TECHNISCHE UNIVERSITÄT
BERGAKADEMIE FREIBERG
The University of Resources. Since 1765.

Three-dimensional individual and joint inversion of direct current resistivity and electromagnetic data

By the Faculty of Geosciences, Geoengineering and Mining
of the Technische Universität Bergakademie Freiberg

approved

Thesis

to attain the academic degree of

Doktor der Naturwissenschaften
(Dr. rer. nat.)

submitted by **Dipl.–Geophys. Julia Weißflog**

born on the 27th of June, 1987 in Stollberg/Erzgeb.

Assessors: Prof. Dr. Klaus Spitzer, Freiberg, Germany
Prof. Dr. Oliver G. Ernst, Chemnitz, Germany
Prof. Dr. Yuguo Li, Qingdao, China

Date of the award: Freiberg, 7th of February, 2017

Abstract

The objective of our studies is the combination of electromagnetic and direct current (DC) resistivity methods in a joint inversion approach to improve the reconstruction of a given conductivity distribution. We utilize the distinct sensitivity patterns of different methods to enhance the overall resolution power and ensure a more reliable imaging result.

In order to simplify the work with more than one electromagnetic method and establish a flexible and state-of-the-art software basis, we developed new DC resistivity and electromagnetic forward modeling and inversion codes based on finite elements of second order on unstructured grids. The forward operators are verified using analytical solutions and convergence studies before we apply a regularized Gauss-Newton scheme and successfully invert synthetic data sets. Finally, we link both codes with each other in a joint inversion.

In contrast to most widely used joint inversion strategies, where different data sets are combined in a single least-squares problem resulting in a large system of equations, we introduce a sequential approach that cycles through the different methods iteratively. This way, we avoid several difficulties such as the determination of the full set of regularization parameters or a weighting of the distinct data sets. The sequential approach makes use of a smoothness regularization operator which penalizes the deviation of the model parameters from a given reference model. In our sequential strategy, we use the result of the preceding individual inversion scheme as reference model for the following one. We successfully apply this approach to synthetic data sets and show that the combination of at least two methods yields a significantly improved parameter model compared to the individual inversion results.

Kurzfassung

Ziel der vorliegenden Arbeit ist die gemeinsame Inversion (*joint inversion*) elektromagnetischer und geoelektrischer Daten zur Verbesserung des rekonstruierten Leitfähigkeitsmodells. Dabei nutzen wir die verschiedenartigen Sensitivitäten der Methoden aus, um die Auflösung zu erhöhen und ein zuverlässigeres Ergebnis zu erhalten.

Um die Arbeit mit mehr als einer Methode zu vereinfachen und eine flexible Softwarebasis auf dem neuesten Stand der Forschung zu etablieren, wurden zwei Codes zur Modellierung und Inversion geoelektrischer als auch elektromagnetischer Daten neu entwickelt, die mit finiten Elementen zweiter Ordnung auf unstrukturierten Gittern arbeiten. Die Vorwärtsoperatoren werden mithilfe analytischer Lösungen und Konvergenzstudien verifiziert, bevor wir ein regularisiertes Gauß-Newton-Verfahren zur Inversion synthetischer Datensätze anwenden.

Im Gegensatz zur meistgenutzten *joint inversion*-Strategie, bei der verschiedene Daten in einem einzigen Minimierungsproblem kombiniert werden, was in einem großen Gleichungssystem resultiert, stellen wir schließlich einen sequentiellen Ansatz vor, der zyklisch durch die einzelnen Methoden iteriert. So vermeiden wir u.a. eine komplizierte Wichtung der verschiedenen Daten und die Bestimmung aller Regularisierungsparameter in einem Schritt. Der sequentielle Ansatz wird über die Anwendung einer Glättungsregularisierung umgesetzt, bei der die Abweichung der Modellparameter zu einem gegebenen Referenzmodell bestraft wird. Wir nutzen das Ergebnis der vorangegangenen Einzelinversion als Referenzmodell für die folgende Inversion. Der Ansatz wird erfolgreich auf synthetische Datensätze angewendet und wir zeigen, dass die Kombination von mehreren Methoden eine erhebliche Verbesserung des Inversionsergebnisses im Vergleich zu den Einzelinversionen liefert.

Contents

List of abbreviations

2D	two-dimensional
3D	three-dimensional
COMMEMI	comparison of modeling methods for electromagnetic induction problems
CSEM	controlled-source electromagnetic
DC	direct current
DtN	Dirichlet-to-Neumann
DOFs	degrees of freedom
MT	magnetotelluric
PCG	preconditioned conjugate gradients
PDE	partial differential equation
RT	Raviart-Thomas
TEM	transient electromagnetic
VLF-R	very low frequency-resistivity

1 Introduction and outline of the thesis

The acquisition and interpretation of geophysical data to investigate the upper part of the Earth's crust has occupied both scientists and exploration companies for decades. Adequate design and sufficiently exact evaluation of a geophysical measurement requires continued work on the existing software and its physical and mathematical background to further improve and possibly accelerate the available algorithms. We have to constantly revise our approaches to eliminate mistakes and replace outdated algorithms with new techniques. Codes have to be adjusted to new developments in our understanding of the measurement of geophysical data and the application of inversion schemes.

As a part of this ongoing research and advancement of available methods, in this thesis we developed two new modeling and inversion codes for dealing with direct current (DC) resistivity and electromagnetic data. These codes are partially based on known algorithms, but we also examined and implemented new approaches. Each of the two individual codes enables us to simulate and invert the particular data sets. Furthermore, we present a sequential approach to joint inversion, where we combine both codes to improve the inversion result. All parts of the software are implemented in MATLAB to achieve standardization of our codes and establish a flexible and state-of-the-art software basis to simplify the work with more than one geophysical method.

Overall, the thesis is divided into three main chapters. Chapters 2 and 3 deal with the modeling and individual inversion of DC resistivity and electromagnetic data, respectively. Here, we describe the historical and physical background of the methods and give a detailed explanation of the mathematical techniques concerning the assembly of the linear system and the solution of the resulting least-squares problem. Therefore, these chapters are more or less a detailed documentation of the codes and we advise future users of the codes to read through the corresponding chapter in order to understand the implementation and to be able to work with it or add further features. An overview over the file structure is given in the Appendix. Chapter 4 explains the sequential approach to joint inversion and shows its applicability and advantages using two synthetic examples. We end with a summary and outlook in Chapter 5.

2 DC resistivity modeling and inversion

The DC resistivity method is one of the most common and cost-effective geophysical exploration techniques. In the past decades, there has been an ongoing advancement and improvement in data acquisition and interpretation for all geophysical methods. The presence of increasingly reliable geoelectrical data sets promoted and still promotes the development of multifunctional simulation and inversion algorithms to evaluate and interpret these huge amounts of data in an appropriate manner. There is a limited number of solution techniques for the three-dimensional (3D) DC resistivity problem discussed in the literature: The integral equation method implemented by Dieter et al. (1969) and Hohmann (1975) is applied to simple model geometries (Bing and Greenhalgh, 2001). The finite difference method (Mufti (1976), Dey and Morrison (1979), Lowry et al. (1989), Park and Van (1991), Spitzer (1995)) approximates the derivatives in the equation of continuity (Section 2.2.1) by finite differences. The newest solution technique is the finite element method (Coggon (1971), Pridmore et al. (1981), Sasaki (1994), Rücker et al. (2006), Yuan et al. (2016)) which approximates the solution of a variational formulation. Referring to Li and Spitzer (2002) who compared finite difference and finite element methods for the DC resistivity method as well as to Bing and Greenhalgh (2001) who emphasized the accuracy and computational efficiency in finite element modeling, we chose to use this approximation method in our studies, too. Furthermore, Rücker et al. (2006) point out important advantages of an unstructured finite element grid, which are "efficient local mesh refinement" and "flexible model geometry description". The inaccuracy of the solution due to singularities in the source term is eliminated by the singularity removal technique introduced by Coggon (1971) and Lowry et al. (1989) and enhanced by Zhao and Yedlin (1996). The inversion schemes based on the three different solution techniques were first published by Park and Van (1991), Ellis and Oldenburg (1994) and Zhang et al. (1995). Most of the inversion approaches are not new and based on Gauss-Newton schemes, which we also use. Finite element inversion schemes were implemented by Sasaki (1989), Yi et al. (2003) and Günther et al. (2006). Here, we present a 3D DC resistivity finite element forward and inversion algorithm using unstructured grids and a smoothness regularization based on a mixed finite element formulation. The resulting least-squares problem is solved with a direct solver or an iterative Krylov subspace method (Hestenes and Stiefel (1952), Paige and Saunders (1982)). The code was developed and implemented in the

plemented in the Geotechnologien project *Three-dimensional Multi-Scale and Multi-Method Inversion to Determine the Electrical Conductivity Distribution of the Subsurface Using Parallel Computing Architectures* (Multi-EM) in close collaboration with Felix Eckhofer.

2.1 Physical principles

The DC resistivity method is a geoelectrical method to explore the upper part of the Earth's crust. It came into use in the first part of the 20th century (Zhdanov and Keller, 1994). Examples of application are mineral exploration, detection of cavities and groundwater or localization of hazardous waste. The aim of a geoelectrical survey is to get a sufficiently exact image of the subsurface regarding electrical properties. In contrast to a geological mapping, geoelectrical methods try to characterize the geological setting by means of petrophysical parameters which are for example the electrical resistivity ρ, its reciprocal parameter the electrical conductivity σ or the electrical permittivity ε. DC resistivity methods make use of artificially generated, stationary electric fields. Figure 2.1 illustrates the setup of a geo-

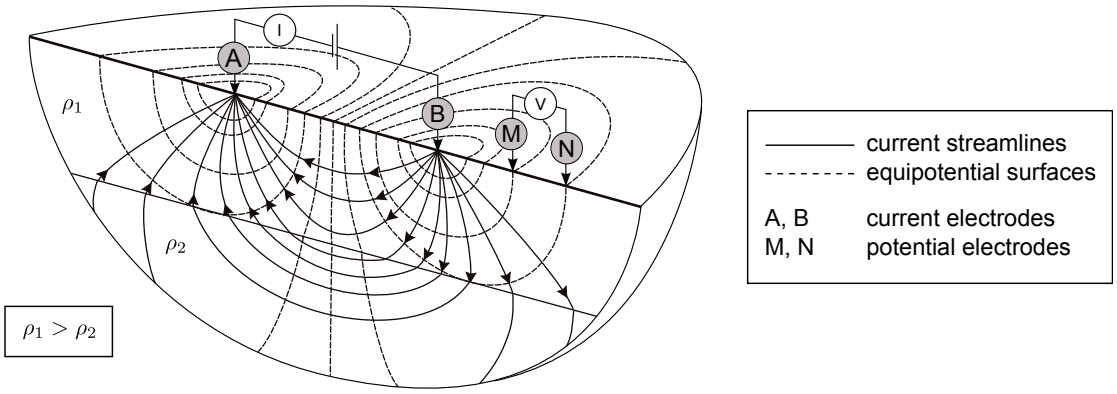

Fig. 2.1: Basic principle of a DC resistivity measurement, adapted from Knödel et al. (2005b).

electrical measurement, where we inject currents into the conductive ground using galvanic contacts which are the current electrodes A and B in Figure 2.1. These currents disseminate into the Earth and evoke a potential field in the subsurface which is influenced and altered by the present conductivity structures as follows: Away from electric current sources in a homogeneous isotropic medium, there holds

$$\nabla \cdot \boldsymbol{j} = 0$$

for the electric current density $\boldsymbol{j}$. Inserting Ohm's law (without displacement currents)

$$\boldsymbol{j} = \sigma \boldsymbol{E}$$

with the electric field $\boldsymbol{E}$, we get

$$\nabla \cdot (\sigma \boldsymbol{E}) = 0$$

$$\nabla\sigma \cdot \boldsymbol{E} + \sigma\nabla \cdot \boldsymbol{E} = 0.$$

We insert the Maxwell equation (see Section 3.1)

$$\nabla \cdot \boldsymbol{D} = \nabla \cdot \varepsilon \boldsymbol{E} = q, \quad \varepsilon = \text{const.}$$

$$\nabla \cdot \boldsymbol{E} = \frac{q}{\varepsilon}$$

with the electric displacement $\boldsymbol{D}$, the electric permittivity ε and the charge density q, which finally leads to

$$q_{\text{s}} = -\frac{\varepsilon}{\sigma}\nabla\sigma \cdot \boldsymbol{E}.$$

This equation describes the evolution of surface charges q_{s} in the interface between areas of different conductivity. These surface charges alter the potential field which results in a visible change in the measured sounding curves. That is why measuring potentials at the Earth's surface and/or in boreholes enables us to evaluate the underlying conductivity distribution. Generally, a configuration of four electrodes is used; one pair of electrodes (sources, A and B in Figure 2.1) transmits the current and the resulting potential field is measured with the second pair (receivers, M and N in Figure 2.1). Figure 2.2 shows several configurations of source and receiver electrodes and each has its own advantages and disadvantages with regards to exploration depth, signal-to-noise ratio, sensitivity and others (Møller et al. (2006), Knödel et al. (2005b)). For example Wenner arrays are less sensitive to noise whereas the dipole-dipole, pole-dipole and gradient arrays are most susceptible to noise, but offer the best resolution (Zhou and Dahlin (2003); Knödel et al. (2005b)). The exploration depth depends on the underlying conductivity structure as well as on the configuration and spacing of the electrodes. In general, this depth increases when the electrode spacing is enlarged. We call the depth range with the highest current density *depth of investigation* and it is possible to define approximate values for this depth for all configurations (Roy and Apparao, 1971). According to the theorem of reciprocity (Geselowitz, 1971) in a four-electrode system the positions of sources and receivers can be interchanged without affecting the results of the measurement. This property is used to examine the accuracy of the measurement or to enhance the signal-to-noise ratio using stacking (Zhou and Dahlin, 2003). The interpretation of DC resistivity measurements is limited by equivalences. For example in a layered medium different combinations of layer thicknesses and conductivities and even different numbers of layers can result in the same potential field (Flathe, 1974). Particularly in the case of an

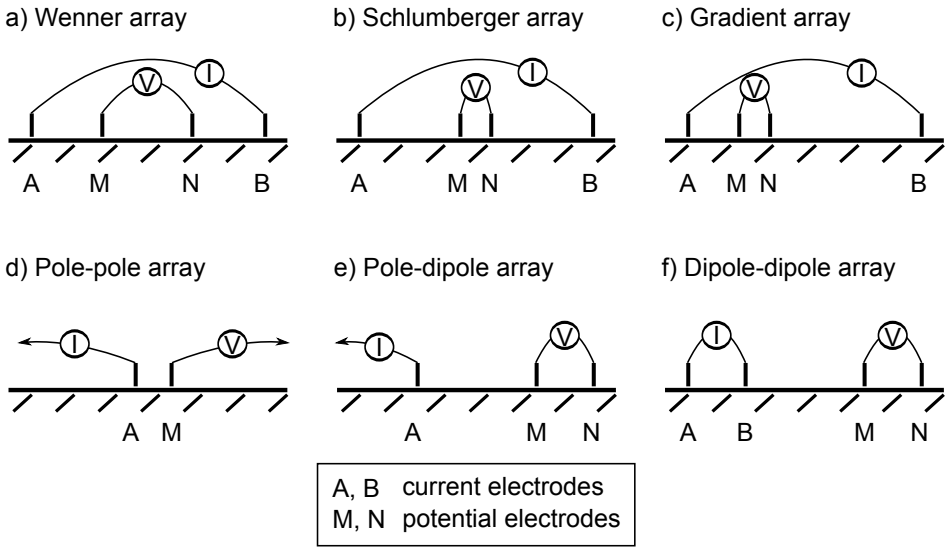

Fig. 2.2: Some of the most common configurations of electrodes, adapted from Møller et al. (2006). I denotes the injected current, whereas V denotes the potential difference measured between M and N.

embedded thin layer only the product or quotient of thickness h and conductivity σ can be determined. Here, the terms of longitudinal conductance and transverse resistance play an important role. If the thin layer is conductive in comparison with the background, the current is going to flow almost horizontally along the thin layer. For a resistive layer, the current will flow almost vertically through the layer. In both cases, the measurement result will not change if the longitudinal conductance $S = h\sigma$ or the transverse resistance $T = \frac{h}{\sigma}$ remains constant. For further explanation we refer to Zhdanov and Keller (1994).

A very important physical quantity in interpreting DC resistivity measurements is the apparent resistivity ρ_a, which can be explained as follows: The potential u of a point source of strength I located on the surface of a homogeneous halfspace is given by

$$u = \frac{\rho I}{2\pi r}$$

at an arbitrary spacing r (Keller and Frischknecht, 1966). For an array of two source electrodes A and B and two measuring electrodes M and N the potential of each source will be added to the resulting potential difference between M and N:

$$\Delta u(r) = \rho I \underbrace{\left[\frac{1}{2\pi}\left(\frac{1}{r_{\mathsf{AM}}} - \frac{1}{r_{\mathsf{BM}}} - \frac{1}{r_{\mathsf{AN}}} + \frac{1}{r_{\mathsf{BN}}}\right)\right]}_{k}.$$

We call this formula *Neumannsche Formel* (ibid.). Then, the resistivity of the homogeneous halfspace is given by

$$\rho = k\frac{\Delta u}{I} \tag{2.1}$$

with the geometric or configuration factor k, which depends on the position and spacing between source and receiver electrodes and therefore, its unit is a measure of length. We find the appropriate factors for a certain configuration e. g. in Roy and Apparao (1971) and Knödel et al. (2005b). Equation (2.1) represents the resistivity for a homogeneous earth but for an inhomogeneous conductivity distribution the values of ρ may differ a lot or even have no relation to the actual resistivities because they are calculated under the assumption of homogeneity. Hence, we call these values *apparent resistivities* ρ_a.

2.2 DC resistivity forward modeling

2.2.1 Governing equations and boundary conditions

If we want to study the propagation of the electric currents and the evolution of a potential field during a DC resistivity measurement we have to evaluate the governing equations using numerical forward modeling. Generally, the forward problem is a nonlinear mapping which generates synthetic data $\boldsymbol{u}$ by applying an operator G to a given parameter distribution $\boldsymbol{m}$:

$$G(\boldsymbol{m}) = \boldsymbol{u}.$$

G implements the laws of physics and processes which are relevant for the given physical problem and usually, it is the discrete representation of a partial differential equation (PDE). To model DC resistivity measurements, we first consider the equation of continuity:

$$-\nabla \cdot (\sigma \nabla u) = I\delta(\boldsymbol{x} - \boldsymbol{x}_0), \tag{2.2}$$

with the potential u, a point source of strength I located at $\boldsymbol{x}_0 = [x_0, y_0, z_0]^T$ on the right-hand side of the equation and a given conductivity distribution $\sigma = \sigma(x, y, z)$. For a homogeneous halfspace the solution of this equation is given by Green's function in two and three dimensions (Keller and Frischknecht (1966) and Zhdanov and Keller (1994)):

$$u(\boldsymbol{x}) = \begin{cases} -\dfrac{I}{\pi\sigma} \log r, \text{ (2D)} & (2.3) \\ \dfrac{I}{2\pi\sigma r}, \quad \text{(3D)} & (2.4) \end{cases}$$

where $r = ||\boldsymbol{x} - \boldsymbol{x}_0||$ is the distance between an arbitrary point $\boldsymbol{x}$ in the modeling domain and the source position $\boldsymbol{x}_0$. Although the electric fields and currents spread in an infinite medium, to find a discrete representation and solve equation (2.2) we have to limit the medium to a small computational domain Ω with appropriate boundary conditions on $\partial\Omega$ which satisfy the

laws of physics. These conditions are essential for a unique solution of the PDE. Depending on the physical phenomenon we want to simulate numerically, we can set Dirichlet or Neumann conditions, which determine the field value on the boundary or the normal derivative of the considered field, respectively. These conditions yield a sufficiently good approximation although the physical field extends beyond the boundary of the modeling domain. A mixed or Robin boundary condition on $\partial\Omega$ includes a mapping of Dirichlet values onto Neumann values and is known as Poincaré-Steklov operator (Oberai et al., 1998) or Dirichlet-to-Neumann (DtN) operator. It maps the values of the electric potential u on the boundary $\partial\Omega$ onto the normal derivative of the potential $\partial_n u$ on $\partial\Omega$. For the two-dimensional (2D) DC resistivity problem, we derive the DtN mapping of Green's function (2.3): With the gradient of the potential

$$\nabla u(\boldsymbol{x}) = \begin{bmatrix} \partial_x u \\ \partial_y u \\ \partial_z u \end{bmatrix} = -\frac{I}{\pi\sigma r} \begin{bmatrix} \partial_x r \\ \partial_y r \\ \partial_z r \end{bmatrix} = -\frac{I}{\pi\sigma r^2} \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} = \frac{\boldsymbol{x} - \boldsymbol{x}_0}{r^2 \log r} u(\boldsymbol{x})$$

we obtain the normal derivative with the unit normal vector $\boldsymbol{n}$ as

$$\partial_n u(\boldsymbol{x}) = \boldsymbol{n} \cdot \nabla u(\boldsymbol{x}) = \boldsymbol{n} \cdot \frac{(\boldsymbol{x} - \boldsymbol{x}_0)}{r^2 \log r}\, u(\boldsymbol{x}),$$

and get the mixed boundary condition

$$\partial_n u - \boldsymbol{n} \cdot \frac{(\boldsymbol{x} - \boldsymbol{x}_0)}{r^2 \log r} u = 0. \tag{2.5}$$

For the 3D case we get the DtN mapping as follows: The gradient of the potential is given by

$$\nabla u(\boldsymbol{x}) = \begin{bmatrix} \partial_x u \\ \partial_y u \\ \partial_z u \end{bmatrix} = -\frac{I}{2\pi\sigma r^2} \begin{bmatrix} \partial_x r \\ \partial_y r \\ \partial_z r \end{bmatrix} = -\frac{I}{2\pi\sigma r^3} \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} = -\frac{\boldsymbol{x} - \boldsymbol{x}_0}{r^2} u(\boldsymbol{x})$$

and the normal derivative reads

$$\partial_n u(\boldsymbol{x}) = \boldsymbol{n} \cdot \nabla u(\boldsymbol{x}) = -\boldsymbol{n} \cdot \frac{(\boldsymbol{x} - \boldsymbol{x}_0)}{r^2}\, u(\boldsymbol{x})$$

resulting in the exact local boundary condition with a different sign compared to the 2D DtN mapping:

$$\partial_n u + \boldsymbol{n} \cdot \frac{(\boldsymbol{x} - \boldsymbol{x}_0)}{r^2} u = 0. \tag{2.6}$$

For the boundary located at the Earth's surface, denoted by $\Gamma_0 \subset \partial\Omega$, we consider homogeneous Neumann conditions because the conductivity of the air is equal to zero and therefore the electric field is tangential to the boundary:

$$\partial_n u \equiv 0 \quad \text{along } \Gamma_0. \tag{2.7}$$

2.2.2 The secondary field approach

In the previous section, we introduced the fundamental equation for DC resistivity modeling:

$$-\nabla \cdot (\sigma \nabla u) = I\delta(\boldsymbol{x} - \boldsymbol{x}_0). \tag{2.8}$$

Discretizing and solving this equation to obtain the electric potential u for a given conductivity distribution is called *total field approach* for the DC resistivity problem. The slope of the dirac delta distribution on the right-hand side of the equation is infinite at the source location and therefore a singularity in the potential field u occurs at $\boldsymbol{x}_0$. This singularity restricts the accuracy of the finite element approximation and results in a deduced convergence rate. We are not able to discretize this discontinuous function adequately but the usage of very fine grids can weaken the effect of the singularity. Unfortunately, this will increase the computational effort rapidly (Lowry et al., 1989). To obtain the full rate of convergence and utilize a moderate grid size, we use the *secondary field approach* described by Coggon (1971) and Lowry et al. (1989) and we develop the corresponding PDE as follows: For a single point source of strength I located at $\boldsymbol{x}_0$ on the surface of a 3D homogeneous halfspace with a constant conductivity σ_0 equation (2.8) changes to

$$-\nabla \cdot \left(\sigma_0 \nabla u_\mathsf{p}\right) = I\delta(\boldsymbol{x} - \boldsymbol{x}_0) \tag{2.9}$$

with the solution for the primary potential u_p given by Green's function (equation (2.4)):

$$u_\mathsf{p}(\boldsymbol{x}) = \frac{I}{2\pi\sigma_0 r} \tag{2.10}$$

at an arbitrary point $\boldsymbol{x}$ and $r = ||\boldsymbol{x} - \boldsymbol{x}_0||$. Now, we assume an anomalous conductivity $\sigma_\mathsf{s} = \sigma(\boldsymbol{x}) - \sigma_0$ and decompose the total potential u into primary (u_p) and secondary (u_s) potential:

$$u = u_\mathsf{p} + u_\mathsf{s}$$

Using this decomposition and inserting it into equation (2.8) we get

$$\begin{aligned} -\nabla \cdot \left(\sigma \nabla (u_\mathsf{p} + u_\mathsf{s})\right) &= I\delta(\boldsymbol{x} - \boldsymbol{x}_0) \\ -\nabla \cdot \left(\sigma \nabla u_\mathsf{p}\right) - \nabla \cdot (\sigma \nabla u_\mathsf{s}) &= I\delta(\boldsymbol{x} - \boldsymbol{x}_0). \end{aligned}$$

The right-hand side is replaced by equation (2.9):

$$\nabla \cdot \left(\sigma \nabla u_\mathsf{p}\right) + \nabla \cdot (\sigma \nabla u_\mathsf{s}) = \nabla \cdot (\sigma_0 \nabla u_\mathsf{p}),$$

which finally yields the fundamental equation for the secondary field approach:

$$\begin{aligned}\nabla \cdot (\sigma \nabla u_s) &= \nabla \cdot \left(\sigma_0 \nabla u_p\right) - \nabla \cdot \left(\sigma \nabla u_p\right) \\ &= \nabla \cdot \left((\sigma_0 - \sigma) \nabla u_p\right) \\ -\nabla \cdot (\sigma \nabla u_s) &= \nabla \cdot (\underbrace{(\sigma - \sigma_0)}_{\sigma_s} \nabla u_p). \end{aligned} \tag{2.11}$$

To model the Earth's surface in a realistic way we have to be able to work with topography which means to use correct boundary conditions for a surface which is far away from being a "flat Earth". For the total potential u, we use the homogeneous Neumann condition given in equation (2.7). To fulfill this condition for the secondary potential u_s we have to implement the following inhomogeneous Neumann boundary condition along Γ_0:

$$\begin{aligned} u_s &= u - u_p \\ \frac{\partial u_s}{\partial n} &= \cancelto{0}{\frac{\partial u}{\partial n}} - \frac{\partial u_p}{\partial n} \\ &= -\frac{\partial u_p}{\partial n} \\ &= -\boldsymbol{n} \cdot \nabla u_p. \end{aligned}$$

We already calculated the gradient of the primary potential for the mixed boundary condition given in equations (2.6) and (2.5). Hence, for the normal derivative of u_s we get two different expressions for the 2D- and 3D-case:

$$\frac{\partial u_s}{\partial n} = \begin{cases} -\boldsymbol{n} \cdot \dfrac{(\boldsymbol{x} - \boldsymbol{x}_0)}{r^2 \log r}\, u_p^{\mathsf{2D}}, & \text{(2D)}, \\ \boldsymbol{n} \cdot \dfrac{(\boldsymbol{x} - \boldsymbol{x}_0)}{r^2}\, u_p^{\mathsf{3D}}, & \text{(3D)}. \end{cases} \tag{2.12}$$

These derivatives yield the correct potential fields for modeling the secondary potential in a realistic Earth model containing topography at the air-Earth interface.

2.2.3 Finite element approximation – Lagrange elements

We are able to develop analytical solutions of PDEs for some simple models. But dealing with complex and realistic model geometries requires us to use approximation procedures to get a solution. There are a lot of different numerical methods to find this solution such as finite element, finite difference, finite volume, or spectral methods. Each of these methods has its own advantages and disadvantages. To solve the DC resistivity problem described in the

previous section, we chose to use the finite element method. It uses an appropriate variational formulation for each PDE and provides large flexibility concerning surface topography and complex geological models.

In the following chapter, we give a short introduction to our finite element discretization and the assembly of the resulting linear system of equations. For further details and explicit derivations see for example Zienkiewicz et al. (2005), Monk (2003) or Braess (2003). We want to remark that almost all detailed explanations concerning basis functions, size of variables etc. are developed in 3D but the DC resistivity code can be used for 2D problems, too (see Appendix A.1.1).

2.2.3.1 Variational formulation

Again, we consider the equation of continuity given in (2.2):

$$-\nabla \cdot (\sigma \nabla u) = f, \tag{2.13}$$

with the right-hand side or source term f and the following boundary conditions:

$$u = g_D \text{ along } \Gamma_D \subset \partial\Omega \tag{2.14}$$

$$\frac{\partial u}{\partial n} = g_N \text{ along } \Gamma_N \subset \partial\Omega \tag{2.15}$$

with the Dirichlet boundary Γ_D, the Neumann boundary Γ_N and $\partial\Omega = \Gamma_D \cup \Gamma_N$. To deduce the basis for a finite element discretization, which is the variational formulation, we need to use the multidimensional equivalent of integration by parts which is the divergence theorem:

Theorem 1 (*Divergence Theorem, see e.g. Monk (2003)*)

Let $\Omega \subset \mathbb{R}^3$, with boundary $\partial\Omega$ and unit outward normal $\boldsymbol{n}$, be a bounded Lipschitz domain. Let $\boldsymbol{u} : \mathbb{R}^3 \to \mathbb{R}^3$ be a vector field with $u^i \in H^1(\Omega)$, $i = 1, 2, 3$. Then

$$\int_\Omega \nabla \cdot \boldsymbol{u}\, d\boldsymbol{x} = \int_\Omega (u^1_{x_1} + u^2_{x_2} + u^3_{x_3})\, d\boldsymbol{x} = \int_{\partial\Omega} \boldsymbol{u} \cdot \boldsymbol{n}\, ds,$$

where $\boldsymbol{u} = [u^1, u^2, u^3]^T$ and ds stands for integration over the boundary $\partial\Omega$.

