Angular Discretisation of the Steady-State Mono-Energetic Linear Boltzmann Equation with Fokker-Planck Approximation for Proton Transport using a Control Volume Method

Bachelor Thesis

Koen de Jong, 1517953
Applied Physics
NERA, TU Delft

07-08-2017

Abstract

This bachelor thesis implements angular discretisation for the steady-state mono-energetic Linear Boltzmann Transport Equation (LBTE) in the Fokker-Planck limit into the in-house program Phantom of the NERA research group, TU Delft. The implementation uses a control volume method (CVM) scheme, whereas the spatial discretisation (which is already implemented in Phantom) of the LBTE uses a Discontinuous Galerkin finite element method (DCGM).

In chapter 2 the CVM discretisation scheme is explained step by step. Furthermore flux error calculations are introduced by use of the method for manufactured solutions. Spherical harmonics are used for the angular dependent part of the flux as these are the eigenvalues of the Laplacian that appears in the Fokker-Planck limit.

In chapter 3 the results of Nauta (2016) are reproduced, to assure correct implementation. Next streaming is enabled. Uniform angular refinement and a quadratic angular depended flux are used to determine the effectiveness of the implementation. This shows angular convergence around first order is achieved for early refinement steps. Higher refinement however results in lower convergence orders. Convergence of the scalar error start out promising, resulting in convergence order $p = 2.37$ and $p = 2.06$ for the Spherical angular mesh refinement method between zeroth and third level and for the Octagon angular mesh refinement method between zeroth and fourth level refinement respectively, after which the convergence breaks down. A cubic angular depended flux shows even more deviation from ideal angular convergence. Scalar error convergence between zeroth and third level refinement is $p = 2.04$ and $p = 1.98$ for the Spherical and Octagon method respectively, after which the convergence breaks down.

Comparing the CVM to results obtained with a DCGM by Hennink and Lathouwers (2017) shows that the CVM is much less effective. For The angular flux error of fourth level refinement achieved with the CVM can be reached with first level refinement using a DCGM. The scalar errors of the CVM with third level refinement are surpassed with first level refinement using a DCGM. These findings merit the use of a DCGM over the CVM for angular discretisation.

This project is conducted at the NERA research group of the TU Delft and supervised by Dr. ir. D. Lathouwers.

Contents

Chapter 1

Introduction

Radiotherapy is one of the main options for cancer treatment. High energy photons are beamed at the tumour in order to ionise the diseased tissue. The incoming radiation can directly ionise the atoms that make up the DNA (direct ionisation) or the photons ionise water molecules that form free radicals, which subsequently damage the DNA (indirect ionisation). This damage can cause the cell to immidiatly die (mitotic catastrophe). As the cancer cells are less able to repair themselves compared to healthy cells and reproduce fast, non-lethal damage is passed on and can cumulatively result in cell death too. To prevent the healthy tissue from undergoing too much damage, the tissue is radiated from different angles causing only the diseased tissue to be exposed to the full dose.

The sensitivity of the tumour towards the radiation depends on the particular type of cancer. Highly sensitive types such as lymphomas and leukimias are highly receptive to radiation and might require doses of 20 to 40 Gy. Types with limited susceptibility as epithelial cancer may require significantly larger doses of 60 to 80 Gy. Radioresitant tumours, such as melanoma, would require unsafe doses. Besides the sensitivity towards radiation therapy, the location and spread of the cancer are of critical importance to the viability of radiation therapy. When the cancer is too spread out (as often the case with leukimias) or they are located next to critical tissue that should not be damaged, the effective use of radiotherapy is very limited at best.

1.1 Dose deposition & Proton Therapy

The damaging of DNA through ionisation resulting into mitotic catastrophe can also be achieved by the radiation of charged particles. Different types of radiation treatments each have their unique dose deposition profile within the tissue. See figure 1.1. This makes those radiation treatments more or less suitable as treatment depending on the particular case. Electron therapy can be used for tumours at the surface, as the dose deposition is very high on entry (entrance dose) and sharply drops off to zero after a certain depth, leaving tissue behind the tumour undamaged. Radiotherapy has a surface sparing effect, as the entrance dose is low. Deeper inside the tissue the dose reaches a maximum after which it will gradually decrease. This leaves an exit dose which damages the tissue located behind the tumour. Heavy charged particles (e.g. protons and carbon ions) show a different profile as well.

As the heavy charged particles move through tissue they lose energy described by the stopping

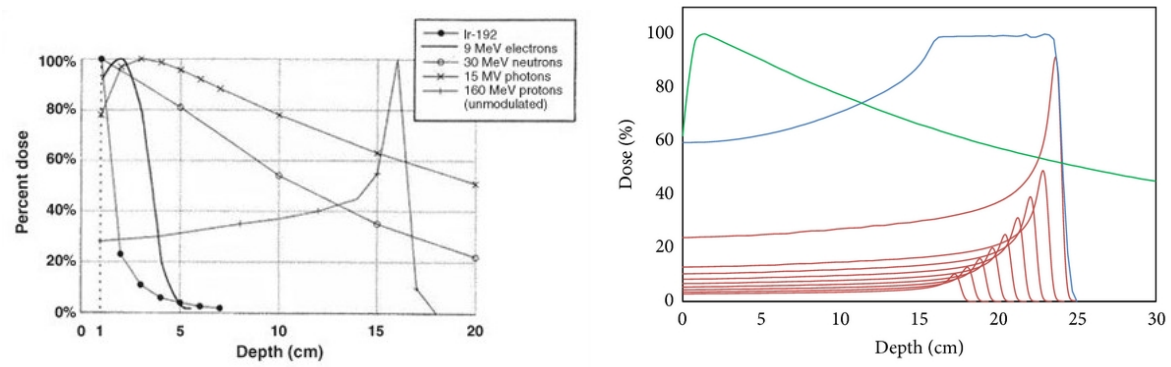


Figure 1.1: Left: The dose deposition patterns plotted for different methods of radiation. For 160MeV protons the characteristic Bragg-peak is cleary visible. Ir-192 is the deposition pattern when using Iridium-192 seeds. The profiles for 9MeV electrons, 30MeV neutrons and 15MV photons are shown as well. This image is taken from Czito et al. (2006). **Right**: Proton depositions of different energies and intensities are shown in red, combined these create a Spread-Out Bragg Peak shown in blue. The green line shows the deposition profile of photons. This image is taken from Grant and Chang (2014)

power, this energy lost (the deposited dose) results in a lower velocity of the particles. As the loss in energy is inversely proportional to the velocity squared, just before the particles coming to a halt the energy deposition will peak. This peak called the Bragg peak. After the Bragg peak the dose is zero for protons and almost zero for heavier charged particles such as carbon ions. Combining proton radiation of different energies allows for coverage of the complete tumour with a spread-out Bragg peak (SOBP), this is shown in figure 1.1. Another advantage of heavy charged particles is that the high mass of the particles causes the beam to be less divergent from the incident axis.

The dose deposition profile of proton radiation makes proton therapy advantageous when the tumour is located next to critical tissue. The lack of an exit dose allows for the critical tissue to be undamaged, while the tumour is being radiated. Another suitable use for proton therapy is when the tumour requires dose escalation as it responds better to higher doses. With the localised character of the Bragg peak, proton therapy allows for these higher doses. This leads to the use of proton therapy for tumours near the eye, near the spinal cord as well as for prostate cancer amongst other applications.

1.2 Dose Prediction

The prediction of the dose deposition is of natural importance for a treatment plan, as the tumour needs to be destroyed while damage to the surrounding tissue needs to be limited as much as can be. For proton therapy this is even more so the case, as the application specializes in tumours next to critical tissue and localised high dose deposition.

Uniform intensity algorithms to predict the dose deposition are simple and fast, but too crude to use in treatment plans. Pencil beam approximations provide a much better alternative, calculation is fast and the results are reasonably accurate. This technique, derived first by

Rossi and Greisen (1941), considers the incident particle beam to be a weighted conglomerate of many much smaller pencil beams. The divergence of these pencil beams is modelled on experimental data. The pencil-beam approximation is however difficult to implement for inhomogeneous media and more general geometries, such as with head and neck treatments (Jiang et al., 2007). Discrepancies arising from dose prediction using pencil-beam approximation might in certain cases have clinical implications (Paganetti et al., 2008).

