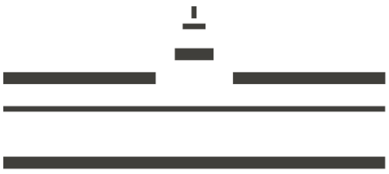

Westfälische
Wilhelms-Universität
Münster

Masterarbeit zum Thema

Discontinuous Galerkin Methods for the EEG Forward Problem

eingereicht von:
Jakob Ludewig

Gutachter:
Prof. Dr. Christian Engwer
Institut für Numerische und Angewandte Mathematik
Priv.-Doz. Dr. Carsten Wolters
Institut für Biomagnetismus und Biosignalanalyse

Münster, 10. Dezember 2013

Abstract

The topic of this thesis is the evaluation of Discontinuous Galerkin (DG) methods for the Electroencephalography (EEG) forward problem. We will start out by presenting the basic physiological, physical and mathematical framework necessary for modelling brain activity and show how the EEG forward problem can be derived from that. We will then briefly introduce the Subtraction Approach as a means of modelling the singularity occurring in the EEG forward problem when choosing a dipole source as the term. Afterwards we will introduce the basic concepts and notations needed to describe Discontinuous Galerkin methods. We will apply and extend this set of tools while deriving a weak DG formulation of the Subtraction Approach for the EEG forward problem. A short discussion about general properties of DG methods and how the Subtraction Approach may benefit from those is followed by a chapter describing the implementation of the discretized problem in the DUNE framework. The remainder of this thesis will be dedicated to the presentation of numerical results obtained by the implementation. In this discussion we will present multiple studies comparing DG with conforming Finite Element Methods (FEM) as a reference in both tetrahedral and hexahedral sphere models. In addition to the validation of the DG methods using sphere models we will present some visualizations of simulations performed on a realistic head model. Furthermore a study evaluating the computational cost of DG methods will be presented. We will conclude this work with a short outlook on possible future applications of Discontinuous Galerkin methods in the field of source analysis.

Acknowledgements

I would like to express my gratitude to everyone who made my studies and this thesis possible:

- Christian Engwer for introducing me to the subject of scientific computing and giving me the opportunity to write this thesis,
- Carsten Wolters for guiding my first steps in the fascinating field of source analysis,
- Johannes Vorwerk for his insights and advice in the early stages of this work and providing me with the models used in this thesis,
- Sven Wagner for helping me visualize some of the results and proof-reading the thesis,
- everyone else who proof-read this thesis and in particular Falk Meyer for their suggestions,
- all my friends in Münster and elsewhere,
- my family and especially my parents for their unconditional moral support and for granting me the privilege of higher education,
- Rebekka for her encouragement and for pulling me out of the rabbit hole now and then.

Contents

Introduction

In modern brain research as well as in everyday clinical practice *Electroencephalography(EEG)* has become an important and powerful tool. It enables scientists and practitioners alike to perform brain research and clinical diagnoses at a moderate cost. The first EEG was recorded in 1924 by neurologist Hans Berger and development of the methodology has continued ever since. The EEG registers potential changes on a subject's scalp which result from electromagnetic fields caused by neural activity in the brain. Visual inspection of the data recorded by the EEG electrodes can help to diagnose the origin of, for example, epileptic seizures or investigate the way in which the brain performs certain tasks. Although a great level of expertise has been achieved at this kind of source analysis, this manual method lacks a sound theoretical basis. It is relying heavily on the experience and knowledge of the analyst and it can not fully incorporate the physiological characteristics of the subject's brain structure, which in general is very complex with regard to electrical conductivity.
With the advancement of *Magnetic Resonance Imaging(MRI)* techniques it has become possible to obtain detailed information about the electrical properties of an individual's brain. This data can be incorporated into the *EEG forward problem* whose solution provides the potential recorded by an EEG given a certain neural activity within the brain. In source analysis the forward problem is a corner stone for solving the *inverse problem* which aims at recovering the source configuration corresponding to a given potential on the head surface. Many complex and highly advanced methods for the solution of the inverse problem have been proposed all of which depend on accurately solving a large number of forward problems.
The concurrent progress in numerical mathematics has produced a wide range of methods to deal with the forward problem in an adequate fashion. Among the most popular in the field of source analysis are the *Boundary Element Method(BEM)* and the conforming *Finite Element Method(FEM)*, each having their distinct advantages and disadvantages. Progress has been made concerning the accuracy as well as efficiency but still the case of sources very close to the brain surface and the incorporation of the brain's anisotropy remain challenging tasks with room for improvement. The *Discontinuous Galerkin Method(DG)* evaluated in this thesis offers reasons for hope that enhancements can be achieved in these scenarios.

1. Physiological and Mathematical Basis of the EEG Forward Problem

In this chapter we will give a short overview of the relevant neurophysiological background necessary to understand the subsequent physical and mathematical derivations that will result in the EEG Forward equation. Since an in-depth examination of this matter would go beyond the scope of this thesis, we will keep the following sections concise. The physiological principles governing the brain activity which is measurable by the EEG are well researched and details can be found in [1] or [2].

1.1. The Physiology of Brain Activity

The electric potentials on a subject's head surface which are registered by the EEG are caused by electromagnetic fields resulting from ion movement inside the brain. We will now take a closer look at the *neuron* as the basic physiological structure in the process of brain activity. The neuron is an electrically excitable cell that is responsible for transmitting and processing information inside the whole human body in general and the brain in particular. The human brain contains approximately up to 10^{12} neurons which form a complex and interconnected network.
On a larger scale a neuron can be divided into two parts: the cell body(*soma*), containing the *nucleus*, and one *axon*, the latter being an elongate projection of the soma (see Figure 1.1). *Dendrites* form extensions to the cell body, acting as receivers of information in the form of electrochemical stimuli, transmitted from other neural cells of the whole body.

In general, the axon acts as a transmitter sending electrical impulses away from the cell body. It has a special structure that allows for the transmission of electrical stimuli in spite of the relatively high resistance of the nerve fibres. The axon is connected to the cell body at a position called the *axon hillock* and at the other end it branches out, making up the so-called *axon terminals*. These extensions are located close to the dendrites of another neuron, forming a conjunction called a *synapse*. Synapses form connections between the axon of one neuron and the dendrites of another neuron (see Figure 1.2). This is the location where the actual information transmission occurs in a fashion we will now describe.

The basis for information transport and processing on a neural level is the change in ion concentrations inside and around the neuron. In an inactive neuron the difference in the extra- and intra-cellular potential ranges from 60 mV up to approximately 100 mV with the

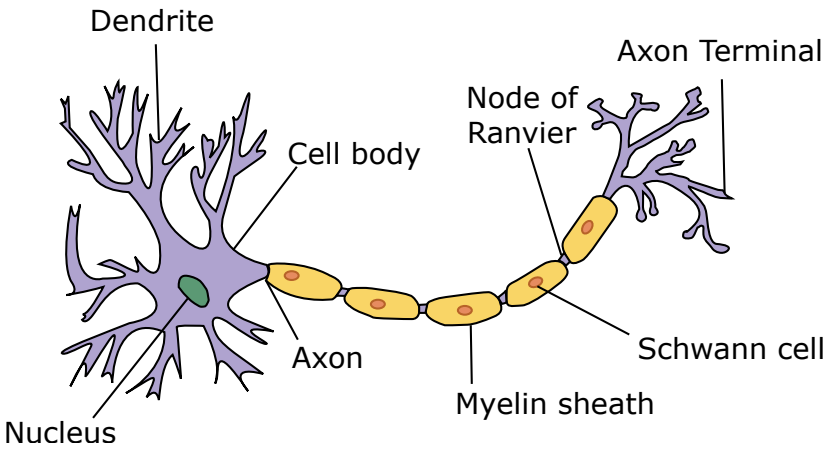

Figure 1.1.: Basic Structure of a Neuron. *Source: Wikimedia Commons*

inside of the cell being negative. This potential equilibrium is maintained by *ion pumps* which transport potassium(K^+), sodium(Na^+) and chlorine(Cl^-) ions (among others) through the membrane of the neuron.
Once a neuron is sufficiently excited it will release *neurotransmitters* at its axon terminals. These neurotransmitters lead to a change in the ion permeability of the dendrite of the neuron to which it is connected through the associated synapse and ultimately to a change in the potential difference between the inside and the outside of this neuron which is called the *postsynaptic potential*. If this potential reaches a certain threshold at the axon hillock, the neuron will *fire*, i. e. it will transmit the so-called *action potential* along its axon and thus depolarizing the neuron. This, again, will result in the release of neurotransmitters at the axon terminals.
Since the potential threshold at the axon hillock is not necessarily reached this form of information transmission can be considered an *all-or-nothing* type of communication. Also it should be noted that the change in ion permeability does not have to result in the excitation of the neuron but also inhibitory changes may occur.

The action potential and the postsynaptic potential differ with respect to their temporal extent as well as the magnitude of the potential. The action potential exhibits temporal scales of about 1 ms and potential magnitudes of about 100 mV. Also the occurrence of multiple action potentials is not very well synchronized, making them unfit for EEG measurement.
The postsynaptic potential on the other hand features time frames in the order of 10 ms and magnitudes of about 10 mV. Furthermore, this kind of activity is rather well synchronized between a multitude of neurons. This behaviour makes the postsynaptic potential suitable for physical modelling as presented in the next section.

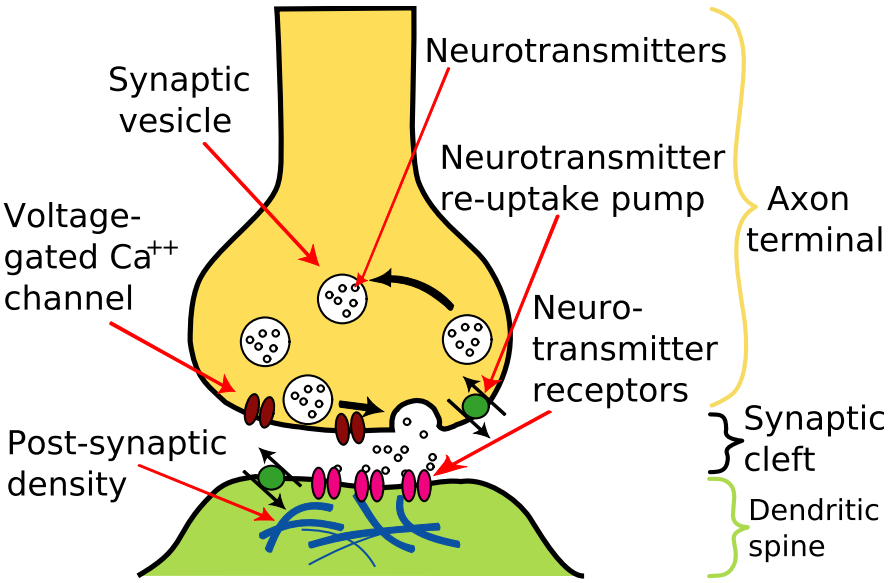

Figure 1.2.: Schematic Illustration of a Synapse. *Source: Wikimedia Commons*

1.2. Physical Modelling of Neural Activity

As described in the previous section, the EEG registers the potentials caused by electromagnetic fields resulting from the post-synaptic potentials. We will now derive a physical model to describe this phenomenon. For a more detailed derivation we refer to [2], [16] or [5].
In order to describe the electromagnetic field we will use *Maxwell's Equations* in the following form

Definition 1.1 (Maxwell's Equations)**.** With $\mathbf{D}$ being the *electric displacement field*, ρ the *charge density*, $\mathbf{E}$ the *electric field*, $\mathbf{B}$ the *magnetic field*, $\mathbf{J}$ the *the current density* and $\mathbf{H}$ the *magnetizing field* the following holds:

$$\nabla \cdot \mathbf{D} = \rho \tag{1.1}$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \tag{1.2}$$

$$\nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial \mathbf{t}} \tag{1.3}$$

$$\nabla \cdot \mathbf{B} = 0. \tag{1.4}$$

Since biological tissue can be considered to be an electrolyte, we can assume the following material equations to be true:

$$\mathbf{D} = \varepsilon\varepsilon_0\mathbf{E} \tag{1.5}$$

$$\mathbf{B} = \mu\mu_0\mathbf{H} \tag{1.6}$$

In the above ε and ε_0 denote the relative permittivity and the permittivity of free space, respectively. The same goes for μ and μ_0 for the case of permeability. We can neglect the magnetic properties of biological tissue, i.e. we can set $\mu = 1$ which leaves us with the permeability of free space μ_0 in (1.6). Using the above equations to transform (1.3) we get

$$\nabla \times \mathbf{B} = \mu_0(\mathbf{J} + \varepsilon\varepsilon_0\frac{\partial \mathbf{E}}{\partial \mathbf{t}}). \tag{1.7}$$

We utilize Ohm's Law $\mathbf{J} = \sigma\mathbf{E}$, with σ being the conductivity of the material. Furthermore we apply a Fourier transformation to $\mathbf{E}$ with the angular frequency ω:

$$\mathbf{E} = E_0 \cdot e^{-i\omega t} \tag{1.8}$$

Using the above term for the temporal derivative in (1.7), we get

$$\nabla \times \mathbf{B} = \mu_0(\sigma\mathbf{E} - i\varepsilon\varepsilon_0\mathbf{E}). \tag{1.9}$$

In source analysis the following assumptions can be made for the human head(see [2]):

$$\sigma \approx 0.3\, S/m$$
$$\varepsilon_r \approx 10^5$$
$$f = \omega/2\pi \approx 100\, Hz$$

Using these premises and the text book value for ε_0 we get $|\varepsilon\varepsilon_0\omega| \ll |\sigma|$ and accordingly the temporal derivative can be neglected.
Furthermore we show that the temporal derivative of the magnetic field is negligible. For that we apply the rotation to (1.2):

$$\nabla \times \nabla \times \mathbf{E} = -\frac{\partial(\nabla \times \mathbf{B})}{\partial \mathbf{t}} \tag{1.10}$$

Using (1.3) and transforming it in the same fashion as before we get

$$\nabla \times \nabla \times \mathbf{E} = -\mu_0\frac{\partial}{\partial \mathbf{t}}(\sigma\mathbf{E} + \varepsilon\varepsilon_0\frac{\partial \mathbf{E}}{\partial \mathbf{t}}) = \mu_0(i\sigma\omega - \varepsilon\varepsilon_0\omega^2)\mathbf{E}. \tag{1.11}$$

The characteristic wavelengths of this equation have a magnitude of:

$$\lambda_c = |\mu_0\sigma\omega(1 - \varepsilon\varepsilon_0\omega/\sigma)|^{-1/2} \approx 65\, m.$$

This is much larger than the diameter of an average human head which means that the contribution of $\mathbf{E}$ can be neglected in this equation.
All in all we arrive at the following quasi-static approximation of our problem:

Definition 1.2 (Quasi-static Maxwell's Equations).

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon \varepsilon_0} \tag{1.12}$$
$$\nabla \times \mathbf{E} = 0 \tag{1.13}$$
$$\nabla \times \mathbf{B} = \mu_0 \mathbf{J} \tag{1.14}$$
$$\nabla \cdot \mathbf{B} = 0 \tag{1.15}$$

1.3. Derivation of the EEG Forward Equation

In (1.14) we can split up the current density $\mathbf{J}$ into two parts

$$\mathbf{J} = \mathbf{J^p} + \sigma \mathbf{E} \tag{1.16}$$

where $\mathbf{J^p}$ is the *primary current* which is a direct result of the neural activity. The *return current* $\sigma \mathbf{E}$ results from the macroscopic electric field on charge carriers in the conducting volume, with σ being the *conductivity tensor*. In contrast to the primary current it is present in the whole medium.
Since the curl of $\mathbf{E}$ is zero (see (1.13)) we can assume $\mathbf{E}$ it to be the gradient field of a scalar potential $\mathbf{V}$:

$$\mathbf{E} = -\nabla \mathbf{V}. \tag{1.17}$$

Inserting the above into (1.16), we get

$$\mathbf{J} = \mathbf{J^p} - \sigma \nabla \mathbf{V} \tag{1.18}$$

which can be inserted in the divergence of (1.14) that turns out to be zero:

$$\begin{aligned} 0 &= \nabla \cdot \mathbf{J^p} - \nabla \cdot (\sigma \nabla \mathbf{V}) \\ \text{or equivalently} \quad \nabla \cdot (\sigma \nabla \mathbf{V}) &= \nabla \cdot \mathbf{J^p} \end{aligned} \tag{1.19}$$

From now on we assume that our volume model can be represented as an open and connected domain Ω with sufficiently regular boundary $\partial\Omega$. Furthermore we make the sound physical assumption that the current density is continuous across the boundary. Also we take into account that the conductivity of the medium surrounding the volume, i.e. air, is zero. Thus we get *homogeneous Neumann boundary conditions* on the surface,

$$\sigma \nabla \mathbf{V} \cdot \vec{n} = 0, \tag{1.20}$$

with $\vec{n}$ being the outward normal unit vector on $\partial\Omega$ at the corresponding position.
Overall we get the following Poisson-type equation with homogeneous Neumann boundary conditions:

$$\begin{aligned} \nabla \cdot \sigma(x)\nabla \mathbf{V}(x) &= f \quad (= \nabla \cdot J^p) && \text{in } \Omega && (1.21)\\ \sigma(x)\nabla \mathbf{V}(x) \cdot \vec{n}(x) &= 0 && \text{on } \partial\Omega && (1.22) \end{aligned}$$

In the above we substituted $\nabla \cdot \mathbf{J^p}$ by the abstract source term f to emphasize the necessity for the choice of an appropriate method to model the source. We will discuss this in the following section.
Before we do so, we need to ensure the uniqueness of the solution to the above equations. Since we are dealing with a pure Neumann problem the solution of the problem is only unique up to a scalar constant and we need to supplement the above equations by a third condition which makes the solution to it unique. For this purpose several different approaches are viable. For now we will use the following condition to ensure uniqueness of the solution:

$$\int_\Omega \mathbf{V}(x)\,\mathrm{d}x = 0. \tag{1.23}$$

Finally, we arrive at the following problem formulation:

Problem 1.1 (EEG Forward Equation)**.**

$$\begin{aligned} \nabla \cdot \sigma(x)\nabla \mathbf{V}(x) &= f \quad (= \nabla \cdot J^p) && \text{in } \Omega && (1.24)\\ \sigma(s)\nabla \mathbf{V}(s) \cdot \vec{n}(s) &= 0 && \text{on } \partial\Omega && (1.25)\\ \int_\Omega \mathbf{V}(x)\,\mathrm{d}x &= 0 && && (1.26) \end{aligned}$$

In the next section we will discuss how the source term f can be chosen and what has to be taken into consideration in order to make the EEG forward equation suitable for discretization methods like the conforming *Finite Element Method(FEM)* and the *Discontinuous Galerkin(DG)* method.

1.4. The Dipole Model for the Source Term

For the forward equation which was derived in the previous section we now need to model the source term f. In order to do so, we introduce the notion of a *mathematical dipole*. With y being the source position, the dipole source is defined as

$$J^p(x) = M \cdot \delta_y(x). \tag{1.27}$$

The vector M denotes the moment of the dipole and δ_y denotes the Dirac delta distribution at point y. The dipole describes the unidirectional primary current that may extend over several square centimetres of the cortex, thus it represents the current resulting from the simultaneous activity of a large group of neurons.
In source analysis it is common practice to use the dipole model for the source term and it has been shown that for distances considered in EEG measurements the assumption of a dipolar source is realistic (see [2] and [20]).

Remark 1.1. *For the following derivations it should be noted that the EEG forward equation as formulated in Problem 1.1 does not have a solution in the strong sense when we assume the source term J^p to be a dipole. A classical solution would have to satisfy $\mathbf{V} \in C^2(\Omega) \cap C^0(\bar{\Omega})$ which is not possible for such a singular source term. So whenever we use a strong formulation as seen above, we will have to assume a stronger regularity to the source term J^p or restrict ourselves to regarding the equations in a purely symbolical sense. That is why we will switch to a weak formulation of the problem once we actually use the dipole as a source model in the following derivations.*

Besides those rather formal problems, the introduction of a dipole source term will pose problems regarding the discretization in a finite element context. The dipole as a source term constitutes a singularity which is per se not suitable for a finite element approximation of the problem and results in further mathematical effort to cope with. Multiple approaches have been proposed to arrive at a problem formulation which is suitable for discretization methods (see for example [3] or [5]).
Since it has the most valid mathematical foundation, we will concentrate on the *Subtraction Approach* in this thesis.

1.5. The Subtraction Approach

Following the concepts presented in [15] we will now describe the basic ideas of the *Full Subtraction Approach* for the EEG forward problem. Note that in this context we will use continuous test functions v. Thus the following derivations are not valid for Discontinuous Galerkin methods but will only act as a prototype for later reference.

We will consider Problem 1.1 derived in the previous section for a dipole source. To conform with the literature on the Subtraction Approach and the common notations in a Finite Element context we will replace $\mathbf{V}$ by u.
In the case of *anisotropic conductivities* $\sigma : \mathbb{R}^3 \to \mathbb{R}^3 \times \mathbb{R}^3$ maps a point in Ω to a symmetric positive matrix. For *isotropic conductivities* this becomes a diagonal matrix and thus can be regarded as the multiplication with a scalar value. The following derivation of the Subtraction Approach will be valid for both cases.

In the setting of an unbounded volume conductor $\Omega = \mathbb{R}^3$ with constant conductivity $\sigma(x) \equiv$

$\sigma^{\infty,y}$ for all $x \in \Omega$ the analytical solution to Problem 1.1 is known and it takes the form of:

$$u^{\infty,y}(x) := \frac{1}{4\pi\sqrt{\det\sigma(y)}} \frac{M(y)\cdot\sigma(y)^{-1}(x-y)}{(\sigma(y)^{-1}(x-y)\cdot(x-y))^{3/2}} \tag{1.28}$$

In the following $u^{\infty,y}$ will be called the *singularity potential*. We will also need its gradient:

$$\nabla u^{\infty,y} := \frac{1}{4\pi\sqrt{\det\sigma(y)}} \cdot \frac{\sigma(y)^{-1}M(y)}{(\sigma(y)^{-1}(x-y)\cdot(x-y))^{3/2}} \tag{1.29}$$

$$- \frac{1}{4\pi\sqrt{\det\sigma(y)}} \cdot \frac{(3\cdot M(y)\cdot\sigma(y)^{-1}(x-y))(\sigma(y)^{-1}(x-y))}{(\sigma(y)^{-1}(x-y)\cdot(x-y))^{5/2}} \tag{1.30}$$

The fundamental idea of the Subtraction Approach is to utilize the singularity potential to split up the potential u into two parts, the singularity potential itself and the *correction potential* $u^{corr,y}$:

$$u = u^{\infty,y} + u^{corr,y}$$

As motivated in Remark 1.1 we will do this using a weak formulation of Problem 1.1. Thus we will multiply equations (1.24) and (1.25) with a test function $v \in H^1(\Omega)$ and integrate them over the domain and the boundary of the domain respectively. Inserting the splitting for u we arrive at the following system of equations:

$$\int_\Omega \nabla\cdot\sigma(x)\nabla(u^{corr,y}(x)+u^{\infty,y}(x))v(x)\,\mathrm{d}x = \int_\Omega f(x)v(x)\,\mathrm{d}x \tag{1.31}$$

$$\int_{\partial\Omega} \sigma(s)\nabla(u^{corr,y}(s)+u^{\infty,y}(s))\cdot\vec{n}(s)v(s)\,\mathrm{d}s = 0 \tag{1.32}$$

$$\int_\Omega (u^{corr,y}(x)+u^{\infty,y}(x))\,\mathrm{d}x = 0 \tag{1.33}$$

Since the singularity potential is known analytically we can move all the terms involving $u^{\infty,y}$ to the right hand side and solve for $u^{corr,y}$ instead of u:

$$\int_\Omega \nabla\cdot\sigma(x)\nabla u^{corr,y}(x)v(x)\,\mathrm{d}x = \int_\Omega (f(x)-\nabla\cdot\sigma(x)\nabla u^{\infty,y}(x))v(x)\,\mathrm{d}x \tag{1.34}$$

$$\int_{\partial\Omega} \sigma(s)\nabla u^{corr,y}(s)\cdot\vec{n}(s)v(s)\,\mathrm{d}s = -\int_{\partial\Omega} \sigma(s)\nabla u^{\infty,y}(s)\cdot\vec{n}(s)v(s)\,\mathrm{d}s \tag{1.35}$$

$$\int_\Omega u^{corr,y}(x)\,\mathrm{d}x = -\int_\Omega u^{\infty,y}(x)\,\mathrm{d}x \tag{1.36}$$

Now we split up σ into $\sigma(x) = \sigma^{corr,y}(x) + \sigma^{\infty,y}$. Knowing that $u^{\infty,y}$ as introduced in (1.28) solves the forward equation in an unbounded conductor with constant conductivity $\sigma^{\infty,y}$ we see that

$$\int_\Omega (f(x)-\nabla\cdot\sigma^{\infty,y}\nabla u^{\infty,y}(x))v(x)\,\mathrm{d}x = 0$$

and we are left with

$$-\int_\Omega \nabla \cdot \sigma^{corr,y}(x)\nabla u^{\infty,y}(x)v(x)\,\mathrm{d}x$$

on the right hand side of (1.34).
Applying partial integration to both sides of (1.34), substituting the resulting boundary term on the left hand side with (1.35) and moving it to the right hand side we arrive at:

Definition 1.3 (Subtraction approach conforming weak formulation)**.**

$$\begin{aligned} -\int_\Omega \sigma(x)\nabla u^{corr,y}(x)\nabla v(x)\,\mathrm{d}x &= \int_\Omega \sigma^{corr,y}(x)\nabla u^{\infty,y}(x)\nabla v(x)\,\mathrm{d}x \qquad (1.37a)\\ &\quad + \int_{\partial\Omega} \sigma^{\infty,y}\nabla u^{\infty,y}\vec{n}(s)v(s)\,\mathrm{d}s \\ \int_\Omega u^{corr,y}(x)\,\mathrm{d}x &= -\int_\Omega u^{\infty,y}(x)\,\mathrm{d}x \qquad (1.37b) \end{aligned}$$

After solving for $u^{corr,y}$ the potential u is recovered by recombining $u^{corr,y}$ with $u^{\infty,y}$.

In order for the Subtraction Approach to be valid the following assumption has to be made:

Assumption 1.1 (Homogeneity Condition)**.** Given a source space $Y \subset \Omega$ there exists $\varepsilon > 0$ for every $y \in Y$ such that the conductivity tensor $\sigma(x)$ is constant in a ball

$$\Omega_\varepsilon^y := \{x \in \Omega \mid \|x-y\|_2 < \varepsilon\}$$

Assumption 1.1 is reasonable for multisphere studies as presented in this thesis but also for the case of realistic head models. In the former conductivities are assigned to each element simulating the conduction properties within different compartments of the brain. In the case of realistic head models with anisotropy *Diffusion Tensor Imaging(DTI)* data is obtained from MRI measurements and each voxel is assigned a specific conductivity value again leading to element wise constant conductivities. For realistic head models with isotropic conductivities each voxel is assigned to a certain compartment of the brain and given the corresponding conductivity. In each of the previous cases the source space Y has to be chosen in such a manner that it excludes source positions in which no definite statement can be made about which element the source belongs to (for example source positions on nodes or element faces). With the above it is obvious that the homogeneity condition holds.
Using the homogeneity condition it can be shown that the right hand side of (1.37b) is in $L^2(\Omega)$ and standard methodology for proving uniqueness and existence of a solution $u^{corr,y} \in H^1(\Omega)$ can be applied. What is more, the problem is now suitable for a finite element discretization. For more information on that we refer once more to [15].

2. The Discontinuous Galerkin Method

In this chapter we will introduce the *Discontinuous Galerkin(DG) Finite Element Method* and derive a Subtraction DG formulation analogous to the conforming formulation which was introduced in the previous chapter. We will start out by giving a short historical overview of the development of DG methods and the origin of its fundamental idea. We will then introduce some basic notations and definitions which will be used to derive the weak formulation of the EEG forward problem as modelled by the Subtraction Approach.