The partial derivatives $u^i_{x_i}$ are weak derivatives and can be described as follows: If there is a continuous partial derivative u_x of the function u with respect to x, then, for every differentiable function ϕ which vanishes on $\partial\Omega$ and $u^1 = u\phi$, $u^2 = u^3 = 0$ there holds

$$\int_\Omega u\phi_x\, d\boldsymbol{x} = -\int_\Omega u_x \phi\, d\boldsymbol{x}, \quad \forall \phi \in C_0^\infty \tag{2.16}$$

according to the divergence theorem. This equation can apply to non-differentiable functions, too: If u and v are integrable functions with the property

$$\int_\Omega u\phi_x \, d\boldsymbol{x} = -\int_\Omega v\phi \, d\boldsymbol{x},$$

then v is the weak derivative of u with respect to x. Using these weak formulations, we deduce the variational formulation for the DC resistivity problem by multiplying (2.13) by a test function v, integrating and applying the divergence theorem:

$$\begin{aligned}(f,v) &= -\int_\Omega v\nabla\cdot(\sigma\nabla u)\, d\boldsymbol{x} \\ &= -\int_\Omega (\nabla\cdot(\sigma v\nabla u) - \nabla u\cdot\sigma\nabla v)\, d\boldsymbol{x} \\ &= \int_\Omega \nabla u\cdot\sigma\nabla v\, d\boldsymbol{x} - \int_{\partial\Omega}\sigma v\nabla u\cdot\boldsymbol{n}\, ds \\ &= \int_\Omega \nabla u\cdot\sigma\nabla v\, d\boldsymbol{x} - \int_{\partial\Omega}\sigma v\frac{\partial u}{\partial n}\, ds,\end{aligned}$$

where $(\cdot,\cdot)$ is the L^2 inner product for vector functions in Ω. Now, we choose a test function v which vanishes on Γ_D. With the Neumann condition given in equation (2.15) and the L^2 inner product $(\cdot,\cdot)_\Gamma$ on Γ we get

$$(\nabla u,\sigma\nabla v) = (f,v) + (\sigma\partial_n u, v)_{\Gamma_N} = (f,v) + (\sigma g_N, v)_{\Gamma_N}$$

with the (bi)linear forms

$$a(u,v) := (\nabla u,\sigma\nabla v) = \int_\Omega \nabla u\cdot\sigma\nabla v\, d\boldsymbol{x}, \tag{2.17}$$

$$l(v) := (f,v) + (\sigma g_N, v)_{\Gamma_N} = \int_\Omega fv\, d\boldsymbol{x} + \int_{\Gamma_N}\sigma g_N v\, ds. \tag{2.18}$$

Finally, with the functions $u, v \in H^1(\Omega)$ and the ansatz and test spaces

$$\mathscr{S} = \left\{u\in H^1(\Omega) : u_{|\Gamma_D} = g_D\right\} \tag{2.19}$$

$$\mathscr{V} = \left\{v\in H^1(\Omega) : v_{|\Gamma_D} = 0\right\} \tag{2.20}$$

we get the variational formulation for the DC resistivity problem given in equation (2.13):

$$\text{Find } u\in\mathscr{S} \text{ such that } a(u,v) = l(v) \quad \forall v\in\mathscr{V}. \tag{2.21}$$

The DC resistivity code is able to work with different approaches for the right-hand side of the PDE discussed in Chapter 2.2.2. For the total field approach the first integral of equation (2.18) is given by

$$\int_\Omega fv\, d\boldsymbol{x} = Iv(\boldsymbol{x}_0)$$

and for the secondary field approach we get

$$\int_\Omega fv\, d\boldsymbol{x} = \int_\Omega \nabla v\cdot(\sigma-\sigma_0)\nabla u_\mathrm{p}\, d\boldsymbol{x} \tag{2.22}$$

which is almost equal to the integral of the bilinear form $a(u,v)$ as given in equation (2.17).

2.2.3.2 Derivation of the Galerkin system

The variational formulation (2.21) enables us to assemble a linear system of equations to solve the DC resistivity problem. The finite element method uses ansatz and test spaces for the ansatz and test functions u and v consisting of piecewise polynomials and the underlying modeling area Ω is divided into subdomains which are for example triangles in two dimensions or tetrahedra, hexahedrons and others in three dimensions. We call the decomposition of Ω into subsets $K \subset \Omega$ triangulation $\mathscr{T}_h$ (Figure 2.3). First of all, the ansatz and test spaces

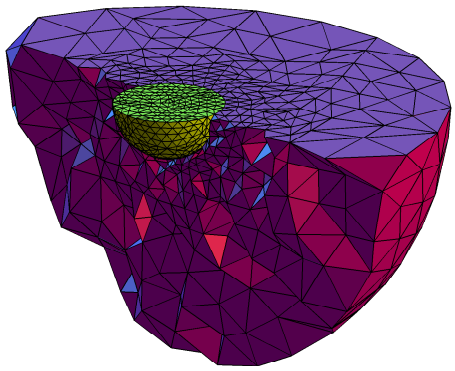

Fig. 2.3: Vertical cut through an exemplary tetrahedral triangulation $\mathscr{T}_h$ of two concentric hemispheres.

given in equations (2.19) and (2.20) are substituted with finite spaces $\mathscr{S}^h$ and $\mathscr{V}^h$. These two spaces are equal except for the boundary conditions and in particular for $g_D \equiv 0$ there holds $\mathscr{S}^h = \mathscr{V}^h$. Therefore, we only consider basis functions of the n-dimensional subspace $\mathscr{V}^h \subset \mathscr{V}$ to describe the solution u^h. The variational formulation (2.21) changes as follows:

$$\text{Find } u^h \in \mathscr{V}^h \text{ such that } a(u^h, v) = l(v) \quad \forall v \in \mathscr{V}^h. \tag{2.23}$$

With a basis of $\mathscr{V}^h$ $\{\phi_1, \phi_2, \ldots, \phi_n\}$ we get the solution u^h by calculating the linear combination of these basis functions:

$$u^h = \sum_{j=1}^{n} u_j \phi_j, \tag{2.24}$$

where u_j are unknown scalar coefficients or degrees of freedom (DOFs). Our code deals with Lagrange elements which are scalar finite elements. Here, we consider linear elements where the DOFs are defined on vertices (Figure 2.4). All linear basis functions $\phi_j \in \mathscr{V}^h$ are uniquely defined by their function values at the vertices $\boldsymbol{x}_i$ of the elements. We use an appropriate nodal basis $\{\phi_1, \phi_2, \ldots, \phi_n\}$, given by

$$\phi_j(\boldsymbol{x}_i) = \delta_{i,j}, \quad i, j = 1, \ldots, n, \tag{2.25}$$

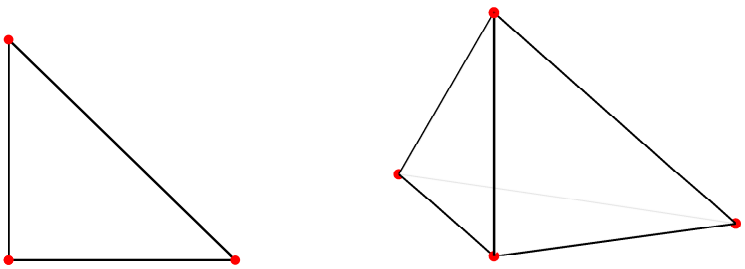

Fig. 2.4: DOFs (red) for linear Lagrange elements in two (left) and three (right) dimensions.

which means that each basis function ϕ_j is equal to one at the associated vertex $\boldsymbol{x}_j$ and vanishes for all other vertices. With the linear combination (2.24) and a test function $v = \phi_i$, the variational formulation (2.23) is equivalent to

$$\sum_{j=1}^{n} u_j a(\phi_j, \phi_i) = l(\phi_i), \quad i = 1, 2, \ldots, n.$$

With the matrix $A \in \mathbb{R}^{n \times n}$ for $[A]_{i,j} = a(\phi_j, \phi_i)$, the right-hand side $\boldsymbol{b} \in \mathbb{R}^n$ for $[b]_i = l(\phi_i)$, the solution $\boldsymbol{u} \in \mathbb{R}^n$ for $[u]_i = u_i$ and the conductivity vector $\boldsymbol{\sigma}$, which contains one conductivity value for each element K, we get the Galerkin system (Zienkiewicz et al., 2005):

$$A(\boldsymbol{\sigma})\boldsymbol{u} = \boldsymbol{b}. \tag{2.26}$$

Because of the nodal basis, the solution $\boldsymbol{u}$, which contains the coefficients (DOFs) of the linear combination (2.24), directly represents the approximated electric potential on each of the nodes in the triangulation $\mathcal{T}_h$. Therefore, we use the vector $\boldsymbol{u}$ for the solution of the forward problem as well as for the electric potential from now on. Furthermore, the equations (2.11) and (2.22) yield the discrete representation of the secondary field equation (Wolters et al. (2007), Rücker et al. (2006)):

$$A(\boldsymbol{\sigma})\boldsymbol{u}_s = A(\boldsymbol{\sigma}_0 - \boldsymbol{\sigma})\boldsymbol{u}_p, \tag{2.27}$$

where $\boldsymbol{\sigma}_0 = \sigma_0 \cdot [1, 1, \ldots, 1, 1]^T$ is a constant vector containing the source conductivity. The sum of the solution $\boldsymbol{u}_s$ (secondary potential) and the primary potential $\boldsymbol{u}_p$ (equation (2.10)) yields the total potential $\boldsymbol{u}$ and therefore an approximated solution of the underlying PDE (2.11).

2.2.3.3 Assembly of the system matrix

To assemble the Galerkin system we have to reformulate the integrals given in equations (2.17) and (2.18) in terms of the basis functions of the finite element space. For each element K of the triangulation $\mathscr{T}^h$ and $i, j = 1, 2, \ldots, n$ it holds

$$\begin{aligned} a(\phi_j, \phi_i) &= \int_\Omega \nabla\phi_j \cdot \sigma(\boldsymbol{x}) \nabla\phi_i \, d\boldsymbol{x} \\ &= \sum_{K \in \mathscr{T}^h} \sigma_K \int_K \nabla\phi_j \cdot \nabla\phi_i \, d\boldsymbol{x} =: \sum_{K \in \mathscr{T}^h} a_K(\phi_j, \phi_i) \end{aligned} \tag{2.28}$$

$$\begin{aligned} l(\phi_i) &= \int_\Omega f\phi_i \, d\boldsymbol{x} + \int_{\Gamma_N} \sigma(\boldsymbol{x}) g_N \phi_i \, ds \\ &= \sum_{K \in \mathscr{T}^h} \left(\int_K f\phi_i \, d\boldsymbol{x} + \sigma_K \int_{K \cap \Gamma_N} g_N \phi_i ds \right) =: \sum_{K \in \mathscr{T}^h} l_K(\phi_i). \end{aligned} \tag{2.29}$$

We assume a piecewise constant conductivity distribution $\sigma(\boldsymbol{x})$ which means that each element K is associated with a constant conductivity σ_K and the element integrals given in equations (2.28) and (2.29) do not depend on sigma anymore. With the element matrix and right-hand side

$$\begin{aligned} [A_K]_{i,j} &:= a_K(\phi_j, \phi_i) \quad i, j = 1, 2, \ldots, n \\ [\boldsymbol{b}_K]_i &:= l_K(\phi_i) \quad i = 1, 2, \ldots, n, \end{aligned}$$

which are defined in a single element, we get the full system matrix and right-hand side

$$A = \sum_{K \in \mathscr{T}^h} A_K \text{ and } \boldsymbol{b} = \sum_{K \in \mathscr{T}^h} \boldsymbol{b}_K.$$

Usually, for a finite element method we define a finite element on the reference element $\hat{K}$ and derive an arbitrary element $K \in \mathscr{T}^h$ using an affine mapping from $\hat{K}$ onto K. One reason for this approach is that the derivation of basis functions and other essential quantities is much easier for the reference element and has to be done only once. From now on, $\hat{K}$ itself and all other quantities associated with the reference element will be marked with a hat in this thesis. We set the reference element to be the unit tetrahedron $\hat{K}$ (Figure 2.5) with the coordinates $\hat{\boldsymbol{x}} = (\xi, \eta, \nu) \in \hat{K}$ in 3D (unit triangle in 2D) and define it as follows:

$$\hat{K} = \{(\xi, \eta, \nu) \in \mathbb{R}^3 \, : \, 0 \leq \xi \leq 1, \, 0 \leq \eta \leq 1 - \xi, \, 0 \leq \nu \leq 1 - \xi - \eta\}, \tag{2.30}$$

with the vertices $\hat{\boldsymbol{v}}_1 = [1, 0, 0]^T$, $\hat{\boldsymbol{v}}_2 = [0, 1, 0]^T$, $\hat{\boldsymbol{v}}_3 = [0, 0, 1]^T$ and $\hat{\boldsymbol{v}}_4 = [0, 0, 0]^T$. We want to use a transformation F_K from $\hat{K}$ onto K to get the element matrix A_K for an arbitrary element. This transformation is given as follows:

$$F_K \, : \, \hat{K} \to K, \quad \hat{K} \ni \hat{\boldsymbol{x}} \mapsto \boldsymbol{x} \in K, \quad \boldsymbol{x} = F_K(\hat{\boldsymbol{x}}) = B_K \hat{\boldsymbol{x}} + \boldsymbol{b}_K \tag{2.31}$$

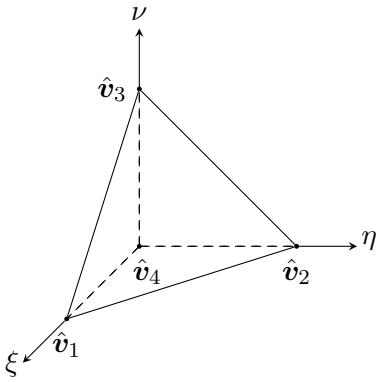

Fig. 2.5: The reference tetrahedron $\hat{K}$.

with

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \underbrace{\begin{bmatrix} x_1 - x_4 & x_2 - x_4 & x_3 - x_4 \\ y_1 - y_4 & y_2 - y_4 & y_3 - y_4 \\ z_1 - z_4 & z_2 - z_4 & z_3 - z_4 \end{bmatrix}}_{B_K} \begin{bmatrix} \xi \\ \eta \\ \nu \end{bmatrix} + \underbrace{\begin{bmatrix} x_4 \\ y_4 \\ z_4 \end{bmatrix}}_{\boldsymbol{b}_K}. \tag{2.32}$$

Then, the affine mapping F_K transforms each vertex of the reference tetrahedron $\hat{K}$ onto an arbitrary tetrahedron K:

$$\begin{aligned} \hat{\boldsymbol{v}}_1 &= [1,0,0]^T \mapsto [x_1, y_1, z_1]^T, \\ \hat{\boldsymbol{v}}_2 &= [0,1,0]^T \mapsto [x_2, y_2, z_2]^T, \\ \hat{\boldsymbol{v}}_3 &= [0,0,1]^T \mapsto [x_3, y_3, z_3]^T, \\ \hat{\boldsymbol{v}}_4 &= [0,0,0]^T \mapsto [x_4, y_4, z_4]^T. \end{aligned}$$

To evaluate the (bi)linear forms (2.28) and (2.29), we have to evaluate the basis functions for an arbitrary element. This can be realized by transforming the basis functions defined in the reference element to the respective element. The local 3D basis functions in the reference element $\hat{K}$ are given by

$$\hat{\phi}_1(\xi,\eta,\nu) = \xi, \quad \hat{\phi}_2(\xi,\eta,\nu) = \eta, \quad \hat{\phi}_3(\xi,\eta,\nu) = \nu, \quad \text{and} \quad \hat{\phi}_4(\xi,\eta,\nu) = 1 - \xi - \eta - \nu$$

and their gradients are

$$\begin{bmatrix} \hat{\nabla}\phi_1 & \hat{\nabla}\phi_2 & \hat{\nabla}\phi_3 & \hat{\nabla}\phi_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}.$$

With equation (2.31) there holds

$$\begin{aligned} \phi(\boldsymbol{x}) &:= \hat{\phi}(\hat{\boldsymbol{x}}) = \hat{\phi}(F_K^{-1}(\boldsymbol{x})) \\ \nabla\phi &= B_K^{-T}\hat{\nabla}\hat{\phi} \\ \hat{\nabla}\hat{\phi} &= \left[\partial_\xi\hat{\phi},\, \partial_\eta\hat{\phi},\, \partial_\nu\hat{\phi}\right]^T. \end{aligned} \tag{2.33}$$

Now, we can reformulate the (bi)linear forms and integrals given in equations (2.28) and (2.29) in terms of the reference element (without boundary conditions):

$$\begin{aligned} a_K(\phi_j, \phi_i) &= \sigma_K \int_{\hat{K}} \left(B_K^{-T}\hat{\nabla}\hat{\phi}_j\right) \cdot \left(B_K^{-T}\hat{\nabla}\hat{\phi}_i\right) |\mathrm{det}B_K|\, d\hat{\boldsymbol{x}} \\ l(\phi_i) &= \int_{\hat{K}} f\hat{\phi}_i |\mathrm{det}B_K|\, d\hat{\boldsymbol{x}}. \end{aligned}$$

Within our assembly routines, we use a Gaussian quadrature rule to approximate the integrals and assemble the matrix A_K and the right-hand side $\boldsymbol{b}_K$ for the element K.

Pseudocode 1 shows the principle of the assembly of the full stiffness matrix A and the right-hand side $\boldsymbol{b}$. We use the notation of MATLAB where $A(i,j)$ denotes the entry of A in the ith row and the jth column. Within our DC resistivity code the coordinates of all nodes of the triangulation $\mathscr{T}^h$ are given in an array `mesh.nodes` of size $3 \times n_{\text{nodes}}$ in 3D. The four nodes a tetrahedron K consists of are given in the array `mesh.elem` of size $4 \times n_K$.

Pseudocode 1 ASSEMBLY OF THE GALERKIN SYSTEM

1: $A = O$, $\boldsymbol{b} = \boldsymbol{0}$
2: **for** $i = 1, 2, \ldots, n_K$ {i: index of element K} **do**
3: Extract nodes
$i_k = \texttt{mesh.elem}(k,\, i),\ k = 1, 2, 3, 4$
4: Calculate A_K and b_K for the element K
5: $A([i_1\ i_2\ i_3\ i_4],\, [i_1\ i_2\ i_3\ i_4]) = A([i_1\ i_2\ i_3\ i_4],\, [i_1\ i_2\ i_3\ i_4]) + A_K$
6: $\boldsymbol{b}([i_1\ i_2\ i_3\ i_4]) = \boldsymbol{b}([i_1\ i_2\ i_3\ i_4]) + \boldsymbol{b}_K$
7: **end for**

Besides refining the mesh, it could be useful to increase the order of the basis functions to improve the finite element approximation. The improvement in the solution depends on the underlying problem and the PDE we have to discretize. In this chapter we just stated the basis functions and DOFs for linear Lagrange elements, but the DC resistivity code is able to use quadratic Lagrange elements in 2D and 3D, too. Therefore, we give the basis functions and DOFs for quadratic Lagrange elements in Appendix A.1.2. The assembly procedure for elements of higher order is similar to that for linear elements.

2.2.3.4 Implementing boundary conditions

The previous section explained the assembly of the stiffness matrix without considering the boundary conditions. In the following, we give a short overview over the three main types of boundary conditions we implemented for our DC resistivity code. In the linear form (2.29) the boundary integral for all elements is given as

$$\int_{\Gamma_N} \sigma(\boldsymbol{x}) g_N \phi_i \, ds = \sum_{K \in \mathscr{T}^h} \left(\sigma_K \int_{K \cap \Gamma_N} g_N \phi_i ds \right).$$

To implement *Neumann* boundary conditions we have to transform the boundary integral to the reference element. Because we want to evaluate this integral on the boundary of the modeling domain, we have to compute a rectangular transformation matrix B_K^{r}. This matrix represents a projection from the original 3D boundary of the modeling domain Ω, consisting of the triangular faces of the boundary tetrahedra, onto the reference triangle $\hat{K}$ which is a 2D object. This results in

$$\sigma_K \int_{K \cap \Gamma_N} g_N \phi_i \, ds = \sigma_K \int_{\hat{K}} g_N \hat{\phi}_i \sqrt{|\det(B_K^{\mathrm{r}T} \cdot B_K^{\mathrm{r}})|} \, d\hat{s}.$$

For homogeneous Neumann conditions, this integral vanishes. If $g_N \neq 0$, we have to add the resulting vector to the right-hand side of the linear system of equations. An inhomogeneous Neumann condition is needed to implement surface topography as described in Section 2.2.1 and in particular equation (2.12).

As given in Section 2.2.1 we want to be able to use *Robin* boundary conditions for all boundaries except the Earth's surface. The associated equation (2.6) was given by

$$g_R = \partial_n u = -\frac{\boldsymbol{n} \cdot (\boldsymbol{x} - \boldsymbol{x}_0)}{r^2} u.$$

Then, for the variational formulation (2.23) we get

$$\begin{aligned} a(u,v) &= l(v) \\ &= \int_\Omega f v \, d\boldsymbol{x} + \int_{\Gamma_N} \sigma g_N v \, ds + \int_{\Gamma_R} \sigma g_R v \, ds \\ \underbrace{a(u,v)}_{\text{stiffness matrix}} + \underbrace{\int_{\Gamma_R} \sigma \frac{\boldsymbol{n} \cdot (\boldsymbol{x} - \boldsymbol{x}_0)}{r^2} uv \, ds}_{\text{Robin contribution to the stiffness matrix}} &= \underbrace{\int_\Omega f v \, d\boldsymbol{x}}_{\text{source term}} + \underbrace{\int_{\Gamma_N} \sigma g_N v \, ds}_{\text{Neumann condition}}. \end{aligned}$$

In terms of the reference element the contribution of the Robin condition to the stiffness matrix is given by

$$\sigma_K \int_{K \cap \Gamma_R} \frac{\boldsymbol{n} \cdot (\boldsymbol{x} - \boldsymbol{x}_0)}{r^2} \phi_i \phi_j \, ds = \sigma_K \int_{\hat{K}} \frac{\boldsymbol{n} \cdot (\boldsymbol{x} - \boldsymbol{x}_0)}{r^2} \hat{\phi}_i \hat{\phi}_j \sqrt{|\det(B_K^{\mathrm{r}T} \cdot B_K^{\mathrm{r}})|} \, d\hat{s}.$$

Again, we need the rectangular transformation matrix B_K^{r}, because we are integrating over the boundary triangles. In contrast to the Neumann condition, the mixed or Robin boundary condition contributes to the stiffness matrix A.

Dirichlet boundaries can be used, if the exact solution on a certain boundary is known. They are not considered in the initial assembly process, but incorporated before solving the system of equations. First, we decompose the DOFs $\boldsymbol{u} \in \mathbb{R}^n$ into inner nodes $\boldsymbol{u}_I$ and Dirichlet boundary nodes $\boldsymbol{u}_D$ with

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_I \\ \boldsymbol{u}_D \end{bmatrix}.$$

Then, the Galerkin system (2.26) can be decomposed as follows:

$$\begin{bmatrix} A_{II} & A_{ID} \\ A_{DI} & A_{DD} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_I \\ \boldsymbol{u}_D \end{bmatrix} = \begin{bmatrix} \boldsymbol{b}_I \\ \boldsymbol{b}_D \end{bmatrix}.$$

The four matrices describe the coupling of the inner DOFs and the Dirichlet nodes with each other and with themselves. With the Dirichlet condition

$$\boldsymbol{u}_D = \boldsymbol{g}_D$$

given in equation (2.14) we are able to separate the unknown inner DOFs from the known Dirichlet values by

$$\begin{bmatrix} A_{II} & A_{ID} \\ O & I \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_I \\ \boldsymbol{u}_D \end{bmatrix} = \begin{bmatrix} \boldsymbol{b}_I \\ \boldsymbol{g}_D \end{bmatrix}$$

and we get the smaller linear system of equations

$$A_{II}\boldsymbol{u}_I = \boldsymbol{b}_I - A_{ID}\boldsymbol{g}_D. \tag{2.34}$$

Now we are able to solve this new, reduced system of equations for $\boldsymbol{u}_I$ and afterwards we combine these values with the known Dirichlet values $\boldsymbol{u}_D$ to get the full solution vector $\boldsymbol{u} = [\boldsymbol{u}_I, \boldsymbol{u}_D]^T$.

In the DC resistivity code, the implementation of the boundary conditions can be found in the functions `fe_get_rhs_bc`, `fe_get_stiffness_robin` and `fe_solve` (see Appendix A.5.1.3). Unless noted otherwise, we use a Neumann condition at the Earth's surface and an exact Robin boundary condition in all following forward simulations.

2.2.3.5 Decomposition of the stiffness matrix

To avoid reassembling the system matrix $A(\boldsymbol{\sigma})$ for a new conductivity vector and simplify the derivative of this matrix, it is beneficial to separate the parameter information from the mesh information. Within this section we want to show two different possible ways to decompose $A(\boldsymbol{\sigma})$.

The approach we are using within our DC resistivity code is the following: We do not assemble the full stiffness matrix $A(\boldsymbol{\sigma})$, but a three-way tensor

$$A'(\boldsymbol{\sigma}) = \frac{\partial A(\boldsymbol{\sigma})}{\partial \boldsymbol{\sigma}},$$

where the number of "slices" equals the number of parameters in $\boldsymbol{\sigma}$ and each slice $\frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}$ contains the derivative of the matrix A with respect to σ_i. For example the first slice belongs to the partial derivative with respect to σ_1 (Figure 2.6). The system matrix $A(\boldsymbol{\sigma})$ depends linearly on each σ_i. The tensor is very sparse and independent of the parameters. To obtain

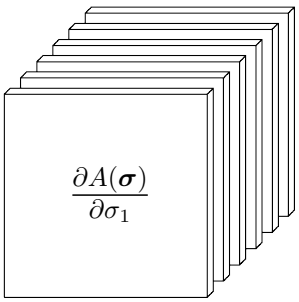

Fig. 2.6: Derivative of the system matrix $A(\boldsymbol{\sigma})$ with respect to all σ_i.

the system matrix, we only have to multiply the tensor by the conductivity vector in the third dimension:

$$A(\boldsymbol{\sigma}) = \frac{\partial A(\boldsymbol{\sigma})}{\partial \boldsymbol{\sigma}} \times_3 \boldsymbol{\sigma}.$$

Therefore, there is no need to reassemble the matrix for a new parameter vector which will become important especially in the inversion procedure later on. The tensor class we use was implemented by Martin Afanasjew.

Alternatively, we can try to find a decomposition of $A \in \mathbb{R}^{n \times n}$ (where n is the number of DOFs) as follows:

$$A(\boldsymbol{\sigma}) = S^T \cdot D(\boldsymbol{\sigma}) \cdot S,$$

where the diagonal matrix D contains the electrical conductivities $\boldsymbol{\sigma}$ and S contains the geometric information of the mesh. This decomposition will simplify the process of deriving the system matrix, which is needed for any inversion algorithm. Basically, we have to decompose the integral (2.28) within the assembly process of the system matrix:

$$\begin{aligned} a_K(\phi_j, \phi_i) &= \int_K \nabla\phi_j \cdot \sigma_K \cdot \nabla\phi_i \, d\boldsymbol{x} \\ &= \int_{\hat{K}} \left(B_K^{-T}\hat{\nabla}\hat{\phi}_j\right) \cdot \sigma_K \cdot \left(B_K^{-T}\hat{\nabla}\hat{\phi}_i\right) |\det B_K| \, d\hat{\boldsymbol{x}} \\ &= \int_{\hat{K}} \underbrace{\sqrt{|\det B_K|}\left(B_K^{-T}\hat{\nabla}\hat{\phi}_j\right)}_{S^T} \cdot \underbrace{\sigma_K}_{D} \cdot \underbrace{\sqrt{|\det B_K|}\left(B_K^{-T}\hat{\nabla}\hat{\phi}_i\right)}_{S} d\hat{\boldsymbol{x}}. \end{aligned}$$

For the ith element we get the following decomposition of the corresponding part of the system matrix:

$$A_i = S_i^T D_i S_i$$

$$S_i = \sqrt{|\det B_{K_i}|} \cdot B_{K_i}^{-T} \cdot \hat{\nabla}\hat{\phi} \in \mathbb{R}^{3\times 4}, \quad S_i^T \in \mathbb{R}^{4\times 3}$$

$$D_i = \begin{bmatrix} \sigma_i & 0 & 0 \\ 0 & \sigma_i & 0 \\ 0 & 0 & \sigma_i \end{bmatrix} \in \mathbb{R}^{3\times 3}.$$

The entries of S_i are filled in a $3 \times n$-matrix $\tilde{S}_i$ such that the first column of S_i becomes the i_1th column of $\tilde{S}_i$, where i_k ($k = 1, 2, 3, 4$) are the global nodes that belong to the ith tetrahedron. Finally, we get the decomposition of the assembled system matrix A for m elements and n DOFs in the following way:

$$A = \underbrace{\begin{bmatrix} | & | & & | \\ \tilde{S}_1^T & \tilde{S}_2^T & \cdots & \tilde{S}_n^T \\ | & | & & | \end{bmatrix}}_{n\times 3m} \underbrace{\begin{bmatrix} D_1 & & & \\ & D_2 & & \\ & & \ddots & \\ & & & D_n \end{bmatrix}}_{3m\times 3m} \underbrace{\begin{bmatrix} — & \tilde{S}_1 & — \\ — & \tilde{S}_2 & — \\ & \vdots & \\ — & \tilde{S}_n & — \end{bmatrix}}_{3m\times n}.$$

To calculate the derivatives with respect to σ_i we only have to consider the block D_i in D, which becomes a 3×3-identity matrix. All other entries in D vanish. This second approach is not implemented in our code, but is a promising alternative and should be included in future tests concerning the performance of the assembly algorithm.

