Delft University of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft Institute of Applied Mathematics

Goal-Oriented Angular Adaptive Algorithm using Sensitivity Analysis for the Transport Equation and Boltzmann-Fokker-Planck Equation

A thesis submitted to the
Delft Institute of Applied Mathematics
in partial fulfillment of the requirements

for the degree

MASTER OF SCIENCE
in
APPLIED MATHEMATICS

by

D.J. Koeze

Delft, the Netherlands
May 2012

MSc THESIS APPLIED MATHEMATICS

"Goal-Oriented Angular Adaptive Algorithm using Sensitivity Analysis for the Transport Equation and Boltzmann-Fokker-Planck Equation"

D.J. Koeze

Delft University of Technology

Daily supervisor

Dr. N.V. Budko

Responsible professor

Prof. dr. ir. C. Vuik

Other thesis committee members

Dr. ir. D. Lathouwers

May 2012

Delft, the Netherlands

Abstract

In this work we examined the discretised form of Boltzmann-like transport, i.e. the neutron transport equation and the Boltzmann-Fokker-Plank (BFP) equation with the discontinuous Galerkin method and polynomial basis functions. In particular we examined an adaptive algorithm, which bases its decision of where to refine on the adjoint problem, as in sensitivity analysis. In this way an accurate detector response should be obtained in an efficient manner.

The goal-orietend criterion uses the adjoint solution as a measure for importance to the detector response. This refinement technique is compared to traditional methods, which base refinement on the change in the solution of a local test refinement, and to the discrete ordinates method.

Problems with one spatial and one angular dimension are used to test the adaptive algorithm. In previous work the same problems were solved with first order polynomials in the spatial direction and zeroth order polynomials in the angular direction. We saw then that constant patches, zeroth order polynomials in the angular part of the domain, could not represent the angular flux in diffusive materials accurately. We furthermore saw that the quality of the error estimate with the global adjoint approximation was reasonable, while with the local adjoint approximation it was poor.

In this work we employed first order polynomials in both the spatial and angular domains. Linear patches, the first order polynomials in the angular domain, provide a better approximation of the angular flux using less unknowns. This can be seen when comparing both the constant and linear patches to the discrete ordinates method. Also the diffusive materials are now much easier to approximate, which has as effect that traditional refinement gives smaller errors than with constant patches.

The quality of the error estimate with linear patches, however, is poor for almost every test case. A cause can be found in the approximation of the exact adjoint solution that is used in the estimate. This approximation is done by refining all the current mesh uniformly one level. It appears the space of the deeper level is only slightly larger than the coarser level on which the forward problem is solved, in the sense that the adjoint solution on that deeper level does not yield information not contained in the coarser level. The solution we propose is to solve the adjoint problem with higher order basis functions than those of the forward problem.

Finally we present some results on the Boltzmann-Fokker-Planck equation discretised with discontinuous Galerkin. The discrete operator one obtains is very similar to the discrete transport operator, with extra bands added within a spatial element. To solve this system of equations one could use a Gauss-Seidel iteration, which means one sometimes uses outdated angular flux values of neighbours, or Kylov subspace methods, as the matrix operator is not computed explicitly.

Contents

List of Figures

List of Symbols

α	Momentum transfer in BFP equation	cm^{-1}
γ	Spatial basis function	
η	Contribution to the error estimator of a patch	$cm^{-3}s^{-1}$
λ	Mean free path of a neutron	cm
μ	Cosine of the polar angle	
μ'	Average polar angle cosine of a patch	
σ_D	Macroscopic detector cross section	cm^{-1}
σ_s	Macroscopic scattering cross section	cm^{-1}
σ_t	Macroscopic total cross section	cm^{-1}
Φ	Scalar neutron flux	$cm^{-2}s^{-1}$
ϕ	Angular neutron flux	$cm^{-2}s^{-1}$
ϕ^*	Exact adjoint solution	
Ω	Direction	
$\hat{\mathbf{\Omega}}$	Unit vector of direction of motion	
$\hat{\mathbf{\Omega}}'$	Average direction of a patch	
ω	Azimuthal angle	
B	Block bands of transport operator	
$B(u,v)$	Bilinear form in weak formulation of transport equation	
c	Scatter ratio, total over scatter cross section	
D	Block diagonal part of transport operator	
E	Energy of a neutron	eV
E	Projection matrix between different refinement levels	
E_{det}	Error in the detector response	$cm^{-2}s^{-1}$
E_{rms}	Node wise root mean square error of scalar flux	$cm^{-3}s^{-1}$
G	Basis function of a patch	

J	Detector response	s^{-1}
$\boldsymbol{K}$	Matrix with products of spatial basis functiosn in streaming term	
L	Transport Equation operator	
$l(v)$	Linear form in weak formulation of transport equation	
L^*	Adjoint transport operator	
$\boldsymbol{M}$	Matrix with products of spatial basis functions	
$\hat{\boldsymbol{n}}$	Normal vector	
P	Continuity relation matrix	
q	Partial scatter source	$cm^{-3}s^{-1}$
$\boldsymbol{r}$	Position vector	cm
R	Volumetric residual	$cm^{-3}s^{-1}$
r	Boundary residual	$cm^{-3}s^{-1}$
S	Scatter operator	
s	External source in operator expression	$cm^{-3}s^{-1}$
s	External source in the transport equation	$cm^{-3}s^{-1}$
T	Transport operator	
t	Time	s
$\boldsymbol{U}$	Matrix with products of spatial basis functions on a boundary	
V	Space in which weak solution of transport equation lies	
v	Element of function space used for Galerkin method	
$v(E)$	Speed of a neutron with energy E	cms^{-1}
V_h	Subspace of functions in which the Galerkin discretised solution lies	
$aprx$	Approximate subscript	
c	Coarse element subscript	
Δ	Size of interval or surface area of element	
δ	Boundary of a set or space	
e	Element index subscript	

f	Element face subscript
f	Fine element subscript
G	Global adjoint approximation subscript
h	Discrete subscript
int, ext	Superscript indicating internal and external with respect to an element
k	Sweep iteration index superscript
L	Local adjoint approximation subscript
l	l times scattered superscript
l	Source iteration index superscript
n	Direction subscript
p	Patch index subscript
U, D	Upwind, Dirichlet boundary subscript
x, y, z	Spatial component subscript

1 Introduction

Nuclear physics is, since the 1940's, part of our daily lives, albeit not conscientiously for most of us. We use it to produce electricity, for medical treatments, in materials science and to build weapons. Several fields of physics are involved in these technologies, for example thermodynamics when heat is produced and all kinds of radiation which are released by the radioactive atoms. In this work we will be considering one kind of radiation in particular; neutral and charged particles that are produced in radioactive decay.

Let us take a closer look at two of the most widely used technologies, energy production in nuclear power plants and medical treatments. The nuclear power plant works very similar to a power plant that burns fossil fuels, a heat source produces steam, which drives the electric generator. Uranium, or another fissile material, is used instead of fossil fuels and the 'burning' process is a bit different. Certain atoms in the reactor core can break apart and in the process produce different kinds of radiation and heat. The heat is used to vaporize water and some of the radiation is used to keep the reactor burning. Neutrons are the driving force in this process, they are absorbed and induce fission in atoms, which in turn produces new neutrons.

Neutrons travel through the reactor core and will interact, through collisions and absorption, with the surrounding material. The neutron transport equation, or the Boltzmann equation, describes the behaviour of the neutrons. Being able to solve the neutron transport equation efficiently and accurately means more efficient and safer reactor core designs.

Neutrons, but also charged particles, can be used in medical treatments, for example to irradiate and treat a tumour. In such a treatment one wants to damage the tumour, but not the surrounding, healthy tissue. Nuclear physics, and more specifically the neutron transport equation or the Boltzmann-Fokker-Plank (BFP) equation, enables us to make accurate predictions as to how a tumour should be treated.

In this work we therefore also investigate the Boltzmann-Fokker-Planck equation, also known as the Fokker-Planck equation, which is the transport equation with some additional terms. The transport equation describes the transport of neutral particles, while the BFP equation describes charged particles. This means the force between particles originating from the charge of the particles is taken into account.

The BFP equation was first introduces by Andrey Kolmogorov in 1931, therefore it is sometimes also called the Kolmogorov forward equation. We will call it the BFP equation, to emphasize the fact that it is an adaptation of the Boltzmann transport equation. There are more uses of this equation apart from charged particle transport, as it describes the time evolution of the probability density function of the velocity of particles in quantum mechanics.

One of the focuses of this work was to investigate how the BFP equation can be solved using the discontinuous Galerkin discretisation method. We present the discretisation and propose a way of solving the system of equations obtained from that discretisation. However, due to unexpected behaviour of the adjoint solution of the transport equation there was a lack of time to implement the BFP equation in a working code.

Another focus of this work is to develop an adaptive algorithm that solves the transport equation efficiently. In each iteration the solution of the differential equation is obtained and from that solution a choice is made where the mesh needs to be refined. In this way one keeps the

computational and memory costs at a minimum, while obtaining an accurate result.

In many cases one is interested in a set-up with a source of neutrons, or other particles, and a detector, where the quantity of interest is the detector response. Our goal is then to minimize the error in the detector response with as few as possible unknowns. To achieve this we need a criterion that will choose where in the mesh the refinement is most efficient, i.e. decreasing the error maximally by minimal refinement.

Traditional methods of refinement are based on observing the change in the solution of the differential equation upon refinement. For example, one refines locally and checks the difference in the solution this produced. Repeating this for each element or cell in the mesh one can find the locations where refinement leads to the largest changes in the solution, hopefully reducing the error efficiently.

In this work we compare a traditional critirion to a refinement criterion which bases its decision on the importance of each particular refinement location for the detector response. This importance is obtained by solving the adjoint problem, which turns out to be almost the same as the forward problem. Exactly how the adjoint problem should be discretised and used is a matter of discussion.

Both equations have a solution that is seven dimentional, with three spatial dimensions, two angular dimensions, energy and time. All these dimensions are of a different nature, calling for different discretisation or simulation techniques.

Here, without loss of much generality, we look into a simplified version of these equations. The problems that we consider are steady-state and have one energy group and one spatial dimension. This means we are left with a two-dimensional problem, with one spatial and one angular variable.

The discretisation technique we apply here is the discontinuous Galerkin method, which has been around for almost a century. This method requires that a differential equation is solved in its weak form on a restricted space.

This master thesis is a continuation of a master thesis done at the applied physics department of the Delft University of Technology. The two projects are to be evaluated separately, but the results of [4] are necessary for a full understanding of the present work.

Titled ‘Goal-oriented angular adaptive neutron transport using a spherical discontinuous Galerkin method’, [4] is an investigation into the performance of a relatively new numerical method. This method enables us to adaptively solve the transport equation in the angular domain, which means we can more efficiently use our computational power and obtain more accurate results. The mathematical description of the method is presented for the three dimensional case, while numerical results for a one dimensional model are presented. A shortened version of these results is presented in this chapter, for a full understanding of the work done in this thesis see [4].

The first section of this thesis is a summary of the applied physics thesis where all relevant information is briefly discussed. This includes the discretisation of the transport equation with constant basis functions in the angular part of the domain, derivation of the goal-oriented and traditional error criteria and the results of the implementation of this discretisation. Since irrelevant parts are left out the summary does not include all considerations and questions that were discussed in the physics thesis.

An investigation into the algorithm used with the constant patches and later on with the linear patches is presented in the second section. Both the method of solving the equation and of estimating the error are subjects of investigation in this section. Also in this section we show that

switching the order of discretisation and taking the adjoint problem is not symmetrical, that is, the adjoint of the discrete forward problem is not the same as the discretised adjoint problem.

The third and fourth sections are on the implementation of the linear basis functions in the angular domain, so we are using linear basis functions to discretise the whole domain. First we will look at what needs to be altered to the discretisation of constant patches to handle linear patches. Also conservation issues are addressed in this section. After the theoretical basis is made we present the results of the implementation of that method. This is done with eight test cases, which can be found in the appendix.

In the fourth section we found that the error estimate that could be used as a stopping criterion is of very poor quality. A cause for this is presented in section five. In the estimate an approximation of the adjoint solution is needed, which in the tests is taken as the solution on a more refined mesh. It turns out this is probably not an accurate approximation for the part of the adjoint solution that does not lie in the restricted Galerkin space. Two ways of improving the approximation are discussed.

Finally we present in the sixth section and investigation into the Boltzmann-Fokker-Planck equation, which was originally one of the focusses of this work. The other focus, linear patches for the transport equation, would have to be implemented in the BFP equation for a good approximation of the angular flux. Since we ran into the loss of symmetry in the derivation of the adjoint, the section on the BFP equation is not very elaborate. However we did discretise the equation and derived in what ways the BFP operator matrix differs from the transport operator. From this structure we also present possible methods of solving this system of equations.

This thesis is based upon a thesis in the field of applied physics at the Delft University of Technology. This thesis is done in the context of the master thesis research of the applied mathematics curriculum at the same university. It was conducted in the Numerical Analysis group of the Electrical Engineering, Mathematics and Computer Science faculty at that university.

2 Neutron Transport with Constant Angular Basis Functions

We start with a short introduction of the neutron transport equation, which is the subject of the physics thesis. Both the physical and mathematical background of the equation will be touched upon. Discretization of the equation is the subject of the next part, where we will both present the discretized equations as well as go into more detail on why this method is to be preferred over other methods. In this section we will find that continuity relations are needed for neutrons crossing a boundary of elements with a different angular discretization. This will be discussed in depth in the third part. The fourth part is then on the development of the adaptive criterion on which the choice refinement is based. This criterion is derived in this section. In the final theoretical section we present an overview of the adaptive algorithm as is used in the one dimensional problems. Finally, we present some results of the one dimensional model.

2.1 Introduction to the Transport Equation

In nuclear reactor physics neutrons are very important, since neutrons are necessary for the chain reaction of fission to continue. In fact, since neutrons can travel through the reactor core and fuel is fixed, neutrons are the main subject in reactor physics. The neutron transport equation is the governing equation of free neutrons in the reactor core, or, for that matter, in any geometry or substance. In certain geometries the transport equation can be solved analytically, however for many real world problems numerical solutions are needed. In the process of designing nuclear reactors or other nuclear facilities accurate approximation of the neutron density or neutron flux are needed to predict for example heat production in the core. New numerical techniques are still developed, which will result in more accurate reactor designs. This section contains some basic remarks on the neutron transport equation, which are necessary for this report. A detailed description of the properties and derivation of the transport equation can be found in many nuclear engineering hand books. [6] [3]

Free neutrons are the neutrons that are important to reactor physics. These neutrons can freely move through the material and take part in reactions with the surrounding material. Common reactions are collisions (scatter) and absorption. Neutrons that are captured in the nucleus of an atom are not important to reactor physics, as these neutrons will not take part in any important reactions. The neutron transport equation therefore considers the free neutrons and describes the rate of reaction as well as the movement (streaming) of the neutrons. A full derivation of the transport equation, as well as many applications, can be found in [3]. The most general form of the transport equation is

$$\begin{aligned}&\frac{1}{v(E)}\frac{\partial\phi(\boldsymbol{r},E,\hat{\boldsymbol{\Omega}},t)}{\partial t}+\hat{\boldsymbol{\Omega}}\cdot\nabla\phi(\boldsymbol{r},E,\hat{\boldsymbol{\Omega}},t)+\sigma_t(\boldsymbol{r},E,t)\phi(\boldsymbol{r},E,\hat{\boldsymbol{\Omega}},t)=\\&\int_{4\pi}d\Omega'\int_0^\infty dE'\sigma_s(\boldsymbol{r},E'\to E,\hat{\boldsymbol{\Omega}}'\to\hat{\boldsymbol{\Omega}},t)\phi(\boldsymbol{r},E',\hat{\boldsymbol{\Omega}}',t)+s(\boldsymbol{r},E,\hat{\boldsymbol{\Omega}},t)\end{aligned}\tag{2.1}$$

Some of the symbols in this equation are introduced here shortly, a nomenclature can be found in the front of the report. The angular flux ϕ is the quantity of interest, it can be interpreted as the

density of the number of neutrons that is at $\boldsymbol{r}$, has an energy E and travels in direction $\hat{\boldsymbol{\Omega}}$ at time t. The total cross section σ_t and the scattering cross section σ_s tell us with what rate the reactions occur.

We will now shortly discuss the physical meaning of each of the terms. The first term on the left hand side is the change in neutron density over time. The next term is the streaming term of the equation, it follows from applying Gauss' theorem on the expression describing the neutrons travelling into a control volume. Finally on the left hand side we have the total removal of neutrons, proportional to the total removal cross section σ_t. The right hand side contains two terms, the second term is an external source, which can be arbitrarily specified. Scattering is described by the first term of the right hand side. This can be considered a source as neutrons with other energies and travelling in other directions can be scattered into the part of phase space considered. Therefore the term contains an integral both over all directions and all energies. Please note that this general form of the transport equation does not describe fission reactions. Fission reactions result in an extra source term that has the same form as the scatter term.

To test the performance of numerical techniques it is not always necessary to consider the general transport equation. Some approximations or simplifications can be made without altering the behaviour or the complexity of the equation. A number of these adjustments is used in this report, which are discussed in this section.

The time dependence is eliminated in most proofs of principle. When solving a time dependent problem one usually approximates a certain state of the problem at time t as a steady-state problem. To solve a time dependent problem a number of steady-state problems is solved sequentially. Therefore we can equate the first term of the general transport equation to zero: $\frac{1}{v(E)}\frac{\partial\phi(\boldsymbol{r},E,\hat{\boldsymbol{\Omega}},t)}{\partial t} = 0$.

Another common discretization in most solvers is that the energy dependence is discretized into groups. All neutrons are put into energy 'bins' and cross sections are used to determine the number of neutrons that switch bins or stay in the same bin. The driving force of switching bins is scatter, since in scatter reactions neutrons may lose or gain energy. The simplest case is to consider just one bin, this effectively eliminates energy dependence from the problem. This report uses this one group approach.

The scatter source term is further simplified by considering isotropic scatter only. This means the angular dependency of the scatter cross section σ_s is neglected. To account for anisotropic scatter multiple techniques exist, the most common being the expansion of the scatter cross section in spherical harmonics or Legendre polynomials. In this report there is no background information on this.

When applying all these simplifications the transport equation becomes:

$$\hat{\boldsymbol{\Omega}} \cdot \nabla\phi(\boldsymbol{r},\hat{\boldsymbol{\Omega}}) + \sigma_t(\boldsymbol{r})\phi(\boldsymbol{r},\hat{\boldsymbol{\Omega}}) = \frac{\sigma_s}{4\pi}\Phi(\boldsymbol{r}) + s(\boldsymbol{r},\hat{\boldsymbol{\Omega}}) \tag{2.2}$$

where the scalar flux Φ is defined as $\Phi(\boldsymbol{r}) = \int_{4\pi}\phi(\boldsymbol{r},\hat{\boldsymbol{\Omega}})d\hat{\boldsymbol{\Omega}}$.

Some properties of the equation are noteworthy, as they play a role in the performance of numeric solvers. First there is the difference between an optically thick and thin medium. A thick medium means the mean free path λ of neutrons is small. The mean free path compared to the size of the domain determines the probability of neutrons to leak out of the problem. When the

domain is much larger than the mean free path the problem is optically thick. The mean free path is determined by the inverse of the total cross section,

$$\lambda = \frac{1}{\sigma_t} \tag{2.3}$$

A thick medium in a small domain can however result in an optically thin problem. Conversely a thin medium, with a large mean free path λ may result in optically thin and optically thick medium. However the size of the domain will be much larger for a thin medium to make a thick problem.

Besides the thickness of the material, one can also vary the ratio with which the two most important reactions, absorption and scatter, occur. This ratio is called the scatter ratio c, and is

$$c = \frac{\sigma_s}{\sigma_t} \tag{2.4}$$

The larger c the more diffusive the material will be. In diffusive materials the transport equation can be approximated by the diffusion equation. In the results it will show that the scatter ratio c of the material influences the performance of numerical methods.

In this work this equation is mostly studied in its one-dimensional form and all numerical results are for this case. To arrive at the one dimensional form we must assume the physical properties of the material vary only in one direction. In mathematical notation this reads

$$\sigma_t(x, y, z) = \sigma_t(z) \tag{2.5}$$
$$\sigma_s(x, y, z) = \sigma_s(z) \tag{2.6}$$
$$s(x, y, z) = s(z) \tag{2.7}$$

Now the angular flux ϕ will only vary along the z direction and along the polar angle θ. This can be shown by considering the flux ϕ at two points that are different only in the x and y components. The physical surroundings of these points is exactly the same, making them indiscernible, therefore the flux will be the same. In other words we know $\phi(x, y, z, \omega, \theta) = \phi(z, \omega, \theta)$. Furthermore we observe that the flux ϕ along ω, the azimuthal angle, does not change as a result of this simplification.

We arrive then at the transport equation for one spatial dimension and one angular dimension,

$$\mu \frac{\partial \phi(z, \mu)}{\partial z} + \sigma_t(z)\phi(z, \mu) = \frac{\sigma_s}{4\pi} \int_{4\pi} \phi(z, \mu) d\hat{\mathbf{\Omega}} + s(z) \tag{2.8}$$

From now on, for conventional purposes, we will call the z directions the x direction. Although there will be thoughts on the three dimensional case in this work, the equation we investigate is the one dimensional transport equation in the form presented above.

2.2 Discretisation of the Transport Equation

To discretise the transport equation some questions have to be answered first. As the goal is to have a discretisation that can handle refinement in the angular domain well we need a discretisation that allows for this. Furthermore the discretised system of equations must be easily solved. Both

questions were addressed in the physics thesis and the main results are repeated here. We will start with the advantages and drawbacks of the methods of discretisation that were considered.

The first method that was discussed is the discrete ordinates method, which is the most widely used method to discretise the angular domain of the transport equation. It comes down to requiring the transport equation to hold for a discrete and finite set of directions, or values for μ in one dimension. The integral of the scalar flux Φ is then approximated by a quadrature rule. In one dimension most often the Gauss-Legendre quadrature is used, as this provides exact integration of polynomials up to a certain degree. In three dimensions the approximation is not as good as in one dimension, since the quadratures for three dimensions introduce larger errors. Note that this only provides us with a discretisation of the angular domain. In the spatial domain another method has to be used. In the test cases presented throughout the work we include results of this method, with linear discontinuous Galerkin in the spatial domain, for referencing purposes.

Then we turned to discontinuous Galerkin, where we have two options for the choice of basis functions. The first is wavelets, which were investigated because of their hierarchical nature. This hierarchy can be used for refinement, as wavelets have the same shape on each level, the numerical computations are the same on each level. However, it turned out the hierarchical structure has a major drawback. The cost of the wavelet algorithm is large, as the wavelets of different levels have overlapping support. This support results in matrices that are largely filled, so we cannot use any algorithms that use sparse matrices. For more on the use of wavelets see [2], [1].

Finally we have the discontinuous Galerkin method with polynomial basis functions both in space and direction. This method can be used in a hierarchical way and results in a relatively cheap algorithm. This is the method used in the numerical tests of the thesis. Below the discretisation of the transport equations using this method is presented.

Several researches have been done on the spatial discretization with Discontinuous Galerkin in the field of neutron transport. This report does not focus on the behavior of the spatial part of the transport equation, but on the angular part. For more on the spatial properties of Discontinuous Galerkin see [9] [10].

We start the discretisation with stating the transport equation

$$\hat{\mathbf{\Omega}} \cdot \nabla \phi + \sigma_t \phi = s + \frac{\sigma_s}{4\pi} \Phi \tag{2.9}$$

where $\hat{\mathbf{\Omega}}$ is the angular vector, ϕ is the angular flux and Φ is the scalar flux. For the application of the discontinuous Galerkin method we first define the space V_h in which the solution will lie. This is done by splitting the angular flux in a spatial and an angular part. The spatial part $\phi_j(\boldsymbol{r})$ will be approximated by first order polynomials in this report, but can in principle be any polynomial. Analogously, the angular part will be approximated by zeroth order polynomials (constant functions) in this report, but can also be other polynomials. This can be formulated as the following sum

$$\phi(\boldsymbol{r}, \hat{\mathbf{\Omega}}) \approx \sum_{e}^{E} \sum_{p}^{P} \phi_{e,p} \gamma_e(\boldsymbol{r}) G_{e,p}(\hat{\mathbf{\Omega}}) \tag{2.10}$$

where E is the number of spatial elements and P is the number of patches in a spatial element, or on a spatial location. We can now multiply the equation from the left by a test function G_p in our

space V_h. Then integrate over the whole angular domain to obtain

$$\int_V \int_{4\pi} \gamma_e(\boldsymbol{r}) G_{e,p}(\hat{\boldsymbol{\Omega}}) \left[(\hat{\boldsymbol{\Omega}} \cdot \nabla + \sigma_t) \sum_{e',p'} \phi_{e',p'} \gamma_{e'} G_{e',p'} - s - \frac{\sigma_s}{4\pi} \Phi \right] d\hat{\boldsymbol{\Omega}} = 0 \tag{2.11}$$

For the scalar flux Φ in this equation we can obtain the following expression, since the patch function G_j is unity on the patch and zero elsewhere.

$$\begin{aligned} \Phi &= \int_{4\pi} \phi d\hat{\boldsymbol{\Omega}} && (2.12)\\ &= \int_{4\pi} \sum_{j=1}^{M} \phi_j G_j d\hat{\boldsymbol{\Omega}} && (2.13)\\ &= \sum_{j=1}^{M} \phi_j \int_{4\pi} G_j d\hat{\boldsymbol{\Omega}} && (2.14)\\ &= \sum_{j=1}^{M} \phi_j Area(G_j) && (2.15) \end{aligned}$$

Let from now on ΔG_j denote the area of patch G_j. Please note that G_j is just an indicator of that patch, G_j is unity on the patch and zero outside the patch. The value of the flux on that patch is in the parameter ϕ_j.

Let us now work out the angular integral of Equation 2.11. We will plug in the expression for the scalar flux and evaluate the integral over two angular basis functions. Since the patches all have compact support we know that

$$\sum_{j=1}^{M} \phi_j \int_{4\pi} G_p G_j d\hat{\boldsymbol{\Omega}} = \phi_p \Delta G_p \tag{2.16}$$

The equation can then be written as

$$\begin{aligned} \int_V \gamma_e \left[\hat{\boldsymbol{\Omega}}'_p \cdot \nabla \sum_{e'} \phi_{e',p} \gamma_{e'} + \sigma_t \sum_{e'} \phi_{e',p} \gamma_{e'} \right] d\boldsymbol{r} = \\ \int_V \gamma_e \left[\frac{\sigma_s}{4\pi} \sum_{e'} \sum_{p} \phi_{e',p} \gamma_{e'} G_{e',p} + s_p(\boldsymbol{r}) \right] d\boldsymbol{r} \end{aligned} \tag{2.17}$$

In this equation we have:

$$\hat{\mathbf{\Omega}}'_p = \frac{1}{\Delta G_p}\int_{\Delta G_p} \hat{\mathbf{\Omega}} d\hat{\mathbf{\Omega}} \tag{2.18}$$

$$\Phi = \sum_{j=1}^{M} \Delta G_j \phi_j \tag{2.19}$$

$$s_p = \frac{1}{\Delta G_p}\int_{4\pi} G_p s d\hat{\mathbf{\Omega}} \tag{2.20}$$

In the spatial part of the problem we use linear basis functions. So working out the spatial integral of Equation 2.17 will result in a matrix equation for each patch. The matrices will be two by two, since we have two spatial basis functions that have overlapping support. Before we work out the integral we first apply the divergence theorem to the streaming term, resulting in

$$\int_{\delta V} \hat{\mathbf{\Omega}}'_p \cdot \hat{\boldsymbol{n}} \gamma_e \sum_{e'} \phi^b_{e',p}\gamma_{e'} d\delta V - \int_V (\hat{\mathbf{\Omega}}'_p \nabla \cdot \sum_{e'} \gamma_{e'})\gamma_e \phi_{e',p} dV + \int_V \gamma_e \sum_{e'} \left[\sigma_t \gamma_{e'}\phi_{e',p} - \frac{\sigma_s}{4\pi}\gamma_{e'}\phi_{e',p} - s_p(\boldsymbol{r})\right] dV = 0 \tag{2.21}$$

In this equation $\phi^b_{e,p}$ is the angular flux at the boundary of a cell. Now we will assign to each element face its angular flux, e is the element index, f is the element face index. To do this we need to compose the total element boundary out of the individual element faces. Each of those individual faces is a plane or line, in order to be able to define an outward normal vector. This gives us the following expression

$$\delta V_e = \sum_{f=1}^{N_{faces}} \delta V_{e,f} \tag{2.22}$$

The flux on in flow boundaries is chosen to be the upwind flux. This means we have to make a distinction between directions as

$$\phi^b_{e,p,f} = \begin{cases} \phi_{e,p} & \text{if} \quad \hat{\mathbf{\Omega}}'_p \cdot \hat{\boldsymbol{n}}_f > 0 \\ \phi^{\text{upwind in } f}_{e,p} & \text{if} \quad \hat{\mathbf{\Omega}}'_p \cdot \hat{\boldsymbol{n}}_f < 0 \end{cases} \tag{2.23}$$

Plugging this into the equation we arrive at a matrix equation that has the final discretized form. The matrices are square and have the size of the number of basis functions that are in one spatial element. The matrix equation that is to be solved for each patch on each element is then

$$\sum_{f=1}^{N_{faces}} \Omega'_p \boldsymbol{U}_{e,p,f}\boldsymbol{\phi}^b_{e,p,f} + (-\Omega'_p \boldsymbol{K}_e + \sigma_t \boldsymbol{M}_e)\boldsymbol{\phi}_{e,p} = \frac{\sigma_s}{4\pi}\boldsymbol{M}_e \boldsymbol{\Phi}_e + \boldsymbol{s}_{e,p} \tag{2.24}$$

where

$$[\boldsymbol{U}]_{e_1,e_2,p,f} = \int_{\delta V_{e,f}} n_f \gamma_{e_1} \gamma_{e_2} d\delta V \tag{2.25}$$

$$[\boldsymbol{K}]_{e_1,e_2} = \int_{V_e} (\nabla \gamma_{e_1}) \gamma_{e_2} dV \tag{2.26}$$

$$[\boldsymbol{M}]_{e_1,e_2} = \int_{V_e} \gamma_{e_1} \gamma_{e_2} dV \tag{2.27}$$

$$s_{e_i,p} = \int_{V_e} \gamma_{e_i} s_p(\boldsymbol{r}) dV \tag{2.28}$$

$$= \int_{V_e} \int_{4\pi} \gamma_{e_i} G_{e_i,p} s(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) d\hat{\boldsymbol{\Omega}} dV \tag{2.29}$$

2.3 Continuity Relations

Two neighbouring elements do not necessarily have the same angular distribution of patches. This is a result of refining in the angular domain, which is not done uniformly but rather adaptively. When such a situation occurs we need interpolation rules to ensure neutron conservation across this boundary.