2.1. A Short Historical Overview

The fundamental idea for Discontinuous Galerkin methods can be traced back to 1971 when Nitsche [6] proposed a method for solving problems with Dirichlet boundary conditions by the use of a penalty term rather than a restriction of the solution space. Consider the elliptic model problem

$$-\Delta u = f \qquad \text{in } \Omega \tag{2.1a}$$

$$u = 0 \qquad \text{on } \partial\Omega \tag{2.1b}$$

with $u \in C^2(\Omega) \cap C^0(\bar{\Omega})$ on a domain $\Omega \subseteq \mathbb{R}^d$. Following the idea of Babuška [7] one can replace (2.1b) by an approximation of the boundary condition:

$$u + \frac{1}{\mu}\nabla u \cdot \vec{n} = 0 \qquad \text{with} \quad \mu > 0. \tag{2.2}$$

However, the resulting weak formulation does not solve the strong formulation of the model problem. In order to fix this lack of consistency Nitsche introduced the following weak formulation:

$$\int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}x - \int_{\partial\Omega} \nabla u \cdot \vec{n} v \,\mathrm{d}s - \int_{\partial\Omega} \nabla v \cdot \vec{n} u \,\mathrm{d}s + \mu \int_{\partial\Omega} uv \,\mathrm{d}s \quad = \quad \int_\Omega fv \,\mathrm{d}x. \tag{2.3}$$

With $u, v \in V \subset H^1(\Omega)$. This formulation arises quite naturally if equation (2.1a) is tested with v and (2.1b) is tested with $v - \frac{1}{\mu}\nabla v \cdot \vec{n}$. Adding up the resulting equations, multiplying the second with μ and applying partial integration on the left hand side of the volume term, one arrives at (2.3).
The additional terms

$$-\int_{\partial\Omega} \nabla v \cdot \vec{n} u \,\mathrm{d}s \tag{2.4}$$

and

$$\mu \int_{\partial\Omega} uv \, \mathrm{d}s \tag{2.5}$$

ensure symmetry and coercivity, respectively.
The development of Discontinuous Galerkin Methods began when Babuška and Zlámal [8] used the above concept to weakly enforce continuity over inter-element boundaries of the solution for a fourth-order problem in 1973. In the following more methods were developed for elliptic, parabolic and hyperbolic equations. Suffering from the lack of efficient solvers and good estimates for the choice of necessary penalty parameters, development of DG methods slowed down at the beginning of the 1980s. However, in the recent past this changed due to new applications in which DG methods proved advantageous, enhanced estimates for the penalty terms arising in the formulation and the publication of a framework that allows for a unified analysis of all DG methods by Arnold [9].

2.2. Basic Framework

We will now introduce some basic geometric notations and definitions which are essential to the following discussion of Discontinuous Galerkin methods.

We begin by defining the *triangulation* $\mathscr{T}$ of a domain $\Omega \subseteq \mathbb{R}^d$:

Definition 2.1 (Triangulation $\mathscr{T}$).

$$\mathscr{T} = \{E_0, \ldots, E_{M-1}\} \quad \text{with} \tag{2.6a}$$
$$E_n \subseteq \Omega \qquad \forall 0 \leq n < M, \quad E_n \text{ open}, \tag{2.6b}$$
$$E_n \cap E_m = \emptyset \qquad \forall 0 \leq n < m < M, \tag{2.6c}$$
$$\overline{\Omega} = \bigcup_{n=0}^{M-1} \overline{E}_n. \tag{2.6d}$$

We will also use the notation $\mathscr{T}_h$ where h is defined as

$$h := \max\{diam(E) \mid E \in \mathscr{T}_h\}.$$

In addition to that we will need the *internal and external skeleton* of the triangulation $\mathscr{T}_h$:

Definition 2.2 (Internal Skeleton Γ_{int}).

$$\Gamma_{int} = \left\{ \gamma_{e,f} = \partial E_e \cap \partial E_f \mid E_e, E_f \in \mathscr{T}_h, E_e \neq E_f, \left|\gamma_{e,f}\right| > 0 \right\} \tag{2.7}$$

Definition 2.3 (External Skeleton Γ_{ext}).

$$\Gamma_{ext} = \{ \gamma_e = \partial E_e \cap \partial \Omega \mid E_e \in \mathscr{T}_h, |\gamma_e| > 0 \} \tag{2.8}$$

Here $|\cdot|$ denotes the codimension one volume. In the case of a three-dimensional triangulation $\mathscr{T}_h$ this means that nodes or edges are not contained in the internal or the external skeleton.
Furthermore we define the *skeleton* of $\mathscr{T}_h$ as the union of the internal and the external skeleton:

Definition 2.4 (Skeleton Γ).

$$\Gamma = \Gamma_{int} \cup \Gamma_{ext} \tag{2.9}$$

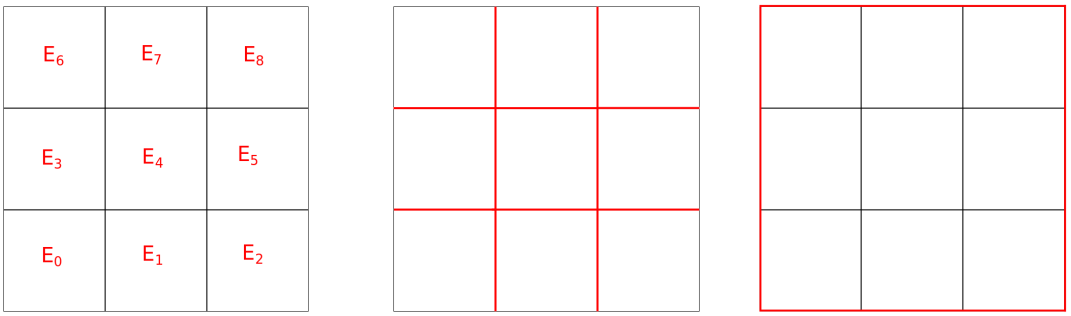

Figure 2.1.: Visualization of the triangulation $\mathscr{T}_h$ (left), the internal skeleton Γ_{int} (middle) and the external skeleton Γ_{ext} (right) for a 2D-example.

Figures 2.1 and 2.2 exemplify the definitions introduced above.

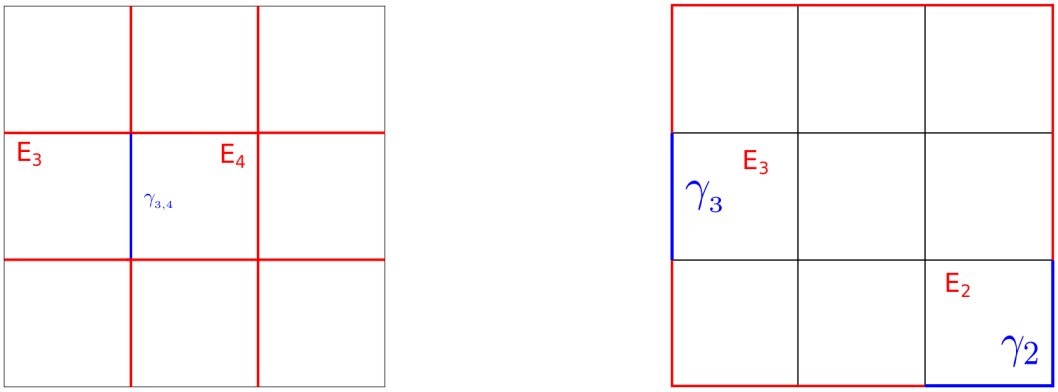

Figure 2.2.: Visualization of the elements $\gamma_{3,4}$ in Γ_{int} (left) and the elements γ_3 and γ_2 in Γ_{ext} (right).

Using this geometric framework we can now derive the weak DG formulation for the EEG forward problem. Further definitions and notations will be added when they are needed in the process.

2.3. The DG Formulation of the Subtraction Approach for the EEG Forward Problem

2.3.1. Derivation of a Consistent Weak Formulation

We will start out by restating the original formulation for the EEG forward problem as introduced in Section 1.3:

Problem 2.1 (Original Problem Formulation of the EEG Forward Problem).

$$\nabla \cdot \sigma(x)\nabla u(x) = f \quad \left(= \nabla \cdot \vec{j}\right) \qquad \text{in } \Omega \tag{2.10a}$$

$$\sigma(x)\nabla u(x) \cdot \vec{n}(x) = 0 \qquad \text{on } \partial\Omega \tag{2.10b}$$

$$\int_{\Omega} u(x)dx = 0 \tag{2.10c}$$

We assume the existence of a triangulation $\mathscr{T}_h$ of Ω as introduced in the previous section.

Test functions will be taken from *broken polynomial spaces*, i. e. either from

$$V_h := \mathbb{P}_d^k(\mathscr{T}_h) \tag{2.11}$$

or

$$V_h := \mathbb{Q}_d^k(\mathscr{T}_h). \tag{2.12}$$

The following derivations will be valid for both cases. For more details on the above function spaces see Definitions A.2 and A.3 in the Appendix and also the discussion in Section 2.5. For now it suffices to say that V_h consists of functions that behave like polynomials on each element of the triangulation but may exhibit *discontinuities* across inter-element boundaries.

Note that in the case of Dirichlet boundary conditions the function space V_h would not be modified but the boundary conditions would rather be enforced weakly through a penalty term, just like the continuity across inter-element boundaries.

Taking the left hand side of equation (2.10a), testing it with $v \in V_h$ and integrating over Ω we get

$$\int_\Omega \nabla \cdot \sigma(x) \nabla u(x) v(x) \, \mathrm{d}x. \tag{2.13}$$

Splitting up the domain Ω into the elements Ω_e of the triangulation $\mathscr{T}_h$ the above becomes:

$$\int_\Omega \nabla \cdot \sigma(x) \nabla u(x) v(x) \, \mathrm{d}x = \sum_{\Omega_e \in \mathscr{T}_h} \int_{\Omega_e} \nabla \cdot \sigma(x) \nabla u(x) v(x) \, \mathrm{d}x. \tag{2.14}$$

Now, in order to account for inter-element discontinuities, element wise partial integration is applied to each of the integral terms and (2.14) becomes:

$$-\sum_{\Omega_e \in \mathscr{T}_h} \int_{\Omega_e} \sigma(x) \nabla u(x) \cdot \nabla v(x) \, \mathrm{d}x + \sum_{\Omega_e \in \mathscr{T}_h} \int_{\partial \Omega_e} \sigma(s) \nabla u(s) \cdot \vec{n} v(s) \, \mathrm{d}s \tag{2.15}$$

Note that functions in V_h are not defined on elements in the skeleton Γ of the triangulation. Thus appropriate trace operators have to be utilised in order to carry out calculations on boundary and intersection elements. Here we assumed the existence of a trace operator

$$\hat{\cdot} : \Omega_e \to \partial \Omega_e$$

for each Ω_e in $\mathscr{T}_h$ that extends u, v and σ to the boundary of Ω_e. For convenience we omit writing $\hat{u}$,$\hat{v}$ or $\hat{\sigma}$ and instead use u, v and σ.
Also note that in general, the outward normal unit vector $\vec{n}$ may be depending on the integration point s on the boundary $\partial \Omega_e$, e.g. in the case of isoparametric elements, but for now

we assume it to be independent.

The first term of (2.15) sums up to be

$$-\sum_{\Omega_e \in \mathscr{T}_h} \int_{\Omega_e} \sigma(x) \nabla u(x) \cdot \nabla v(x) \, \mathrm{d}x = -\int_{\Omega} \sigma(x) \nabla u(x) \cdot \nabla v(x) \, \mathrm{d}x \tag{2.16}$$

whereas for the second sum two cases have to be distinguished:
On the one hand we sum over all the segments that each Ω_e might share with the boundary of the domain Ω, thus over the external skeleton Γ_{ext} which adds up to the boundary $\partial\Omega$. However, this term vanishes due to the homogeneous Neumann boundary conditions (see (2.10b)).
On the other hand the sum includes *two* integrals over each $\gamma_{e,f} \in \Gamma_{int}$, one as a boundary segment of Ω_e and the other as a boundary segment of Ω_f. These segments will also be called *interfaces* in the following. For each of the two integrals the respective local trace operators have to be used, thus the integrands under the two integrals over $\gamma_{e,f}$ are not necessarily equal.
Keeping this in mind we can now rewrite the second sum of (2.15) to be:

$$\begin{aligned}
&\sum_{\Omega_e \in \mathscr{T}_h} \int_{\partial\Omega_e} \sigma(s) \nabla u(s) \cdot \vec{n} v(s) \, \mathrm{d}s \\
&= \underbrace{\int_{\partial\Omega} \sigma(s) \nabla u(s) \cdot \vec{n} v(s) \, \mathrm{d}s}_{=0, \text{ see (2.10b)}} \\
&+ \sum_{\gamma_{e,f} \in \Gamma_{int}} \left(\int_{\gamma_{e,f} \cap \partial\Omega_e} \sigma(s) \nabla u(s) \cdot \vec{n}_{e,f} v(s) \, \mathrm{d}s + \int_{\gamma_{e,f} \cap \partial\Omega_f} \sigma(s) \nabla u(s) \cdot \vec{n}_{f,e} v(s) \, \mathrm{d}s \right) \\
&= \sum_{\gamma_{e,f} \in \Gamma_{int}} \left(\int_{\gamma_{e,f} \cap \partial\Omega_e} \sigma(s) \nabla u(s) \cdot \vec{n}_{e,f} v(s) \, \mathrm{d}s + \int_{\gamma_{e,f} \cap \partial\Omega_f} \sigma(s) \nabla u(s) \cdot \vec{n}_{f,e} v(s) \, \mathrm{d}s \right)
\end{aligned} \tag{2.17}$$

In the above $\vec{n}_{e,f}$ denotes the outward normal unit vector on the boundary of Ω_e, pointing outward from Ω_e to Ω_f. With this notation it is clear that $\vec{n}_{e,f} = -\vec{n}_{f,e}$. Accordingly for the case where the integrands under the interface integrals are continuous across elements the integrals would cancel each other out.

We introduce some additional notation for the interface terms:

Definition 2.5 (Jump Operator). The *jump* of a piecewise continuous function x on the interface between two adjacent elements Ω_e and Ω_f is denoted by the linear operator

$$[\![x]\!]_{e,f} = x|_{\partial\Omega_e}\vec{n}_{e,f} + x|_{\partial\Omega_f}\vec{n}_{f,e}$$

Note that for scalar x the jump operator $[\![x]\!]_{e,f}$ becomes vector valued and for vector valued x it becomes a scalar.

Definition 2.6 (Average Operator). The *average* of a piecewise continuous function x on the interface is given by the arithmetic mean

$$\langle x\rangle_{e,f} = \frac{1}{2}\left(x|_{\partial\Omega_e} + x|_{\partial\Omega_f}\right)$$

Remark 2.1. *Note that the choice of the arithmetic mean for the average operator is not necessarily optimal. In fact Ern showed in [10] that weighted averages depending on the values of σ on each side of the intersection yield optimal convergence results for heterogeneous diffusion problems. Weighted averages will be used in this thesis as well but for the sake of simplicity we will postpone presenting details for now. The following derivations will be valid for both choices for the average.*

The jump and average operator fulfil the following basic identity which will be used shortly:

Theorem 2.1. For two piecewise continuous functions x and y the following equation holds:

$$[\![xy]\!]_{e,f} = [\![x]\!]_{e,f}\langle y\rangle_{e,f} + [\![y]\!]_{e,f}\langle x\rangle_{e,f} \tag{2.18}$$

With the notations introduced in Definitions 2.5 and 2.6 the term in (2.17) becomes

$$\begin{aligned}&\sum_{\gamma_{e,f}\in\Gamma_{int}}\left(\int_{\gamma_{e,f}\cap\partial\Omega_e}\sigma(s)\nabla u(s)\cdot\vec{n}_{e,f}v(s)\,\mathrm{d}s + \int_{\gamma_{e,f}\cap\partial\Omega_f}\sigma(s)\nabla u(s)\cdot\vec{n}_{f,e}v(s)\,\mathrm{d}s\right)\\ &= \sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![\sigma\nabla uv]\!]_{e,f}(s)\,\mathrm{d}s.\end{aligned} \tag{2.19}$$

Since it should always be clear from the context on which interface the jump/average is evaluated we will write $[\![x]\!]$ instead of $[\![x]\!]_{e,f}$ and $\langle x\rangle$ instead of $\langle x\rangle_{e,f}$.

Using property (2.18) we can expand (2.19) to be

$$\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![\sigma\nabla uv]\!](s)\,\mathrm{d}s = \sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![\sigma\nabla u]\!](s)\langle v\rangle(s)+\langle\sigma\nabla u\rangle(s)[\![v]\!](s)\,\mathrm{d}s. \tag{2.20}$$

Considering the fact that u is the solution of (2.10a) and making the physically sound assumption that the source term $\vec{j}$ is continuous across element borders, we see that $[\![\sigma\nabla u]\!] = [\![\vec{j}]\!] = 0$. Thus the first summand on the right hand side of (2.20) vanishes and together with (2.16) we get the following left hand side for the weak formulation:

$$-\int_\Omega \sigma(x)\nabla u(x)\cdot\nabla v(x)\,\mathrm{d}x+\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma\nabla u\rangle(s)[\![v]\!](s)\,\mathrm{d}s \tag{2.21}$$

Following the idea of the Subtraction Approach introduced in Section 1.5 we split up

$$u = u^{corr,y}+u^{\infty,y} \quad \text{and} \quad \sigma = \sigma^{corr,y}+\sigma^{\infty,y} \tag{2.22}$$

which leaves us at

$$\begin{aligned}&-\int_\Omega \sigma(x)\nabla u^{corr,y}(x)\cdot\nabla v(x)\,\mathrm{d}x-\int_\Omega \sigma(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)\,\mathrm{d}x\\ &+\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma\nabla u^{corr,y}\rangle(s)[\![v]\!](s)\,\mathrm{d}s+\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)\,\mathrm{d}s.\end{aligned} \tag{2.23}$$

Since we reformulated the problem using the Subtraction Approach we are now solving for $u^{corr,y}$ and the terms involving $u^{\infty,y}$ are moved to the right hand side of the problem as they are not part to the bilinear form.

On the right hand side of (2.10a) we proceed in the same way as on the left hand side by testing the source term f with a test function $v\in V_h$ and integrating over Ω. Together with the terms that were moved from the left hand side we obtain

$$\int_\Omega fv dx+\int_\Omega \sigma(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)dx-\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)ds$$

for the right hand side.
Splitting up σ as described in (2.22) the above becomes

$$\begin{aligned}&\int_\Omega fv dx+\int_\Omega \sigma^{\infty,y}(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)\,\mathrm{d}x\\ &+\int_\Omega \sigma^{corr,y}(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)\,\mathrm{d}x-\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)\,\mathrm{d}s\end{aligned} \tag{2.24}$$

A general formula for the element wise partial integration that was used on the left hand side can be found in A.6. Using this formula in the form (A.7) on the second term of 2.24 gives us:

$$\int_\Omega f v \,\mathrm{d}x - \int_\Omega \nabla \cdot \sigma^{\infty,y}(x)\nabla u^{\infty,y}(x)v(x)dx \tag{2.25}$$

$$+\int_{\partial\Omega} \sigma^{\infty,y}(s)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)\,\mathrm{d}s + \sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}} [\![\sigma^{\infty,y}\nabla u^{\infty,y}v]\!](s)\,\mathrm{d}s \tag{2.26}$$

$$+\int_\Omega \sigma^{corr,y}(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)\,\mathrm{d}x - \sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}} \langle\sigma\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)\,\mathrm{d}s \tag{2.27}$$

The terms in line (2.25) vanish due to the assumptions of the Subtraction Approach(see Section 1.5).
The skeleton term in line (2.26) can be treated as in (2.20) using property (2.18):

$$\begin{aligned}&\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}} [\![\sigma^{\infty,y}\nabla u^{\infty,y}v]\!](s)ds\\ &=\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}} [\![\sigma^{\infty,y}\nabla u^{\infty,y}]\!](s)\langle v\rangle(s)+\langle\sigma^{\infty,y}\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)ds\end{aligned} \tag{2.28}$$

Using the fact that $\sigma^{\infty,y}\nabla u^{\infty,y}$ fulfils the partial differential equation in a homogeneous volume and therefore does not jump (i.e. $[\![\sigma^{\infty,y}\nabla u^{\infty,y}]\!]=0$), the first summand in line (2.28) vanishes. The remaining term in this line can be combined with the last term in (2.27) using $\sigma^{\infty,y}-\sigma = -\sigma^{corr,y}$ and the linearity of the average operator.
Overall we arrive at the following formulation for the right hand side

$$\begin{aligned}&\int_\Omega \sigma^{corr,y}(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)dx+\int_{\partial\Omega}\sigma^{\infty,y}(s)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)ds\\ &-\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma^{corr,y}\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)ds.\end{aligned}$$

Together with the left hand side the problem has the following weak formulation:

Definition 2.7 (Consistent Weak DG Formulation)**.**

$$\begin{aligned}&-\int_\Omega\sigma(x)\nabla u^{corr,y}(x)\cdot\nabla v(x)\,\mathrm{d}x+\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma\nabla u^{corr,y}\rangle(s)[\![v]\!](s)\,\mathrm{d}s\\ &=\int_\Omega\sigma^{corr,y}(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)\,\mathrm{d}x+\int_{\partial\Omega}\sigma^{\infty,y}(s)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)\,\mathrm{d}s\\ &-\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma^{corr,y}\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)\,\mathrm{d}s\end{aligned}$$

Consistency

It can be shown that the solution $u^{corr,y} \in H^1(\Omega)$ for the weak formulation of the Subtraction Approach as described in Definition 1.3 fulfils the equation above. As a matter of fact the derivation of the above formulation was conducted in such a fashion that consistency was obtained. Since the proof of consistency for the method amounts in large parts to the reversal of the above shown equalities we forgo the proof at this point and refer to Chapter B of the Appendix.

2.3.2. Stabilization of the Weak Formulation

As we have just stated, the above formulation is by construction consistent. However the method is not stable due to its lack of coercivity. For more details on this see for example [27] or [4]. We will now extend the bilinear form introduced above to obtain a coercive formulation.

As already discussed in Remark 2.1 we will now revisit the definition of the average operator and switch to a formulation using weighted averages. Note that this reformulation is not necessary to obtain coercivity of the method but will be used because it is better suited for the kind of problem we are dealing with. Following Ern [10] the average operator $\langle x \rangle_{e,f}$ will now be defined as

$$\langle x \rangle_{e,f} := \omega_{e,f} x|_{\partial\Omega_e} + \omega_{f,e} x|_{\partial\Omega_f}$$

where

$$\omega_{e,f} := \frac{\sigma_f}{\sigma_e + \sigma f}, \quad \text{respectively} \quad \omega_{f,e} := \frac{\sigma_e}{\sigma_e + \sigma_f}$$

In the above σ_i denotes the (constant) conductivity in element i for each element in the partition $\mathscr{T}_h$ or rather the scalar value that can be identified with the conductivity in the case of isotropic diffusion.

Remark 2.2. *The following derivations can be made for the case of anisotropic conductivities in a similar fashion. Obviously adjustments with regard to the dimensionality of the weights have to be made. For more on that we refer once more to [27].*

Using the modified average operator the consistent bilinear form can be supplemented with the following term to obtain coercivitiy:

$$-\eta \frac{\kappa_{e,f}}{h_{e,f}} \sum_{\gamma_{e,f} \subset \Gamma_{int}} \int_{\gamma_{e,f}} [\![u^{corr,y}]\!](s) [\![v]\!](s) \, \mathrm{d}s \tag{2.29}$$

In this term $h_{e,f}$ denotes the area of $\gamma_{e,f}$ and is actually depending on the intersection $\gamma_{e,f}$ in the case of non-uniform grids. The term $\kappa_{e,f}$ denotes the harmonic average of the conductivities on the elements e and f:

$$\kappa_{e,f} = \frac{2\sigma_e \sigma f}{\sigma_e + \sigma_f}$$

The value of the penalty parameter η is a positive scalar and has to be chosen large enough in order to ensure the coercivity of the method. We will derive a lower bound for η for which coercivity can be ensured later on.

Another term living on the internal skeleton of the triangulation is added:

$$\varepsilon \sum_{\gamma_{e,f} \in \Gamma_{int}} \int_{\gamma_{e,f}} \langle \sigma \nabla v \rangle (s) [\![u^{corr,y}]\!] (s) \, \mathrm{d}s \tag{2.30}$$

For $\varepsilon \in \{-1, 0, 1\}$ this term switches between the *Non Symmetric Interior Penalty Galerkin Method*(NIPG), the *Incomplete Interior Penalty Galerkin Method*(IIPG) and the *Symmetric Interior Penalty Galerkin Method*(SIPG) respectively. For an overview of different DG methods see [9].
In this thesis we will limit ourselves to the SIPG method. In the case of weighted averages this method is also called *Symmetric Weighted Interior Penalty Galerkin Method(SWIPG or SWIP)*. Accordingly we will choose ε to be 1 from now on.

Remark 2.3. *The addition of the two new terms does not spoil the consistency of the method. Since* $[\![u^{corr,y}]\!] = 0$ *for the exact solution of the equation, the additional terms are equal to zero and consistency is obtained in the same fashion as described in the Appendix.*

Adding the terms (2.29) and (2.30) to the left hand side of the formulation in Definition 2.7, we arrive at the final weak formulation of the problem:

Definition 2.8 (Weak DG Formulation of the Subtraction Approach).

$$
\begin{aligned}
&-\int_\Omega \sigma(x)\nabla u^{corr,y}(x)\cdot\nabla v(x)\,\mathrm{d}x+\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma\nabla u^{corr,y}\rangle(s)[\![v]\!](s)+\langle\sigma\nabla v\rangle(s)[\![u^{corr,y}]\!](s)\,\mathrm{d}s\\
&-\eta\frac{\kappa_{e,f}}{h_{e,f}}\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![u^{corr,y}]\!](s)[\![v]\!](s)\,\mathrm{d}s\\
&=\int_\Omega \sigma^{corr,y}(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)\,\mathrm{d}x+\int_{\partial\Omega}\sigma^{\infty,y}(s)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)\,\mathrm{d}s\\
&-\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma^{corr,y}\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)\,\mathrm{d}s
\end{aligned}
$$

Defining the left hand side to be the bilinear form $\alpha(u^{corr,y},v)$ and the right hand side to be the linear form $\lambda(v)$ the problem can be restated in the following way:

$$\alpha(u^{corr,y},v) \quad = \quad \lambda(v) \tag{2.31}$$

Remark 2.4. *In addition to the above equality another condition must be met in order to ensure the uniqueness of the solution since we are dealing with a homogeneous Neumann problem (see Section 1.3). This will especially become relevant when implementing the problem. We will come back to this issue when discussing the numerical simulations in Chapter 4.*

Coercivity

We will go on to show the coercivity of the weak DG formulation closely following [29]. Then using the Lax-Milgram Lemma (see A.1 in the Appendix) we obtain the well-posedness of the problem.
In order to do so we turn to the equivalent problem formulation

$$-\alpha(u^{corr,y},v)=-\lambda(v)$$

and show its coercivity, i. e.

$$\forall v_h\in V_h,\quad -\alpha(v_h,v_h)\geq C\|v_h\|_\sigma^2 \tag{2.32}$$

with a constant $C>0$ and the energy norm $\|v_h\|_\sigma$ defined by

$$\|v\|_\sigma:=\left(\|\sigma^{1/2}\nabla_h v\|^2_{[L^2(\Omega)]^d}+|v|^2_{J,\sigma}\right)^{1/2}. \tag{2.33}$$

The *jump seminorm* is defined by

$$|v|_{J,\sigma} := \left(\sum_{\gamma_{e,f} \in \Gamma_{int}} |v|^2_{J,\sigma,(e,f)} \right)^{1/2} \quad \text{and } |v|_{J,\sigma,(e,f)} := \left(\frac{\kappa_{e,f}}{h_{e,f}} \right)^{1/2} \|[\![v]\!]\|_{L^2(\gamma_{e,f})}. \tag{2.34}$$

Here $\nabla_h v$ denotes the *broken gradient* of v (see Appendix A for more information on that). Using this norm we can now show the coercivity of the bilinear form:

$$\begin{aligned} -\alpha(v_h, v_h) = &\int_\Omega \sigma(x) \nabla v_h(x) \cdot \nabla v_h(x) \, \mathrm{d}x \\ &- \sum_{\gamma_{e,f} \in \Gamma_{int}} \int_{\gamma_{e,f}} \langle \sigma \nabla v_h \rangle(s) [\![v_h]\!](s) + \langle \sigma \nabla v_h \rangle(s) [\![v_h]\!](s) \, \mathrm{d}s \\ &+ \eta \frac{\kappa_{e,f}}{h_{e,f}} \sum_{\gamma_{e,f} \in \Gamma_{int}} \int_{\gamma_{e,f}} [\![v_h]\!](s) [\![v_h]\!](s) \, \mathrm{d}s \\ = &\|\sigma^{1/2} \nabla_h v\|^2_{[L^2(\Omega)]^d} + \eta |v_h|^2_{J,\sigma} \\ &- 2 \sum_{\gamma_{e,f} \in \Gamma_{int}} \int_{\gamma_{e,f}} \langle \sigma \nabla v_h \rangle(s) [\![v_h]\!](s) \, \mathrm{d}s \end{aligned}$$

It can be inferred (see Lemma A.4 in the Appendix) that

$$\left| \sum_{\gamma_{e,f} \in \Gamma_{int}} \int_{\gamma_{e,f}} \langle \sigma \nabla_h v_h \rangle [\![v_h]\!] \right| \leq |v_h|_{\dagger,\sigma,2} |v_h|_{J,\sigma}$$

with

$$|v|_{\dagger,\sigma,2} := \left(\sum_{E \in \mathscr{T}_h} \sum_{\gamma \in \Gamma_{int} : \gamma \subset \partial E} h_\gamma \|\sigma^{1/2} \nabla v \,|_E \cdot \vec{n}_\gamma\|^2_{L^2(\gamma)} \right)^{1/2} \tag{2.35}$$

where $\vec{n}_\gamma$ denotes the outward normal unit vector on the interface γ.
Thus we have

$$\begin{aligned} -\alpha(v_h, v_h) \geq &\|\sigma^{1/2} \nabla_h v\|^2_{[L^2(\Omega)]^d} + \eta |v_h|^2_{J,\sigma} \\ &- 2 |v_h|_{\dagger,\sigma,2} |v_h|_{J,\sigma} \end{aligned}$$

Using the discrete trace inequality A.2 in the Appendix, we can derive the inequality A.3:

$$|v_h|_{\dagger,\sigma,2} \leq C_{tr} N_\partial^{1/2} \|\sigma^{1/2} \nabla_h v_h\|_{[L^2(\Omega)]^d} \tag{2.36}$$

Here C_{tr} is a positive scalar and N_∂ denotes the maximum number of interfaces for all the elements per element in the triangulation $\mathscr{T}_h$.