2.2.4 The measurement operator

2.2.4.1 Implementation for the electric potential

If we want to generate a synthetic DC resistivity data set $\boldsymbol{b} \in \mathbb{R}^{n_{\text{receivers}}}$ to compare it with an analytical solution and validate the implemented algorithms, we have to evaluate the numerical solution $\boldsymbol{u} \in \mathscr{V}^h$ at the associated receiver positions. Usually, the receivers are not bound to the nodes in the mesh. The finite element discretization enables us to calculate synthetic potential data $\boldsymbol{u}$ at any point within the discretized area. The measurement operator $Q \in \mathbb{R}^{n_{\text{receivers}} \times n_{\text{DOFs}}}$ can be used to calculate this data by a linear combination of the finite element basis functions $\hat{\phi}$ in the following way:

$$\boldsymbol{b} = \begin{bmatrix} \boldsymbol{u}(\boldsymbol{x}_1) \\ \boldsymbol{u}(\boldsymbol{x}_2) \\ \vdots \\ \boldsymbol{u}(\boldsymbol{x}_{n_{\text{receivers}}}) \end{bmatrix} = \sum_{i}^{n_{\text{DOFs}}} u_i \phi_i(\boldsymbol{x}_j) = Q\boldsymbol{u} = \begin{bmatrix} \phi_1(\boldsymbol{x}_1) & \dots & \phi_{n_{\text{DOFs}}}(\boldsymbol{x}_1) \\ \vdots & & \vdots \\ \phi_1(\boldsymbol{x}_{n_{\text{receivers}}}) & \dots & \phi_{n_{\text{DOFs}}}(\boldsymbol{x}_{n_{\text{receivers}}}) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n_{\text{DOFs}}} \end{bmatrix}. \tag{2.35}$$

This equation shows two generic rows of Q and each row belongs to one receiver position. For linear elements, only the four basis functions which are associated with the element K the receiver is located in yield a nonzero contribution to the linear combination. They are evaluated within the reference element and therefore, we transform the receiver location $\boldsymbol{x} \in K$ to a point $\hat{\boldsymbol{x}} \in \hat{K}$ (equation (2.31)). For quadratic elements, we have to evaluate ten basis functions belonging to the receiver element.

Pseudocode 2 CALCULATE J$^{\text{TH}}$ ROW OF Q FOR J$^{\text{TH}}$ RECEIVER

1: Calculate index i of enclosing element K for j$^{\text{th}}$ receiver
2: Extract corresponding DOFs $i_1, \dots, i_4$
3: Solve $B_K \hat{\boldsymbol{x}} = (\text{receivers}(j,:)^T - \boldsymbol{b}_K)$
4: **for** $i = 1, \dots, 4$ **do**
5: $\quad \boldsymbol{q}_K(i) \leftarrow \hat{\phi}_i(\hat{\boldsymbol{x}})$
6: **end for**
7: $Q(j, [i_1\ i_2\ i_3\ i_4]) \leftarrow \boldsymbol{q}_K^T$

Pseudocode 2 gives an overview over the assembly of the j$^{\text{th}}$ row of the measurement operator which is implemented in the function `fe_assemble_observation`. First of all, we use the MATLAB function `tsearchn` to find the enclosing element K for the j$^{\text{th}}$ receiver. Afterwards,

we assemble the transformation matrix B_K and the vector $\boldsymbol{b}_K$ (equation (2.32)) to transform the coordinates of the receiver location (stored in a variable called `receivers` $\in \mathbb{R}^{n_{\text{receivers}} \times 3}$) to the reference element and finally evaluate the basis functions to get the entries of Q.

2.2.4.2 Implementation for the apparent resistivity

If we carry out a DC resistivity measurement, the data calculated from the injected currents and measured potential differences are apparent resistivities ρ_{a} (Section 2.1). The measurement operator Q, described in the previous subsection, extracts potential values at the receiver locations. To convert these values into apparent resistivities, we have to apply the *Neumannsche Formel* as stated in equation (2.1):

$$\rho_{\text{a}} = k\frac{\Delta u}{I}$$

with $k = 2\pi a$ being the geometric factor derived from Green's function (2.4) for a pole-pole configuration (Roy and Apparao, 1971) and a being the spacing between the current source A and the potential electrode M (Figure 2.2). For a current I of 1 A, we implemented the configuration factor k as a diagonal matrix $D \in \mathbb{R}^{n_{\text{receivers}} \times n_{\text{receivers}}}$:

$$D_{\rho_{\text{a}}} = 2\pi \begin{bmatrix} a_1 & & & & \\ & \ddots & & & \\ & & a_j & & \\ & & & \ddots & \\ & & & & a_{n_{\text{receivers}}} \end{bmatrix}, \quad j = 1, \ldots, n_{\text{receivers}}$$

and a_j is the spacing between the current electrode A and the jth receiver electrode M$_j$. Thus, we get the data vector consisting of apparent resistivities as follows:

$$\boldsymbol{b}_{\rho_{\text{a}}} = \rho_{\text{a}}(\boldsymbol{x}) = D_{\rho_{\text{a}}} Q \boldsymbol{u}.$$

This measurement operator is implemented in the function `rhoA_get_transformation` (Appendix A.5.1.3) for 2D and 3D problems. According to the 2D Green's function (2.3), the configuration factor for the pole-pole configuration in a 2D modeling domain is given by

$$k = -\frac{\pi}{\log(a)}.$$

2.2.5 Incorporation of multiple sources

To improve the resolution of a DC resistivity measurement, it is not only important to use several receiver positions, but an appropriate number of sources, too. For multiple sources we have to build up block diagonal matrices of the system matrix $A(\boldsymbol{\sigma})$ and the measurement operator Q with one diagonal block for each source:

$$A_{\text{blk}} = \begin{bmatrix} A(\boldsymbol{\sigma}) & & & \\ & A(\boldsymbol{\sigma}) & & \\ & & \ddots & \\ & & & A(\boldsymbol{\sigma}) \end{bmatrix}, \quad Q_{\text{blk}} = \begin{bmatrix} Q & & & \\ & Q & & \\ & & \ddots & \\ & & & Q \end{bmatrix}.$$

The entries of $A(\boldsymbol{\sigma})$ and Q do not depend on the source position. Therefore, we have n_{s} equal blocks on the main diagonal of both block matrices for n_{s} sources. These block matrices are used later on within the Gauss-Newton scheme (see Section 2.3.1) to expand the sensitivity matrix J for multiple sources. The data vector $\boldsymbol{u}$ is extended for multiple sources by solving the forward problem for each source individually and writing the resulting vector $\boldsymbol{u}_{\text{s}_k}$ $(k = 1, \ldots, n_{\text{s}})$ one below the other:

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_{\text{s}_1} \\ \boldsymbol{u}_{\text{s}_2} \\ \vdots \\ \boldsymbol{u}_{\text{s}_{n_{\text{s}}}} \end{bmatrix}.$$

The spacing a between the source and receiver electrodes changes for multiple source positions. Therefore, we have to extend the measurement operator for the apparent resistivities (see Section 2.2.4.2) as follows:

$$D_{\rho_{\text{a}}}^{\text{blk}} = \begin{bmatrix} D_{\rho_{\text{a}}}^{\text{s}_1} & & & \\ & D_{\rho_{\text{a}}}^{\text{s}_2} & & \\ & & \ddots & \\ & & & D_{\rho_{\text{a}}}^{\text{s}_{n_{\text{s}}}} \end{bmatrix}$$

with $D_{\rho_{\text{a}}}^{\text{blk}} \in \mathbb{R}^{n_{\text{receivers}} \cdot n_{\text{s}} \times n_{\text{receivers}} \cdot n_{\text{s}}}$.

2.2.6 Reference solutions

The previous sections explain the discrete forward problem for DC resistivity modeling. To verify this forward operator, we calculate the electric response for some simple models, where the analytical solution is known and can be compared with the numerical solution. An application of our DC resistivity forward modeling code to a realistic geological setting in comparison with other geophysical methods is shown in detail in Börner et al. (2015a).

There are different possibilities to solve the assembled forward problem, i.e. the Galerkin system given in equation (2.26). For small systems the easiest way is to use MATLAB's direct solver `mldivide`. Our code is able to use both this solver and a multigrid method (Weißflog et al., 2013). Felix Eckhofer implemented an algebraic multigrid (AMG) solver to take advantage of the good scaling properties of this method. Within our code, he integrated HSL-MI20 (Boyle et al. (2007), Boyle et al. (2010)) from the HSL Mathematical Software Library.

2.2.6.1 Concentric semicircles and hemispheres

We want to compare the results of our DC resistivity forward modeling code with the analytical solution for a hemispherical anomaly (given in light grey in Figure 2.7) in a homogeneous halfspace. The point source is of strength I and located at the origin above a 3D halfspace

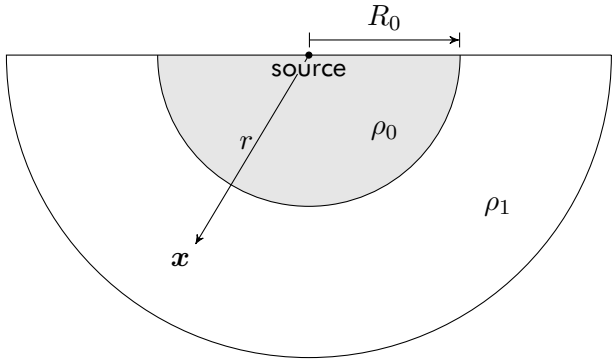

Fig. 2.7: Frontal slice of spherical shell model.

with a resistivity ρ_1 and we have an anomaly with the shape of a hemisphere, a constant resistivity ρ_0 and a radius of $R_0 > 0$. For this model, the analytical potential u at an arbitrary point $\boldsymbol{x} = [x, y, z]^T$ depends on the radial distance r to the source, as the potential of the homogeneous halfspace given in equation (2.4) does, too and is given by

$$u_i = u_i(r) = c_i + \frac{d_i}{r},$$

with $i = 0, 1$ and $r = ||\boldsymbol{x}||$ in both subdomains. The constants c_1, d_1 for the halfspace and c_0, d_0 for the hemispherical anomaly have to be determined. The potential in the subdomain that contains the source is given by u_0 and the second part of it decays with increasing r and contains d_0. This constant can therefore be calculated from the analytical potential for a point source of strength I above a homogeneous halfspace (equation (2.4)):

$$d_0 = \frac{\rho_0 I}{2\pi}.$$

Because the potential decays linearly with increasing r it vanishes at infinity and therefore, c_1 has to be equal to zero. To determine d_1 and c_0 we consider continuity conditions for the potential and the flux at the interface $r = R_0$. For the potential we get:

$$\begin{aligned} u(R_0-) &= u(R_0+) \\ c_0 + \frac{\overbrace{d_0}^{\frac{\rho_0 I}{2\pi}}}{R_0} &= \overbrace{c_1}^{0} + \frac{d_1}{R_0} \\ c_0 &= \frac{d_1}{R_0} - \frac{\rho_0 I}{2\pi R_0} \\ c_0 &= \frac{2\pi d_1 - \rho_0 I}{2\pi R_0}. \end{aligned}$$

The flux yields

$$\begin{aligned} \frac{1}{\rho_0} u'(R_0-) &= \frac{1}{\rho_1} u'(R_0+) \\ -\frac{\overbrace{d_0}^{\frac{\rho_0 I}{2\pi}}}{\rho_0 R_0^2} &= -\frac{d_1}{\rho_1 R_0^2} \\ d_1 &= \frac{\rho_1 I}{2\pi}, \end{aligned}$$

which leads to

$$c_0 = \frac{I(\rho_1 - \rho_0)}{2\pi R_0}.$$

Finally, we arrive at the solution

$$u(r) = \begin{cases} \dfrac{I(\rho_1 - \rho_0)}{2\pi R_0} + \dfrac{\rho_0 I}{2\pi r}, & 0 < r \leq R_0, \\ \dfrac{\rho_1 I}{2\pi r}, & r > R_0. \end{cases}$$

Following the derivation above, we can similarly find a solution for a 2D semicircular halfspace and an anomaly in the shape of a semicircle:

$$u(r) = \begin{cases} \dfrac{(\rho_1 - \rho_0) I}{\pi} \log \dfrac{1}{R_0} + \dfrac{\rho_0 I}{\pi} \log \dfrac{1}{r}, & 0 < r \leq R_0, \\ \dfrac{\rho_1 I}{\pi} \log \dfrac{1}{r}, & r > R_0. \end{cases}$$

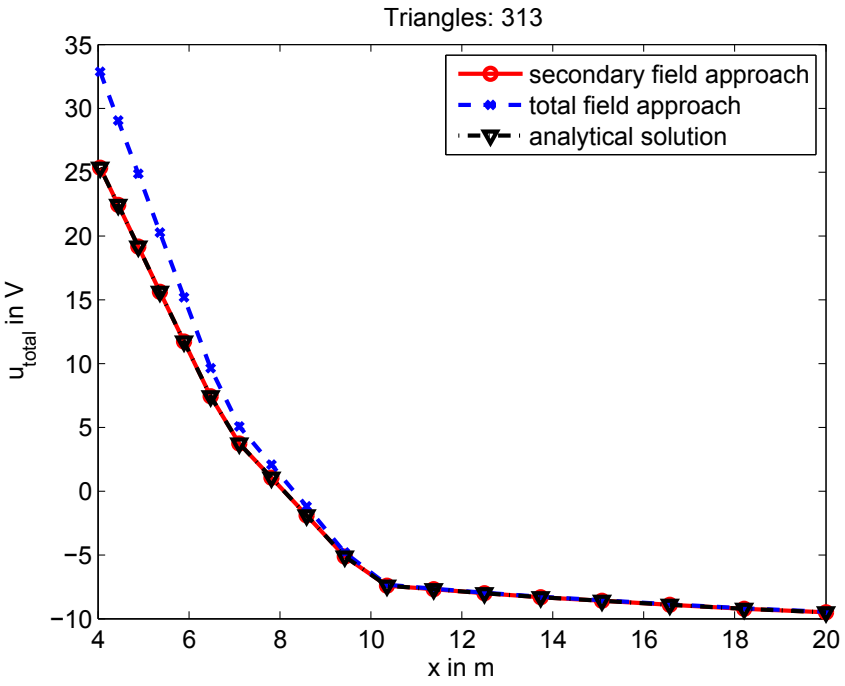

Fig. 2.8: Simulated and analytical total potential for the 2D semicircle model (313 elements).

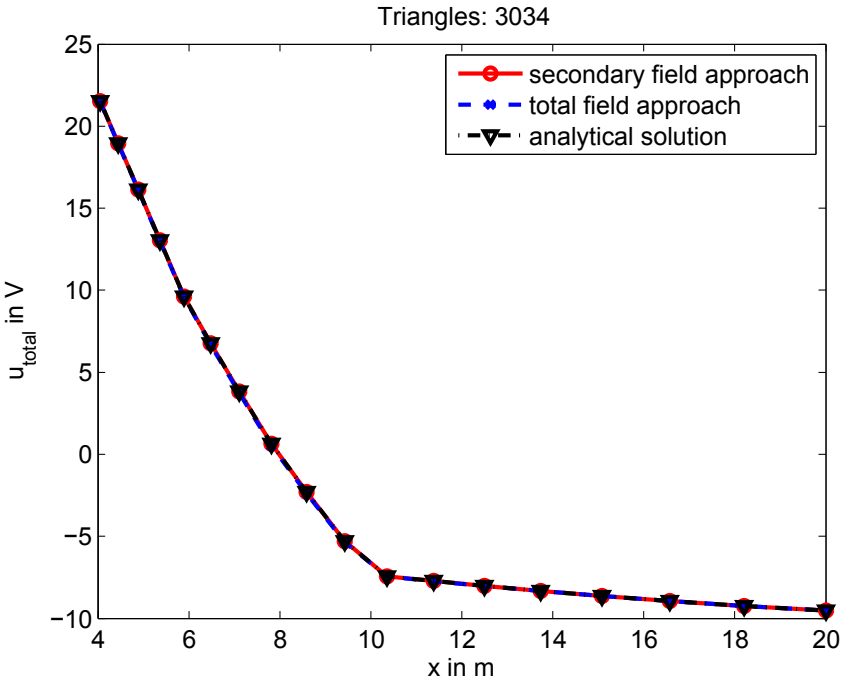

Fig. 2.9: Simulated and analytical total potential for the 2D semicircle model (3034 elements).

Figures 2.8 and 2.9 show the total potential for the 2D model with the parameters $R_0 = 10\,\text{m}$, $\rho_1 = 10\,\Omega\text{m}$ and $\rho_0 = 100\,\Omega\text{m}$. We plotted the potential calculated with our forward modeling code for the total and the secondary field approach compared with the analytical solution $u(r)$

given in the equation above for a coarse (313 triangles) and a finer mesh (3034 triangles). The Figures clearly show that our DC resistivity software approximates the analytical solution very accurately and the secondary field approach yields a better approximation than the total field approach for coarse meshes (Figure 2.8 and Section 2.2.2). The modeling results for the 3D hemispherical model are similar to the 2D results.

2.2.6.2 The layered halfspace

To verify the forward operator we examine the $\frac{1}{r}$-dependence of the electric potential (equation (2.4)) and calculate a geoelectrical sounding curve for a conductive layer in a resistive halfspace with the resistivities $\rho = [100, 10, 100]\,\Omega$m, thicknesses $h = [5, 1, \infty]$ m and a current of $I = 1$ A. We simulate a pole-pole configuration and therefore, the apparent resistivity ρ_a is a function of the spacing a as given in equation (2.1):

$$\rho_a(a) = k \cdot \frac{\Delta u}{I},$$

where

$$k = 2\pi a.$$

We define 20 logarithmically spaced measuring points for an electrode spacing between $a = 2$ m and $a = 200$ m. For the pole-pole configuration, the best indication of a conductive layer in a halfspace can be measured at a spacing which is about three times the depth of the layer (see Roy and Apparao (1971) and Section 2.1). Because the depth of investigation primarily depends on the conductivity distribution this is only an approximate value. For our model we will expect this maximum at a spacing of about 15 m. Figure 2.10 shows the sounding curves for the measuring points at the Earth's surface in negative and positive y-direction at $x = 0$ m. If we measure very close to or far from the source we get the indication of the halfspace with about 100 Ωm because the depth of investigation increases with the electrode spacing and therefore, a large spacing enables us to measure the response of the underlying halfspace. The conductive layer at 5 m depth induces the minimum apparent resistivity at a spacing of about 12 m which corresponds to the estimated depth of about 15 m. Due to the radial symmetry of the electric potential field to the point source, the apparent resistivity calculated from the surface potential should be equal for identical radial distances r to the source position. The small deviations between positive and negative y-direction can be explained by the relatively coarse mesh at the surface of the model.

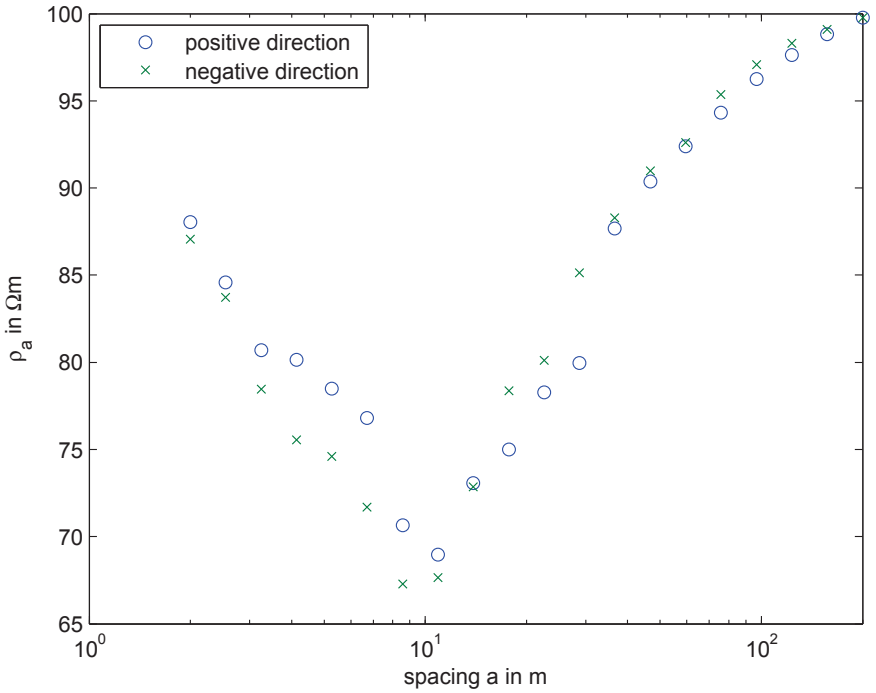

Fig. 2.10: Sounding curve for a pole-pole configuration.

2.3 Inversion of DC resistivity data

The main goal of DC resistivity measurements is to find a parameter model for the subsurface that explains the measured data within the limits of the laws of physics. The previous chapter explains the technique of forward modeling where we produce synthetic data for a given parameter model in order to study measurement configurations and other important issues when preparing a geophysical measurement. The inverse or parameter estimation problem on the other hand solves a minimization problem to find parameters which explain the measured data best (Figure 2.11). Within the following sections, we explain the main

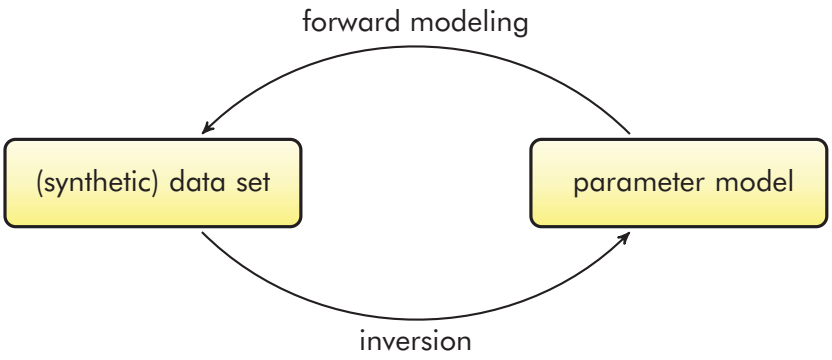

Fig. 2.11: Relation between forward modeling and inverse problems.

ideas of the inverse problem for the DC resistivity method and illustrate used functions and solution techniques. Obviously mathematical algorithms do not necessarily respect physical properties of the real world. In particular an optimal parameter distribution could contain unphysical negative values of the electrical conductivity. To avoid this problem, we use a simple transformation which works well for parameters of the same order of magnitude:

$$\boldsymbol{\sigma} = e^{\boldsymbol{m}}$$

$$\boldsymbol{m} = \ln \boldsymbol{\sigma}.$$

Using these logarithmized parameters, the Galerkin matrix $A(\boldsymbol{\sigma})$ introduced in equation (2.26) has the form

$$A(\boldsymbol{\sigma}) = A(e^{\boldsymbol{m}}) \rightarrow A(\boldsymbol{m}) \tag{2.36}$$

with the model parameter vector $\boldsymbol{m}$. For reasons of clarity, we describe the inversion algorithm and all related properties for electric potential data $\boldsymbol{b} = Q\boldsymbol{u}$. For an inversion using apparent resistivities ρ_a, the measurement operator Q must always be multiplied by D_{ρ_a} as described in Section 2.2.4.2.

2.3.1 The Gauss-Newton method

To solve the inverse problem, an appropriate and sufficiently smooth parameter model must be reconstructed, such that the deviation between measured and modeled data is minimized. Hence, the inversion process requires us to solve a minimization problem which combines the data residual and a regularization operator $R(\boldsymbol{m})$:

$$\begin{aligned} \Phi(\boldsymbol{m}) &= \tfrac{1}{2}\|\boldsymbol{g}(\boldsymbol{m}) - \boldsymbol{b}\|_2^2 + \beta R(\boldsymbol{m} - \boldsymbol{m}_\mathrm{ref}) \rightarrow \min_{\boldsymbol{m}} \\ &= \underbrace{\tfrac{1}{2}\|Q\boldsymbol{u} - \boldsymbol{b}\|_2^2}_{\text{data residual}} + \underbrace{\tfrac{\beta}{2}||W(\boldsymbol{m} - \boldsymbol{m}_\mathrm{ref})||^2}_{\text{regularization norm}} \rightarrow \min_{\boldsymbol{m}} \\ &\text{subject to } A(\boldsymbol{m})\boldsymbol{u} = \boldsymbol{f}. \end{aligned} \tag{2.37}$$

Here, $\boldsymbol{b}$ is the measured data, $\boldsymbol{g}(\boldsymbol{m}) = QA(\boldsymbol{m})^{-1}\boldsymbol{f} = Q\boldsymbol{u}$ the modeled data, $\boldsymbol{m}$ are the model parameters, Q is some measurement operator ($Q\boldsymbol{u} \approx \boldsymbol{b}$), W is the regularization matrix, $\boldsymbol{m}_\mathrm{ref}$ is the reference model and β represents the regularization parameter. The regularization norm is not always an L^2 norm but can be defined in different spaces and therefore, we leave a fixed index out. We will explain the regularization operator and the role and determination of β later in Section 2.3.5.

Applying a Gauss-Newton scheme (Schwetlick (1979), Nocedal and Wright (2006)) to solve this non-linear least squares problem requires us to linearize $\Phi(\boldsymbol{m})$. Linearizing at $\boldsymbol{m} = \boldsymbol{m}_0$ using the Taylor series

$$\boldsymbol{g}(\boldsymbol{m}) = \boldsymbol{g}(\boldsymbol{m}_0) + \boldsymbol{g}'(\boldsymbol{m}_0)\Delta\boldsymbol{m} + \mathcal{O}(|\Delta\boldsymbol{m}|^2) \quad \text{where} \quad \Delta\boldsymbol{m} = \boldsymbol{m} - \boldsymbol{m}_0 \tag{2.38}$$

the minimization problem (2.37) changes to

$$\begin{aligned}\Phi(\boldsymbol{m}) &= \tfrac{1}{2}\|\boldsymbol{g}(\boldsymbol{m}_0) + \boldsymbol{g}'(\boldsymbol{m}_0)\Delta\boldsymbol{m} - \boldsymbol{b}\|_2^2 + \tfrac{\beta}{2}||W(\Delta\boldsymbol{m} + \boldsymbol{m}_0 - \boldsymbol{m}_{\text{ref}})||^2 \\ &= \tfrac{1}{2}\|\underbrace{[\boldsymbol{g}(\boldsymbol{m}_0) - \boldsymbol{b}]}_{\text{data residual}} + \underbrace{\boldsymbol{g}'(\boldsymbol{m}_0)\Delta\boldsymbol{m}}_{\text{Jacobian matrix}}\|_2^2 + \tfrac{\beta}{2}||W(\Delta\boldsymbol{m} + \boldsymbol{m}_0 - \boldsymbol{m}_{\text{ref}})||^2 \to \min_{\boldsymbol{m}}.\end{aligned} \tag{2.39}$$

Here, $\boldsymbol{g}'(\boldsymbol{m}_0)$ is the Jacobian or sensitivity matrix. To minimize the objective function $\Phi(\boldsymbol{m})$ according to equation (2.39), its derivative with respect to $\boldsymbol{m}$ has to vanish. Considering the differentiation rules for vector-valued functions in Appendix A.3, we arrive at

$$\begin{aligned}\frac{\partial\Phi(\boldsymbol{m})}{\partial\boldsymbol{m}} = \frac{\partial\Phi(\boldsymbol{m})}{\partial\Delta\boldsymbol{m}} \cdot \frac{\partial\Delta\boldsymbol{m}}{\partial\boldsymbol{m}}^{1} = \boldsymbol{g}'^{T}(\boldsymbol{m}_0)\big([\boldsymbol{g}(\boldsymbol{m}_0) - \boldsymbol{b}] + \boldsymbol{g}'(\boldsymbol{m}_0)\Delta\boldsymbol{m}\big) + \dots \\ \dots + \beta W^T W\left(\Delta\boldsymbol{m} + \boldsymbol{m}_0 - \boldsymbol{m}_{\text{ref}}\right) = \boldsymbol{0}.\end{aligned} \tag{2.40}$$

Finally, equation (2.40) yields the normal equations:

$$\left(\boldsymbol{g}'^{T}(\boldsymbol{m}_0)\boldsymbol{g}'(\boldsymbol{m}_0) + \beta W^T W\right)\Delta\boldsymbol{m} = \boldsymbol{g}'^{T}(\boldsymbol{m}_0)[\boldsymbol{b} - \boldsymbol{g}(\boldsymbol{m}_0)] + \beta W^T W(\boldsymbol{m}_{\text{ref}} - \boldsymbol{m}_0).$$

For an arbitrary model update $\Delta\boldsymbol{m}_k$ we get

$$\left(\boldsymbol{g}'^{T}(\boldsymbol{m}_k)\boldsymbol{g}'(\boldsymbol{m}_k) + \beta W^T W\right)\Delta\boldsymbol{m}_k = \boldsymbol{g}'^{T}(\boldsymbol{m}_k)[\boldsymbol{b} - \boldsymbol{g}(\boldsymbol{m}_k)] + \beta W^T W(\boldsymbol{m}_{\text{ref}} - \boldsymbol{m}_k). \tag{2.41}$$

The solution of the normal equations yields a new approximation to the model parameters given by

$$\boldsymbol{m}_{k+1} = \boldsymbol{m}_k + \Delta\boldsymbol{m}_k.$$

Within our DC resistivity code we establish the normal equations explicitly for sufficiently small model problems and solve them using MATLAB's built-in sparse solver `mldivide` or using Intel MKL PARDISO or the preconditioned conjugate gradients (PCG) Krylov subspace method (see Section 2.3.2). For large systems of equations it can be useful to apply an appropriate Krylov subspace method such as LSQR to the original least squares problem instead of solving the normal equations. For this reason we have to reformulate equation (2.37). The main idea is the combination of both the data residual and the regularization operator in one norm. With

$$G(\boldsymbol{m}) = \begin{bmatrix} QA(\boldsymbol{m})^{-1}\boldsymbol{f} \\ \sqrt{\beta}W\boldsymbol{m} \end{bmatrix} \quad \text{and} \quad \boldsymbol{a} = \begin{bmatrix} \boldsymbol{b} \\ \sqrt{\beta}W\boldsymbol{m}_{\text{ref}} \end{bmatrix}$$

we get

$$\begin{aligned}\Phi(\boldsymbol{m}) &= \tfrac{1}{2}||G(\boldsymbol{m}) - \boldsymbol{a}||_2^2 \rightarrow \min_{\boldsymbol{m}} \\ &= \tfrac{1}{2}\left\| \begin{bmatrix} QA(\boldsymbol{m})^{-1}\boldsymbol{f} \\ \sqrt{\beta}W\boldsymbol{m} \end{bmatrix} - \begin{bmatrix} \boldsymbol{b} \\ \sqrt{\beta}W\boldsymbol{m}_{\text{ref}} \end{bmatrix} \right\|_2^2 \rightarrow \min_{\boldsymbol{m}}.\end{aligned}$$

Using the Taylor series (2.38) we are again able to linearize the least squares problem:

$$\begin{aligned}\Phi(\boldsymbol{m}_0) &= \tfrac{1}{2}\|[G(\boldsymbol{m}_0) - \boldsymbol{a}] + G'(\boldsymbol{m}_0)\Delta\boldsymbol{m}\|_2^2 \\ &= \frac{1}{2}\left\| \underbrace{\begin{bmatrix} \boldsymbol{b} - Q\boldsymbol{u} \\ \sqrt{\beta}W(\boldsymbol{m}_{\text{ref}} - \boldsymbol{m}) \end{bmatrix}}_{\text{extended data residual}} - \underbrace{\begin{bmatrix} g'(\boldsymbol{m}) \\ \sqrt{\beta}W \end{bmatrix}}_{\text{extended Jacobian matrix}} \Delta\boldsymbol{m} \right\|_2^2 \rightarrow \min_{\boldsymbol{m}}.\end{aligned} \tag{2.42}$$

This reformulation leads to additional entries in the data residual and Jacobian matrix regarding the regularization norm. Now, the solution of this minimization problem can be determined by the application of appropriate Krylov subspace methods which are explained in the following section.