There are two cases that can be discerned, (a) neutrons flowing from a coarse to a fine element and (b) from a fine to a coarse element. In general neutron conservation for neutrons crossing a plane can be formulated as

$$\int_{\hat{\boldsymbol{\Omega}} \cdot \hat{\boldsymbol{n}}^{\text{upwind}} < 0} \hat{\boldsymbol{\Omega}} \cdot \hat{\boldsymbol{n}}^{\text{upwind}} \phi^{\text{upwind}}(\hat{\boldsymbol{\Omega}}) d\hat{\boldsymbol{\Omega}} = \int_{\hat{\boldsymbol{\Omega}} \cdot \hat{\boldsymbol{n}}^{\text{downwind}} > 0} \hat{\boldsymbol{\Omega}} \cdot \hat{\boldsymbol{n}}^{\text{downwind}} \phi^{\text{downwind}}(\hat{\boldsymbol{\Omega}}) d\hat{\boldsymbol{\Omega}} \tag{2.30}$$

[8]. In this equation ϕ^{upwind} is the flux in the upwind element and ϕ^{downwind} is the flux in the downwind element. Similar notation is used for the outward normal vectors $\hat{\boldsymbol{n}}$.This condition should be met in all points $\boldsymbol{r}$ along the boundary of the two elements, as this insures continuity of the neutron current. When the patches on the sphere are constant this will result in some simple continuity relations, they are derived here.

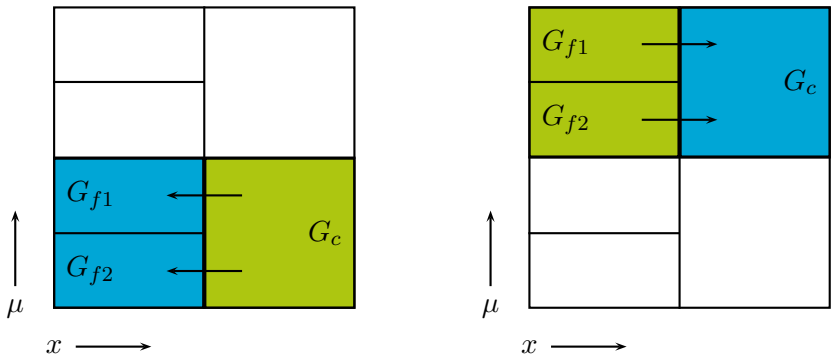
Take two elements A and B and consider the angles such that neutrons flow from A to B. Suppose element A has one patch G_c and element B has two patches G_{f1} and G_{f2}, this is illustrated in Figure 1(a). The continuity relation in Equation 2.30 can be expressed as follows

$$\int_{\Delta G_c} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} \phi(\hat{\boldsymbol{\Omega}}) d\hat{\boldsymbol{\Omega}} = \int_{\Delta G_{f1}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} \phi(\hat{\boldsymbol{\Omega}}) d\hat{\boldsymbol{\Omega}} + \int_{\Delta G_{f2}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} \phi(\hat{\boldsymbol{\Omega}}) d\hat{\boldsymbol{\Omega}} \tag{2.31}$$

$$\int_{\Delta G_c} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} \phi_c G_c d\hat{\boldsymbol{\Omega}} = \int_{\Delta G_{f1}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} \phi_{f1} G_{f1} d\hat{\boldsymbol{\Omega}} + \int_{\Delta G_{f2}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} \phi_{f2} G_{f2} d\hat{\boldsymbol{\Omega}} \tag{2.32}$$

$$\phi_c \int_{\Delta G_c} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} = \phi_{f1} \int_{\Delta G_{f1}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} + \phi_{f2} \int_{\Delta G_{f2}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} \tag{2.33}$$

Since the union of the support of patches G_{f1} and G_{f2} is equal to the support of G_c ($G_{f1} \bigcup G_{f2} = G_c$) we can simplify this to



(a) Coarse to fine (b) Fine to coarse

Figure 1: *Two cases of different angular distribution of patches in neighbouring elements. To ensure neutron conservation relations between the discrete patches need to be derived.*

$$\phi_c \int_{\Delta G_c} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} = \phi_{f1} \left[\int_{\Delta G_{f1}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} + \int_{\Delta G_{f2}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} \right] \tag{2.34}$$

$$\phi_c \int_{\Delta G_c} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} = \phi_{f1} \int_{\Delta G_{f1} \bigcup \Delta G_{f2}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} \tag{2.35}$$

$$\phi_c = \phi_{f1} \tag{2.36}$$

by assuming that $\phi_{f1} = \phi_{f2}$. It is natural to assume this, as the two patches G_{f1} and G_{f2} are equally as important. This result is also what one intuitively would expect. Even though the downwind element B can handle a more accurate solution of the flux, that information is not available. So the two patches G_{f1} and G_{f2} will represent the exact same angular flux as in the upwind element.

Now suppose we have a refined element C and a coarser element D. On C we have two patches, G_{f1} and G_{f2}, and on element D there is only one patch G_c. This is illustrated in Figure 1(b). We will use the same continuity condition as for the former case, which can be found in Equation 2.30. Applying this condition yields

$$\int_{\Delta G_c} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} \phi_c d\hat{\boldsymbol{\Omega}} = \int_{\Delta G_{f1}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} \phi_{f1} d\hat{\boldsymbol{\Omega}} + \int_{\Delta G_{f2}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} \phi_{f2} d\hat{\boldsymbol{\Omega}} \tag{2.37}$$

$$\phi_c \int_{\Delta G_c} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} = \phi_{f1} \int_{\Delta G_{f1}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} + \phi_{f2} \int_{\Delta G_{f2}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} \tag{2.38}$$

Note that since ϕ is constant on a patch we can pull out this variable. Rewriting this equation brings us to the final expression

$$\phi_c = \frac{\phi_{f1} \int_{\Delta G_{f1}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} + \phi_{f2} \int_{\Delta G_{f2}} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}}}{\int_{\Delta G_c} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}}} \tag{2.39}$$

This result can be interpreted as a weighted sum of the current through the two refined patches G_{f1} and G_{f2}, which will be clear after rewriting the equation as follows

$$\phi_c = \frac{\phi_{f1} \hat{\boldsymbol{n}} \cdot \int_{\Delta G_{f1}} \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}} + \phi_{f2} \hat{\boldsymbol{n}} \cdot \int_{\Delta G_{f2}} \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}}}{\hat{\boldsymbol{n}} \cdot \int_{\Delta G_c} \hat{\boldsymbol{\Omega}} d\hat{\boldsymbol{\Omega}}} \tag{2.40}$$

$$= \frac{\phi_{f1} \Delta G_{f1} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}}'_d + \phi_{f2} \Delta G_{f2} \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}}'_e}{\Delta G_c \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{\Omega}}'_f} \tag{2.41}$$

The sum of the fluxes ϕ_{f1} and ϕ_{f2} is weighted by the patch size and component of the average direction in the outward normal of the element. Please note that by Equation 2.18 we have $\Delta G_c \hat{\boldsymbol{\Omega}}'_f = \Delta G_{f1} \hat{\boldsymbol{\Omega}}'_d + \Delta G_{f2} \hat{\boldsymbol{\Omega}}'_e$. Using these continuity relations we ensure particle conservation.

2.4 Error Estimation and Adaptive Criterion

In this section we explain the different criteria that were used for determining where to refine patches. In other words, we must compute how large the contribution to the error is of each patch, so we can refine the patches that contribute most. Since we are interested in a good approximation of the detector response we will start by defining the error measure of the detector response, as well as an alternate measure for the error. This alternate measure is a global error measure.

Thereafter two criteria are formulated, the first being the goal-oriented method. This is called the goal-oriented method since it should make the error in the detector response as small as possible. The second criterion is called traditional refinement, since it considers how much the solution is changed by introducing local refinement. We consider this criterion to compare the goal-oriented refinement with more widely used methods.

2.4.1 Error Definition

The first error measure is that of the detector response, for which we first have to define the detector response. Many kinds of detector configurations are possible, but only a limited amount is physically relevant. Only the volumetric detector is used in this report.

$$J(\phi) = \int_V \int_{4\pi} \sigma_D \phi(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) d\hat{\boldsymbol{\Omega}} dV \tag{2.42}$$

In this expression, the cross section σ_D determines the rate at which neutrons are being measured. The dimensions of the detector are contained in this cross section. In regions where the detector is not present we take the cross section to be zero. Since we integrate over the whole detector region without discrimination in angle, neutrons in each direction have the same contribution to the detector.

With these detector responses we can define an absolute error measure, which is

$$E_{det} = |J_{ref} - J_h| \tag{2.43}$$

where J_{ref} is the 'exact' detector response and J_h is the detector response calculated with the current discrete solution.

When there is a comparison to the exact error in this report we do not use the exact error. A reference error is computed using a very fine angular mesh and this is taken to be equal to the exact error. The spatial elements are small, in the sense that the spatial part of the problem is converged. There will not be any spatial component in the error.

The other error measure with which we can compare methods is the root mean square error of the scalar flux Φ. This error is taken node wise, instead of integral wise:

$$E_{rms} = \sqrt{(\mathbf{\Phi}_{ref} - \mathbf{\Phi}_h) \cdot (\mathbf{\Phi}_{ref} - \mathbf{\Phi}_h)} \tag{2.44}$$

In this equation $\mathbf{\Phi}_h$ is the vector of scalar fluxes on the nodes of the current discrete solution. $\mathbf{\Phi}_{ref}$ is again the 'exact' solution, however, it is now a vector quantity.

The is a difference in what an accurate solution looks like according to the two norms, especially in geometries where the source and detector regions do not overlap. Then the absolute error of the detector response requires the solution in the detector to be very accurate, while in other regions it is not necessarily accurate. However, the rms error will require a solution that lies overall close to the exact solution, which may introduce a larger error in the detector response.

2.4.2 Goal-Oriented Adaptivity

First we will derive the goal-oriented criterion. We start with an alternate expression for the transport equation, which is in terms of the operator L and the external source S

$$L\phi = S \tag{2.45}$$

The adjoint operator or equation can be derived using inner products. An inner product in this case is an integral over the whole phase space of a product of two functions, or in mathematical notation

$$< f, g >= \int_V \int_{4\pi} f(\boldsymbol{r}, \hat{\mathbf{\Omega}}) g(\boldsymbol{r}, \hat{\mathbf{\Omega}}) d\hat{\mathbf{\Omega}} dV \tag{2.46}$$

With this notation the adjoint operator can be derived as follows

$$< \phi^*, L\phi >=< \phi^*, S > \qquad \Longleftrightarrow \qquad < L^*\phi^*, \phi >=< \phi^*, S > \tag{2.47}$$

where L is the forward operator and L^* is the adjoint operator. We can chose the right hand side of the adjoint problem equal to the detector cross section, in other words $L^*\phi^* = \sigma_D$. This leads to

$$< L^*\phi^*, \phi >=< \sigma_D, \phi >=< \phi^*, S >= J \tag{2.48}$$

where J is the detector response. The relation $< \sigma_D, \phi >=< \phi^*, S >$ is known as the duality relation. The explicit expression for the forward operator L in our case is

First we introduce some sets that are needed for expressions later on. The domain in phase space of one element can be written as a set that takes care of the spatial part and one that takes care of the angular part.

$$e = \{\boldsymbol{r} \in \text{element e}\} \tag{2.49}$$
$$\Omega = \{\hat{\boldsymbol{\Omega}} \in 4\pi\} \tag{2.50}$$

We can divide the angular set Ω into two sets, one for in flowing and one for outflowing directions on an edge δe.

$$\Omega^+ = \{\hat{\boldsymbol{\Omega}} \in \Omega | \hat{\boldsymbol{\Omega}} \cdot \hat{\boldsymbol{n}}_{\delta e} > 0\} \tag{2.51}$$
$$\Omega^- = \{\hat{\boldsymbol{\Omega}} \in \Omega | \hat{\boldsymbol{\Omega}} \cdot \hat{\boldsymbol{n}}_{\delta e} < 0\} \tag{2.52}$$

Furthermore we need to discern the edges and directions that are specified by the boundary conditions of the problem. BC in the following expression is the abbreviation for boundary condition, which can be any of the elements of $\{U, D\}$, upwind and Dirichlet boundary conditions respectively. There is an upwind 'boundary condition' as a result of applying the discontinuous Galerkin method is that all patches can be solved independently, therefore the upwind flux can be taken as a boundary condition for that patch.

$$\Omega_i^- = \{\hat{\boldsymbol{\Omega}} \in \Omega^- | BC = i\} \tag{2.53}$$
$$\partial V_i = \{\boldsymbol{r} \in \partial e^- | BC = i\} \tag{2.54}$$

The Galerkin procedure for spatial elements consists of multiplying the equation by a test function $v_{e,p}(\boldsymbol{r}, \hat{\boldsymbol{\Omega}})$ and integrating over the domain. The indices e and p are respectively the element and the patch index. This results in

$$\begin{aligned} -\int_{4\pi}\int_e \phi\hat{\boldsymbol{\Omega}}\cdot\nabla v d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} + \int_{4\pi}\int_e \sigma_t\phi^{int}v^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} - \int_{4\pi}\int_e \frac{\sigma_s}{4\pi}\Phi v^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} \\ + \int_{\Omega^+}\int_{\partial e^+}(\hat{\boldsymbol{\Omega}}\cdot\hat{\boldsymbol{n}})\phi^{int}v^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} + \int_{\Omega_U^-}\int_{\partial e^-\setminus\{\partial V_R\cup\partial V_D\}}(\hat{\boldsymbol{\Omega}}\cdot\hat{\boldsymbol{n}})\phi^{ext}v^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} \\ = \int_{4\pi}\int_e Sv^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} - \int_{\Omega_D^-}\int_{\partial e^-\cap\partial V_D}(\hat{\boldsymbol{\Omega}}\cdot\hat{\boldsymbol{n}})gv^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} \end{aligned} \tag{2.55}$$

By summing this equation over all elements we can identify a bilinear form and linear form such that we can write the discretized transport equation as

$$B(\phi, v) = l(v), \forall v \in V_h \tag{2.56}$$

where V_h is the space of all test functions. An explicit expression for B and l is given by

$$
\begin{aligned}
B(\phi, v) &= \sum_e \Big\{ -\int_{4\pi}\int_e \phi\hat{\boldsymbol{\Omega}}\cdot\nabla v d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} + \int_{4\pi}\int_e \sigma_t\phi^{int}v^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} - \int_{4\pi}\int_e \frac{\sigma_s}{4\pi}\Phi v^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} \\
&\quad + \int_{\Omega^+}\int_{\partial e^+} (\hat{\boldsymbol{\Omega}}\cdot\hat{\boldsymbol{n}})\phi^{int}v^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} \\
&\quad + \int_{\Omega_U^-}\int_{\partial e^-\setminus\{\partial V_R\cup\partial V_D\}} (\hat{\boldsymbol{\Omega}}\cdot\hat{\boldsymbol{n}})\phi^{ext}v^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} \qquad (2.57) \\
l(v) &= \sum_e \Big\{ \int_{4\pi}\int_e Sv^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}} - \int_{\Omega_D^-}\int_{\partial e^-\cap\partial V_D} (\hat{\boldsymbol{\Omega}}\cdot\hat{\boldsymbol{n}})gv^{int}d\boldsymbol{r}d\hat{\boldsymbol{\Omega}}\Big\} \qquad (2.58)
\end{aligned}
$$

We will now introduce patches with constant basis functions. The test function $v_{e,p}$ is therefore assumed to be of the form

$$v_{e,p}(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) = \phi_e(\boldsymbol{r})G_{e,p}(\hat{\boldsymbol{\Omega}}) \qquad (2.59)$$

The spatial part, $\phi_e(\boldsymbol{r})$, consists of linear functions. Each patch, $G_{e,p}(\hat{\boldsymbol{\Omega}})$, has a constant basis function. The linear form can then be written as:

$$l(v) = \sum_e \Big\{ \sum_p \int_e \Delta G_{e,p}S_{e,p}v_{e,p}^{int}d\boldsymbol{r} - \sum_{p\in\Omega_D^-}\int_{\partial e^-\cap\partial V_D} (\hat{\boldsymbol{\Omega}}_{e,p}\cdot\hat{\boldsymbol{n}})G_{e,p}v_{e,p}^{int}d\boldsymbol{r}\Big\} \qquad (2.60)$$

where $\hat{\boldsymbol{\Omega}}_{e,p}$ denotes $\int_{\Delta G_{e,p}}\hat{\boldsymbol{\Omega}}d\hat{\boldsymbol{\Omega}}$. With this notation no approximation is made, the integrals over the directions are exact. Since the patches have a constant basis function we can write the 4π integrals as sums with the size of the patch as weights, as stated earlier.

The bilinear form will become:

$$
\begin{aligned}
B(\phi, v) &= \sum_e \Big\{ -\sum_p \int_e \phi\hat{\boldsymbol{\Omega}}_{e,p}\cdot\nabla v_{e,p}d\boldsymbol{r} + \sum_p \int_e \Delta G_{e,p}\sigma_t\phi^{int}v_{e,p}^{int}d\boldsymbol{r} \\
&\quad - \sum_p \int_e \Delta G_{e,p}\frac{\sigma_s}{4\pi}\Phi v_{e,p}^{int}d\boldsymbol{r} \\
&\quad + \sum_{p\in\Omega^+}\int_{\partial e^+} (\hat{\boldsymbol{\Omega}}_{e,p}\cdot\hat{\boldsymbol{n}})\phi^{int}v_{e,p}^{int}d\boldsymbol{r} \\
&\quad + \sum_{p\in\Omega_U^-}\int_{\partial e^-\setminus\partial V_D} (\hat{\boldsymbol{\Omega}}_{e,p}\cdot\hat{\boldsymbol{n}})\phi^{ext}v_{e,p}^{int}d\boldsymbol{r} \qquad (2.61)
\end{aligned}
$$

A short derivation shows how we can write the error as a function of the linear and bilinear form, using respectively: linearity, dual problem, Galerkin orthogonality and consistency [5].

$$\begin{aligned}
\Delta J &= J(\phi) - J(\phi_h) && (2.62)\\
&= J(\phi - \phi_h) && (2.63)\\
&= B(\phi - \phi_h, \phi^*) && (2.64)\\
&= B(\phi - \phi_h, \phi^* - \phi_h^*) && (2.65)\\
&= l(\phi^* - \phi_h^*) - B(\phi_h, \phi^* - \phi_h^*) && (2.66)
\end{aligned}$$

Here ϕ_h is the computed solution and ϕ is the exact or reference solution. Since the exact solution is not always available one can use an approximation by using a solution on a very fine mesh.

Plugging in our expressions for l and B and subsequent partial integration yields

$$\begin{aligned}
\Delta J = &\sum_e \sum_p \Big\{ \int_e \Delta G_{e,p} S_{e,p} (\phi^* - \phi_h^*) d\boldsymbol{r} - \mathbf{1}_{p \in \Omega_D^-} \int_{\partial e^- \cap \partial V_D} (\hat{\boldsymbol{\Omega}}_{e,p} \cdot \hat{\boldsymbol{n}}) g_{e,p} (\phi^* - \phi_h^*) d\boldsymbol{r}\\
&+ \int_e \phi \hat{\boldsymbol{\Omega}}_{e,p} \cdot \nabla (\phi^* - \phi_h^*) d\boldsymbol{r} - \int_e \Delta G_{e,p} \sigma_t \phi_h^{int} (\phi^* - \phi_h^*) d\boldsymbol{r}\\
&+ \int_e \Delta G_{e,p} \frac{\sigma_s}{4\pi} \Phi (\phi^* - \phi_h^*) d\boldsymbol{r} - \mathbf{1}_{p \in \Omega^+} \int_{\partial e^+} (\hat{\boldsymbol{\Omega}}_{e,p} \cdot \hat{\boldsymbol{n}}) \phi_h^{int} (\phi^* - \phi_h^*) d\boldsymbol{r}\\
&- \mathbf{1}_{p \in \Omega_U^-} \int_{\partial e^- \setminus \partial V_D} (\hat{\boldsymbol{\Omega}}_{e,p} \cdot \hat{\boldsymbol{n}}) \phi_h^{ext} (\phi^* - \phi_h^*) d\boldsymbol{r} \qquad (2.67)
\end{aligned}$$

which is actually a sum over all patches. The sum is weighted by the patch size, which becomes clear when we interpret $\hat{\boldsymbol{\Omega}}_{e,p}$ as the 'average' angle times the size of the patch. The sum over all the patches then becomes

$$\begin{aligned}
\Delta J = &\sum_e \sum_p \Delta G_{e,p} \Big\{ \int_e S_{e,p} (\phi^* - \phi_h^*) d\boldsymbol{r} - \mathbf{1}_{p \in \Omega_D^-} \int_{\partial e^- \cap \partial V_D} (\hat{\boldsymbol{\Omega}}'_{e,p} \cdot \hat{\boldsymbol{n}}) g_{e,p} (\phi^* - \phi_h^*) d\boldsymbol{r}\\
&+ \int_e \phi^{int} \hat{\boldsymbol{\Omega}}'_{e,p} \cdot \nabla (\phi^* - \phi_h^*) d\boldsymbol{r} - \int_e \sigma_t \phi_h^{int} (\phi^* - \phi_h^*) d\boldsymbol{r} + \int_e \frac{\sigma_s}{4\pi} \Phi (\phi^* - \phi_h^*) d\boldsymbol{r}\\
&- \mathbf{1}_{p \in \Omega^+} \int_{\partial e^+} (\hat{\boldsymbol{\Omega}}'_{e,p} \cdot \hat{\boldsymbol{n}}) \phi_h^{int} (\phi^* - \phi_h^*) d\boldsymbol{r}\\
&- \mathbf{1}_{p \in \Omega_U^-} \int_{\partial e^- \setminus \partial V_D} (\hat{\boldsymbol{\Omega}}'_{e,p} \cdot \hat{\boldsymbol{n}}) \phi_h^{ext} (\phi^* - \phi_h^*) d\boldsymbol{r} \qquad (2.68)
\end{aligned}$$

It is interesting to compare the weights of the sum in this expression and the weights of a quadrature set that is used in the discrete ordinates method. In that method one chooses a set of directions and weights on which one demands the equation to hold. The integrals over the angular domain are then performed by computing a sum of weighted angular fluxes. In this case the angular integrals change into weighted sums naturally, where the weights are not free to chose, but are equal to the size of the patch.

Since the expression for ΔJ is a sum over all elements and all patches we can define an error contribution for each patch, call it $\eta_{e,p}$. The $\eta_{e,p}$'s will eventually tell us where to refine or coarsen the spatial or angular discretization. The total error estimate then looks like:

$$\Delta J = \sum_e \sum_p \Delta G_{e,p} \eta_{e,p} \tag{2.69}$$

after some manipulation and partial integration we see that $\eta_{e,p}$ is of the form:

$$\begin{aligned} \eta_{e,p} &= \int_e R_h(\phi^* - \phi_h^*) d\boldsymbol{r} + \mathbf{1}_{p \in \Omega_U^-} \int_{\partial e^- \setminus \{\partial V_D \cup \partial V_R\}} (\hat{\boldsymbol{\Omega}}'_{e,p} \cdot \hat{\boldsymbol{n}}) r_{h,U} (\phi^* - \phi_h^*)^{int} d\boldsymbol{r} \\ &\quad + \mathbf{1}_{p \in \Omega_D^-} \int_{\partial e^- \cap \partial V_D} (\hat{\boldsymbol{\Omega}}'_{e,p} \cdot \hat{\boldsymbol{n}}) r_{h,D} (\phi^* - \phi_h^*)^{int} d\boldsymbol{r} \end{aligned} \tag{2.70}$$

with:

$$R_h = S_{e,p} + \frac{\sigma_s}{4\pi} \Phi_h - \hat{\boldsymbol{\Omega}}'_{e,p} \cdot \nabla \phi_{h;e,p} - \sigma_t \phi_{h;e,p} \tag{2.71}$$

$$r_{h,U} = \phi_{h;e,p}^{int} - \phi_{h;e,p}^{ext} \tag{2.72}$$

$$r_{h,D} = \phi_{h;e,p}^{int} - g_{e,p} \tag{2.73}$$

The error contribution of a patch, $\eta_{e,p}$, is now an integral over the domain in phase space of that patch. The integrand is the product of the residual and the importance of the location in phase space, since R_h, $r_{h,U}$, $r_{h,D}$ and $r_{h,R}$ turn out to be the residual of the discrete transport equation. The importance is given by the adjoint solution. Finally the contribution $\eta_{e,p}$ is multiplied by the size of the patch $\Delta G_{e,p}$.

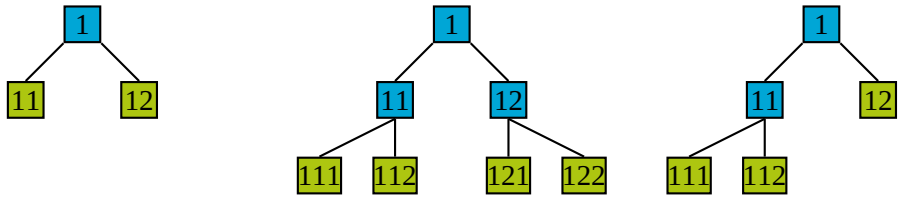
R_h is the spatial and the r's are the boundary residuals of the equation. $r_{h,D}$ and $r_{h,R}$ are the Dirichlet and reflective boundary residuals, while $r_{h,U}$ is the upwind residual. This upwind residual is non zero between elements (it is zero on the boundary of the domain). This residual is a result of the 'jumps' that are allowed in the solution in the discontinuous Galerkin method.

We can use this expression to estimate the error of the solution, without the need for an explicit expression of the exact solution. However, we can also formulate the criterion for refinement from this expression. We now have a contribution to the error of each patch $\Delta G_{e,p} \eta_{e,p}$. When these contributions are sorted we find which patches contribute most to the error. A fixed percentage of patches will be refined in each refinement iteration, which leads to the natural choice of refining the patches that contribute most to the error.

The only question that remains is how the exact adjoint is to be computed. Only in rare cases can this be done analytically, therefore we will make an approximation. Two slightly different approximations are used, the global and the local adjoint approximation. In the global adjoint approximation we compute the adjoint on one level deeper than the forward solution. This means that for each patch of the forward there are two adjoint patches. The adjoint at this level is computed as best as possible.

Turning to the local adjoint, we also use a refinement of one level deeper than the forward in this case. However, the adjoint is not calculated in the normal way, rather the adjoint solution on

the same level as the forward is first calculated. After that we can locally refine the adjoint one level, for each patch separately, and compute the solution on the deeper level. This is done with the 'old' scalar flux and angular flux values of neighbours, in other words the values of the adjoint solution on the same level as the forward. Both approximations are presented schematically in Figure 2.



(a) Forward (b) Global adjoint approximation (c) Local adjoint approximation

Figure 2: *Schematic representation of the two approximations of the exact adjoint used in the goal-oriented adaptive criterion. The global approximation consists of computing the adjoint on a level deeper, while the local approximation consists of local refinements (one such local refinements is presented).*

2.4.3 Traditional Adaptivity

Traditional refinement methods can be found in many areas of mathematics and physics, however the method described here does not necessarily apply to all these areas. It is however widely used in numerical neutron transport and other fields. All traditional methods are based on the same idea. A local refinement and solution is computed and compared to the original non-refined solution. The elements or patches that have the largest change in solution will keep the refinement, the other elements will go back to their original distribution.

We now need to quantify 'change in the solution'. The solution in this case is the angular flux ϕ and the change will be looked at in phase space. In other words, we will look at the square of the change in the angular flux integrated over phase space of a patch η, which reads

$$\begin{aligned} \eta &= \int_{\Delta x}\int_{G_i}(\phi_{h/2}-\phi_h)^2 d\hat{\mathbf{\Omega}}dx && (2.74)\\ &= \int_{G_i} d\hat{\mathbf{\Omega}}\int_{\Delta x}(\phi_{h/2}-\phi_h)^2 dx && (2.75)\\ &= 2\pi\Delta G_i\int_{\Delta x}(\phi_{h/2}-\phi_h)^2 dx && (2.76)\end{aligned}$$

Since the integrand of the angular integral is a constant function, we can pull it out. That leaves us with the spatial integral, which can be formulated in terms of the matrices in Equation 2.27:

$$\eta = \int_{\Delta x}\int_{G_i}(\phi_{h/2}-\phi_h)^2 d\hat{\mathbf{\Omega}}dx = 2\pi\Delta G_i(\boldsymbol{\phi}_{h/2}-\boldsymbol{\phi}_h)^T\boldsymbol{M}(\boldsymbol{\phi}_{h/2}-\boldsymbol{\phi}_h) \quad (2.77)$$

In this report $\phi_{h/2}$ is determined in the local approximation. A patch is locally refined and the new solution is computed on this patch. This means the upwind flux values that are used in the computation of this patch are of the coarse level and are not updated. Also the scalar flux Φ is not updated before computing the solution on the finer patches. After the change has been computed that patch is coarsened again, before going on to the next patch.

This method clearly tries to get an accurate solution of ϕ on the whole domain, it's goal is to let ϕ vary as little as possible. However, we are interested in an accurate calculation of the detector response. The goal of the traditional refinement strategy does not necessarily result in an accurate detector response. As opposed to traditional methods goal-oriented methods take into account the quantity one wants to determine accurately, in this case the detector response.