Accordingly we get:

$$\begin{aligned} -\alpha(v_h, v_h) \geq & \|\sigma^{1/2}\nabla_h v\|^2_{[L^2(\Omega)]^d} + \eta |v_h|^2_{J,\sigma} \\ & - 2C_{tr}N_\partial^{1/2} \|\sigma^{1/2}\nabla_h v_h\|_{[L^2(\Omega)]^d} |v_h|_{J,\sigma} \end{aligned}$$

For all $x, y \in \mathbb{R}$ and $\beta > 0$ the basic inequality

$$x^2 - 2\beta xy + \eta y^2 \geq \frac{\eta - \beta^2}{1+\eta}(x^2 + y^2)$$

holds. With $x = \|\sigma^{1/2}\nabla_h v_h\|_{[L^2(\Omega)]^d}$, $y = |v_h|_{J,\sigma}$ and $\beta = C_{tr}N_\partial^{1/2}$, we get:

$$-\alpha(v_h, v_h) \geq \underbrace{\frac{\eta - \beta^2}{1+\eta}}_{=:C} \underbrace{(\|\sigma^{1/2}\nabla_h v_h\|^2_{[L^2(\Omega)]^d} + |v_h|^2_{J,\sigma})}_{=\|v_h\|^2_\sigma} \tag{2.37}$$

Thus we can now ensure coercivity for the bilinear form $-\alpha(u, v)$ for choices of $\eta > C_{tr}^2 N_\partial$. □

It should be noted that the value of C_{tr} only depends on the dimension d of the problem and the degree k of the basis polynomials. Using this inequality, optimal values for η can be found heuristically. This is a one-time setup cost. Also the implementation automatically adapts to the polynomial degree and the dimensionality of the problem. Accordingly the value of η has to be determined only once and can then be used for every other polynomial degree.

2.4. Convergence Analysis

In Ern [29] it was shown that for the model problem

$$\begin{aligned} -\nabla \cdot (\sigma \nabla u) &= f \qquad \text{in } \Omega, \\ u &= 0 \qquad \text{on } \partial\Omega. \end{aligned} \tag{2.38}$$

with $f \in L^2(\Omega)$, $\sigma \in L^\infty(\Omega)$ and σ bounded, the corresponding bilinear form converges with optimal algebraic rates in the energy norm if $u \in H^p(\Omega)$ with $p \in (6/5, 2]$.
In our case the regularity for the solution $u^{corr,y}$ can not be assumed to be higher than $u^{corr,y} \in H^1(\Omega)$ without making further assumptions. However, as already discussed in Section 1.5, we can assume the conductivity tensor σ to be constant on each element $E \in \mathscr{T}$. In addition to that we can restrict ourselves to the case in which the elements in our triangulation $\mathscr{T}_h$ are open polyhedra. Then, following the argumentation in [30], we can infer the existence of a $p \in (1, 2]$ such that

$$u^{corr,y} \in H^p.$$

For this kind of minimal regularity solutions optimal convergence rates can not be shown, however in [29] convergence with non-optimal rates is proven for the model problem (2.38). Due to the Neumann boundary conditions and the non standard right hand side of our problem the existing proof can not be adapted without further modifications. Especially finding estimates for the right hand side terms does not seem to be a trivial task.
The proof as presented in [29] is based on the reformulation of the bilinear form using the *lifting operator* $r^l_{h,\gamma_{e,f}} : L^2(\gamma_{e,f}) \to \left[\mathbb{P}^l_3(\mathscr{T}_h)\right]^3$ on all $\gamma_{e,f} \in \Gamma_{int}$ such that for all $\zeta \in L^2(\gamma_{e,f})$ and all integers $l \geq 0$

$$\int_\Omega r^l_{h,\gamma_{e,f}}(\zeta) \cdot \tau_h \,\mathrm{d}x = \int_{\gamma_{e,f}} \zeta \langle \tau_h \rangle \,\mathrm{d}s \qquad \forall \tau_h \in \left[\mathbb{P}^l_3(\mathscr{T}_h)\right]^3, \tag{2.39}$$

the *global lifting*

$$R^l_h(\zeta) := \sum_{\gamma_{e,f} \in \Gamma_{int}} r^l_{h,\gamma_{e,f}}(\zeta). \tag{2.40}$$

and the *discrete gradient*

$$G^l_h(v_h) := \nabla_h v_h - R^l_h(\llbracket v_h \rrbracket). \tag{2.41}$$

With these notations the bilinear form can be rewritten as

$$-\alpha(u_h, v_h) = \int_\Omega \sigma G^l_h(u_h) \cdot G^l_h(v_h) + j_h(u_h, v_h) \tag{2.42}$$

where $j_h(u_h, v_h) := -\int_\Omega \sigma R^l_h(\llbracket u_h \rrbracket) \cdot R^l_h(\llbracket v_h \rrbracket)\,\mathrm{d}x + \sum_{\gamma_{e,f} \in \Gamma_{int}} \frac{\eta \kappa_{e,f}}{h_{e,f}} \int_{\gamma_{e,f}} \llbracket u_h \rrbracket \llbracket v_h \rrbracket \,\mathrm{d}s$. Using certain properties of the discrete gradient and an estimate for the right hand side, the convergence of $u_h \to u$ strongly in $L^2(\Omega)$, $\nabla_h u_h \to \nabla u$ strongly in $\left[L^2(\Omega)\right]^3$ and $|u_h|_J \to 0$ can be shown, where

$$|u_h|_J := \left(\sum_{\gamma_{e,f} \in \Gamma_{int}} h^{-1}_{e,f} \| \llbracket u_h \rrbracket \|^2_{L^2(\gamma_{e,f})} \right)^{1/2} \tag{2.43}$$

However the proof relies on estimates for the right hand side. In the case of $\lambda(v)$ as stated in Definition 2.8 this does not seems require a different approach. Accordingly, in this thesis we will have to rely on numerical observations with regard to this aspect.

2.5. Choice of Basis Functions

The choice of the element local basis functions for the discretized space impacts the accuracy and stability of the method. Especially when dealing with higher degree basis functions it is important to use a basis which does not spoil the condition of the resulting system. To

obtain this property L^2-*orthonormal* basis functions can be used, i. e. basis functions which are orthonormal with respect to the standard inner product on L^2. Furthermore the basis should be designed in a hierarchical sense, i.e. the basis of degree k should include the basis functions of degree less than k. In the numerical implementation of the DG method for this thesis two types of basis functions were used that meet both requirements and correspond to the broken polynomial spaces introduced in (2.11) and (2.12).
On tetrahedral meshes an orthogonalized $\mathbb{P}_k$ basis was used. While there is of course a much higher number of degrees of freedom and discontinuities are allowed to occur across elements, this choice can be seen as an emulation of Lagrange basis polynomials on simplices in the conforming finite element case. In the 2D-case the $\mathbb{P}_1$ polynomials would correspond to the basis $\{1,x,y\}$ and $\mathbb{P}_2$ to $\{1,x,y,xy,x^2,y^2\}$ (which both would still have to be orthogonalized). Thus the $\mathbb{P}_k$ basis can be identified with the monomials whose exponents have 1-norm less or equal to k.
On hexahedral meshes the orthogonalized $\mathbb{Q}_k$ basis was used which includes all the monomials of exponents less or equal to k in the ∞-norm. In the 2D-case $\mathbb{P}_1$ would consist of an orthogonalized version of $\{1,x,y,xy\}$ and for $\mathbb{P}_2$ the basis would become $\{1,x,y,xy,x^2,y^2,xy^2,x^2y,x^2y^2\}$. This choice leads to a much larger number of degrees of freedom in the resulting matrix system but corresponds much better to the choice of $\mathbb{Q}_k$ Lagrange basis functions in the conforming finite element case. See the table below for an overview of the basis and the number of local basis functions corresponding to it for the case of a three dimensional problem. Also see Appendix A for a more formal discussion of the broken polynomial spaces.

	$\mathbb{P}_k^3$	$\mathbb{Q}_k^3$
k=1	4	8
k=2	10	27
k=3	20	64

Table 2.1.: Overview of the number of local basis functions corresponding to the respective basis.

2.6. Additional Properties of Discontinuous Galerkin Methods

We will now touch on some of the most prominent features of DG methods. We will not discuss many of the analytical properties but restrict ourselves to those features that are of special relevance to the application in EEG source analysis. For further details on properties of DG methods we again refer to [9] or [27].

2.6.1. Matrix Structure

Considering the weak formulation introduced in Definition 2.8 discretized on a $3 \times 3 \times 3$ hexahedral decomposition of the unit cube using basis functions of polynomial degree one we would get the matrix structure shown on the left hand side of Figure 2.3.

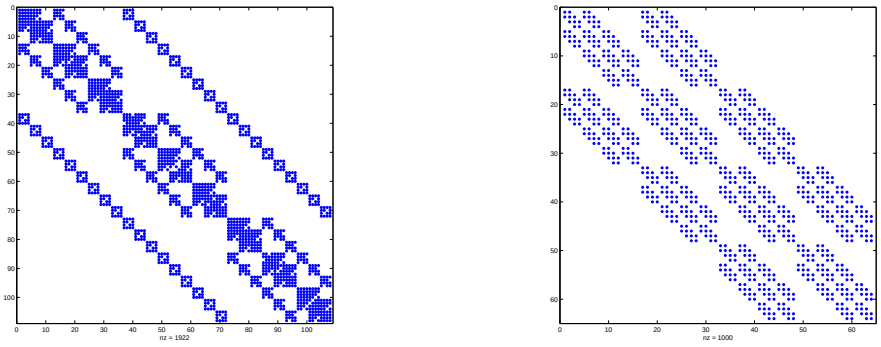

Figure 2.3.: Matrix Structure of a DG discretization (*left*) on a 3x3x3 cubic domain using polynomial of degree 1 for the local basis functions compared to that of a conforming FEM discretization using Lagrange basis functions (*right*).

As one can see the DG discretization leads to a matrix sparsity pattern that allows for blockwise storing of the matrix. The block-size is determined by the number of basis functions on each element. If there are n basis functions per element the block-size of the sparsity pattern becomes $n \times n$. In the above discretization four basis functions were used on each element, leading to a sparsity pattern with block size four. For the purpose of a more compact visualization we used $\mathbb{P}_k^3$ basis polynomials whereas for the simulations later on a $\mathbb{Q}_k^3$ basis will be used on hexahedral meshes. The choice of $\mathbb{Q}_k^3$ results in an overall larger matrix with a bigger block size while the block sparsity pattern remains unchanged.

Each diagonal block in the matrix holds the coupling between the local basis functions of one element. These couplings result from the volume term of the problem's left hand side but also from some of the skeleton terms. This can be seen by expanding the average- and jump-operator terms in the problem's formulation (see Definition 2.8). If only the volume terms of the discretized bilinear form were assembled the resulting matrix structure would be diagonal due to the L^2-orthonormality of the basis functions.
The off-diagonal blocks represent the coupling between basis functions of different elements. These couplings occur if the elements belonging to the respective basis functions share a face.

Then this face is in Γ_{int} and the corresponding skeleton term is part of the bilinear form.
Compared to the conforming FEM discretization (see right hand side in Figure 2.3) it can be seen that the conforming FEM does not allow for a block-wise storage of the matrix entries (excluding the trivial block size of one) and thus the matrix can not be stored as efficiently since more effort has to be undertaken to store the sparsity pattern. This is a result of the node based formulation of conforming FEM with Lagrange basis functions.
What is more the sparsity pattern of the FEM matrix will only get more complex if the polynomial degree is increased. For Discontinuous Galerkin methods however, the block-pattern of the matrix remains unchanged, if another local basis is used or the degree of the basis functions is increased on each element. The sparsity pattern only depends on the geometric properties of the grid, i.e. which element shares an interface with another element. The only thing that changes with respect to the matrix structure is the size of the blocks. If in the above scenario basis functions of polynomial order two were used we would get blocks of size 10 as shown in Figure 2.4.

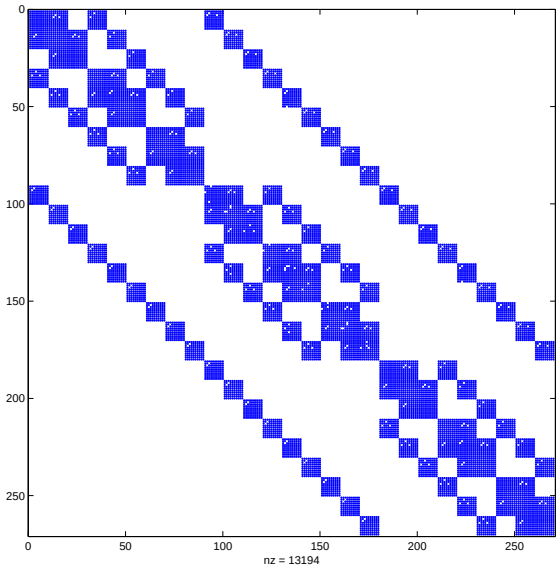

Figure 2.4.: Matrix Structure for basis functions of polynomial degree two.

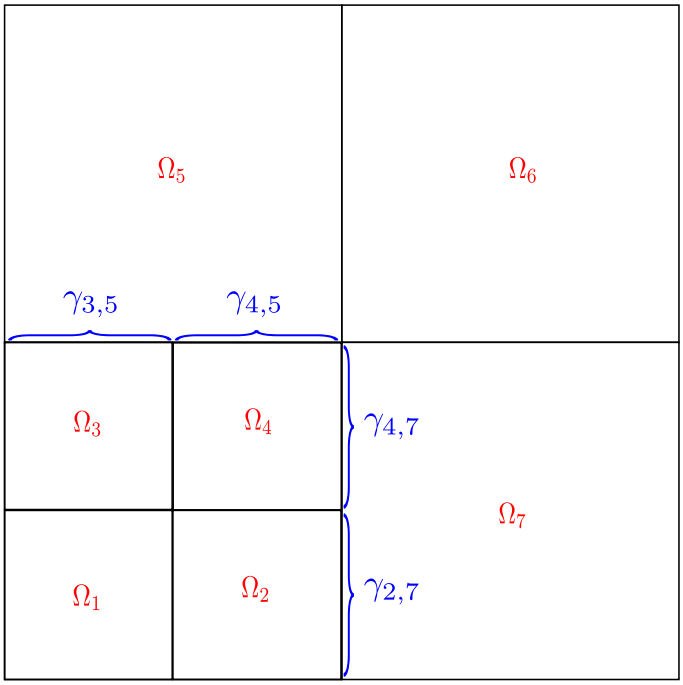

Figure 2.5.: Skeleton structure of a 2×2 domain with local refinement in the bottom left element.

2.6.2. Non-Conforming Meshes

One of the main qualities of the DG formulation introduced in the previous sections is that, unlike conforming FEM formulations, it is not node-based. Vertices of the mesh are not associated with degrees of freedom and they are only relevant for the geometric description of elements and their interfaces.
Therefore DG methods are by design well suited to handle problems on *non-conforming* meshes. A non-conforming mesh may be the result of local refinement, e. g. by the employment of an error estimator, and can be defined as a triangulation for which interfaces between two elements do not necessarily have to be an entire face of both elements. The definition of a triangulation in this chapter (see Definition 2.1) did not exclude this kind of meshes. Accordingly all the previous derivations are valid for non-conforming meshes as well. Figure 2.5 shows an example of a non-conforming mesh.
In the case of a conforming FEM discretization a lot of effort has to be put into the treatment of the degrees of freedom associated with the hanging nodes and the matrix structure

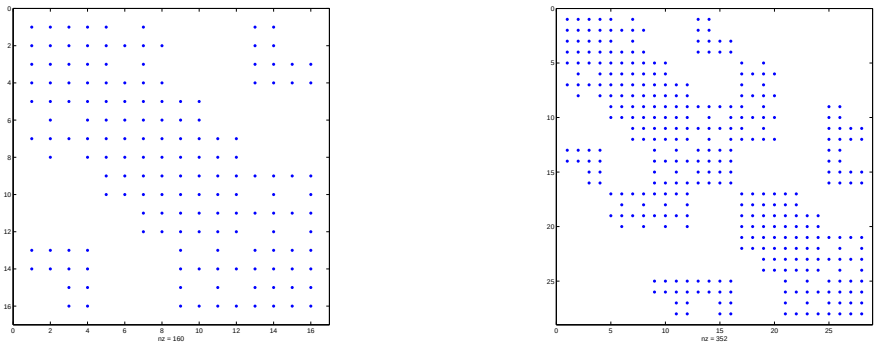

Figure 2.6.: Matrix structure of a DG discretization for a 2×2 domain (left) and the same domain with one refined element (right).

becomes unfit for standard solver techniques without further optimization. DG methods however do not suffer from that issue. The additional interfaces that occur when comparing a refined mesh with an unrefined mesh simply result in additional blocks in the matrix structure representing the coupling of the basis functions on the respective elements. For a comparison of matrix structures in that case see Figure 2.6. The matrix stays symmetric and can be treated in the same way as in the case of a conforming mesh.

2.6.3. More General Approximation Space

As we have mentioned before, the solution space for Discontinuous Galerkin methods includes functions with discontinuities across element borders. This makes them more suitable for problems involving discontinuities. Especially in the case of rough coefficients in heterogeneous diffusion problems like the EEG forward problem a discontinuous function might be a better approximation of the solution than a function with a forced continuity. To exemplify this issue we will present a one-dimensional example similar to that found in [27, p. 152]:
Consider the general diffusion problem

$$\begin{aligned} -\nabla \cdot (\sigma \nabla u) &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{2.44}$$

with $f \in L^2(\Omega)$.
We choose $\Omega = (-1,1)$ and the triangulation $\mathcal{T}_h = \{\Omega_1 = (-1,0), \Omega_2 = (0,1)\}$. On Ω_1 we set $\sigma|_{\Omega_1} = 0.0056 =: \sigma_1$ and on $\Omega_2 = \sigma|_{\Omega_2} = 1$. This choice of parameters approximately corresponds to the diffusion heterogeneity that can be observed at the conductivity border of the skull and the cerebrospinal fluid(CSF) when using isotropic conductivities. It also is the

maximal diffusion heterogeneity that can be expected when using a four layer sphere model with the usual isotropic conductivities.
The exact solution to the model problem (2.44) can be derived to be:

$$u(x) = \begin{cases} a_1(1+x)^2 + b_1(1+x) & \text{if } x \in \Omega_1 \\ a_2(x-1)^2 + b_2(x-1) & \text{if } x \in \Omega_2, \end{cases} \tag{2.45}$$

where $a_1 = -\frac{1}{2\sigma_1}$, $a_2 = -\frac{1}{2}$, $b_1 = \frac{1+3\sigma_1}{2\sigma_1(1+\sigma_1)}$ and $b_2 = -\frac{\sigma_1+3}{2(1+\sigma_1)}$.

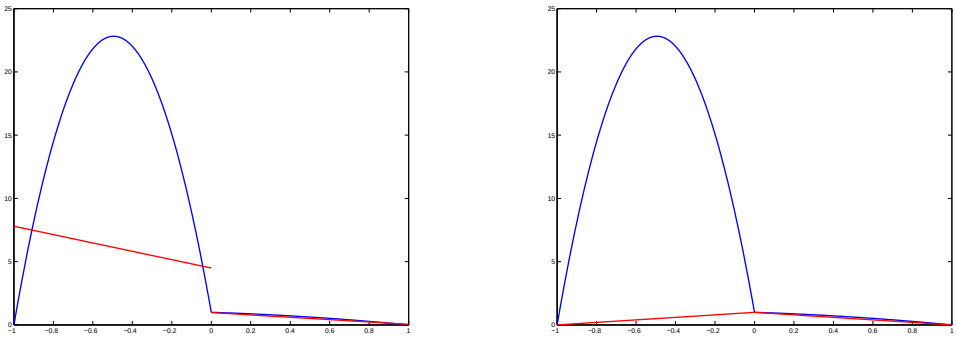

Figure 2.7.: Exact Solution (blue) of the 1D model problem approximated by linear basis functions. Solutions obtained by the DG method (left) and by conforming FEM (right).

The exact solution and its approximation obtained by the DG method and conforming FEM can be seen in Figure 2.7. In both cases linear basis functions were used. Obviously both solutions suffer from the fact that the quadratic term in the left element can not be properly approximated with linear basis functions. In the DG case (left figure) however it is obvious that the L^2-error is greatly reduced when compared to the conforming FEM solution. This increase in L^2 accuracy comes with a lack of continuity and fulfilment of the boundary condition in the solution. Of course a decision has to be made whether the loss of continuity is acceptable for the gain in L^2 accuracy.
Using a basis of order two, the quadratic term on the left hand side could be solved exactly making this example a candidate for local p-adaptivity: Increasing the degree of the local basis functions by one on the left element would produce the exact solution on the left subdomain.

2.6.4. Higher Order Basis Functions

As mentioned before, the implementation of higher order basis functions for DG methods is pretty straightforward. The polynomial degree of the basis functions can be chosen quite arbitrarily, limited of course by the computational costs in practical applications. In contrast to conforming FEM the use of higher order approximations is the norm rather than the exception and one of the main advantages when using DG methods. Naturally a better approximation can be expected when using DG methods of higher degree. What is more the polynomial degree of the local basis functions can, at least in theory, be adapted individually for each element.
For example in the case of the EEG forward problem the polynomial degree at the conductivity jumps could be increased in order to better suit the jump in conductivity and the consequently steep gradients. Combined with the capability for handling locally refined meshes (see Subsection 2.6.2) this makes DG methods a natural candidate for *hp-adaptivity*.

2.6.5. Parallelization

In scientific computing parallelization has become a major focus as a means to efficiently solve large problems on parallel architectures. The higher computational cost that comes with local basis functions makes parallelization an appealing feature for DG methods, especially in the case of higher order basis functions. In this case the memory demand, even on hexahedral meshes of relatively moderate size, easily surpasses the capacity of average modern desktop computers (for more on that see Section 4.5). Parallelization on a distributed memory architecture could help amend this problem. Although it was not applied in this thesis, parallelization of a DG formulated problem should be pretty straightforward:
Based on a matrix structure as shown for example in Figure 2.4, each computational node would assemble and store the diagonal matrix blocks that correspond to the elements that were assigned to it by the domain decomposition. Furthermore all the local couplings, i. e. the off-diagonal blocks that only involve elements contained in the node's own subdomain will be stored. Communication is only necessary for the DOFs associated with off-diagonal blocks that describe couplings between elements in different subdomains of the domain decomposition. This behaviour only depends on the geometrical information of the mesh and the domain decomposition respectively. In particular it is independent of the polynomial degree of the basis functions which is another advantage of DG methods over conforming finite element methods where with the complexity of the sparsity pattern the communication tends to become more involved. In the DG case the only thing that changes is the number of the communicated DOFs while the communication structure remains the same.

2.6.6. Mass Conservation

Mass conservation is a basic physical property and therefore it is a desirable property for a numerical method as well. Consider a simple Poisson problem:

$$-\Delta u = f \quad \text{in } \Omega \qquad \text{and} \qquad u = 0 \quad \text{on } \partial\Omega \tag{2.46}$$

For $E \in \mathscr{T}_h$ and $v \in \mathbb{P}_d^k(E)$ the exact solution fulfils the equality:

$$\int_E f v \,\mathrm{d}x = -\int_E v \Delta u \,\mathrm{d}x = \int_E \nabla u \cdot \nabla v \,\mathrm{d}x - \int_{\partial E} \nabla u \cdot \vec{n} v \,\mathrm{d}s \tag{2.47}$$

Choosing $v \equiv 1$ and defining $\Phi_{\partial E}(u) := -\nabla u \cdot \vec{n}$ we obtain

$$\int_E f \,\mathrm{d}x = -\int_{\partial E} \nabla u \cdot \vec{n} \,\mathrm{d}s = \int_{\partial E} \Phi_{\partial E}(u) \,\mathrm{d}s. \tag{2.48}$$

The term $\Phi_{\partial E}(u)$ is called the *exact numerical flux* and the above equality describes the local conservation property fulfilled by the exact solution u. A similar relationship can be established for the DG method corresponding to the model problem (2.46) on a discrete level: The discretized problem in DG formulation reads

$$\text{Find } u_h \in V_h \text{ s. t. } \alpha_h^{poiss}(u_h, v_h) = \int_\Omega f v_h \,\mathrm{d}x \quad \text{for all } v_h \in V_h \tag{2.49}$$

where the bilinear form α_h^{poiss} is derived in a similar fashion as for the subtraction formulation and turns out to be

$$\begin{aligned} \alpha_h^{poiss}(u_h, v_h) = &\int_\Omega \nabla u_h \cdot \nabla v_h \,\mathrm{d}x - \sum_{\gamma_{e,f} \in \Gamma} \int_{\gamma_{e,f}} \langle \nabla u_h \rangle [\![v_h]\!] + [\![u_h]\!] \langle \nabla v_h \rangle \,\mathrm{d}s \\ &+ \sum_{\gamma_{e,f} \in \Gamma} \frac{\eta}{h_{e,f}} \int_{\gamma_{e,f}} [\![u_h]\!] [\![v_h]\!] \,\mathrm{d}s. \end{aligned} \tag{2.50}$$

Let χ_E denote the characteristic function of E. Then we can insert the element local basis function $v_h = 1\chi_E$ into the above bilinear form and obtain:

$$\begin{aligned} \alpha_h^{poiss}(u_h, 1\chi_E) &= -\int_{\partial E} \langle \nabla u_h \rangle \vec{n} + \frac{\eta}{h} [\![u_h]\!] \vec{n} \,\mathrm{d}s \\ &= \int_{\partial E} \underbrace{(-\langle \nabla u_h \rangle + \frac{\eta}{h} [\![u_h]\!])\vec{n}}_{:=\Phi_{\partial E}^{num}} \,\mathrm{d}s \end{aligned} \tag{2.51}$$

The term $\Phi_{\partial E}^{num}$ is called the *numerical flux*. Since the exact solution u is continuous across element faces we can see that the numerical flux $\Phi_{\partial E}^{num}(u)$ of the exact solution u is equal to the exact flux $\Phi_{\partial E}$:

$$\Phi_{\partial E}^{num} = \int_{\partial E} (-\underbrace{\langle \nabla u \rangle}_{=\nabla u} + \frac{\eta}{h} \underbrace{[\![u]\!]}_{=0}) \vec{n} \,\mathrm{d}s = \Phi_{\partial E} \tag{2.52}$$

The conservation property discussed above is the same that can be derived for Finite Volume methods. It should be noted, however, that the choice of constant basis functions is only a special case of a more general conservation property that can be derived for DG methods. For more details on this we refer to [27, p. 142].

3. Implementation in *DUNE*

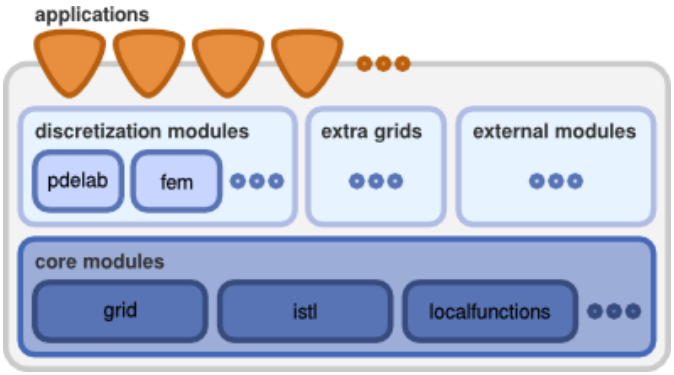

Figure 3.1.: Design of the DUNE toolbox. *Source: DUNE Project Homepage*

In the context of this thesis, both a Discontinuous Galerkin(DG) and a conforming Finite Element Method(FEM) version of the Subtraction Approach were implemented. The implementation was performed using the *Distributed and Unified Numerics Environment(DUNE)*[1] framework. The goal of this chapter is to give a concise overview of the design of the *DUNE* framework and the associated discretization module *PDELab*[2] as well as the implementation of the Subtraction Approach performed therein.

3.1. The *DUNE* Framework

The DUNE project is a modular C++ toolbox for solving partial differential equations(*PDEs*) using grid-based methods. It is published under the GNU General Public License[3] which makes it free and open-source software. DUNE makes extensive use of advanced object oriented C++ programming techniques such as template parameters and static polymorphism. This makes it possible to provide very general and abstract interfaces for added functionality and maintainability without having the computational overhead of evaluating template parameters at run-time.