2.3.2 Krylov subspace methods

Krylov subspace methods are iterative solvers for large sparse linear systems of equations. Unless noted otherwise, we will explain the main ideas and properties according to Saad (2003). First, we consider the linear system of equations

$$A\boldsymbol{x} = \boldsymbol{b} \tag{2.43}$$

with the regular matrix $A \in \mathbb{R}^{n\times n}$. The unique solution of equation (2.43) is given by $\boldsymbol{x}_* = A^{-1}\boldsymbol{b} \in \mathbb{R}^n$. Krylov subspace methods calculate a solution $\boldsymbol{x}_k$ which approximates $\boldsymbol{x}_*$ and $\boldsymbol{x}_k$ is contained in a shifted Krylov subspace:

$$\boldsymbol{x}_k \ \in \ \boldsymbol{x}_0 + \mathscr{K}_k(A, \boldsymbol{r}_0). \tag{2.44}$$

$\boldsymbol{x}_0$ is an arbitrary initial guess for the iterative algorithm with the corresponding residual $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0 \in \mathbb{R}^n$. The kth Krylov subspace given in equation (2.44) is defined as follows:

$$\begin{aligned}\mathscr{K}_k(A, \boldsymbol{r}_0) &:= \text{span}\{\boldsymbol{r}_0, A\boldsymbol{r}_0, A^2\boldsymbol{r}_0, \ldots, A^{k-1}\boldsymbol{r}_0\} \\ &= \{\alpha_0\boldsymbol{r}_0 + \alpha_1 A\boldsymbol{r}_0 + \alpha_2 A^2\boldsymbol{r}_0 + \cdots + \alpha_{k-1}A^{k-1}\boldsymbol{r}_0, \ \alpha_j \in \mathbb{R}\} \\ &= \{p(A)\boldsymbol{r}_0 \ : \ p \in \mathscr{P}_{k-1}\},\end{aligned} \tag{2.45}$$

where $\mathscr{P}_{k-1}$ describes the space containing all real polynomials with a maximal degree of $k-1$. Consequently, $\mathscr{K}_k(A, \boldsymbol{r}_0)$ is generated by a linear combination of the vectors $\boldsymbol{r}_0$, $A\boldsymbol{r}_0$, $A^2\boldsymbol{r}_0, \ldots, A^{k-1}\boldsymbol{r}_0$. It is easy to see that the dimension of the Krylov subspaces which belong to a linear system of equations grows if we add another vector $A^j\boldsymbol{r}_0$ and each new, larger space contains all previous subspaces:

$$\{\mathbf{0}\} = \mathscr{K}_0 \subseteq \text{span}\{\boldsymbol{r}_0\} = \mathscr{K}_1 \subseteq \cdots \subseteq \mathscr{K}_k \subseteq \mathscr{K}_{k+1} \subseteq \cdots \subseteq \mathbb{R}^n.$$

The dimension of Krylov subspaces cannot grow arbitrarily. Because each new vector in the linear span (equation (2.45)) is given by a multiplication of the residual $\boldsymbol{r}_0 \in \mathbb{R}^n$ with a power of the system matrix $A \in \mathbb{R}^{n\times n}$, all of these vectors are in $\mathbb{R}^n$, too and can only generate a space which is of maximal dimension n (in case of linear independence). If the index $k = L$, for which the dimension of the subspaces does not grow anymore is reached, we call the space stationary. For all $k \geq L$, the exact solution $\boldsymbol{x}_* = A^{-1}\boldsymbol{b}$ is contained in the shifted Krylov subspace $\boldsymbol{x}_0 + \mathscr{K}_k$. Hence, Krylov subspace methods are also exact solvers because (without rounding errors) the exact solution of the linear system of equations will be found at the latest after n steps. Since $\boldsymbol{x}_0 + \mathscr{K}_k$ often contains an appropriate approximation of $\boldsymbol{x}_*$ for small values of k, the main field of application is the iterative determination of approximate solutions with sufficiently small computational effort. These approximations are determined according to equation (2.45) as follows:

$$\begin{aligned}\boldsymbol{x}_* = A^{-1}\boldsymbol{b} \approx \boldsymbol{x}_k &= \boldsymbol{x}_0 + p_{k-1}(A)\boldsymbol{r}_0 \\ &= \boldsymbol{x}_0 + p_{k-1}(A)(\boldsymbol{b} - A\boldsymbol{x}_0)\end{aligned}$$

and in the simplest case (for $\boldsymbol{x}_0 = \mathbf{0}$) there holds:

$$A^{-1}\boldsymbol{b} \approx p_{k-1}(A)\boldsymbol{b}.$$

Thus, the inverse of the matrix A is approximated by a polynomial of lower degree in A. Krylov subspace methods can be divided into two groups: The first ones are called minimal residual (or MR) methods. They determine an approximate solution $\boldsymbol{x}_k^{\text{MR}}$ in the shifted Krylov subspace $\boldsymbol{x}_0 + \mathscr{K}_k(A, \boldsymbol{r}_0)$ such that the residual norm is minimized:

$$||\boldsymbol{b} - A\boldsymbol{x}_k^{\text{MR}}|| = \min_{\boldsymbol{x}\in\boldsymbol{x}_0+\mathscr{K}_k(A,\boldsymbol{r}_0)} ||\boldsymbol{b} - A\boldsymbol{x}||.$$

In contrast, the orthogonal residual (or OR) methods built up the solution $\boldsymbol{x}_k^{\text{OR}}$ in the shifted Krylov subspace on the following condition:

$$\boldsymbol{b} - A\boldsymbol{x}_k^{\text{OR}} \perp \mathscr{K}_k(A, \boldsymbol{r}_0).$$

Here, the residual $\boldsymbol{r}_k = \boldsymbol{b} - A\boldsymbol{x}_k^{\mathrm{OR}}$ has to be orthogonal to the kth Krylov subspace.

Within our DC inversion code we apply the MATLAB routines for the Krylov subspace method CG (Hestenes and Stiefel, 1952) directly to the normal equations and the LSQR routine (Paige and Saunders, 1982) to the extended least squares problem given in equation (2.42).

2.3.3 Explicit calculation of the sensitivity matrix

To apply the Gauss-Newton method we need to calculate the so-called Jacobian or sensitivity matrix $J = g'(\boldsymbol{m})$ as given in equation (2.41). This matrix contains the derivative of the synthetic data $Q\boldsymbol{u}$ with respect to the parameter vector $\boldsymbol{m}$ and describes the influence of the underlying conductivity structure on the measured data set.

Conveniently, we choose to use the physical parameter vector $\boldsymbol{\sigma}$ to establish the Jacobian matrix and replace it by the logarithmic parameter vector $\boldsymbol{m}$ as given in equation (2.36) later on. The derivative of the primary potential $\boldsymbol{u}_{\mathrm{p}}$ for a homogeneous halfspace defined in equation (2.4) is given by:

$$\frac{\partial \boldsymbol{u}_{\mathrm{p}}}{\partial \sigma_i} = \begin{cases} \mathbf{0}, & \sigma_i \neq \sigma_0, \\ -\dfrac{1}{\sigma_0}\boldsymbol{u}_{\mathrm{p}}, & \sigma_i = \sigma_0. \end{cases} \tag{2.46}$$

It only yields a contribution for conductivities equal to the source conductivity. To derive the secondary potential $\boldsymbol{u}_{\mathrm{s}}$, we apply the product rule to the secondary field equation (2.27):

$$\frac{\partial}{\partial \sigma_i}\Big(A(\boldsymbol{\sigma})\boldsymbol{u}_{\mathrm{s}}\Big) = \frac{\partial}{\partial \sigma_i}\Big(A(\boldsymbol{\sigma}_0 - \boldsymbol{\sigma})\boldsymbol{u}_{\mathrm{p}}\Big).$$

Using the decomposition of the right-hand side

$$A(\boldsymbol{\sigma}_0 - \boldsymbol{\sigma}) = \sigma_0 A(\mathbf{1}) - A(\boldsymbol{\sigma})$$

due to Rücker et al. (2006) and because A is linear in $\boldsymbol{\sigma}$ we arrive at

$$\frac{\partial}{\partial \sigma_i}\Big(A(\boldsymbol{\sigma})\boldsymbol{u}_{\mathrm{s}}\Big) = \frac{\partial}{\partial \sigma_i}\Big(\sigma_0 A(\mathbf{1})\boldsymbol{u}_{\mathrm{p}} - A(\boldsymbol{\sigma})\boldsymbol{u}_{\mathrm{p}}\Big)$$

and finally get the derivative:

$$\frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}\boldsymbol{u}_{\mathrm{s}} + A(\boldsymbol{\sigma})\frac{\partial \boldsymbol{u}_{\mathrm{s}}}{\partial \sigma_i} = \frac{\partial \sigma_0 A(\mathbf{1})}{\partial \sigma_i}\boldsymbol{u}_{\mathrm{p}} + \sigma_0 A(\mathbf{1})\frac{\partial \boldsymbol{u}_{\mathrm{p}}}{\partial \sigma_i} - \frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}\boldsymbol{u}_{\mathrm{p}} - A(\boldsymbol{\sigma})\frac{\partial \boldsymbol{u}_{\mathrm{p}}}{\partial \sigma_i}. \tag{2.47}$$

Now, we have to differentiate between the cases where σ_i is equal to the source conductivity σ_0 and where it is not. For all conductivities that differ from σ_0 several terms in (2.47) vanish:

$$\frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}\boldsymbol{u}_\mathsf{s} + A(\boldsymbol{\sigma})\frac{\partial \boldsymbol{u}_\mathsf{s}}{\partial \sigma_i} = \cancelto{0}{\frac{\partial \sigma_0 A(\mathbf{1})}{\partial \sigma_i}}\boldsymbol{u}_\mathsf{p} + \sigma_0 A(\mathbf{1})\cancelto{0}{\frac{\partial \boldsymbol{u}_\mathsf{p}}{\partial \sigma_i}} - \frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}\boldsymbol{u}_\mathsf{p} - A(\boldsymbol{\sigma})\cancelto{0}{\frac{\partial \boldsymbol{u}_\mathsf{p}}{\partial \sigma_i}}$$

$$A(\boldsymbol{\sigma})\frac{\partial \boldsymbol{u}_\mathsf{s}}{\partial \sigma_i} = -\frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}\left(\boldsymbol{u}_\mathsf{p} + \boldsymbol{u}_\mathsf{s}\right)$$

$$\frac{\partial \boldsymbol{u}_\mathsf{s}}{\partial \sigma_i} = -A(\boldsymbol{\sigma})^{-1}\left(\frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}\left(\boldsymbol{u}_\mathsf{p} + \boldsymbol{u}_\mathsf{s}\right)\right). \tag{2.48}$$

If σ_i equals the source conductivity, we get the following expressions:

$$\frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}\boldsymbol{u}_\mathsf{s} + A(\boldsymbol{\sigma})\frac{\partial \boldsymbol{u}_\mathsf{s}}{\partial \sigma_i} = \cancelto{A(\mathbf{1})}{\frac{\partial \sigma_0 A(\mathbf{1})}{\partial \sigma_i}}\boldsymbol{u}_\mathsf{p} + \sigma_0 A(\mathbf{1})\cancelto{-\frac{1}{\sigma_0}\boldsymbol{u}_\mathsf{p}}{\frac{\partial \boldsymbol{u}_\mathsf{p}}{\partial \sigma_i}} - \frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}\boldsymbol{u}_\mathsf{p} - A(\boldsymbol{\sigma})\cancelto{-\frac{1}{\sigma_0}\boldsymbol{u}_\mathsf{p}}{\frac{\partial \boldsymbol{u}_\mathsf{p}}{\partial \sigma_i}}$$

$$A(\boldsymbol{\sigma})\frac{\partial \boldsymbol{u}_\mathsf{s}}{\partial \sigma_i} = -\frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}\left(\boldsymbol{u}_\mathsf{p} + \boldsymbol{u}_\mathsf{s}\right) + A(\boldsymbol{\sigma})\frac{1}{\sigma_0}\boldsymbol{u}_\mathsf{p}$$

$$\frac{\partial \boldsymbol{u}_\mathsf{s}}{\partial \sigma_i} = -A(\boldsymbol{\sigma})^{-1}\left(\frac{\partial A(\boldsymbol{\sigma})}{\partial \sigma_i}\left(\boldsymbol{u}_\mathsf{p} + \boldsymbol{u}_\mathsf{s}\right)\right) + \frac{1}{\sigma_0}\boldsymbol{u}_\mathsf{p}. \tag{2.49}$$

It is easy to see, that the derivatives (2.48) and (2.49) differ only in the additional term $\frac{1}{\sigma_0}\boldsymbol{u}_\mathsf{p}$ which has to be added for conductivities equal to the source conductivity. If we add the derivatives for the primary and secondary potential given in equations (2.46), (2.48) and (2.49), we get the following expression regarding the total field $\boldsymbol{u} = \boldsymbol{u}_\mathsf{p} + \boldsymbol{u}_\mathsf{s}$:

$$J = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{\sigma}} = -A(\boldsymbol{\sigma})^{-1}\left(\frac{\partial A(\boldsymbol{\sigma})}{\partial \boldsymbol{\sigma}} \times_2 \boldsymbol{u}\right).$$

The ith ($i = 1 \dots n_{\text{parameters}}$) column of J contains the sensitivities with respect to the ith parameter and the kth ($k = 1 \dots n_{\text{DOFs}}$) row contains the sensitivities for the kth degree of freedom. To extract the relevant receiver positions, we have to multiply the matrix J by the measurement operator Q. We finally use the logarithmic parameters and calculate J as follows:

$$J = \frac{\partial Q\boldsymbol{u}}{\partial \boldsymbol{m}} = -QA(\boldsymbol{m})^{-1}\left(\frac{\partial A(\boldsymbol{m})}{\partial \boldsymbol{m}} \times_2 \boldsymbol{u}\right). \tag{2.50}$$

To simplify this expression, we use the term $A'(\boldsymbol{m})$ for the three-way tensor instead of $\frac{\partial A(\boldsymbol{m})}{\partial \boldsymbol{m}}$.

To show the influence of an embedded conductive or resistive body in a layered halfspace on the resolution of the pole-pole configuration, we did some sensitivity studies for a simple 2D model given in Figure 2.12. Besides a layered background, the conductivity distribution

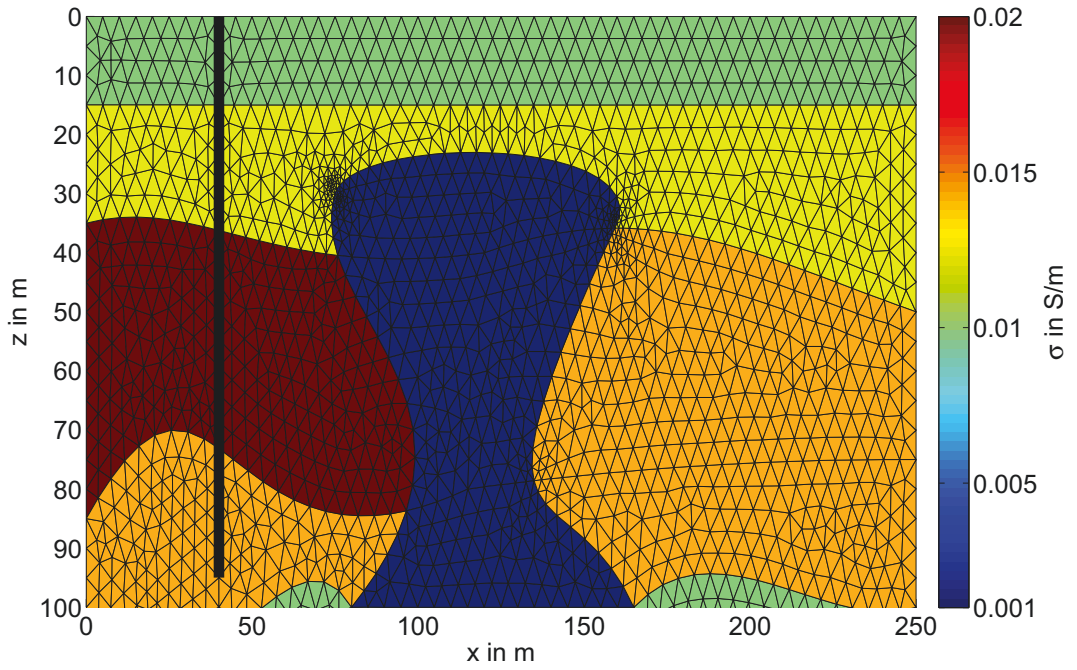

Fig. 2.12: 2D model with the source electrode A located in a borehole at $x = 40\,\text{m}$.

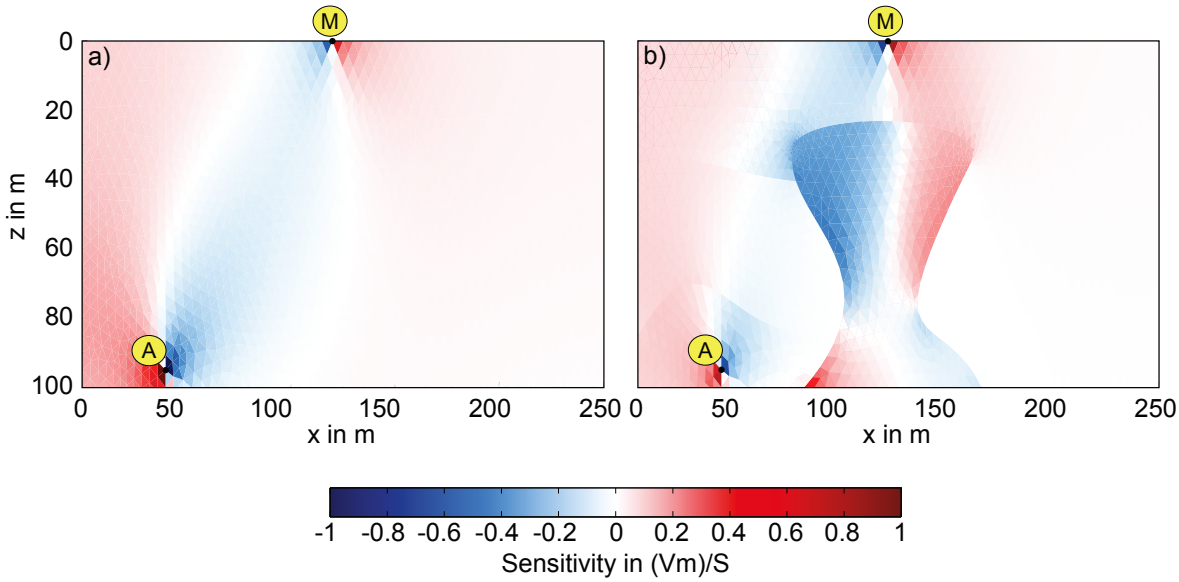

Fig. 2.13: Sensitivities for a homogeneous halfspace (a) and for the inhomogeneous model (b) given in Figure 2.12.

contains a salt dome with a very high resistivity of 1000 Ωm. The source A is located in a borehole at $x = 40\,\text{m}$ and $z = 95\,\text{m}$, whereas the measuring electrode M is located at $x = 120\,\text{m}$ and $z = 0\,\text{m}$. Figures 2.13 a) and b) show the sensitivity distribution for a homogeneous halfspace in comparison with the given conductivity model. In the homogeneous case we see the well-known shape of negative sensitivity between A and M. In Figure 2.13b), the body causes serious distortions of the sensitivity pattern.

2.3.4 Implicit calculation of the sensitivity matrix

Although the tensor (see Section 2.2.3.5), which contains the partial derivatives with respect to all entries of the parameter vector $\boldsymbol{m}$, is sparse, we cannot afford to store the full sensitivity matrix J (equation (2.50)) because it is very large and dense. To apply an iterative and especially a Krylov subspace method to the least squares problem given in equation (2.42) we do not need to built up the full sensitivity matrix, but we have to be able to multiply J by an arbitrary vector $\boldsymbol{v}$ (cf. Mackie and Madden (1993), Siripunvaraporn and Egbert (2007)):

$$\boldsymbol{y} = J\boldsymbol{v} = -QA(\boldsymbol{m})^{-1}\left(A'(\boldsymbol{m}) \times_2 \boldsymbol{u}\right)\boldsymbol{v} \tag{2.51}$$

or for the transposed case

$$\boldsymbol{y} = J^T\boldsymbol{v} = -\left(A'(\boldsymbol{m}) \times_2 \boldsymbol{u}\right)^T A(\boldsymbol{m})^{-T}Q^T\boldsymbol{v}.$$

We want to avoid to solve a system of equations with several right-hand sides in $(A'(\boldsymbol{m}) \times_2 \boldsymbol{u})$ because this matrix is very large and dense. Therefore, we rewrite equation (2.51) and solve the system with the measurement operator Q as the right-hand side, which is smaller and not as dense (see Section 2.2.4):

$$QA(\boldsymbol{m})^{-1} = \left(A(\boldsymbol{m})^{-T}Q^T\right)^T.$$

The implicit multiplication with J is implemented in the function `multJ`. Pseudocode 3 only shows a simple implementation of the products $\boldsymbol{y} = J \cdot \boldsymbol{v}$ and $\boldsymbol{y} = J^T \cdot \boldsymbol{v}$ without the regularization functional as shown in equation (2.37).

Pseudocode 3 IMPLICIT MULTIPLICATION WITH J

1: $G \leftarrow A'(\boldsymbol{m}) \times_2 \boldsymbol{u}$
2: **if** transpose **then**
3: $\quad \boldsymbol{w} \leftarrow Q^T\boldsymbol{v}$
4: $\quad$ Solve $-A(\boldsymbol{m})^T\boldsymbol{z} = \boldsymbol{w}$
5: $\quad \boldsymbol{y} \leftarrow G^T\boldsymbol{z}$
6: $\quad$ **return** $\boldsymbol{y}$
7: **else**
8: $\quad \boldsymbol{z} \leftarrow G\boldsymbol{v}$
9: $\quad$ Solve $-A(\boldsymbol{m})^T X = Q^T$
10: $\quad \boldsymbol{y} \leftarrow X^T\boldsymbol{z}$
11: $\quad$ **return** $\boldsymbol{y}$
12: **end if**

Including this contribution we get

$$\boldsymbol{y} = J \cdot \boldsymbol{v} = \begin{bmatrix} J_1 \cdot \boldsymbol{v} \\ J_2 \cdot \boldsymbol{v} \end{bmatrix} = \begin{bmatrix} -QA(\boldsymbol{m})^{-1} \left(A'(\boldsymbol{m}) \times_2 \boldsymbol{u}\right) \cdot \boldsymbol{v} \\ \sqrt{\beta} W \cdot \boldsymbol{v} \end{bmatrix},$$

$$\boldsymbol{y} = J^T \cdot \boldsymbol{v} = \begin{bmatrix} J_1^T & J_2^T \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix} = \left[J_1^T \boldsymbol{w}_1 + J_2^T \boldsymbol{w}_2 \right], \text{ and } \boldsymbol{v} = \begin{bmatrix} \boldsymbol{w}_1 \\ \boldsymbol{w}_2 \end{bmatrix}.$$

Then, the algorithm changes to the function we are using within our DC resistivity code (Pseudocode 4). To do sensitivity studies (discussed in Section 2.3.3) it is advantageous to be able to calculate the full sensitivity matrix or at least one column or one row of it. To achieve this, we can pass the ith unit vector to the function `multJ` to get the ith column or row (transposed case) of the matrix J.

Pseudocode 4 MULTJ.M

1: $G \leftarrow A'(\boldsymbol{m}) \times_2 \boldsymbol{u}$
2: **if** transpose **then**
3: $\quad \boldsymbol{w}_1 \leftarrow \boldsymbol{v}(1 : \texttt{end} - \texttt{size}(\texttt{W}, 1))$
4: $\quad \boldsymbol{w}_2 \leftarrow \boldsymbol{v}(\texttt{end} - \texttt{size}(\texttt{W}, 1) + 1 : \texttt{end})$
5: $\quad \boldsymbol{w}_1 \leftarrow Q^T \boldsymbol{w}_1$
6: $\quad$ Solve $-A(\boldsymbol{m})^T \boldsymbol{z} = \boldsymbol{w}_1$
7: $\quad \boldsymbol{y}_1 \leftarrow G^T \boldsymbol{z}$
8: $\quad \boldsymbol{y}_2 \leftarrow \sqrt{\beta} W^T \cdot \boldsymbol{w}_2$
9: $\quad$ **return** $\boldsymbol{y}_1 + \boldsymbol{y}_2$
10: **else**
11: $\quad \boldsymbol{z} \leftarrow G\boldsymbol{v}$
12: $\quad$ Solve $-A(\boldsymbol{m})^T X = Q^T$
13: $\quad \boldsymbol{y}_1 \leftarrow X^T \cdot \boldsymbol{z}$
14: $\quad \boldsymbol{y}_2 \leftarrow \sqrt{\beta} W \cdot \boldsymbol{v}$
15: $\quad$ **return** $\begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \end{bmatrix}$
16: **end if**

For iterative methods which are applied directly to the normal equations given in equation (2.41) we also need an algorithm which implements the product of $J^T J$ with a vector $\boldsymbol{v}$. We apply the function `multJ` given in Pseudocode 5.

Pseudocode 5 MULTJTJ.M

1: $\boldsymbol{Jv} \leftarrow \texttt{multJ}(\ldots, \texttt{v}, '\texttt{notransp}')$;

2: $\boldsymbol{J^T Jv} \leftarrow \texttt{multJ}(\ldots, \texttt{Jv}, '\texttt{transp}')$;

3: **return** $\boldsymbol{J^T Jv}$

Here, we hand over the vector $\boldsymbol{v}$ to the function `multJ` and get the matrix vector product $\boldsymbol{y} = J \cdot \boldsymbol{v}$. After that, `multJ` is evaluated for the second time with the option `'transp'` which calculates the product of J^T with the vector $\boldsymbol{y}$ to get the required result $J^T J \cdot \boldsymbol{v}$.

2.3.5 Smoothness regularization

Inverse problems are often unstable and ill-posed in the sense of Hadamard (Engl et al., 2000), which means that one of the following conditions is violated:

- A solution exists,
- the solution is unique,
- the solution depends continuously on the data.

The violation of the last condition leads to huge deviations in the result for small variations in the initial data. An appropriate regularization operator stabilizes the inversion procedure and provides additional information to avoid ambiguities. Consequently, regularization can be used to find a solution of the inverse problem which is less sensitive to perturbations by enforcing smoothness and suppressing unwanted oscillations or noisy data (Aster et al., 2013).

Again, we state the minimization problem given in equation (2.37) which combines the data residual and a regularization functional:

$$\Phi(\boldsymbol{m}) = \underbrace{\tfrac{1}{2}\|Q\boldsymbol{u} - \boldsymbol{b}\|_2^2}_{\text{data residual}} + \beta \underbrace{R(\boldsymbol{m} - \boldsymbol{m}_{\text{ref}})}_{\text{regularization functional}} \rightarrow \min_{\boldsymbol{m}} \tag{2.52}$$

subject to $A(\boldsymbol{m})\boldsymbol{u} = \boldsymbol{f}$. Our inversion approach is based on a finite element discretization of the potential equation (2.2) using a piecewise constant representation of the conductivity model. This requires a regularization functional applicable to piecewise constant model parameters on unstructured grids. We have implemented a smoothness regularization

(Schwarzbach and Haber, 2013) in which the penalty function measures the norm of a weak gradient of the conductivity field. Its continuous formulation reads

$$R(m) = \tfrac{1}{2} \int_\Omega |\nabla(m - m_{\text{ref}})|^2 \, dV, \tag{2.53}$$

with $m \in H^1(\Omega)$. Because the parameters m are piecewise constant and not differentiable across element boundaries, we use a generalized formulation to minimize this functional with respect to m. As laid out in Brezzi and Fortin (1991) the mixed (or primal-dual, Strang (1986)) formulation of the problem of minimizing $R(m)$ is fo find a stationary point of

$$\Phi(m, \boldsymbol{p}) = \; -\beta \left(\tfrac{1}{2} \int_\Omega |\boldsymbol{p}|^2 \, dV + \int_\Omega (m - m_{\text{ref}}) \nabla \cdot \boldsymbol{p} \, dV \right), \tag{2.54}$$

where

- $m \in L^2(\Omega)$
- $\boldsymbol{p} \in H_0(\text{div}; \Omega) = \{\boldsymbol{p} \in L^2(\Omega)^3; \nabla \cdot \boldsymbol{p} \in L^2(\Omega); \boldsymbol{n} \cdot \boldsymbol{p}|_{\partial\Omega} = 0\}$.