2.5 Overview of Algorithm

A schematic overview of the algorithm is presented in Figure 3. Before we will explain the overview of the algorithm a short explanation of the method used for solving the system of equations obtained from discretising the transport equation is given. This method is described extensively in Section 3.2, therefore we will only briefly explain why we have two loops in this overview. The outer iteration is the refinement iteration, which is discussed below. The inner iteration is the source iteration, where the transport equation is solved on a certain mesh. This iteration consists of splitting the matrix and applying a Jacobi iteration on these matrices.

We start with setting up the finite element method in an hierarchical data structure. The hierarchy can be used for easy reference when refining. After which we can start with the first refinement iteration, which itself starts with the first source iteration. These iterations are represented by respectively the outer and the inner box of dashed lines.

In the source iteration we start by updating the source term, that is the sum of the external source and the scatter terms. In the first iteration the scatter term simply adds nothing to this sum. We can then update the angular flux by solving the transport part of the transport equation. This results in the angular flux of one-time scattered neutrons. From this angular flux we can update the scatter term, after which we enter the second source iteration. This is repeated a fixed number of times, however, in the tests presented here it is made sure the source iteration has converged up to machine precision.

After the solution of the angular flux at the current refinement level is computed we can start the first refinement procedure. The first step is to calculate the error contributions of each patch and ordering this list of contributions in decreasing absolute contribution. When the contributions are added the error estimate for this iteration is obtained. The other step consists of refining the patches with the largest contributions, in the tests presented below we refine thirty per cent of all patches in each iteration.

The first refinement iteration is then complete. We can now repeat the whole process until the error in the detector response is as small as one wants it to be. The detector response can be computed at the end of the algorithm, when one is solely interested in an accurate detector response. For testing purposes we compute the detector response in each refinement iteration.

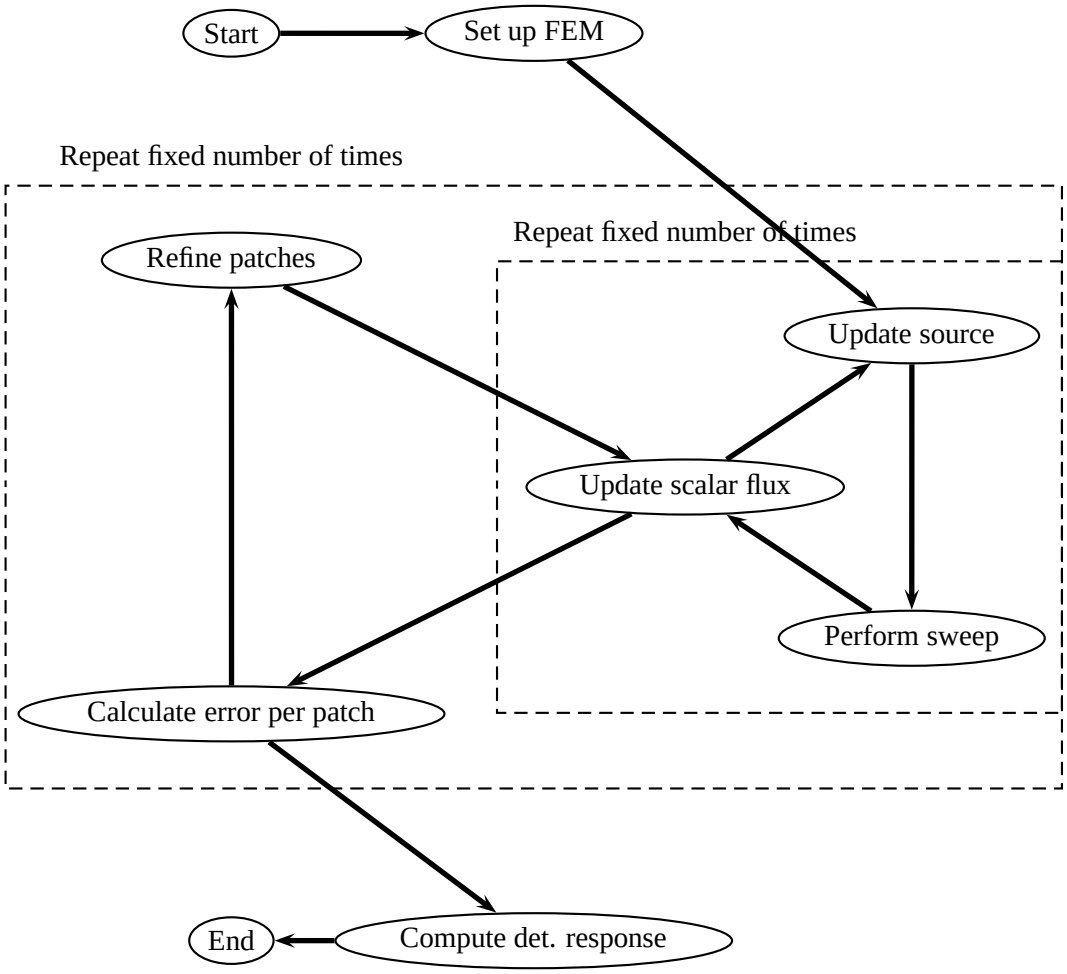


Figure 3: *Schematic overview of the adaptive algorithm.*

2.6 One-Dimensional Results

In this section the results of the various adaptive algorithms for several test cases are presented. Eight cases are used to illustrate the performance of the algorithm, these test cases can be found in Appendix A. Each of the cases will test different aspects of the adaptive methods. In all test cases the adaptive algorithms refine 30 per cent of the patches in each refinement iteration, unless stated otherwise.

For all test cases a figure with the error plotted against the total number of patches. This shows how the refinement in each step improves the solution. In this way different adaptive methods can be compared on performance. The discrete ordinates method is also included in these plots, although it is not an adaptive method. However, one can compare the cost in number of unknowns

of the discrete ordinates method and the adaptive methods.

The quality of the error estimator for the two goal-oriented methods is also examined. To this end we compare the estimated error with the reference error, which is computed at a much deeper level. We examine the quality to find out whether we can use the error estimator as a trustworthy representation of the error. This would mean that in future work the reference error does not necessarily have to be computed.

A final comparison between the adaptive methods is an investigation into the effect of the refinement ratio, that is the percentage of patches that is refined in each refinement iteration. The adaptive algorithm is run several times with different percentages. The error versus the total number of patches for all methods is then plotted in one figure, which shows the most effective refinement ratio.

The first test cases have a homogeneous slab geometry, with different materials. The next test cases have a separate source and detector in a homogeneous material. Thereafter the source detector test case is extended by placing a shielding region in the middle of the domain, between the source and detector.

A final remark has to be made on all plots that are shown in this section, the legend holds names for the different method that are presented in this work. 'Uniform' is the discontinuous Galerkin method with uniform refinement, i.e. in each refinement iteration all patches are refined. 'Traditional' refers to the traditional refinement criterion that can be used with the discontinuous Galerkin method. The two goal-oriented adaptive methods are referred to as 'full adjoint' and 'local adjoint'. The full adjoint is the criterion that bases its decision for which patches to refine on the adjoint solution of patches that are one level deeper (they are refined once more) than the patches of the forward solution. The other criterion bases its decision on a local refinement of the adjoint solution and is therefore called local adjoint. We also have the discrete ordinates method which is referred to as 'discrete ordinates'. Finally a line that represents second order convergence is plotted for convenience, this line is referred to as '2nd order'.

2.6.1 Homogeneous Slab (cases A and B)

The first test case is the homogeneous slab, both optically thick and thin. The exact specifications of these test cases are presented in Appendix A. In Figure 4(a) one can find the plot of the detector response error versus the total number of patches of test case A, the thick scattering slab. In this plot we note that convergence is eventually second order for all methods that are presented. The error with uniform refinement of patches decreases constantly, while the error of all other methods decreases faster in the beginning, i.e. with few patches. Therefore the traditional, full and local adjoint and discrete ordinates methods eventually have an advantage of an estimated factor of two, measured in the number of patches needed to get a certain error. The similarity in error decrease of the different methods is probably due to the homogeneity of the problem.

Turning to the plot of detector response error decrease versus the total number of patches in case B, Figure 4(b), we see that all methods again perform somewhat similar. This time the uniform method (all patches are refined in each refinement iteration) performs slightly better than the other methods. The error decrease for all methods is again about second order.

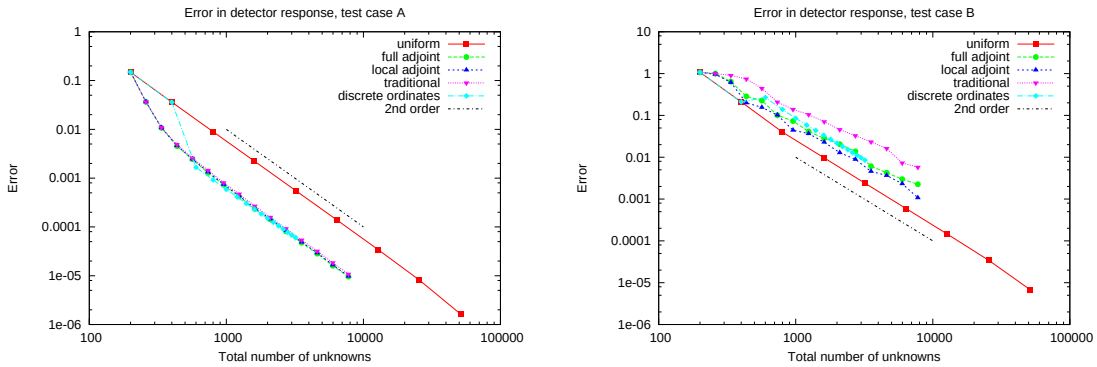
In both cases the adaptive algorithms perform about the same, also the Discrete Ordinates method performs about the same as the adaptive algorithms. The uniform refinement performs,

however, different in case A and B. In case A the adaptive algorithms perform slightly better, while in case B the uniform refinement performs slightly better. An answer to this can be found in the spatial patch distribution, which is shown in Figure 5. In these plots the number of patches in an element is plotted against the x position of that element.

Case B has quite a uniform patch distribution in space. Therefore one would expect the uniform and adaptive methods to work similarly. However in case A the patch distribution is far from uniform, more patches are used near the edge of the domain. Near the edges the angular flux is more difficult to compute, because of the leakage. In the middle of the domain there is little effect of the leakage, because the material is optically thick. Therefore the adaptive algorithms can provide a better solution.

Another remarkable result is the steps in the patch distribution of test case A. These are an artefact of the refinement algorithm, when a smaller fraction of patches is refined these steps disappear and the distribution becomes exponential. The exponential behaviour might be due to the fact that effects of leakage decrease exponentially when propagating through the domain, because a ray of neutrons also decreases exponentially.

The lack of difference between the three adaptive methods can be explained by the homogeneity of the problem. Since the volumetric detector is in the whole domain there is little difference between the traditional and goal oriented criteria. One could say that a refinement that changes the solution of the angular flux ϕ most will probably also result in a better detector response and vice versa. Therefore we will turn to other test cases.



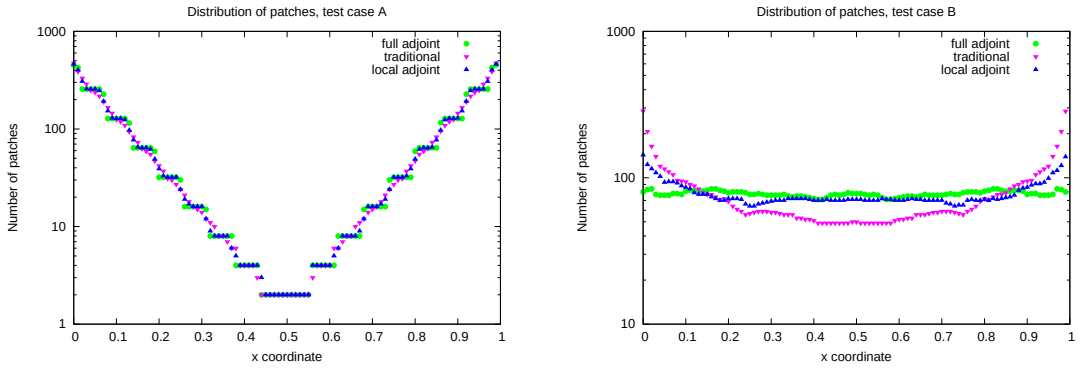
(a) *Test case A, thick highly scattering medium.*

(b) *Test case B, thin scattering medium.*

Figure 4: *Error in the detector response of the two homogeneous slab test cases. In the optically thick case all methods perform better than the uniform method, because the optimal distribution of patches is not flat. However in the optically thin case the optimal distribution is almost flat, therefore the uniform refinement works well.*

2.6.2 Separate Source Setector (cases C, D, E and F)

In these test cases a geometry with a separate source and detector for different materials is presented. The four materials are thick scattering, thin scattering, thin absorbing and a material with



(a) *Test case A, thick highly scattering medium.*

(b) *Test case B, thin scattering medium.*

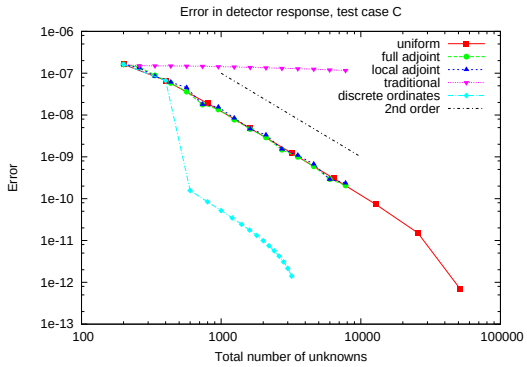
Figure 5: *Spatial patch distribution, in one spatial element all patches are counted and plotted at its position. The steps in the optically thick case disappear when a smaller refinement ratio is used. The optimal distribution of patches in the optically thin case is almost uniform.*

medium thickness and some scattering. For each of the test cases the material properties are listed in Appendix A.

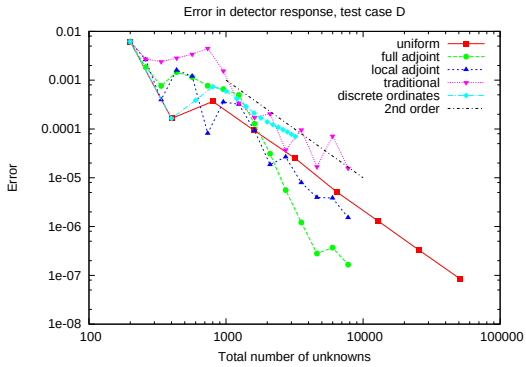
Figure 6 shows the error in the detector response versus the total number of patches of all source detector test cases. We see for all four test cases the same second order convergence of the uniform method. For test cases C, E and F we see somewhat the same behaviour. The discrete ordinates method outperforms all other methods, while the traditional refinement criterion with discontinuous Galerkin performs the worst of all, since there is almost no error decrease. The two goal-oriented adaptive methods, full and local adjoint, converge second order and perform at least as good as the uniform method. In case D the full adjoint goal-oriented method performs best, while all other methods perform comparably.

An explanation for this behaviour can be found looking at the angular flux profile. In the thick and diffusive cases the flux profile is almost linear while in the absorption case it is exponential. Since we're using constant basic functions (patches) different behaviour can be expected when the solution has a different angular flux profile. A linear function is harder to approximate with constant patches, as it needs a fine representation for all directions. An accurate approximation of exponential functions with constant functions only needs a fine representation where the derivative of the exponential function is largest.

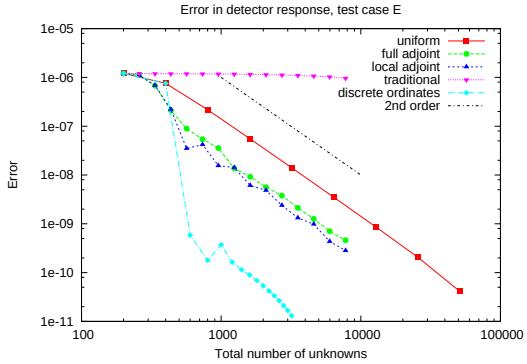
When comparing the uniform and adaptive methods one finds that the goal-oriented adaptive methods are at least as good as the uniform method. The traditional adaptive method does a very poor job. These differences between the methods can be explained by the patch distributions, which are shown in Figure 7. In this figure we will take a closer look at the spatial patch distribution of cases C and E. In case C we saw that the goal-orientated methods have an equal error reduction as the uniform method, which can be explained by the diffusivity of the problem. Constant patches cannot approximate diffusive problems well, because the angular flux profile is linear. Since we need a fine mesh to approximate a linear function by constant basis functions, this means the whole domain of the problem will be refined. Figure 7(a) shows the flat spatial distribu-



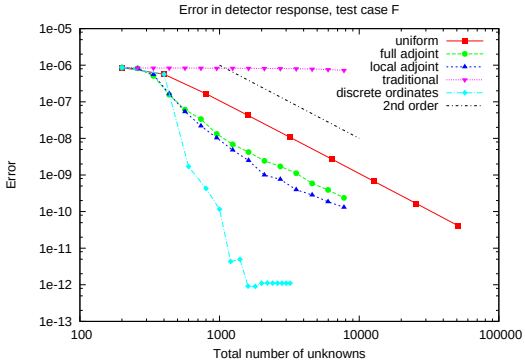
(a) *Test case C, thick highly scattering medium.*



(b) *Test case D, thin scattering medium.*



(c) *Test case E, absorbing medium.*



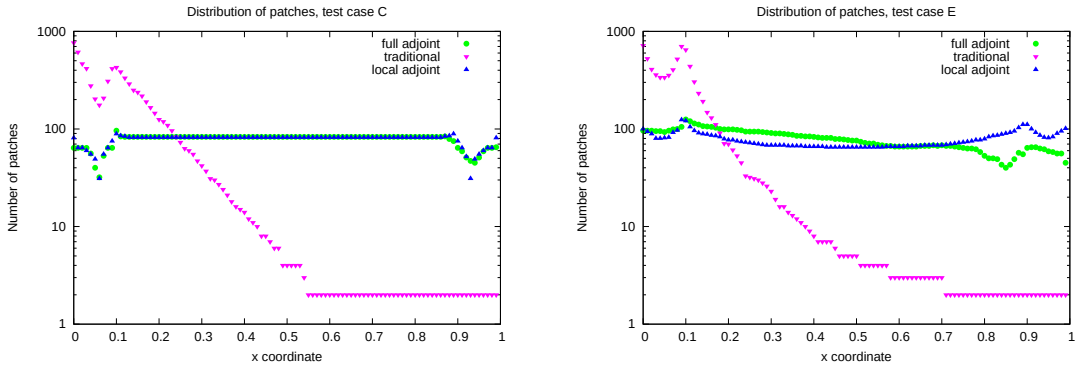
(d) *Test case F, purely absorbing medium.*

Figure 6: *Error in the detector response of all source detector test cases.*

tion of the two goal-oriented adaptive methods. This figure also shows why the traditional method performs so poorly. Most refinement is on the left hand side of the problem, while the detector is on the right hand side of the problem, which results in a bad detector response. The traditional criterion refines patches of which the solution changes most when they are refined, which will be around the source region in the domain and not around the detector region, as the solution is much larger in the source region.

The patch distribution of test case E shows why the goal orientated adaptive methods outperform the uniform method. Even though the spatial distribution is still quite flat, more refinement took place on the border of the source region and the detector region. Since this problem is less scattering the patch distribution is less flat. Also in this case the traditional method refines locally around the source and not at all around the detector.

It is also worth looking at the node wise root mean square error (rms error) of the flux. Since this error measure looks at the whole domain of the problem instead of just the detector region. Figure 8 shows a plot with the rms error versus the total number of patches. The traditional adaptive method provides us with the smallest error of all methods and discrete while the goal oriented adaptive methods show almost no decrease in error. This behaviour can be expected since



(a) *Test case C, thick highly scattering medium.*

(b) *Test case E, absorbing medium.*

Figure 7: *Spatial patch distribution, in one spatial element all patches are counted and plotted at its position. The traditional adaptive method mostly refines around the source, not around the detector. The absorbing medium has a larger refinement near source and detector for the goal oriented adaptive methods, while the thick case has an almost flat distribution. This is needed for an accurate representation of a diffusive problem.*

the goal of the goal-oriented methods is to get the detector response as accurate as possible, which not necessarily needs an accurate description of the solution in the whole domain. We furthermore note that the uniform method converges about second order and that the discrete ordinates method performs similar to the uniform method.

2.6.3 Quality of Error Estimator

Besides using the error estimate as a criterion for refinement we can also use it as an estimator for the error in the detector response. In many cases the exact error is not available, as we do not have an exact solution to the neutron transport problem. To reliably use the estimator as error indicator we first need to test its performance. This test consists of examining the error ratio, that is the ratio between the estimated and exact error. Since the exact error is not available we will use a reference error, which is computed on a much finer patch distribution. In this section we will look at three test cases (A, E and G) where the estimator behaves differently.

In each of the figures presented here three data sets are plotted, ‘uniform’, ‘full adjoint’ and ‘local adjoint’. We will treat them in reverse order, starting with the local adjoint. The local adjoint criterion consists of the goal-oriented error estimator (see Section 2.4.2), where the exact adjoint solution is approximated by a local refinement of the associated adjoint patch. That is, to compute the error contribution of a certain patch in the forward patch distribution, we refine the associated adjoint patch once locally, resulting in a representation on one level deeper. Besides using the error estimator the refinement criterion based on the error contribution is also used. Turning to the full adjoint criterion, this method uses the same criterion as the local adjoint, only the exact adjoint solution is now approximated by computing the adjoint solution on a patch distribution that is refined to one level deeper than the forward patch distribution. The last data set is obtained by using the full adjoint error estimator on a uniform refined patch distribution. This means in

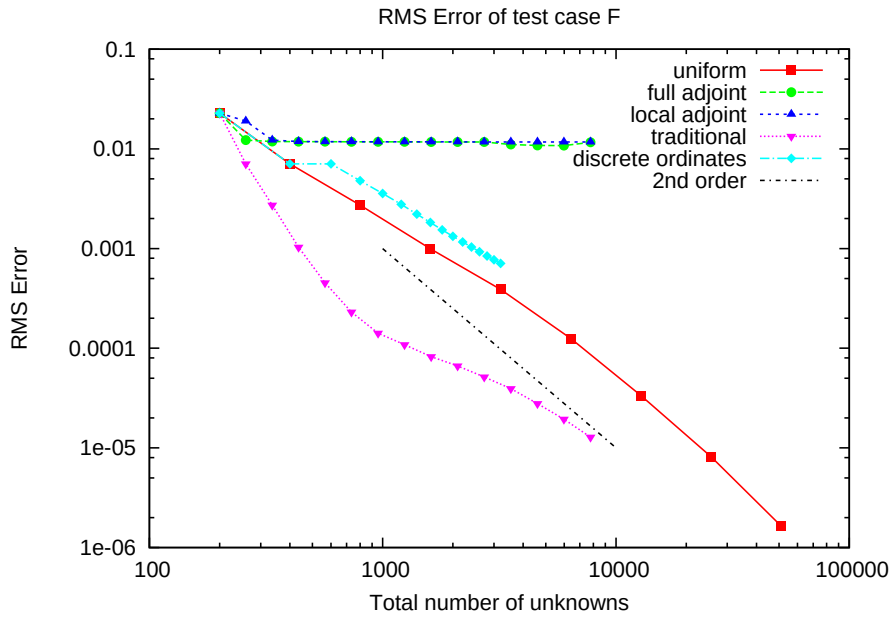


Figure 8: *Node wise root mean square error of scalar flux of test case F. In the root mean square error measure we see the traditional adaptive method being more effective than the goal oriented adaptive methods. The goal of the goal oriented adaptive methods, an accurate detector response, is therefore not the same as an accurate overall solution.*

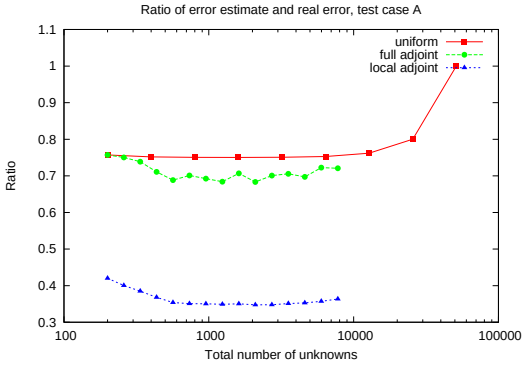
each refinement iteration all patches are refined, so the adaptive criterion for refinement is not used. However, we can test the error estimator that is obtained from the adaptive criterion using this patch distribution.

The error ratio's for the thick homogeneous slab, test case A, are shown in Figure 9(a), where the ratio of the error estimator and the reference error is plotted against the total number of patches. We expect the error estimator to asymptotically go to the reference error, this means the ratio should tend to unity. We can see that for case A the ratio tends to unity for the uniform method, however not asymptotically. It is expected that this will happen when linear patches are used, as constant patches cannot approximate linear flux profiles well. The same criterion but with adaptive refinement, the full adjoint case, has somewhat the same behaviour. It has some wiggles, but the trend is the same as the uniform case. In contrast to this, the local adjoint case has a very different behaviour. The ratio lies significantly below the reference error and it even becomes a worse estimator when more patches are used. For this case it cannot be used as an error indicator.

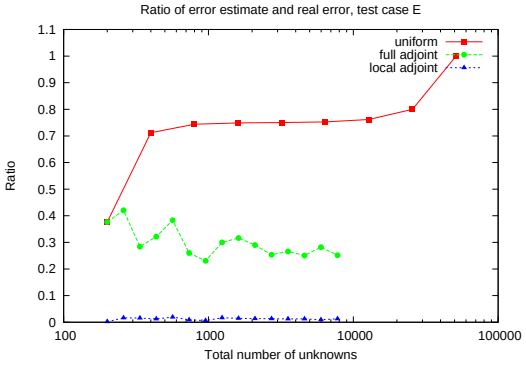
Turning to test case E, the absorption source detector problem, we see in Figure 9(b) that the full adjoint case does not always provide an accurate error estimator. The error ratio does not tend to unity as patches are refined, which means we cannot use it as a reliable indicator for the error in this case. The uniform refined method still tends to unity, but again not asymptotically. The local adjoint estimator performs even worse in this case, compared to case A. We can certainly not use this as an indicator for the error.

The final case in this section is test case G, the thick boundary detector case. The error ratio's are shown in Figure 9(c). In this case the full adjoint the uniform methods give almost the same error estimator, which tends to unity. However, the local adjoint estimator is still off. In all cases the local adjoint estimator seems to provide us with an underestimate of the error, which cannot be used as an indicator for the error.

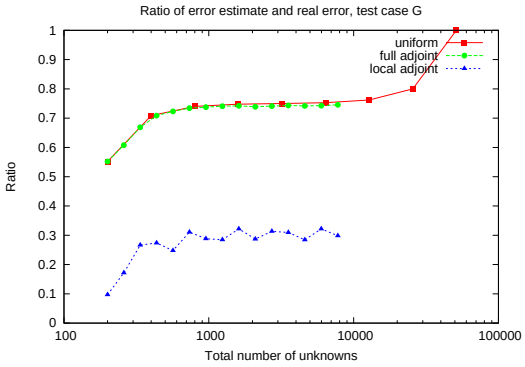
An interesting fact is that the local and full adjoint error estimators do not agree on the error estimator, the full adjoint provides us with a reasonably good estimator, while the local adjoint estimator is too much off. The two methods do, however, refine almost the same patches, resulting in similar patch distributions. This can be seen in the patch distribution plots shown earlier. This means that when the error estimator is not important, the local adjoint method, which is cheaper than the full adjoint method, can be used to decide which patches to refine.



(a) *Ratio of error estimator and reference error of test case A, the thick scattering homogeneous slab problem. The full adjoint gives an accurate error when uniformly refined, the estimator is also quite accurate when refined adaptively. However, the local adjoint estimator gives an inaccurate estimator.*



(b) *Ratio of error estimator and reference error of test case E, the absorption source volumetric detector problem. In this case the full adjoint estimator gives an accurate estimator when refinement takes place uniformly, however not when the refinement takes place adaptively. The local adjoint estimator does not provide an accurate estimator.*



(c) Ratio of error estimator and reference error of test case G, the thick source boundary detector problem. The full adjoint refinement criterion gives us an accurate estimator, in contrast to the local adjoint estimator.

Figure 9: *Ratio's of error estimators and reference error. The reference error is computed using a very deep refinement, much deeper than where the tests took place.*

3 Analysis of the Adaptive Algorithm

In this section we will investigate on the algorithm used to solve the transport equation. This is mainly to describe the algorithm more accurately and also explain some of the results we have seen for the constant patches. There are three parts in this section. The first part is on constructing the full transport operator matrix. We will need this full form in order to investigate the algorithm further. The full matrix is never constructed during the tests with the constant patches.

The second part involves the method of solving the transport equation itself. Two iterations are used to solve the system, the source iteration and the sweep iteration. These iterations are used in a nested way, which is described in mathematical terms in this part. Furthermore we look at the question of convergence of the iterations in this part.

In the algorithm the adjoint operator of the transport equation is used, that is, the adjoint of the continuous operator. We can also define a discrete adjoint operator and examine how it relates to the discretised continous adjoint operator. A derivation and comparison between the two ways of deriving a discrete adjoint problem is presented in the third part of this section.

Finally we have a part considering the behaviour of the local and global error estimator we saw in the previous section. Both the global and local error estimator refine patches in a way that is to be expected and seems to be, in general, the most effective refinement. However the error estimate itself differs between the methods. The global error estimator provides us with a fairly accurate estimate of the error, while the estimate of the local adjoint is off by too much to be reliable. Why this is possible with the error estimation as described in the previous section is the subject of the last part.

3.1 Transport Equation Matrix

The goal of this part is to construct the full transport operator matrix, when discretised using the discontinuous Galerkin method with linear basis functions in the spatial domain and constant basis functions in the angular domain. We will start with deriving the shape of the matrix for a uniformly discretized problem term by term. Let us assume there are E elements, in the spatial direction, and P patches on each element.