The most precise method of dose prediction is the use of Monte Carlo simulation where the path and interactions of a single particle are tracked many times over. The aggregate of these paths gives a very precise prediction of the dose distribution. The high precision allows for Monte Carlo simulations to act as a benchmark for other prediction models. High computational costs however make these simulations of limited use for individual treatment plans. It is expected that with the rise of computing power these simulations will become much more used in treatment planning. Paganetti et al. (2008).

Another direction that has been explored to predict the dose distribution is discretely solving the transport equation for protons (Morel, 1981; Uilkema, 2012). In theory solving the equation would achieve a similar level of accuracy as with the Monte Carlo method and has several other benefits. It is much faster in calculation than the Monte Carlo method, the dose distribution for the entire region is calculated and this method allows for sensitivity analysis used to calculate uncertainties, for example caused by movements of the patient or as a result of chemical decomposition inside the body (Uilkema, 2012).

To discretely solve in the (in this case) linear Boltzmann transport equation, different methods have been applied. The S_N method uses a finite set of discrete-ordinates to approximate the possible directions of travel and subsequently discretises the equation in all variables. The method was developed by nuclear reactor physicists to calculate de neutron distribution in a reactor core. The large number of small scatter interactions in proton transport, lead to a nearly singular differential cross section. Making expansion of the differential cross section in Legendre polynomials as is traditional with neutron distributions unfeasible Uilkema (2012). The Fokker-Planck approximation is used for the small angle scatter interactions instead, this introduces unfortunately an error claimed to be greater than the error due to the pencil beam approximation (Borgers and Larsen, 1996; Hennink, 2015). However, the great versatility of a discrete solution makes this approach still very much desirable.

As the S_N method is not wholly suitable for charged particle transport a fully adaptable deterministic numerical approach has been described by Hennink (2015). This approach is based on the discontinuous Galerkin method (DCGM) for both the spatial and angular parts. The inhouse software Phantom of NERA (TU Delft), mainly developed by Danny Lathouwers, included this DG discretisation in space of the transport equation at the start of this project. The DG discretisation in the angular part was implemented during this project. The goal of this bachelor thesis is to implement a more simplistic and computational lighter angular discretisation method for the transport equation in Phantom. The method being a control volume method used for the Fokker-Planck equation by Nauta (2016).

Chapter 2

Theoretical Overview

2.1 Linear Boltzmann Transport Equation and Fokker-Planck

Proton transport can be described by the steady-state Linear Boltzmann Transport Equation (LBTE). For the validity of the underlying assumptions see Uilkema (2012); Duderstadt and Hamilton (1976). In this particular work, the energy dependency is neglected. This dependency is not implemented in the existing transport code and doing so is not the topic of this research. Leaving out the energy dependencies, the equation for proton transport is the following steady-state mono-energetic LBTE (Hennink, 2015):

$$\boldsymbol{\Omega}\left(\frac{\partial}{\partial \boldsymbol{r}}\right)\varphi(\boldsymbol{r},\boldsymbol{\Omega})+\Sigma_a(\boldsymbol{r})\varphi(\boldsymbol{r},\boldsymbol{\Omega})-Q\varphi(\boldsymbol{r},\boldsymbol{\Omega})=S(\boldsymbol{r},\boldsymbol{\Omega}) \tag{2.1}$$

The equation describes the angular proton flux (φ), which can be considered the particle density moving in a certain direction ($\boldsymbol{\Omega}$) on a specific location ($\boldsymbol{r}$). $S(\boldsymbol{r},\boldsymbol{\Omega})$ is the angular source term. $\Sigma_a(\boldsymbol{r})$ is the macroscopic absorption cross-section. The macroscopic scatter cross-section is represented by Q. This cross section describes the probability density function of small deviations mostly due to coulomb scattering with electrons and elastic scattering with nuclei resulting in a change of direction of the incident protons.

2.1.1 Focker-Planck

The Fokker-Plack (FP) method can be used to approximate the macroscopic cross-section term Q. For the derivation see Uilkema (2012); Morel (1981). As only steady-sate is considered, the Fokker-Planck equation will lose its stochastic character and becomes an ordinary differential equation. The approximation leads to:

$$Q\,\varphi(\boldsymbol{r},\boldsymbol{\Omega})\approx\frac{\sigma_{tr}}{2}\nabla^2_{\boldsymbol{\Omega}}\,\varphi(\boldsymbol{r},\boldsymbol{\Omega}) \tag{2.2}$$

Here σ_{tr} is the macroscopic transport cross-section. This cross-section can be measured experimentally or predicted from theoretical models. σ_{tr} is energy depended, which is as stated earlier neglected in this work. $\nabla^2_{\boldsymbol{\Omega}}$ is the Laplacian in the angular coordinates of $\boldsymbol{\Omega}$. The direction is represented on a unity sphere by the spherical coordinates θ and χ, the azimuthal and zenith angle respectively. The Laplacian and Divergence operator in these

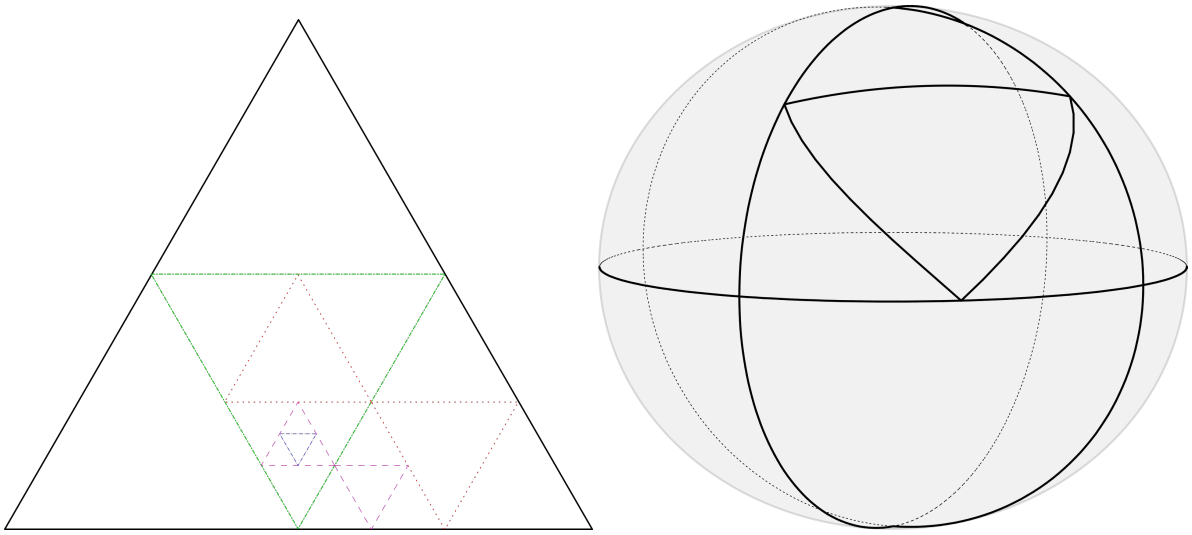


Figure 2.1: **left**: Local refinement of the angular mesh on a octahedron face. The angular elements are up to level 4. **right**: Angular mesh on sphere, one octant has a refinement of level 1

coordinates are:

$$\nabla^2_{\mathbf{\Omega}} f = \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}(\sin\theta\frac{\partial f}{\partial\theta}) + \frac{1}{\sin^2\theta}\frac{\partial^2 f}{\partial\chi^2} \qquad \nabla_{\mathbf{\Omega}} \cdot A = \frac{1}{\sin\theta}\frac{\partial A_\theta \sin\theta}{\partial\theta} + \frac{1}{\sin\theta}\frac{\partial A_\chi}{\partial\chi}$$

The Fokker-Planck approximation describes a diffusive process of the angular flux and is hence called the diffusion term. Whereas $\mathbf{\Omega}\left(\frac{\partial}{\partial \boldsymbol{r}}\right)\varphi(\boldsymbol{r},\mathbf{\Omega})$ is called the streaming part. The LBTE becomes:

$$\mathbf{\Omega}\left(\frac{\partial}{\partial \boldsymbol{r}}\right)\varphi(\boldsymbol{r},\mathbf{\Omega}) + \Sigma_a S\varphi(\boldsymbol{r},\mathbf{\Omega}) - \frac{\sigma_{tr}}{2}\nabla^2_{\mathbf{\Omega}}\varphi(\boldsymbol{r},\mathbf{\Omega}) = S(\boldsymbol{r},\mathbf{\Omega}) \tag{2.3}$$

2.2 Angular Elements

As mentioned earlier, the direction of flux can be described by the two angular coordinates, the azimuthal and zenith angle. An angular mesh is created on a octahedron. Each of the eight triangle faces is an angular element or so-called patch. These elements can be refined by dividing the triangle up in to four triangles. This is done by connecting the three bisection-points of the original patch. This way first level (refined) angular elements are created (The original element being zeroth level of refinement). This process can be repeated to create higher level elements. When all angular elements have an equal level, there is said to be uniform refinement. If this is not the case, there is local refinement. See figure 2.1.