For m, $m_{\text{ref}} \in H^1(\Omega)$, our regularization operator (2.53) is equal to this mixed formulation or more specifically to the saddle point problem

$$\inf_{\boldsymbol{p} \in H_0(\text{div};\Omega)} \sup_{m \in L^2(\Omega)} \left(\tfrac{1}{2} \int_\Omega |\boldsymbol{p}|^2 \, dV + \int_\Omega (m - m_{\text{ref}}) \nabla \cdot \boldsymbol{p} \, dV \right). \tag{2.55}$$

The divergence theorem and especially equation (2.16) yield

$$\int_\Omega m \nabla \cdot \boldsymbol{p} \, dV = - \int_\Omega \nabla m \cdot \boldsymbol{p} \, dV$$

since $\boldsymbol{n} \cdot \boldsymbol{p} = 0$ on $\partial\Omega$. This changes equation (2.55) to

$$\inf_{\boldsymbol{p} \in H_0(\text{div};\Omega)} \sup_{m \in L^2(\Omega)} \left(\tfrac{1}{2} \int_\Omega |\boldsymbol{p}|^2 \, dV - \int_\Omega \nabla(m - m_{\text{ref}}) \cdot \boldsymbol{p} \, dV \right).$$

To get the infimum, we establish the derivative with respect to $\boldsymbol{p}$ and set it to zero:

$$\begin{aligned} \boldsymbol{p} - \nabla(m - m_{\text{ref}}) &\stackrel{!}{=} \boldsymbol{0} \\ \boldsymbol{p} &= \nabla(m - m_{\text{ref}}). \end{aligned}$$

Using this equivalence, we obtain

$$\sup_{m \in L^2(\Omega)} \; - \tfrac{1}{2} \int_\Omega |\nabla(m - m_{\text{ref}})|^2 \, dV.$$

Changing the sign, we finally get

$$\inf_{m \in L^2(\Omega)} \tfrac{1}{2} \int_\Omega |\nabla(m - m_{\text{ref}})|^2 \, dV$$

which is our regularization functional $R(m)$. Although both formulations are equivalent for $m, m_{\text{ref}} \in H^1(\Omega)$, equation (2.53) includes derivatives of the parameters m which are not continuous across element boundaries in our parameter model. In contrast, the dual formulation only contains derivatives of the dual variable $\boldsymbol{p}$ which fits our piecewise constant conductivity $m \in L^2(\Omega)$. Therefore, we base our regularization on the dual formulation.

Because this mixed formulation contains variables defined in different variational spaces, we use mixed finite element subspaces to approximate the variables m and $\boldsymbol{p}$ (Brezzi and Fortin (1991), Arnold et al. (2010)). The parameters m will be approximated by piecewise constant basis functions because we assume that the conductivity will not change within a single element. To achieve a conforming discretization for the dual variable $\boldsymbol{p}$ which is defined in $H_0(\text{div}; \Omega)$ where normal components vanish on the boundary, we use the divergence conforming Raviart-Thomas (RT) elements of lowest order (RT_0). This ensures continuity of normal components across elements. Figures 2.14 and 2.15 show the DOFs for RT elements of different order and the vector-valued basis functions for the RT_0 elements in 2D. The derivation of DOFs and basis functions is explained in Braess (2003). Figure 2.15 shows the basis functions for the reference triangle, each associated with one edge of the triangle.

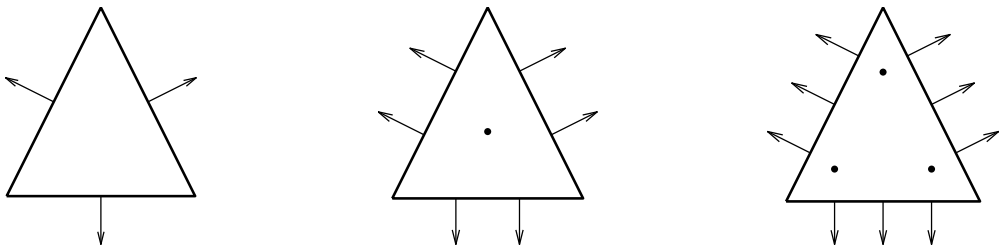

Fig. 2.14: Degrees of freedom for RT functions of order 0 (left), 1 (middle) and 2 (right).

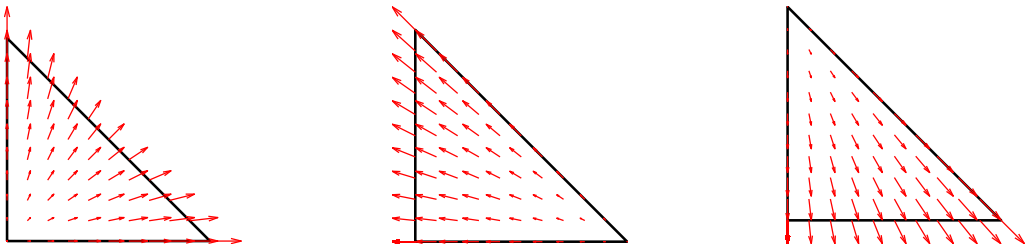

Fig. 2.15: Basis functions of RT_0 elements: $\boldsymbol{\phi}_1(\hat{\boldsymbol{x}}) = [\hat{x}, \hat{y}]^T$ (left), $\boldsymbol{\phi}_2(\hat{\boldsymbol{x}}) = [\hat{x} - 1, \hat{y}]^T$ (middle) and $\boldsymbol{\phi}_3(\hat{\boldsymbol{x}}) = [\hat{x}, \hat{y} - 1]^T$ (right).

They have a nonzero normal component on the associated edge and on the other two edges the normal component is zero, hence there are only tangential components. In contrast to Lagrange elements where the transformation of basis functions from the reference element to an arbitrary element is rather simple (equation (2.33)), we have to use the so-called Piola transform given by

$$\boldsymbol{\phi}(\boldsymbol{x}) := \frac{1}{\det B_K} B_K \hat{\boldsymbol{\phi}}(\hat{\boldsymbol{x}}(\boldsymbol{x}))$$

in order to preserve the normal components and ensure their continuity between neighboring elements. Figure 2.16 illustrates the basis functions associated with the same edge in two

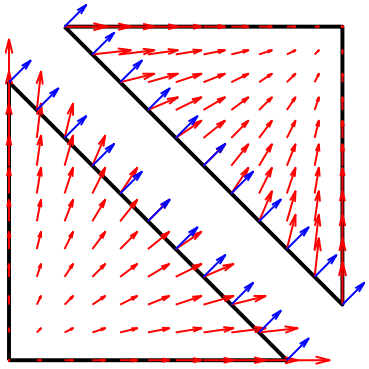

Fig. 2.16: Correct orientation of RT_0 basis functions between neighboring elements.

adjacent triangles. The normal components (blue arrows) are continuous across the element boundary. In our code, we ensure this orientation within the function `orientationRT0`.

As given in equation (2.24) for Lagrange elements, we represent the approximated dual variable $\boldsymbol{p}^h \approx \boldsymbol{p}$ as a linear combination of the RT_0 basis functions $\boldsymbol{\phi}$:

$$\boldsymbol{p}^h = \sum_{j=1}^{n} \xi_j \boldsymbol{\phi}_j,$$

with the scalar coefficients ξ_j. Using this expression and the piecewise constant representation

$$m^h = \sum_{j=1}^{n} m_j q_j, \quad q \in L^2(\Omega),\ q_j = \text{const.} \equiv 1$$

for the parameters m, the discrete representation of equation (2.54) then reads:

$$\Phi(\boldsymbol{m}, \boldsymbol{\xi}) = -\tfrac{\beta}{2} \boldsymbol{\xi}^T M \boldsymbol{\xi} - \beta(\boldsymbol{m} - \boldsymbol{m}_{\text{ref}})^T D \boldsymbol{\xi}. \tag{2.56}$$

M denotes the RT_0 mass matrix and D the discrete divergence operator with

$$M_{i,j} = \int_\Omega \boldsymbol{\phi}_i \cdot \boldsymbol{\phi}_j \, d\boldsymbol{x}$$
$$D_{i,j} = \int_\Omega q_i \nabla \cdot \boldsymbol{\phi}_j \, d\boldsymbol{x} = \int_\Omega \nabla \cdot \boldsymbol{\phi}_j \, d\boldsymbol{x}, \quad q_i = 1 \, \forall \, K \in \mathscr{T}_h.$$

In order to minimize the objective function (2.56) the gradient of $\Phi(\boldsymbol{m}, \boldsymbol{\xi})$ with respect to the dual coefficient vector $\boldsymbol{\xi}$ has to vanish:

$$\nabla_{\boldsymbol{\xi}} \Phi(\boldsymbol{m}, \boldsymbol{\xi}) = -\beta M \boldsymbol{\xi} - \beta D^T (\boldsymbol{m} - \boldsymbol{m}_{\mathsf{ref}}) = \mathbf{0}.$$

This results in

$$\boldsymbol{\xi} = -M^{-1} D^T (\boldsymbol{m} - \boldsymbol{m}_{\mathsf{ref}}).$$

Using this relation, we eliminate the dual variable $\boldsymbol{\xi}$ and arrive at the final objective function:

$$\Phi(\boldsymbol{m}) = \tfrac{\beta}{2} (\boldsymbol{m} - \boldsymbol{m}_{\mathsf{ref}})^T D M^{-1} D^T (\boldsymbol{m} - \boldsymbol{m}_{\mathsf{ref}}) \to \min_{\boldsymbol{m}}.$$

To expose the relationship to the normal equations (2.41), we expand the general formulation of the regularization operator given in equation (2.37):

$$\begin{aligned} \tfrac{\beta}{2} ||W(\boldsymbol{m} - \boldsymbol{m}_{\mathsf{ref}})||_2^2 &= \tfrac{\beta}{2} (W(\boldsymbol{m} - \boldsymbol{m}_{\mathsf{ref}}))^T \cdot (W(\boldsymbol{m} - \boldsymbol{m}_{\mathsf{ref}})) \\ &= \tfrac{\beta}{2} (\boldsymbol{m} - \boldsymbol{m}_{\mathsf{ref}})^T W^T W (\boldsymbol{m} - \boldsymbol{m}_{\mathsf{ref}}). \end{aligned}$$

Using this formulation it is easy to see that the matrix $W^T W$ can be replaced by $D M^{-1} D^T$. If we want to apply an iterative method to the original least squares problem as described in equation (2.42) we need to calculate the matrix W and therefore decompose $D M^{-1} D^T$. This can be done using a Cholesky decomposition of M calculated by the MATLAB function `chol`: With

$$M \overset{\texttt{chol}}{=} L L^T$$

we set

$$W := L^{-1} D^T$$

which yields

$$\begin{aligned} W^T W &= D L^{-T} L^{-1} D^T \\ &= D \left(L L^T \right)^{-1} D^T \\ &= D M^{-1} D^T. \end{aligned}$$

Now, we are able to apply the smoothness regularization within our inversion algorithm using a direct solver as well as an iterative method.

2.3.6 Damping

Each Gauss-Newton scheme yields a direction of descent $\Delta \boldsymbol{m}_k$ by evaluating the system of normal equations (see Schwetlick (1979) and Section 2.3.1). Now, we want to apply a simple damping algorithm to the Gauss-Newton iteration in order to add only a sufficiently small amount of the direction $\Delta \boldsymbol{m}_k$ to ensure the convergence to the desired minimum. Therefore, the search direction is multiplied by a factor α and the new parameter vector is given by $\boldsymbol{m}_{k+1} = \boldsymbol{m}_k + \alpha \Delta \boldsymbol{m}_k$. We calculate the synthetic data $\boldsymbol{u}$ and the data residual using the new approximation $\boldsymbol{m}_{k+1}$ and compare it with the data residual of the previous iteration step. α starts at a value of 1 and is reduced as long as the following criterion is fulfilled:

$$||\boldsymbol{b} - Q\boldsymbol{u}(\boldsymbol{m}_{k+1})|| > \left(1 - \tfrac{1}{4}\alpha\right) ||\boldsymbol{b} - Q\boldsymbol{u}(\boldsymbol{m}_k)||.$$

The damping algorithm is implemented as follows:

Pseudocode 6 DAMPED GAUSS-NEWTON SCHEME

Choose $\boldsymbol{m}_0$
for $k = 0, 1, 2, \ldots$ **do**
  $J_k \leftarrow \boldsymbol{u}'(\boldsymbol{m}_k)$
  $\boldsymbol{r}_k \leftarrow \boldsymbol{b} - Q\boldsymbol{u}(\boldsymbol{m}_k)$
  Solve $||\boldsymbol{r}_k - J_k \Delta \boldsymbol{m}_k|| \rightarrow \min_{\Delta \boldsymbol{m}_k}$
  $\alpha \leftarrow 1$
  $\boldsymbol{m} \leftarrow \boldsymbol{m}_k + \alpha \Delta \boldsymbol{m}_k$
  Solve $A(\boldsymbol{m})\boldsymbol{u} = \boldsymbol{f}$
  while $||\boldsymbol{b} - Q\boldsymbol{u}(\boldsymbol{m})|| > \left(1 - \tfrac{1}{4}\alpha\right) ||\boldsymbol{r}_k||$ **do**
    $\alpha \leftarrow \frac{1}{2}\alpha$
    if $\alpha < 2^{-10}$ **then**
      return $\boldsymbol{m}_k$
    end if
    $\boldsymbol{m} \leftarrow \boldsymbol{m}_k + \alpha \Delta \boldsymbol{m}_k$
    Solve $A(\boldsymbol{m})\boldsymbol{u} = \boldsymbol{f}$
  end while
  $\boldsymbol{m}_{k+1} \leftarrow \boldsymbol{m}$
end for

2.3.7 Matrix analysis and inversion results

We discussed the theory of all properties and features of the inversion algorithm of the DC resistivity code in the previous sections. In order to show the need for regularization and validate the functionality of the inversion algorithm, we apply the code to a synthetic model problem and try to reconstruct the conductivity distribution shown in Figure 2.17 from a synthetic apparent resistivity data set. We want to point out that the model is a scientific example developed to validate the inversion algorithm. Its dimensions as well as the large source-receiver configuration are only a matter of scaling. The conductivities of the homogeneous

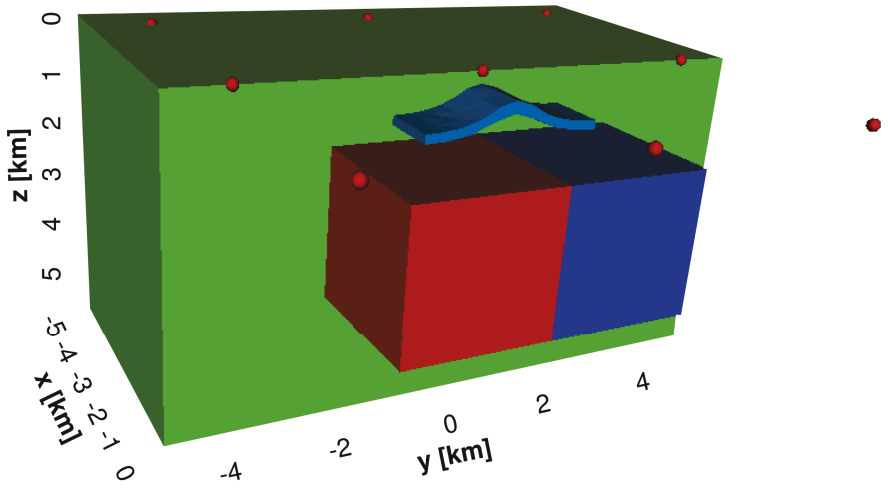

Fig. 2.17: Synthetic model: $\sigma_1 = 0.1\ \frac{S}{m}$ (green), $\sigma_2 = 0.01\ \frac{S}{m}$ (top blue), $\sigma_3 = 1\ \frac{S}{m}$ (red) and $\sigma_4 = 0.002\ \frac{S}{m}$ (bottom blue).

background and the embedded bodies are given in the caption of the figure. The red dots mark nine source positions we use. First of all, we want to investigate the properties of the system matrix of the normal equations (2.41), which we have to use within the Gauss-Newton scheme:

$$J^T J + \beta W^T W.$$

The first part contains the sensitivity matrix J which depends on parameters and data and the second part is a regularization term which changes with the value of the regularization parameter β. The properties of the system matrix regarding the solvability of the linear system of equations can be described by the set of eigenvalues and the absolute ratio of largest and smallest eigenvalue, respectively (Hansen, 2010). The inverse problem can be ill-conditioned or even ill-posed if the condition number is too high or even infinite. In the following, we calculate the condition number of $J^T J$ for the model given in Figure 2.17 with a relatively

coarse mesh of about 13 000 tetrahedra. We use just one source at $\boldsymbol{x}_0 = [-4, 0, 0]$ km and 289 measuring points at the Earth's surface ($z = 0$ km) with x- and y-coordinates between minus four and four kilometers and a step size of 500 meters. The starting model $\boldsymbol{m}_0$ is a constant vector containing the logarithmized source conductivity $\sigma_0 = 0.1\ \frac{\mathrm{S}}{\mathrm{m}}$.

The MATLAB function `cond` returns `Inf` for the condition number of the matrix $J^T J$, which is the representation of positive infinity. Hence, this number is very large which characterizes a severely ill-posed problem. Therefore, we add the smoothness regularization operator $\beta W^T W = \beta D M^{-1} D^T$ (see Section 2.3.5) to improve the condition of the linear system of equations. The assembly of the regularization matrix $DM^{-1}D^T$ requires a lot of computing time because it includes the solution of a linear system with the RT_0 mass matrix M which is very large (dimension: $n_{\mathrm{faces}} \times n_{\mathrm{faces}}$). Therefore, we choose to use only the diagonal of M for the following calculations:

$$M^* = \begin{bmatrix} m_{1,1} & & & \\ & m_{2,2} & & \\ & & \ddots & \\ & & & m_{n_{\mathrm{faces}},n_{\mathrm{faces}}} \end{bmatrix}.$$

The number of nonzero entries of M^* is decreased by a factor of seven in comparison with the full mass matrix M. As a consequence, the ratio of nonzero entries between $W^T W$ assembled with the full matrix M and assembled with its diagonal counterpart is $3{\cdot}10^3$ which results in a much shorter time for solving linear systems of equations with this matrix. In addition, we expanded the regularization matrix by a diagonal matrix which contains the cell volumes on the main diagonal:

$$W = \begin{bmatrix} k_1 W_{\mathrm{SR}} \\ k_2 I \end{bmatrix}$$

with the smoothness matrix W_{SR} and the diagonal matrix

$$I = \begin{bmatrix} v_1 & & & \\ & v_2 & & \\ & & \ddots & \\ & & & v_n \end{bmatrix}$$

with the cell volumes $v_1, v_2, \ldots, v_n$. The scaling factors k_1 and k_2 define the ratio between the two matrices. We chose to use this combination of matrices for the regularization operator in order to construct a model that is close to a known reference model on the one hand and varies smoothly between neighboring elements on the other hand. We investigate the influence of the scaling factors on the inversion result later on in this section.

The choice of the regularization parameter β is another difficult issue, because we have to find a compromise between fulfilling the data residual norm on the one hand and the regularization norm on the other hand (equation (2.37)). A large β forces the solution to be smooth and neglects data matching. For the following inversion results we chose to balance the data residual and regularization norm and use the ratio of both as a starting value for β:

$$\beta_1 = \frac{||Q\boldsymbol{u} - \boldsymbol{b}||_2}{||W(\boldsymbol{m} - \boldsymbol{m}_{\text{ref}})||_2}. \tag{2.57}$$

This ensures the correct search direction in the beginning of the inversion algorithm and stabilizes the inversion. Throughout the Gauss-Newton iteration, we apply a continuation or cooling approach (Newman and Hoversten, 2000) and decrease β in each Gauss-Newton step k to give more importance to the data residual norm as the iteration progresses:

$$\beta_k = \max\left\{\frac{\beta_{k-1}}{10}, \beta_{\text{min}}\right\} \quad \text{with } k = 2, \ldots, n.$$

We want to avoid a situation where the smoothness regularization has almost no influence and the inversion result becomes noisy: The smallest regularization parameter β_{min} has to be chosen according to the noise level of the input data.

The model given in Figure 2.17 was coarsely discretized into about 13 000 elements and allowed the parameter $\boldsymbol{m}$ to vary on each tetrahedron. We use one source at $\boldsymbol{x}_0 = [4, 0, 0]$ km and a uniform grid of receiver locations at the Earth's surface between -4 and 4 kilometers in x- and y-direction with a distance of 500 meters between the single receivers. This configuration generates about 250 data points to which we add three percent random noise to avoid an *inverse crime* (Kaipio and Somersalo (2007), Mueller and Siltanen (2012) and Section 3.3.2.1). For the starting model $\boldsymbol{m}_0$ we chose to use a constant vector containing the logarithmized mean of the true parameter distribution given by $\sigma_{\text{start}} = 0.15\ \frac{\text{S}}{\text{m}}$. The reference model $\boldsymbol{m}_{\text{ref}}$ was set to the logarithm of the background conductivity $\sigma = 0.1\ \frac{\text{S}}{\text{m}}$ and we calculate $\beta_1 \approx 69$ according to equation (2.57) and set $\beta_{\text{min}} = 10^{-9}$. First of all, we present the development of the relative data residual norm r given by

$$r_{\text{rel}} = \frac{||Q\boldsymbol{u}_k - \boldsymbol{b}||_2}{||Q\boldsymbol{u}_1 - \boldsymbol{b}||_2} \tag{2.58}$$

in the kth Gauss-Newton step. The different curves in Figure 2.18 are calculated for different

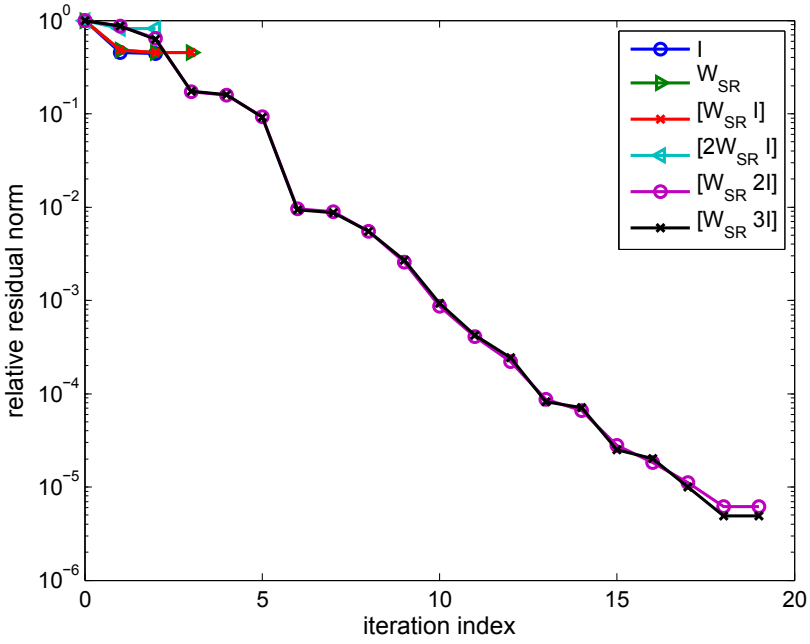

Fig. 2.18: Development of the relative data residual norm during the inversion of data generated from one source for different weighting matrices.

ratios of the diagonal and smoothness matrix in the regularization operator W. The iteration terminated once the new search direction yields no improvement to the solution according to the damping criterion described in Pseudocode 6. The figure shows that it is not to easy to find an appropriate composition of the regularization matrix W. Almost all combinations stopped within the damping algorithm during the first two or three Gauss-Newton steps. The successful iterations give more weight to the diagonal matrix, which seems to be essential for the regularization term. On the other hand, it is not possible to neglect the smoothness regularization matrix W_{SR}. From the residual curves we are able to select the scaling factors $k_1 = 1$ for the smoothness matrix and $k_2 > 1$ for the diagonal matrix. The final value for k_2 was determined by the resulting parameter model which was not smooth enough for values larger than two. Therefore, we chose to use $k_2 = 2$ (violet line) for the Gauss-Newton iteration with one source and show a frontal slice of the inversion result at $x = 0$ km in Figure 2.19.

Although we are only using a single source and 289 data points for 13 000 parameters which is a factor of about 45, the resolution of the underlying conductivity model is quite good: We can get an idea of the two large blocks. The resistive one (blue) on the right-hand side appears to be larger than the more conductive block on the left. The small bent layer

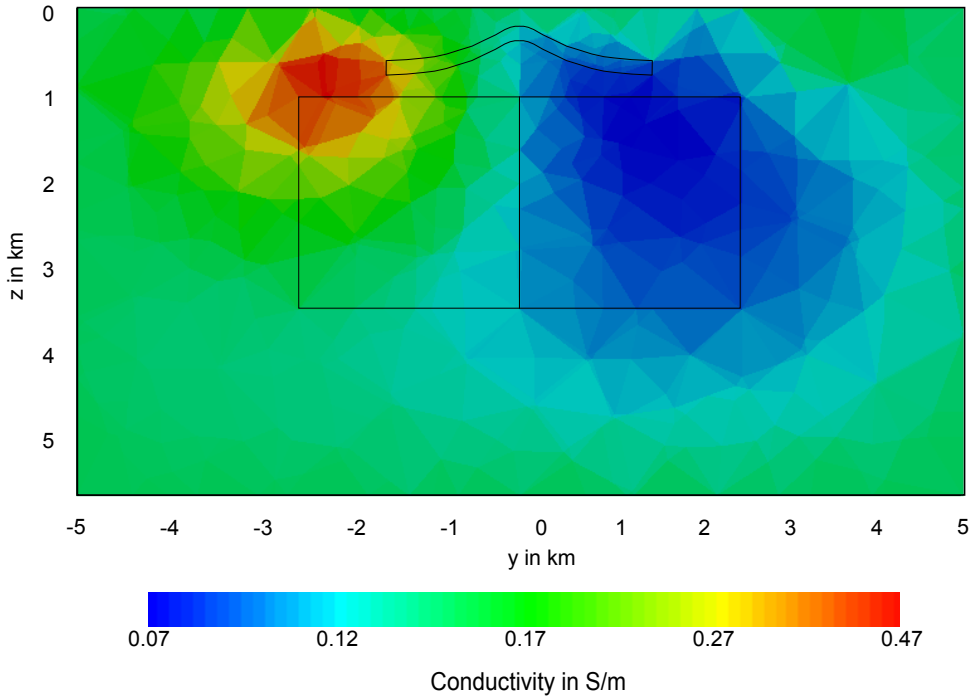

Fig. 2.19: Inversion result for one source.

close to the surface is not seen yet or is connected with the resistive block. Furthermore, the resistive block smears out at greater depths because we do not get any information about its boundaries with the source-receiver configuration we used. A source in a borehole might overcome these difficulties.

If we use more sources and therefore, more data points and complementary information, the inverse problem changes a lot. All control variables such as the regularization parameter β, the weighting factors k_1 and k_2 as well as the damping parameter α might have to be chosen totally different. Additional sources yield additional information about the subsurface and might help to get a more precise image of the buried bodies. We expanded the number of sources to a regular grid of nine point sources as it is shown by the red dots in Figure 2.17. The receiver nodes stayed the same. The iteration was terminated once we reached a relative data residual norm of 10^{-4}. Again, we calculate the starting value for the regularization parameter according to equation (2.57): $\beta_1 \approx 71$. We apply the continuation approach and set the lower bound to be 10^{-8}. Figure 2.20 shows a frontal slice of the inversion result at $x = 0\,\text{km}$. As the number of sources is increased, the reconstruction of the embedded conductive and resistive bodies becomes progressively more accurate because we add additional information about the subsurface. The exact shape of the two blocks as well as the thin bent layer was reconstructed quite well. Figure 2.21 shows the development of the relative data

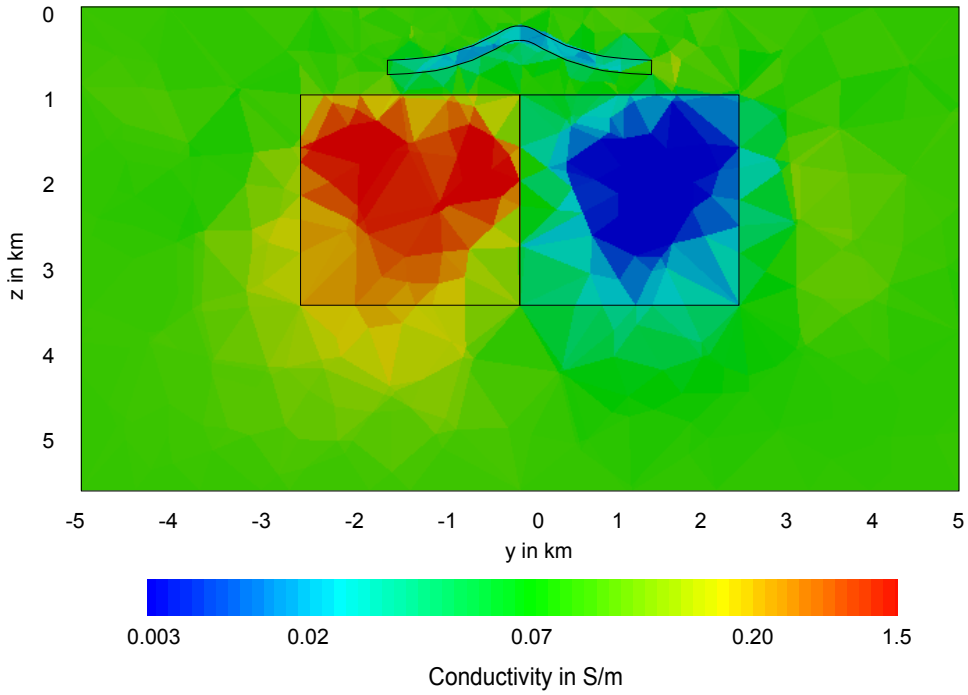

Fig. 2.20: Inversion result for the nine sources shown in Figure 2.17.

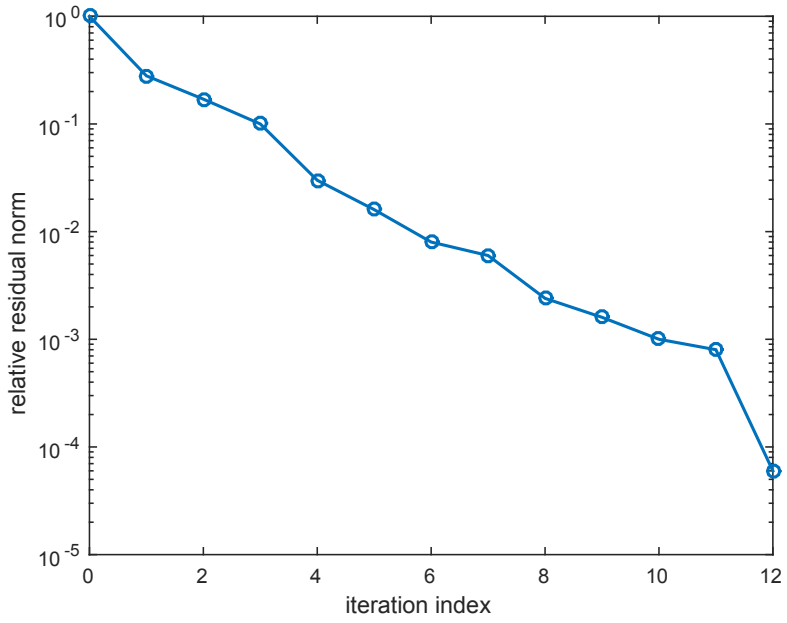

Fig. 2.21: Development of the relative data residual norm during the Gauss-Newton iteration.

residual norm as defined in equation (2.58). Even for a relatively small lower bound of β the regularization operator stabilized the iteration and guaranteed the convergence of the Gauss-Newton scheme to a sufficiently smooth model.