Starting with the removal term $\sigma_t\phi$, we see that this term adds a block matrix for each cell in the spatial-angular domain. Since applying the Galerkin procedure to this term results in

$$\int_p \int_e \sigma_t \phi_{e,p} \gamma_e G_{e,p} \sum_{e,p} \gamma_e G_{e,p} dx d\mu = \begin{bmatrix} \gamma_{e_1}\gamma_{e_1} G^2_{e,p} & \gamma_{e_1}\gamma_{e_2} G^2_{e,p} \\ \gamma_{e_2}\gamma_{e_2} G^2_{e,p} & \gamma_{e_2}\gamma_{e_2} G^2_{e,p} \end{bmatrix} \begin{bmatrix} \phi_{e1,p} \\ \phi_{e2,p} \end{bmatrix} \tag{3.1}$$

and the spatial basis functions γ are linear and the angular basis functions $G_{e,p}$ are constant, we get a block diagonal matrix with the following entries

$$\begin{bmatrix} \gamma_{e_1}\gamma_{e_1} G^2_{e,p} & \gamma_{e_1}\gamma_{e_2} G^2_{e,p} \\ \gamma_{e_2}\gamma_{e_2} G^2_{e,p} & \gamma_{e_2}\gamma_{e_2} G^2_{e,p} \end{bmatrix} \begin{bmatrix} \phi_{e1,p} \\ \phi_{e2,p} \end{bmatrix} = \begin{bmatrix} \frac{1}{3}\Delta x \Delta\mu & \frac{1}{6}\Delta x \Delta\mu \\ \frac{1}{6}\Delta x \Delta\mu & \frac{1}{3}\Delta x \Delta\mu \end{bmatrix} \begin{bmatrix} \phi_{e1,p} \\ \phi_{e2,p} \end{bmatrix} \tag{3.2}$$

This two by two matrix block is added to the diagonal of the large matrix L_h, so there are in total $E \cdot P$ of these blocks.

The scatter term $\frac{\sigma_s}{4\pi}\Phi$ is a little more involved, since this couples all patches in an element. Φ is an integral of a piece-wise constant function and easily computed. For a uniformly refined discretisation with P patches on an element, this results in the following matrix for one element

$$\begin{aligned}
&\int_p\int_e \frac{\sigma_s}{4\pi}\Phi_e \sum_{e,p}\gamma_e G_{e,p} dx d\mu \\
&= \begin{bmatrix} \gamma_{e_1}\gamma_{e_1}G_{e,1} & \gamma_{e_1}\gamma_{e_2}G_{e,1} & \gamma_{e_1}\gamma_{e_1}G_{e,2} & \cdots & \gamma_{e_1}\gamma_{e_2}G_{e,P} \\ \gamma_{e_2}\gamma_{e_1}G_{e,1} & \gamma_{e_2}\gamma_{e_2}G_{e,1} & \gamma_{e_2}\gamma_{e_1}G_{e,2} & \cdots & \gamma_{e_2}\gamma_{e_2}G_{e,P} \\ \vdots & & \ddots & & \vdots \\ \gamma_{e_2}\gamma_{e_1}G_{e,1} & \gamma_{e_2}\gamma_{e_2}G_{e,1} & \gamma_{e_2}\gamma_{e_1}G_{e,2} & \cdots & \gamma_{e_2}\gamma_{e_2}G_{e,P} \end{bmatrix} \begin{bmatrix} \Phi_{e1,p} \\ \Phi_{e2,p} \end{bmatrix} \qquad (3.3) \\
&= \begin{bmatrix} \frac{1}{3}\Delta x\Delta\mu & \frac{1}{6}\Delta x\Delta\mu & \frac{1}{3}\Delta x\Delta\mu & \cdots & \frac{1}{6}\Delta x\Delta\mu \\ \frac{1}{6}\Delta x\Delta\mu & \frac{1}{3}\Delta x\Delta\mu & \frac{1}{6}\Delta x\Delta\mu & \cdots & \frac{1}{3}\Delta x\Delta\mu \\ \vdots & & \ddots & & \vdots \\ \frac{1}{6}\Delta x\Delta\mu & \frac{1}{3}\Delta x\Delta\mu & \frac{1}{6}\Delta x\Delta\mu & \cdots & \frac{1}{3}\Delta x\Delta\mu \end{bmatrix} \begin{bmatrix} \Phi_{e1,p} \\ \Phi_{e2,p} \end{bmatrix} \qquad (3.4)
\end{aligned}$$

This block matrix is also added to the diagonal of the large matrix L_h, however since this block is $2P$ by $2P$, it is added E times.

The streaming term was divided into two terms, a volumetric and a boundary term. We will first discuss the volumetric term, which is again a two by two block matrix that will be added to the diagonal of the large matrix L_h. This block matrix is derived as follows

$$\begin{aligned}
\int_p\int_e \mu\frac{\partial\phi_{e,p}\gamma_e G_{e,p}}{\partial x}\sum_{e,p}\gamma_e G_{e,p} dx d\mu &= \begin{bmatrix} \gamma'_{e_1}\gamma_{e_1}\langle G^2_{e,p}\rangle & \gamma'_{e_1}\gamma_{e_2}\langle G^2_{e,p}\rangle \\ \gamma'_{e_2}\gamma_{e_1}\langle G^2_{e,p}\rangle & \gamma'_{e_2}\gamma_{e_2}\langle G^2_{e,p}\rangle \end{bmatrix} \begin{bmatrix} \phi_{e1,p} \\ \phi_{e2,p} \end{bmatrix} \qquad (3.5) \\
&= \begin{bmatrix} -\frac{1}{2}\Delta\mu & -\frac{1}{2}\Delta\mu \\ \frac{1}{2}\Delta\mu & \frac{1}{2}\Delta\mu \end{bmatrix} \begin{bmatrix} \phi_{e1,p} \\ \phi_{e2,p} \end{bmatrix} \qquad (3.6)
\end{aligned}$$

Now we have the most involved term left, the boundary streaming term. This term will not produce block diagonal matrices, instead it will couple patches from different elements to each other. We have a different situation for left and right going patches, that is, for patches where either $\int_p d\mu > 0$ or $\int_p d\mu < 0$. In general we have

$$\begin{aligned}
\int_p \mu\gamma_e G_{e,p}\sum_{e,p}\frac{\partial\gamma_e}{\partial x}d\mu = &\begin{bmatrix} \gamma'_{e_1}\gamma_{e_1}\langle G^2_{e,p}\rangle & 0 \\ 0 & 0 \end{bmatrix}_{x=e1} \begin{bmatrix} \phi_{e1,p} \\ \phi_{e2,p} \end{bmatrix}_{x=e1} \\
&+ \begin{bmatrix} 0 & 0 \\ 0 & \gamma'_{e_2}\gamma_{e_2}\langle G^2_{e,p}\rangle \end{bmatrix}_{x=e2} \begin{bmatrix} \phi_{e1,p} \\ \phi_{e2,p} \end{bmatrix}_{x=e2} \qquad (3.7)
\end{aligned}$$

At a boundary we chose the angular flux values ϕ upwind when the patch is streaming into the element and we chose the flux values of the element itself when the patch is streaming out of the element. This gives us for $\int_p d\mu > 0$

$$\int_p \mu\gamma_e G_{e,p} \sum_{e,p} \frac{\partial\gamma_e}{\partial x} d\mu = \begin{bmatrix} \langle\mu\rangle & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \phi_{e1-1,p} \\ \phi_{e2-1,p} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \langle\mu\rangle \end{bmatrix} \begin{bmatrix} \phi_{e1,p} \\ \phi_{e2,p} \end{bmatrix} \tag{3.8}$$

Analogously we treat patches where $\int_p d\mu < 0$, only now the fluxes at the other side of the element are taken from the neighbouring element.

Adding all these terms, we can construct the full matrix as is solved in the code. Note that in the code the full matrix is never constructed, for large problems this is very costly computational and memory wise. In Figure 10 the non-zero elements of L_h are shown.

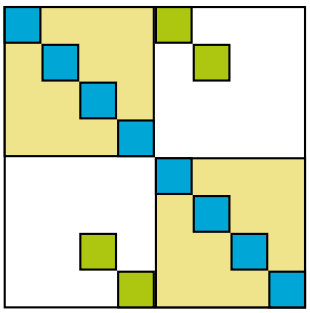


Figure 10: *Structure of the matrix L_h. The blue blocks represent the blocks on the diagonal which arise from the removal and volumetric streaming terms. The boundary streaming terms result in the green blocks. The large yellow blocks arise from the scatter term. This is the matrix corresponding to a system with two spatial elements with each four patches.*

For non-uniform meshes the structure of the matrix will be the same, however since the number of patches per spatial element is variable the yellow blocks in Figure 10 have variable size. The elements that arise from the boundary streaming term, green in that figure, will correspondingly shift. This does not alter the iteration with which the system is solved, as can be seen in the next section.

3.2 Iterative Solver

Now we will focus on the method of solving the system of equations that was obtained above. Since the matrix is sparse, especially for large systems, a lot of computational and memory costs can be avoided by using methods that do not need the explicit matrix. To achieve this two iterations are used in a nested way, first the source iteration is applied and then the sweep iteration.

Source iteration is a method that is also inspired by physical reasons, since it is an iteration where neutrons are scattered more and more with each iteration, more on this below. The sweep iteration is a way of solving the angular flux with the dependencies on upwind angular fluxes. As it turns out the sweep iteration can also be understood as a reordering of the matrix.

3.2.1 Source Iteration

We will start with applying source iteration to the discretised system of linear equations, which will turn out to be an implementation of a block Jacobi iteration. However, the method also has a physical explanation, for practical reasons we will use the continuous equation to explain the physical aspect of source iteration, but the reasoning holds for the discrete domain.

The two sides of the transport equation as in Equation 2.2 describe a different process. The left hand side of the transport equation,

$$\hat{\boldsymbol{\Omega}} \cdot \nabla \phi(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) + \sigma_t(\boldsymbol{r}) \phi(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) = \frac{\sigma_s}{4\pi} \Phi(\boldsymbol{r}) + s(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) \tag{3.9}$$

describes streaming and removal of neutrons, while the right hand side is a source of a particular point in the spatial-angular space. This source side consists of two terms, the external source s and the scatter source $\frac{\sigma_s}{4\pi}\Phi(\boldsymbol{r})$. This term can be considered as a source since it 'redistributes' neutrons from all directions to the point in spatial-angular space we are considering, in other words, for this point the neutrons scattering to this point are a source.

In many numerical implementations this term is approximated with the source iteration. In this iteration we start with the transport equation without any scatter source,

$$\hat{\boldsymbol{\Omega}} \cdot \nabla \phi^0(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) + \sigma_t(\boldsymbol{r}) \phi^0(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) = s(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) \tag{3.10}$$

where we solve for ϕ^0, the zero-times scattered flux. The solution ϕ^0 we found is an approximation to the solution of the transport equation where all scattering is ignored. We can use this 'guess' to obtain a better approximation for the angular flux ϕ of Equation 3.9 by substituting ϕ^0 in the scatter term and solving the equation

$$\hat{\boldsymbol{\Omega}} \cdot \nabla \phi^1(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) + \sigma_t(\boldsymbol{r}) \phi^1(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) = \frac{\sigma_s}{4\pi} \int_{4\pi} \phi^0(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) d\hat{\boldsymbol{\Omega}} + s(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) \tag{3.11}$$

We now obtained a 'one scattered' angular flux ϕ^1, which is a better approximation to the angular flux of the transport equation ϕ. In ϕ^1 we now take into account all neutrons that have scattered once, but we disregard all neutrons that have scattered more times. This procedure can be repeated until the approximation ϕ^l is close enough to ϕ. In general this iteration can be written as

$$\hat{\boldsymbol{\Omega}} \cdot \nabla \phi^{l+1}(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) + \sigma_t(\boldsymbol{r}) \phi^{l+1}(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) = \frac{\sigma_s}{4\pi} \int_{4\pi} \phi^l(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) d\hat{\boldsymbol{\Omega}} + s(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) \tag{3.12}$$

We will now continue with the numerical description of this iterative method, which turns out to be a block Jacobi method. Some words on convergence of the complete iterative method will be spent at the end of this section. From now on we will again consider the discrete system of linear equations $L_h \phi = s$.

To apply the block Jacobi method we split the matrix in two matrices,

$$L_h = T_h + S_h \tag{3.13}$$

where T_h is the operator that handles transport and removal, while S_h is the scatter operator. T_h is a block diagonal matrix with bands. The blocks are two by two, so the blocks describe the transport

equation within one cell in the spatial-angular space. Boundary streaming terms, originating in Equation 3.7 form the bands of the matrix. The entries in S_h all follow from the scatter term in Equation 3.4, so this matrix is also block diagonal, but with blocks of $2P \times 2P$.

Although the matrix T_h is not block diagonal we will apply the Jacobi iteration method to this splitting of L_h. This can be done since we will be able to solve the system obtained from the Jacobi iteration later on. Our iteration now looks like

$$\phi^{l+1} = T_h^{-1} S_h \phi^l + T_h^{-1} s \tag{3.14}$$

3.2.2 Sweep Iteration

Now we can turn to the question how to solve the transport part of the equation, that is, a system of the form $T_h x = b$. We can utilise the special form of T_h to solve this with a direct method. To this end we split the matrix T_h into two matrices and thus, effectively, splitting the matrix L_h into three matrices. This results in the following matrices

$$T_h = B_h + D_h \tag{3.15}$$
$$L_h = B_h + D_h + S_h \tag{3.16}$$

where B_h is the block diagonal matrix in T_h and D_h is the matrix containing all off diagonal blocks of T_h. Before writing the iteration as an expression of these matrices, we will take a look at the inner parts of these matrices.

First we take a look at matrix B_h and let us assume the matrix is $n \times n$. Since this is a block diagonal matrix with blocks of two by two we can solve a system $B_h x = b$ by solving $\frac{n}{2}$ decoupled systems of two by two. We would like to be able to solve the system $(B_h + D_h)x = b$ in this way, since it will reduce the computational cost of an implementation of this system.

The diagonal blocks of D_h can be put in the right hand side of the two by two systems if the column entry of x is known, or, for block (i, i) in B_h and block (i, j) in D_h we can put $D_{h;i,j}$ in the right hand side of

$$(B_h + D_h)x = b \Longrightarrow B_{h;i,i} x_i = b_{i,i} - D_{h;i,j} x_j \tag{3.17}$$

when x_j is known. In Figure 11 the sweeping iteration is performed on the matrix $B_h + D_h$. In step (2) one can see how Equation 3.17 is constructed for each block in B_h. As shown in step (1) there are two blocks without a dependence of other blocks in B_h, these will have to be solved first. After the first block is solved, the dependence to another block in B_h is known, therefore this block can be directly solved. This procedure will have to be repeated to solve every other block on the diagonal, after which the same procedure can be followed, but going back up the matrix.

There are two ways of interpreting this method, a physical and mathematical one. To start with the mathematical interpretation, we note that this method solves the entries of the vector x in a certain order. When rearranging the matrix in this order, we see that the matrix is actually a triangular matrix, either upper or lower. Since a triangular matrix is directly solvable, this method is a direct method.

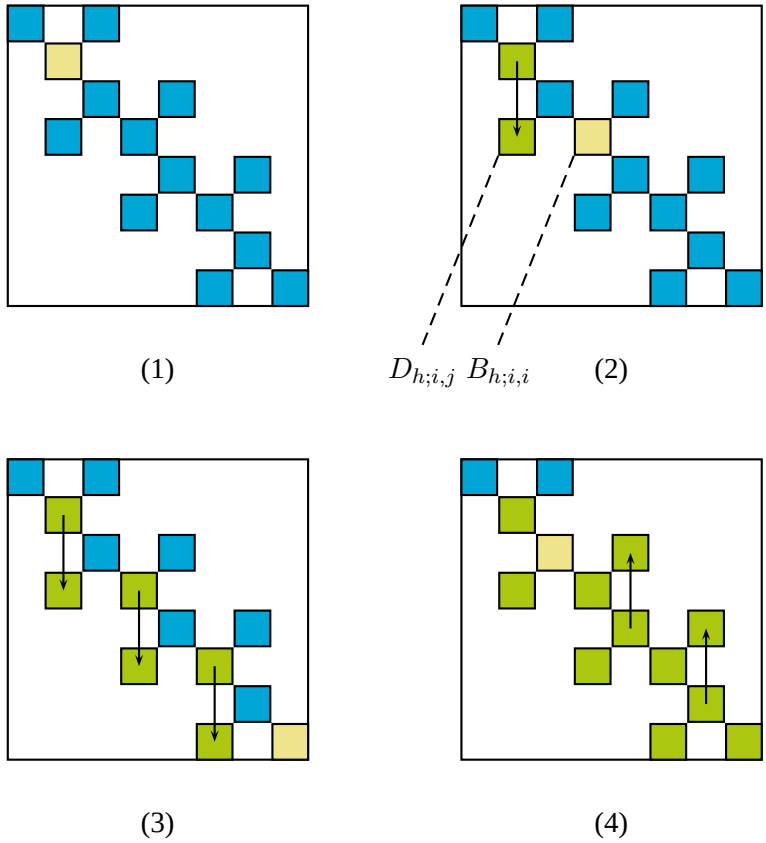


Figure 11: *Sweeping iteration in matrix $B_h + D_h$. (1) shows which block in B_h can be solved without using D_h. In (2) one can see how Equation 3.17 is constructed. Situation (3) is the end of the sweep going to the right. In (4) the sweep iteration is nearing completion.*

Let us turn to the physical interpretation, in Equation 3.7 we made the choice to only use upwind information of the angular flux. This means we must first solve the upwind angular flux of a patch before we can solve the angular flux of the patch itself. The only exception are the boundaries, where Dirichlet boundary conditions stipulate the upwind angular flux. In Figure 12 an illustration of the sweep algorithm in the spatial-angular space can be found. The arrows indicate the direction of flow and therefore the dependence of the patches, which results in the ordering of the patches.

We can now give an expression for the whole iteration, that is, both source and sweep iteration combined. We will use the superscript l for the source iteration and k for the sweep iteration. Note that the source iteration and the sweep iteration are different kinds of iterations. The source iteration is truly an iteration, where convergence questions can be asked. This means every instance of the source iteration is an approximation of the solution. The sweep iteration, however, is an iteration of the rows in the system of equations. It is actually a direct method, which becomes clear when one looks at rows separately. Therefore we mean by index $k+1$ the next row under consideration and by k all rows of which the equation is already computed. The full iteration can

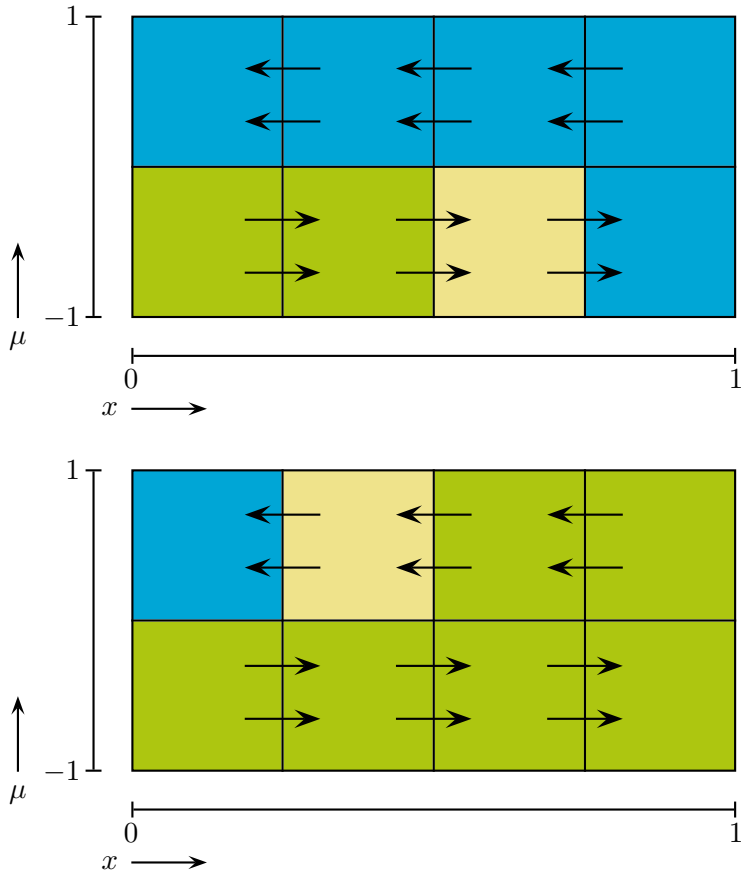


Figure 12: *Sweep iteration in the spatial-angular space. The direction of flow, and therefore the dependence between patches, is shown by the arrows. In (1) the right going directions are computed, while in (2) the left going directions are computed. After both directions are finished, the system is solved*

then be written as

$$\phi^{l+1,k+1} = B_h^{-1} S_h \phi^{l,k} - B_h^{-1} D_h \phi^{l+1,k} + s_h^k \tag{3.18}$$

For the Jacobi iteration sufficient conditions for convergence have been proven. The iteration converges when the matrix L_h is diagonally dominant. That is, when the absolute sum of the off-diagonal elements is smaller than the absolute diagonal element.

We will show that the matrix L_h satisfies this condition. To that end we will look at a general row in the matrix. For the diagonal element we can write

$$L_{h;i,i} = \frac{1}{3}\Delta x \Delta \mu_{p'} \sigma_t \pm \frac{1}{2} < \mu_{p'} > + < \mu_{p'} > + \frac{1}{3}\Delta x \Delta \mu_{p'}^2 \frac{\sigma_s}{4\pi} \tag{3.19}$$

where the terms originate from, respectively, removal, volumetric streaming, boundary streaming and scatter. p' denotes the index of the patch of this row and p is the index of the other patches in this element. For the absolute sum of the off-diagonal elements we can write

$$\sum_{i \neq j} |L_{h;i,j}| = \frac{1}{6}\Delta x \Delta \mu_{p'} \sigma_t \pm < \mu_{p'} > + < \mu_{p'} > + \frac{\sigma_s}{4\pi}\left[\frac{1}{6}\Delta x \Delta \mu_{p'}^2 + \frac{1}{2}\sum_{p \neq p'} \Delta x \Delta \mu_{p'}^2\right] \tag{3.20}$$

The inequality we need to prove for convergence can then be written as

$$\frac{1}{6}\Delta x \Delta \mu_{p'} \sigma_t > \frac{\sigma_s}{4\pi}\left[\frac{1}{6}\Delta x \Delta \mu_{p'}^2 + \sum_{p \neq p'} \frac{1}{2}\Delta x \Delta \mu_p^2\right] \tag{3.21}$$

$$4\pi \Delta \mu_{p'} > \Delta \mu_{p'}^2 + 3\sum_{p \neq p'} \Delta \mu_p^2 \tag{3.22}$$

We can further simplify this inequality to an inequality that surely holds, since $\Delta \mu_{p'} > 0$. Therefore the inequality above holds and the Jacobi iteration is surely convergent.

$$4\pi > \Delta \mu_{p'} + 3\sum_{p \neq p'} \Delta \mu_p \tag{3.23}$$

$$> \Delta \mu_{p'} + 3(2 - \Delta \mu_{p'}) \tag{3.24}$$

$$> 6 - 2\Delta \mu_{p'} \tag{3.25}$$

In these steps we used that $\Delta \mu_{p'} + \sum_{p \neq p'} \Delta \mu_p = 2$. Note that this is a sufficient, not necessary, condition for convergence of a Jacobi method.

3.3 Derivation of Adjoint Operator.

Here we will look more closely at the derivation of the adjoint problem. To that end we write the transport problem in the following way

$$L\phi = s \tag{3.26}$$

In Section 2.6 we derived the adjoint transport operator and discretised it in the same way as the forward operator. For clarity we will repeat the definition of the adjoint operator and both operators here. The adjoint operator L^* of any operator L is defined as

$$< \phi^*, L\phi > = < L^*\phi^*, \phi > \tag{3.27}$$

where ϕ and ϕ^* are functions from the space L and L^* work on, in general this is L^2. $< \cdot, \cdot >$ denotes the standard inner product on this space. Furthermore we required that the adjoint operator satisfies the following equation

$$L^*\phi^* = \sigma_D \tag{3.28}$$

where σ_D is the macroscopic detector cross section. This leads to the following equality, which we will call the duality relation

$$< s, \phi >=< \phi^*, \sigma_D >= J \tag{3.29}$$

in which J is the detector response. With properly chosen boundary conditions, we have shown that the forward and adjoint transport operator can be expressed as

$$L = \hat{\mathbf{\Omega}} \cdot \nabla + \sigma_t - \frac{\sigma_s}{4\pi} \int_{4\pi} d\hat{\mathbf{\Omega}} \tag{3.30}$$

$$L^* = -\hat{\mathbf{\Omega}} \cdot \nabla + \sigma_t - \frac{\sigma_s}{4\pi} \int_{4\pi} d\hat{\mathbf{\Omega}} \tag{3.31}$$

Since these operators and their associated equations have a similar form, we can discretise them in the same way. This leads to a discretisation where in the adjoint operator the volumetric and boundary streaming matrices are subtracted in stead of added to the matrix L_h^*.

The derivation above is one way to obtain a discretised expression for the adjoint transport equation, however there is another way to obtain such an expression. Figure 13 shows the two routes to L_h^*, where we have now explored the upper route, that is first deriving the adjoint operator L^*, then discretising this operator. The other route starts by discretising the forward operator L, obtaining L_h, of which we can the adjoint operator.

The discrete transport equation can be written as follows, which is a matrix equation,

$$L_h\phi_h = s_h \tag{3.32}$$

and let us define the detector response by

$$J = d_h^T\phi_h \tag{3.33}$$

$$= d_h^T L_h^{-1} s_h \tag{3.34}$$

Since the detector response is a scalar we know that $J = J^T$. So we can write

$$J = s_h^T (L^{-1})^T d_h \tag{3.35}$$

$$= s_h^T (L^T)^{-1} d_h \tag{3.36}$$

$$= s_h^T \phi_h^* \tag{3.37}$$

where ϕ_h^* is given by the adjoint problem

$$L^T\phi_h^* = d_h \tag{3.38}$$

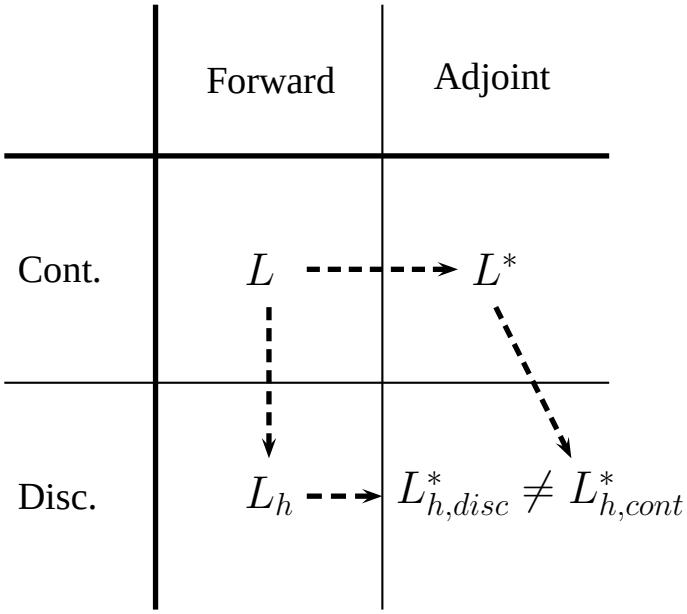


Figure 13: *Schematic overview of the continuous and discrete forms of the forward and adjoint equation. There are two routes that can be taken to arrive at a discretised adjoint equation starting from the continuous forward equation. The routes can result in a different expression of the discretised adjoint.*

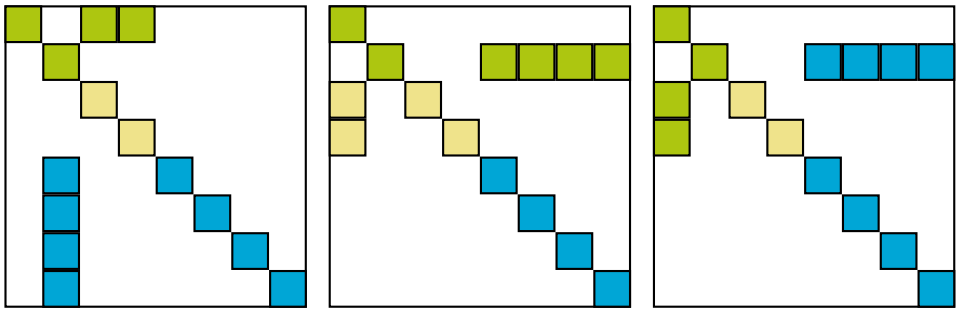
from which we can deduce that the adjoint operator is simply the transpose of the forward operator, $L_h^* = L_h^T$.

We can now examine whether these two expressions for the discrete adjoint are the same. The elements on the diagonal are not affected by transposing a matrix, therefore we will start by looking at these elements. All terms but the boundary streaming terms that are taken from the upwind flux add to the diagonal elements. In the continuous adjoint we see all streaming terms are subtracted from the diagonal elements, while in the discrete adjoint they are still added to the diagonal elements, since we are only taking the transpose of the forward matrix.

This means the volumetric streaming term will always make a difference, for uniformly and non-uniformly refined meshes. However the boundary streaming term does not introduce a difference between the two adjoints for uniformly refined meshes. In the results of the constant patches we already noted that the error estimator works better with uniformly refined meshes. For linear patches the only case where the error estimator provides a correct answer is with a uniformly refined mesh. This confirms the hypothesis that the origin of the bad quality of the error estimator is the difference in the continuous and discrete adjoints.

We will now discuss why for non-uniform meshes the boundary streaming term can introduce a difference between the continuous and discrete adjoints. Figure 14(a) shows the matrix of the forward operator for a non-uniform mesh. The mesh consists of two spatial elements with two patches on the left element and six on the right element. On the left element we have one left-going patch and one right-going. In the right element there are two left-going patches and four

right-going patches. Blue, green and yellow blocks have different entries, since the position of the patch is different, that is, the μ range is different.



(a) *Forward operator* (b) *Discretised continous adjoint* (c) *Discrete adjoint*

Figure 14: *Difference between the discretised continuous adjoint and the discrete adjoint for the boundary streaming term. This difference arises from using different patch sizes and therefore makes the difference between the adjoints only larger in non-uniformly refined meshes.*

The first remark is that transposing the forward matrix looks a lot like reversing the direction of flow. We see the same non-entries in both Figures 14(b) and 14(c). However, the patches on which the values of the entries are based are different. When constructing the discretised continuous adjoint the patch sizes of the current patch are considered, while in the discrete adjoint the patch sizes of the forward associated patches are used.