[1] Homepage of the DUNE project: `http://www.dune-project.org`
[2] Homepage of the dune-pdelab module: `http://www.dune-project.org/pdelab/index.html`
[3] Version 2, with an andditional runtime exception, for more details see `http://dune-project.org/license`

The DUNE toolbox is divided into different modules (see Figure 3.1). Each of the core modules provides a certain set of basic functionality like for instance access to different kinds of grids through a common interface(*dune-grid*), implementations of basic geometric structures(*dune-geometry*) or access to a whole range of iterative solvers(*dune-istl*). Built on top of that are discretization modules like *dune-pdelab* or *dune-fem* and other modules, e. g. external modules or additional grid implementations.
The discretization modules act as the basic framework in which the partial differential equations are formulated and discretized. They take different approaches as to how this task is carried out. In this thesis *dune-pdelab* was used and we will later discuss some of its main features and design concepts.
Another feature of the DUNE framework is the reuse of already available Finite Element toolboxes and software. For example, the implementations created for this thesis make use of the *Adaptive, Load-balanced and Unstructured Grid* library(*ALUGrid*, see [17] and [18]) which serves as the *grid manager*, i. e. it handles the creation and manipulation of the grid. It is accessed through the common interface of the *dune-grid* module.
The modular structure of the DUNE project does not only make it easier to maintain and develop but also is beneficial to the user as any changes or improvements to the core modules, e. g. the implementation of a new solver in *dune-istl*, can be implemented by simply updating the respective core module and adding the new feature with just a few changes to the user's implementation.

3.2. The *DUNE-PDELab* Module

As mentioned above the *dune-pdelab* module was used for the discretization of the Subtraction DG Approach as well as for the conforming Finite Element Approach. In the following we will only describe the implementation of the DG approach as the implementation of the conforming Finite Element method is in many ways analogous to that of the DG approach but less relevant for this thesis.

In order to discretize the partial differential equations PDELab utilizes an *operator concept*: The user has to construct a class object, the so called *local operator* which provides an element local formulation of the PDE it is supposed to discretize. It provides a set of methods each representing a certain aspect of the problem's formulation. These methods can be categorized by the way they contribute to the PDE:

- `alpha_`-methods, representing contributions of the bilinear form α on the left hand side
- `lambda_`-methods, representing contributions of the linear form λ on the right hand side

See the weak formulation of the DG Subtraction Approach in Definition 2.8 for the shape that α and λ take in our case.

Both `alpha_`- and `lambda_`-methods are again organized into different groups depending on which kind of grid entity the corresponding integral lives on. Assuming a triangulation of our domain Ω there are three cases on which the terms in the α and λ forms can be defined:

- `_volume`, for volume terms
- `_boundary`, for boundary terms
- `_skeleton`, for skeleton terms

So for the Subtraction DG Approach as introduced before we have the following relationships for the `alpha_`-methods

$$\begin{aligned}
\texttt{alpha_volume} \rightarrow \quad & -\int_\Omega \sigma(x)\nabla u^{corr,y}(x)\cdot\nabla v(x)\,\mathrm{d}x \\
\texttt{alpha_boundary} \rightarrow \quad & \textit{not present in our problem} \\
\texttt{alpha_skeleton} \rightarrow \quad & \sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}} (\langle\sigma\nabla u^{corr,y}\rangle(s)[\![v]\!](s)+\varepsilon\langle\sigma\nabla v\rangle(s)[\![u^{corr,y}]\!](s) \\
& +\frac{\eta}{h}[\![u^{corr,y}]\!](s)[\![v]\!](s))\,\mathrm{d}s,
\end{aligned}$$

and for the `lambda_`-methods

$$\begin{aligned}
\texttt{lambda_volume} \quad &\rightarrow \quad \int_\Omega \sigma^{corr,y}(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)\,\mathrm{d}x \\
\texttt{lambda_boundary} \quad &\rightarrow \quad \int_{\partial\Omega} \sigma^{\infty,y}(s)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)\,\mathrm{d}s \\
\texttt{lambda_skeleton} \quad &\rightarrow \quad -\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma^{corr,y}\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)\,\mathrm{d}s
\end{aligned}$$

Remark 3.1. *It should be noted here that PDELab utilizes a residual formulation for the problem, i. e. all the terms are moved to either the right hand side or the left hand side and the resulting system is solved for zero. Thus we need to change the sign in front of either all the `alpha_`-terms or all the `lambda_`-terms when considering the implementation of the problem.*

Furthermore, the assembly of the matrix for the linear system requires the implementation of `jacobian_`-methods. The `jacobian_` part in the name of the methods is motivated by the fact that the matrix can be identified with the jacobian of a discretized version of the residual itself, that is

$$\frac{\partial}{\partial u}(Au-b)=A.$$

As mentioned earlier, all of these methods have to be implemented in the local operator. With the help of the local operator a global *grid operator* is constructed which visits all the entities in the grid and calls the corresponding method:`_volume`-methods for codim-0 entities, `_boundary`-methods for codim-1 entities making up the boundary of the domain and `_skeleton`-methods for codim-1 entities forming the internal skeleton of the mesh. Once the grid operator performed the assembly of the residual and the matrix we are left with a linear system to be handed over to an appropriate solver. We will discuss the implementation of the operator again in more detail in the next section.

3.3. Implementation and Work Flow

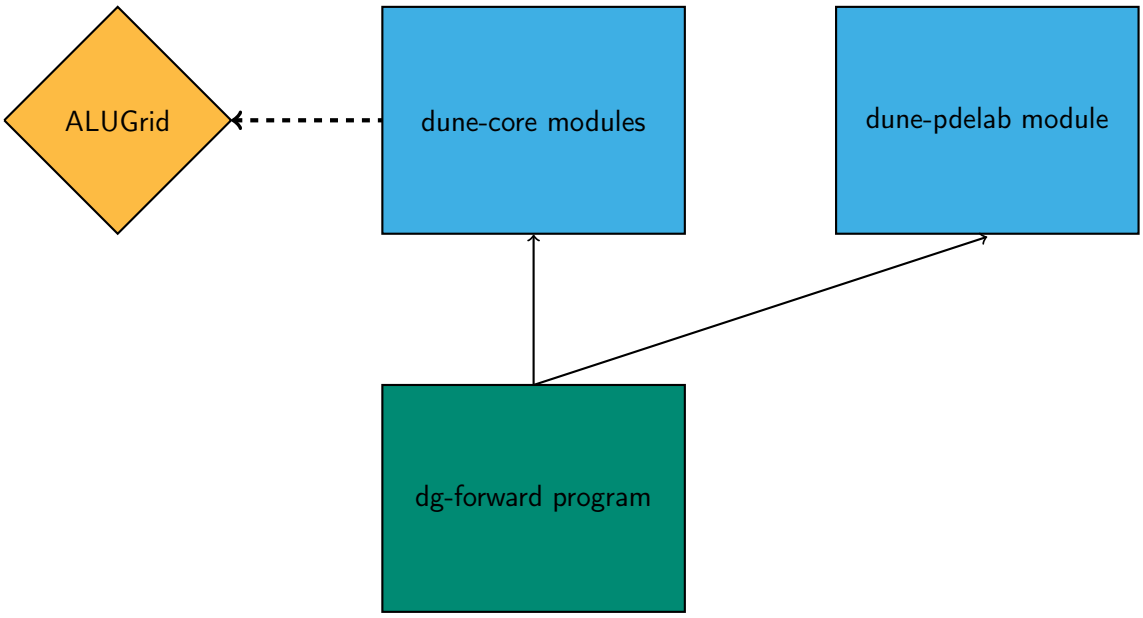

Figure 3.2.: Dependency structure of the *dg-forward* program.

In the following we will have a closer look at how the implementation was carried out and describe the work flow of the *dg-forward* application that was implemented to solve the DG forward problem. Again the implementation of the conforming FEM is very much alike and accordingly will not be discussed here.

On a macroscopic level the dependency structure between the *dg-forward* program and the DUNE core modules can be illustrated by Figure 3.2.
The dashed arrow from the box representing the *dune-core modules* to the *ALUGrid* rectangle

symbolizes an interface relationship. The external *ALUGrid* module is interfaced through the *dune-grid* module. However, this is not a hard dependency of the *dune-grid* module or the core modules as a whole but rather a choice between different options for the grid manager we made for this implementation. The current implementation reads the mesh using the `GmshReader` class which allows to import `.msh`-files[4].
The main application *dg-forward* directly depends on the core modules and on the *pdelab module* which is signified by the solid arrows starting from the program's box going to the respective rectangles. As illustrated before in Figure 3.1, the pdelab module also depends on the core modules but we neglected this dependency since it is not specific to the *dg-forward* application but a general property of the DUNE framework.

As described in the previous section, a crucial part of the implementation was the formulation of a local operator class. Here the programming effort was significantly reduced by the presence of a suitable operator called `ConvectionDiffusionDG` provided for the generic implementation of standard DG convection diffusion problems in the *dune-pdelab* module. This operator needs to be parametrised with a *parameter class* which provides the `lambda_-`, `alpha_-` and `jacobian_-`methods with the terms arising in the problem's formulation. However, due to the non standard nature of the right hand side for the Subtraction Approach the `lambda_-`methods of the `ConvectionDiffusionDG` class could not be inherited as they were:

- The standard implementation only allows for v to be used as a test function in the volume term of the right hand side. As described above we need to test against ∇v. Thus the `lambda_volume` method had to be reimplemented.
- `lambda_boundary` had to be newly implemented since the standard implementation does not provide this method.
- Furthermore it was necessary to implement `lambda_skeleton` since it was not present for the standard convection diffusion problem implemented in `ConvectionDiffusionDG`.

[4] The .msh-format is the file format for the 3D Finite Element grid generator *Gmsh*. For further information on that see `http://geuz.org/gmsh/` and also [22]

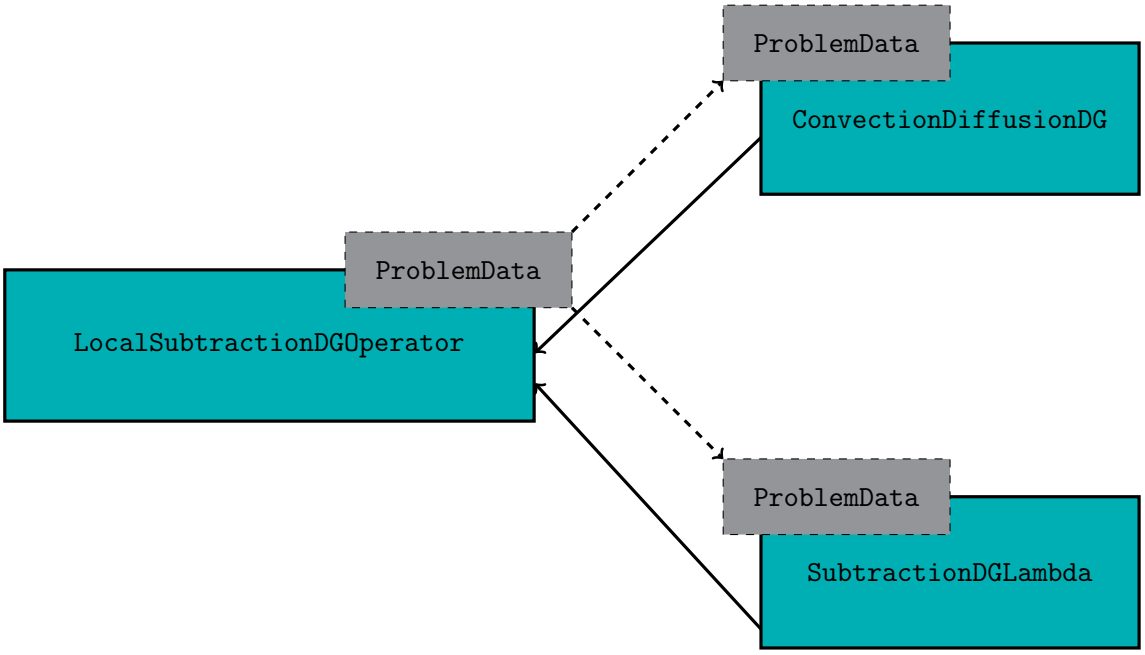

Figure 3.3.: Inheritance diagram of the `LocalSubtractionDGOperator`.

Those modified and new `lambda_`-methods were implemented in a class called `SubtractionDGLambda`. The class that will serve as the local operator was written and is called `LocalSubtractionDGOperator`. It does not have any methods implemented by itself but rather inherits them from `ConvectionDiffusionDG` and `SubtractionDGLambda`. The parameter class `ProblemDataSubtractionDG` was created which provides all the relevant terms occurring in the integrals which are needed by the local operator. This class parametrizes all the other classes, see Figure 3.3 for a visualization. Although the modifications required to the `ConvectionDiffusionDG` operator do not make our problem a text book example for generic programming, it should still be obvious that the implementation benefited greatly from this basic design principle.

Regarding the overall work flow of the *dg-forward* program we will give a rough overview in Figure 3.4, hiding most of the implementation details. Each source file (blue boxes on the left in the flowchart) corresponds to a specific C++ function performing the tasks explained in the light grey boxes on the right side. The modularity of this design allows for easy modification of the work flow. For instance, in order to implement a different dipole model (e. g. the Venant Approach) the `options_interface` routine could be extended to make a call to a newly implemented routine sitting in `venant_dg_approach.hh` while most of the other

source files would remain untouched. However the implementation of a new local operator would have to be carried out which then would be called in the `venant_dg_approach.hh` instead of the `LocalSubtractionDGOperator`. The programming work load for this would be rather mild compared to that of the Subtraction Approach since the right hand side of the Venant Approach should be comparably easy to implement. Nevertheless effort has to be put in adapting the node based formulation of the Venant Approach for conforming Finite Element methods into a Discontinuous Galerkin context.

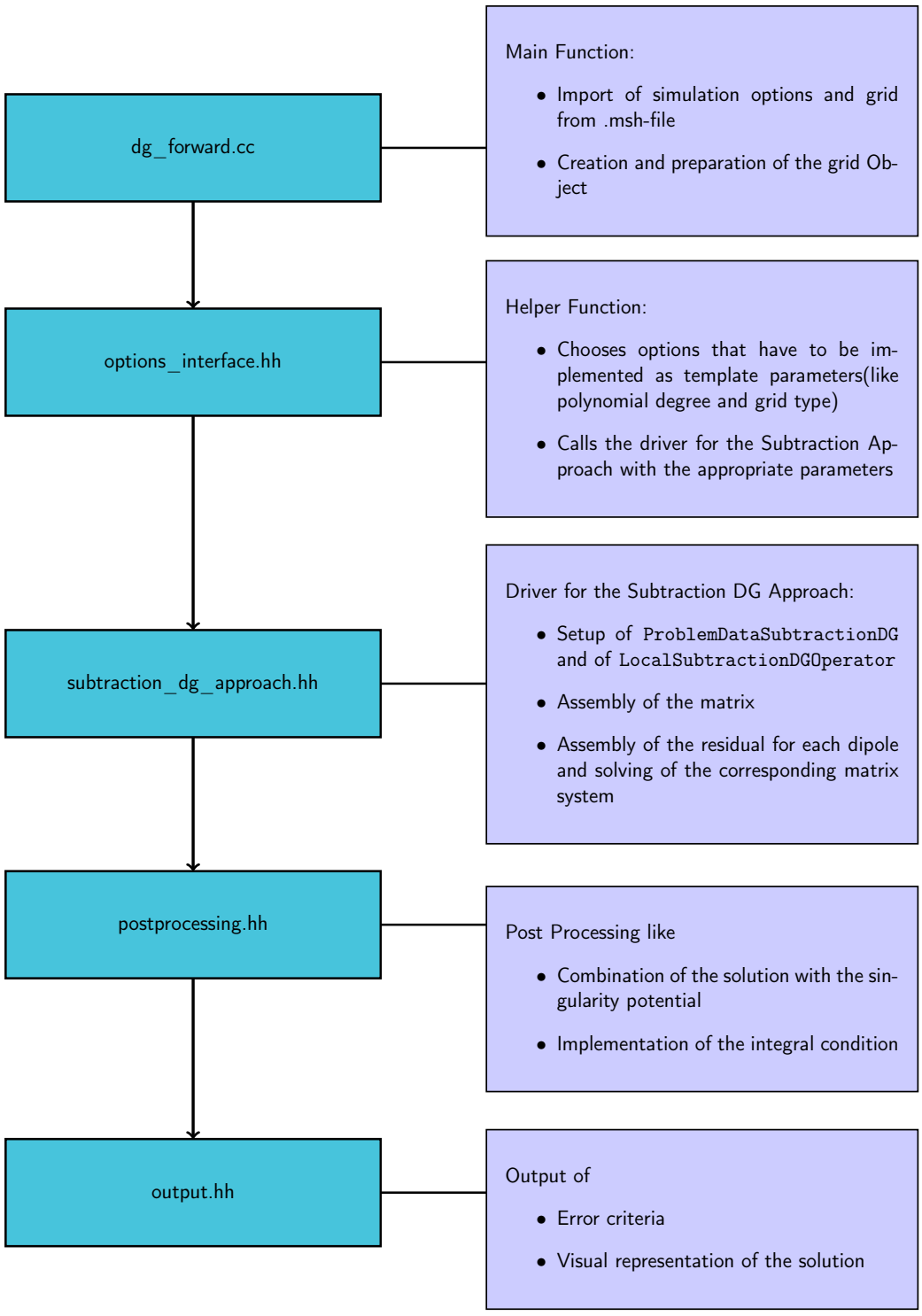

Figure 3.4.: Work flow of the *dg-forward* program.

4. Numerical Simulations

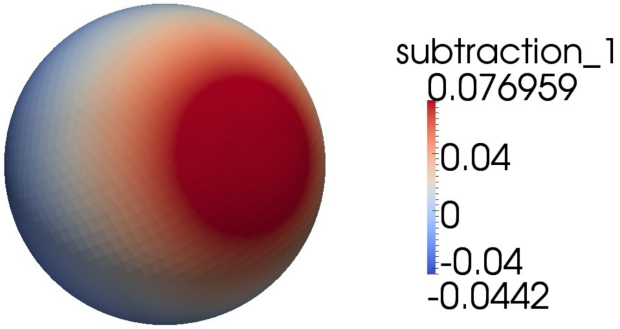

Figure 4.1.: Visualization of the potential obtained from a numerical simulation.

In this chapter we will present numerical results from the implementation described in the previous chapter.

4.1. Simulation Setup

For our studies we used *sphere models*, i. e. approximations of the head by nested spheres with each layer representing a different tissue compartment of the head. While they are obviously a very rough approximation of a real human head, sphere models are still used in source analysis due to the existence of an analytical solution. However, when using Finite Element methods, the goal should certainly be to apply them to real head models generated by MRI scans. Nevertheless the existence of an analytical solution makes sphere models very well suited for the validation of numerical methods.

4.1.1. Analytical Solution for Sphere Models

In [21] de Munck showed that an analytical solution for the potential in a layered spherical volume conductor with compartment wise constant isotropic or anisotropic conductivities can

be derived. Furthermore, in [19] it was shown by de Munck et al. how the analytical solution in such multisphere models can be computed in a rapid fashion using an approximating series:

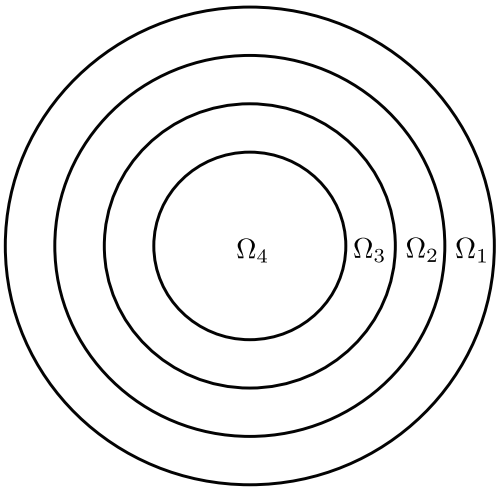

Figure 4.2.: 2D Visualization of a Four Layer Sphere Model for the Head.

Given a multilayer sphere model with S shells (see Figure 4.2), i. e. a collection of S concentric spheres nested within in each other with radii $r_S < r_{S-1} < \ldots < r_1$ and constant conductivities inside each of those shells, the approximation formula for the analytical solution becomes:

$$u_{ana}(x_0, x_e) = \frac{1}{4\pi}\langle \vec{M}, S_0 \frac{x_e}{r_e} + (S_1 - cos\omega_{0_e})\frac{x_0}{r_0}\rangle$$

In the above formula ω_{0_e} denotes the angular distance between the source's position x_0 and the electrode's position x_e. The vector $\vec{M}$ denotes the dipole moment, r_e and r_0 denote the radial coordinate of the electrode and the source respectively. In order for the approximation to be valid it has to be assumed that the source is located in a more interior layer of the sphere than the electrode position.
Regarding the terms

$$\begin{aligned} S_0 =& \frac{F_0}{r_0}\frac{\Lambda}{(1-2\Lambda\cos\omega_{0_e}+\Lambda^2)^{3/2}} \\ &+ \frac{1}{r_0}\sum_{n=1}^{\infty}((2n+1)R_n(r_0,r_e) - F_0\Lambda^n)P_n{}'(\cos\omega_{0_e}) \end{aligned}$$

and

$$S_1 = F_1 \frac{\Lambda \cos \omega_{0_e} - \Lambda^2}{(1 - 2\Lambda \cos \omega_{0_e} + \Lambda^2)^{3/2}} \sum_{n=1}^{\infty} ((2n+1) R_n'(r_0, r_e) - F_1 n \Lambda^n) P_n(\cos \omega_{0_e})$$

it suffices to say that the terms in the series tend to become independent of n. The computation of the series can be stopped when the following criterion is fulfilled:

$$\frac{t_k}{t_0} \leq v, \quad t_k := (2k+1) R_k' - F_1 k \Lambda^k$$

for v chosen to be small enough. For more details on the definitions of the other terms occurring in the above formulas we refer to [15] and the detailed derivation in [19].
This analytical solution was used as a validation reference for the DG methods and the conforming FEM in the numerical studies. The source code for the computation of the analytical solution was adapted from the *SimBio toolbox*[1] and implemented directly into the work flow of the program.

4.1.2. Electrodes

For the evaluation of the numerical solution a set of 222 electrodes were used in all of the following studies. The electrodes were distributed equally on the surface of the outermost sphere and are then mapped onto the closest surface node of the mesh. In the case of hexahedral meshes an additional constraint was used to make sure that the projected position did not lie outside of the maximal radius of the analytical sphere. This was done to ensure the validity of the analytical solution. Figure 4.3 shows a visualization of the electrodes.

4.1.3. Error Criteria

The following error criteria were used for the validation of the method:
The *Relative Error(RE)* is defined by

$$RE(u^{ana}, u^{num}) := \frac{\|u^{num} - u^{ana}\|_2}{\|u^{ana}\|_2} \tag{4.1}$$

where u^{ana} denotes the vector holding the values of the analytical solution at the electrodes and u^{num} the numerical solution obtained by the numerical methods. Accordingly $\|\cdot\|_2$ denotes the l^2 vector norm. In the field of source analysis two additional error criteria are used to

[1] SimBio Development Group homepage: `https://www.mrt.uni-jena.de/simbio`

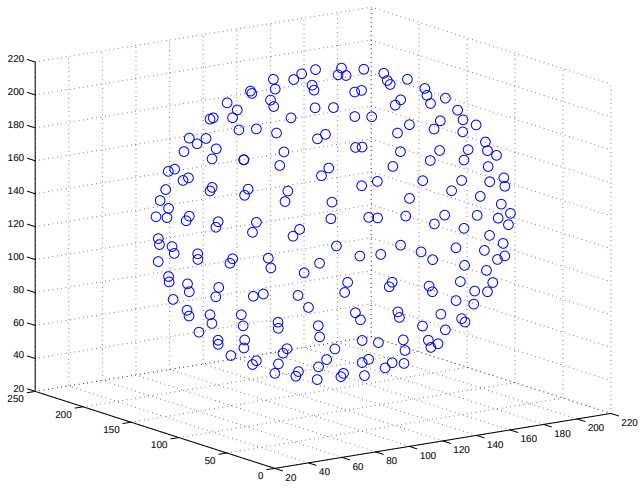

Figure 4.3.: Visualization of the electrodes distributed on the sphere model.

distinguish between the error in magnitude and the error in topography of the solution. The *magnitude error(MAG)* is defined by

$$MAG(u^{ana}, u^{num}) := \frac{\|u^{num}\|_2}{\|u^{ana}\|_2} \tag{4.2}$$

and the *relative distance measure(RDM)*

$$RDM(u^{ana}, u^{num}) := \left\| \frac{u^{num}}{\|u^{num}\|_2} - \frac{u^{ana}}{\|u^{ana}\|_2} \right\|_2 \tag{4.3}$$

measures the topography error. Since it only considers the normalized solutions the RDM does not incorporate the error in magnitude of the solution whereas the MAG only measures the magnitude error. The optimal value for the MAG is 1.

4.1.4. Integration Order

The integration orders for the `alpha_`- and `jacobian_`-methods were automatically chosen by the preimplemented DG operator according to the degree of the basis polynomials involved in the terms. For the newly implemented `lambda_`-methods it was experimentally observed that convergence could only by guaranteed if the integration order for the `lambda_ boundary` method was increased. For the simulations in this thesis the integration order for that method

was set to ten to eliminate any error that may stem from this particular source. On top of that all the precomputed integration orders were increased by two to avoid instabilities introduced by the singularity in our equations.

4.1.5. Choice of Solvers

For solving the matrix system resulting from the DG methods an *Algebraic Multigrid(AMG)* solver was utilized. For the conforming FEM a sequential conjugate gradient solver was used. In both cases Krylov subspace methods were used which preserve the algebraic average of the initial guess for our solution. This makes it unnecessary to ensure the uniqueness of the solution using an additional condition as discussed in Remark 2.4. To obtain a solution at the electrodes which is comparable to the analytical solution both vectors were adapted to have zero mean.
For all simulations the corresponding matrix systems were solved up to a defect of 10^{-8}.

4.2. Study 1: Tetrahedra Models

4.2.1. Model Properties

For this study two tetrahedral four layer sphere models were used. The first model is made up of 190060 elements while the second model has approximately 3.16 million elements. Thus the first model is a much coarser approximation of the multi sphere geometry than the second one. Both models consist of four layers representing the following compartments of the human head (from outside to inside): *Scalp*, *Skull*, *Cerebrospinal Fluid(CSF)* and *Brain*. Isotropic conductivities for each layer were chosen according to literature (see for example [23], [26] or [25]). For more details on the properties of the two models see Table 4.1 and 4.2, also see Figure 4.4 for a graphical comparison.

Model	Number of Elements	Number of Vertices	Number of Layers
tet190k	190060	31627	4
tet3159k	3159575	518730	4

Table 4.1.: Properties of the tetrahedra models

4.2.2. Simulation Parameters

For all of the following simulations dipoles were placed in the innermost compartment which represents the brain in our model. To describe its position within this compartment we will use the *eccentricity*. For a source position x_0 and the center of the model c the eccentricity measures the radial position of the dipole relative to the outer radius r_4 of the brain

Layer	Physiological Tissue	Outer Radius(in mm)	Conductivity(in S/m)
1	Scalp	92.0	0.43
2	Skull	86.0	0.01
3	CSF	80.0	1.79
4	Brain	78.0	0.33

Table 4.2.: Conductivities and radii for the tetrahedra models

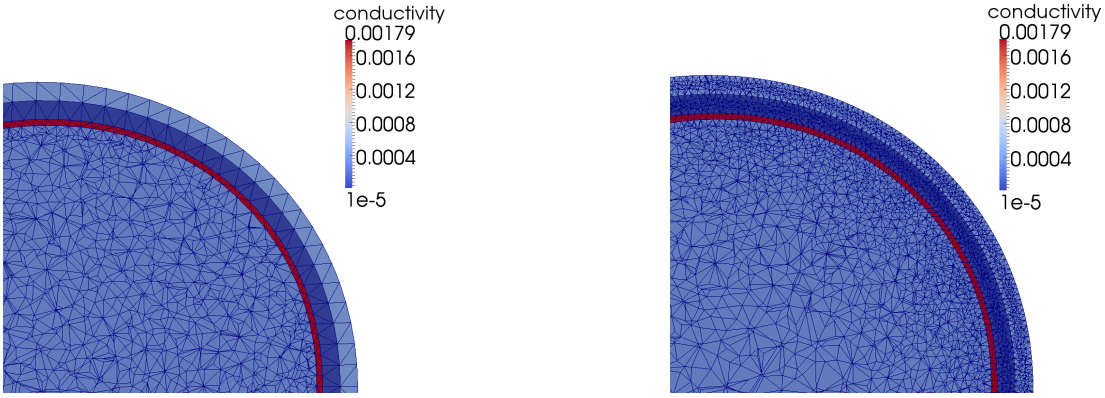

Figure 4.4.: Detail view of the tetrahedral sphere models. Left figure shows the coarse mesh, right figure shows the fine mesh. Visualization done using *ParaView*. Note that the conductivity scale is in S/mm.

compartment:

$$ecc(x_0) = \frac{\|x_0 - c\|_2}{r_4}$$

The eccentricity gives a percentage value describing the proximity of the source to the boundary with the neighbouring compartment and thus to the jump in conductivity. Previous FEM studies (for example [15]) have shown that the error of the numerical solution remains quite stable for lower eccentricities and tends to grow exponentially when the conductivity jump is approached. For the studies in this thesis we chose logarithmic steps towards the conductivity jump in order to put the focus on the area were the growth in error can be expected and away from the region where the error is expected to stay more or less constant.