In the previous sections we explained the physical and mathematical background and the implementation of our DC resistivity forward operator and inversion scheme. Overviews of the structure and main input parameters of the code are given in Appendix A.5.1. Our code can be used to do sensitivity studies in order to find an appropriate configuration in preparation of a measurement. Furthermore, we are able to invert DC resistivity data to reconstruct the underlying conductivity distribution. Although we can vary the measurement configuration and add further sources and receivers to get complementary data including deeper structures, the resolution of the DC resistivity method is limited. Therefore, we explain the combination of DC resistivity data with an electromagnetic data set in Chapter 4.

3 Electromagnetic modeling and inversion

In the following chapter, we develop and implement a forward and inversion algorithm for electromagnetic methods. Here, we derive the fundamental equation which is almost equal for all EM methods except for the right-hand side given by the mathematical representation of the particular source configuration. In case of the magnetotelluric (MT) method, there are no artificial sources and the source term is equal to zero. Therefore, we deduce all algorithms for the MT problem to establish a basic instrument for modeling and inversion of electromagnetic data. Then, an enhancement for any other electromagnetic method can be achieved by simply implementing another source in the form of the right-hand side of the curl-curl equation (see equation (3.7)).

The use of the MT method goes back to the 1950s and was mentioned for example by Cagniard (1953) and Vozoff (1990) explained the main principles later on. Just like for the DC resistivity method (Chapter 2) the number of 3D electromagnetic and especially MT surveys increased (Heise et al. (2008), Patro and Egbert (2008)) and this goes along with a higher demand for a continuous improvement of the evaluation software. For example Schmucker and Weidelt (1975) have introduced the integral equation technique and there are some forward modeling algorithms using this technique such as Xiong (1992) or Zhdanov et al. (2000). Furthermore, a number of inversion algorithms – mainly based on finite difference or finite element forward algorithms – have been developed for example by Mackie and Madden (1993), Newman and Alumbaugh (2000), Sasaki (2004), Siripunvaraporn and Egbert (2007), Avdeev and Avdeeva (2009) and Siripunvaraporn and Sarakorn (2011) and an application to mineral exploration data was given by Farquharson and Craven (2009). The most frequent inversions schemes are Gauss-Newton (Sasaki, 2004) and quasi-Newton schemes (Haber (2005), Avdeev and Avdeeva (2009)) as well as some variations regarding efficiency or improved storage requirements (Siripunvaraporn et al., 2005). Here, we present a 3D MT forward modeling and inversion code for one polarization direction. It is based on a discretization using unstructured grids and – as opposed to the discretization of the DC resistivity problem – we use the vector-valued Nédélec elements. The inversion algorithm works with complex field quantities and is based on a regularized Gauss-Newton scheme as well. We demonstrate the accuracy with both scientific and realistic data sets.

3.1 Physical principles

In contrast to the DC resistivity method, electromagnetic methods are based on natural or artificial time-varying source fields and measure electric and/or magnetic fields. Electromagnetic fields are described by four complex vector functions (Monk, 2003), which are

- the electric field intensity $\boldsymbol{e}$ in $\frac{\mathrm{V}}{\mathrm{m}}$,
- the electric displacement $\boldsymbol{d}$ in $\frac{\mathrm{As}}{\mathrm{m}^2}$,
- the magnetic field intensity $\boldsymbol{h}$ in $\frac{\mathrm{A}}{\mathrm{m}}$,
- and the magnetic induction $\boldsymbol{b}$ in $\mathrm{T}= \frac{\mathrm{Vs}}{\mathrm{m}^2}$

as well as Maxwell's equations in the time domain which relate these vector functions with each other:

$$\nabla \times \boldsymbol{e} = -\frac{\partial \boldsymbol{b}}{\partial t}$$

$$\nabla \cdot \boldsymbol{d} = q \tag{3.1}$$

$$\nabla \times \boldsymbol{h} = \boldsymbol{j} + \frac{\partial \boldsymbol{d}}{\partial t} \tag{3.2}$$

$$\nabla \cdot \boldsymbol{b} = 0$$

with the electric charge density q in $\frac{\mathrm{As}}{\mathrm{m}^3}$ and the electric current density $\boldsymbol{j}$ in $\frac{\mathrm{A}}{\mathrm{m}^2}$. The first equation is called Faraday's law and describes the evolution of eddy currents when the magnetic field changes with time. Equation (3.1) – Gauss' law – shows the influence of the electric charge density on the electric displacement. The third equation is Ampère's circuital law which describes the connectedness of the magnetic eddy current field and the ohmic and displacement current (Nabighian and Macnae, 1988a). The fourth equation explains that the magnetic field lines are solenoidal and the magnetic induction has no sources. Because Maxwell's equations are decoupled, we need additional constitutive equations that depend on the medium the electromagnetic field is in:

$$\boldsymbol{b} = \mu \boldsymbol{h} = \mu_0 \mu_r \boldsymbol{h},$$

$$\boldsymbol{d} = \varepsilon \boldsymbol{e} = \varepsilon_0 \varepsilon_r \boldsymbol{e},$$

$$\boldsymbol{j} = \sigma \boldsymbol{e},$$

with the vacuum and relative magnetic permeability μ_0 and μ_r in $\frac{\mathrm{Vs}}{\mathrm{Am}}$, the vacuum and relative electrical permittivity ε_0 and ε_r in $\frac{\mathrm{As}}{\mathrm{Vm}}$ and the electrical conductivity σ in $\frac{\mathrm{S}}{\mathrm{m}}$. In the following isotropic considerations, we set the relative values of μ and ε to be equal to one and use the vacuum permeability and permittivity with

$$\mu = \mu_0 = 4\pi \cdot 10^{-7} \, \frac{\mathrm{Vs}}{\mathrm{Am}}$$
$$\varepsilon = \varepsilon_0 \approx 8.854 \cdot 10^{-12} \, \frac{\mathrm{As}}{\mathrm{Vm}}.$$

The spreading of electromagnetic fields is influenced by the skin effect which is caused by the attenuation during the propagation of the electromagnetic fields through rocks and structures in the subsurface. In consequence, we can observe an exponential decay of the source field's amplitude with growing distance from the source. A very important physical quantity is the skin depth which is a characteristic depth where the source field has decayed to the e-th part of the surface amplitude (ibid.):

$$\begin{aligned} \delta &= \sqrt{\frac{2}{\sigma\mu\omega}} \\ &\approx 503\sqrt{\frac{1}{\sigma f}}, \end{aligned} \tag{3.3}$$

with the frequency f and the angular frequency $\omega = 2\pi f$. This depth yields important information for designing the virtual experiment, for example the size of the modeling area. There is a rule of thumb which says that this area has to be at least as large as five times the skin depth to ensure a full decayment of the initial source field. Furthermore, we can suppose that the largest amount of current density, and therefore the most interesting part of the measurement, is located in the meter range below the surface which is characterized by the skin depth. Hence, the sensitivity distribution of the source-receiver configuration and the skin depth are directly related.

Electromagnetic methods are called passive methods if they are using natural sources, such as current systems in the ionosphere and magnetosphere which generate magnetic fields or artificial sources which are far away from the investigated area such as for the MT method. In contrast, active electromagnetic methods, such as the controlled-source electromagnetic (CSEM) method, use artificial, near-field (portable) sources to generate the electromagnetic primary (transmitting) field (Knödel et al., 2005a), which induces currents inside the Earth. The response of the underlying conductivity structures is called secondary field and is measured with the receivers. The inductive coupling of the sources and receivers enables measurements in areas which might be inaccessible for methods which need galvanic coupling such as DC

resistivity methods. The most important source and receiver types are the horizontal and vertical electric dipole (HED and VED) or the horizontal and vertical magnetic dipole (HMD and VMD) and combinations of them.

3.2 Electromagnetic forward modeling

3.2.1 Governing equations

We deduce the MT problem in the frequency domain for one polarization direction with an electric field oriented in x-direction given by

$$\boldsymbol{E} = \begin{bmatrix} E_x(z) \\ 0 \\ 0 \end{bmatrix}. \tag{3.4}$$

With a time-dependence of $\boldsymbol{E}$ and $\boldsymbol{H}$ given by $e^{i\omega t}$ and the derivatives $\frac{\partial \boldsymbol{E}}{\partial t} = i\omega \boldsymbol{E}$ as well as $\frac{\partial \boldsymbol{H}}{\partial t} = i\omega \boldsymbol{H}$, the third Maxwell equation – Faraday's law (equation (3.2)) – is given by

$$\nabla \times \boldsymbol{E} = -i\omega\mu \boldsymbol{H}, \tag{3.5}$$

with the angular frequency $\omega = 2\pi f$ and the imaginary unit i. Applying the curl operator to the previous equation, we get

$$\nabla \times \nabla \times \boldsymbol{E} = -i\omega\mu \nabla \times \boldsymbol{H}.$$

Inserting Ampère's law given by

$$\nabla \times \boldsymbol{H} = \sigma \boldsymbol{E} \tag{3.6}$$

we obtain the homogeneous Helmholtz equation for the electric field $\boldsymbol{E}$:

$$\nabla \times \nabla \times \boldsymbol{E} - k^2 \boldsymbol{E} = \boldsymbol{0} \tag{3.7}$$

with

$$k^2 = -i\omega\mu\sigma.$$

To deduce the analytical solution of this model problem for a homogeneous halfspace we set

$$\boldsymbol{H} = [0, 1, 0]^T \text{A/m}$$

for a constant magnetic source field which is equal to a plane wave in the air halfspace ($z \leq 0$). Then, the magnetic field fulfills the following homogeneous Helmholtz equation in a homogeneous halfspace ($z \geq 0$):

$$\partial_{zz}^2 H_y(z) + k^2 H_y(z) = 0, \quad k^2 = -i\omega\mu\sigma,$$

where σ is the conductivity of the halfspace. The solution is given by

$$H_y(z) = H_y(0)e^{-ikz}, \quad k = \sqrt{-i\omega\mu\sigma}$$

and with the plane wave $H_y(0) = 1$ we arrive at

$$H_y(z) = \begin{cases} 1 & z < 0 \\ e^{-ikz} & z \geq 0. \end{cases}$$

Applying Ampère's law (3.6), we obtain the electric field in $z > 0$:

$$E_x(z) = \frac{ik}{\sigma} e^{-ikz} \tag{3.8}$$

and for the air halfspace $z < 0$ Faraday's law (3.5) leads to

$$\partial_z E_x = -i\omega\mu.$$

We get the electric field by integration:

$$\int_0^z \partial_z E_x(z) dz = E_x(z) - E_x(0) = -\int_0^z i\omega\mu dz = -i\omega\mu z.$$

Using equation (3.8) we get

$$E_x(z) = E_x(0) - i\omega\mu z = \frac{ik}{\sigma} - i\omega\mu z.$$

To summarize, the analytical solution for our model problem which we can use as inhomogeneous Dirichlet boundary conditions to confirm the correctness of our simulation in a homogeneous halfspace is given by

$$E_x(z) = \begin{cases} \frac{ik}{\sigma} - i\omega\mu z & z < 0 \\ \frac{ik}{\sigma} e^{-ikz} & z \geq 0. \end{cases} \tag{3.9}$$

For a layered halfspace we refer to Wait (1953), Schmucker and Weidelt (1975) and Ward and Hohmann (1988) and calculate the field values on the boundary according to Wait's algorithm in the function `getE1dMT` which was implemented by Ralph-Uwe Börner. We give a brief overview on this algorithm in Appendix A.4.

3.2.2 Finite element approximation – Nédélec elements

3.2.2.1 Variational formulation

As shown in Chapter 2, the finite element method is a powerful tool for the numerical modeling of (geo)physical equations. In contrast to the equation of continuity (equation (2.2)), the electromagnetic PDE (3.7) contains no gradient but a curl operator and the relevant fields are vector fields. Therefore, we need another class of finite elements to discretize this equation appropriately. In Section 2.3.5 we apply the divergence conforming Raviart-Thomas elements, which provide the continuity of normal components across element interfaces. The physical fields we want to approximate by solving the Helmholtz equation (3.7) fulfill continuity conditions across the element interfaces regarding their tangential components. The curl-conforming elements, which ensure these continuity conditions are Nédélec elements (Nédélec, 1980). To deduce a finite element discretization, we derive the variational formulation for the curl-curl equation (3.7).

By analogy with the derivation for the equation of continuity given in Section 2.2.3.1 we need an equivalent to integration by parts for equation (3.7) which contains the curl operator. Monk (2003) uses the divergence theorem 1 to establish the following Corollary:

Corollary 1 (*adapted from Monk (ibid.)*)

Let $\Omega \subset \mathbb{R}^3$ be a bounded Lipschitz domain with boundary $\partial\Omega$ and unit outward normal $\boldsymbol{n}$. Suppose $\boldsymbol{u}$ and $\boldsymbol{v}$ are in $(\mathcal{C}^1(\overline{\Omega}))^3$. Then

$$\int_\Omega \boldsymbol{v} \cdot (\nabla \times \boldsymbol{u})\, d\boldsymbol{x} = \int_\Omega \boldsymbol{u} \cdot (\nabla \times \boldsymbol{v})\, d\boldsymbol{x} + \int_{\partial\Omega} (\boldsymbol{n} \times \boldsymbol{u}) \cdot \boldsymbol{v}_t\, ds,$$

with the tangential component of $\boldsymbol{v}$ given by $\boldsymbol{v}_t = \boldsymbol{n} \times (\boldsymbol{v} \times \boldsymbol{n})$.

To ensure that these integrals are defined, the vector fields $\boldsymbol{u}$ and $\boldsymbol{v}$ as well as $\nabla \times \boldsymbol{u}$ and $\nabla \times \boldsymbol{v}$ have to be square-integrable. Appropriate function spaces are

$$\begin{aligned} \boldsymbol{L}^2(\Omega) &:= \left\{ \boldsymbol{u} \,:\, \Omega \to \mathbb{C}^3 \,:\, \int_\Omega |\boldsymbol{u}|^2\, d\boldsymbol{x} < \infty \right\} = L^2(\Omega)^3, \\ \boldsymbol{H}(\mathrm{curl};\Omega) &:= \left\{ \boldsymbol{u} \in \boldsymbol{L}^2(\Omega) \,:\, \nabla \times \boldsymbol{u} \in \boldsymbol{L}^2(\Omega) \right\}. \end{aligned} \tag{3.10}$$

If there holds

$$\int_\Omega \boldsymbol{u} \cdot (\nabla \times \boldsymbol{\phi})\, d\boldsymbol{x} = \int_\Omega \boldsymbol{v} \cdot \boldsymbol{\phi}\, d\boldsymbol{x}$$

for each differentiable vector field $\boldsymbol{\phi}$ which vanishes on $\partial\Omega$, then the vector field $\boldsymbol{v} \in \boldsymbol{L}^2(\Omega)$ is called weak curl of $\boldsymbol{u} \in \boldsymbol{L}^2(\Omega)$. The compatible boundary condition in $\boldsymbol{H}(\text{curl};\Omega)$ is defined by

$$\boldsymbol{H}_0(\text{curl};\Omega) := \{\boldsymbol{u} \in \boldsymbol{H}(\text{curl};\Omega) \,:\, \boldsymbol{n} \times \boldsymbol{u} = \boldsymbol{0} \text{ on } \partial\Omega\}\,. \tag{3.11}$$

Now, we multiply equation (3.7) by a test function $\boldsymbol{v}$, integrate and apply the partial integration rule given in Corollary 1 to get the variational formulation:

$$\int_\Omega (\nabla \times \boldsymbol{E}) \cdot (\nabla \times \boldsymbol{v})\, d\boldsymbol{x} + \int_{\partial\Omega} \boldsymbol{n} \times (\nabla \times \boldsymbol{E}) \cdot \boldsymbol{v}_t\, ds + i\omega\mu_0 \int_\Omega \sigma(\boldsymbol{x}) \boldsymbol{E} \cdot \boldsymbol{v}\, d\boldsymbol{x} = \boldsymbol{0} \tag{3.12}$$

for all test functions $\boldsymbol{v} \in \boldsymbol{H}(\text{curl};\Omega)$. The ansatz and test spaces are

$$\mathscr{S} := \{\boldsymbol{E} \in \boldsymbol{H}(\text{curl};\Omega) \,:\, \boldsymbol{n} \times \boldsymbol{E} = \boldsymbol{E}_D \text{ on } \Gamma_D\}, \tag{3.13}$$

$$\mathscr{V} := \{\boldsymbol{v} \in \boldsymbol{H}(\text{curl};\Omega) \,:\, \boldsymbol{n} \times \boldsymbol{v} = \boldsymbol{0} \text{ on } \Gamma_D\}. \tag{3.14}$$

As described in the previous section and equation (3.9), the given electromagnetic boundary value problem has only Dirichlet boundary conditions and because we choose the test space $\mathscr{V}$ such that $\boldsymbol{n} \times \boldsymbol{v}$ vanishes on the Dirichlet boundary Γ_D, the second integral in equation (3.12) is equal to zero. Then, the variational formulation of the boundary value problem (3.7) reads as follows:
Find $\boldsymbol{E} \in \mathscr{S}$ such that

$$a(\boldsymbol{E}, \boldsymbol{v}) = l(\boldsymbol{v}) \quad \forall \boldsymbol{v} \in \mathscr{V}$$

with

$$a(\boldsymbol{E}, \boldsymbol{v}) = \int_\Omega \left((\nabla \times \boldsymbol{E}) \cdot (\nabla \times \boldsymbol{v}) + i\omega\mu_0\sigma(\boldsymbol{x})\boldsymbol{E} \cdot \boldsymbol{v}\right) d\boldsymbol{x}, \tag{3.15}$$

$$l(\boldsymbol{v}) = \boldsymbol{0}.$$

3.2.2.2 Derivation of the Galerkin system

As for DC resistivity modeling, we build a triangulation $\mathscr{T}_h$ and decompose the modeling domain Ω into tetrahedral subsets $K \subset \Omega$. We move to finite ansatz and test spaces $\mathscr{S}^h$ and $\mathscr{V}^h$ which are equal except for the boundary conditions and there holds $\mathscr{S}^h = \mathscr{V}^h$ for homogeneous Dirichlet boundary conditions given by $\boldsymbol{E}_D \equiv \boldsymbol{0}$ (equation (3.13) and (3.14)). Hence, we are able to find the solution $\boldsymbol{E}^h$ using basis functions of the n-dimensional subspace $\mathscr{V}^h \subset \mathscr{V}$. The variational formulation changes to

$$\text{Find } \boldsymbol{E}^h \in \mathscr{V}^h \text{ such that } a(\boldsymbol{E}^h, \boldsymbol{v}) = \boldsymbol{0} \quad \forall \boldsymbol{v} \in \mathscr{V}^h. \tag{3.16}$$

Using this formulation, we can derive a discrete representation of our PDE (3.7) based on Nédélec elements. With an appropriate basis $\{\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \boldsymbol{\phi}_n\}$ of $\mathscr{V}^h$, the solution $\boldsymbol{E}^h \in \mathscr{V}^h$ can be described as a linear combination of the basis functions:

$$\boldsymbol{E}^h = \sum_{i=1}^{n} u_i \boldsymbol{\phi}_i, \tag{3.17}$$

where the unique scalar coefficients $\{u_i\}_{i=1}^n$ are directly related to the DOFs of the finite element approximation. In contrast to Lagrange elements, where the DOFs are the field values at the vertices of the element, the DOFs for the vector-valued Nédélec elements are moments on edges or more specifically "the average value of the tangential component of the considered vector field on each edge" (Monk, 2003) for linear elements. Elements of higher order are described by face and volume moments as well. With a polynomial space R_k (equation (3.26)), these moments and the curl-conforming Nédélec elements are defined as follows:

Definition 1 (*Curl-conforming element, adapted from Monk (ibid.)*)

The curl-conforming finite element $(\hat{K}, P_{\hat{K}}, \psi_{\hat{K}})$ *is defined by*

- $\hat{K}$ *is the reference tetrahedron,*
- $P_{\hat{K}} = R_k$ *is the associated function space,*
- *The degrees of freedom* $\psi_{\hat{K}} = M_{\hat{e}} \cup M_{\hat{f}} \cup M_{\hat{K}}$ *are of three types associated with edges* $\hat{e}$ *of* $\hat{K}$*, faces* $\hat{f}$ *of* $\hat{K}$ *and* $\hat{K}$ *itself. We denote by* $\hat{\boldsymbol{\tau}}$ *a unit vector in the direction of* $\hat{e}$*. We define three different degrees of freedom as follows:*

  (1) *the first set is associated with edges of the element:*

$$M_{\hat{e}}(\hat{\boldsymbol{u}}) = \left\{ \int_{\hat{e}} \hat{\boldsymbol{u}} \cdot \hat{\boldsymbol{\tau}} \hat{q} \, d\hat{s} \quad \text{for all } \hat{q} \in P_{k-1}(\hat{e}) \text{ for each edge } \hat{e} \text{ of } \hat{K} \right\}, \tag{3.18}$$

  (2) *the second set is associated with faces of the element:*

$$M_{\hat{f}}(\hat{\boldsymbol{u}}) = \left\{ \frac{1}{|\hat{f}|} \int_{\hat{f}} \hat{\boldsymbol{u}} \cdot \hat{\boldsymbol{q}} \, d\hat{A} \quad \text{for all faces } \hat{f}, \hat{\boldsymbol{q}} \in (P_{k-2}(\hat{f}))^3, \, \hat{\boldsymbol{q}} \cdot \hat{\boldsymbol{n}} = 0 \right\},$$

  (3) *the last set is associated with the volume:*

$$M_{\hat{K}}(\hat{\boldsymbol{u}}) = \left\{ \int_{\hat{K}} \hat{\boldsymbol{u}} \cdot \hat{\boldsymbol{q}} \, d\hat{V} \quad \text{for all } \hat{\boldsymbol{q}} \in (P_{k-3}(\hat{K}))^3 \right\}.$$

Following the theorem

Theorem 2 (*adapted from Monk (ibid.)*)

The finite element defined in Definition 1 is $\boldsymbol{H}(curl;\Omega)$ conforming and unisolvent.

the local interpolant $\boldsymbol{E}^h$ for the element K fulfills the following condition:

$$M_j(\boldsymbol{E}^h) = M_j(\boldsymbol{E}).$$

Applying this condition to equation (3.17) for the basis $\hat{\boldsymbol{\phi}}_i$ defined in the reference element we get

$$\begin{aligned} M_j(\boldsymbol{E}^h) &= M_j\left(\sum_{i=1}^{n} u_i \boldsymbol{\phi}_i\right) \\ &= \sum_{i=1}^{n} u_i M_j(\boldsymbol{\phi}_i) \stackrel{!}{=} M_j(\boldsymbol{E}). \end{aligned}$$

Thus, the coefficients u_i which are needed to evaluate the linear combination are linked to the moments M_j by the following linear system of equations:

$$M(\boldsymbol{\phi})\boldsymbol{u} = \boldsymbol{m}(\boldsymbol{E}) \tag{3.19}$$

with $M(\boldsymbol{\phi})$ being a matrix containing the moments of the basis functions and the vector $\boldsymbol{m}(\boldsymbol{E})$ containing the moments of the solution $\boldsymbol{E}$. For a nodal basis $\{\boldsymbol{\phi}_j\}_{j=1,\dots,n}$ of $\mathscr{V}^h$ and linear Nédélec elements, the moments are given as follows:

$$M_i(\boldsymbol{\phi}_j) = \delta_{i,j}, \qquad 1 \leq i,j \leq n, \tag{3.20}$$

which means that the moments M_i of the basis functions are equal to one on the associated edge and zero on all other edges. This is similar to the characteristic property of a nodal basis for Lagrange elements we described in Section 2.2.3.2 and especially in equation (2.25). In this case, the matrix $M(\boldsymbol{\phi})$ equals the identity matrix and thus, the coefficients $\boldsymbol{u}$ are equal to the moments $\boldsymbol{m}(\boldsymbol{E})$. With the linear combination (3.17) and a test function $\boldsymbol{v} = \boldsymbol{\phi}_i$, the variational formulation (3.16) is equivalent to

$$\sum_{j=1}^{n} u_j a(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i) = \boldsymbol{0}, \quad i = 1,2,\dots,n,$$

and with $A \in \mathbb{R}^{n\times n}$ for $[A]_{i,j} = a(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i)$, the solution $\boldsymbol{u} \in \mathbb{R}^n$ for $[u]_i = u_i$ and the conductivity vector $\boldsymbol{\sigma}$ which contains one conductivity value for each element K, we get the so-called Galerkin system (Zienkiewicz et al., 2005)

$$A(\boldsymbol{\sigma})\boldsymbol{u} = \boldsymbol{0}.$$

In contrast to Lagrange elements, the solution $\boldsymbol{u}$ of the Galerkin system is related to the moments of the approximated physical quantity and we have to evaluate the linear combination (3.17) to obtain the approximated electric field $\boldsymbol{E}^h$. Therefore, we assemble a measurement operator Q which is explained in detail in Section 3.2.3.

3.2.2.3 Assembly of the system matrix

To assemble the system matrix $A(\boldsymbol{\sigma})$ we have to reformulate the bilinear integrals of equation (3.15) in terms of the basis functions of the finite element space for each element K in the triangulation $\mathscr{T}^h$:

$$\begin{aligned} a(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i) &= \int_\Omega ((\nabla \times \boldsymbol{\phi}_j) \cdot (\nabla \times \boldsymbol{\phi}_i) + i\omega\mu_0\sigma(\boldsymbol{x})\boldsymbol{\phi}_j \cdot \boldsymbol{\phi}_i)\, d\boldsymbol{x} \\ &= \int_\Omega (\nabla \times \boldsymbol{\phi}_j) \cdot (\nabla \times \boldsymbol{\phi}_i)\, d\boldsymbol{x} + i\omega\mu_0 \int_\Omega \sigma(\boldsymbol{x})\boldsymbol{\phi}_j \cdot \boldsymbol{\phi}_i\, d\boldsymbol{x} \\ &= \underbrace{\sum_{K\in\mathscr{T}^h} \int_K (\nabla \times \boldsymbol{\phi}_j) \cdot (\nabla \times \boldsymbol{\phi}_i)\, d\boldsymbol{x}}_{\text{stiffness matrix}} + i\omega\mu_0 \underbrace{\sum_{K\in\mathscr{T}^h} \sigma_K \int_K \boldsymbol{\phi}_j \cdot \boldsymbol{\phi}_i\, d\boldsymbol{x}}_{\text{mass matrix}} \\ &= \sum_{K\in\mathscr{T}^h} a_K^{\text{stiffness}}(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i) + i\omega\mu_0 \sum_{K\in\mathscr{T}^h} a_K^{\text{mass}}(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i). \end{aligned} \tag{3.21}$$

The system matrix can be split in two parts: The stiffness matrix containing the curl-curl operator and the mass matrix depending on the parameter vector $\boldsymbol{\sigma}$. As for the DC resistivity problem, we consider a piecewise constant conductivity distribution $\sigma(\boldsymbol{x})$, where each element K is associated with a constant conductivity σ_K and the element integral for the mass matrix does not depend on sigma anymore. We assemble the element matrix for a single element:

$$[A_K]_{i,j} := a_K(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i) \quad i,j = 1,2,\ldots,n.$$

Then, the full system matrix can be obtained by forming

$$A = \sum_{K\in\mathscr{T}^h} A_K.$$

Again, we use tetrahedral grids and derive the basis functions and moments within the reference tetrahedron $\hat{K}$, which is the unit simplex in Figure 3.1 with the vertices

$$\hat{\boldsymbol{v}}_1 = [1,0,0]^T,\ \hat{\boldsymbol{v}}_2 = [0,1,0]^T,\ \hat{\boldsymbol{v}}_3 = [0,0,1]^T \quad \text{and} \quad \hat{\boldsymbol{v}}_4 = [0,0,0]^T. \tag{3.22}$$

Its oriented edges, denoted by $\hat{e}_j$ and $j = 1,\ldots,6$, are given by

$$\hat{e_1} : \hat{\boldsymbol{v}}_1 \to \hat{\boldsymbol{v}}_2,\ \hat{e_2} : \hat{\boldsymbol{v}}_1 \to \hat{\boldsymbol{v}}_3,\ \hat{e_3} : \hat{\boldsymbol{v}}_1 \to \hat{\boldsymbol{v}}_4,\ \hat{e_4} : \hat{\boldsymbol{v}}_2 \to \hat{\boldsymbol{v}}_3,\ \hat{e_5} : \hat{\boldsymbol{v}}_2 \to \hat{\boldsymbol{v}}_4,\ \hat{e_6} : \hat{\boldsymbol{v}}_3 \to \hat{\boldsymbol{v}}_4.$$

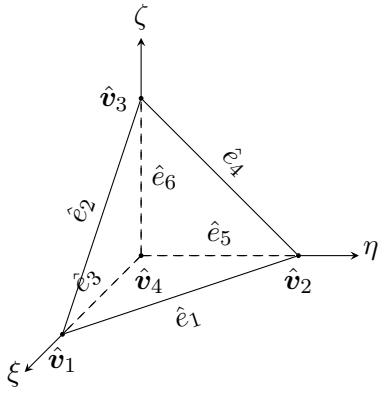

Fig. 3.1: The reference tetrahedron.