This only introduces a difference between the adjoints for non-uniform meshes, since all patch sizes are the same for uniformly refined meshes. So the discrete adjoint and discretised continuous adjoints of uniform meshes only differ in the volumetric streaming term, which makes the error estimator better in this case.

In the goal-oriented adaptive algorithm the breaking of this symmetry should not lead to problems, as only the discretised continuous adjoint problem is used. The error estimate and criterion for refinement are both formulated in the continuous domain, not the discrete domain. More on this can be found in Section 6.

3.4 Error Estimation and Adjoint Approximation

The fourth and final part of this section is an investigation into the goal-oriented error estimate used with the constant patches to decide where refinement should take place. The goal-oriented methods make use of the adjoint solution ϕ^* and try to refine in such a way that the detector response error is as small as possible. Both methods refine roughly the same patches, as can be seen in the spatial patch distributions in the last section. However, the estimate of the error they provide is very different, the error of the global adjoint approximation is a reasonable indication of the real error, while that of the local adjoint is not reliable. An explanation for this behaviour is presented below.

We will start by defining both the forward and adjoint transport problems in discrete form. As before the h subscript denotes this is the discrete form, we will use $h+1$ for a level deeper in

refinement, in other words with more unkowns. Conversely $h-1$ will denote the discrete problem on a level higher in refinement. The current level for the forward problem is h:

$$L_h \phi_h = s_h \tag{3.39}$$

To represent the error estimator in matrix form we need the adjoint problem solved on two levels, both h and $h+1$:

$$L_h^* \phi_h^* = d_h \tag{3.40}$$
$$L_{h+1}^* \phi_{h+1}^* = d_{h+1} \tag{3.41}$$

Before defining the error estimator we need a matrix that projects vectors from a refinement level h to levels $h+1$ and $h-1$, which we will call respectively E_h^{h+1} and E_h^{h-1}. In the case we are considering, with constant patches and a division into two equally sized patches upon refinement, these matrices have the following form

$$E_h^{h+1} = \begin{bmatrix} 1 & 0 & & \emptyset \\ 1 & 0 & & \\ 0 & 1 & 0 & \\ 0 & 1 & 0 & \\ & \ddots & \ddots & \ddots \\ & & 0 & 1 \\ \emptyset & & 0 & 1 \end{bmatrix} \tag{3.42}$$

$$E_h^{h-1} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 0 & & & & \emptyset \\ & 0 & 1 & 1 & 0 & & \\ & & \ddots & \ddots & \ddots & \ddots & \\ \emptyset & & & & 0 & 1 & 1 \end{bmatrix} \tag{3.43}$$

where E_h^{h+1} is $n \times 2n$ and E_h^{h-1} is $\frac{n}{2} \times n$.

Now we can look at the way the error estimator is built up and consecutively how a different approximation of the exact adjoint influences the estimate. For this investigation we do not have to be concerned with the forward solution on different levels, we will look at the influence of the adjoint on the error calculation in one refinement iteration. This means we can represent the error estimate per patch as

$$\eta = R(E_h^{h+1} \phi_h^* - \phi_{aprx}^*) \tag{3.44}$$

where η is the vector holding all contributions of the patches and ϕ_{aprx}^* is the adjoint angular flux approximation that is chosen, this can be either the global adjoint ϕ_G^* or local adjoint ϕ_L^*. Note that R is depending on physical properties of the problem, discrete parameters of our discretisation and the forward angular flux. Furthermore the estimate of the error itself is given by $\Delta J = \mathbf{1}^T \eta$. The global adjoint is simply given by

$$\phi_G^* = \phi_{h+1}^* \tag{3.45}$$
$$= (L_{h+1}^*)^{-1} d_h \tag{3.46}$$

The local adjoint approximation is more involved. Before we can give an explicit expression for ϕ_L^* we will need to split the matrix L_h^* in the block diagonal part B_h^* and the scatter and boundary transport part $D_h^* + S_h^*$ to obtain the following

$$B_h^* \phi_h^* = s_h - (D_h^* + S_h^*)\phi_h^* \tag{3.47}$$

We can now locally refine the left hand side of this equation without altering the block diagonal form of the matrix. If we perform such a local refinement to each patch we find the following equation

$$B_{h+1}^* \phi_L^* = s_{h+1} - E_h^{h+1}(D_h^* + S_h^*)\phi_h^* \tag{3.48}$$

where $\phi_h^* = L_h^* s_h$. In this way a local refinement is performed, without altering neighbouring patches, to give an approximation of the adjoint solution on $h+1$. This constitutes the local approximation of the adjoint.

We can now explore to what extend the two approximations of the exact adjoint solution provide us with a similar error estimate or with a similar ordering of the contribution of the patches. First we will take a closer look at whether the ordering of the contribution is the same in the two cases. This would explain why the two methods roughly refine the same regions in the domain.

In order to easily compare the two estimator expressions they are presented here

$$\phi_L^* = (B_{h+1}^*)^{-1} \left[d_{h+1} - E_h^{h+1}(D_h^* + S_h^*)\phi_h^* \right] \tag{3.49}$$
$$\phi_G^* = (B_{h+1}^*)^{-1} \left[d_{h+1} - (D_{h+1}^* + S_{h+1}^*)\phi_{h+1}^* \right] \tag{3.50}$$

Note that the difference between the two estimators is the right most term, which we can see as a difference in the right hand side of a system of equations. Considering the difference between ϕ_h and ϕ_{h+1}, we can justify the following assumption. Take two entries or patches in ϕ_h, patch A and patch B. Let us assume without loss of generality that in this case $\phi_A < \phi_B$. When both patches are refined, thus obtaining entries in ϕ_{h+1} which we will call patches A1, A2, B1 and B2, the assumption will be that $\phi_{A1}, \phi_{A2} < \phi_{B1}, \phi_{B2}$. Or, to put it into words, the order of the entries of ϕ_h is preserved as the same order of pairs of ϕ_{h+1}.

The operators D and S do not affect the ordering of the vector, when the vector is multiplied by it, by their nature. The matrix E_h^{h+1} increases the length of a vector, but again the ordering is not affected in the same sense of ordering as used above. From this we can conclude that the ordering of the right hand side of Equations 3.49 and 3.50 are roughly the same, as our assumptions holds in most cases but not all. Since this is the only difference between the local and global error estimator we can conclude that in general the ordering of the error contributions, and thus the decisions which patches are refined, is the same.

The matrix B^*_{h+1} is block diagonal, with blocks of two by two for the constant patches. This means the inverse of that matrix is also block diagonal. Multiplying the right hand side by this inverse will therefore only introduce local dependence between entries in the right hand side.

In Equation 3.44 we see that the the error contribution η depends on the difference of $E_h^{h+1}\phi_h^*$ and the estimator flux. Since these two vectors are almost the same, as they solve the same problem, a small perturbation in one of them will lead to large relative changes. This can cause the total estimated error to be off for the local adjoint estimator.

4 Neutron Transport with Linear Angular Basis Functions

After examining the algorithm with constant patches in more detail we can turn to using linear basis functions in the angular domain as well. This means we use linear basis functions in both spatial and angular domain and we will call this method the linear patch method (as opposed to the constant patches).

There are two issues when switching from constant patches to linear patches. First of all we need to discretise the transport equation with our new basis functions, which leads to a larger transport matrix with the same number of patches, as there are more unknowns with higher order basis functions. Secondly we need to look at the conservation of neutrons with the new basis functions. When the angular refinement of two neighbouring elements is different the continuity relations that we used with the constant patches, see Section 2.3, do not ensure particle conservation.

4.1 Discretisation with Linear Basis Functions

The Galerkin discretization procedure is the same for constant and linear patches, in the sense that only the evaluation of the integral will be different. In both cases the space of test functions V is the product space of the spatial and angular test functions, respectively V_s and V_a.

$$V = V_s \otimes V_a \tag{4.1}$$

The difference is that for linear patches the space of angular basis functions has a larger dimensionality. In Figure 15 an illustration of the basis functions is presented.

One can see that the basis functions are linear in the direction of the coordinates and quadratic in any other direction. Using basis functions defined as a product of functions allows us to discretize in two steps, spatial and angular domains separately. To large extend the discretisation is the same as that of constant basis functions, which is presented in Section 2.2. We start with the same approximation as in that section,

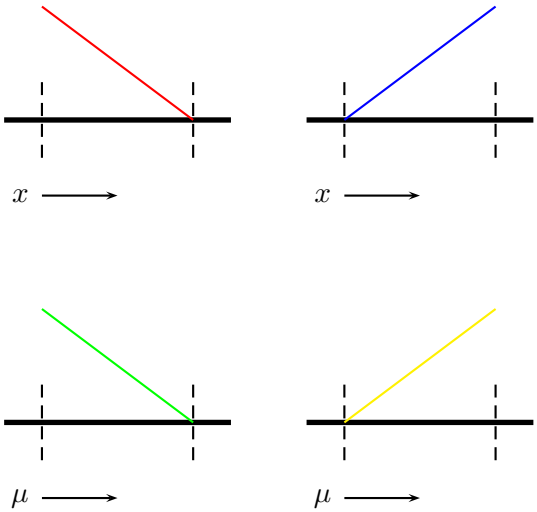
$$\phi(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}) \approx \sum_e \sum_p \phi_{e,p} \gamma_e(\boldsymbol{r}) G_{e,p}(\hat{\boldsymbol{\Omega}}) \tag{4.2}$$

which leads to equation 2.11,

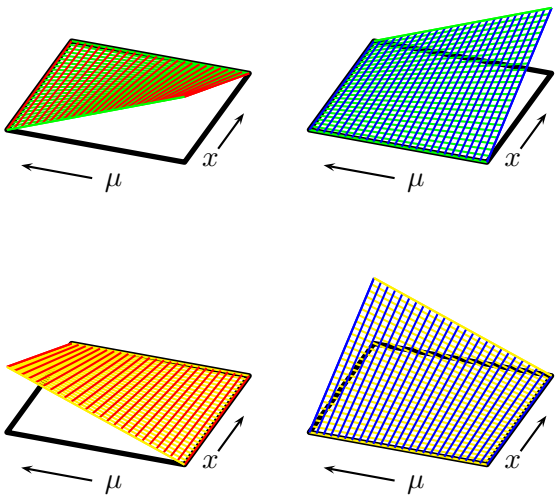
$$\int_V \int_{4\pi} \gamma_e(\boldsymbol{r}) G_{e,p}(\hat{\boldsymbol{\Omega}}) \left[(\hat{\boldsymbol{\Omega}} \cdot \nabla + \sigma_t) \sum_{e',p'} \phi_{e',p'} \gamma_{e'} G_{e',p'} - s - \frac{\sigma_s}{4\pi} \Phi \right] d\hat{\boldsymbol{\Omega}} = 0 \tag{4.3}$$

Now we apply again the divergence theorem to this expression, resulting in

$$\begin{aligned} &\int_{4\pi} \int_V \gamma_e(\boldsymbol{r}) G_{e,p}(\hat{\boldsymbol{\Omega}}) \left[(\hat{\boldsymbol{\Omega}} \nabla \cdot + \sigma_t) \sum_{e',p'} \phi_{e',p'} \gamma_{e'} G_{e',p'} - s - \frac{\sigma_s}{4\pi} \Phi \right] dV \\ &+ \int_{\delta V} \hat{\boldsymbol{\Omega}} \cdot n \sum_{e',p'} \phi_{e',p'} \gamma_{e'} G_{e',p'} d\delta V d\hat{\boldsymbol{\Omega}} = 0 \end{aligned} \tag{4.4}$$



(a) Basis functions of each space separate.



(b) Basis functions for the combined spatial-angular domain.

Figure 15: *Basis of the linear test space for the one dimensional transport equation.*

Where we will take the boundary flux values in the same way as for the constant patches. When a face of an spatial element is an outflow boundary for the current patch, we take the angular flux of the element itself. When it is in an inflow boundary, we take the upwind flux. This is completely analogous to what was done in the case of the constant patches.

In the case of linear patches we can write the scalar flux Φ as

$$\Phi = \int_{4\pi} \phi d\hat{\mathbf{\Omega}} \tag{4.5}$$

$$\Phi_e = \int_{4\pi} \sum_p \phi_{e,p} G_{e,p} d\hat{\mathbf{\Omega}} \tag{4.6}$$

$$= \sum_p \phi_{e,p} \int_{4\pi} G_{e,p} d\hat{\mathbf{\Omega}} \tag{4.7}$$

$$= \sum_p \phi_{e,p} \frac{1}{2} \Delta\mu \tag{4.8}$$

The matrix obtained when evaluating the integrals in Equation 4.4 has the same structure as that of the constant patches, which is derived in Section 3.1. This means we can apply the same method of solving the equation as for the constant patches. However, the blocks on the diagonal will now by four by four instead of two by two. In those blocks we evaluate all the integrals over the basis functions exactly, even if a product leads to a second order basis function.

4.2 Continuity Relations

In the algorithm we allow for each element to have a different distribution of patches. Since we use flux values of neighbouring elements when updating the flux of a patch, we need interpolation rules when the distribution of patches is not the same. More specific, we need interpolation rules when the patch we are considering does not have a 'sister' patch, with the same interval in μ, in the neighbouring element. Since the continuity relations used for constant patches are not generally applicable, we need to derive new relations.

Suppose we have two neighbouring elements, e_c and e_f, and a patch distribution on these elements such that interpolation rules are needed. We can distinguish two cases, where neutrons are streaming from a coarse distribution to a fine one and vice versa. An illustration of this can be found in Figure 16.

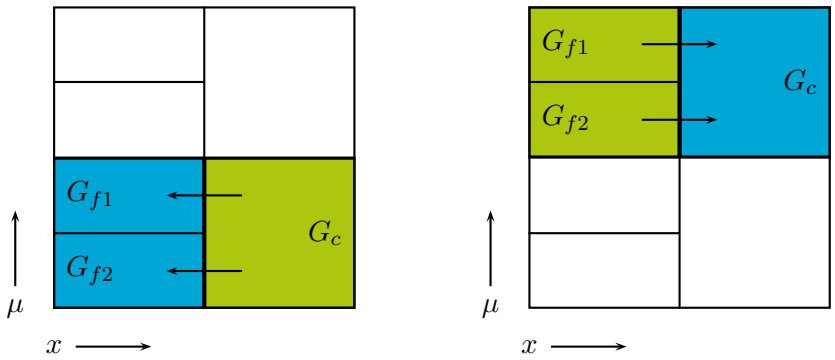
The interpolation rules are derived from a conservation principle regarding the number of neutrons. The transport equation considers 'free neutrons', neutrons that are not bound by a nucleus, and describes free neutron sources and sinks. Taking these sources into account the number of neutrons must be conserved. Therefore in our interpolation rules no neutrons may be lost or created.

Conservation of neutrons crossing the boundary between the elements e_c and e_f can be formulated as

$$\int_{p_c} \hat{\mathbf{\Omega}} \cdot \hat{\boldsymbol{n}} \phi^{e_c}(\boldsymbol{r}_e, \hat{\mathbf{\Omega}}) d\hat{\mathbf{\Omega}} = \int_{p_{f1} \cup p_{f2}} \hat{\mathbf{\Omega}} \cdot \hat{\boldsymbol{n}} \phi^{e_f}(\boldsymbol{r}_e, \hat{\mathbf{\Omega}}) d\hat{\mathbf{\Omega}} \tag{4.9}$$

or, for one spatial dimension,

$$\int_{p_c} \mu \phi^{e_c}(x_e, \mu) d\mu = \int_{p_{f1} \cup p_{f2}} \mu \phi^{e_f}(x_e, \mu) d\mu \tag{4.10}$$



(a) Coarse to fine (b) Fine to coarse

Figure 16: *Two cases of different angular distribution of patches in neighbouring elements. To ensure neutron conservation interpolation relations between the discrete patches need to be derived.*

This expression comes down to counting the neutrons that cross the boundary from one side to the other. We do not need to consider the whole angular domain for this continuity relation, since we can have the situation where only one patch is refined more in element e_f. This means the interpolation has to be local, in other words only relate to the coarse patch p_c on element e_c and the fine patches p_{f1} and p_{f2} on element e_f.

To find general interpolation rules it is sufficient to find rules for two cases, streaming from two to one patch and vice versa. Upon refinement a patch is divided into two equal new patches, so by applying the simple interpolation rule multiple times we can apply it to any situation we might encounter.

4.2.1 A First Attempt

We can easily formulate an interpolation rule to go from the coarse element e_c to the fine element e_f, since we can exactly represent the coarse function in the fine function space. Take the coarse flux to be represented by the two coefficients $\phi_{e_c,p1}$ and $\phi_{e_c,p2}$, where $p1$ and $p2$ denote, respectively, the left and right angular basis function on the boundary of the elements. We can likewise do this for the fine patches, only now we have four angular basis functions.

The interpolation from the coarse to the fine element can then be written as

$$\begin{bmatrix} \phi_{e_f,p1} \\ \phi_{e_f,p2} \\ \phi_{e_f,p3} \\ \phi_{e_f,p4} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \phi_{e_c,p1} \\ \phi_{e_c,p2} \end{bmatrix} \tag{4.11}$$

which can be used to formulate an interpolation rule of the converse direction. Let P denote the interpolation matrix as defined above, ϕ_{e_c} denote the vector of coefficients in the coarse element

and ϕ_{e_f} denote that of the fine element. Then we can rewrite the interpolation as

$$\begin{aligned} \phi_{e_f} &= P\phi_{e_c} && (4.12)\\ P^T\phi_{e_f} &= P^TP\phi_{e_c} && (4.13)\\ \phi_{e_c} &= (P^TP)^{-1}P^T\phi_{e_f} && (4.14) \end{aligned}$$

where we have that

$$(P^TP)^{-1}P^T = \begin{bmatrix} \frac{3}{4} & \frac{1}{4} & \frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{3}{4} \end{bmatrix} \quad (4.15)$$

It turns out this interpolation rule violates the conservation of neutrons when crossing a boundary. This will introduce errors when non-uniform refined meshes are used and is therefore not preferred. An approach where the conservation of neutrons is ensured is therefore now examined.

4.2.2 Continuity as Minimization Problem with Constraints

We can evaluate this expression by substituting the expansion of the flux in basis functions. The angular basis functions are linear in the angular component μ. Substituting linear functions like this yields an expression of the continuity relation in terms of the space of basis functions.

$$\begin{aligned} \int_{p_c} \mu\phi^{e_c}(x_e,\mu)d\mu &= \int_{p_{f1}\cup p_{f2}} \mu\phi^{e_f}(x_e,\mu)d\mu && (4.16)\\ \int_{p_c} \mu \sum_{coarse} v_n^c\phi_n^{e_c}(x_e,\mu)d\mu &= \int_{p_{f1}\cup p_{f2}} \mu \sum_{fine} v_m^f\phi_m^{e_f}(x_e,\mu)d\mu && (4.17)\\ \sum_{coarse} v_n^c \int_{p_c} \mu\phi_n^{e_c}(x_e,\mu)d\mu &= \sum_{fine} v_m^f \int_{p_{f1}\cup p_{f2}} \mu\phi_m^{e_f}(x_e,\mu)d\mu && (4.18)\\ \boldsymbol{w}^{cT}\boldsymbol{v}^c &= \boldsymbol{w}^{fT}\boldsymbol{v}^f && (4.19) \end{aligned}$$

The vector $\boldsymbol{w}^c$ contains the integrals over each of the coarse basis functions and $\boldsymbol{w}^f$ those of the fine basis functions. Note that the vectors of the coarse space are two-vectors, while those of the fine space are four-vectors, respectively the number of basis functions on the interval of μ we are considering. We call this condition the 'hard condition', as we want this equality to hold exactly.

Using only the hard condition we cannot yet determine an interpolated flux in each case. To this end we need another condition. The most natural condition is that the interpolated flux should resemble the original flux. The most resembling flux can be written as the solution of a minimization problem. Let $||.||$ denote a norm, then the minimization problem is

$$\min_{\boldsymbol{v}^c \text{or} \boldsymbol{v}^f} ||\phi^{e_c} - \phi^{e_f}|| \qquad \text{subject to} \qquad \boldsymbol{w}^{cT}\boldsymbol{v}^c = \boldsymbol{w}^{fT}\boldsymbol{v}^f \quad (4.20)$$

Now consider the first case of different distributions, where neutrons stream from a coarse element e_c to a fine element e_f. In this case we need to determine the interpolated flux on the fine

element e_f. Since the space of basis functions of the coarse element e_c is contained in that of the fine element e_f, we can represent the flux in the fine space exactly. In other words, we can find $\boldsymbol{v}^f$ such that $||\phi^f - \phi^c|| = 0$.

The inclusion of the coarse space in the fine space also means that, in general, we will need an approximation of the flux in the other case, where neutrons are streaming from a fine element e_f to a coarse element e_c. By choosing a norm in which the difference is to be minimized we can derive different interpolation rules.

A natural way to start is by looking at the two-norm, since we are dealing with physically relevant functions. The two-norm is induced by the standard inner product, which means

$$< f, g > = \int f(\mu)g(\mu)d\mu \tag{4.21}$$

$$||f||_2^2 = < f, f > \tag{4.22}$$

Let us turn to the minimization of $||\phi_{e_c} - \phi_{e_f}||$ when ϕ_{e_f} is given, so in the case of neutrons streaming from a fine to a coarse element. We can project ϕ_{e_f} on the coarse space of basis functions, which results in the following split of ϕ_{e_f},

$$\phi_{e_f} = \phi_{e_f}^{||} + \phi_{e_f}^{\perp} \tag{4.23}$$

where $\phi_{e_f}^{||}$ lies in the space of coarse basis functions and $\phi_{e_f}^{\perp}$ is orthogonal to that space. This projection, or orthogonal projection is done with respect to the standard inner product. Since the two-norm is induced by this inner product, we find the minimum of our expression by taking

$$\phi_{e_c} = \phi_{e_f}^{||} \tag{4.24}$$

which results in the minimum being attained with

$$||\phi_{e_c} - \phi_{e_f}|| = ||\phi_{e_f}^{||} - \phi_{e_f}^{||} - \phi_{e_f}^{\perp}|| \tag{4.25}$$

$$= ||\phi_{e_f}^{\perp}|| \tag{4.26}$$

By working out the integrals of the continuity expression in Equation 4.10 we conclude that the conservation constraint is satisfied. This derivation is not presented here, as it is tedious and long. We have now found an interpolation rule that conserves neutrons and finds the best approximate flux in the two-norm. This is implemented in the code for linear patches.

5 Results of One-Dimensional Linear Patches

The linear basis functions as described in the previous section were implemented using the same method of solving as was used for the constant patches. Therefore we can compare the constant and linear patches with each other. Besides this comparison we can also compare the three adaptive criteria that were formulated in Section 2.4. As a short recapitulation we summarize what the three criteria are.

Starting with the traditional criterion, which is a more commonly used method for determining where to refine. This method bases its decision on how much the solution locally changes with a local refinement. The other two criteria are both called goal-oriented criteria, since they have as goal to approximate the detector response as good as possible. In this criterion the exact adjoint solution ϕ^* is needed, since this is not available we have to approximate it. The global approximation is ϕ^*_{h+1}, when the level of the forward solution ϕ_h is h, or in other words the adjoint has twice the amount of patches as the forward has. The local adjoint is computed by taking only local refinements of the adjoint from level h to $h+1$. So only when the contribution of a patch is computed the adjoint is refined to a level deeper on that patch.

As a reference we also plotted the data of the discrete ordinates method for one dimension using a Gauss-Legendre quadrature. This is a widely used method, which is especially effective in problems with one spatial dimension. As stated before, this method does lend itself well for adaptive refinement.

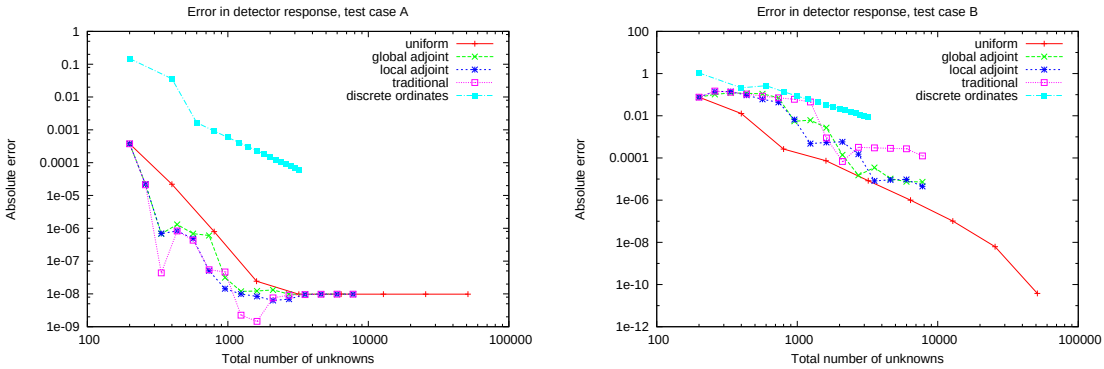
Several kinds of plots are used to compare the various methods. In general we look at the error versus the total number of unknowns, the error can be either the detector response error or the root mean square error, which is a more global error measure. This is an indication of the performance of the method, since the number of unknowns is also a measure of time and cost when the number of patches that are refined is taken equal between methods. In order to explain certain results spatial distributions of patches are used. These plots show the total number of patches within an element versus the position of that element. The quality of the error estimators will also be investigated, which is done by dividing the estimate by the exact error. This ratio should go to one for large numbers of unknowns when the estimator is accurate.

The obtained results are presented per kind of test case, that is, there are two homogeneous test cases, four separate source detector cases and two shielding cases. A short explanation of the test case will be given before the results are presented. Furthermore in Appendix A all test cases are listed with material properties and overviews of the geometry. The final results that are presented here are on the effect of the ratio of patches that is refined in each refinement iteration.

5.1 Homogeneous Slab (cases A and B)

Two homogeneous test cases are used, one with an optically thick and highly scattering medium (A) and the other with an optically thin and little scattering medium (B). Homogeneity here means that the material properties, that is the cross sections, are the same throughout the domain. Also the source and detector are present in the whole domain.

Figure 17 shows the error reduction in the detector response versus the number of unknowns. For patches with linear functions in both the spatial and angular direction we have four unknowns per patch. For both test cases we see that the discrete ordinates converges slower and needs

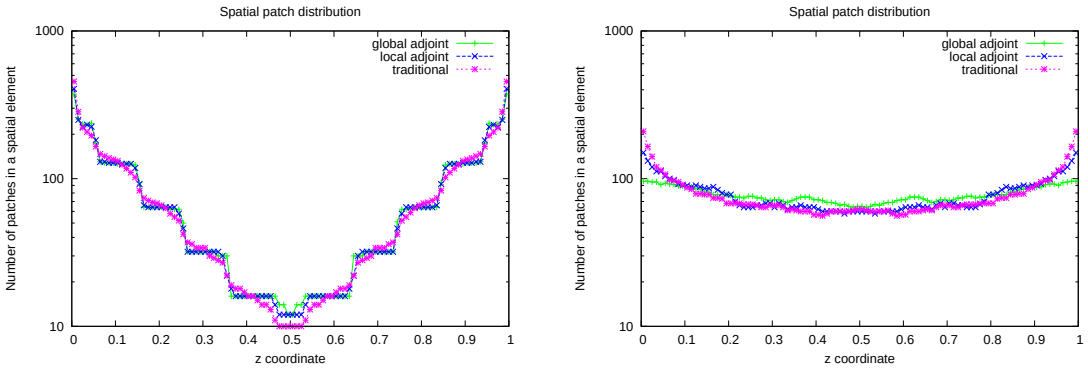


(a) *Test case A* (b) *Test case B*

Figure 17: *Absolute error in the detector response versus the number of unknowns of test cases A and B.*

more unknowns than the discontinuous Galerkin method with linear patches. Comparing the error reduction of the linear patches in this figure with that of the error reduction using constant patches in Figure 4 we can conclude that the linear patches work much better than the constant patches.

Also note that the difference is much larger in test case A, with the highly scattering material. This is due to the quadrature that is used with the discrete ordinates method, which is the same throughout the domain. However, the adaptive methods can refine certain parts of the domain, making that even more efficient. From Figure 18 we can conclude that this is the case. Case B has an almost flat distribution of patches as the outcome of several refinement iterations, while case A has an exponential distribution of patches.

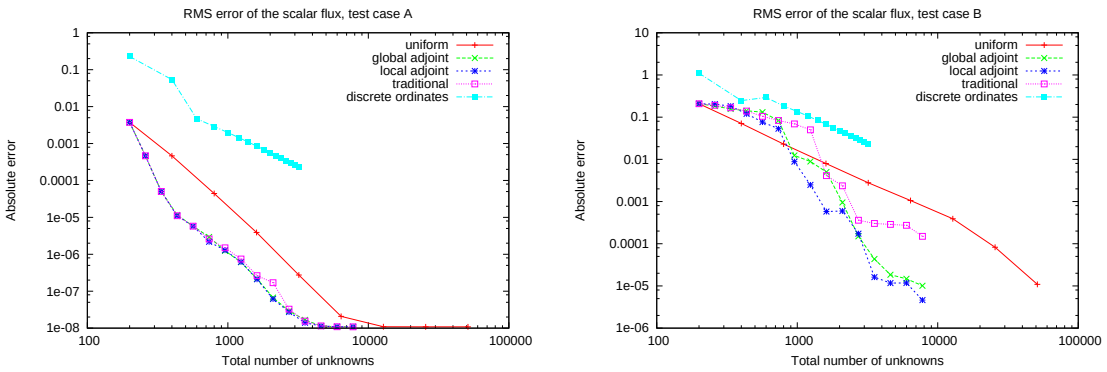


(a) *Test case A* (b) *Test case B*

Figure 18: *Distribution of patches along the spatial coordinate. The total number of patches in a spatial element is plotted agains the position of that element.*

We can investigate the method by comparing not only the error in the detector response, but also the root mean square error of the scalar flux. Figure 19 shows this error measure of the solution for the homogeneous test cases. For case A we can see in both error measures, the

detector response and the rms error of the scalar flux, that the adaptive methods perform similarly. It is especially noteworthy that the traditional refinement method behaves in the same way as the goal-oriented adaptive methods. This can also be seen in the spatial distribution of this test case, see Figure 18(a). In this distribution we also see the reason why all adaptive methods perform better than uniform refinement, the boundaries of the domain are more difficult than the middle to approximate well.