Eccentricities									
0.0100	0.5022	0.7483	0.8714	0.9329	0.9637	0.9790	0.9867	0.9906	0.9925

Table 4.3.: Eccentricities for the coarse tetrahedra model study

For the study on the coarse tetrahedral model the minimal eccentricity was chosen to be 1% and the maximal eccentricity 99.25% corresponding to a minimal distance of about 0.585 mm to the conductivity jump for the most eccentric sources. See Table 4.3 for an overview of the eccentricities. For each eccentricity 10 dipole positions were randomly generated with radial orientation and another set of 10 dipoles were generated with tangential moment. For the total of 200 sources conforming FEM simulations and DG simulations with polynomial degrees of one, two and three were performed.

For the finer tetrahedra model dipoles were placed on a beam, starting close to the center of the model with an eccentricity of 0.22%, approaching the inner shell's radius up to an eccentricity of 99.5% which corresponds to a proximity of 0.39 mm. Radial dipoles were distributed along the x-axis and their moments were aligned along the same axis. Tangential dipoles were placed along the y-axis and oriented in z-direction. Again logarithmic steps were used while approaching the maximal eccentricity and 12 dipoles were placed for both types of orientation in order to keep the computational time manageable. Polynomial degrees of one and two were used.

Eccentricities					
0.002	0.498	0.747	0.871	0.933	0.964
0.980	0.987	0.992	0.993	0.994	0.995

Table 4.4.: Eccentricities for the fine tetrahedra model Study

4.2.3. Results for the Coarse Tetrahedral Mesh

For the coarse tetrahedra model an increase in accuracy can be observed for all error measures. Figures 4.5 to 4.7 show the mean errors per eccentricity obtained by the DG simulations and the conforming FEM. The conforming Finite Element method will be used as a reference for the evaluation of the DG methods. The detail plots on the right hand side of the figures show the averaged errors for the last five eccentricities on a logarithmic scale y-scale (except for the MAG plots where linear scales were used).

The two plots in the upper row of Figure 4.5 show that the RDM of the conforming FEM can be reduced by a factor of four when using DG with polynomial order three for radial sources with maximal eccentricity. While in both cases the absolute values of 0.4136 and 0.105 respectively are probably still too high to be considered acceptable, the eccentricity

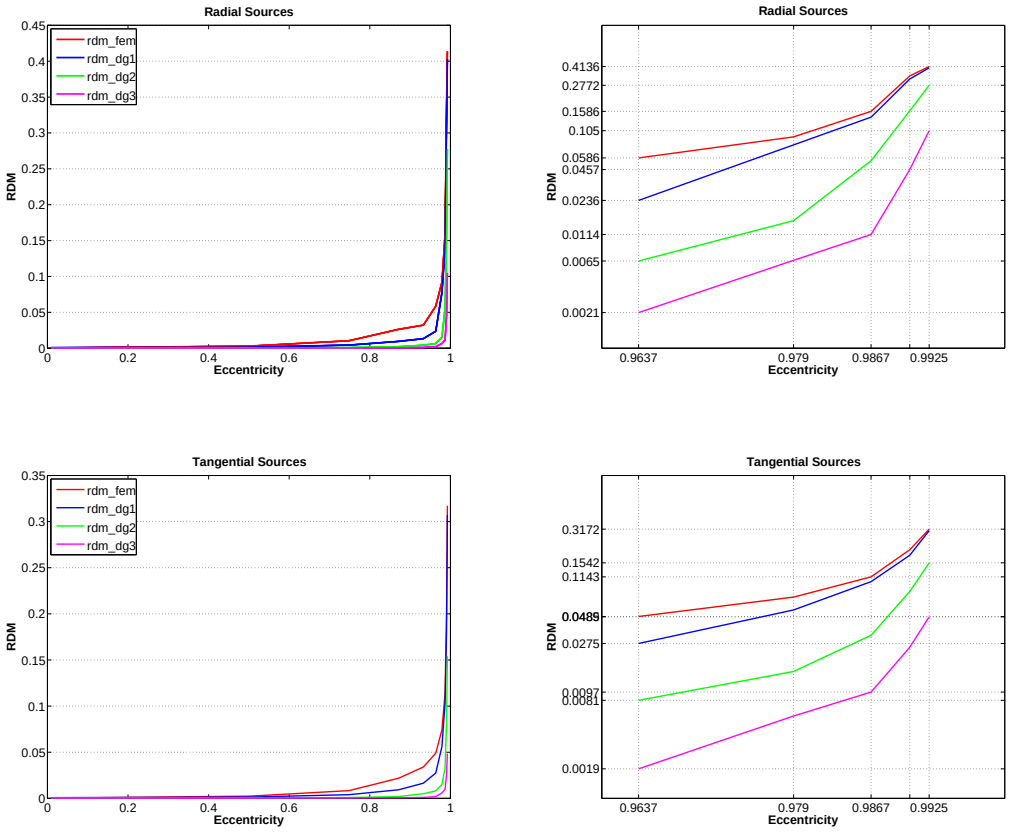

Figure 4.5.: Plot of the mean RDM error over the eccentricities for the radial sources (top) and the tangential sources (bottom) on the coarse tetrahedra model. The right hand side figures show the last five eccentricities on a logarithmical y-scale.

threshold for which solutions with RDM of less than 0.05 can be expected is considerably lower for the DG method of polynomial order three. In this case eccentricities of up to 99.1% could be chosen whereas the conforming Finite Element method did not obtain RDM values lower than 0.05 for any of the last five eccentricities. The DG method with polynomial degree one performed slightly better, yielding errors below five percent for eccentricities lower than 0.9637.

In the case of tangential sources (bottom plots) the RDM values are significantly lower than for radial sources. The error at the maximal eccentricity is already at 0.0485 for polynomial degree three and also the eccentricity threshold for sources with acceptable errors is higher for all methods. Apart from the absolute error being lower the error still seems to grow

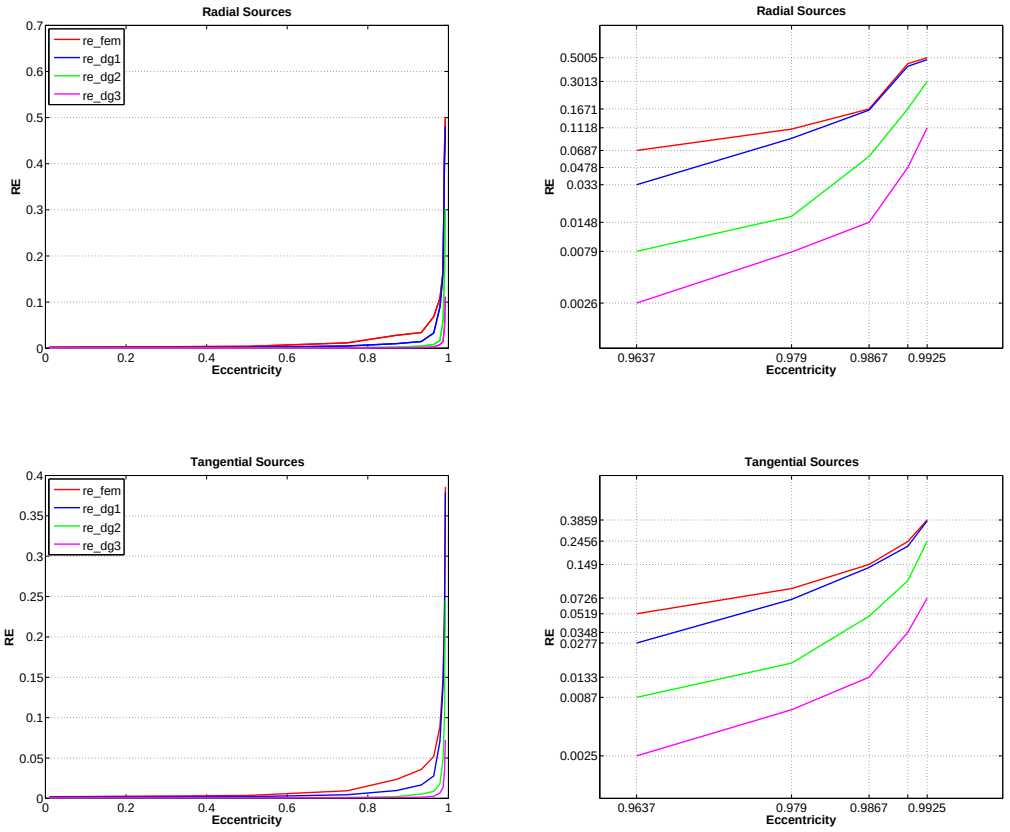

Figure 4.6.: Plot of the mean relative error over the eccentricities for the radial sources (top) and the tangential sources (bottom) on the coarse tetrahedra model. The right hand side figures show the last five eccentricities on a logarithmical y-scale.

exponentially as observed in the case of radial sources and shown in previous studies using conforming FEM.
It also should be noted that the for the eccentricity of 0.9867 which corresponds to sources roughly 1 mm up to the conductivity jump the errors could be greatly reduced. For radial sources the FEM solution yields an error of 0.1586 while the third order DG approximation produces errors of about 0.0114. In the case of the tangential sources the picture looks very alike where the FEM error could be improved from 0.1143 to just under 0.01 using the DG scheme with polynomial degree three.

Due to the close relationship between the relative error and the RDM, the plots for the relative

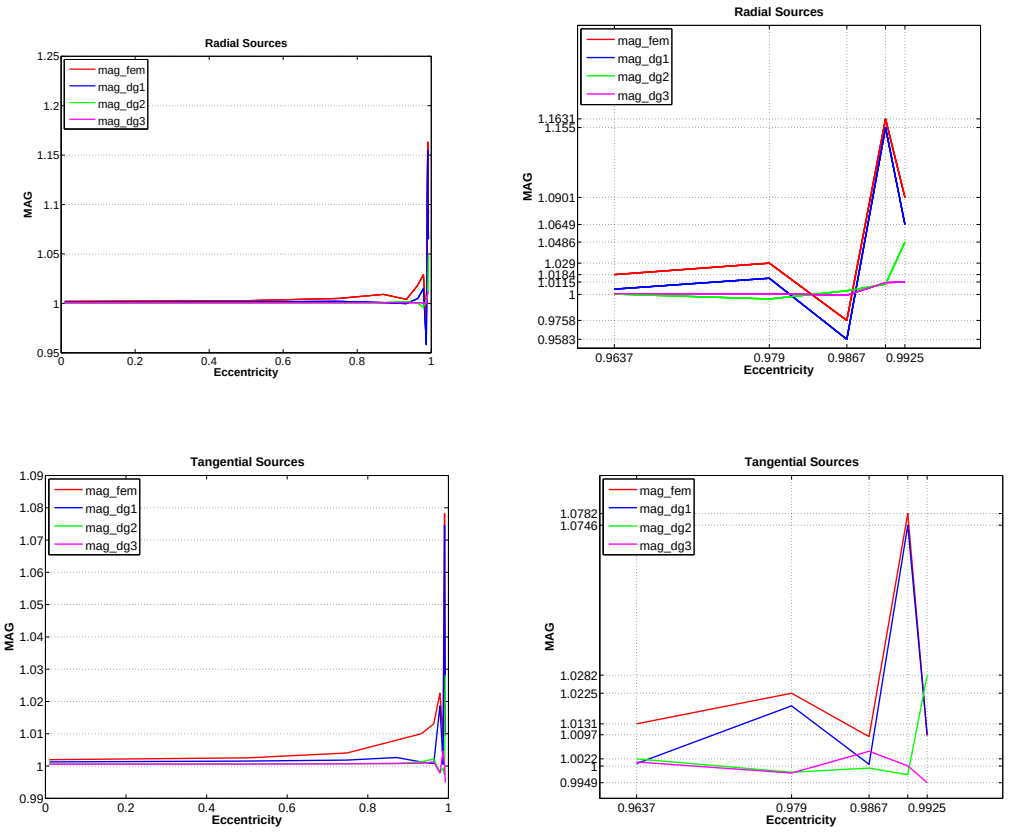

Figure 4.7.: Plot of the mean MAG error over the eccentricities for the radial sources (top) and the tangential sources (bottom) on the coarse tetrahedra model. The right hand side figures show the last five eccentricities.

error in Figure 4.6 exhibit very similar behaviour when compared to the RDM. As previously discussed, unlike the RDM, the relative error also incorporates the error in magnitude of the solution.

The magnitude error (see Figure 4.7) shows some oscillations at higher eccentricities for both the radial and the tangential sources where again tangential sources overall produce the better results. Despite the fact that the oscillations do not allow for a clear growth pattern to emerge, it can be stated that for the available sample the DG methods on average can improve the results of the conforming Finite Element solution. Nevertheless it may be worthwhile to re-examine this phenomenon using a bigger sample size than ten dipoles per eccentricity.

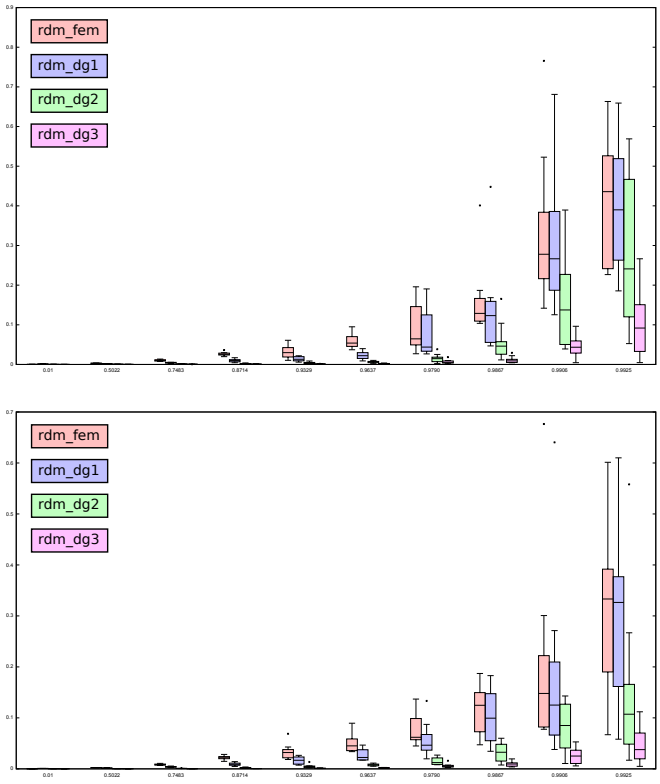

Figure 4.8.: Boxplot of the RDM error for the radial (top) and tangential (bottom) sources. Outliers are marked by crosses. For better readability the eccentricities on the x axis are not drawn to scale.

For a simulation setup as used in this study, where multiple sources are used for each eccentricity, the evaluation of the method can not be based solely on averaged errors since they do not visualize the distribution of the errors on a per source basis. In particular for the MAG on higher eccentricities we could see that the error does not seem to follow a steady growth pattern but rather oscillates. This may hint at the fact that the errors do not fall into a close range but are distributed with quite high variance. This is of course an undesirable property since we are looking for a method that gives us low errors with little variance. In order to assess this quality of the methods we use boxplots as presented in Figure 4.8 for the RDM and in Figure 4.9 for the MAG. The boxes represent the area in which the middle 50% of the values can be found. The horizontal bar within each box represents the median. The whiskers are limited to 1.5 times the interquartile range. Outliers, i. e. values that lie outside the whiskers, are marked by crosses. Additional boxplots can be found in Appendix C.

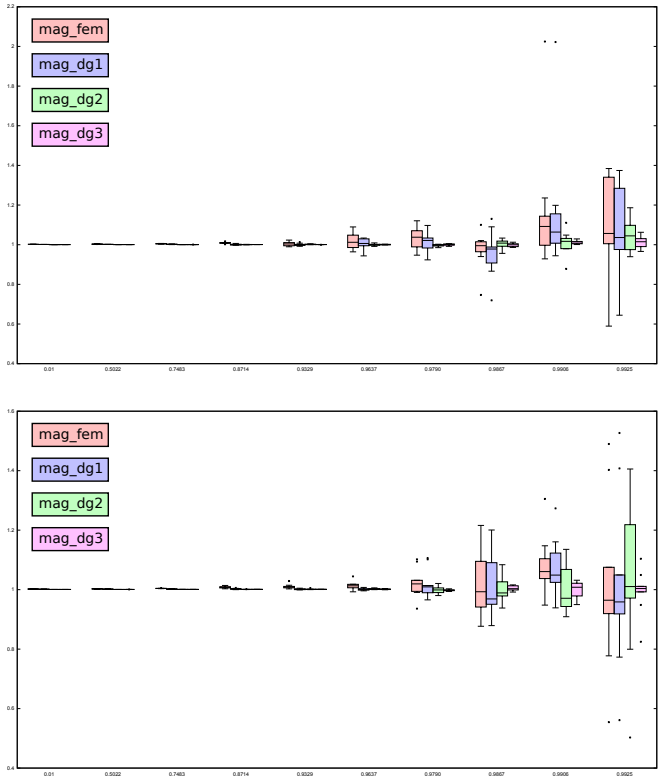

Figure 4.9.: Boxplot of the MAG error for the radial (top) and tangential (bottom) sources. Outliers are marked by crosses. For better readability the eccentricities on the x axis are not drawn to scale.

From the boxplots it can be seen that for both radial and tangential source orientations the DG method of degree three produces the best result, i. e. the smallest error variance in a low range. Also it produces very few outliers, all of which still assume tolerable absolute values. The error range for the DG method of degree two always lies under that of the FEM and the DG method of degree one but seems to produce a larger scattering than the former two in some cases. With respect to variance of the error it is hard to tell whether the DG method with linear basis functions or the conforming FEM does a better job for this particular sample. For the MAG boxplots as shown in Figure 4.9 it becomes obvious again that the DG method of degree three performs best, although it has to be noted that, especially for the most eccentric tangential sources, a lot of outliers are produced. Also the variance for the second order DG method in the tangential case seems very high which might call for a bigger sample size.

4.2.4. Results for the Fine Tetrahedral Mesh

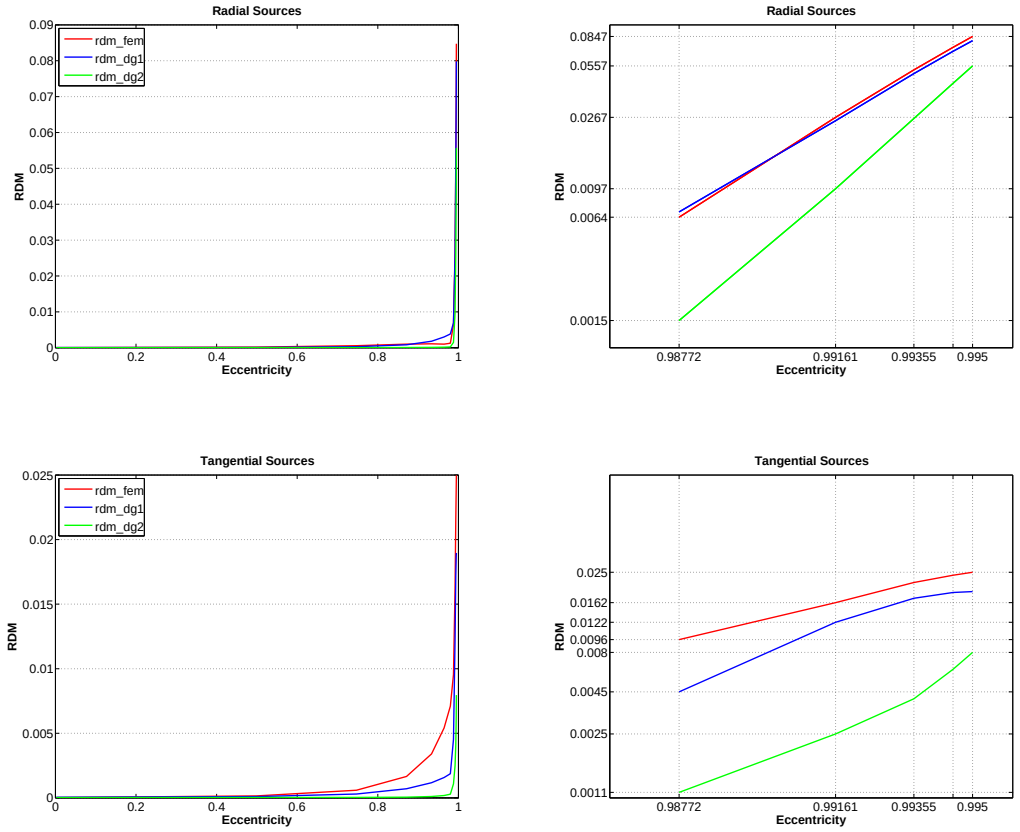

Figure 4.10.: Plot of the RDM over the eccentricities for the radial sources (top) and the tangential sources (bottom) for the fine tetrahedra model. The right hand side figures show the last five eccentricities on a logarithmical y-scale.

On the fine tetrahedral mesh, similarly to the previous results on the coarser tetrahedral mesh, an improvement in all error measures could be observed for the DG methods over the conforming FEM solutions. Figures 4.10 through 4.12 show plots of the errors over the eccentricities. Conforming FEM and DG with polynomial degree one again perform quite similar to each other while the DG method almost always has slightly better errors, the exception being some radial sources with lower eccentricity. Due to the much finer resolution of the mesh the RDM for all the methods never exceeds 10%. Nevertheless the DG methods can offer improvement, especially in the case of second order basis functions. The tangential sources offer much better results, yielding maximal RDM values of below 1% for the second

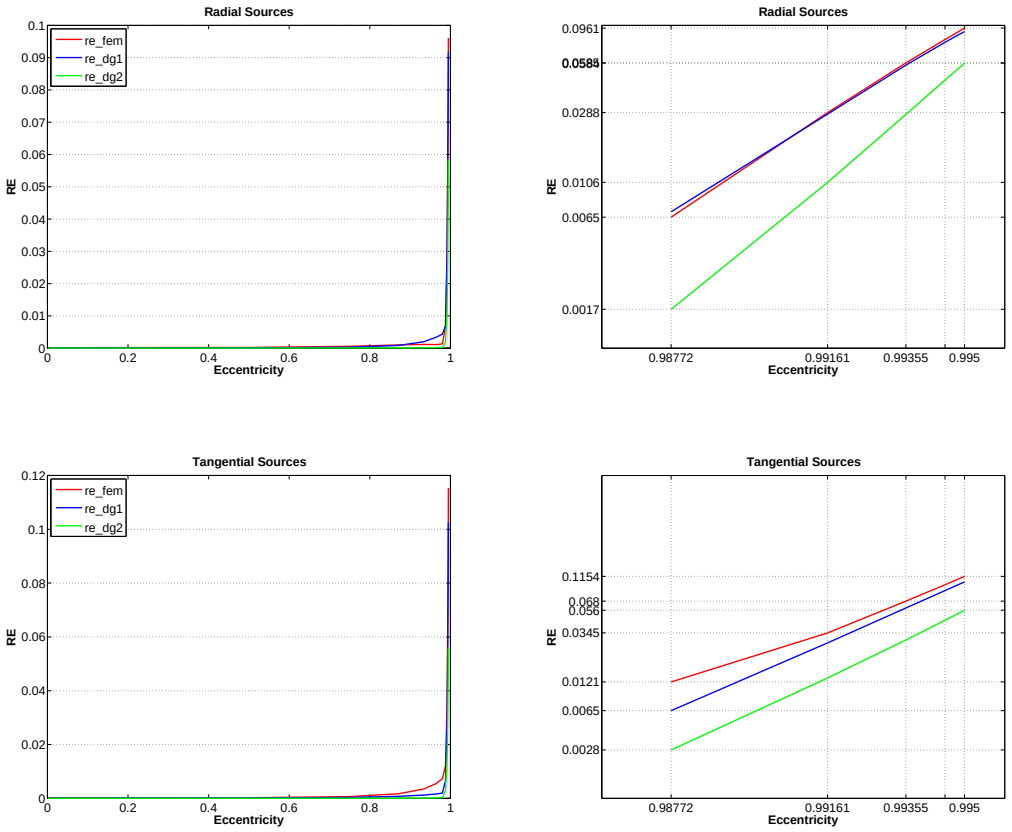

Figure 4.11.: Plot of the relative error over the eccentricities for the radial sources (top) and the tangential sources (bottom) for the fine tetrahedra model. The right hand side figures show the last five eccentricities on a logarithmical y-scale.

order DG method. The error still seems to grow exponentially although the semilogarithmic plot for the tangential sources indicates a slowdown for DG of order one and the conforming FEM method.

The relative error as shown in Figure 4.11 shows a behaviour similar to that of the RDM, again owing to their close relationship.

With respect to the MAG the fine tetrahedra study shows a much clearer picture when compared to the coarse tetrahedra results. This is most likely a result of the choice of sources along a straight line instead of randomly choosing them for each eccentricity but might also

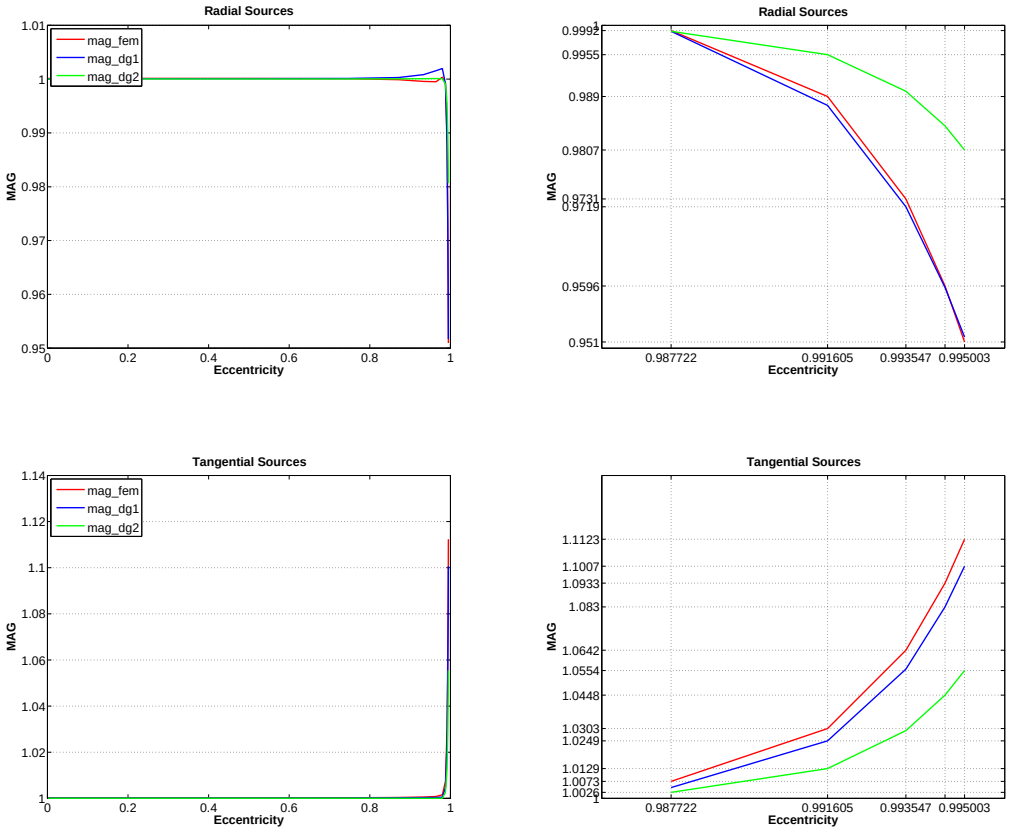

Figure 4.12.: Plot of the MAG error over the eccentricities for the radial sources (top) and the tangential sources (bottom) for the fine tetrahedra model. The right hand side figures show the last five eccentricities.

result from the better approximation properties of the model. The DG method of order two only deviates about 1.93% from the optimal MAG value of 1 for radial sources and about 5.54% for the tangential sources, while the FEM method shows a divergence of approximately 4.9% for radial and 11.23% for tangential sources. The DG method of order one is slightly better than the FEM method in both cases.

As computational cost had to be limited third order basis functions were not evaluated. Furthermore this rather small sample with dipoles placed along a straight line through the model can only give a first estimate of the errors that can be expected on this type of mesh. Another evaluation with randomized sources, possibly using parallelized computations, might

be necessary to obtain more reliable results.

4.3. Study 2: Hexahedra Models

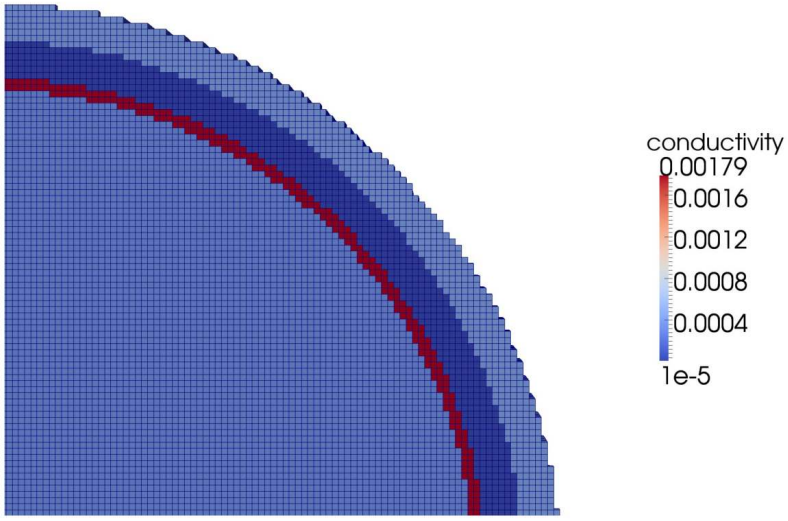

Figure 4.13.: Detail view of the hexahedral sphere model. Visualization done using *ParaView*. Note that the conductivity scale is in S/mm.