With each edge we associate the unit (tangent) vector $\hat{\boldsymbol{\tau}}_j$ in the edge direction:

$$\hat{\boldsymbol{\tau}}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix}-1\\1\\0\end{bmatrix}, \hat{\boldsymbol{\tau}}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix}-1\\0\\1\end{bmatrix}, \hat{\boldsymbol{\tau}}_3 = \begin{bmatrix}-1\\0\\0\end{bmatrix}, \hat{\boldsymbol{\tau}}_4 = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\-1\\1\end{bmatrix}, \hat{\boldsymbol{\tau}}_5 = \begin{bmatrix}0\\-1\\0\end{bmatrix}, \hat{\boldsymbol{\tau}}_6 = \begin{bmatrix}0\\0\\-1\end{bmatrix}.$$

Finally, we define the faces $\hat{f}_j$ with $j = 1, \ldots, 4$. These are spanned by the edges as follows:

$$\hat{f}_1 : \hat{e}_1, \hat{e}_2, (\hat{e}_4), \qquad \hat{f}_2 : \hat{e}_1, \hat{e}_3, (\hat{e}_5), \qquad \hat{f}_3 : \hat{e}_2, \hat{e}_3, (\hat{e}_6), \qquad \hat{f}_4 : \hat{e}_4, \hat{e}_5, (\hat{e}_6), \tag{3.23}$$

with their corresponding outward unit normals

$$\hat{\boldsymbol{n}}_1 = \frac{1}{\sqrt{3}}[1,1,1]^T, \quad \hat{\boldsymbol{n}}_2 = [0,0,-1]^T, \quad \hat{\boldsymbol{n}}_3 = [0,-1,0]^T \quad \text{and} \quad \hat{\boldsymbol{n}}_4 = [-1,0,0]^T. \tag{3.24}$$

With the reference tetrahedron $\hat{K}$ we can obtain any tetrahedron K in the triangulation by using an affine mapping (Monk, 2003). This transformation does not differ from that for Lagrange elements stated in equation (2.31): For the map $F_K : \hat{K} \to K$ with $F_K(\hat{K}) = K$ it holds

$$\boldsymbol{x} = F_K(\hat{\boldsymbol{x}}) = B_K\hat{\boldsymbol{x}} + \boldsymbol{b}_K, \quad \hat{K} \ni \hat{\boldsymbol{x}} \mapsto \boldsymbol{x} \in K, \tag{3.25}$$

where B_K and $\boldsymbol{b}_K$ are of the same structure as described for Lagrange elements in equation (2.32). To evaluate the bilinear forms for the mass and stiffness matrix given in equation (3.21), we have to find an appropriate basis of the Nédélec space R_k. Following Monk (ibid.) and Gopalakrishnan et al. (2005), the space of polynomials R_k is obtained by

$$R_k = (P_{k-1})^3 \oplus \mathcal{S}_k. \tag{3.26}$$

$\mathcal{S}_k$ is a subspace of vector polynomials:

$$\mathcal{S}_k = \left\{ \boldsymbol{p} \in (\tilde{P}_k)^3 \,|\, \boldsymbol{x} \cdot \boldsymbol{p} = 0 \right\},$$

with $\boldsymbol{x} = (x, y, z)$, $\boldsymbol{x} \cdot \boldsymbol{p} \in \tilde{P}_{k+1}$ and $(\tilde{P}_k)^3$ being the space of vector polynomials where each component of the vector belongs to $\tilde{P}_k$, the space of all homogeneous polynomials of degree k. The number of the DOFs for each edge element is defined by the dimension of R_k:

$$\dim(R_k) = 3\dim(P_{k-1}) + \dim \mathcal{S}_k = \tfrac{1}{2}(k+3)(k+2)k.$$

For the Nédélec elements of order $k = 1$ and $k = 2$ we will deal with the dimension of R_k and therefore, the number of DOFs and basis functions per element is $\dim R_1 = 6$ and $\dim R_2 = 20$. In the following, we derive the basis functions for R_1. Whitney (1957) describes a method to determine these functions by usage of barycentric coordinates λ, which are given by

$$\lambda_1(\hat{\boldsymbol{x}}) = \xi, \quad \lambda_2(\hat{\boldsymbol{x}}) = \eta, \quad \lambda_3(\hat{\boldsymbol{x}}) = \zeta, \quad \lambda_4(\hat{\boldsymbol{x}}) = 1 - \xi - \eta - \zeta$$

for an arbitrary point $\hat{\boldsymbol{x}} = (\xi, \eta, \zeta) \in \hat{K}$. Then, the basis functions can be deduced from the Whitney form

$$\boldsymbol{\theta}_{ij} := \lambda_i \nabla\lambda_j - \lambda_j \nabla\lambda_i, \qquad 1 \le i < j \le 6,$$

with the gradients of λ given by

$$\nabla\lambda_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \nabla\lambda_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \nabla\lambda_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \nabla\lambda_4 = \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix}.$$

Consequently, the basis is given by

$$\hat{\boldsymbol{\phi}}_1(\hat{\boldsymbol{x}}) = \begin{bmatrix} -\eta \\ \xi \\ 0 \end{bmatrix}, \qquad \hat{\boldsymbol{\phi}}_2(\hat{\boldsymbol{x}}) = \begin{bmatrix} -\zeta \\ 0 \\ \xi \end{bmatrix}, \qquad \hat{\boldsymbol{\phi}}_3(\hat{\boldsymbol{x}}) = \begin{bmatrix} \eta + \zeta - 1 \\ -\xi \\ -\xi \end{bmatrix},$$

$$\hat{\boldsymbol{\phi}}_4(\hat{\boldsymbol{x}}) = \begin{bmatrix} 0 \\ -\zeta \\ \eta \end{bmatrix}, \qquad \hat{\boldsymbol{\phi}}_5(\hat{\boldsymbol{x}}) = \begin{bmatrix} -\eta \\ \xi + \zeta - 1 \\ -\eta \end{bmatrix}, \qquad \hat{\boldsymbol{\phi}}_6(\hat{\boldsymbol{x}}) = \begin{bmatrix} -\zeta \\ -\zeta \\ \xi + \eta - 1 \end{bmatrix}. \tag{3.27}$$

Just like for linear elements, we use Whitney forms to determine a basis of R_2 on the reference element and use barycentric coordinates to define 20 basis functions. The detailed derivation and explicit functions can be looked up in Appendix A.2.

The preceding explanations and equations define moments and basis functions on the reference tetrahedron. To calculate these quantities on an arbitrary element we need to use the affine mapping defined in equation (3.25). For the transformation of Nédélec basis functions,

which are vector functions, we have to multiply by a matrix to conserve their properties:

$$\boldsymbol{\phi}(\boldsymbol{x}) := (dF_K)^{-T}\hat{\boldsymbol{\phi}}(F_K^{-1}(\boldsymbol{x})) = B_K^{-T}\hat{\boldsymbol{\phi}}(\hat{\boldsymbol{x}}). \tag{3.28}$$

Furthermore, we get the gradient of a scalar function ϕ in an arbitrary tetrahedron by

$$(\nabla\phi) \circ F_K = B_K^{-T}\hat{\nabla}\hat{\phi}$$

and the curl of a vector-valued function $\hat{\boldsymbol{\phi}}$ is obtained by

$$\nabla \times \boldsymbol{\phi} = \frac{1}{\det B_K} B_K(\hat{\nabla} \times \hat{\boldsymbol{\phi}}). \tag{3.29}$$

Finally, the unit tangent vectors $\hat{\boldsymbol{\tau}}$ along an edge $\hat{e}$ of $\hat{K}$ can be transformed as follows:

$$\boldsymbol{\tau} := \frac{B_K\hat{\boldsymbol{\tau}}}{\|B_K\hat{\boldsymbol{\tau}}\|}. \tag{3.30}$$

Now, to assemble the system matrix $A(\boldsymbol{\sigma})$, we can reformulate the integrals for the stiffness and mass matrix given in equation (3.21) in terms of the reference element using the affine covariant Piola transformation (equation (3.28)). The mass matrix M will be assembled as follows:

$$\begin{aligned} M(\boldsymbol{\sigma}) = a_K^{\text{mass}}(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i)_K &= \sigma_K \int_K \boldsymbol{\phi}_j \cdot \boldsymbol{\phi}_i \, d\boldsymbol{x} \\ &= \sigma_K \int_{\hat{K}} (B_K^{-T}\hat{\boldsymbol{\phi}}_j) \cdot (B_K^{-T}\hat{\boldsymbol{\phi}}_i) |\det B_K| \, d\hat{\boldsymbol{x}}. \end{aligned}$$

And, with the transform of the curl given in equation (3.29), we can assemble the stiffness or curl-curl matrix K, which is independent of the conductivity σ_K:

$$\begin{aligned} C = a_K^{\text{stiffness}}(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i) &= \int_K (\nabla \times \boldsymbol{\phi}_j) \cdot (\nabla \times \boldsymbol{\phi}_i) \, d\boldsymbol{x} \\ &= \frac{1}{|\det B_K|} \int_{\hat{K}} (B_K \nabla \times \hat{\boldsymbol{\phi}}_j) \cdot (B_K \nabla \times \hat{\boldsymbol{\phi}}_i) \, d\hat{\boldsymbol{x}}. \end{aligned}$$

The assembly of the mass and stiffness matrix yields all parts for the discrete representation of our geophysical model problem stated in equation (3.7) which is the associated Galerkin system:

$$(C + i\omega\mu_0 M(\boldsymbol{\sigma}))\boldsymbol{u} = \boldsymbol{0}. \tag{3.31}$$

As for DC resistivity forward modeling we do not assemble the mass matrix $M(\boldsymbol{\sigma})$ directly, but we calculate the derivative of M with respect to sigma as a three-way tensor (see Section 2.2.3.5). We can obtain the full mass matrix by multiplying this tensor by the full parameter vector:

$$M(\boldsymbol{\sigma}) = \frac{\partial M(\boldsymbol{\sigma})}{\partial \boldsymbol{\sigma}} \times_3 \boldsymbol{\sigma}.$$

3.2.2.4 Implementing boundary conditions

We refrain from describing the implementation of the boundary conditions in detail because of the similarities to the DC resistivity method and the variety of appropriate literature. As stated in the previous section, we implement a Dirichlet boundary condition and refer to Section 2.2.3.4 and especially equation (2.34):

$$A_{II}\boldsymbol{u}_I = \boldsymbol{b}_I - A_{ID}\boldsymbol{g}_D. \tag{3.32}$$

For Nédélec elements, the DOFs in $\boldsymbol{u}_I$ are not equal to electric field values but linked to the moments of the electric field at the boundary. Therefore, to implement the Dirichlet conditions correctly, we have to evaluate the Nédélec interpolant at the boundary and calculate the respective moments. We defined the curl-conforming Nédélec element and the edge, face and volume moments in Definition 1. For example the edge moments M_e can be calculated after defining a basis of the polynomial space $P_0(\hat{e})$. We choose the simplest constant polynomial given by $\hat{q} \equiv 1$ in the linear case. Then, with the transformation given in equation (3.30) we can determine the DOFs on the edges e_j for an arbitrary boundary element by evaluating the integral (3.18):

$$\begin{aligned} M_{e_j}(\boldsymbol{E}(\boldsymbol{x})) &= \int_{e_j} \boldsymbol{E}(\boldsymbol{x}) \cdot \boldsymbol{\tau}_j \, ds, \quad j = 1, \ldots, n_{\text{boundary edges}} \\ &= \int_{e_j} \boldsymbol{E}(\boldsymbol{x}) \cdot \frac{B_K \hat{\boldsymbol{\tau}}_j}{\|B_K \hat{\boldsymbol{\tau}}_j\|} \, ds. \end{aligned}$$

For each boundary edge e_j we calculate one moment M_{e_j} and finally solve the linear system (3.19) to determine the Dirichlet coefficients $\boldsymbol{u}_D$ for all boundary edges:

$$\boldsymbol{u}_D = M(\boldsymbol{\phi})^{-1}\boldsymbol{m}(\boldsymbol{E}).$$

The matrix $M(\boldsymbol{\phi})$ is equal to the identity matrix for linear Nédélec elements (equation (3.18)) and there holds

$$\boldsymbol{u}_D^{\text{linear}} = \boldsymbol{m}(\boldsymbol{E}).$$

We insert the boundary coefficients $\boldsymbol{u}_D$ into the vector $\boldsymbol{g}_D$ to finally evaluate equation (3.32). After implementing these conditions, the solution of the electromagnetic Galerkin system given in equation (3.31) yields the vector $\boldsymbol{u}$ which contains the coefficients of the linear combination (3.17) which are used to obtain the field values in the following section.

3.2.3 The measurement operator

3.2.3.1 Implementation for the electric field

In contrast to the DC resistivity method, where the solution $\boldsymbol{u}$ of the Galerkin system contains the explicit values of the investigated physical field, for the electromagnetic problem and for every problem discretized with Nédélec elements, the entries of $\boldsymbol{u}$ are not field values but related to the moments on the edges of the elements. Therefore, to get the electric field $\boldsymbol{E}$ at certain measurement points in a data vector $\boldsymbol{b}_E$, we have to calculate the linear combination of these moments with the associated basis functions. This is done with the measurement operator Q_E. The basic idea is the same as for Lagrange elements (see Section 2.2.4): We implement the linear combination as a matrix-vector product:

$$\boldsymbol{b}_E = \sum_{i}^{n_{\text{DOFs}}} u_i \boldsymbol{\phi}_i(\boldsymbol{x}) = Q_E \boldsymbol{u} = Q_E \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n_{\text{DOFs}}} \end{bmatrix}, \; j = 1, \ldots, n_{\text{receivers}}, \tag{3.33}$$

where each row of Q_E represents one location and a single frequency and contains the evaluated basis functions of the tetrahedron the receiver belongs to (Pseudocode 7). For linear elements, we have at least six basis functions and for quadratic elements there are 20. In comparison to Pseudocode 2 for Lagrange elements, the code changes for Nédélec elements because we have to use the covariant Piola transformation described in equation (3.28) to conserve the properties of the basis functions:

Pseudocode 7 CALCULATE J$^{\text{TH}}$ ROW OF Q_E FOR J$^{\text{TH}}$ RECEIVER

1: Calculate index i of enclosing element K for j$^{\text{th}}$ receiver
2: Extract corresponding DOFs $i_1, \ldots, i_6$
3: Solve $B_K \hat{\boldsymbol{x}} = (\text{receivers}(j, :) - \boldsymbol{b}_K)$
4: **for** $i = 1, \ldots, 6$ **do**
5: Solve $B_K^T \boldsymbol{\phi}_i(\boldsymbol{x}) = \hat{\boldsymbol{\phi}}_i(\hat{\boldsymbol{x}})$
6: $\boldsymbol{q}_K(i) \leftarrow \boldsymbol{\phi}_i(\boldsymbol{x})$
7: **end for**
8: $Q_E(j, [i_1\ i_2\ i_3\ i_4\ i_5\ i_6]) \leftarrow \boldsymbol{q}_K^T$

The code is given for linear basis functions. In the quadratic case, the upper bound of the for-loop has to be 20.

3.2.3.2 Implementation for the magnetic field

The measured physical quantities for a lot of electromagnetic methods are not only electric field components, but components of the magnetic field intensity, too. Therefore, we have to implement a measurement operator Q_H for the magnetic field. We rewrite Faraday's law given in equation (3.5):

$$\boldsymbol{H} = -\frac{1}{i\omega\mu}\nabla \times \boldsymbol{E}.$$

Hence, we have to assemble the curl of the electric field and multiply it by a factor to get the magnetic field $\boldsymbol{H}$. The approach is similar to the implementation of the measurement operator Q_E: The linear combination of the curl of different basis functions is implemented as a matrix-vector product:

$$\boldsymbol{b}_H = -\frac{1}{i\omega\mu}\sum_{i}^{n_{\text{DOFs}}} u_i \nabla \times \boldsymbol{\phi}_i(\boldsymbol{x}_j) = Q_H \boldsymbol{u} = Q_H \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n_{\text{DOFs}}} \end{bmatrix}, \; j = 1, \ldots, n_{\text{receivers}}. \tag{3.34}$$

Again, the ith row of Q_H contains the curls of all basis functions associated with the element the ith receiver belongs to.

Pseudocode 8 CALCULATE J$^{\text{TH}}$ ROW OF Q_H FOR J$^{\text{TH}}$ RECEIVER

1: Calculate index i of enclosing element K for j$^{\text{th}}$ receiver
2: Extract corresponding DOFs $i_1, \ldots, i_6$
3: Solve $B_K \hat{\boldsymbol{x}} = (\text{receivers}(j, :) - \boldsymbol{b}_K)$
4: $d \leftarrow \det(B_K)$
5: **for** $i = 1, \ldots, 6$ **do**
6: $\quad \nabla \times \boldsymbol{\phi}_i(\boldsymbol{x}) \leftarrow \frac{1}{d} B_K \nabla \times \hat{\boldsymbol{\phi}}_i(\hat{\boldsymbol{x}})$
7: $\quad \boldsymbol{q}_K(i) \leftarrow \nabla \times \boldsymbol{\phi}_i(\boldsymbol{x})$
8: **end for**
9: $Q_H(j, [i_1\ i_2\ i_3\ i_4\ i_5\ i_6]) \leftarrow -\frac{1}{i\omega\mu}\boldsymbol{q}_K^T$

The implementation for linear elements follows Pseudocode 8. We apply the Piola transformation given in equation (3.29) to obtain the curl of the basis functions in an arbitrary element (line 6). A further method to assemble this measurement operator is the integration of the electric field $\boldsymbol{E}$ over a small coil as done in the function `get_Q_H`, but we will only use the curl of $\boldsymbol{E}$ in the following sections.

3.2.4 Incorporation of multiple frequencies

Almost all electromagnetic methods use more than only one frequency to investigate the subsurface. As described in Section 3.1, the behavior and especially the resolution depth of an electromagnetic method depends on the frequency that is used during the measurement. If we include a set of frequencies, the information will be much more detailed for a larger depth range and this hopefully improves the inversion result.

Multiple frequencies in electromagnetics correspond to multiple source positions for the DC resistivity method. We introduced modeling and inversion for multiple sources in Section 2.2.5 and use block diagonal matrices containing n_s equivalent blocks of the system matrix A and the measurement operator Q on their main diagonal for the inversion of data generated by n_s sources. For the electromagnetic code we have to keep in mind that the system matrix (see Section 3.2.2)

$$A(\boldsymbol{\sigma}) = C + i\omega\mu_0 M(\boldsymbol{\sigma}), \quad \omega = 2\pi f$$

as well as the measurement operator for the magnetic field Q_H (see Section 3.2.3) derived from Faraday's law (3.5) depends on the frequency f. Hence, we can assemble block diagonal matrices, too, but the individual blocks are different for each frequency f_k ($k = 1 \dots n_f$):

$$A_{\mathsf{blk}} = \begin{bmatrix} A^{f_1}(\boldsymbol{\sigma}) & & & \\ & A^{f_2}(\boldsymbol{\sigma}) & & \\ & & \ddots & \\ & & & A^{f_k}(\boldsymbol{\sigma}) \end{bmatrix}, \quad Q_H^{\mathsf{blk}} = \begin{bmatrix} Q_H^{f_1} & & & \\ & Q_H^{f_2} & & \\ & & \ddots & \\ & & & Q_H^{f_k} \end{bmatrix}.$$

The measurement operator for the electric field is independent of f and has to be assembled only once:

$$Q_E^{\mathsf{blk}} = \begin{bmatrix} Q_E & & & \\ & Q_E & & \\ & & \ddots & \\ & & & Q_E \end{bmatrix}.$$

The solution $\boldsymbol{u}$ of the Galerkin system is extended for multiple frequencies by solving this system for each frequency individually and writing the resulting vectors $\boldsymbol{u}_{f_k}$ one below the other:

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_{f_1} \\ \boldsymbol{u}_{f_2} \\ \vdots \\ \boldsymbol{u}_{f_k} \end{bmatrix}.$$

Later on, the block matrices are used as input for the inversion algorithm (see Section 3.3.1) to expand the sensitivity matrix for multiple frequencies in an appropriate way.

Now we are able to solve the electromagnetic forward problem for one polarization and an arbitrary conductivity distribution. The observation matrices Q_E and Q_H enable us to extract synthetic data sets at arbitrary receiver locations for an arbitrary number of frequencies. The following section describes the application of the forward modeling code to different model problems.

3.2.5 Reference solutions

To verify the electromagnetic forward operator, it is important to calculate the numerical solution for sufficiently simple models, for which we know the analytical solution so we can compare the data sets.

The forward modeling code is able to use two different solvers for the Galerkin system which are MATLAB's built-in sparse solver `mldivide` and Intel MKL PARDISO. The efficiency of the PARDISO solver will be explained in Section 3.2.5.3.

3.2.5.1 The cavity problem

First of all, we examine the cavity problem for a plane wave in a cavity (Monk, 2003). The vector field

$$\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{a}\exp(i\omega\boldsymbol{k}\cdot\boldsymbol{x}) \tag{3.35}$$

describes a plane wave with the angular frequency $\omega = 2\pi f$, the frequency f and the wave vector $\boldsymbol{k} \in \mathbb{R}^3$ which spreads in the bounded domain Ω with the boundary $\partial\Omega$. The constant vector $\boldsymbol{a} \in \mathbb{R}^3$ contains the amplitude of the wave in all directions. The vector field $\boldsymbol{u}$ is the solution of a PDE, which we derive as follows: Applying the curl operator to the vector field $\boldsymbol{u}$, we arrive at

$$\nabla \times \boldsymbol{u} = i\omega(\boldsymbol{k}\times\boldsymbol{a})\exp(i\omega\boldsymbol{k}\cdot\boldsymbol{x}).$$

Using the identity $\boldsymbol{p}\times(\boldsymbol{q}\times\boldsymbol{r}) = (\boldsymbol{p}\cdot\boldsymbol{r})\boldsymbol{q} - (\boldsymbol{p}\cdot\boldsymbol{q})\boldsymbol{r}$ leads to

$$\begin{aligned}\nabla\times\nabla\times\boldsymbol{u} &= (i\omega)^2(\boldsymbol{k}\times(\boldsymbol{k}\times\boldsymbol{a}))\exp(i\omega\boldsymbol{k}\cdot\boldsymbol{x})\\ &= -\omega^2((\boldsymbol{k}\cdot\boldsymbol{a})\boldsymbol{k} - (\boldsymbol{k}\cdot\boldsymbol{k})\boldsymbol{a})\exp(i\omega\boldsymbol{k}\cdot\boldsymbol{x}).\end{aligned}$$

Now, we choose the vectors $\boldsymbol{a}$ and $\boldsymbol{k}$ to be perpendicular to each other ($\boldsymbol{a} \perp \boldsymbol{k}$) and $|\boldsymbol{k}| = 1$ to simplify this equation:

$$\nabla \times \nabla \times \boldsymbol{u} = \omega^2 \underbrace{\boldsymbol{a} \exp(i\omega \boldsymbol{k} \cdot \boldsymbol{x})}_{\boldsymbol{u}}.$$

Under these assumptions, $\boldsymbol{u}$ satisfies the homogeneous, linear PDE

$$\nabla \times \nabla \times \boldsymbol{u} - \omega^2 \boldsymbol{u} = \boldsymbol{0}.$$

This PDE is similar to the electromagnetic curl-curl equation (3.7) as well as to its discrete representation given in equation (3.31) except for the prefactor of the mass matrix M which is $-\omega^2$ for the cavity problem and $i\omega\mu_0$ for the MT method. Its discrete representation then reads

$$(C - \omega^2 M(\boldsymbol{\sigma}))\boldsymbol{u} = \boldsymbol{0}$$

with a constant conductivity vector $\boldsymbol{\sigma} \equiv \boldsymbol{1}$. To verify the implementation of this problem with Nédélec elements, we have to add appropriate boundary conditions along $\partial\Omega$, similar to the electromagnetic boundary value problem given in equations (3.7) and (3.9). Here, we use Dirichlet conditions and evaluate the analytical solution (3.35) at the boundary edges. We assemble the mass and stiffness matrix as described in Section 3.2.2 and solve this equation in the bounded unit cube. The numerical solution is compared with the analytical expression for $\boldsymbol{u}$ given in equation (3.35). To ensure our assumptions are true, we choose the normalized amplitude and wave vector to be

$$\boldsymbol{k} = \frac{\hat{\boldsymbol{k}}}{||\hat{\boldsymbol{k}}||} = \frac{1}{||\hat{\boldsymbol{k}}||} \begin{bmatrix} -1.5 \\ 1 \\ 0 \end{bmatrix} \quad \text{and} \quad \boldsymbol{a} = \frac{\hat{\boldsymbol{a}}}{||\hat{\boldsymbol{a}}||} = \frac{1}{||\hat{\boldsymbol{a}}||} \begin{bmatrix} 1 \\ 1.5 \\ 0.5 \end{bmatrix}.$$

The angular frequency ω is set to one. For illustration purposes, we calculate and show the field $\boldsymbol{u}$ for a very simple and coarse mesh consisting of only 24 tetrahedra and 49 DOFs for linear elements and 218 DOFs for quadratic elements. Figure 3.2 gives a first impression of the quality of the modeling code. It shows the real part of the field vectors $\boldsymbol{u}(\boldsymbol{x})$ at the midpoints of all tetrahedra belonging to the mesh. We see the analytical solution given in equation (3.35) on the left-hand side and the numerical solution for quadratic elements on the right-hand side. For these full vectors, it is difficult to see differences or inaccuracies of the numerical approximation. Therefore, we show further details for the modeled individual field components for linear and quadratic elements in Figure 3.3 a) and b). There is almost no difference between the analytical solution (red) and the modeled data for quadratic elements (blue). Even for this coarse mesh with only 24 tetrahedra, the approximation is of very good

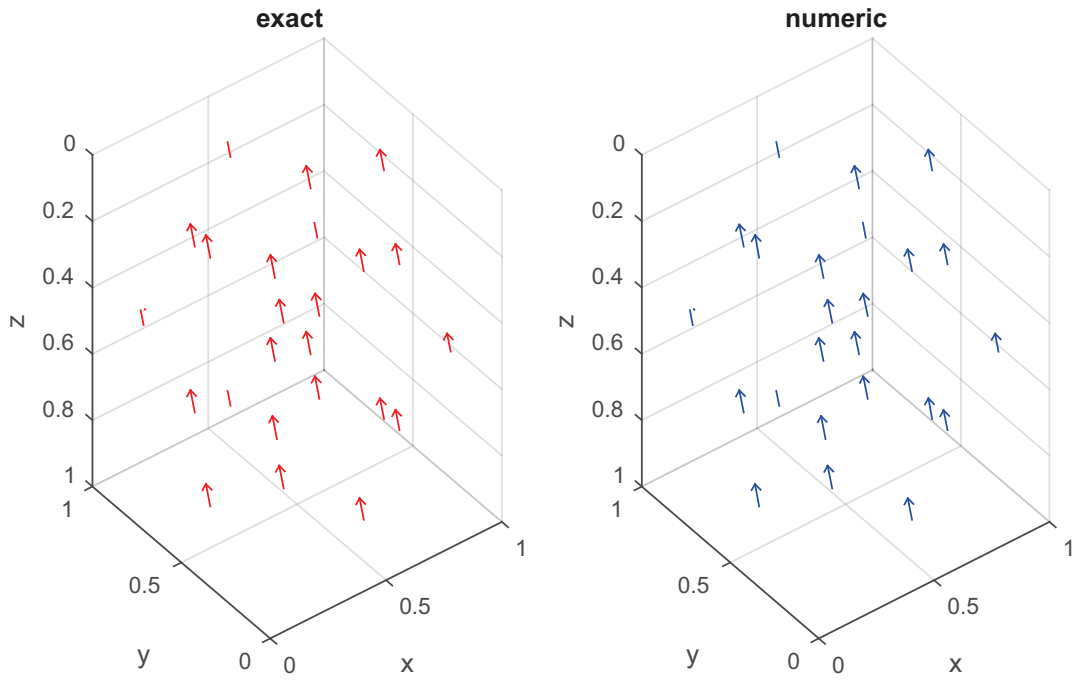

Fig. 3.2: Field vectors $\mathrm{Re}(\boldsymbol{u}(\boldsymbol{x}))$ at the midpoints of all tetrahedra in the unit cube for quadratic Nédélec elements.

quality. The linear approximation shows some inaccuracies which will become smaller for refined meshes. To quantify the behavior of the error when decreasing the element size, we can provide error estimates for the numerical solution $\boldsymbol{u}_h$ depending on the element diameter h. Referring to Monk (2003), we know that the L^2 error $e(h) = ||\boldsymbol{u} - \boldsymbol{u}_h||_{L^2}$ is proportional to h. For a constant c, the ansatz

$$e(h) = ch^\alpha$$
$$e\left(\frac{h}{2}\right) = c\left(\frac{h}{2}\right)^\alpha$$

leads to the convergence rate estimate α in terms of two error measurements $e(h)$ and $e(h/2)$:

$$\alpha = \log_2\left(\frac{e(h)}{e(h/2)}\right).$$

This formula yields the convergence rates for linear and quadratic elements for four different meshes given in Table 3.1. Nédélec elements of orders one and two converge with the expected convergence rate of one in the linear case and two in the quadratic case. The table shows that the difference of the L^2 error between order one and two is more than one decimal power. Consequently, the use of finite elements of higher order should be obligatory, whenever it is computationally affordable and the solution is sufficiently regular.