(a) *Test case A* (b) *Test case B*

Figure 19: *Reduction in the root mean square error of the scalar flux for the homogeneous test cases. The error is plotted against the total number of unknowns needed to obtain this error.*

The difference between the adaptive methods in both cases A and B can be neglected. The goal-oriented methods have a slightly larger decrease in error in case B, but that is not significant until the end of the graph. Comparing the spatial distribution in Figure 18(b) we can conclude that the traditional method refines too much on the edges of the domain, compared to the goal-oriented methods. This is is qualified as a minor difference between the methods.

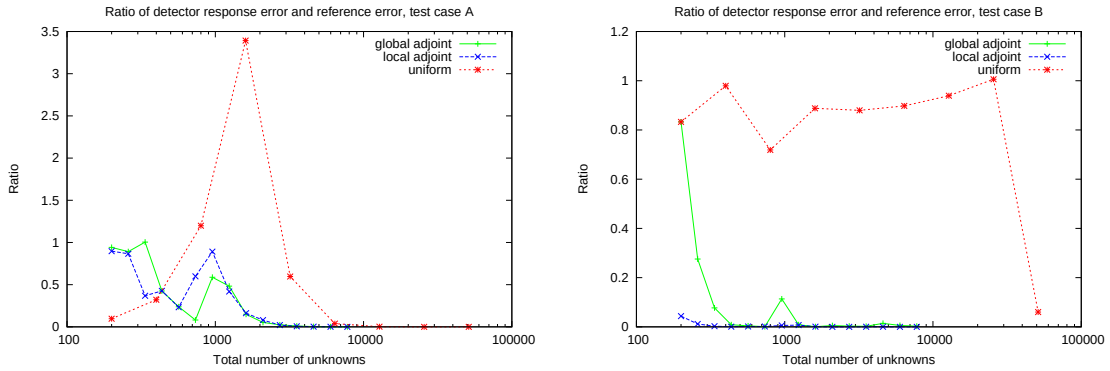
More interesting is the difference between the adaptive methods and uniform refinement in the detector response error. In case A we see that the adaptive methods have a smaller error than the uniform, while in case B this is the other way around. An explanation for this can be found when looking at the spatial distribution in Figure 18 and the solution of the scalar flux Φ, the integrated angular flux over the angular domain only, of the two cases in Figures 33(a) and 34(a). A correlation exists between the number of patches after refinement in an element and the change in the scalar flux Φ along the spatial direction. In the middle of the domain of case A we see little refinement and the scalar flux Φ is nearly constant in the spatial direction. However in the whole of domain B and on the edges of the domain of A we see much refinement and change in the scalar flux Φ. From this we can deduce that a uniform distribution of patches will work well in case B, while it will not work well in case A. Adaptivity therefore has an advantage in case A and the adaptive methods in case B cannot do much better than the uniform method. In the best case, for the adaptive methods, they will meet the performance of the uniform method.

As a last remark on convergence we would like to point out that the convergence rate of all adaptive methods and the uniform refinement are, before hitting the floor, fourth order or almost fourth order. Since the convergence of constant patches was second order, we expected then

that the convergence of linear patches would be fourth order. This indicates that this expectation was justified. We will see that for all cases the same conclusion can be drawn on the order of convergence.

The last observation on the homogeneous test cases is on the quality of the error estimator. We can quantify this quality as a ratio of the estimated error over the 'exact' error, where the exact error is approximated by looking at the detector response of the reference solution. The reference solution is computed at a deeper level than the tests take place, in this case with 1024 patches on each spatial element. This is one level deeper than the deepest uniform refinement level. In Figure 20 the error ratio's of both test cases are presented.

Only the global adjoint estimator with uniform refinement in case B produces an accurate estimator. We can disregard the final point in this graph as this point is close to the reference solution. The approximation of the exact error can for this point be too far off for an accurate quality. It will turn out that only this estimator has a good quality. All other test cases with all estimators provide useless error estimators. An explanation for this behaviour is proposed in the next section, if the reader wants to know more on this we refer to Section 6.



(a) *Test case A* (b) *Test case B*

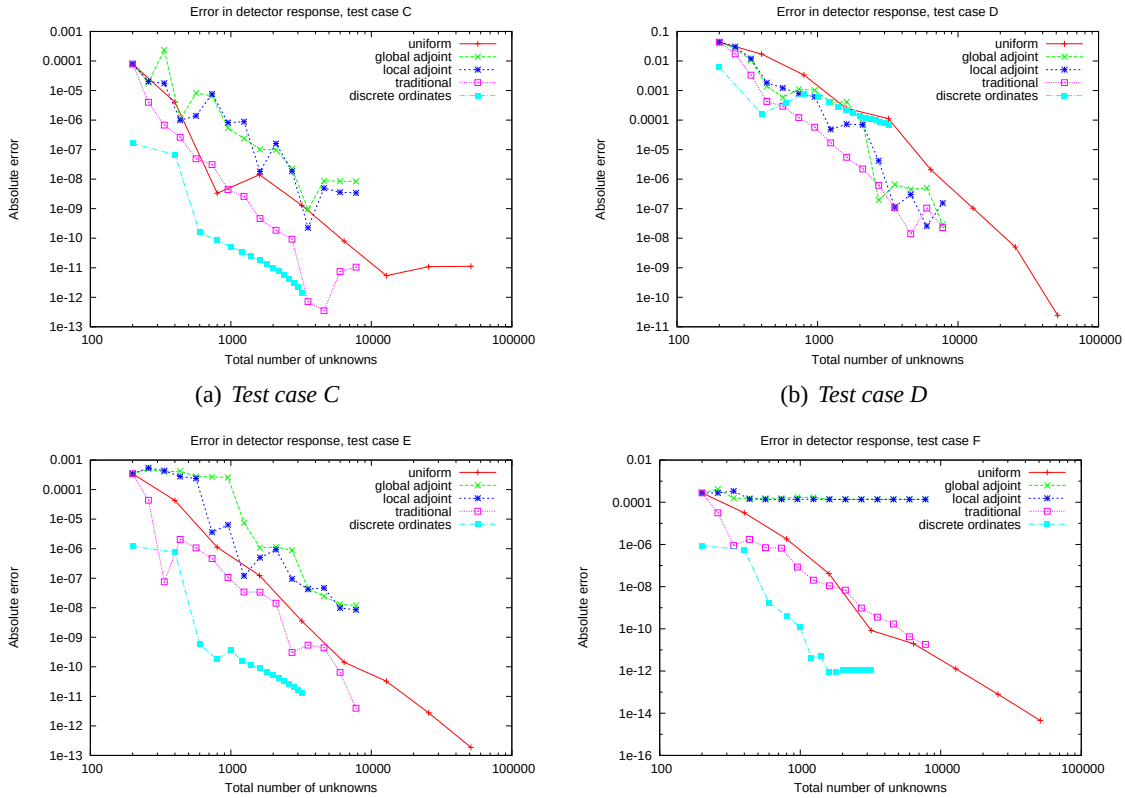
Figure 20: *Ratio of the estimated error over the 'exact' error. The exact error is determined by comparing the current detector response with that of the reference solution, computed with a very refined mesh.*

5.2 Separate Source and Detector (cases C, D, E and F)

The next set of test cases we present the results of are the separate source and detector geometries. This means one tenth of the domain on the left is the source region, with an isotropic homogeneous source. One tenth of the domain on the right of the domain is the detector region, where the detector is also isotropic and homogeneous. To obtain an accurate detector response for these test cases it is important to represent the neutrons streaming from source to detector well. However, in order to get an accurate overall solution, measured with the root mean square error of the scalar flux Φ, we need an accurate representation of the angular flux ϕ around the source.

A first observation of the error in the detector response of these test cases presented in Figure 21 is that the discrete ordinates method provides a more accurate detector response in three out

of four cases. This is somewhat the same behaviour as for the constant patches, except that the difference between discrete ordinates and discontinuous Galerkin has become smaller by using linear patches, see Figure 6.



(a) *Test case C* (b) *Test case D*

(c) *Test case E* (d) *Test case F*

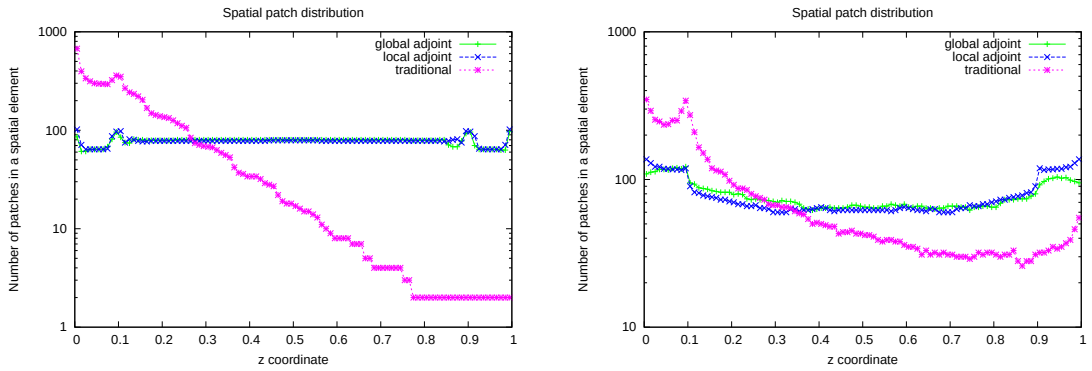
Figure 21: *Decrease in the detector response error versus the total number of unknowns for the separate source ande detector test cases.*

Case D, with different behaviour, has a scatter to total cross section ratio of 2, while case C has a much larger ratio and cases E and F have much smaller ratio's. In this fact we can find an explanation for the observation that linear patches work better in case D. In case C we have an almost linear flux profile along the angular direction, this can be well described by the linear patches. However, the Gauss-Legendre quadrature of discrete ordinates performs very well in with these properties too. Therefore we see that linear patches do 'gain' error reduction on discrete ordinates, but do not perform better.

In cases E and F we have very little and no scattering respectively, therefore the quadrature of discrete ordinates does not play any role in achieving and accurate detector response. In one dimension we use the Gauss-Legendre quadrature also for choosing the directions used in discrete ordinates. These directions can very efficiently represent the angular flux. Linear patches can describe that angular flux more efficiently than constant patches, that is why the error of linear patches is smaller than that of constant patches, using the same number of unknowns. However,

it is not yet as efficient as discrete ordinates. Then in case D we can see the situation where the linear patches perform well, since in this case the quadrature of discrete ordinates is important and the flux profile is not linear. Linear patches can efficiently represent an exponential angular flux, while the quadrature of discrete ordinates performs less well in these conditions.

Comparing the different refinement methods of discontinuous Galerkin we see that in general the traditional method provides the most accurate detector response. When we examine the spatial distribution in Figure 22 we can see that the accurate solution is achieved when there is more refinement in the source region than in the detector region. There are some differences between the two test cases in the spatial distribution of the traditional method. Figure 22(a) shows that in case C there is no refinement in the detector region at all, while in case D (Figure 22(b)) we see that the refinement in the detector region is about one tenth of that in the source region. This is due to the different angular flux profile in the two cases, linear in C and more exponential in D. To represent a linear profile with linear basis functions requires few patches, while representing an exponential function requires more patches.



(a) *Test case C* (b) *Test case D*

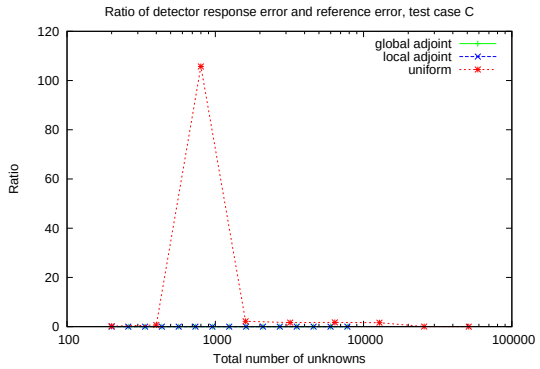
Figure 22: *Distribution of patches along the spatial coordinate. The total number of patches in a spatial element is plotted agains the position of that element.*

In general the goal-oriented adaptive methods provide a more or less uniform distribution of patches, as can be seen in Figure 22. This explains why it in general has the same error reduction as the uniform refinement. That, however, does not provide the solution with the smallest error in both error measures. The difference between the goal-oriented refinement and traditional refinement is the use of the adjoint. The solution of the adjoint problem can be used as the importance of a location in the domain to the detector response. Therefore the fact that goal-oriented methods produce an almost uniform distribution of patches would mean that the adjoint puts too much importance on patches near the detector. In the next section we present an explanation for this behaviour, which is likely connected with the problem of the error estimator.

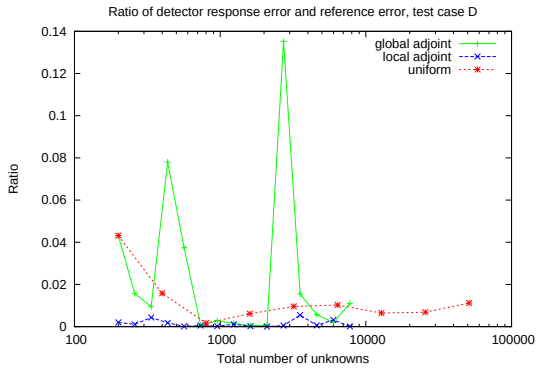
Furthermore it is remarkable to see the error reduction in the detector response error with the traditional refinement criterion. Comparing Figure 21 with Figure 6 we see that with constant patches the traditional method provides us with a poorly refined mesh. Poor in the sense that it provides an erroneous detector response. This is not the case with linear patches, where the

method works very well. This shows that linear patches are much more efficient in representing the angular flux well. This can also be seen in the fact that convergence in these four cases is again approximately fourth order.

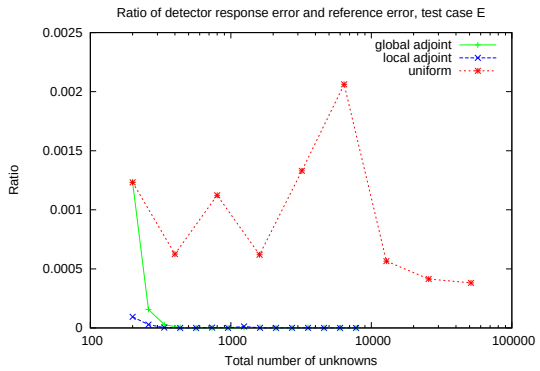
The last remark we would like to make concerning the separate source detector test cases C, D, E and F is that the error estimator is off by a large factor. Figure 23 shows the error ratio of each relevant refinement method. In all cases, except one, we have seen so far the estimated error is too small. In the next section we will present a more in-depth investigation of this problem. However, the distribution of patches after refinement is like what one would expect in most cases. The distribution is also similar to that of the constant patches. A final observation is that the global and local adjoint methods produce almost the same distributions. Taking all this into account it is likely that the refinement criterion is, generally speaking, correct and provides accurate results.



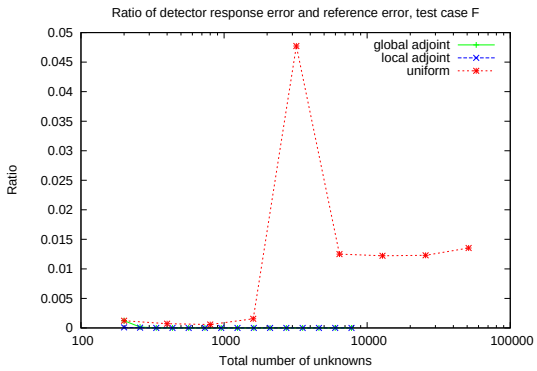
(a) *Test case C*



(b) *Test case D*



(c) *Test case E*



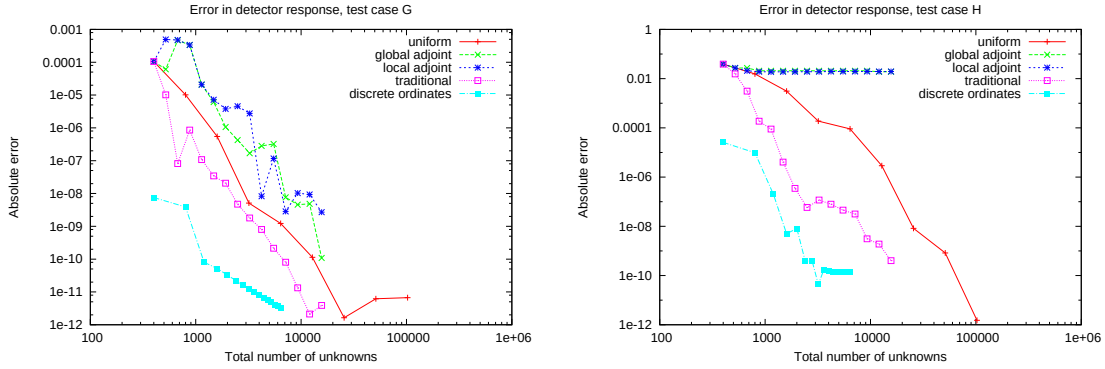
(d) *Test case F*

Figure 23: *Ratio of the estimated error over the 'exact' error. The exact error is determined by comparing the current detector response with that of the reference solution, computed with a very refined mesh.*

5.3 Shielding (cases G and H)

This section holds the results of the final two test cases, cases G and H. Both cases have an added shielding region in the middle of the domain compared to the previous cases. So we have a source region on the left of domain and for the neutrons to reach the detector region on the right of the domain they have to pass through a shielding region in the middle. Since few neutrons will pass through the domain to the detector it is important to refine the directions leading to the detector in order to obtain an accurate detector response. Furthermore we can see what the effect of discontinuities in the materials properties is.

Figure 24 shows the detector response error of the two shielding test cases. Besides the error of the two goal-oriented adaptive methods for test case H we generally see the same behaviour as in the separate source detector test cases. For the same reasoning as in the previous test cases we think this behaviour can be explained by looking at the adjoint. We refer to the next section for more on this.



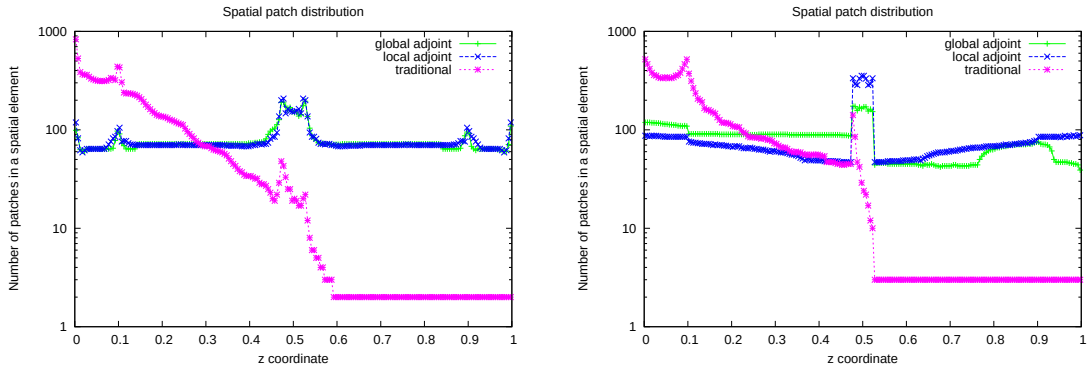
(a) *Test case G* (b) *Test case H*

Figure 24: *Error of the detector response of the two shielding test cases versus the total number of unknowns.*

The traditional refinement method performs best of the adaptive methods in both cases. By looking at Figure 25 we see a clear distinction in the spatial distribution of the traditional and goal-oriented methods. The goal-oriented method refines 'too much' on the right side of the shielding region, around the detector, to be efficient. Note that the distribution the traditional method produces is more efficient, but not necessarily the most efficient distribution. It is possible that more refinement on the right of the shielding region will result in an even more accurate detector response.

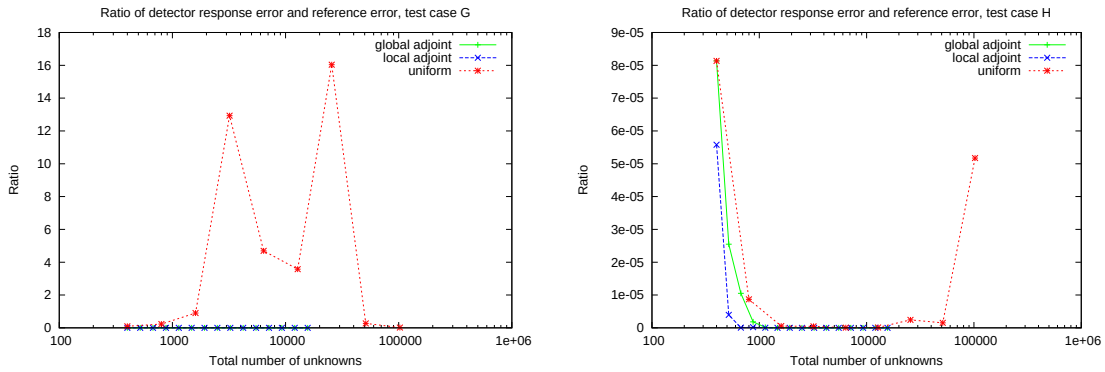
In the spatial distribution we can furthermore see that the discontinuities in the material properties are more difficult to approximate, according to all adaptive methods. Especially in the shielding region of Figure 25(b) we see that such a discontinuity needs more refinement than the discontinuity introduces by a source or detector region.

Also in these test cases we have an estimated error that is too small, as can be seen in Figure 26. In Section 6 an explanation for this result is presented.



(a) *Test case G* (b) *Test case H*

Figure 25: *Spatial distribution of patches of the two shielding test cases, G and H.*



(a) *Test case G* (b) *Test case H*

Figure 26: *Error ratio of the two shielding test cases. The ratio of the estimated error over the exact error is plotted versus the total number of unknowns.*

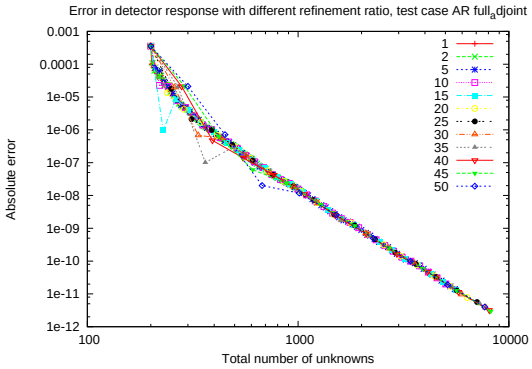
5.4 Effects of the Refinement Ratio

As final result we discuss the effect of the number of patches that is refined in each iteration. In this work a percentage of patches is refined in each iteration. In all cases up till now we used thirty per cent refinement per iteration.

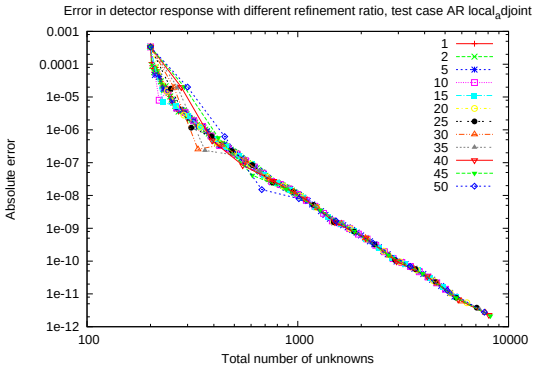
Figures with plots of the detector response error versus the total number of unknowns are once more presented, but now each plot holds the error for different percentages of only one adaptive method. We will only discuss some exemplary cases here.

Figures 27(a) and 28(a) show the error for the global adjoint method and Figures 27(b) and 28(b) are the plots of the detector response error of the local adjoint method. In general we can say that for the global and local adjoint methods the refinement percentage does not affect the error much. For test case C we see a little oscillation of the error and the lower percentages have a slightly smaller error in this oscillation. This difference is negligible.

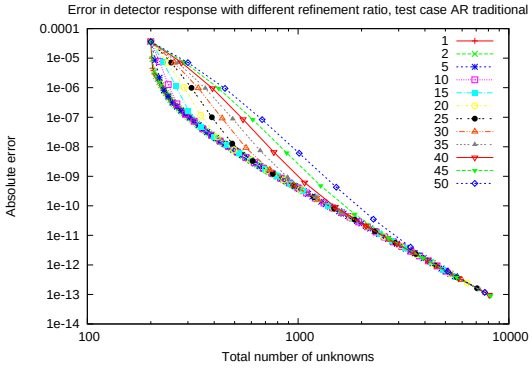
Turning to Figures 27(c) and 28(c), showing the error of the traditional method, we see there



(a) *Global adjoint criterion*



(b) *Local adjoint criterion*



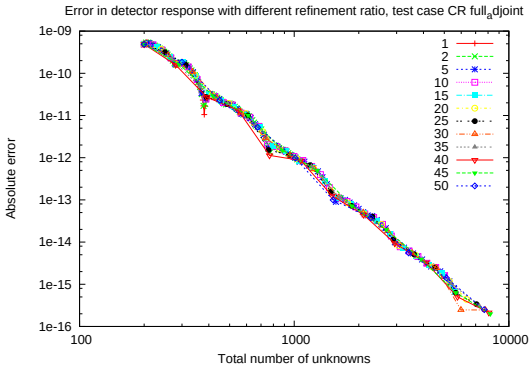
(c) *Traditional criterion*

Figure 27: *Error in the detector response versus the total number of unknowns for different refinement percentages of case A.*

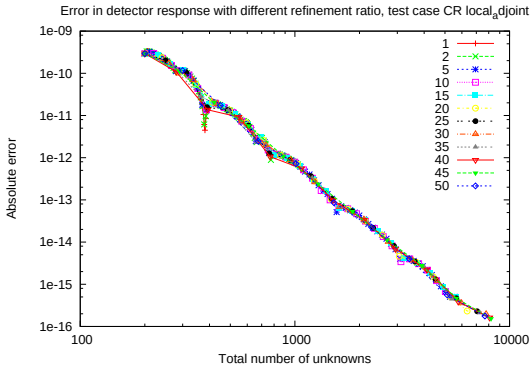
is more difference between the percentages here. In both figures we see that small percentages achieve a smaller error when less unknowns are used, while for large enough number of unknowns the percentage return to the level of the small percentages.

Smaller percentages will refine only the absolute necessary patches, thus achieving a high efficiency (small error for a given number of unknowns). Using larger percentages means that there will be patches that are refined that are not necessary, especially at the start of the iteration. At the start of the iteration cutting a patch into two new patches has a great effect on the error in the detector response. When many patches are already present in the problem, refining a patch does not assort a large effect. The patches that are refined in the beginning can become useful later in the refinement iteration, which is clearly what happens with the two cases presented here.

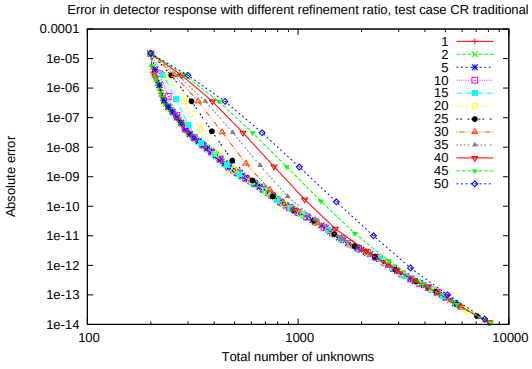
As a final remark we would like to point out that a method that produces a small error for a given number of unknowns is not necessarily a good choice. The plots presented here show the efficiency of a method in the cost of number of unknowns, not in the cost in memory or computational power. Refining more patches each iteration means one obtains an accurate detector response more quickly, in computation time, than with small percentages. Therefore in this work not one per cent was used, but thirty.



(a) *Global adjoint criterion*



(b) *Local adjoint criterion*



(c) *Traditional criterion*

Figure 28: *Error in the detector response versus the total number of unknowns for different refinement percentages of case C.*

6 Quality of the Error Estimate

In the previous section we have seen that the error estimate is of poor quality, while the refinement criterion produced a distribution of patches that in some cases is better than uniform refinement and in some cases it is not. In this section we will look more closely at the error estimate, as this cannot be used reliably as an indicator of the error.

To further investigate the behaviour of the error estimator we will look into the definition of the estimator. In Section 2.4 the error estimator is derived in terms of the transport equation, to be able to formulate an expression of the estimator that can be coded in an algorithm. Here we will present the same derivation, but now we will look more closely at the precise definition of the transport equation operator and its adjoint.

Starting with the definition of the problem at hand we define a space H in which the solution lies. The forward transport problem is then

$$\text{find} \quad \tilde{\phi} \in H \quad \text{such that} \quad L\tilde{\phi} = s \tag{6.1}$$

We will call this the strong formulation of the problem, as opposed to the weak definition of the problem, which is

$$\text{find} \quad \tilde{\phi} \in H \quad \text{such that} \quad \langle L\tilde{\phi}, \psi \rangle = \langle s, \psi \rangle \quad \forall \psi \in H \tag{6.2}$$

So far we use the full space H, i.e. these formulations are for the continuous problem. Discretisation means that we restrict ourselves to a space $V \subset H$, consisting of all functions in H that can be represented by all combinations of basis functions. The numerical results presented in Section 5 are, for example, with the space V the space of all piece-wise constant functions with a restriction on the size of the linear pieces. Before formulating the discretised problem we formulate the continuous restricted problem,

$$\text{find} \quad \tilde{\phi}_v \in V \quad \text{such that} \quad \langle L\tilde{\phi}_v, \psi_v \rangle = \langle s, \psi_v \rangle \quad \forall \psi_v \in V \tag{6.3}$$

Consequently we can represent the space V by vectors that lie in $\mathbb{R}^N$, which are the coefficients corresponding to a basis in V. Let P_h^v be a projector that projects these vectors onto the space V, i.e. the projector P_h^v is the expansion in basis functions,

$$P_h^v \phi_h = \sum_n \gamma(n) \phi_h(n) \tag{6.4}$$

where $\gamma(n)$ are the basis functions that span the space V. With this projection the discretised problem can be formulated as

$$\text{find} \quad \tilde{\phi}_h \in \mathbb{R}^N \quad \text{such that} \quad \langle L P_h^v \tilde{\phi}_h, P_h^v \psi_h \rangle = \langle s, P_h^v \psi_h \rangle \quad \forall \psi_h \in \mathbb{R}^N \tag{6.5}$$

Here it is important to note that we apply the Galerkin discretisation method by choosing the test functions ψ_h from the same space in which the solution lies, i.e. V or $\mathbb{R}^N$ with the projection P_h^v.