The elements on the octahedron are projected on the surface of the unity sphere to create the final angular mesh. There are different ways to construct the angular mesh, see Hennink (2015). One way would be to take the halfway points of the sides of an octant projected on the unity sphere and connect those using the shortest path (Great-circle navigation). Phantom has implemented two ways to calculate the angular mesh. The 'octagon method' calculates the mesh refinements on the octagon before projecting it on a unity sphere. The 'spherical method' immediately calculates the angular grid refinement on the unity sphere.

2.3 (Angular) Discretisation of FP-LBTE

The goal is to discretise mono-energetic steady-state LBTE with the FP approximation. The discretisation in space has already been implemented in Phantom using the Discontinious Galerkin Finite Element method. For the angular discretatsion a more simplistic method will be used, a control volume method. This particular method was used in the bachelor thesis work of Nauta (2016). For a specific spatial element k and angular element l, using the Discontinuous polynomial approximation, the angular proton flux can be considered as follows. Lathouwers (2014):

$$\varphi^{k,l}(\boldsymbol{r},\boldsymbol{\Omega}) \approx \sum_{j=1}^{N}\sum_{i=1}^{M} \varphi_{i,j}^{k,l} g_{i,j}^{k,l}(\boldsymbol{r}) h_{i,j}^{k,l}(\boldsymbol{\Omega}) \tag{2.4}$$

Here $h(\boldsymbol{\Omega})$ and $g(\boldsymbol{r})$ are the angular and spatial basis functions. The control volume method, used for the angular discretisation, has only one angular basis function $h(\boldsymbol{\Omega}) = 1$ as the proton flux is taken to be constant for each angular element. This simplifies the discontinuous polynomial approximation to:

$$\varphi(\boldsymbol{r},\boldsymbol{\Omega}) \approx \sum_{i=1}^{M} \varphi_i^{k,l} g_i^{k,l}(\boldsymbol{r}) \tag{2.5}$$

To find the expansion coefficients using the Discontinuous Galerkin Method (DCGM), the original equation is multiplied by a test function (in the case of DCGM, the test function is the basis function). The integrate over the spatial and angular element of the subsequent equation must hold for the numerical solution. For the Diffusion part of the equation this leads to the following integral for spatial element k and angular element l:

$$\int_k \int_l -\frac{\sigma_{tr}}{2}\nabla_{\boldsymbol{\Omega}}^2 \varphi(\boldsymbol{r},\boldsymbol{\Omega}) g_i^{k,l}(\boldsymbol{r}) \mathrm{d}\boldsymbol{\Omega}\mathrm{d}\boldsymbol{r} \approx \sum_{v=1}^{M} \int_l -\frac{\sigma_{tr}}{2}\nabla_{\boldsymbol{\Omega}}^2 \varphi_v^{k,l}(\boldsymbol{\Omega})\,\mathrm{d}\boldsymbol{\Omega} \int_k g_v^{k,l}(\boldsymbol{r}) g_i^{k,l}(\boldsymbol{r})\mathrm{d}\boldsymbol{r} \qquad i = 1, ..., M \tag{2.6}$$

Here the polynomial approximation of equation (2.5) is used, but only for the spatial dependencies, thus $\varphi_i^{k,l} = \varphi_i^{k,l}(\boldsymbol{\Omega})$. The equation is divided in an angular en spatial part. Concentrating on the integral over the angular element (patch) and using Green's theorem ($\iint_S \nabla \boldsymbol{F}\,\mathrm{d}a = \oint_C \boldsymbol{F}\cdot\hat{\boldsymbol{n}}\,\mathrm{d}l$). The integral over the (angular) patch can be rewritten as:

$$\int_l -\frac{\sigma_{tr}}{2}\nabla_{\boldsymbol{\Omega}}^2 \varphi_v^{k,l}(\boldsymbol{\Omega})\,\mathrm{d}\boldsymbol{\Omega} = -\frac{\sigma_{tr}}{2}\int_l \nabla_{\boldsymbol{\Omega}}(\nabla_{\boldsymbol{\Omega}}\varphi_v^{k,l}(\boldsymbol{\Omega}))\,\mathrm{d}\boldsymbol{\Omega} = -\frac{\sigma_{tr}}{2}\oint_l \nabla_{\boldsymbol{\Omega}}\varphi_v^{k,l}(\boldsymbol{\Omega})\cdot\hat{\boldsymbol{n}}\,\mathrm{d}l \tag{2.7}$$

2.3.1 Divergencce of flux

The Control Volume Method (CVM) assumes a constant angular flux on each patch, the value of the angular flux is taken to be on the centre of mass of the patch. In order to calculate the centre of mass (CoM) the vectors from the origin of the unity sphere to the three vertices of the patch are summed and subsequently normalised. This constructed vector is taken to point from the origin to the centre of mass.

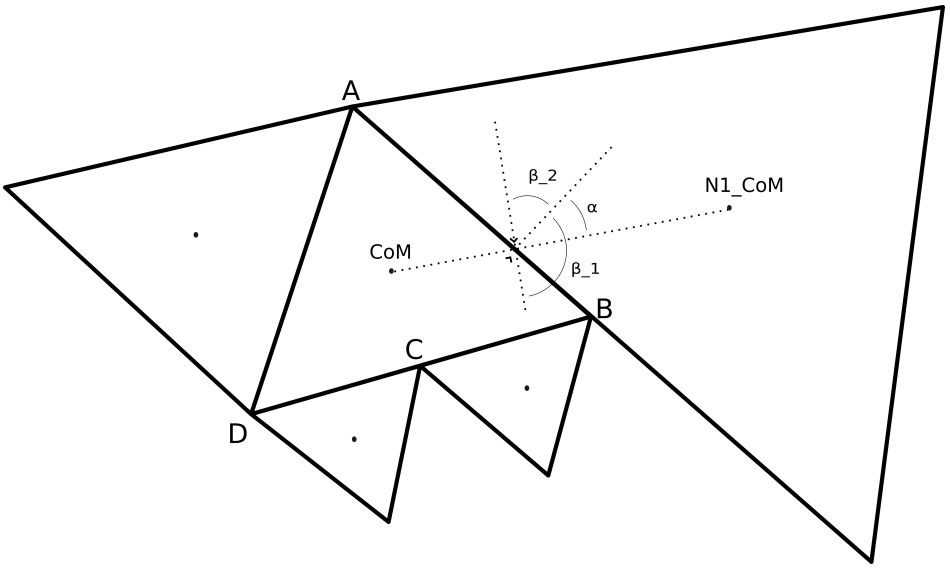


Figure 2.2: A patch with four subfaces. α is the angle between the normal of the (sub)face and the difference vector of the CoM and it's (sub)face-neighbours CoM.

In figure 2.2 it is shown that while every patch has three faces, due to local refinement it can have more than three face-neighbours and therefore has sub-faces that need to be taken into account. The lines BC and CD are the sub-faces of BD.

The divergence of the flux in equation (2.7) is approximated for each (sub-)face in the following manner. The difference in flux between the patch and it's particular face-neighbour is taken and divided by the distance between their respective CoM (Δs). The gradient direction is taken to be the normalised difference vector between the two centres of mass, $\hat{\nabla}_{\mathbf{\Omega}}$. The dot-product between gradient direction $\hat{\nabla}_{\mathbf{\Omega}}$ and the normal vector (pointing outwards) of the face between the two patches $\hat{n}$ gives $\hat{\nabla}_{\mathbf{\Omega}} \cdot \hat{\boldsymbol{n}} = \cos\alpha$. Where α is the angle between the two vectors. See figure 2.2

The normalised cross-product between the two vectors from the origin to the edges of the (sub)face creates a vector $\hat{\boldsymbol{f}}_n$ parallel to the normal of the (sub)face. The cross-product of the two CoM-vectors is taken to construct a vector $\hat{\boldsymbol{a}}_n$ perpendicular to the gradient direction. The angle between these normalised vectors is calculated using the cross-product $|\hat{\boldsymbol{f}}_n \times \hat{\boldsymbol{a}}_n| = \sin\beta$.