4.3.1. Model Properties

Model	Number of Elements	Number of Vertices	Number of Layers
hex3262k	3262312	3342701	4

Table 4.5.: Properties of the hexahedra Model

For this study a mesh with regular hexahedral elements consisting of four layers was used. As can be seen in Figure 4.13 the surface of the hexahedra model does not approximate the surfaces of the nested shells nearly as well as the tetrahedral models do. Nevertheless hexahedral meshes hold a very big relevance in practical applications since the generation of high quality tetrahedra meshes is a very involved and time consuming task. The edge lengths of the hexahedra are chosen to be 1 mm corresponding to the voxel size which is obtained from high-resolution MRI measurements.
Partly due to our choice of basis functions the computational effort for hexahedral meshes of

this size is enormous, which is why we will limit ourselves to a smaller amount of dipoles and will not increase the polynomial degree of the basis functions. Since the models used for this study were not geometry-adapted it is questionable whether an increase in polynomial order would have produced better results or whether they would be limited by the bad geometrical approximation. The properties of the mesh can be seen in Table 4.5. The conductivities chosen for this simulation were the same as for the tetrahedra study in Table 4.2.

Remark 4.1. *Although computational cost is a big factor when using DG methods it should be noted that it is not a general obstacle for using fine hexahedral meshes. Parallelization on a distributed memory architecture could both handle the huge memory demand and decrease the solving and assembly time for the matrix system. This thesis, however, will be limited to sequential computations.*

4.3.2. Simulation Parameters

Eccentricities									
0.0124	0.5030	0.7500	0.8735	0.9353	0.9661	0.9816	0.9893	0.9931	0.9951

Table 4.6.: Eccentricities for the hexahedra model study

For this study ten dipoles were distributed along a beam close to the center along the x-axis using the eccentricities shown in Table 4.6. Radial dipoles were given the unit vector in x direction as moments and tangential dipoles were oriented along the z axis, both using the same positions.

4.3.3. Results for the Hexahedral Mesh

The results for the hexahedral mesh are quite similar to those on the tetrahedral meshes although some unexpected behaviour especially for the MAG was observed.

The RDM for the highest eccentricities could be brought down using the DG method of polynomial degree one, see Figure 4.14. In the case of radial sources the value for the FEM result could be reduced from 0.0905 to 0.0383 for the outermost source. For all of the five sources with highest eccentricity an improvement could be observed although the behaviour for the DG method of the third but last source seems somewhat contradictory to formerly produced results, not exhibiting the usual exponential growth. For the lower eccentricities the FEM seems to produce the better results but it should be mentioned that the absolute error values in this case are still acceptable and in practice sources with higher eccentricity are more relevant. The tangential sources show comparable results for the RDM although in this case only for the last source position an improvement could be made.

The relative error in Figure 4.15 again shows a slight improvement for the outermost source in

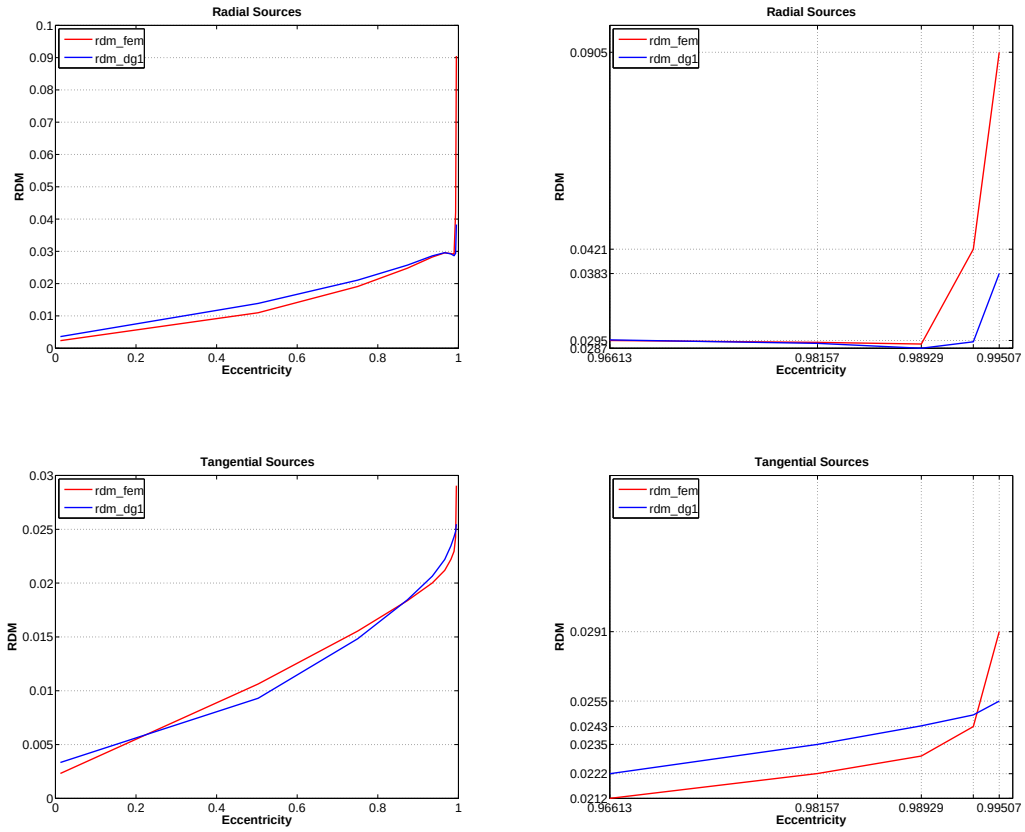

Figure 4.14.: Plot of the RDM over the eccentricities for the radial sources (top) and the tangential sources (bottom) for the hexahedral model. The right hand side figures show the last five eccentricities.

the radial case where the FEM error could be improved from 0.1039 to 0.0843 using the DG method. For the tangential sources the overall behaviour again is not quite as expected from previous simulations but it shows a clearer advantage of the DG method over the conforming FEM. Except for sources of relatively low eccentricity the DG method produces better results and can bring down the error by approximately 26% for the outermost source.

As already mentioned the MAG showed some surprising behaviour as can be seen in Figure 4.16. First of all it should be noted that the DG method does not improve the magnitude error in the case of radial sources but shows worse results than for the FEM simulations. The MAG for the last source position lies around 1.074 for the DG method and 1.047 for the FEM.

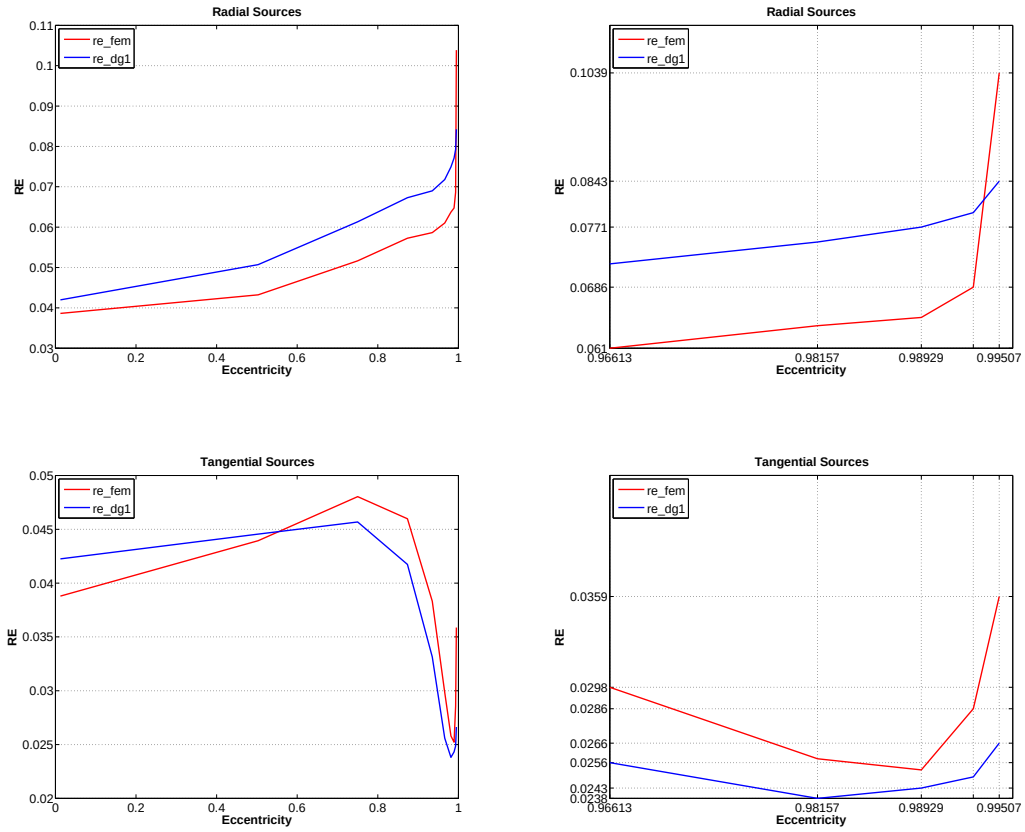

Figure 4.15.: Plot of the relative error over the eccentricities for the radial sources (top) and the tangential sources (bottom) for the hexahedral model. The right hand side figures show the last five eccentricities.

What is more the FEM results show a drop in error for the last two eccentricities which clearly should not be expected. For the case of tangential sources the DG method could once again improve the results of the conforming Finite Element method for the last six eccentricities.

It should be kept in mind that the evaluation for hexahedral meshes in this thesis can only give a first glimpse at the behaviour of DG methods on this type of meshes. In particular regarding the somewhat strange error plots for the MAG it seems necessary to perform further simulations, possibly using parallelized computations. Furthermore the DG methods seem to suffer from the very rough approximation of the spheres' surfaces brought on by the use of hexahedral elements. Simulations using geometry adapted-meshes should help to clear things

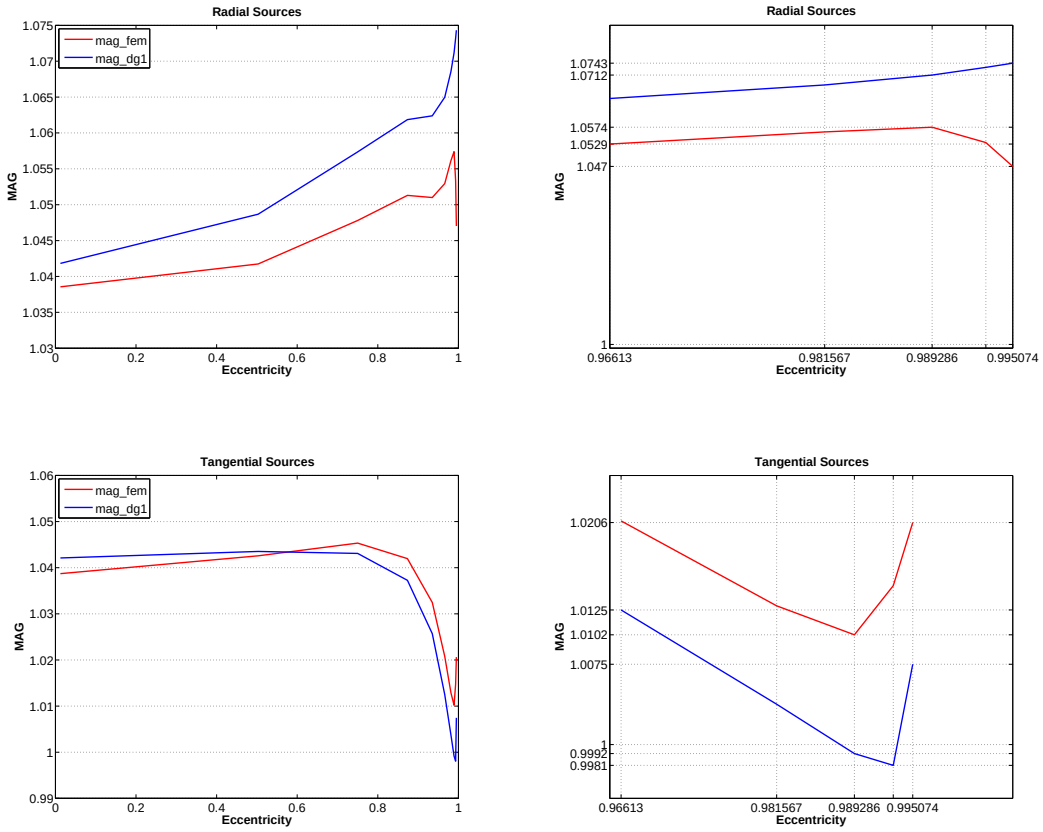

Figure 4.16.: Plot of the MAG error over the eccentricities for the radial sources (top) and the tangential sources (bottom) for the hexahedral model. The right hand side figures show the last five eccentricities.

up and might make the use of higher order basis functions more rewarding. We will come back to this issue in the conclusion in Chapter 5.

4.4. Study 3: Real Head Model

In this section we will present visualizations of results obtained by simulations performed on a realistic approximation of a human head.
Since there is no analytic solution available for this type of simulation, we are not able to conduct an error analysis employing the error measures used in the previous sections. Although it would be possible to compare the Discontinuous Galerkin solution to a conforming FEM result on a heavily refined mesh we will not do so because the effort needed to compute the refined FEM solution would probably be disproportionate. The results from the previous sections should suffice to back the hypothesis that Discontinuous Galerkin methods can produce results superior to that of the conforming Finite Element method. Therefore we will limit ourselves to present visualizations of the simulation results. All the visualizations in this section were performed using the *SCIRun*[2] environment.

Model	Number of Elements	Number of Vertices	Number of Compartments
Real Head Model	6107561	984569	10

Table 4.7.: Properties of the real head model.

Physiological Tissue	Conductivity(in S/m)
Skin	0.43
Eye	0.43
Skull Compacta	0.008
Skull Spongiosa	0.025
Brain Stem	0.14
Cerebellum Gray Matter	0.33
Cerebellum White Matter	0.14
Gray Matter	0.33
White Matter	0.14
CSF	1.79

Table 4.8.: Physiological tissues of the real head model and the corresponding isotropic conductivities.

[2] A Problem Solving Environment(PSE) for modelling, simulation and visualization of scientific problems. See: `http://www.sci.utah.edu/cibc-software/scirun.html`

Source Configuration			
Position	154.405	114.846	178.102
Moment	-0.3365	0.9126	0.2323

Table 4.9.: Source position and moment for the real head model simulation.

The head model used in this study was obtained by MRI measurements from a healthy 25 year old male subject. The voxel size for the MRI scans was 1 mm and constrained Delaunay tetrahedralization(CDT) was applied to the image data resulting in a tetrahedral mesh with the properties presented in Table 4.7. Ten different compartments were distinguished and assigned the conductivities shown in Table 4.8. In the following visual results of the simulation will be presented.

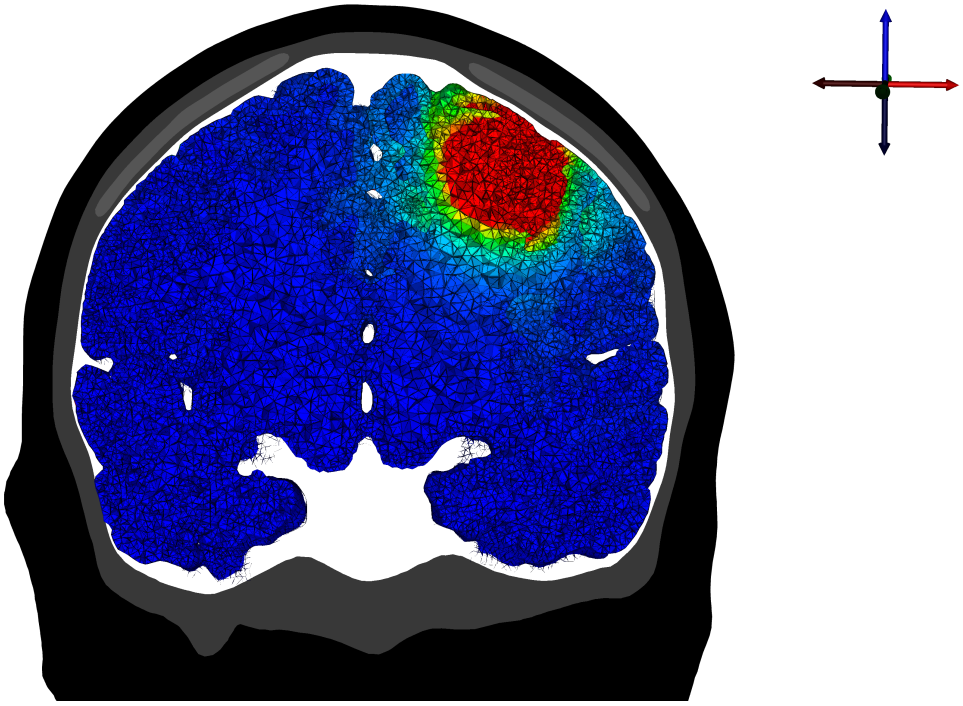

Figure 4.17.: Visualization of the norm of the current density $\sigma \nabla u$ using a coronal slice at position y=114. Figure provided by Sven Wagner.

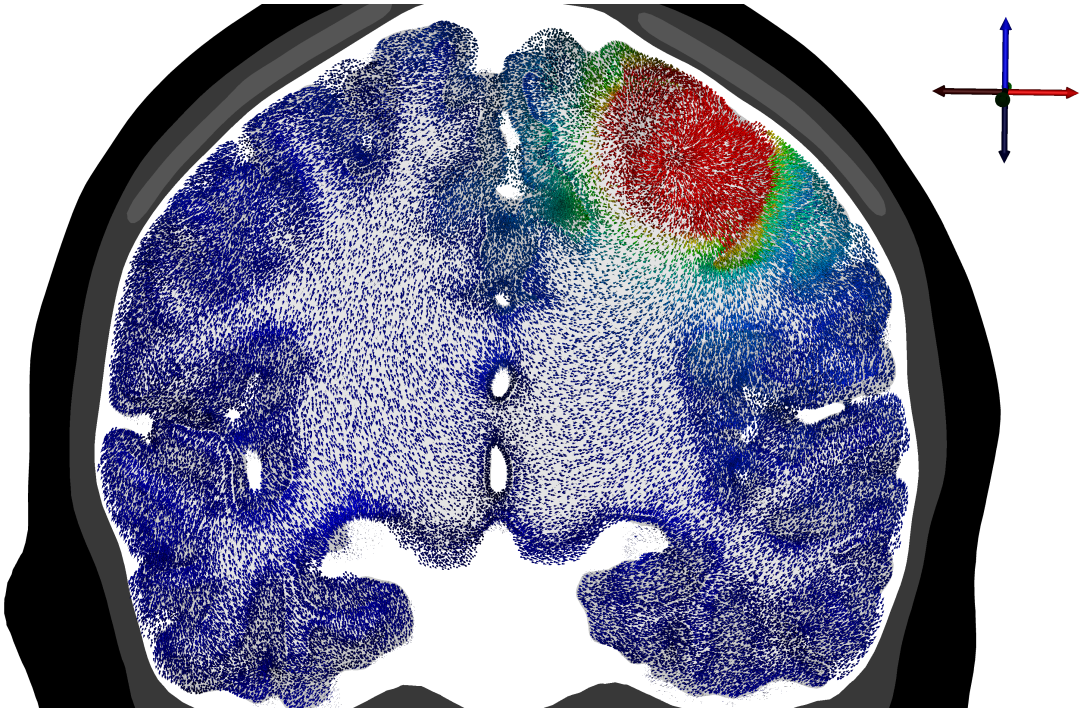

Figure 4.18.: Visualization of the current density $\sigma \nabla u$ using a coronal slice at position y=114. Figure provided by Sven Wagner.

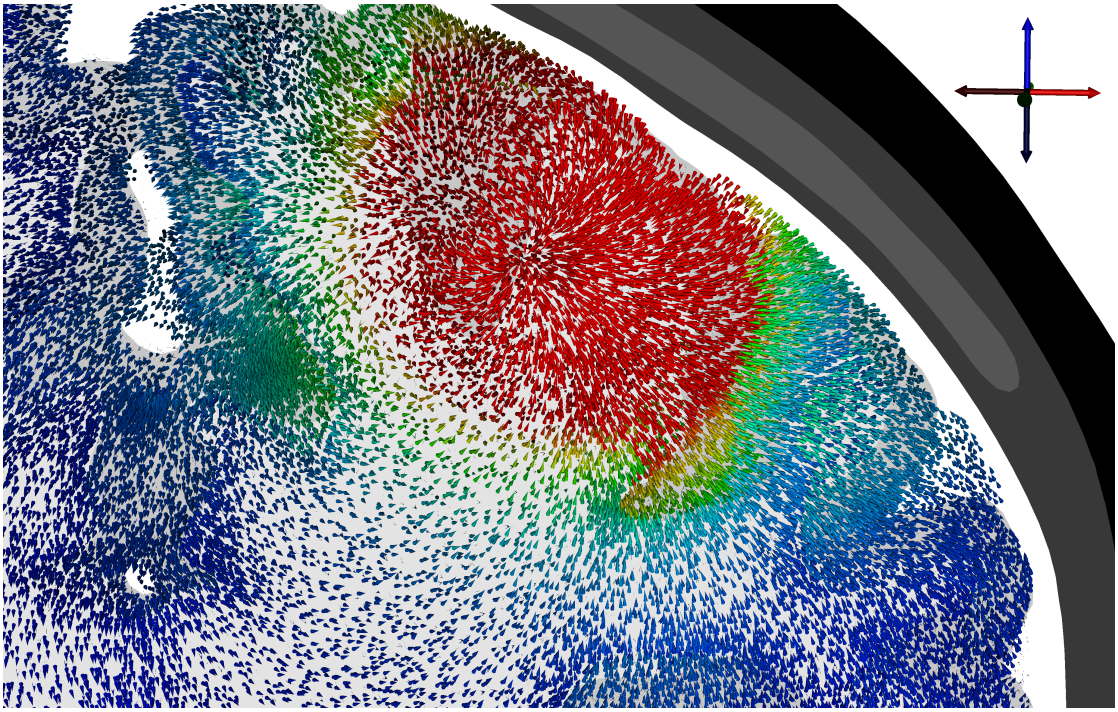

Figure 4.19.: Visualization of the current density $\sigma \nabla u$ using a coronal slice at position y=114 zoomed in around the source position. Figure provided by Sven Wagner.

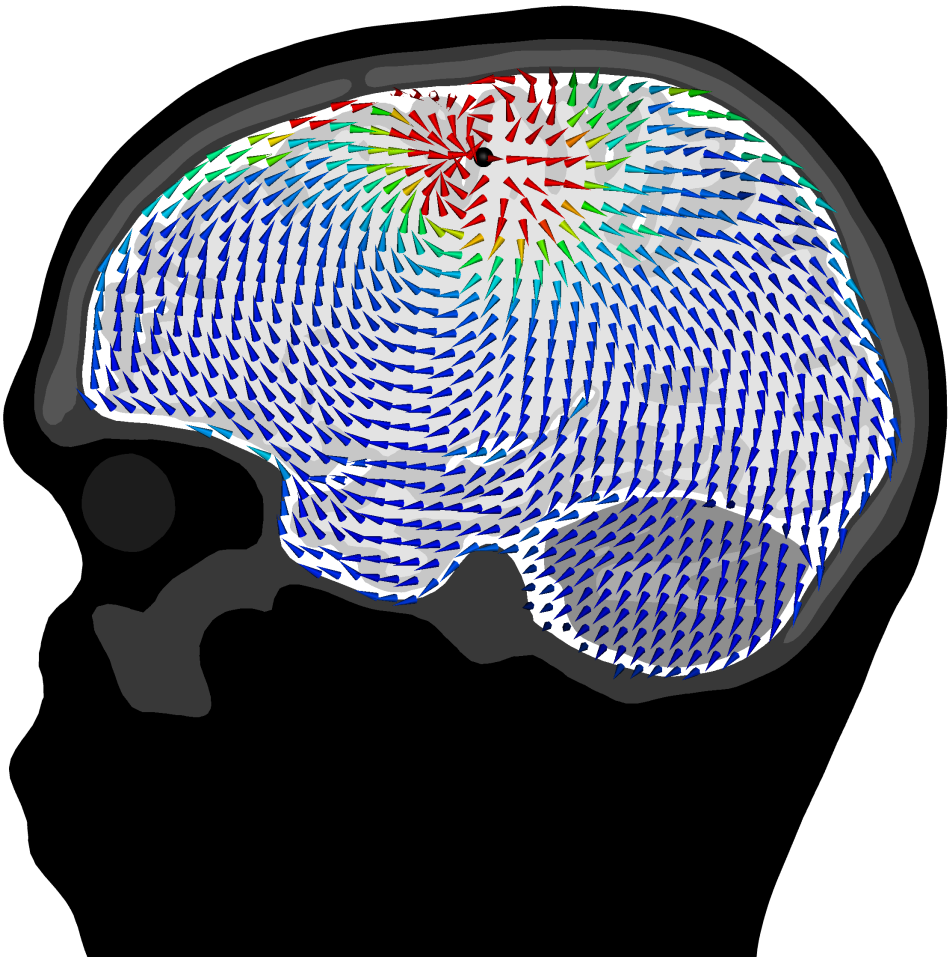

Figure 4.20.: Mapping of the current density $\sigma \nabla u$ on a regular 4 mm cube grid. Saggital slice through x=154. Figure provided by Sven Wagner.

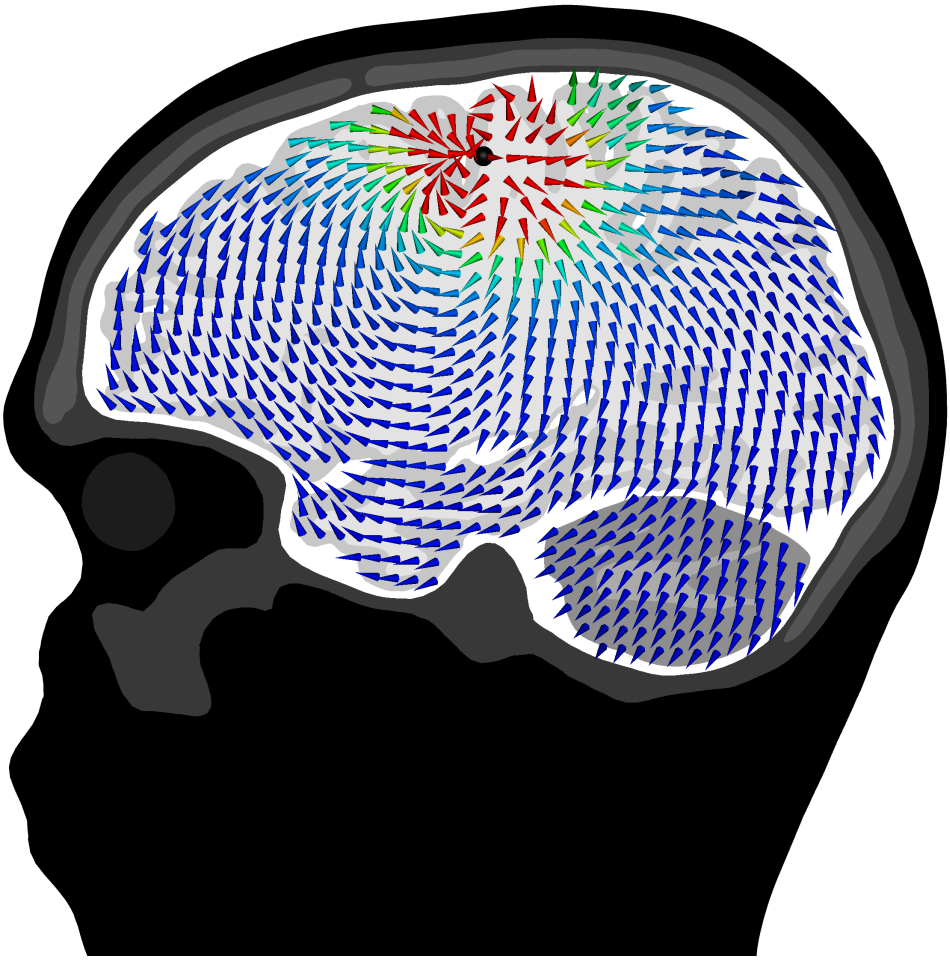

Figure 4.21.: Mapping of the current density $\sigma \nabla u$ on a regular 4 mm cube grid, excluding the CSF compartment. Saggital slice through x=154. Figure provided by Sven Wagner.

4.5. Study 4: Performance Analysis

The previous studies on tetrahedral and hexahedral multi sphere models have shown that Discontinuous Galerkin methods can indeed improve upon the results obtained by FEM simulations. As described before the gain in accuracy obtained by DG methods stems in great part from the introduction of local basis functions which, especially for the case of basis orders larger than one, results in a much higher number of degrees of freedoms and thus a much bigger matrix system. Although the storage of the matrix can benefit from the matrix structure of the DG method by using blockwise storage patterns (see Section 2.6.1), the higher number of degrees of freedom still results in a much larger memory consumption and higher solving time.