The detector response J is defined as the following inner product, which is the same expression as in Equation 2.42,

$$J = \langle \tilde{\phi}, \sigma_D \rangle \tag{6.6}$$

Now we can define the error in the detector response as

$$\Delta J = \langle \tilde{\phi}, \sigma_D \rangle - \langle \tilde{\phi}_v, \sigma_D \rangle \tag{6.7}$$

where the subscript v denotes that it is the restriction to the space $V \subset H$. By linearity we can also write this as

$$\Delta J = \langle \tilde{\phi} - \tilde{\phi}_v, \sigma_D \rangle \tag{6.8}$$

At this point in the derivation in Section 2.4 the adjoint equation and its solution are substituted in this expression. We will therefore define the adjoint problem in the same way as the forward problem. We start with the strong formulation,

$$\text{find} \quad \tilde{\phi}^* \in H \quad \text{such that} \quad L^* \tilde{\phi}^* = \sigma_D \tag{6.9}$$

Note that the space H is the same space as in the definition of the forward equation. The weak form of the adjoint equation is defined analogously,

$$\text{find} \quad \tilde{\phi}^* \in H \quad \text{such that} \quad \langle L^* \tilde{\phi}^*, \psi \rangle = \langle \sigma_D, \psi \rangle \quad \forall \psi \in H \tag{6.10}$$

Restricting the space H to the same space V again we arrive at

$$\text{find} \quad \tilde{\phi}^*_v \in V \quad \text{such that} \quad \langle L^* \tilde{\phi}^*_v, \psi_v \rangle = \langle \sigma_D, \psi_v \rangle \quad \forall \psi_v \in V \tag{6.11}$$

Finally, we can also write the adjoint problem in terms of N-dimensional vectors,

$$\text{find} \quad \tilde{\phi}^*_h \in \mathbb{R}^N \quad \text{such that} \quad \langle L^* P^v_h \tilde{\phi}^*_h, P^v_h \psi_h \rangle = \langle s, P^v_h \psi_h \rangle \quad \forall \psi_h \in \mathbb{R}^N \tag{6.12}$$

Again we applied the Galerkin discretisation procedure on the adjoint problem.

Now we can rewrite the expression for the error estimator in Equation 6.8 by substituting $L^* \phi^*$ for the right hand side of the adjoint problem, σ_D,

$$\Delta J = \langle \tilde{\phi} - \tilde{\phi}_v, L^* \tilde{\phi}^* \rangle \tag{6.13}$$

Equivalently we can write this estimator as

$$\Delta J = \langle L(\tilde{\phi} - \tilde{\phi}_v), \tilde{\phi}^* \rangle \tag{6.14}$$

We now arrived at Equation 2.64 of Section 2.4. The error in the detector is now expressed in terms of the bilinear form. Analogous to the derivation from Equation 2.64 to Equation 2.65, we will apply Galerkin orthogonality. We can identify $\langle L\tilde{\phi}, \psi_v \rangle$ as the bilinear form and $\langle s, \psi_v \rangle$ as the linear form of our problem. Galerkin orthogonality is the fact that the error $\tilde{\phi} - \tilde{\phi}_v$ is orthogonal to our restricted space V,

$$\begin{aligned}\langle L(\tilde{\phi}-\tilde{\phi}_v),\psi_v\rangle &= \langle L\tilde{\phi},\psi_v\rangle - \langle L\tilde{\phi}_v),\psi_v\rangle && (6.15)\\ &= \langle s,\psi_v\rangle - \langle s,\psi_v\rangle && (6.16)\\ &= 0 && (6.17)\end{aligned}$$

We can express Equation 2.65 therefore as

$$\Delta J = \langle L(\tilde{\phi}-\tilde{\phi}_v),\tilde{\phi}^*-\psi_v\rangle \tag{6.18}$$

where ψ_v can be any element of the space V. In particular we can choose ψ_v to be the solution of the weak restricted adjoint problem, $\tilde{\phi}^*_v$, resulting in

$$\Delta J = \langle L(\tilde{\phi}-\tilde{\phi}_v),\tilde{\phi}^*-\tilde{\phi}^*_v\rangle \tag{6.19}$$

The error can then be expressed in terms of the discrete forward problem as follows

$$\Delta J = \langle P^v_h s_h - LP^v_h\tilde{\phi}_h,\tilde{\phi}^*-P^v_h\tilde{\phi}^*_h\rangle \tag{6.20}$$

Here we assume that both the source s and the detector cross section σ_D obey the following relation

$$\begin{aligned}s &= P^v_h s_h && (6.21)\\ \sigma_D &= P^v_h\sigma_{D;h} && (6.22)\end{aligned}$$

Similarly, by continuing from Equation 6.14 instead of Equation 6.13 we can derive the following expression for the error, which we will call the dual form of the error,

$$\Delta J = \langle\tilde{\phi}-P^v_h\tilde{\phi}_h,P^v_h\sigma_{D;h}-L^*P^v_h\tilde{\phi}^*_h\rangle \tag{6.23}$$

Note that we again used the Galerkin orthogonality property, however now on the adjoint problem.

On the space H we have an expression for the detector response J derived from the adjoint problem. This property is sometimes called the duality relation and can be proven by substituting the adjoint problem in the following way

$$\begin{aligned}J &= \langle\tilde{\phi},\sigma_D\rangle && (6.24)\\ &= \langle\tilde{\phi},L^*\tilde{\phi}^*\rangle && (6.25)\\ &= \langle L\tilde{\phi},\tilde{\phi}^*\rangle && (6.26)\\ &= \langle s,\tilde{\phi}^*\rangle && (6.27)\end{aligned}$$

The duality relation also holds on the restricted space V, which can be shown by rewriting Equation 6.13 as

$$\begin{aligned}\Delta J &= \langle \tilde{\phi} - \tilde{\phi}_v, L^* \tilde{\phi}^* \rangle && (6.28)\\ &= \langle L(\tilde{\phi} - \tilde{\phi}_v), \tilde{\phi}^* \rangle && (6.29)\\ &= \langle s - L\tilde{\phi}_v, \tilde{\phi}^* \rangle && (6.30)\\ &= \langle s, \tilde{\phi}^* \rangle - \langle L\tilde{\phi}_v, \tilde{\phi}^* \rangle && (6.31)\end{aligned}$$

Furthermore, by rewriting Equation 6.13 by applying the Galerkin orthogonality principle twice (on the forward and adjoint problem respectively) we obtain

$$\begin{aligned}\Delta J &= \langle L(\tilde{\phi} - \tilde{\phi}_v), \tilde{\phi}^* \rangle && (6.32)\\ &= \langle L(\tilde{\phi} - \tilde{\phi}_v), \tilde{\phi}^* - \tilde{\phi}_v^* \rangle && (6.33)\\ &= \langle \tilde{\phi}, L^*(\tilde{\phi}^* - \tilde{\phi}_v^*) \rangle && (6.34)\\ &= \langle s, \tilde{\phi}^* - \tilde{\phi}_v^* \rangle && (6.35)\\ &= \langle s, \tilde{\phi}^* \rangle - \langle s, \tilde{\phi}_v^* \rangle && (6.36)\end{aligned}$$

Combining these two expressions for the error in the detector response we see that we must have equivalence between two terms, resulting in

$$\langle s, \tilde{\phi}_v^* \rangle = \langle \tilde{\phi}_v, \sigma_D \rangle \tag{6.37}$$

which is the duality relation on the space V.

To this point no approximation is made, the expression we have for the error in the adjoint is exact. When the exact solution of respectively the adjoint or forward problem is available we can compute the error made by restricting the space H to the space V. However, in general this exact solution is not available, which makes an approximation of thi solution necessary. The approximation we choose is perform a refinement on all basis functions of the space V, call this space U. We then have the following nested spaces,

$$V \subset U \subseteq H \tag{6.38}$$

Let us denote the adjoint solution in the space U by $\tilde{\phi}_{h+1}^*$ in the discretised problem formulation. We then have $\tilde{\phi}_h^* \in \mathbb{R}^N$ and $\tilde{\phi}_{h+1}^* \in \mathbb{R}^M$, with $M \geq N$, more specifically in the case of the linear basis functions we have that $M = 2N$. Analogously we define the solution of the discretised forward problem on the space U. Substituting this approximation in the discrete form of the error we obtain

$$\begin{aligned}\Delta J &\approx \langle P_h^v s_h - L P_h^v \tilde{\phi}_h, P_{h+1}^u \tilde{\phi}_{h+1}^* - P_h^v \tilde{\phi}_h^* \rangle && (6.39)\\ \Delta J &\approx \langle P_{h+1}^u \tilde{\phi}_{h+1} - P_h^v \tilde{\phi}_h, P_h^v \sigma_{D;h} - L^* P_h^v \tilde{\phi}_h^* \rangle && (6.40)\end{aligned}$$

In the results of the linear basis functions in Section 5 the version in Equation 6.39 is used as error estimate. Therefore we will look into this expression to find the cause of the underestimation of the error estimator.

Since the ratio of the error estimator over the reference error is very small, we know the estimate is too small. The inner product in Equation 6.39 will become small when the two functions in the arguments are orthogonal to each other. We can distinguish four cases in which this happens for different reasons.

1. The residual of the forward problem $s - LP_h^v\tilde{\phi}_h$ decreases.
2. The exact adjoint error $\tilde{\phi}^* - P_h^v\tilde{\phi}_h^*$ decreases, hence the approximate adjoint error will most likely also decrease.
3. There is a possibility that the approximate adjoint error $P_{h+1}^u\tilde{\phi}_{h+1}^* - P_h^v\tilde{\phi}_h^*$ decreases while the exact adjoint error does not decrease. This could be due to stagnation between consecutive refinements.
4. Finally, the residual of the forward problem can be orthogonal to the approximate adjoint error, while neither of these are zero.

Several tests were performed to determine which of the above reasons is the cause of the poor quality of the error estimator. The step where the adjoint solution $P_h^V\tilde{\phi}_h^*$ is added to the error expression by applying Galerkin orthogonality is not necessary to arrive at a correct expression for the error. We verified that leaving out this term leads to similar results, i.e. also to an error estimator with poor quality. This means that option 3) is not the cause of the poor quality, as there cannot be stagnation in the approximate error of the adjoint when it is not used. Figure 29 shows a plot of the error estimate of test case D without the term $P_h^V\tilde{\phi}_h^*$ in the second argument of the inner product, where the quality of the error estimate is still poor.

Also $P_{h+1}^u\tilde{\phi}_{h+1}^*$ is not small, as it is the adjoint solution of our problem.

Furthermore we computed the norm of the forward residual vector and together with the norm of the approximate adjoint, which are shown in Figure 30 for test case D. Considering Figures 21(b) and 23(b) we can conclude that the residual is too large to account for the small value of the inner product, reducing the possible options to numbers 2) and 4).

Furthermore we conclude that reason 2) cannot be the cause of the problem with the error estimator, since this would mean that the discrete adjoint solution $P_h^v\tilde{\phi}_h^*$ is very accurate, which in turn would mean that the detector response computed by the duality relation (see Equation 6.27) is very accurate. This is not the case, as the adjoint detector response sometimes stagnates for several refinement iterations.

Only option 4) now remains, hence the forward residual and approximate adjoint error are orthogonal, but neither are zero. We also conclude that, by Galerking orthogonality, the forward residual is orthogonal to the adjoint approximation $P_{h+1}^u\tilde{\phi}_{h+1}^*$. The most likely cause for this is the introduction of the approximation in Equation 6.39.

The residual is computed in the space V and the adjoint approximation in the space U, where U is obtained by refining V once. In the results of Section 5 and of [4] piece-wise linear and piece-wise constant angular basis functions were used respectively. The spatial basis functions were in both cases piece-wise linear. We will concentrate on the angular basis functions, as this is the part of the domain where refinement takes place. The basis functions of the space U have

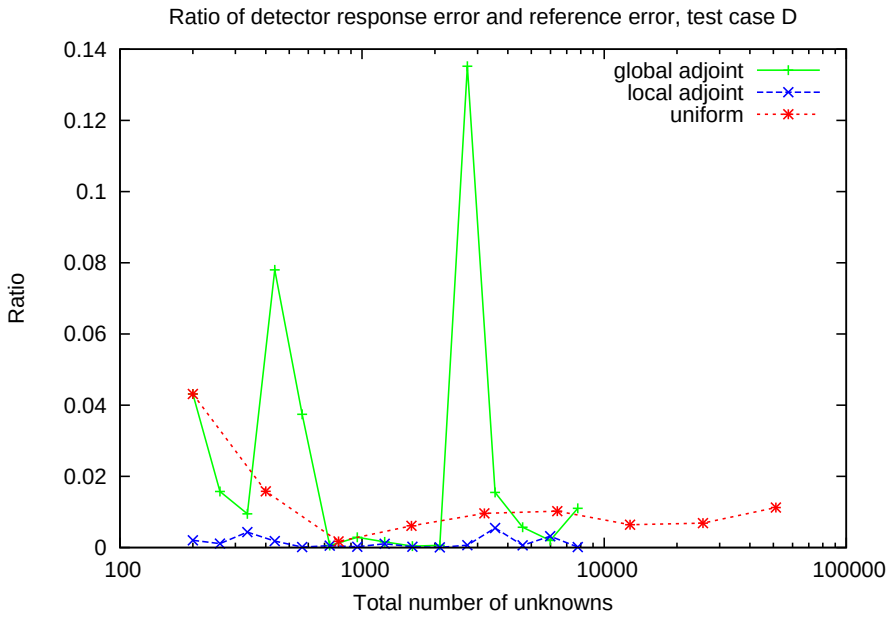


Figure 29: *Ratio of the error estimate over the reference error without applying the Galerkin orthogonality to the error estimate. The quality of this error estimate is also poor, meaning that stagnation in the adjoint error cannot be the cause of the poor quality.*

half the support of the basis functions in V. These spaces are in a sense very similar, which could cause stagnation in refinement.

The exact error in the detector response $\tilde{\phi}^* - P_h^v \tilde{\phi}_h^*$ lies in the space H, we could write this error as

$$\begin{aligned} \tilde{\phi}^* - P_h^v \tilde{\phi}_h^* &= e && (6.41) \\ &= e^{||} + e^{\perp} && (6.42) \end{aligned}$$

where $e^{||} \in V$ and $e^{\perp}$ orthogonal to V. The approximate error can then be written as

$$\begin{aligned} P_{h+1}^u \tilde{\phi}_{h+1}^* - P_h^v \tilde{\phi}_h^* &= e_h && (6.43) \\ &= e_h^{||} + e_h^{\perp} && (6.44) \end{aligned}$$

where again $e_h^{||} \in V$ and $e_h^{\perp}$ orthogonal to V.

The approximation of the adjoint solution in the derivation of the error estimate is an approximation of the form $e \approx e_h$. However, the part of e_h that lies in the space V, i.e. $e^{||}$, does actually not contribute to the error estimate, as Galerkin orthogonality applies to all functions in V. Therefore an approximation of the error e that leads to an accurate error estimate must approximate $e^{\perp}$ well, this leads to the following condition

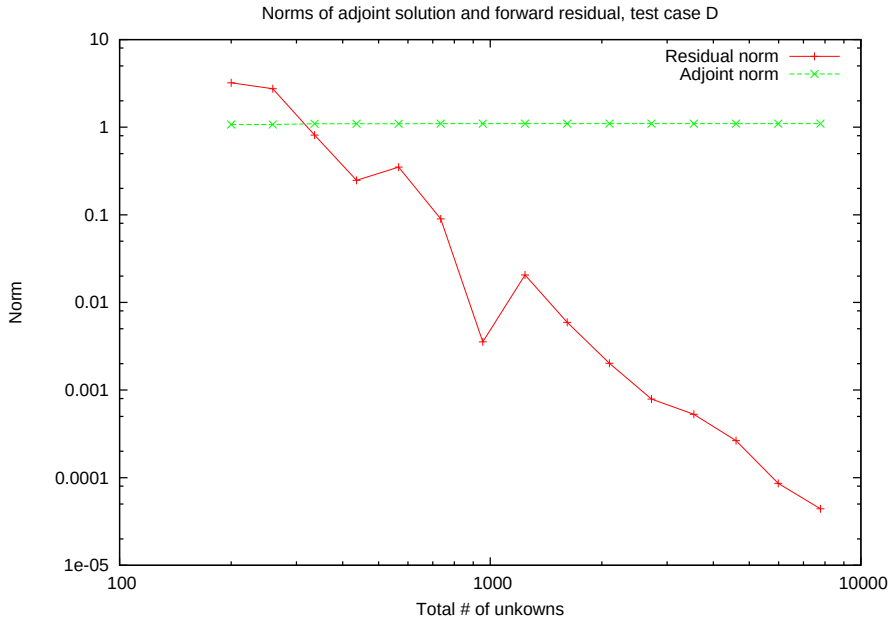


Figure 30: *The norms of the approximate adjoint and forward residual of test case D are plotted against the total number of unknowns. This shows that the error estimate, which is the inner product of the two, is not small because the adjoint or the forward residual is small.*

$$e^{\perp} \approx e_h^{\perp} \tag{6.45}$$

Furthermore, we still want that the the error e is approximated well by e_h, which leads us to the following two conditions for the approximation of $\tilde{\phi}^*$

$$e^{\perp} \approx e_h^{\perp} \tag{6.46}$$

$$e^{||} \approx e_h^{||} \tag{6.47}$$

which implies $e \approx e_h$. Note that the converse is not true. For example, assume that $e^{||} >> e^{\perp}$, in which case the perpendicular part of the error is not important in the approximation of the whole error e.

To test this hypothesis a more refined approximation of the exact adjoint could be used, for example refine each basis function in the space V twice to obtain the basis functions of U. This approximation should lead to an error estimate of better quality, however the computational costs will go up rather quickly, as they grow exponentially with the number of refinement levels.

Another way to test this hypothesis, that leads to cheaper algorithms is to use different basis functions for the adjoint problem, that is, solve the adjoint problem not on the restricted space V, but on a different restricted space V'. When piece-wise polynomials span the the space V, higher order polynomials could be chosen to span the space V', for example solve the adjoint approximation on piece-wise quadratic basis functions. Refinement in basis function order is sometimes referred to as p refinement, as opposed to refinement in mesh, which is then referred to as h refinement. p refinement has the advantage of the computational costs scaling linear with the number of refinement levels, however, it is not always feasible to use higher order polynomials or other higher order basis functions.

7 Notes on the Boltzmann-Fokker-Planck Equation

One of the goals of this work was to investigate whether the discretisation method that is used with the transport equation can be used for the Boltzmann-Fokker-Planck equation. In this section the results of that are presented. Since the loss of symmetry in the derivation of the discrete adjoint transport operator the investigation is not very thorough. However, we have discretised the equation and a proposal for solving the discrete system of equations. This loss of symmetry is also present in the BFP equation, since the terms from which this results are also present in this equation.

7.1 Discretisation of the BFP Equation

The Boltzmann-Fokker-Planck equation reads

$$\mu\frac{\partial}{\partial x}\phi(x,\hat{\mathbf{\Omega}}) + \sigma_t\phi(x,\mu) + \frac{\alpha}{2}\frac{\partial}{\partial\mu}\left[(1-\mu^2)\frac{\partial}{\partial\mu}\phi(x,\mu)\right] = \frac{\sigma_s}{2}\int_{-1}^{1}\phi(x,\mu)d\mu \tag{7.1}$$

which is the transport equation as in Equation 2.8, with two extra terms on the left hand side [7]. These terms describe a diffusion process in the angular variables. The parameter α is called the momentum transfer, which determines the rate at which the direction of the neutrons diffuses over the angular coordinates.

To discretise this equation we use the same discretisation for all terms as already obtained for the transport equation, see Section 4.1. Below we will only discretise the extra terms, using the same method. To that end we start with rewriting the terms as

$$\begin{aligned}\frac{\alpha}{2}\frac{\partial}{\partial\mu}\left[(1-\mu^2)\frac{\partial}{\partial\mu}\phi(z,\mu)\right] &= \frac{\alpha}{2}\left[\frac{\partial}{\partial\mu}(1-\mu^2)\frac{\partial}{\partial\mu}\phi(z,\mu) + (1-\mu^2)\frac{\partial^2}{\partial\mu^2}\phi(z,\mu)\right] &(7.2)\\ &= \frac{\alpha}{2}\left[-2\mu\frac{\partial}{\partial\mu}\phi(z,\mu) + (1-\mu^2)\frac{\partial^2}{\partial\mu^2}\phi(z,\mu)\right] &(7.3)\end{aligned}$$

Now we can substitute a function from our test space for the angular flux in this term. The function is a double sum of elements and patches on the elements, which can be expressed as

$$\phi_h = \sum_e\sum_p \phi_{e,p}\gamma_e G_{e,p} \tag{7.4}$$

Substitution of this expression and integration over the whole spatial-angular space leads to

$$\frac{\alpha}{2}\int_x 2\pi\int_{-1}^{1}\gamma_e G_{e,p}\left[(1-\mu^2)\sum_{e'}\sum_{p'}\gamma_{e'}\frac{\partial^2 G_{e',p'}}{\partial\mu^2} - 2\mu\sum_{e'}\sum_{p'}\frac{\partial G_{e',p'}}{\partial\mu}\right]d\mu dx \tag{7.5}$$

In this expression there is a second order derivative of one of the linear basis functions. We want at most a first order derivative with these basis functions, since otherwise the term would vanish. Therefore we use partial integration to obtain

$$\begin{aligned}\tfrac{\alpha}{2}\int_x 2\pi\int_{-1}^{1} \quad & \gamma_e G_{e,p}\left[(1-\mu^2)\sum_{e'}\sum_{p'}\gamma_{e'}\frac{\partial^2 G_{e',p'}}{\partial\mu^2} - 2\mu\sum_{e'}\sum_{p'}\frac{\partial G_{e',p'}}{\partial\mu}\right] d\mu dx = \\ & -\frac{\alpha}{2}\int_x 2\pi\int_{-1}^{1}(1-\mu^2)\sum_{e'}\sum_{p'}\gamma_e\gamma_{e'}\frac{\partial G_{e,p}}{\partial\mu}\frac{\partial G_{e',p'}}{\partial\mu}d\mu dx \\ & +\frac{\alpha}{2}\int_x\left[(1-\mu^2)\gamma_e G_{e,p}\sum_{e'}\sum_{p'}\gamma_{e'}\frac{\partial G_{e',p'}}{\partial\mu}\right]_{\mu=\mu_-}^{\mu+} dx \end{aligned} \tag{7.6}$$

The next step is to see how each term is added to the large matrix P_h. We will start with the first term in 7.6, which is the equivalent of the volumetric streaming term in the discretisation of the streaming term of the transport equation. However, it is now a volumetric diffusion term, since we have a second order derivative in this term. The block matrix for one patch, which are added as block diagonal matrix to P_h, looks like

$$\begin{aligned}&\alpha\pi\int_x\int_{-1}^{1}(1-\mu^2)\sum_{e'}\sum_{p'}\gamma_e\gamma_{e'}\frac{\partial G_{e,p}}{\partial\mu}\frac{\partial G_{e',p'}}{\partial\mu}d\mu dx \\ &= \alpha\pi\int_x\int_{-1}^{1}(1-\mu^2) \\ &\begin{bmatrix} \gamma_{e_1}\gamma_{e_1}G'_{e_1,p_1}G'_{e_1,p_1} & \gamma_{e_1}\gamma_{e_1}G'_{e_1,p_1}G'_{e_1,p_2} & \gamma_{e_1}\gamma_{e_2}G'_{e_1,p_1}G'_{e_2,p_1} & \gamma_{e_1}\gamma_{e_2}G'_{e_1,p_1}G'_{e_2,p_2} \\ \gamma_{e_1}\gamma_{e_1}G'_{e_1,p_2}G'_{e_1,p_1} & \gamma_{e_1}\gamma_{e_1}G'_{e_1,p_2}G'_{e_1,p_2} & \gamma_{e_1}\gamma_{e_2}G'_{e_1,p_2}G'_{e_2,p_1} & \gamma_{e_1}\gamma_{e_2}G'_{e_1,p_2}G'_{e_2,p_2} \\ \gamma_{e_2}\gamma_{e_1}G'_{e_2,p_1}G'_{e_1,p_1} & \gamma_{e_2}\gamma_{e_1}G'_{e_2,p_1}G'_{e_1,p_2} & \gamma_{e_2}\gamma_{e_2}G'_{e_2,p_1}G'_{e_2,p_1} & \gamma_{e_2}\gamma_{e_2}G'_{e_2,p_1}G'_{e_2,p_2} \\ \gamma_{e_2}\gamma_{e_1}G'_{e_2,p_2}G'_{e_1,p_1} & \gamma_{e_2}\gamma_{e_1}G'_{e_2,p_2}G'_{e_1,p_2} & \gamma_{e_2}\gamma_{e_2}G'_{e_2,p_2}G'_{e_2,p_1} & \gamma_{e_2}\gamma_{e_2}G'_{e_2,p_2}G'_{e_2,p_2} \end{bmatrix} \\ &\qquad\qquad \begin{bmatrix}\phi_{e_1,p_1}\\ \phi_{e_1,p_2}\\ \phi_{e_2,p_1}\\ \phi_{e_2,p_2}\end{bmatrix} d\mu dx \end{aligned} \tag{7.7}$$

$$= \alpha\pi\Delta x\frac{1-\mu_+^2-\mu_-\mu_+-\mu_-^2}{\Delta\mu}\begin{bmatrix} \frac{1}{3} & -\frac{1}{3} & \frac{1}{6} & -\frac{1}{6} \\ -\frac{1}{3} & \frac{1}{3} & -\frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & -\frac{1}{6} & \frac{1}{3} & -\frac{1}{3} \\ -\frac{1}{6} & \frac{1}{6} & -\frac{1}{3} & \frac{1}{3} \end{bmatrix}\begin{bmatrix}\phi_{e_1,p_1}\\ \phi_{e_1,p_2}\\ \phi_{e_2,p_1}\\ \phi_{e_2,p_2}\end{bmatrix} \tag{7.8}$$

Now we can turn to the second term in Equation 7.6, which is equivalent to the boundary streaming term for the streaming term in the transport equation. Since there is no upwind direction we will take at both sides of the patch the neighbouring patch angular flux values. This results in the following blocks matrices being added to diagonal bands of P_h

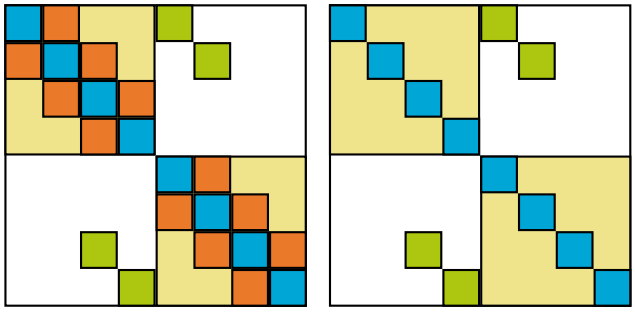
$$
\begin{aligned}
&\frac{\alpha}{2}\int_x \left[(1-\mu^2)\gamma_e G_{e,p}\sum_{e'}\sum_{p'}\gamma_{e'}\frac{\partial G_{e',p'}}{\partial \mu}\right]_{\mu=\mu_-}^{\mu_+} \\
&= \frac{\alpha}{2}\int_x (1-\mu_+^2) \\
&\begin{bmatrix}
0 & 0 & 0 & 0 \\
\gamma_{e_1}\gamma_{e_1}G'_{e_1,p_2}G'_{e_1,p_1} & \gamma_{e_1}\gamma_{e_1}G'_{e_1,p_2}G'_{e_1,p_2} & \gamma_{e_1}\gamma_{e_2}G'_{e_1,p_2}G'_{e_2,p_1} & \gamma_{e_1}\gamma_{e_2}G'_{e_1,p_2}G'_{e_2,p_2} \\
0 & 0 & 0 & 0 \\
\gamma_{e_2}\gamma_{e_1}G'_{e_2,p_2}G'_{e_1,p_1} & \gamma_{e_2}\gamma_{e_1}G'_{e_2,p_2}G'_{e_1,p_2} & \gamma_{e_2}\gamma_{e_2}G'_{e_2,p_2}G'_{e_2,p_1} & \gamma_{e_2}\gamma_{e_2}G'_{e_2,p_2}G'_{e_2,p_2}
\end{bmatrix}
\begin{bmatrix}
\phi_{e_1,p_1} \\ \phi_{e_1,p_2} \\ \phi_{e_2,p_1} \\ \phi_{e_2,p_2}
\end{bmatrix} dx \\
&+\frac{\alpha}{2}\int_x (1-\mu_-^2) \\
&\begin{bmatrix}
\gamma_{e_1}\gamma_{e_1}G'_{e_1,p_1}G'_{e_1,p_1} & \gamma_{e_1}\gamma_{e_1}G'_{e_1,p_1}G'_{e_1,p_2} & \gamma_{e_1}\gamma_{e_2}G'_{e_1,p_1}G'_{e_2,p_1} & \gamma_{e_1}\gamma_{e_2}G'_{e_1,p_1}G'_{e_2,p_2} \\
0 & 0 & 0 & 0 \\
\gamma_{e_2}\gamma_{e_1}G'_{e_2,p_1}G'_{e_1,p_1} & \gamma_{e_2}\gamma_{e_1}G'_{e_2,p_1}G'_{e_1,p_2} & \gamma_{e_2}\gamma_{e_2}G'_{e_2,p_1}G'_{e_2,p_1} & \gamma_{e_2}\gamma_{e_2}G'_{e_2,p_1}G'_{e_2,p_2} \\
0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\phi_{e_1,p_1} \\ \phi_{e_1,p_2} \\ \phi_{e_2,p_1} \\ \phi_{e_2,p_2}
\end{bmatrix} dx
\end{aligned}
\tag{7.9}
$$

$$
\begin{aligned}
&= \frac{\alpha}{2}(1-\mu_+^2)\begin{bmatrix}
0 & 0 & 0 & 0 \\
\frac{1}{3}\Delta x & \frac{1}{3}\Delta x & \frac{1}{6}\Delta x & \frac{1}{6}\Delta x \\
0 & 0 & 0 & 0 \\
\frac{1}{6}\Delta x & \frac{1}{6}\Delta x & \frac{1}{3}\Delta x & \frac{1}{3}\Delta x
\end{bmatrix}
\begin{bmatrix}
\phi_{e_1,p_1} \\ \phi_{e_1,p_2} \\ \phi_{e_2,p_1} \\ \phi_{e_2,p_2}
\end{bmatrix} \\
&+\frac{\alpha}{2}(1-\mu_-^2)\begin{bmatrix}
\frac{1}{3}\Delta x & \frac{1}{3}\Delta x & \frac{1}{6}\Delta x & \frac{1}{6}\Delta x \\
0 & 0 & 0 & 0 \\
\frac{1}{6}\Delta x & \frac{1}{6}\Delta x & \frac{1}{3}\Delta x & \frac{1}{3}\Delta x \\
0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\phi_{e_1,p_1} \\ \phi_{e_1,p_2} \\ \phi_{e_2,p_1} \\ \phi_{e_2,p_2}
\end{bmatrix}
\end{aligned}
\tag{7.10}
$$

7.2 Solving the BFP Equation

To compare the matrix of the operator P_h to that of L_h we present in Figure 31 both matrices. We see in the transport matrix, Figure 31(b), the diagonal blocks in blue, which determine the patch internal calculations. The green blocks represent the coupling between spatial elements, in other words, they represent the neutrons streaming from one element to the next. The large yellow blocks represent the scatter term, these blocks are larger because the scatter term couples all patches within an element with each other.