As $\hat{\boldsymbol{a}}_n$ is perpendicular to the gradient direction and $\hat{\boldsymbol{f}}_n$ is parallel to normal of the face, it follows that $\alpha = \frac{\pi}{2} - \beta_2$ or $\alpha = \beta_1 - \frac{\pi}{2}$ depending on the direction of $\hat{\boldsymbol{a}}$. In either case $\hat{\nabla}_{\mathbf{\Omega}} \cdot \hat{n} = |\hat{\boldsymbol{f}}_n \times \hat{\boldsymbol{a}}_n|$.

Summing over all the (sub)faces this leads to the line integral over the patch being discretised

as:

$$\frac{\sigma_{tr}}{2}\oint_l \nabla_{\mathbf{\Omega}}\,\varphi\cdot\hat{\boldsymbol{n}}\,\mathrm{d}l \approx \sum_{i}^{\#(sub)faces} \frac{\sigma_{tr}}{2}\frac{\varphi_{NCoM,i}-\varphi_{CoM}}{\Delta s_i}\hat{\nabla}_{\mathbf{\Omega},i}\cdot\hat{n}_i\,\Delta l_i$$
$$= \sum_{i}^{\#(sub)faces} \frac{\sigma_{tr}}{2}(\varphi_{NCoM,i}-\varphi_{CoM})|\hat{\boldsymbol{f}}_{n,i}\times\hat{\boldsymbol{a}}_{n,i}|\,\frac{\partial l_i}{\partial s_i} \tag{2.8}$$

Here Δl_i is the length of the specific (sub-)face. φ_{CoM} is the angular flux value taking at CoM of the patch, where $\varphi_{NCoM,i}$ is the flux value of the specific neighbour. The term $\frac{\Delta l_i}{\Delta s_i}$ can be calculated by dividing angle between the vectors from the origin to the two edges of the (sub-)face by angle between the two CoM-vectors. This way there is no need to take the curvature of the distances in account, which simplifies the calculation.

The discrete expression from equation (2.8) combined with equation (2.6) deals with the the discretisation of diffusion term. Note however that the subscripts i from both equations refer to different variables, the (sub-)face number in equation (2.8) and the particular spatial test function in equation (2.6).

2.3.2 Non-diffusion terms

Having discretised the angular dependency of the diffusive part, the other terms of equation (2.3) are still left. The sources term is depended on both $\boldsymbol{r}$ and $\mathbf{\Omega}$. The integral over spatial element k and angular element l of the source multiplied by the test-function can be easily evaluated with the local quadrature sets on both the spatial element and angular patch.

$$\int_k\int_l S(\boldsymbol{r},\mathbf{\Omega})g_i(\boldsymbol{r})\mathrm{d}\mathbf{\Omega}\mathrm{d}\boldsymbol{r} \approx \sum_{p}^{\#lqp}\sum_{o}^{\#kqp} g_{i,o}q_{o,p}S_{o,p} \qquad i=1,...,M \tag{2.9}$$

Here the source term is calculated for each spatial and angular quadrature point and multiplied by its total quadrature weight $q_{o,p}$. Summation over all the quadrature points leads to the discrete approximation of the integral of the spatial element and patch. The adsorption term can be discretised in a similar fashion, but is a bit more complicated as the flux appears in this term:

$$\Sigma_a\int_k\int_l \varphi g_i^{k,l}(\boldsymbol{r})\mathrm{d}\boldsymbol{r}\mathrm{d}\mathbf{\Omega} \approx \Sigma_a\sum_{v=1}^{M}\varphi_v^{k,l}\int_k g_v^{k,l}g_i^{k,l}\mathrm{d}\boldsymbol{r}\int_l \mathrm{d}\mathbf{\Omega} \qquad i=1,...,M \tag{2.10}$$

For this project the macroscopic adsorption coefficient Σ_a is taken to be constant. In cases where it has spatial dependencies, taking the coefficient constant on each spatial element is an option or it can be evaluated with the spatial quadrature sets. The integral over the spatial test functions can be calculated and appears just as it did with the discretisation of the diffusion term in equation (2.6).

Discretisation of the streaming term is more complex. As this part of the function is already implemented in Phantom code using DCGM and as it is considered to be out of the scope of this project the report will not further expand on it. Combining the implemented streaming term with the equations (2.6) (2.8) (2.9) (2.10) leaves a system of M equations that can be solved to obtain the flux coefficients $\varphi_v^{k,l}$.

2.4 Method of manufactured solutions

In order to test the CVM implementation, the method of manufactured solutions is being used. By choosing the source term S in such a way that the flux can be analytically calculated, a comparison can be made between the analytical and numerical solutions. The 'analytical' calculation is done at all the local quadrature points of the spatial and angular elements.

A constant solution would be a trivial solution. When taking the angular flux to be a product of an angular depended and a spatial depended part ($\varphi(\boldsymbol{r}, \boldsymbol{\Omega}) = R(\boldsymbol{r})\Phi(\boldsymbol{\Omega})$), more interesting solutions can be constructed. Taking a look at only the angular depended part first. The streaming term disappears from the equation, as there is no space dependency.

The eigenfunctions are the simplest non-trivial analytic solutions. Analogue to the Schrödinger equation, the eigenfunctions of the Laplacian are the spherical harmonics:

$$\Phi(\boldsymbol{\Omega}) = Y_{m,l}(\boldsymbol{\Omega}) \tag{2.11}$$

$$S(\boldsymbol{\Omega}) = (\frac{\sigma_{tr}}{2} l(l+1) + \Sigma_a)\Phi(\boldsymbol{\Omega}) \tag{2.12}$$

Here the function $Y_{m,l}(\boldsymbol{\Omega})$ is a spherical harmonic, any linear combination of the spherical harmonics would of course be a solution as well. Substituting equation (2.11) into equation (2.3) results in the angular depended source term given by (2.12). When the spatial part is considered to be constant ($R(\boldsymbol{r}) = 1$), spatial streaming is non-existing. This is used to compare the implementation to the work of Nauta (2016).

If streaming is taken into account and thus the spatial part of the flux is non-trivial, more useful assessments can be made about the effectiveness of the implementation in Phantom. The spatial part is taken as follows:

$$R(\boldsymbol{r}) = x(1-x)y(1-y)z(1-z) \tag{2.13}$$

This is for a 3D spatial domain. In the case of 2D spatial dependency the z term disappears from the equation (2.13) and as such becomes $R(\boldsymbol{r}) = (x - x^2)(y - y^2)$. The contribution of the streaming term to the source term is:

$$S_{streaming} = (\Omega_x \frac{\partial R(\boldsymbol{r})}{\partial x} + \Omega_y \frac{\partial R(\boldsymbol{r})}{\partial y} + \Omega_z \frac{\partial R(\boldsymbol{r})}{\partial z})\Phi(\boldsymbol{\Omega}) \tag{2.14}$$

Here the angular coordinates are expressed in there Cartesian form to allow for an easier treatment of the streaming term and it's source contribution. The total source can now be constructed for which the analytical flux is known:

$$\varphi(\boldsymbol{r}, \boldsymbol{\Omega}) = R(\boldsymbol{r})\Phi(\boldsymbol{\Omega}) = Y_{m,l}(\boldsymbol{\Omega})(x - x^2)(y - y^2)(z - z^2) \tag{2.15}$$

$$S(\boldsymbol{r}, \boldsymbol{\Omega}) = (\frac{\sigma_{tr}}{2} l(l+1) + \Sigma_a)\Phi(\boldsymbol{\Omega})R(\boldsymbol{r}) + (\Omega_x \frac{\partial R(\boldsymbol{r})}{\partial x} + \Omega_y \frac{\partial R(\boldsymbol{r})}{\partial y} + \Omega_z \frac{\partial R(\boldsymbol{r})}{\partial z})\Phi(\boldsymbol{\Omega}) \tag{2.16}$$

Using this constructed source term the flux is computed. The (relative) error of computation can now be determined as the analytical flux is known. By increasing the number of angular elements the relative error should decrease. The relation between the relative error and the

angular patch size (error convergence) gives information about the effectiveness of the angular discretisation and computation. The relative angular flux error is calculated as follows:

$$\epsilon_{flux} = \frac{\int\limits_{\boldsymbol{r}} \int\limits_{\boldsymbol{\Omega}} (\varphi^{num} - \varphi)^2}{\int\limits_{\boldsymbol{r}} \int\limits_{\boldsymbol{\Omega}} \varphi^2} \tag{2.17}$$

The integrals are evaluated by summing the local integrals over all the spatial elements and patches. The local integrals are calculated with their particular quadrature sets for each element. Furthermore φ^{num} is the numerically calculated flux value and φ is the 'analytical' flux.