In this section we will finish our analysis of Discontinuous Galerkin methods by presenting a few figures concerning the memory consumption and solving time which is to be expected for the different methods on the various meshes. The objective of this section is not to give a detailed analysis of this matter but to provide an estimate of the hardware requirements for the use of DG methods in source analysis applications.

All the simulations in the previous section and measurements in this section were carried out on a machine running the Linux distribution Ubuntu 12.04 with 96685.63 Megabytes of RAM and an Intel Xeon X5660 CPU running at 2.8 Ghz.

Solving Time

In order to solve the matrix systems resulting from the DG discretizations a highly optimized algebraic multigrid solver was used in the previously presented numerical simulations, see [34] for more details. For the comparisons in this section, additional simulations were carried out using a conjugate gradient solver with incomplete LU decomposition as a preconditioner. This way both the DG and the FEM methods could be evaluated using the same type of solver and a more accurate estimate of the total effort needed can be given. We will however also plot the solving time for the AMG since it can give us a more realistic idea of solving times that can be expected. The solving times presented in this section are the averaged solving times obtained from simulations for five dipoles with different eccentricities and orientations. For each source the system was solved up to a total defect of 10^{-8}.

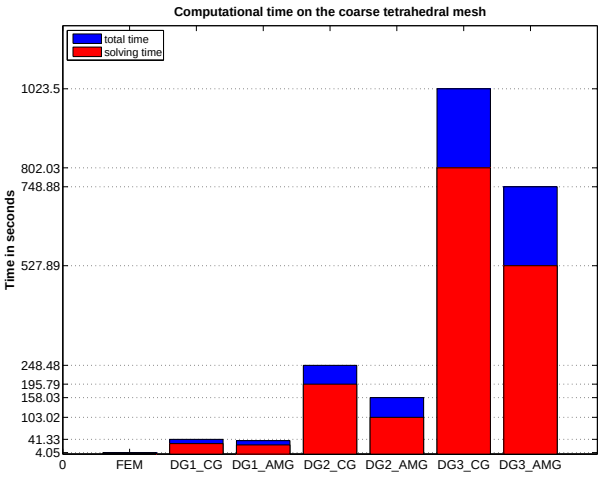

Figure 4.22.: Bar graph of the computational time per dipole on the coarse tetrahedral mesh.

The computational times on the coarse tetrahedral mesh have been visualized in Figure 4.22 using bar graphs. The total time in this figure includes the assembly time for the residual, the solving of the system and all of the post processing like the calculation and output of the error criteria. The FEM computations have an average computation time of roughly 4 seconds where the solving time takes up only 0.3376 seconds making it very hard to improve upon in terms of speed. Obviously that is an unrealistic endeavour since we are dealing with a much larger number of degrees of freedom when using DG methods. In the case of the conforming FEM there is one DOF for each vertex in the grid while the number of DOFs in the DG case can be calculated as #(local basis functions) $\times$ #elements. Using the values for the number of basis functions for the $\mathbb{P}_k^3$ basis in Table 2.1 and the geometric properties of the coarse tetrahedral mesh as shown in Table 4.1 the number of DOFs turn out to be as shown in Table 4.10.

	No. of DOFs			
	tet190k	tet3159k	hex3262k	realhead
FEM	31.627	518.730	3.342.701	984.569
DG1	760.240	12.638.300	26.098.496	24.430.244
DG2	1.900.600	31.595.750	88.082.424	61.075.610
DG3	3.801.200	63.191.500	208.787.968	122.151.220

Table 4.10.: Number of DOFs for each method on the tetrahedral meshes, the hexahedral meshes and the real head model.

Keeping these numbers in mind it is obvious that a comparison between computational times for DG methods and the conforming Finite Element method is not very helpful. Also the increase in accuracy observed for the DG methods which comes with a much larger number of DOFs can not be expected to come without any drawbacks with regard to computational cost.
The absolute values for the computational time on the coarse tetrahedral mesh still seem to be practical. Especially when considering the fact that for the calculation of the inverse problem transfer matrices will be used which only require the computation of #electrodes right hand sides. What is more the AMG solver can greatly improve the performance, especially for the case of higher degree basis functions. In the case of polynomial order three the solving time can be reduced by 34.2% and for the second order approximation by about 47.4%. Parallelization should further speed up not only the solving time but also the remaining portion of the total time as the assembly of the residual would also be done in parallel.

For the fine tetrahedral mesh the computational time turned out to be as shown in Figure 4.23.

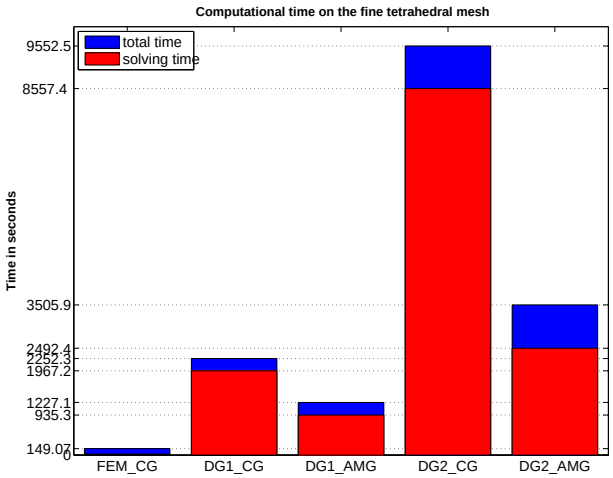

Figure 4.23.: Bar graph of the computational time per dipole on the fine tetrahedral mesh.

Once more the comparison between FEM and the DG methods does not make much sense with regard to the computational time. From the figure it becomes obvious that the solving time takes up a big portion of the total computational time. With the CG solver taking about 8557.4 seconds to solve one right hand side for the DG method of polynomial order two it is certainly questionable whether this method is applicable for everyday applications using

this specific solver. Nevertheless the AMG solver shows potential to make this method more handleable with respect to computational cost bringing down the solving time by roughly 71%. Again parallelization may increase this gain in speed even more.

The values obtained for the hexahedral mesh are very similar as can be seen in Figure 4.24.

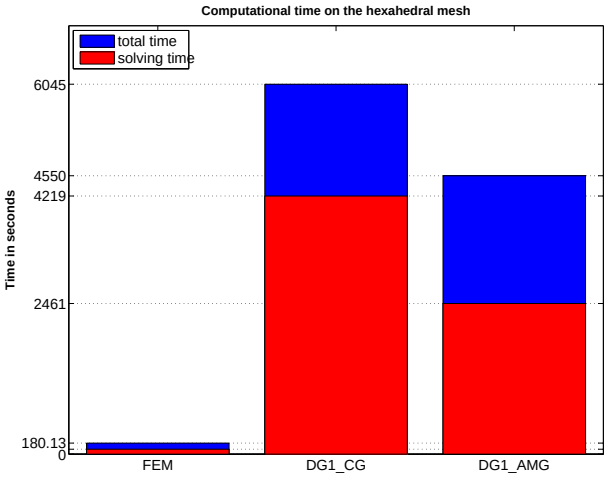

Figure 4.24.: Bar graph of the computational time per dipole on the hexahedral meshes.

For the simulation on the real head model the total computation time was 3028.98 seconds of which 2346.69 seconds were spent solving the system using the AMG solver.

Memory Usage

To obtain the measurements presented in this section, the Linux system tool `htop` was used. All the observations were made during the solving of the matrix system which should give a good estimate of the total memory demand as this is the time where the both the matrix and the right hand side vector have to be stored.

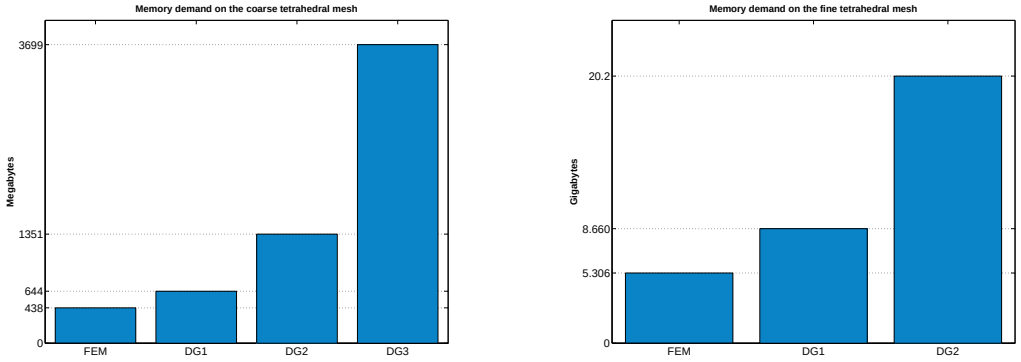

Figure 4.25.: Bar graph of the memory cost on the tetrahedral coarse tetrahedral mesh (left figure) and the fine tetrahedral mesh (right figure).

Figure 4.25 shows the memory demand for the different methods used on the coarse tetrahedral mesh. While all methods still have a quite moderate memory demand due to the comparably small number of elements in the mesh it still becomes clear that the demand in memory grows in a rapid fashion. As presented in Definition A.2 the number of basis functions for a local $\mathbb{P}_k^d$ basis is equal to $\binom{k+d}{k}$ and thus for fixed dimension $d = 3$ this becomes $(k+3)(k+2)(k+1)/6$ meaning that the number of DOFs will follow a cubic growth pattern when increasing the polynomial degree. The memory demand can be expected to grow accordingly. The issue becomes more pressing when looking at the memory consumption on the fine tetrahedral mesh. Even DG of polynomial degree one exceeds the capabilities of average desktop computers with a demand of 8.66 Gigabytes of memory.
For the real head model a memory demand of approximately 16.4 Gigabytes was observed. The hexahedral mesh shows comparably high values as shown in Figure 4.26 making it clear that DG simulations will in practice have to be run on machines with high memory capacities.

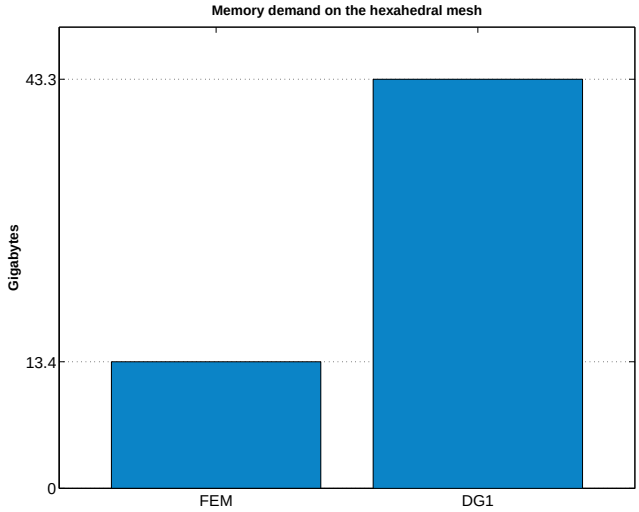

Figure 4.26.: Bar graph of the memory cost on the hexahedral.

Still it remains to say that the DG method can handle a large number of DOFs much more efficient than conforming Finite Element methods with respect to memory consumption. When neglecting any other source on memory cost for our implementation the ratio of memory per DOF on the coarse tetrahedral mesh becomes 0.01385 Megabytes/DOF for the conforming FEM while the DG methods handle the degrees of freedom with 0.000847, 0.000711 and 0.00094817 for the DG methods of polynomial degree one, two and three respectively. Similar values can be obtained for the other geometries. This fact is a result of the block sparsity pattern of the matrix as described in Section 2.6.1.

5. Conclusion and Outlook

In this thesis we have presented a method for dealing with the EEG Forward Problem that to the best of our knowledge has not been used in the field of source analysis as of yet. We have discussed the general properties of the EEG Forward Problem and the difficulties that arise by the introduction of a dipole model into this formulation. As a means to deal with this complication we presented the Subtraction Approach for conforming FEM. The basic framework for Discontinuous Galerkin methods was discussed and we used it to transfer the ideas of the Subtraction approach to a DG context. As a result the weak Discontinuous Galerkin formulation of the Subtraction Approach was presented. We analysed some basic qualities of this formulation and DG methods in general and motivated their application to the field of source analysis.
We presented the implementation of the DG method for the Subtraction Approach that was carried out in the context of this thesis using the DUNE framework. The main design features and work flow of the program were explained and we pointed out how the program can be extended.
Using the implementation we carried out a number of simulations on tetrahedral and hexahedral sphere models which were presented and discussed. We showed how the results obtained by a state of the art method like conforming FEM can be improved upon by the use of Discontinuous Galerkin methods with differing polynomial degrees. Especially in the case of sources on high eccentricities a big improvement could be observed when using higher order DG methods on tetrahedral meshes. Although not as conclusive as the results on the tetrahedra models and limited to polynomial order one the simulations on the hexahedral mesh showed some promising results which may have to be studied in more depth.
We concluded the evaluation of the numerical studies by discussing the computational cost of the DG methods giving an estimate for hardware requirements for practical applications.

With regard to further applications of Discontinuous Galerkin methods in the field of source analysis a number of possibilities are feasible:

As discussed in the derivation of the Subtraction DG Approach the method still lacks a proper proof of convergence. The work done by Ern [29] for heterogeneous diffusion problems with low-regularity solutions might serve as a blueprint for further efforts in this direction but will probably require some modifications and extensions due to the nature of our right hand side.

Since computational cost is still a big factor for the DG methods it would be advantageous to

make the implementation capable of *parallel computations* on a memory distributed architecture. This way both the high computational time and memory demand could be addressed. Initial work has already been put into this but still some more effort has to be made.

As already mentioned, one of the main applications of the EEG forward simulation is its key role in the solution of the inverse problem. If Discontinuous Galerkin simulations are to be used in this context it seems unavoidable to implement the generation of *transfer matrices* for rapid computations of the Forward Problem [35]. The standard methodology for the conforming FEM case will have to be adapted in order to compute transfer matrices using DG forward simulations.

With further regard to computational cost it should be said that compared to different dipole models for the EEG Forward Problem the Subtraction Approach requires the most effort due to its fully populated right hand side vector. Other dipole models, among the most popular the *Venant Approach* and the *Partial Integration Approach* [3], only produce sparse right hand side vectors resulting in greatly reduced computational effort. Also these methods are strong competitors with respect to the accuracy of the produced numerical results [5]. However, the methods as described in existing literature are only described for a node based formulation such as conforming FEM making some adaptations for the application of Discontinuous Galerkin methods necessary. Given a valid DG formulation of the approaches the task of implementing these methods in DUNE should prove rather straightforward using the implementation of the Subtraction Approach as a basis.

Another topic of growing interest within the field of brain research is that of *brain stimulation* using techniques like *transcranial direct current stimulation(tDCS)* and *transcranial alternating current stimulation(tACS)*. Those are non-invasive, easy-to-use and inexpensive techniques which have shown potential for the treating neurological conditions like Alzheimer's disease, Parkinson's disease and epilepsy. Further applications include the enhancement of cognitive functions and support of rehabilitation of motor functions after strokes. Finite Element simulations on very complex and detailed models have been performed [37] and produced more insight into this matter. Since methods like tDCS are performed by delivering a low electric current through the brain it may be beneficial to carry out numerical simulations using Discontinuous Galerkin methods due to their flux formulation and numerical conservation properties.

For practical applications the use of *real head models with anisotropic conductivities* is of course much more relevant than the study of rather artificial sphere models. The implementation of the local operator class of the Subtraction DG Approach is capable of handling anisotropic conductivities but some modifications would have to be made to the surrounding infrastructure in the program, e.g. the import of the conductivities for each element.

Real head models will most likely be present in the form of hexahedral meshes generated from MRI measurements as the generation of high quality tetrahedral models is a very sophisticated process. Since hexahedral models were only discussed briefly in this thesis and the results obtained in the simulations seem to leave room for improvement it might be necessary to carry out further simulations on hexahedra models first. The DG methods seems to suffer much more from the rough approximation of the sphere's surface than the conforming FEM. Accordingly it would be interesting to see if this issue can be improved when using *geometry-adapted meshes* [36]. The beneficial effects when using geometry-adapted hexahedral meshes for FEM based source analysis were studied in [39]. The current implementation will probably have to be modified slightly in order to be able to handle this kind of meshes.

A wholly different approach with regard to the use of meshes could be taken by using the *Unfitted Discontinuous Galerkin Method* [4]. This method combines the Unfitted Finite Element Method with Discontinuous Galerkin methods resulting in a setting where the computational mesh does not have to resolve the geometry of the domain. The information necessary for the approximation of the domain is taken directly from a source like MRI data and is subsequently evaluated during the matrix assembly using local triangulation techniques. By doing so the UDG method produces a high quality approximation of the domain's geometry even on relatively coarse computational meshes. The application of the Unfitted Discontinuous Galerkin method in the field of source analysis may not only alter the standard work flow of practical applications but may also help to avoid the high effort necessary to obtain high high quality tetrahedra meshes.

Regarding the accuracy of the DG methods of higher polynomial order than one the use of conforming Finite Element methods of degree one as a reference can only serve as a first orientation point. A more extensive study would include the comparison of Discontinuous Galerkin methods of a fixed polynomial degree with *Finite Element methods of corresponding degree*. The implementation of the FEM program used in this thesis allows for higher degree basis functions on tetrahedral meshes but including results from such simulation would have gone beyond the scope of this thesis.

We pointed out the qualities of DG methods for simulations on *locally refined meshes* and local *p-adaptivity* which might be another area in which applications in source analysis might be able to profit from Discontinuous Galerkin methods. The effects of an adaptive Finite Element method on the Subtraction Approach have been studied in [38] and might be a possible starting point. The combination of the grid refinement with local adaptation of the degree for the polynomial basis functions may prove fruitful as well.

A. Functional Analysis for Discontinuous Galerkin Methods

Here we will collect some general results from functional analysis and the theory of partial differential equations and also some notations and results that are important when analysing DG methods. In this we closely follow [27]. For proofs and more background we refer to [28] and [27].

In the following X and Y are Hilbert spaces and $\mathscr{L}(X,Y)$ denotes the vector space spanned by bounded linear operators from X to Y equipped with the norm

$$\|A\|_{\mathscr{L}(X,Y)} := \sup_{v\in X\setminus 0} \frac{\|Av\|_Y}{\|v\|_X} \qquad \forall A \in \mathscr{L}(X,Y).$$

Furthermore X' denotes the duality space of X.

Definition A.1 (Coercivity). Let X be a Hilbert space and let $a \in \mathscr{L}(X\times X,\mathbb{R})$. The bilinear form a is called *coercive* on X if there is a $C_{sta} > 0$ such that

$$\forall v \in X \quad C_{sta}\|v\|_X^2 \leq a(v,v).$$

Using coercivity, the well-posedness of the problem

$$\text{Find } u \in X \text{ s. t. } a(u,w) = \langle f, w\rangle_{X',X} \text{ f. a. } w \in X \tag{A.1}$$

where $\langle\cdot,\cdot\rangle_{X',X}$ denotes the duality pairing between X' and X, can be shown:

Lemma A.1 (Lax-Milgram Lemma). For $a \in \mathscr{L}(X\times X,\mathbb{R})$ and $f \in X'$ the problem (A.1) is well-posed if the bilinear form a is coercive on X. Furthermore it can be shown that:

$$\|u\|_X \leq \frac{1}{C_{sta}}\|f\|_{X'}$$

Definition A.2 (Polynomial Space $\mathbb{P}_d^k$)**.** Let $k \geq 0$ be an integer, $A_d^k := \left\{\alpha \in \mathbb{N}^d \mid |\alpha|_{l^1} \leq k\right\}$ and $|\cdot|_{l^p}$ the p-norm. Then the polynomial space $\mathbb{P}_d^k$ of polynomials of d variables, of total degree at most k, is defined as

$$\mathbb{P}_d^k := \left\{ p : \mathbb{R}^d \ni x \mapsto p(x) \in \mathbb{R} \mid \exists (\gamma_\alpha)_{\alpha \in A_d^k} \in \mathbb{R}^{card(A_d^k)} \text{ s. t. } p(x) = \sum_{\alpha \in A_d^k} \gamma_\alpha x^\alpha \right\} \quad \text{(A.2)}$$

The dimension of $\mathbb{P}_k^d$ is equal to $card(A_d^k) = \binom{k+d}{k}$.

Definition A.3 (Polynomial Space $\mathbb{Q}_d^k$)**.** Let $k \geq 0$ be an integer and $B_d^k := \left\{\alpha \in \mathbb{N}^d \mid |\alpha|_{l^\infty} \leq k\right\}$. Then the polynomial space $\mathbb{Q}_d^k$ of polynomials of degree at most k in each variable is defined as

$$\mathbb{Q}_d^k := \left\{ p : \mathbb{R}^d \ni x \mapsto p(x) \in \mathbb{R} \mid \exists (\gamma_\alpha)_{\alpha \in B_d^k} \in \mathbb{R}^{card(B_d^k)} \text{ s. t. } p(x) = \sum_{\alpha \in B_d^k} \gamma_\alpha x^\alpha \right\} \quad \text{(A.3)}$$

The dimension of $\mathbb{Q}_k^d$ is equal to $card(B_d^k) = (k+1)^d$.

Definition A.4 (Broken Polynomial Space)**.** Given a triangulation $\mathscr{T}_h$ and using Definitions A.2 and A.3 we define the broken polynomial spaces as

$$\mathbb{P}_d^k(\mathscr{T}_h) := \left\{ v \in L^2(\Omega) \mid \forall T \in \mathscr{T}_h, v|_T \in \mathbb{P}_d^k(T) \right\}$$

and

$$\mathbb{Q}_d^k(\mathscr{T}_h) := \left\{ v \in L^2(\Omega) \mid \forall T \in \mathscr{T}_h, v|_T \in \mathbb{Q}_d^k(T) \right\}$$

Definition A.5 (Sobolev Space $W^{m,p}(\Omega)$). Let Ω be a domain, $m \geq 0$ an integer, $1 \leq p \leq \infty$ a real number and $L^p(\Omega)$ the Lebesgue Space. Then the Sobolev Space $W^{m,p}(\Omega)$ is defined by

$$W^{m,p}(\Omega) := \{v \in L^p(\Omega) \mid \forall \alpha \in A_d^m, \partial^\alpha v \in L^p(\Omega)\}, \tag{A.4}$$

where A_d^m defined as in Definition A.2 and ∂^α the distributional derivative.
For $p = 2$ we write $H^m := W^{m,2}$. H^m becomes a Hilbert space when equipped with the scalar product

$$(v, w)_{H^m(\Omega)} := \sum_{\alpha \in A_d^m} (\partial^\alpha v, \partial^\alpha w)_{L^2(\Omega)}. \tag{A.5}$$

Definition A.6 (Element Wise Partial Integration Formula). For $\psi, \phi \in \mathbb{P}_d^k(\mathscr{T}_h)$(or $\psi, \phi \in \mathbb{Q}_d^k(\mathscr{T}_h)$) the following holds:

$$\begin{aligned} \int_\Omega \nabla \cdot \nabla \psi \phi \, \mathrm{d}x = &- \int_\Omega \nabla \psi \cdot \nabla \phi \, \mathrm{d}x + \int_{\partial\Omega} \nabla \psi \cdot \vec{n} \phi \, \mathrm{d}s \\ &+ \sum_{\gamma_{e,f} \in \Gamma_{int}} \int_{\gamma_{e,f}} [\![\nabla \psi \phi]\!] \, \mathrm{d}s \end{aligned} \tag{A.6}$$

Or equivalently

$$\begin{aligned} \int_\Omega \nabla \psi \cdot \nabla \phi \, \mathrm{d}x = &- \int_\Omega \nabla \cdot \nabla \psi \phi \, \mathrm{d}x + \int_{\partial\Omega} \nabla \psi \cdot \vec{n} \phi \, \mathrm{d}s \\ &+ \sum_{\gamma_{e,f} \in \Gamma_{int}} \int_{\gamma_{e,f}} [\![\nabla \psi]\!] \langle \phi \rangle + \langle \nabla \psi \rangle [\![\phi]\!] \, \mathrm{d}s \end{aligned} \tag{A.7}$$

Definition A.7 (Broken Sobolev Space $W^{m,p}(\mathscr{T}_h)$)**.** Given a triangulation $\mathscr{T}_h$ of the domain Ω, an integer $m \geq 0$ and a real number $1 \leq p \leq \infty$, we define the broken Sobolev space

$$W^{m,p}(\mathscr{T}_h) := \{v \in L^p(\Omega) \mid \forall T \in \mathscr{T}_h, v|_T \in W^{m,p}(T)\}$$

and

$$H^m := \left\{v \in L^2(\Omega) \mid \forall T \in \mathscr{T}_h, v|_T \in H^m(T)\right\}.$$

Here $W^{m,p}(T)$ and $H^m(T)$ denote the Sobolev spaces as introduced by Definition A.5.

Definition A.8 (Broken Gradient)**.** On the broken Sobolev space $W^{1,p}(\mathscr{T}_h)$ we define the *broken gradient* $\nabla_h : W^{1,p}(\mathscr{T}_h) \to [L^p(\Omega)]^d$ such that for all $v \in W^{1,p}(\mathscr{T}_h)$,

$$\forall T \in \mathscr{T}_h, \quad (\nabla_h v)|_T := \nabla(v|_T).$$

The normal Sobolev spaces are subspaces of the broken Sobolev spaces and on them broken gradient is equal to the distributional gradient. Furthermore the subscript h can be left out when the broken gradient appears inside an integral over a fixed mesh element $T \in \mathscr{T}_h$(see [27] for details).

Lemma A.2 (Discrete Trace Inequality)**.** For all $y_h \in V_h$, $\gamma \in \Gamma$ and $\gamma \subset \partial E$ for an $E \in \mathscr{T}_h$ we have

$$h_\gamma^{1/2} \|y_h\|_{L^q(\gamma)} \leq C_{tr} \|y_h\|_{L^2(E)} \tag{A.8}$$

where $C_t r$ only depends on the dimension d, the polynomial degree k of the functions in V_h and mesh regularity.

Proof. See for example [33] or [32]. □

Lemma A.3. For $v_h \in V_h$ and σ bounded we have

$$|v_h|_{\dagger,\sigma,2} \leq C_{tr} N_\partial^{1/2} \|\sigma^{1/2} \nabla_h v_h\|_{[L^2(\Omega)]^d} \tag{A.9}$$

with $|v_h|_{\dagger,\sigma,2}$ defined as in (2.35) and C_{tr} from Lemma A.2.