Turning to the matrix of the BFP equation, Figure 31(a), we see there are extra blocks in this matrix. The orange blocks represent the extra coupling between patches due to the diffusion in the angular component. This coupling is therefore only present between patches of the same spatial element.



(a) *BFP equation* (b) *Transport equaiton*

Figure 31: *Structure of the matrices of the transport equation and the Boltzmann-Fokker-Planck equation. The structure of the BFP equation is different because the patches are now coupled. This coupling originates from particles that diffuse in the angular direction. The orange blocks are the blocks that couple the patches.*

The source iteration that is used for the transport equation, see Section 3.2.1, can still be used for this matrix, since in that part no changes were introduced. However, the sweep iteration cannot be applied any more, since there is coupling between patches in such a way that we cannot make a lower triangular matrix by solving the system in a certain order.

A Gauss Seidel iteration can be applied to the BFP equation, which is similar to the sweep iteration. Before applying this iteration the matrix will need to be reordered in the same way as with the sweep iteration. We then end up with an iteration that is 'almost direct', in the sense that the neighbour angular flux values of the stream term and one of two neighbour angular flux values of the diffusive term are up-to-date and only one of the two flux values of the diffusive term is from the previous iteration. The stability of this method will have to be investigated.

Another possibility for solving the system of equations obtained from the discretisation of the BFP equation is using Krylov subspace methods. The Krylov subspace of a matrix A of size $n \times n$ and a vector b of size n is a subspace of the n-sized vectors like

$$K_r = \text{span}\left(b, A^1 b, A^2 b, \ddots, A^{r-1} b\right) \tag{7.11}$$

where r is the order of the Krylov subspace. Several iterative methods have been constructed that make use of this space and do not use the full form of matrix A. The matrix is only used in a multiplication with a vector, removing the need to construct the full matrix. This can be applied to the BFP equation, since multiplying P_h by a vector can be easily done. P_h is a sparse matrix and is never constructed in the code that was written for this project. In the same way an efficient multiplication algorithm for sparse matrices can be used.

8 Discussion

Here the main conclusions are presented, which can all be found in the main text. However, this gives an overview of the most important conclusions and puts them into perspective. We will roughly present the conclusions grouped per section and by relevance to each other. Besides the conclusion possible future work is also discussed here. The future work can basically be divided into three domains, implementing this method for three dimensional problems (possibly combining spatial adaptivity with angular adaptivity), investigating the loss of symmetry between the discrete adjoint operator obtained through reversing discretising and deriving adjoint and finally investigating and implementing the Boltzmann-Fokker-Planck equation.

8.1 Conclusions

In the first section that is new for this report, we derived the full form of the matrix and have used this to show what the algorithm that solves the equation does. Essentially the source iteration combined with the sweep iteration is a block Jacobi iteration. This can be done since the matrix can be split in three parts, the diagonal blocks, the larger scatter diagonal blocks and the streaming blocks that form a band. By moving specific blocks to the right hand side, or reordering the unknowns in the matrix, we can solve the transport part directly and iterate only the scatter term.

Furthermore we have looked at the definition of the discrete adjoint problem that is used for error estimation. There are two routes to follow to the discrete adjoint problem, either discretising the continous adjoint or taking the adjoint operator of the discretised forward problem. It turns out the symmetry between these routes is broken for transport equation operator. The transport term of the equation has a different sign in the two cases. Since the error estimation criterion only uses the discretised continuous adjoint problem this loss of symmetry will not affect the estimate, however, it is possible to formulate a different estimate using the discrete adjoint.

We also looked at the difference in quality of the two goal-oriented adaptive methods, the global and local adjoint approximations. By defining all problems in discrete terms and substituting them in the error estimate we saw that the ordering in magnitude of the global and local adjoint does not change. Successive pairs of unknowns can switch in the ordering in magnitude, but that does not matter, since this unknown represents the same patch but a different spatial location. By a small perturbation in magnitude of the adjoint vector we saw that this can greatly affect the total error estimate.

Discretisation of the transport equation using linear basis functions in the angular domain, linear patches, is not much different from using constant basis functions. The matrix will be larger, since there are more unknowns for the same number of patches, but the structure remains the same. Therefore we can keep using the same algorithm for solving the equation.

The other question involved in discretising with linear angular basis functions was the continuity between two elements with a different angular refinement. We can write the continuity relations in two conditions, a hard condition and a soft condition. The hard condition ensures the continuity of the number of neutrons between elements, while the soft condition can be formulated as a minimization problem that lets the angular fluxes of the two elements be similar, in some norm. By choosing in what norm the difference between the elements must be minimized, we also

choose what interpolation rules must be used. The orthogonal projection of basis functions leads to the minimized difference in the two-norm.

After implementation of the linear patches we presented the results of the same test cases as were used for the constant patches. In general we saw that linear patches provide a much better approximation of the angular flux, since the convergence is generally fourth order. We also see this in the convergence of the traditional criterion, as this is in most cases much better than with the constant patches.

Considering the distribution of patches the goal-oriented methods produces, we can conclude that they work as expected. The spatial distribution are somewhat similar to the distributions of the constant patches and are what one would expect. As with the constant patches the performance of the methods in the error versus the number of unknowns varies between the test cases. The linear patches do however show a significant decrease in error relative to the discrete ordinates method, which indicates that linear patches can approximate the angular flux well.

Although the results in general show the refinement criterion is an effective way of refinement in some cases we see that uniform refinement results in a more efficient calculation of the detector response. Efficiency in this case means acquiring an accurate detector response with a smaller total number of unknowns. This is not unexpected when comparing it to the constant patch results, where this was also found in some test cases. The cause of this is likely similar to those of the poor quality of the error estimate.

The quality of the error estimate is very poor. Only with uniform refinement in the homogeneous test case B is the error estimate accurate. The estimator cannot be used as a reliable indicator of the error, for example for a stopping criterion. We have identified the cause for this poor quality in the approximation of the exact adjoint that is used in the estimate. In order to obtain an accurate error estimate we need an approximation of the error in the adjoint solution that does not lie in the same space as we are solving the forward problem in, as only the orthogonal part will lead to a contribution in the error estimate. The hypothesis is that stagnation takes place because the approximation of the adjoint on a level deeper does not contain a large orthogonal component. This could be tested by using a different approximation of the adjoint, most likely by solving the adjoing with higher order basis functions.

We think this is also the cause of the problem with the decision of refinement in some test cases, in particular the cases where uniform refinement provides more accurate results than goal-oriented adaptive refinement. As the refinement decision is a local form of the error estimate the same approximation of the exact adjoint is made. However, besides this the effects of boundary residuals could also play a role. This should be investigated further.

Finally we looked at the Boltzmann-Fokker-Planck equation, which was one of the goals at the start of the project. Due to the unexpected discovery that the derivation of the adjoint is not symmetrical, not much time was spend on this part. However, we did derive the full matrix operator of the BFP equation and looked at possible ways of solving it. The difference with the transport equation is that extra bands are added within an element, making it impossible to use the sweep iteration in the same way, in other words, we cannot reorder the matrix in such a way that the bands form a lower or upper diagonal matrix. A solution to this is to use a Gauss-Seidel iteration as sweep, which effectively means that we use outdated values for the angular flux on one of the neighbouring patches. The other possibility is to use Krylov subspace methods, as the full matrix is not needed in these methods.

8.2 Future Work

As in the physics thesis, one of the possible expansions is to implement the methods for three dimensional problems. One has to take into account that the difference in the adjoint will remain in this case. Another area of investigation is the combination with spatial refinement, where the central question is on the choice of refinement. Are we completely free to choose where to refine or are there restrictions? Another question is, do we have a choice in refining either in the spatial domain or in the angular domain, or are these refinements coupled?

The hypothesis that the poor quality of the error estimate arises from an inaccurate approximation of the exact adjoint has to be verified by implementing an approximation that takes into account the orthogonal part of the exact adjoint to the current restricted space. Also the effects of this on the refinement criterion were not yet fully investigated.

In this work we did not implement the BFP equation, however we explored possible methods of solving this equation. There are two possible ways to implement the equation, either by a Gauss-Seidel iteration, which uses outdated angular flux values of one of the neighbouring patches. The other possibility is using Krylov subspace methods, where the full matrix is not needed.

References

[1] Christian Blatter. *Wavelets: a primer*. CRC Press, January 1999. ISBN 1-5688-1095-4.

[2] A.G. Buchan, C.C. Pain, M.D. Eaton, R.P. Smedley-Stevenson, and A.J.H. Goddard. Linear and quadratic octahedral wavelets on the sphere for angular discretisations of the Boltzmann transport equation. *Annals of Nuclear Energy*, 32:1224–1273, January 2005.

[3] James J. Duderstadt and Louis J. Hamilton. *Nuclear Reactor Analysis*. John Wiley and Sons, 1942. ISBN 0-471-22363-8.

[4] Dion J Koeze. Goal-oriented angular adaptive neutron transport using a spherical discontinuous galerkin method. Master thesis, Delft University of Technology, December 2011.

[5] Danny Lathouwers. Goal oriented spatial adaptivity for the S_N equations on unstructured triangular meshes. *Annals of Nuclear Energy*, 38:1373–1381, January 2011.

[6] E.E. Lewis and W.F. Miller. *Computational methods of neutron transport*. Wiley-Interscience, 1984. ISBN 0-471-09245-2.

[7] Jim E. Morel. Boltzmann-Fokker-Planck calculations using standard discrete-ordinates codes. In *Computational Methods in High Temperature Physics*. Los Alamos National Laboratory, February 1987.

[8] Joseph C. Stone and Marvin L. Adams. Adaptive discrete-ordinates algorithms and strategies. In *Mathematics and computation, supercomputing, reactor physics and nuclear and biological applications*, page 5/12, Palais des Papes, Avignon, France, September 2005. American Nuclear Society (ANS).

[9] Todd A. Warening, John M. McGhee, Jim E. Morel, and Shawn D. Pautz. Discontinuous finite element S_N methods on three-dimensional unstructured grids. *Nuclear Science and Engineering*, 138:256–268, 2001.

[10] James S. Warsa. A continuous finite element-based, discontinuous finite element method for S_N transport. *Nuclear Science and Engineering*, 160:385–400, March 2008.

A Test Cases

Here we present the test cases used throughout this work, with specified geometries and material properties. For each test case we supply a short description, expected result, geometry and material properties. In the geometry diagram blue represents a volumetric source and green represents a volumetric detector.

A.1 Case A, Thick slab

The first test case is a uniform slab with a homogeneous source and detector. It is optically thick, which means that the neutrons have a small mean free path. The dimension of the slab is $1\ cm$. In Figure 32 a diagram of the geometry can be found and in Table 1 the material properties are listed. The boundary conditions of the slab are vacuum boundaries on both sides.

Since the boundary conditions are hard to satisfy properly it is expected that the mesh near the edges of the slab will be very fine.

Figure 32: *Homogeneous slab geometry.*

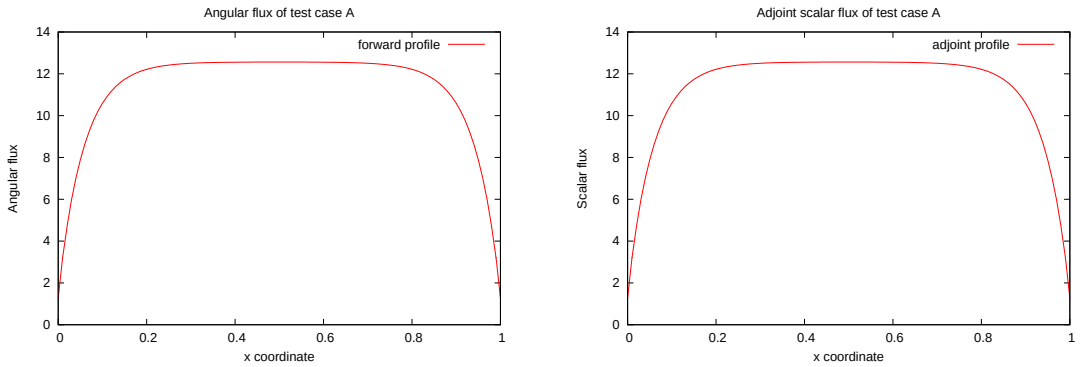
σ_t	$100\ cm^{-1}$
σ_s	$99\ cm^{-1}$
Source	$1\ cm^{-1}s^{-1}rad^{-1}$
Detector	$4\pi\ cm^{-1}$

Table 1: *Material properties for test problem A*

A.2 Case B, Thin slab

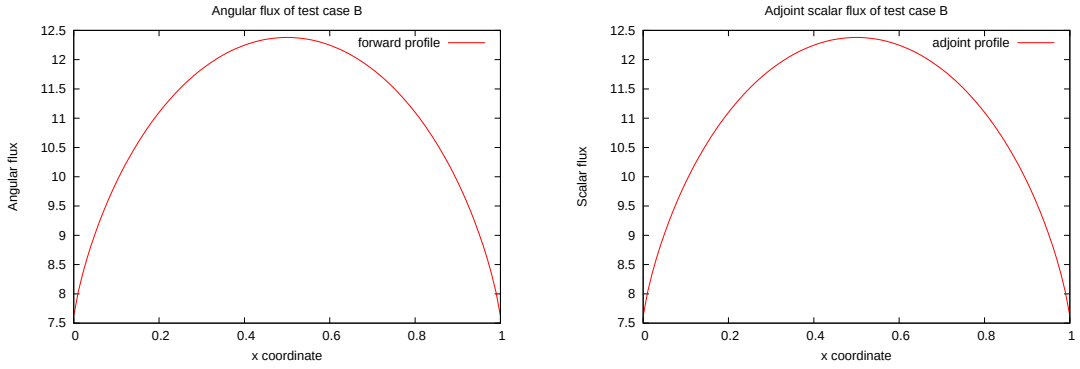
Test case B is again a homogeneous slab with the same dimensions as the test cases above. However, since the total cross section is much lower the mean free path of the neutrons is larger, which makes this an optically thin problem. Again the geometry can be found in Figure 32, while the material properties can be found in Table 2. This test case also has vacuum boundary conditions on both sides.

Refinement is expected to be similar to the refinement in test case A, although the effects of the edges will propagate much further into the domain, as this is an optically thin problem.



(a) *Solution of the forward scalar flux of test case A* (b) *Solution of the adjoint scalar flux of test case A*

Figure 33: *Forward and adjoint solution of test case A.*



(a) *Solution of the forward scalar flux of test case B* (b) *Solution of the adjoint scalar flux of test case B*

Figure 34: *Forward and adjoint solution of test case B.*

σ_t	$1\ cm^{-1}$
σ_s	$.5\ cm^{-1}$
Source	$1\ cm^{-1}s^{-1}rad^{-1}$
Detector	$4\pi\ cm^{-1}$

Table 2: *Material properties for test problem B*

A.3 Case C, Thick source detector

This test case has a separate source and detector region in a homogeneous material. The boundaries of this slab geometry are again vacuum boundaries. A diagram of the geometry can be found in Figure 35. The material properties are listed in Table 3.

In this test case it is important to have an accurate solution in the source and detector regions. Since the source is at the left hand side of the domain we also need an accurate solution of right going directions. Therefore we expect refinement in the detector source and regions, as well as refinement of right going directions.

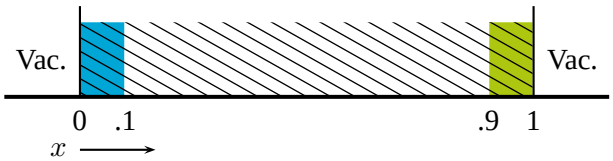


Figure 35: *Source detector slab geometry.*

σ_t	$100\ cm^{-1}$
σ_s	$99\ cm^{-1}$
Source	$1\ cm^{-1}s^{-1}rad^{-1}$
Detector	$4\pi\ cm^{-1}$

Table 3: *Material properties for test problem C*

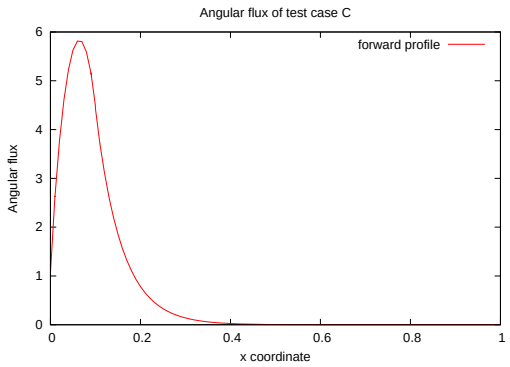
A.4 case D, Thin source detector

This test case is also a source detector problem, but now with optically thin material. An illustration of the geometry can be found in Figure 35. Table 4 lists the material properties.

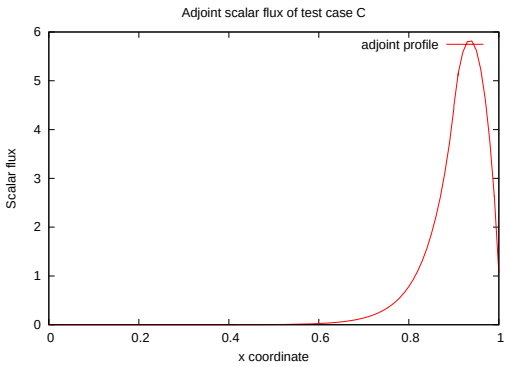
The expected behaviour is similar to that of test case C. However, since this is an optically thin problem the effects of the edges will propagate much further into the domain.

A.5 Case E, Highly absorbing source detector

This is also a source detector geometry as shown in Figure 35. However, the homogeneous material that is used in this problem is strongly absorbing. The material properties are listed in Table

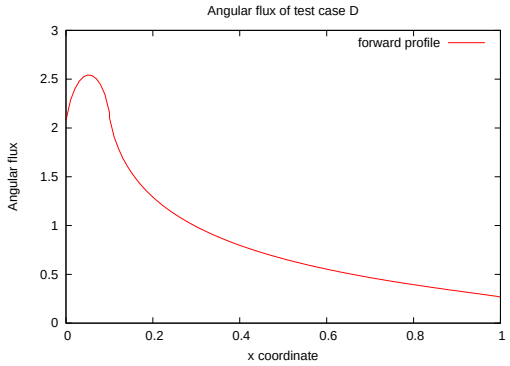


(a) *Solution of the forward scalar flux of test case C*

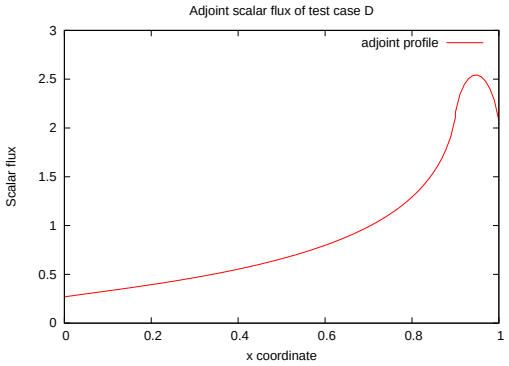


(b) *Solution of the adjoint scalar flux of test case C*

Figure 36: *Forward and adjoint solution of test case C.*



(a) *Solution of the forward scalar flux of test case D*

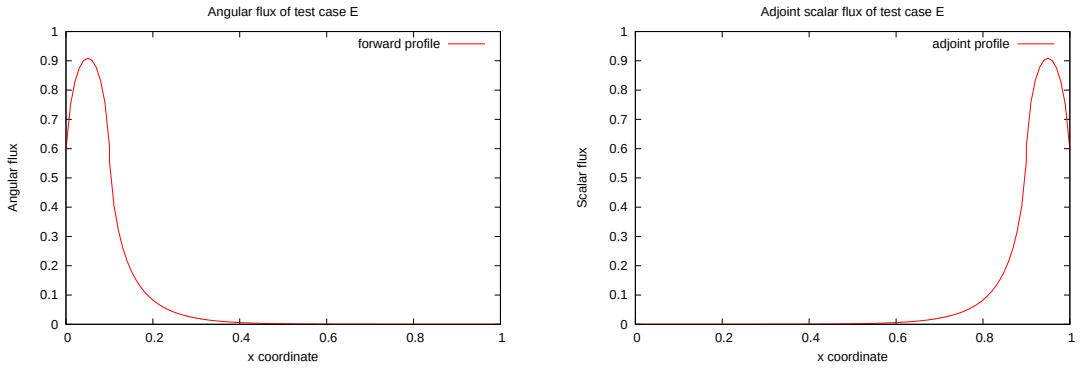


(b) *Solution of the adjoint scalar flux of test case D*

Figure 37: *Forward and adjoint solution of test case D.*

σ_t	$1\ cm^{-1}$
σ_s	$.5\ cm^{-1}$
thickness	$1\ cm$
Source	$1\ cm^{-1}s^{-1}rad^{-1}$
Detector	$4\pi\ cm^{-1}$

Table 4: *Material properties for test problem D*



(a) *Solution of the forward scalar flux of test case E* (b) *Solution of the adjoint scalar flux of test case E*

Figure 38: *Forward and adjoint solution of test case E.*

5.

In this test case we expect the same results as for test case C, as this is also an optically thick problem.

σ_t	$10\ cm^{-1}$
σ_s	$1\ cm^{-1}$
Source	$1\ cm^{-1}s^{-1}rad^{-1}$
Detector	$4\pi\ cm^{-1}$

Table 5: *Material properties for test problem E*

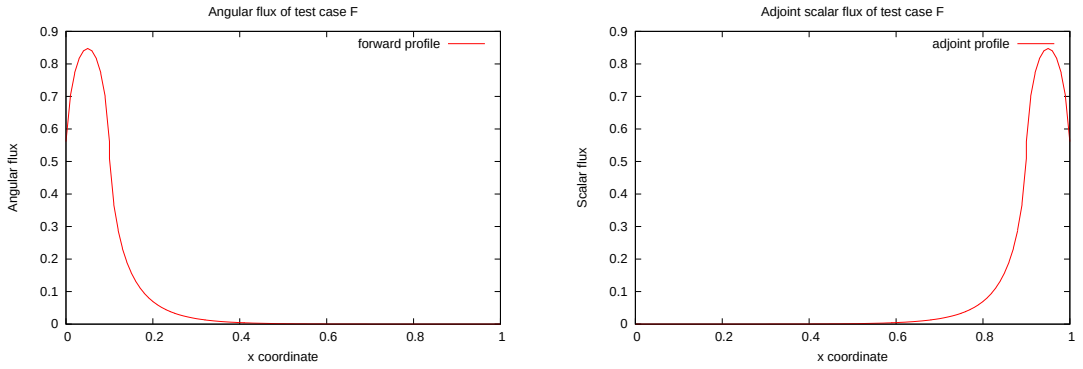
A.6 Case F, Purely absorbing source detector

Test case F is the last separate source detector geometry as shown in Figure 35. The material is now purely absorbing, which means there is no scatter source in the right hand side of the transport equation. The material properties of this test case are listed in Table 6.

In this case there is no coupling between directions through the scatter, which will yield results on the choice of directions of the discontinuous Galerkin method.

σ_t	$10\ cm^{-1}$
σ_s	$0\ cm^{-1}$
Source	$1\ cm^{-1}s^{-1}rad^{-1}$
Detector	$4\pi\ cm^{-1}$

Table 6: *Material properties for test problem F*



(a) *Solution of the forward scalar flux of test case F* (b) *Solution of the adjoint scalar flux of test case F*

Figure 39: *Forward and adjoint solution of test case F.*

A.7 Case G, Shielding

This test case is an extension of test case C. There is again a separate source and detector, however, the detector is behind a shield. A region of $0.05\ cm$ in the middle of the slab has a large total cross section, making it a neutron shield. The properties of the other regions remain the same. A diagram of the geometry is shown in Figure 40. The material properties are listed in Table 7, the source and detector are only present in the specified regions.

As this test case is similar to test case C we expect the refinement to be almost the same. Only the source region is not very important now, as only a few neutrons will traverse the shielding. Therefore it is more important to get an accurate flux in the shielding region. It is expected that refinement takes place in the shielding and detector region, as well as for right going directions.

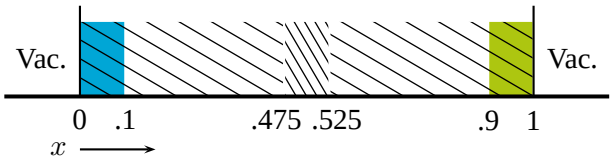
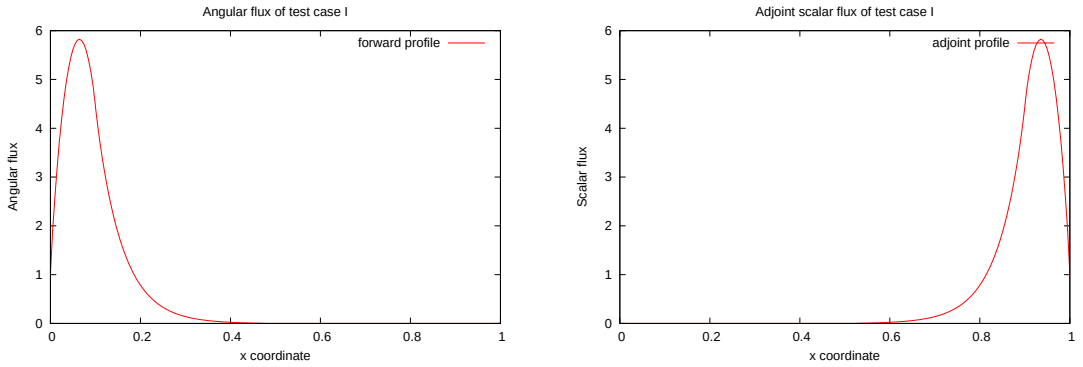


Figure 40: *Shielding slab geometry.*

Property	Material	Shielding
σ_t	$100\ cm^{-1}$	$50\ cm^{-1}$
σ_s	$99\ cm^{-1}$	$5\ cm^{-1}$
thickness	$2 \times .475\ cm$	$0.05\ cm$
Source	$1\ cm^{-1}s^{-1}rad^{-1}$	$0\ cm^{-1}s^{-1}rad^{-1}$
Detector	$4\pi\ cm^{-1}$	$0\ cm^{-1}$

Table 7: *Material properties for test problem G.*



(a) *Solution of the forward scalar flux of test case G*

(b) *Solution of the adjoint scalar flux of test case G*

Figure 41: *Forward and adjoint solution of test case G.*

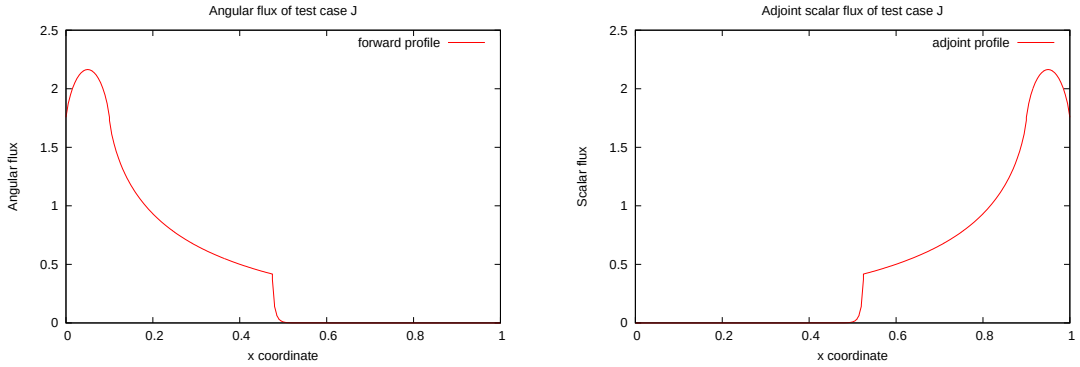
A.8 Case H, Purely absorbing shielding

Test case H has the same geometry as the previous test case, shown in Figure 40. Now both the medium and the neutron shield are purely absorbing. The material properties are listed in Table 8.

Expectations on refinement are the same as in the other shielding test case, case G.

Property	Material	Shielding
σ_t	$1\ cm^{-1}$	$100\ cm^{-1}$
σ_s	$0\ cm^{-1}$	$0\ cm^{-1}$
thickness	$2 \times .475\ cm$	$0.05\ cm$
Source	$1\ cm^{-1}s^{-1}rad^{-1}$	$0\ cm^{-1}s^{-1}rad^{-1}$
Detector	$4\pi\ cm^{-1}$	$0\ cm^{-1}$

Table 8: *Material properties for test problem H*



(a) *Solution of the forward scalar flux of test case H*

(b) *Solution of the adjoint scalar flux of test case H*

Figure 42: *Forward and adjoint solution of test case H.*