When the angular mesh is refined by one level, the number of patches is increased to four times the former amount. The number of patches is hence given by: $N_i = 8 * 4^i$, here i is the level of refinement. The characteristic size of the patch is $h_i = \sqrt{\frac{4\pi}{N_i}}$. The relative error is proportional to a power function of the characteristic length: $\epsilon_i \propto h_i^p$, here p is the order of convergence. For the control volume method being used the theoretical error is of order $p = 1$. The order of convergence that is achieved with calculation is given by:

$$p = \frac{\ln\left(\frac{\epsilon_{i+k}}{\epsilon_i}\right)}{\ln\sqrt{\frac{1}{4^k}}} \tag{2.18}$$

So between each refinment step the order of convergence is $p = \ln\left(\frac{\epsilon_{i+1}}{\epsilon_i}\right) / \ln(0.5)$.

In dose predictions the total error in each location is much more important than the angular error. This error can be calculated with the scalar flux which is given by $\varphi_{scal} = \int_{\boldsymbol{\Omega}} \varphi(\boldsymbol{r}, \boldsymbol{\Omega}) d\boldsymbol{\Omega}$. The relative scalar flux error (ϵ_{scal}) can be calculated in a similar manner as the angular flux error:

$$\epsilon_{scal} = \frac{\int\limits_{\boldsymbol{r}} (\varphi^{num}_{scal} - \varphi_{scal})^2}{\int\limits_{\boldsymbol{r}} \varphi^2_{scal}} \tag{2.19}$$

A third method of error calculation is used. The numerically calculated flux value, which is taken to be on the CoM of each patch, is compared to the actual flux value in the CoM. This gives the following CoM flux error:

$$\epsilon_{CoM} = \frac{\int\limits_{\boldsymbol{r}} \sum\limits_{l}^{\#patches} (\varphi^l_{num} - \varphi^l(\boldsymbol{r}, \boldsymbol{\Omega}_{CoM}))^2}{\int\limits_{\boldsymbol{r}} \sum\limits_{l}^{\#patches} \varphi^l(\boldsymbol{r}, \boldsymbol{\Omega}_{CoM}))^2} \tag{2.20}$$

Chapter 3

Results

3.1 Uniform refinement, no streaming

As this work is a continuation of the work of Nauta (2016). First the results obtained in that research will be reproduced. To accomplish that, the streaming part of the transport equation is turned off and the flux is considered to be only angular dependent. The following combination of spherical harmonics is as used as flux for the method of manufactured solutions.

$$\varphi(\mathbf{\Omega}) = 2\sqrt{\pi}(0.7Y_{0,0}(\mathbf{\Omega}) - 0.5Y_{1,1}(\mathbf{\Omega}) + 0.1Y_{2,0}(\mathbf{\Omega}) + 0.3Y_{3,2}(\mathbf{\Omega})) \tag{3.1}$$

Furthermore, $\Sigma_a = 10\,\mathrm{cm}^{-1}$ and $\sigma_{tr} = 25\,\mathrm{cm}^{-1}$. The program Phantom has the feature to use the coordinates on the octagon and use these for the creation of the refined angular mesh before projecting this mesh on the sphere. Another option is to immediately refine the mesh on the unity sphere. For both ways of mesh creation the angular discretisation is tested. The results using the flux from equation (3.1) are shown in table 3.1.

With the spherical method the results of Nauta (2016) can not be reproduced. The convergence rate lacks behind on the theoretical expectation. The octagon method gives better convergence compared to the spherical method and it matches the results produced in Nauta (2016) exactly. For the first two refinement levels (0 and 1) there is no difference between

Table 3.1: Angular flux error for φ given by equation (3.1), $\Sigma_a = 10\,\mathrm{cm}^{-1}$, $\sigma_{tr} = 25\,\mathrm{cm}^{-1}$, uniform refinement.

	Spherical Method			Octagon Method		
N	Angular flux error	Steps	Order of conv (p)	Angular flux error	Steps	Order of conv (p)
8	4.459E-01	6		4.459E-01	6	
32	3.052E-01	12	0.547	3.052E-01	12	0.547
128	1.496E-01	20	1.03	1.442E-01	20	1.08
512	7.791E-02	38	0.941	7.420E-02	37	0.958
2048	4.408E-02	76	0.822	3.857E-02	74	0.944
8192	3.034E-02	148	0.539	2.178E-02	139	0.825
32768	2.593E-02	285	0.227	1.489E-02	278	0.548

the spherical and octagon method, for higher refinement however the methods produce different results. Even though the octagon method shows better convergence, when the angular refinement goes up the convergence becomes less strong. The octagon method does not only show a better rate of convergence, it takes less steps in calculation as well and is therefore computationally more effective in this case.

3.2 Uniform refinement, with streaming

Taking the effects of streaming into account, error calculations are made to see the effectiveness of the implementation of the CVM in Phantom. A four by four spatial grid is used. The angular dependencies are calculated for half a sphere as this is possible due to symmetry and saves computing costs. The spatial dependent flux is taken as described by the 2D equivalent of equation (2.13). The angular dependent part is given by: $\Phi(\mathbf{\Omega}) = C_0 + C_1\Omega_x + C_2\Omega_y + C_3(3\Omega_x^2 - 1)$ with $C_{0:3} = 5; 1; 2; 1$, a linear combination of different spherical harmonics. Here $\Sigma_a = 10\,\mathrm{cm}^{-1}$ and $\sigma_{tr} = 2\,\mathrm{cm}^{-1}$. Fourth order spatial basis functions are used As the spatial dependency of the flux is of second order, this should eliminate spatial error dominance. The results are shown in Table 3.2 for the spherical method and Table 3.3 for the octagon method. Results for the no-streaming case, while using the same manufactured solution apart from having no spatial dependencies, are presented in Appendix A.1

Table 3.2: Flux error using the Spherical method, $\Sigma_a = 10\,\mathrm{cm}^{-1}$,$\sigma_{tr} = 2\,\mathrm{cm}^{-1}$, $C_{0:3} = 5; 1; 2; 1$, uniform refinement

N	Steps	Angular flux error	Order of conv. (p)	Scalar flux error	Order of conv. (p)	CoM flux error	Order of conv. (p)
8	4	2.112E-01		9.639E-03		3.086E-02	
32	6	1.058E-01	0.998	2.952E-03	1.71	1.736E-02	0.830
128	15	5.661E-02	0.902	6.844E-04	2.11	9.746E-03	0.833
512	35	2.955E-02	0.938	6.992E-05	3.29	9.190E-03	0.085
2048	89	1.661E-02	0.831	1.683E-04	-1.27	9.222E-03	-0.005
8192	174	1.121E-02	0.567	2.235E-04	-0.409	9.249E-03	-0.004
32768	336	9.391E-03	0.256	2.431E-04	-0.121	9.258E-03	-0.001

Table 3.3: Flux error using the Octagon method, $\Sigma_a = 10\,\mathrm{cm}^{-1}$,$\sigma_{tr} = 2\,\mathrm{cm}^{-1}$, $C_{0:3} = 5; 1; 2; 1$, uniform refinement

N	Steps	Angular flux error	Order of conv. (p)	Scalar flux error	Order of conv. (p)	CoM flux error	Order of conv. (p)
8	4	2.112E-01		9.639E-03		3.086E-02	
32	6	1.058E-01	0.998	2.952E-03	1.71	1.736E-02	0.830
128	15	5.712E-02	0.889	7.576E-04	1.96	5.814E-03	1.58
512	35	2.912E-02	0.972	1.561E-04	2.28	3.552E-03	0.711
2048	75	1.490E-02	0.966	3.149E-05	2.31	3.324E-03	0.096
8192	151	7.999E-03	0.898	6.381E-05	-1.02	3.323E-03	0.000
32768	312	4.940E-03	0.695	7.642E-05	-0.260	3.334E-03	-0.005

In table 3.2 it can be seen that the order of convergence of the angular flux error starts begins close to the theoretical value of $p = 1$ for early refinement. Up to a refinement of level 3 the order stays within 10% of this theoretical value. After that the convergence starts to break down however and the order decreases for each consecutive step. The scalar error starts out with even better convergence which then even significantly increases up to $p = 3.29$ between refinement levels 2 and 3 . After this level the error starts to increase again, as the error due to spatial discretisation becomes dominant. The CoM error converges quickly to a limit and stays at this limit for higher refinement.
With the octagon method in table 3.3 similar behaviour is seen as with the spherical method. The angular flux order of convergence drops down less quickly. The increase in order for the scalar convergence is more gradual but continues for longer achieving the most precise calculation for refinement level 4. The CoM error calculation again converges to a limit. The error in this limit is 2.8 times smaller as with the spherical method.