Proof.

$$\begin{aligned}
|v_h|_{\dagger,\sigma,2} &= \left(\sum_{E \in \mathcal{T}_h} \sum_{\gamma \in \Gamma_{int} : \gamma \subset \partial E} \underbrace{h_\gamma \|\sigma^{1/2} \nabla v_h \mid_E \cdot \vec{n}_\gamma\|^2_{L^2(\gamma)}}_{\leq C_{tr}^2 \|\sigma^{1/2} \nabla_h v_h\|^2_{[L^2(E)]^d}} \right)^{1/2} \\
&\leq C_{tr} \left(\sum_{E \in \mathcal{T}_h} \underbrace{\sum_{\gamma \in \Gamma_{int} : \gamma \subset \partial E} \|\sigma^{1/2} \nabla_h v_h\|^2_{[L^2(E)]^d}}_{\leq N_\partial \|\sigma^{1/2} \nabla_h v_h\|^2_{[L^2(E)]^d}} \right)^{1/2} \\
&\leq C_{tr} N_\partial^{1/2} \left(\sum_{E \in \mathcal{T}_h} \|\sigma^{1/2} \nabla_h v_h\|^2_{[L^2(E)]^d} \right)^{1/2} \\
&= C_{tr} N_\partial^{1/2} \|\sigma^{1/2} \nabla_h v_h\|_{[L^2(\Omega)]^d}
\end{aligned}$$

□

Lemma A.4. For $v_h \in V_h$ and $w \in V_{\dagger h}$ we have

$$\left| \sum_{\gamma_{e,f} \in \Gamma_{int}} \int_{\gamma_{e,f}} \langle \sigma \nabla_h v_h \rangle [\![w_h]\!] \right| \leq |v_h|_{\dagger,\sigma,2} |w|_{J,\sigma}. \tag{A.10}$$

Proof. Using the Cauchy-Schwarz inequality, see [29]. □

B. Discontinuous Galerkin Subtraction Approach

Consistency of the Weak DG Formulation for the Subtraction Approach

Taking the weak DG formulation of the subtraction approach as described in definition 2.7 we need to show that

$$\alpha_{DG}^{cons}(u^{corr,y}, v) = \lambda(v)$$

for the solution $u^{corr,y} \in H^1(\Omega)$ of the subtraction approach as described in definition 1.3 with

$$\alpha_{DG}^{cons}(u,v) := -\int_\Omega \sigma(x)\nabla u(x)\cdot\nabla v(x)dx + \sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}} \langle\sigma\nabla u\rangle(s)[\![v]\!](s)ds$$

and

$$\lambda(v) := \int_\Omega \sigma^{corr,y}(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)dx + \int_{\partial\Omega}\sigma^{\infty,y}(x)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)ds - \sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma^{corr,y}\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)ds$$

Taking $u^{corr,y}$ as described above and inserting it into $\alpha_{DG}^{cons}(u,v)$ we get:

$$\begin{aligned}
&\alpha_{DG}^{cons}(u,v) = -\int_\Omega \sigma(x)\nabla u^{corr,y}(x)\cdot\nabla v(x)dx + \sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma\nabla u^{corr,y}\rangle(s)[\![v]\!](s)ds\\
&= \int_\Omega \nabla\cdot\sigma(x)\nabla u^{corr,y}v(x)dx - \int_{\partial\Omega}\sigma\nabla u^{corr,y}(s)\vec{n}v(s)ds\\
&- \sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![\sigma\nabla u^{corr,y}]\!](s)\langle v\rangle(s) + \langle\sigma\nabla u^{corr,y}\rangle(s)[\![v]\!](s)ds + \sum_{\gamma\in\Gamma_{int}}\int_{\gamma_{e,f}}\langle\sigma\nabla u^{corr,y}\rangle(s)[\![v]\!](s)ds\\
&= \int_\Omega \nabla\cdot\sigma(x)\nabla u^{corr,y}(x)v(x)dx - \int_{\partial\Omega}\sigma\nabla u^{corr,y}(s)\vec{n}v(s)ds - \sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![\sigma\nabla u^{corr,y}]\!](s)\langle v\rangle(s)ds
\end{aligned}$$

In the above the partial integration formula was used for the second equality.
Using the Neumann boundary condition the boundary term can be substituted by

$$-\int_{\partial\Omega}\sigma\nabla u^{corr,y}(s)\vec{n}v(s)ds = \int_{\partial\Omega}\sigma\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)ds$$

and split up into

$$\int_{\partial\Omega}\sigma(s)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)ds=\int_{\partial\Omega}\sigma^{corr,y}(s)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)ds+\int_{\partial\Omega}\sigma^{\infty,y}(s)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)ds$$

arriving at

$$\begin{aligned}&\int_{\Omega}\nabla\cdot\sigma(x)\nabla u^{corr,y}(x)v(x)dx+\int_{\partial\Omega}\sigma^{corr,y}(s)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)ds\\&+\int_{\partial\Omega}\sigma^{\infty,y}(s)\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)ds-\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![\sigma\nabla u^{corr,y}]\!](s)\langle v\rangle(s)ds\end{aligned}$$

The partial integration formula is applied to the second term giving us

$$\begin{aligned}&\int_{\Omega}\nabla\cdot\sigma(x)\nabla u^{corr,y}(x)v(x)dx+\int_{\Omega}\nabla\cdot\sigma^{corr,y}(x)\nabla u^{\infty,y}(x)dx+\int_{\Omega}\sigma^{corr,y}(x)\nabla u^{\infty,y}(x)\cdot\nabla v(x)dx\\&-\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![\sigma^{corr,y}\nabla u^{\infty,y}]\!](s)\langle v\rangle(s)+\langle\sigma^{corr,y}\nabla u^{\infty,y}\rangle(s)[\![v]\!](s)ds\\&+\int_{\partial\Omega}\sigma^{\infty,y}\nabla u^{\infty,y}(s)\cdot\vec{n}v(s)ds-\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![\sigma\nabla u^{corr,y}]\!](s)\langle v\rangle(s)ds\end{aligned}$$

The first two terms add up to zero since

$$\begin{aligned}&\int_{\Omega}\nabla\cdot\sigma(x)\nabla u^{corr,y}(x)v(x)dx+\int_{\Omega}\nabla\cdot\underbrace{\sigma^{corr,y}}_{\sigma^{corr,y}=\sigma-\sigma^{\infty,y}}(x)\nabla u^{\infty,y}(x)dx\\&=\int_{\Omega}\nabla\cdot\sigma(x)\nabla u^{corr,y}(x)v(x)dx+\int_{\Omega}\nabla\cdot\sigma(x)\nabla u^{\infty,y}(x)dx-\int_{\Omega}\nabla\cdot\sigma^{\infty,y}(x)\nabla u^{\infty,y}(x)dx\\&=\underbrace{\int_{\Omega}\nabla\cdot\sigma(x)\nabla u(x)v(x)dx}_{\int_{\Omega}f(x)v(x)dx}-\int_{\Omega}\nabla\cdot\sigma^{\infty,y}(x)\nabla u^{\infty,y}(x)dx=0\end{aligned}$$

where in the last step the property of the singularity potential $u^{\infty,y}$ was used(see section 1.5). In order to show the consistency it remains to show that

$$\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![\sigma^{corr,y}\nabla u^{\infty,y}]\!](s)\langle v\rangle(s)+[\![\sigma\nabla u^{corr,y}]\!](s)\langle v\rangle(s)ds=0$$

This holds true since using $\sigma^{corr,y}=\sigma-\sigma^{\infty,y}$ on the first term yields

$$\begin{aligned}&\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}[\![\sigma\nabla u^{\infty,y}]\!](s)\langle v\rangle(s)+[\![\sigma\nabla u^{corr,y}]\!](s)\langle v\rangle(s)-[\![\sigma^{\infty,y}\nabla u^{\infty,y}]\!](s)\langle v\rangle(s)ds\\&=\sum_{\gamma_{e,f}\in\Gamma_{int}}\int_{\gamma_{e,f}}\underbrace{[\![\sigma\nabla u]\!]}_{=0}(s)\langle v\rangle(s)-\underbrace{[\![\sigma^{\infty,y}\nabla u^{\infty,y}]\!]}_{=0}(s)\langle v\rangle(s)ds=0\end{aligned}$$

and consistency has been shown. □

C. Additional Plots and Figures

Study 1: Coarse Tetrahedra Model

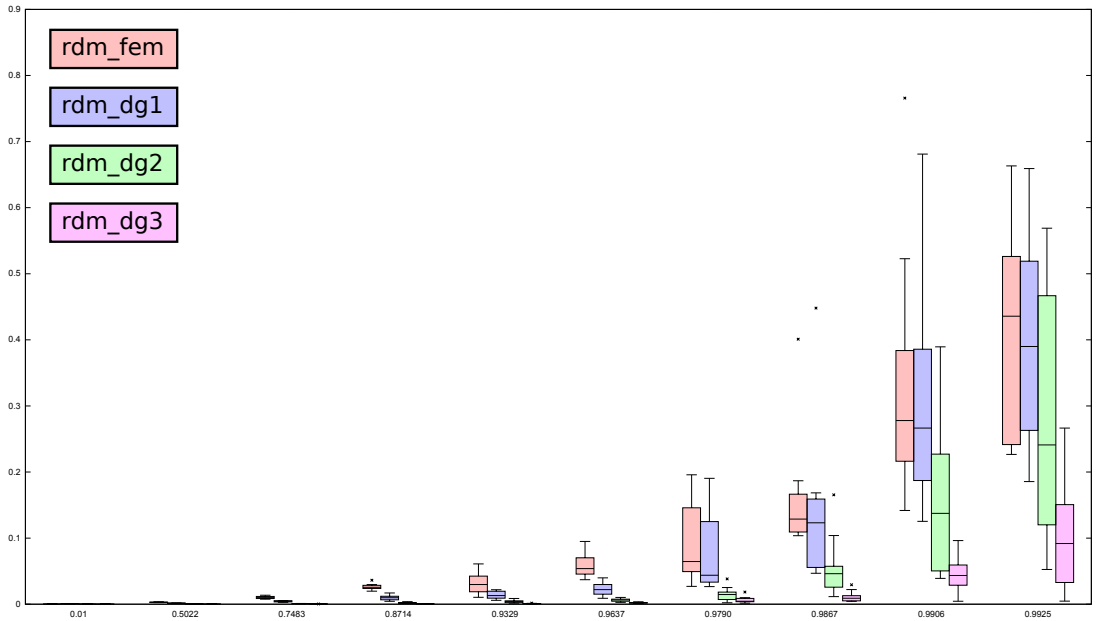

Figure C.1.: Boxplot for the RDM of the radial sources on the coarse tetrahedral model.

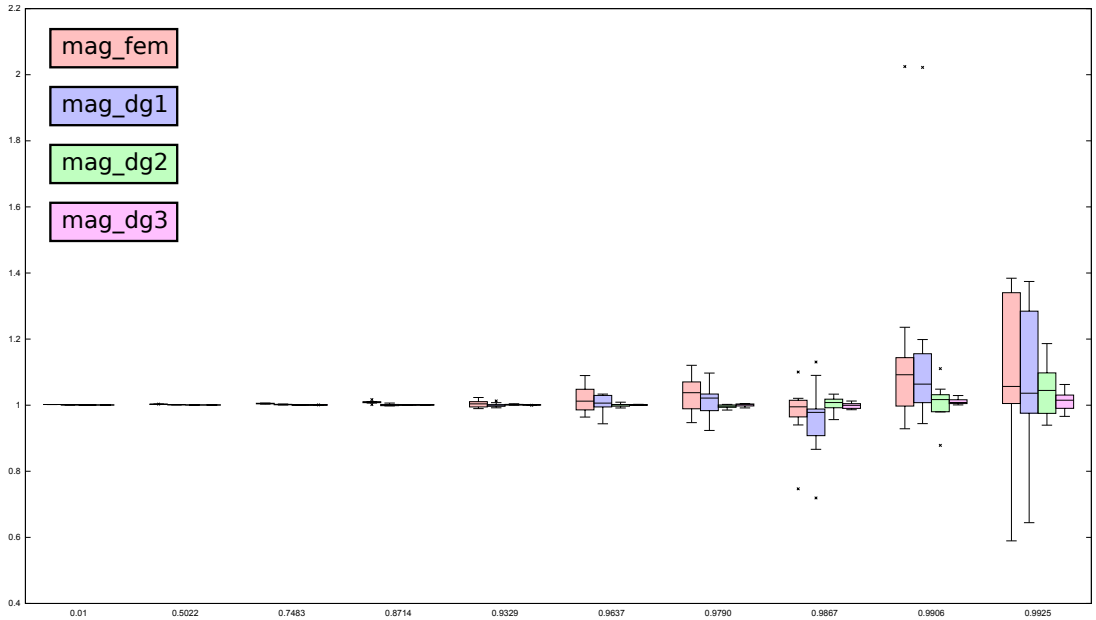

Figure C.2.: Boxplot for the MAG of the radial sources on the coarse tetrahedral model.

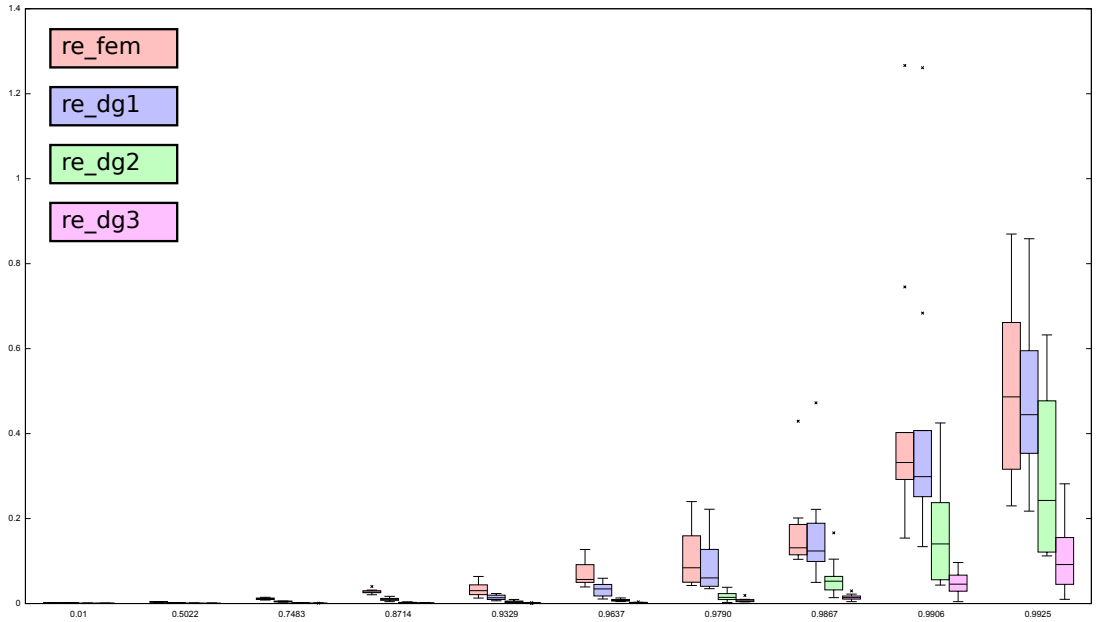

Figure C.3.: Boxplot for the relative error of the radial sources on the coarse tetrahedral model.

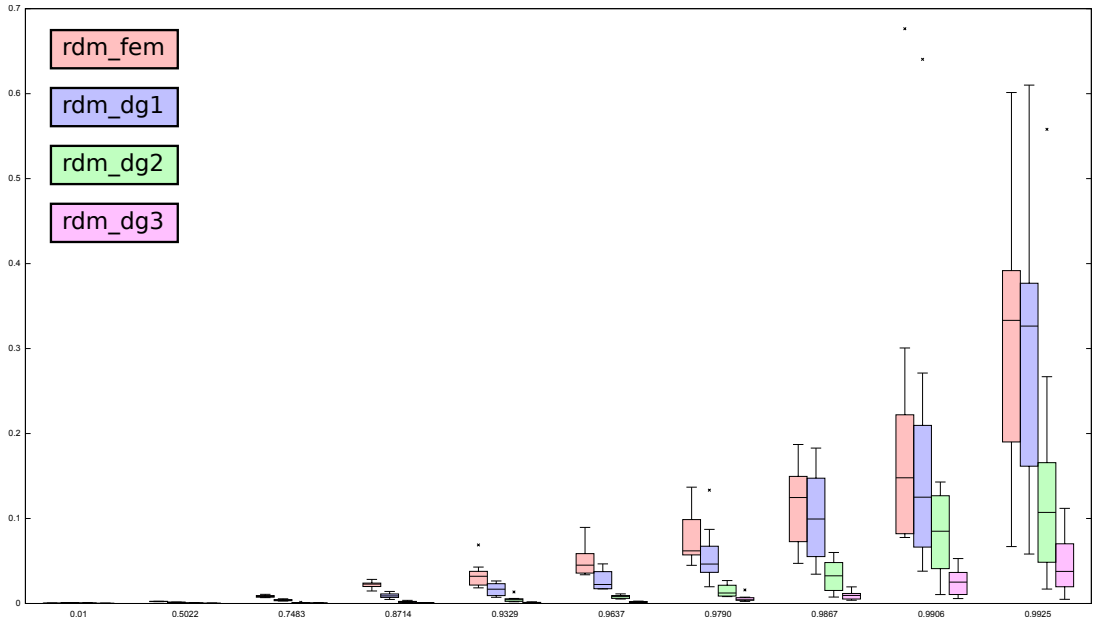

Figure C.4.: Boxplot for the RDM of the tangential sources on the coarse tetrahedral model.

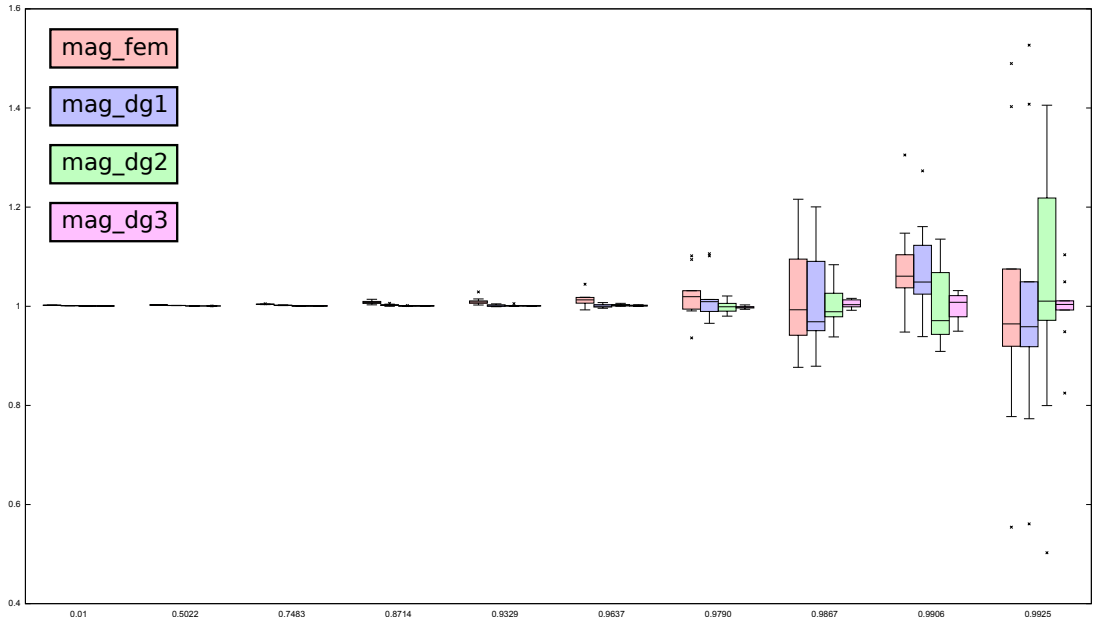

Figure C.5.: Boxplot for the MAG of the tangential sources on the coarse tetrahedral model.

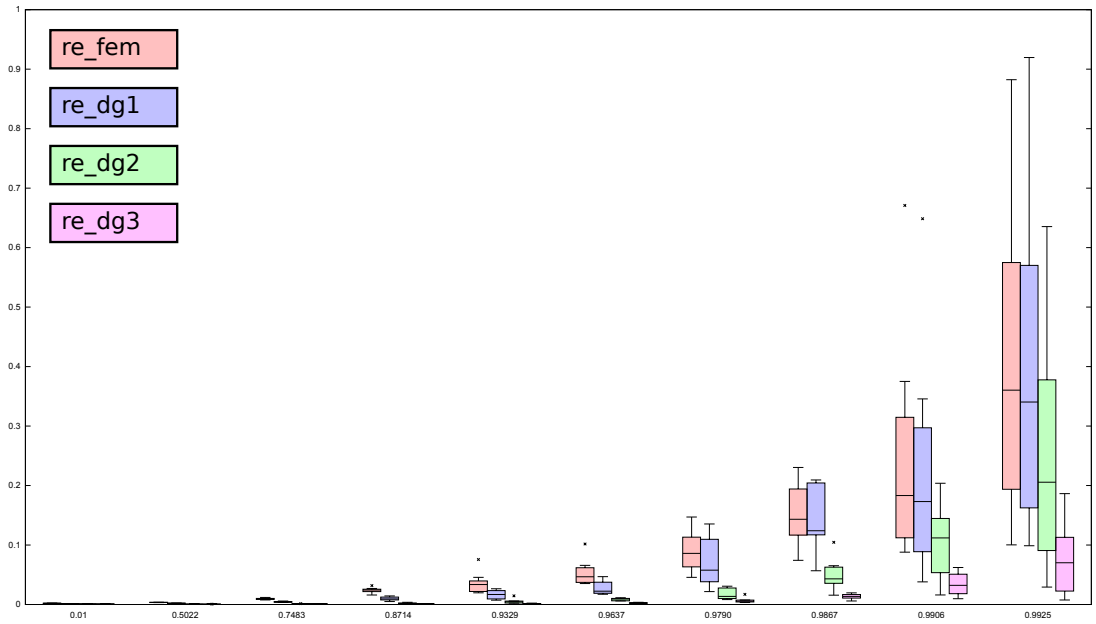

Figure C.6.: Boxplot for the relative error of the tangential sources on the coarse tetrahedral model.

List of Figures

List of Tables

Bibliography

[1] S. Murakami and Y.C. Okada, *Contributions of principal neocortical neurons to magnetoencephalography and electroencephalography signals*. The Journal of Physiology, 575(3):925–936, 2006.

[2] M. Hämäläinen, R. Hari, R. Ilmoniemi, J. Knuutila, and O. Lounasmaa, *Magnetoencephalography – theory, instrumentation, and applications to noninvasive studies of the working human brain*. Reviews of Modern Physics, 65(2): 413–497, 1993.

[3] C.H. Wolters, H. Köstler, C. Möller, J. Härdtlein, A. Anwandler, *Numerical approaches for dipole modelling in finite element method based source analysis*. International Congress Series 130 (2007): 189-192, 2007

[4] C. Engwer, *An Unfitted Discontinuous Galerkin Scheme for Micro-scale Simulations and Numerical Upscaling*. Naturwissenschaftlich-Mathematische Gesamtfakultät der Ruprecht-Karls-Universität Heidelberg, 2009.

[5] J. Vorwerk, *Comparison of Numerical Approaches to the EEG Forward Problem*. Fachbereich 10 Mathematik und Informatik, Institut für Biomagnetismus und Biosignalanalyse, Westfälische Wilhelms-Universität Münster, 2011.

[6] J. Nitsche, *Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind*. Abhandlungen des Mathematischen Seminars der Universität Hamburg, 36: 9-15, 1971.

[7] I. Babuška, *Numerical solution of boundary value problems by the perturbed variational principle*. Technical Note BN-624, Institute of Fluid Dynamics and Applied Mathematics, University of Maryland, 1969.

[8] I. Babuška, M. Zlámal, *Nonconforming elements in the finite element method with penalty*. SIAM Journal on Numerical Analysis, Volume 10, No. 5: 863-875, 1973.

[9] D. N. Arnold, F. Brezzi, B. Cockburn, L. D. Marini, *Unified Analysis of Discontinuous Galerkin Methods for Elliptic Problems*. SIAM Journal on Numerical Analysis, Volume 39, No.5: 1749-1779, 2002.

[10] A. Ern, A. F. Stephansen, P. Zunino, *A Discontinuous Galerkin method with weighted averages for advection-diffusion equations with locally small and anisotropic diffusivity*. IMA Journal of Numerical Analysis, Volume 29, No.2: 235-256, 2009.

[11] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, M. Ohlberger, O. Sander, *A Generic Grid Interface for Parallel and Adaptive Scientific Computing. Part I: Abstract Framework*. Computing, 82(2-3): 103-119, 2008.

[12] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, R. Kornhuber, M. Ohlberger, O. Sander, *A Generic Grid Interface for Parallel and Adaptive Scientific Computing. Part II: Implementation and Tests in DUNE*. Computing, 82(2-3): 121-138, 2008.

[13] M. Blatt, P. Bastian, *The Iterative Solver Template Library*. In B. Kåström, E. Elmroth, J. Dongarra and J. Wasniewski, Applied Parallel Computing. State of the Art in Scientific Computing, Volume 4699 of Lecture Notes in Scientific Computing, Springer: 666-675, 2007.

[14] P. Bastian, M. Blatt, *On the Generic Parallelisation of Iterative Solvers for the Finite Element Method*. In Int. J. Computational Science and Engineering, 4(1):56-69, 2008.

[15] F. Drechsler, C. H. Wolters, T. Dierkes, H. Si, L. Grasedyck, *A full subtraction approach for finite element method based source analysis using constrained Delaunay tetrahedralisation*. Neuroimage 46: 1055-165, 2009.

[16] C. H. Wolters, *Mathematical methods in bioelectromagnetism and in the analysis of biosignals*. Fakultät für Mathematik und Informatik, Westfälische Wilhelms-Universität Münster, http://www.sci.utah.edu/~wolters/LiteraturZurVorlesung/Vorlesungsskriptum, 2012.

[17] A. Dedner, C. Rohde, B. Schupp, M. Wesenberg, *A Parallel, Load Balanced MHD Code on Locally Adapted, Unstructured Grids in 3D*. Preprint Mathematisches Institut, Universität Freiburg No. 30/2002, Computing and Visualization in Science, 7, 79-96, 2004.

[18] A. Burri, A. Dedner, R. Klöfkorn, M. Ohlberger, *An efficient implementation of an adaptive and parallel grid in DUNE*. Proceedings of the 2nd Russian-German Advanced Research Workshop on Computational Science and High Performance Computing, 2005.

[19] J. C. de Munck, M. J. Peters, *A Fast Method to Compute the Potential in the Multisphere Model*. IEEE Transactions on Biomedical Engineering, Vol. 40, No. 11, 1993.

[20] J. C. de Munck, B. W. van Dijk, H. Spekreijse, *Mathematical Dipoles are Adequate to Describe Realistic Generators of Human Brain Activity*. IEEE Transactions on Biomedical Engineering, Vol. 35, No. 11, 1988.

[21] J. C. de Munck, *The potential distribution in a layered anisotropic spheroidal volume conductor*. Journal of Applied Physics, Vol. 64, No. 2: 464-470, 1988.

[22] C. Geuzaine, J.-F. Remacle, *Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities*. International Journal for Numerical Methods in Engineering 79(11): 1309-1331, 2009.

[23] M. Dannhauer, B. Lanfer, C. H. Wolters and T. R. Knösche, *Modeling of the Human Skull in EEG Source Analysis*. Human Brain Mapping 32(9): 1383-1399, 2011.

[24] M. Ahktari, H. C. Bryant, A. N. Mamelak, L. Heller, J. J. Shih, M. Mandelkern, A. Matlachov, D. M. Ranken, E. D. Best, W. W. Sutherling, *Conductivities of Three-Layer Human Skull*. Brain Topography 13(1)(September 1): 29-42, 2000.

[25] S. B. Baumann, D. R. Wozny, S. K. Kelly, F. M. Meno, *The Electrical Conductivity of Human Cerebrospinal Fluid at Body Temperature*. Biomedical Engineering, IEEE Transactions On 44(3)(March): 220-223, 1997.

[26] C. Ramon, P. Schimpf, J. Haueisen, M. Holmes, A. Ishimaru, *Role of Soft Bone, CSF and Gray Matter in EEG Simulations*. Brain Topography 16(4): 245-248, 2004.

[27] D. A. Di Pietro, A. Ern, *Mathematical Aspects of Discontinuous Galerkin Methods*, Springer, 2012.

[28] L. C. Evans, *Partial Differential Equations*. American Mathematical Society, 2010.

[29] D. A. Di Pietro, A. Ern, *Analysis of a Discontinuous Galerkin Method for Heterogeneous Diffusion Problems with Low-Regularity Solutions*. Numer. Methods Partial Differ. Equations, 28: 1161-1171, 2012.

[30] S. Nicaise, A. M. Sändig, *General Interface Problems I, II*. Math. Methods Appl. Sci., 17(6):395-429, 431-450, 1994.

[31] D. A. Di Pietro, A. Ern, *Discrete functional analysis tools for discontinuous Galerkin methods with application to the incompressible Navier-Stokes equations*. Math. Comp., 89:1303-1330, 2010.

[32] A. Ern, J.-L. Guermond, *Theory and Practice of Finite Elements*. Applied Mathematical Scineces, Volume 159, Springer, 2004.

[33] S. C. Brenner, L. R. Scott, *The mathematical theory of finite element methods*. Texts in Applied Mathematics, Volume 15, Springer, 2008.

[34] P. Bastian, M. Blatt, R. Scheichl, *Algebraic multigrid for discontinuous galerkin discretizations of heterogenous elliptic problems*. Numerical Linear Algebra with Applications, 19(2): 367-388, 2012.

[35] C. H. Wolters, L. Grasedyck, W. Hackbusch, *Efficient computation of lead field bases and influence matrix for the FEM-based EEG and MEG inverse problems*. Inverse Problems, 20(4):1099-1116, 2004.

[36] D. Camacho, R. Hopper, G. Lin, B. Myers, *An improved method for finite element mesh generation of geometrically complex structures with application to the skullbase*. Journal of Biomechanics, 30(10):1067-1070, 1997.

[37] S. Wagner, S. M. Rampersad, Ü. Aydin, J. Vorwerk, T. F. Oostendorp, T. Nestling, C. S. Hermann, D. F. Stegeman, C. H. Wolters, *Investigation of tDCS volume conduction effects in a highly realistic head model*. Journal of Neural Engineering, 2014. Preprint viewed on 19th November, 2013: `http://www.sci.utah.edu/~wolters/PaperWolters/2014/WagnerRampersadAydinVorwerkOostendorpNeulingHerrmannStegemanWolters_JNeuralEng_2014_Webversion.pdf`

[38] F. Meyer, *Convergent adaptive Finite Element Methods for the solution of the EEG forward problem with the help of the subtraction method*. Master thesis, Fachbereich 10 Mathematik und Informatik, Institut für Biomagnetismus und Biosignalanalyse, Westfälische Wilhelms-Universität Münster, 2013.

[39] C. H. Wolters, A. Anwander, G. Berti, U. Hartmann, *Geometry-Adapted Hexahedral Meshes Improve Accuracy of Finite-Element-Method-Based EEG Source Analysis*. IEEE Transactions on Biomedical Engineering, 54(8): 1446-1453, 2007.

Erklärung der Eigenständigkeit

Hiermit erkläre ich, Jakob Ludewig, dass die vorliegende Arbeit selbständig verfasst worden ist, dass keine anderen Quellen und Hilfsmittel als die angegebenen benutzt worden sind und dass die Stellen der Arbeit, die anderen Werken - auch elektronischen Medien - dem Wortlaut oder Sinn nach entnommen wurden, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht worden sind.
Die Kugelmodelle und das realistische Kopfmodell wurden mir zur Verfügung gestellt von Dipl.-Math. Johannes Vorwerk. Die Visualisierungen in Abschnitt 4.4 wurden von Dipl.-Math. Sven Wagner vorgenommen.
Die auf der beiliegenden CD eingereichten Programme wurden, aufbauend auf frei verfügbarer Software, selbständig programmiert.

Ort, Datum

Unterschrift