In Appendix A.2 similar results are achieved when the angular depended part of the flux is a cubic instead of a quadratic function. While the angular flux error of the spherical method lacks behind compared to the octagon method. The scalar error decrease more rapidly and reaches its minimum for refinement level 3 as opposed to level 4 for the octagon method. The order of angular flux convergence drops much faster in the cubic case as opposed to quadratic angular dependency.

The angular and scalar flux errors are plotted against the characteristic patch size h for both the quadratic and cubic cases in Appendix A.3.

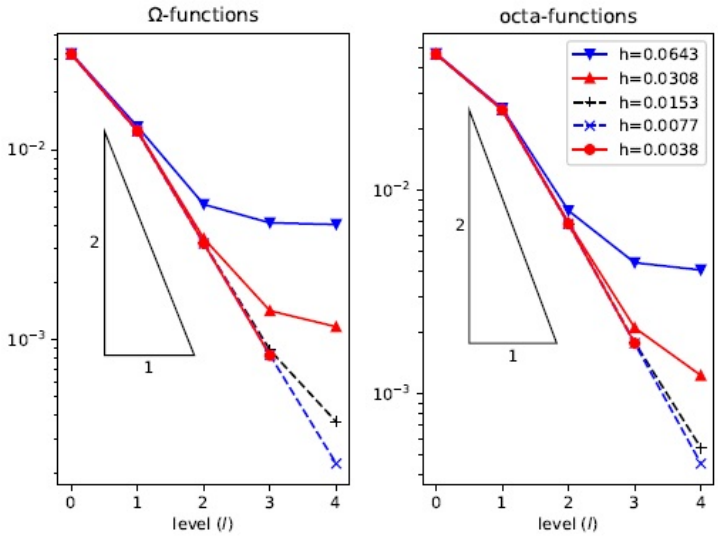
3.3 Comparisson CVM to DCGM

Data from the currently unpublished work of Hennink and Lathouwers (2017) is used to make a comparison and see how the CVM implementation holds up against a full Discontinious Galerkin Method for angular discretisation. The data contains the relative angular- and scalar- flux errors achieved with the DCGM using the same flux as used for tables 3.2 and 3.3. The only differences being the spatial grids being used and $\sigma_{tr} = 1\text{cm}^{-1}$. The spatial grid being a 2D unibox (Domain $(0, 1)^2$) consisting of triangles of approximately equal surface. Different spatial refinement is used: $h = 0.0643$, $h = 0.0308$, $h = 0.0153$, $h = 0.0077$ and $h = 0.0038$. Where the characteristic length is given by $h = 1/\sqrt{(N_{elem})}$, with N_{elem} being the number of spatial elements. First order spatial basis functions are used.

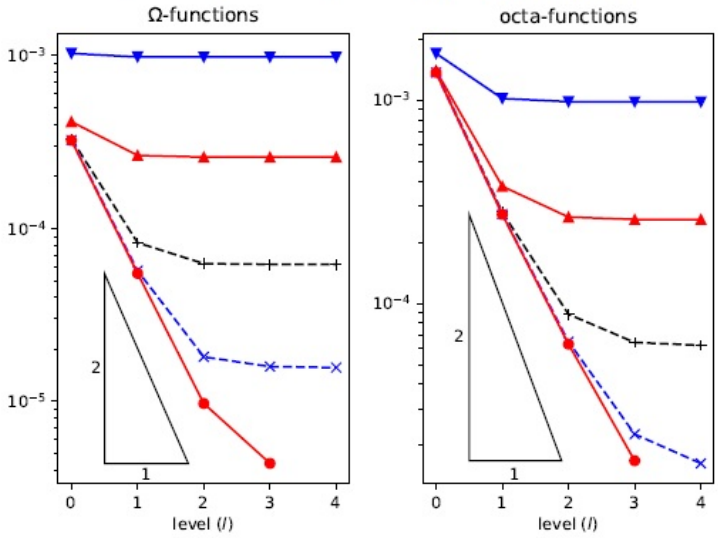
In Hennink and Lathouwers (2017) up to second order convergence is achieved for both the angular and scalar flux errors. This can be seen in Figure 3.1, copied from the work of Hennink and Lathouwers (2017). For larger spatial grid size, the spatial error dominance cuts off the convergence however. As the scalar error is in general of an order smaller than the angular error, it will reach the convergence boundary imposed by the spatial grid error more quickly. The spherical method shows better results than the octagon method does. Reaching the convergence boundary earlier, as the errors are smaller.

Using the CVM scheme under the same conditions, close to first order convergence is achieved for the angular flux and up to second order convergence for the scalar flux. The results are plotted in Figure 3.2. For the angular error no spatial grid error dominance is encountered

is the error starts out much higher than with the DCGM scheme and does not converge fast enough to reach this boundary. The scalar flux error starts out angular dominated and therefore higher as with the DCGM. The lower order of the scalar flux error (compared to the angular flux error) combined with its higher convergence makes that the convergence boundary is reached for the less refined spatial meshes. However, this does not happen as fast as with the DCGM. The differences between the spherical and octagon method are not as clear as in Figure 3.1.



(a) relative L_2-error of angular flux $(|\varphi_h - \varphi|_{T\times\mathbb{S}^2}/|\varphi|_{T\times\mathbb{S}^2})$



(b) relative L_2-error of scalar flux $(|\phi_h - \phi|_T/|\phi|_T)$

Figure 3.1: This figure is taken from Hennink and Lathouwers (2017). Here the convergence of the angular (top) and scalar (bottom) flux errors are shown for increasing angular refinement levels and different spatial mesh size (h). Both spherical (left) and octagon (right) method are plotted

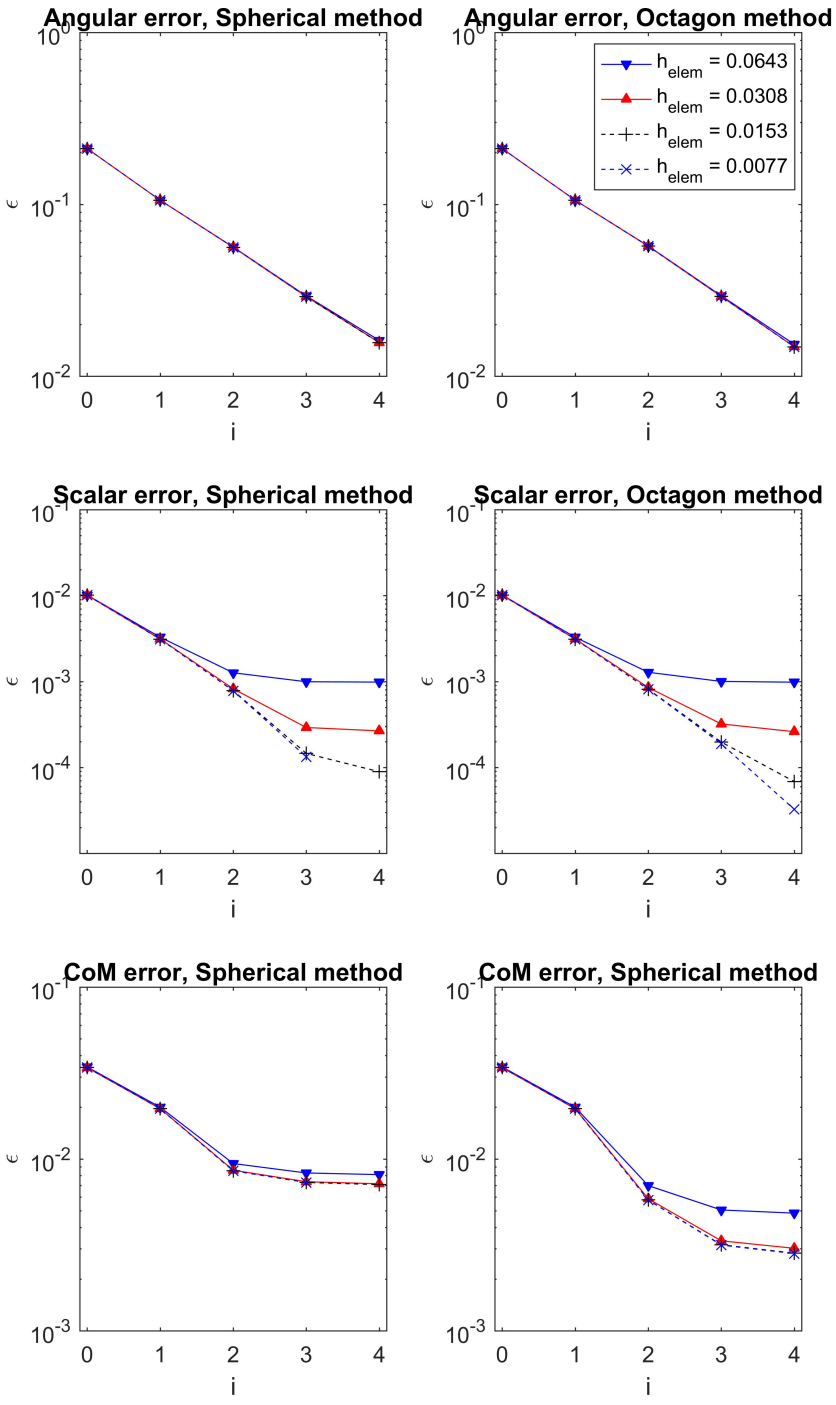


Figure 3.2: Angular (top), scalar (middle) and CoM (bottom) flux errors are shown for increasing angular refinement levels (i) and different spatial mesh size (h_{elem}). Both spherical (left) and octagon (right) method are plotted

Chapter 4

Conclusion and Discussion

The results for uniform refinement of the patches obtained with this CVM scheme by Nauta (2016) can be exactly reproduced using the octagon method of angular grid refinement (table 3.1). The first step of refinement shows unexpectedly poor convergence of order $p = 0.547$. With next three refinement steps the convergence recovers and stays within 10% of the theoretical expected value of $p = 1$. After the initial recovery the order of convergence starts to drift away more and more from the expected and desired $p = 1$. Using the spherical method for angular mesh refinement less desirable results are achieved. From the second refinement step, the calculated error stays behind compared to octagon method. The order of convergence walks away more quickly from the theoretical first order, while taking more steps of calculation to produce the results.

The results obtained when the streaming part of the LBTE is turned on show a convergence of the angular flux error that starts out immediately around the theoretical desired value. The octagon method stays close to this value longer than the spherical method does, but for both methods the convergence starts to break down eventually. The scalar error decreases much more rapidly. With an order of up to $p = 3.29$ for the Spherical method (table 3.2) and an order of $p = 2.37$ from refinement level zero to three, after which the convergence breaks down. The octagon method (table 3.3) shows a maximum convergence order of $p = 2.31$. The total convergence order, before break down after the fourth refinement level, is $p = 2.06$. A break down of scalar error convergence can be explained by the error due to spatial grid size becoming the dominant error. Fourth order spatial basis functions and a flux of second order spatial dependency should have prevented this. It does also not explain the increase in scalar error after break down, as the error is expected to remain relatively constant. The error in the numerical flux value at CoM-coordinates and the analytical value in these coordinates quickly converges to a limit, disallowing the error to reduce further. The angular flux error, starting out larger, appears to converge to a similar limit, although the dataset is not sufficiently extensive to confidently state this is indeed the case. It merits more research here to discover the exact cause of this limit, it being due to spatial grid size errors, a natural result from the approximations made with the CVM discretisation scheme or it having a different cause.

The octagon method achieves smaller errors. The scalar flux error converges slighty better in the spherical case however, but breaks down earlier and at a larger error than the octagon method does. In the no-streaming case of Section A.1 similar angular and CoM as for the streaming case are achieved as for the streaming case.

When the flux is assumed to be cubic in its angular dependencies the angular error convergence shifts away from ideal behaviour much earlier as compared to the quadratic case (Tables A.3 and A.4). The scalar error convergence does again much better with a maximum of $p = 2.19$ and $p = 2.01$ for the Spherical and Octagon method respectively. The total convergence from zeroth to third level refinement being $p = 2.04$ and $p = 1.98$ respectively. Further refinement results in breakdown of scalar convergence.

While scalar error convergence of second order and higher is achieved. The implemented CVM scheme lives up to its potential only in part. The achieved angular error convergence starts out for low level refinement around first order convergence as expected, but as the refinement increases, the order decreases and the error deviates more and more away from first order convergence. This unfortunate behaviour limits the effectiveness of the implementation and warrants the search for different discretisation schemes. Within Phantom a Discontinous Galerkin Method has been implemented for the angular discretisation during the length of this project. Altough more computational costly, the non-ideal behaviour of this CVM scheme and the higher theoretical angular convergence (order $p = 2$) of the DCGM make this a desirable alternative. The results in Section 3.3 show the CVM implementation is indeed inferior to the DCGM. The DCGM achieves second order convergence for both angular and scalar flux errors as is shown in Hennink and Lathouwers (2017). With the CVM, the achieved errors are much larger. As can be seen from Figures 3.1 and 3.2. The angular convergence is close to first order, while the angular error starts out much larger. First level refinement with the DCGM shows a similar angular error as fourth level refinement with the CVM scheme. And while the scalar convergence is promising for the CVM, as up to second order convergence is achieved, the scalar error starts out much higher to begin with. Scalar errors close to 10^{-4} can be achieved with refinement level 3, the DCGM surpasses this at refinement level 1 however. These results show the DCGM is indeed much more effective than the implemented CVM method.

Another method that might be feasible is described and used by Shahbazi et al. (2007). This method discretely solves the Poisson equation for each patch, not only using the patch his neighbours in calculation, but the next-nearest patch neighbours as well. This method shows promise and second order convergence.

As the CVM has non-ideal behaviour for larger angular refinement, low angular flux convergence and as there are more effective techniques such as the DCGM available, it is recommended to use such discretisation schemes over the CVM.

Appendix A

A.1 Results alternative manufactured solution, no streaming

Using the angular manufactured solution of $\Phi(\mathbf{\Omega}) = C_0 + C_1\Omega_x + C_2\Omega_y + C_3(3\Omega_x^2 - 1)$ with $C_{0:3} = 1; 3; 2; 1$ and the spatial depended part being $R(\boldsymbol{r}) = 1$, no streaming. The manufactured solution is the same as used for the results of tables 3.2 and 3.3 but without streaming and spatial dependencies. Table A.1 gives the results for angular flux errors and the CoM flux errors are presented in table A.2.

Table A.1: Angular flux error, $\Sigma_a = 10\,\mathrm{cm}^{-1}$, $\sigma_{tr} = 2\,\mathrm{cm}^{-1}$,$C_{0:3} = 1; 3; 2; 1$ uniform refinement.

	Spherical Method			Octagon Method		
N	Angular flux error	Steps	Order of conv (p)	Angular flux error	Steps	Order of conv (p)
8	2.106E-01	4		2.106E-01	4	
32	1.057E-01	6	0.995	1.057E-01	6	0.995
128	5.660E-02	12	0.901	5.711E-02	11	0.888
512	2.957E-02	21	0.937	2.912E-02	21	0.972
2048	1.664E-02	40	0.830	1.491E-02	42	0.966
8192	1.125E-02	78	0.564	8.002E-03	82	0.897
32768	9.441E-03	144	0.253	4.945E-03	157	0.694

Table A.2: CoM flux error, $\Sigma_a = 10\,\mathrm{cm}^{-1}$, $\sigma_{tr} = 2\,\mathrm{cm}^{-1}$,$C_{0:3} = 1; 3; 2; 1$ uniform refinement.

	Spherical Method			Octagon Method		
N	CoM flux error	Steps	Order of conv (p)	CoM flux error	Steps	Order of conv (p)
8	2.650E-02	4		2.650E-02	4	
32	1.690E-02	6	0.650	1.690E-02	6	0.650
128	9.694E-03	12	0.801	5.745E-03	11	1.56
512	9.228E-03	21	0.071	3.550E-03	21	0.695
2048	9.277E-03	40	-0.008	3.329E-03	42	0.093
8192	9.309E-03	78	-0.005	3.329E-03	82	0.000
32768	9.319E-03	144	-0.002	3.340E-03	157	-0.005

A.2 Results cubic angular dependency, with streaming

Using the angular manufactured solution of $\Phi(\mathbf{\Omega}) = C_0 + C_1\Omega_x + C_2\Omega_y + C_3(3\Omega_x^2 - 1) + C_4(5\Omega_y^3 - 3\Omega_y)$ with $C_{0:5} = 1; 3; 2; 1; 0.5$ and the spatial depended part of $R(\boldsymbol{r}) = x(1-x)y(1-y)$ the angular and scalar errors and calculated as well as their order of convergence with regards to angular refinement. Fourth order spatial basis functions are used. The results for the spherical and octagon method are presented in tables A.3 and A.4 respectively.

Table A.3: Spherical method, $\Sigma_a = 10\,\mathrm{cm}^{-1}$,$\sigma_{tr} = 2\,\mathrm{cm}^{-1}$, $C_{0:4} = 1; 3; 1; 0.5$, uniform refinement

N	Steps	Angular flux error	Order of conv. (p)	Scalar flux error	Order of conv. (p)
8	4	6.025E-01		8.702E-02	
32	7	3.007E-01	1.00	2.347E-02	1.89
128	16	1.700E-01	0.823	5.135E-03	2.19
512	39	1.029E-01	0.724	1.264E-03	2.02
2048	91	7.683E-02	0.422	2.184E-03	-0.788
8192	190	6.880E-02	0.159	2.572E-03	-0.236

Table A.4: Octagon method, $\Sigma_a = 10\,\mathrm{cm}^{-1}$,$\sigma_{tr} = 2\,\mathrm{cm}^{-1}$, $C_{0:4} = 1; 3; 1; 0.5$, uniform refinement

N	Steps	Angular flux error	Order of conv. (p)	Scalar flux error	Order of conv. (p)
8	4	6.025E-01		8.702E-02	
32	7	3.007E-01	1.00	2.347E-02	1.89
128	16	1.748E-01	0.783	6.035E-03	1.96
512	39	1.072E-01	0.706	1.411E-03	2.01
2048	79	8.101E-02	0.404	1.126E-03	0.325
8192	154	7.298E-02	0.151	1.304E-03	-0.212

A.3 Plots scalar and angular flux errors

Plotting the scalar and angular error against the angular grid size (h) to visualise the convergence using the data from Tables 3.2 and 3.3 in Figure A.1 and Tables A.3 and A.4 in Figure A.2.

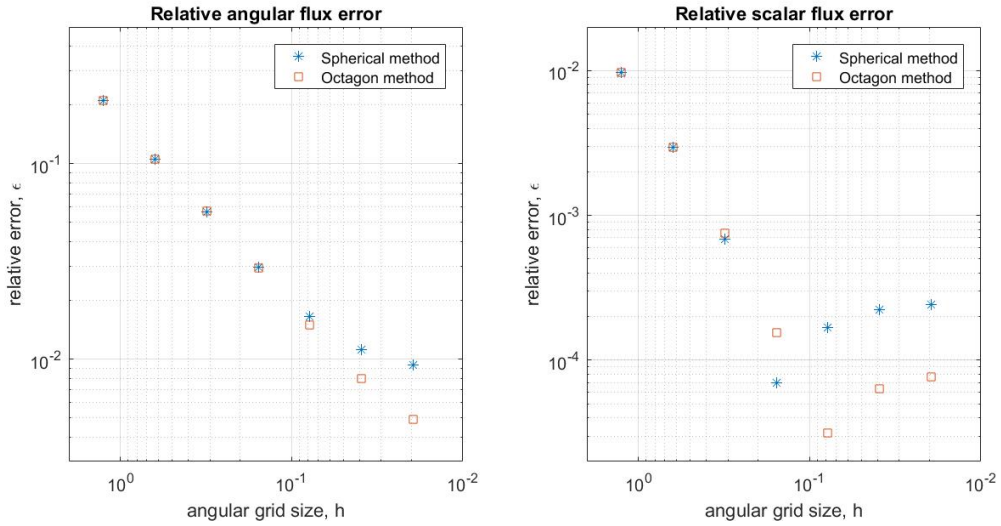


Figure A.1: The angular (left) and scalar (right) flux error plotted against the characteristic patch size (h) using the data of tables 3.2 and 3.3

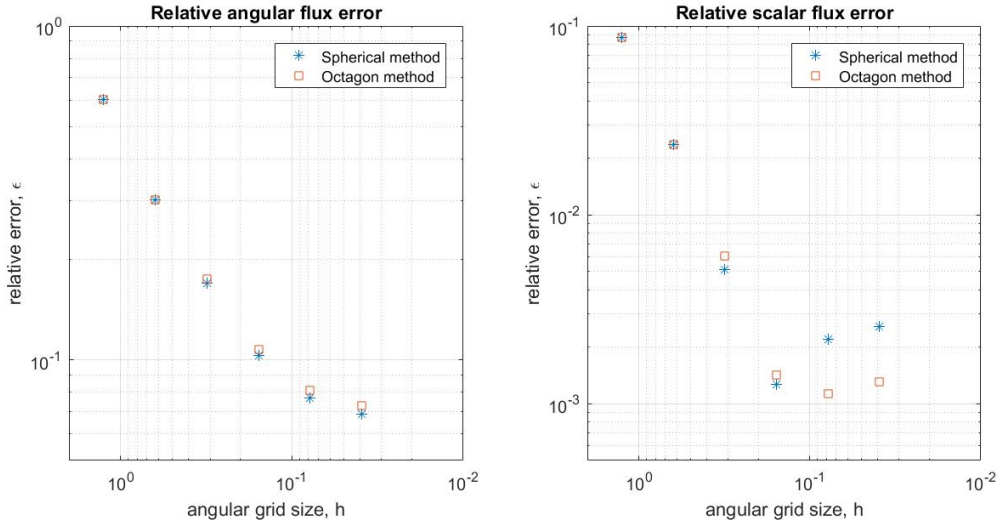


Figure A.2: The angular (left) and scalar (right) flux error plotted against the characteristic patch size (h) using the data of tables A.3 and A.4.

Bibliography

Borgers, C. and Larsen, E. W.: 1996, On the accuracy of the fokker-planck and fermi pencil beam equations for charged particle transport, *Medical Physics* **23**(10), 1749–1759.

Czito, B. G., Anscher, M. S. and Willet, C. G.: 2006, Radiation therapy in the treatment of cholangiocarcinoma, *BioMed Research International* **20**(8), 873–874.

Duderstadt, J. J. and Hamilton, L. J.: 1976, *Nuclear Reactor Analysis*, John & Wiley Inc.

Grant, J. D. and Chang, J. Y.: 2014, Proton-based stereotactic ablative radiotherapy in early-stage non-small-cell lung cancer, *BioMed Research International* .

Hennink, A.: 2015, *A discontinious galerkin method for charged particle transport in the fokker-planck limit*, master thesis, TU Delft.

Hennink, A. and Lathouwers, D.: 2017, A flexible discontinious galerkin method for the mono-energetic fokker-planck equation, *Unpublished* .

Jiang, H., Seco, J. and Paganetti, H.: 2007, Effects of hounsfield number conversion on ct based proton monte carlo dose calculations, *Medical Physics* **34**(4), 1439–1449.

Lathouwers, D.: 2014, Essentials of discontinuous galerkin finite element discretiation for neutron transport, pdf handout.

Morel, J. E.: 1981, Fokker-planck calculations using standard discrete ordinates transport codes, *Nuclear Science and Engineering* **79**, 340–356.

Nauta, J. B.: 2016, *Finit volume method for charged particle transport in the fokker-planck limit*, bachelor thesis, TU Delft.

Paganetti, H., Jiang, H., Parodi, K., Slopsema, R. and Engelsman, M.: 2008, Clinical implementation of full monte carlo dose calculation in proton beam therapy, *Physics in Medicine and Biology* **53**, 4825–4853.

Pantankar, S. V.: 1980, *Numerical Heat Transfer and Fluid Flow*, Taylor And Francis. Used as study material for understanding the control volume method (Chapter 4).

Rossi, B. and Greisen, K.: 1941, Cosmic-ray theory, *Review of Modern Physics* **13**(4), 240.

Shahbazi, K., Fischer, P. F. and Ethier, C. R.: 2007, A high-orderdiscontinous galerkin method for the unsteady incompressible navier-stokes equations, *Journal of Computational Physics* **222**(1), 391–407.

Uilkema, S. B.: 2012, *Proton therapy planning using the s_n method with the fokker-planck approximation*, master thesis, TU Delft.