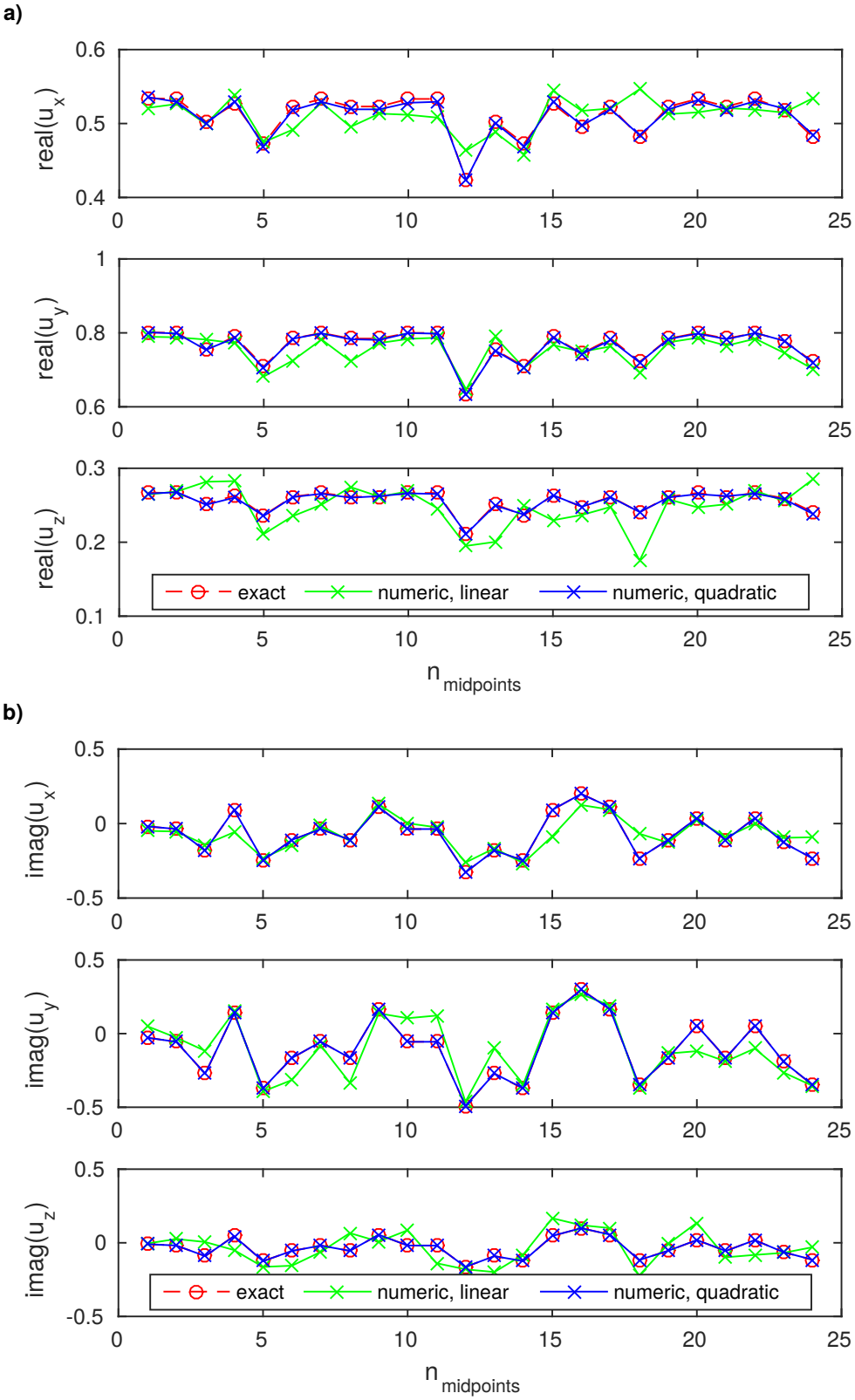

Fig. 3.3: Real (a) and imaginary (b) part of the components of $\boldsymbol{u}(\boldsymbol{x})$ at the midpoints of all tetrahedra in the unit cube for analytical and modeled data.

elements	DOFs	L^2 error, linear	α_{linear}	DOFs	L^2 error, quadratic	$\alpha_{\text{quad.}}$
24	49	7.07e-2	-	218	3.62e-3	-
192	302	3.56e-2	0.99	1 468	0.91e-3	1.99
1 536	2 092	1.77e-2	1.01	10 712	0.23e-3	2.01
12 288	15 512	0.87e-2	1.02	81 712	0.06e-3	2.02

Tab. 3.1: Convergence rates for elements of linear and quadratic order and different meshes.

3.2.5.2 MT on the unit cube

The consistency of the modeled forward solution with the analytical solution for the field $\boldsymbol{u}(\boldsymbol{x})$ as well as the convergence rates for different meshes ensure that the implemented forward modeling code works correctly. However, the cavity problem described in the previous section is just a theoretical model problem in a homogeneous unit cube. In order to exclude errors for inhomogeneous modeling domains and to establish the connection to electromagnetics, we implemented the MT problem for one polarization introduced in Section 3.2.1. The modeling domain consists of two unit cubes – one for the homogeneous Earth of constant conductivity and the other for the air halfspace. The numerical solution is compared with the analytic solution for a homogeneous halfspace given in equation (3.9). To determine an appropriate

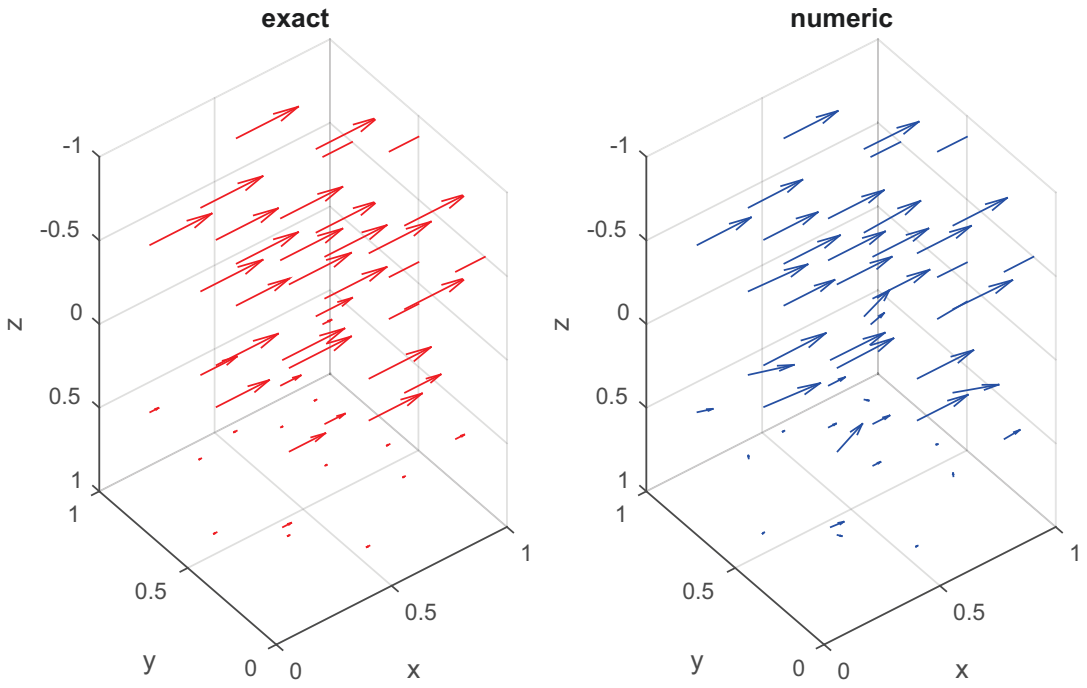

Fig. 3.4: Field vectors $\mathrm{Re}(\boldsymbol{E}(\boldsymbol{x}))$ at the midpoints of all tetrahedra in the unit cube for linear Nédélec elements.

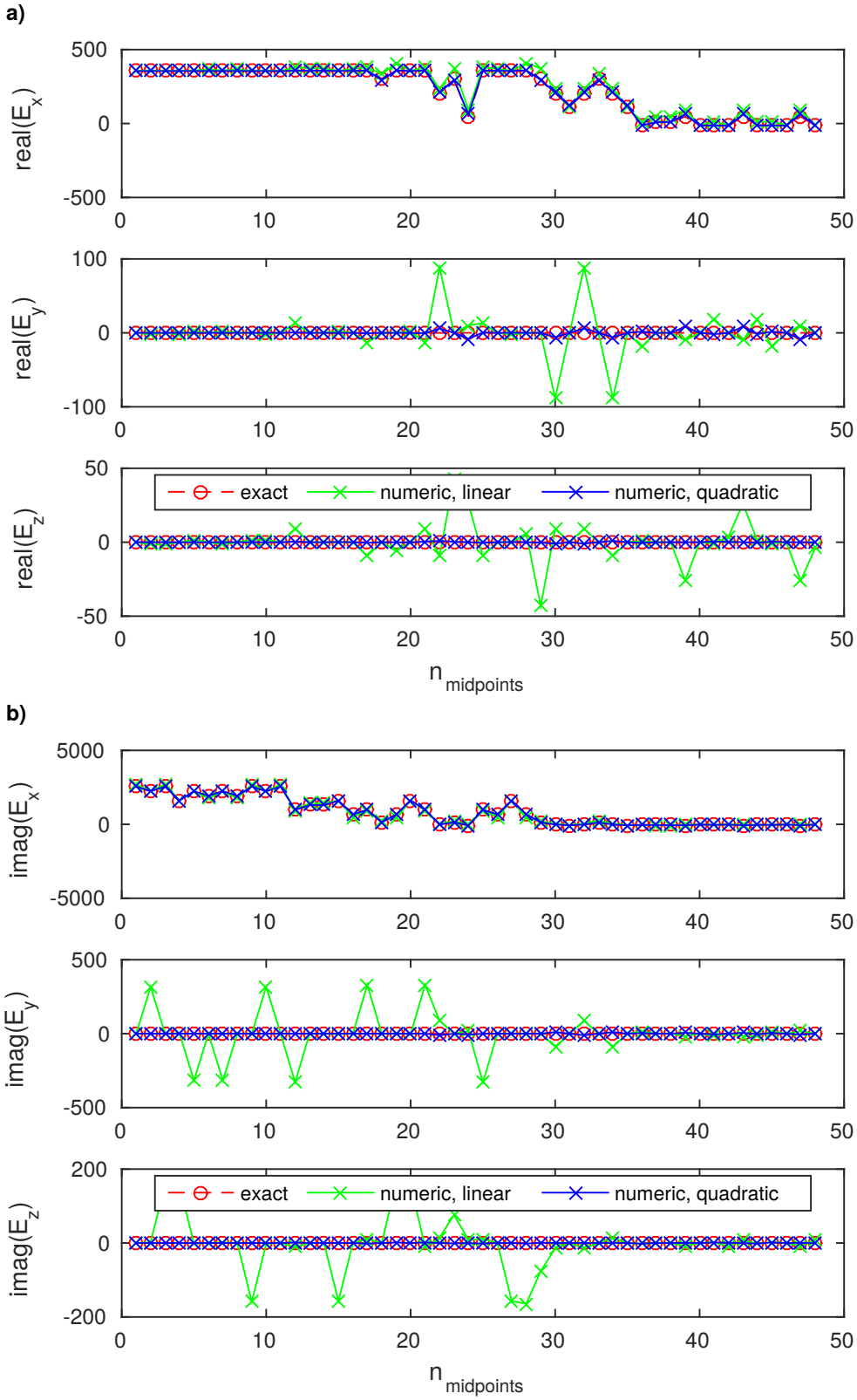

Fig. 3.5: Real (a) and imaginary (b) part of the components of $\boldsymbol{E}(\boldsymbol{x})$ at the midpoints of all tetrahedra in the unit cube for analytical and modeled data.

frequency $f = \frac{\omega}{2\pi}$ for this simple MT problem, we have to consider the skin effect. In Section 3.1 we derived a formula for the skin depth. Simulating a homogeneous halfspace with a constant conductivity of $\sigma = 0.01\ \frac{\mathrm{S}}{\mathrm{m}}$ and a size of $z_{\mathrm{max}} = 1$ m in z-direction we arrive at a frequency of $f = 2{\cdot}10^9$ Hz. The air conductivity is set to $10^{-9}\ \frac{\mathrm{S}}{\mathrm{m}}$ and we use Dirichlet boundary conditions according to the analytical solution (3.9). The mesh consists of 48 elements, 90 DOFs for linear elements and 412 DOFs in the quadratic case. We show the forward solution for linear elements, which are the field vectors for the real part of the electromagnetic field, in Figure 3.4. Again, we achieve a good approximation and see only few differences between the exact and the numerical solution. The field decays within the conductive cube and stays almost homogeneous in the resistive air cube. Because we use only one polarization, the electric field has only one component depending on the depth z as shown in Section 3.2.1. To compare the quality for elements of different order, we show the individual field components in Figures 3.5a) and b). For this coarse mesh, we observe a good approximation for elements of second order and some differences in the linear case which will become smaller with further refinement of the mesh because of the first and second order convergence of the L^2 error as shown in Table 3.1.

3.2.5.3 MT for the 3D-2 COMMEMI model

Finally, we apply our electromagnetic forward code to a more complicated model. Zhdanov et al. (1997) published electromagnetic modeling results for a number of models. These results can be used to test our code against it. The modeling is carried out for the 3D-2 COMMEMI model (Comparison of Modeling Methods for ElectroMagnetic Induction problems) given in Figure 3.6 and a frequency of 0.01 Hz. It mainly consists of three layers and two blocks in the first layer where one is conductive and the other is resistive compared to the background conductivity of the first layer. Within the paper, they developed the governing electromagnetic equations for a time dependence $e^{-i\omega t}$ which differs from our ansatz in Section 3.2.2 in the sign of the exponent. Therefore, we have to change the sign in the assembled Galerkin system (equation (3.31)) as well:

$$(C - i\omega\mu_0 M(\boldsymbol{\sigma}))\boldsymbol{u} = \mathbf{0}.$$

The forward problem has been assembled and solved for linear and quadratic elements and a set of uniformly refined meshes (Table 3.2). Our discretized model extends from –100 km to 100 km in both horizontal directions and down to a depth of 100 km in z-direction. The air-layer is set to be of 100 km thickness, too. We impose Dirichlet conditions for a layered halfspace and calculate the analytical boundary values with the function `getE1dMT` written by

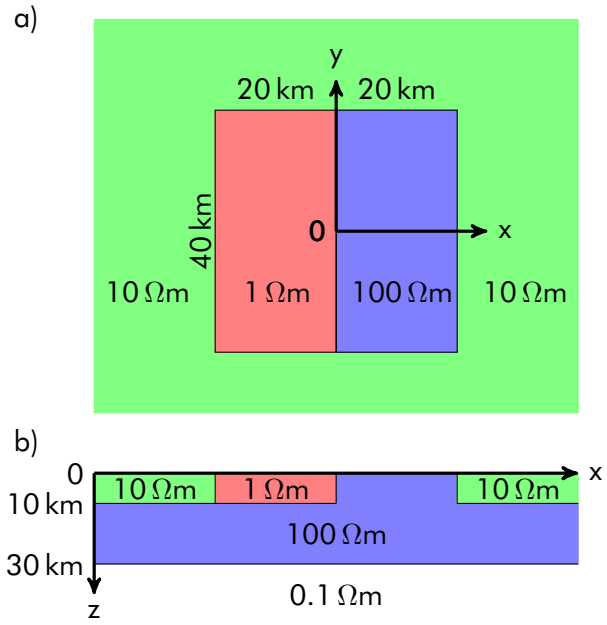

Fig. 3.6: Plan view (a) and slice (b) of the 3D-2 COMMEMI model.

Ralph-Uwe Börner according to Wait's algorithm (Wait, 1953). The assembly of the stiffness and mass matrix C and M is done only once for each model problem. Even for large meshes with about 1.5 million DOFs the required assembly time is only about four minutes. Referring to Section 3.2.5, besides assembly times, we compare two different solvers for our forward problem in the table. Compared to MATLAB's `mldivide`, the PARDISO solver is faster by a factor of six to ten for reasonably sized meshes and should be preferred.

Order	1	1	1	2	2	2
No. of elements	3 687	29 496	235 968	3 687	29 496	235 968
No. of DOFs	4 657	35 804	280 804	24 516	191 408	1 512 744
Assembly time (s)	0.98	7.19	60.29	3.90	32.26	268.45
Solution time (s), `mldivide`	0.20	12.00	1092.19	6.89	469.02	-
PARDISO	0.25	2.29	88.88	1.21	49.55	2721.44

Tab. 3.2: Timings for elements of 1st and 2nd order and different meshes done on an SMP machine with four AMD Opteron 6136 CPUs ("Magny-Cours", 2.4 GHz) and 256 GB RAM.

Figure 3.7 shows the normalized real and imaginary part of the x-component of the electric field $\boldsymbol{E}$ and the normalized real and imaginary part of the y-component of the magnetic field $\boldsymbol{H}$, which are the interesting components for modeling one polarization in the MT method. All results are calculated using quadratic Nédélec elements and about 1.5 million DOFs as shown in the last column of Table 3.2. The modeling results by Dr. Z. Xiong (blue circles)

were computed using an integral equation code (Xiong, 1992) and a model consisting of 16 800 prismatic cells and 18 receiver locations between $x = -70$ km and $x = 70$ km and $y = z = 0$ km. Our results fit very well with Xiong's data. Even at the interfaces of the first layer and one of the embedded blocks the step sizes in the electric field component almost coincide with each other.

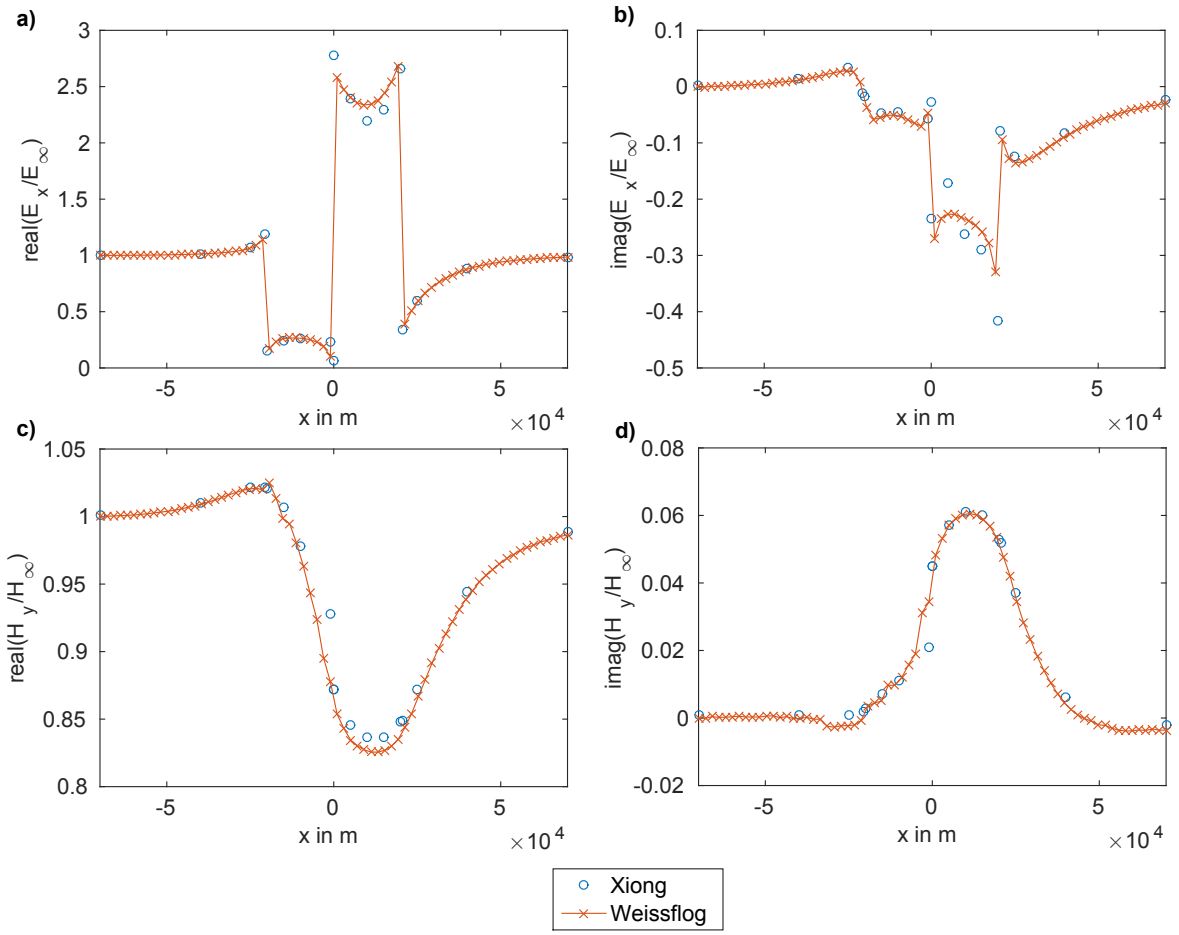

Fig. 3.7: Simulated real and imaginary part of the electric field component E_x (Figures a and b) as well as both parts of the magnetic field component H_y (Figures c and d) in comparison with Xiong's results published in Zhdanov et al. (1997).

The previous modeling results, the comparison with reference solutions and the convergence study for the cavity problem in Section 3.2.5.1 examined and confirmed the reliability and correctness of our forward modeling code. Especially the convergence study showed that the usage of quadratic Nédélec elements leads to a notably improved result. Now, this code can be used as part of an inversion algorithm.

3.3 Inversion of electromagnetic data

3.3.1 The Gauss-Newton method for complex data sets

Basically, the inversion algorithm is the same as described in Section 2.3.1 for the DC resistivity method. We refer to this section and confine ourselves to explaining the main differences. Again, we switch over to the logarithmic parameter vector $\boldsymbol{m} = \log \boldsymbol{\sigma}$ for the inversion algorithm as described in equation (2.36). For an electromagnetic problem, we have to keep in mind that we now deal with complex field quantities and system matrices and therefore the normal equations change considerably. We refer to the least squares problem in equation (2.39):

$$\begin{aligned}\Phi(\boldsymbol{m}) &= \tfrac{1}{2}\|[\underbrace{\boldsymbol{g}(\boldsymbol{m}_0) - \boldsymbol{b}}_{\text{data residual}}] + \underbrace{\boldsymbol{g}'(\boldsymbol{m}_0)}_{\text{Jacobian matrix}} \Delta\boldsymbol{m}\|_2^2 + \tfrac{\beta}{2}||W(\Delta\boldsymbol{m} + \boldsymbol{m}_0 - \boldsymbol{m}_{\text{ref}})||^2 \to \min_{\boldsymbol{m}} \\ &= \tfrac{1}{2}\|\Delta\boldsymbol{d} + J\Delta\boldsymbol{m}\|_2^2 + \tfrac{\beta}{2}||W(\Delta\boldsymbol{m} + \boldsymbol{m}_0 - \boldsymbol{m}_{\text{ref}})||^2 \to \min_{\boldsymbol{m}}\end{aligned}$$

with the Jacobian or sensitivity matrix J. For the derivation of the normal equations for complex quantities we leave out the second norm, which is the regularization part, because the matrix W and all parameter vectors $\boldsymbol{m}_0$, $\boldsymbol{m}_{\text{ref}}$ and $\Delta\boldsymbol{m}$ are real-valued and there is no difference in this part compared to Section 2.3.1. We call this shortened objective function $\Phi^*(\boldsymbol{m}_0)$. For the data residual norm there holds

$$\begin{aligned}\Phi^*(\boldsymbol{m}) &= \tfrac{1}{2}\|\Delta\boldsymbol{d} + J\Delta\boldsymbol{m}\|_2^2 = \tfrac{1}{2}\left(\Delta\boldsymbol{d} + J\Delta\boldsymbol{m}\right)^H\left(\Delta\boldsymbol{d} + J\Delta\boldsymbol{m}\right) \\ &= \tfrac{1}{2}\left(\Delta\boldsymbol{d}^H\Delta\boldsymbol{d} + \Delta\boldsymbol{d}^H J\Delta\boldsymbol{m} + \Delta\boldsymbol{m}^H J^H\Delta\boldsymbol{d} + \Delta\boldsymbol{m}^H J^H J\Delta\boldsymbol{m}\right). \qquad (3.36)\end{aligned}$$

Because the parameters $\boldsymbol{m}$ are real-valued, $\Delta\boldsymbol{m}^H$ is equivalent to $\Delta\boldsymbol{m}^T$. For the second and third terms in equation (3.36) we get

$$\begin{aligned}\Delta\boldsymbol{d}^H J\Delta\boldsymbol{m} + \Delta\boldsymbol{m}^H J^H\Delta\boldsymbol{d} &= \Delta\boldsymbol{m}^T J^T\overline{\Delta\boldsymbol{d}} + \Delta\boldsymbol{m}^T J^H\Delta\boldsymbol{d} \\ &= \Delta\boldsymbol{m}^T\underbrace{\overline{J^H\Delta\boldsymbol{d}}}_{\overline{\boldsymbol{c}}} + \Delta\boldsymbol{m}^T\underbrace{J^H\Delta\boldsymbol{d}}_{\boldsymbol{c}}.\end{aligned}$$

Considering the sum of a complex number $z = \mathsf{Re}(z) + i\mathsf{Im}(z)$ (with the imaginary unit i) and its complex conjugate $\overline{z} = \mathsf{Re}(z) - i\mathsf{Im}(z)$

$$z + \overline{z} = \mathsf{Re}(z) + i\mathsf{Im}(z) + \mathsf{Re}(z) - i\mathsf{Im}(z) = 2\mathsf{Re}(z) \qquad (3.37)$$

we arrive at

$$\Phi^*(\boldsymbol{m}) = \tfrac{1}{2}\left(\Delta\boldsymbol{d}^H\Delta\boldsymbol{d} + 2\mathsf{Re}(J^H\Delta\boldsymbol{d})\Delta\boldsymbol{m} + \Delta\boldsymbol{m}^H J^H J\Delta\boldsymbol{m}\right).$$

Using the differentiation rules given in Appendix A.3 and again equation (3.37) the gradient of $\Phi^*(\boldsymbol{m})$ is given by

$$\frac{\partial \Phi^*(\boldsymbol{m})}{\partial \boldsymbol{m}} = \frac{\partial \Phi^*(\boldsymbol{m})}{\partial \Delta \boldsymbol{m}} \cdot \cancelto{1}{\frac{\partial \Delta \boldsymbol{m}}{\partial \boldsymbol{m}}} = \tfrac{1}{2}\left(\mathbf{0} + 2\mathsf{Re}(J^H \Delta \boldsymbol{d}) + 2\mathsf{Re}(J^H J)\Delta \boldsymbol{m}\right)$$
$$= \mathsf{Re}(J^H \Delta \boldsymbol{d}) + \mathsf{Re}(J^H J)\Delta \boldsymbol{m}.$$

Adding the regularization part as given in equation (2.41) we finally get the normal equations for complex data:

$$(\mathsf{Re}(J^H J) + \beta W^T W)\Delta \boldsymbol{m}_k = \mathsf{Re}(J^H \Delta \boldsymbol{d}) + \beta W^T W(\boldsymbol{m}_{\mathsf{ref}} - \boldsymbol{m}_k). \tag{3.38}$$

As described in Section 2.3.1 for the DC resistivity method, this linear system of equations can be solved directly using MATLAB's `mldivide` or PARDISO. But for direct solving we have to assemble the full system matrix $\mathsf{Re}(J^H J) + \beta W^T W$ which is very large and dense especially for 3D problems. Therefore, we suggest and prefer to use an iterative solver and more precisely again the Krylov subspace method (see Section 2.3.2) PCG. The algorithms which calculate the product of J or $J^H J$ with a vector $\boldsymbol{v}$ in order to avoid the explicit assembly of the sensitivity matrix J (see Section 2.3.3 and Pseudocodes 4 and 5) need to be changed according to (3.38) for complex valued-data as shown in Pseudocode 9 and 10.

Pseudocode 9 MULTJ.M

1: $G \leftarrow A'(\boldsymbol{m}) \times_2 \boldsymbol{u}$
2: **if** transpose **then**
3: $\boldsymbol{w}_1 \leftarrow \boldsymbol{v}(1 : \mathtt{end} - \mathtt{size}(\mathtt{W}, 1))$
4: $\boldsymbol{w}_2 \leftarrow \boldsymbol{v}(\mathtt{end} - \mathtt{size}(\mathtt{W}, 1) + 1 : \mathtt{end})$
5: $\boldsymbol{w}_1 \leftarrow Q^H \boldsymbol{w}_1$
6: Solve $-A(\boldsymbol{m})^H \boldsymbol{z} = \boldsymbol{w}_1$
7: $\boldsymbol{y}_1 \leftarrow G^T \boldsymbol{z}$
8: $\boldsymbol{y}_2 \leftarrow \sqrt{\beta} W^T \cdot \boldsymbol{w}_2$
9: **return** $\boldsymbol{y}_1 + \boldsymbol{y}_2$
10: **else**
11: $\boldsymbol{z} \leftarrow G\boldsymbol{v}$
12: Solve $-A(\boldsymbol{m})^T X = Q^T$
13: $\boldsymbol{y}_1 \leftarrow X^T \cdot \boldsymbol{z}$
14: $\boldsymbol{y}_2 \leftarrow \sqrt{\beta} W \cdot \boldsymbol{v}$
15: **return** $[\boldsymbol{y}_1, \boldsymbol{y}_2]^T$
16: **end if**

Pseudocode 10 MULTJHJ.M

1: $\boldsymbol{Jv} \leftarrow$ `multJ(..., v, 'notransp')`;
2: $\boldsymbol{J^H Jv} \leftarrow$ `multJ(..., Jv, 'transp')`;
3: **return** Re($\boldsymbol{J^H Jv}$) $\leftarrow$ `real(J^H Jv)`;

The correct calculation of the derivative of the system matrix A as well as the assembly of the full sensitivity matrix is verified within `taylor_test.m` and explained in Appendix A.5.2.6.

The complex-valued Gauss-Newton scheme as well as the frequency-dependence of the system matrix and measurement operator (see Section 3.2.3) are the most important differences in comparison with the DC resistivity inversion algorithm. All other features are carried over from the DC code. Again, we use the smoothness regularization and the damping algorithm described there. For further details see Section 2.3.5 and the following section.

3.3.2 Inversion results

3.3.2.1 Inversion of electric field components

For the inversion of MT data we choose to use the 3D-2 COMMEMI model as described in Section 3.2.5.3. From the mathematical point of view, we want to use all available information for the inversion, which means three components of the electric field $\boldsymbol{E}$. As shown for example in Figure 3.5, the y- and z-component of $\boldsymbol{E}$ differ only a little from zero for numerical reasons and they are supposed to be zero as given in equation (3.4). Nevertheless, we show all components of the synthetic data set in Figure 3.8 (black circles). The most important input parameters are given in Table 3.3. We use a relatively coarse mesh consisting of 3 687 tetra-

No. of elements	3 687
No. of DOFs	24 516
No. of parameters	2 241 (without air layer)
Frequency	0.01 Hz
Data	50 receivers between $x = 70$ km and $x = -70$ km ($y = z = 0$ m) $\rightarrow$ 150 electric field components, no noise
Starting model	homogeneous, $\sigma_0 = 0.01 \frac{\text{S}}{\text{m}}$
Regularization	$\beta_1 = 10^2$, $\beta_i = \max(\beta_{i-1}/10, 10^{-8})$, layered halfspace

Tab. 3.3: Input parameter for the inversion algorithm

hedra. We need the air layer to calculate a correct forward response which is the synthetic data and the air conductivity is set to be $10^{-9}\,\frac{S}{m}$. We set Dirichlet conditions for a layered halfspace and calculate the analytical boundary values with the function `getE1dMT` (Section

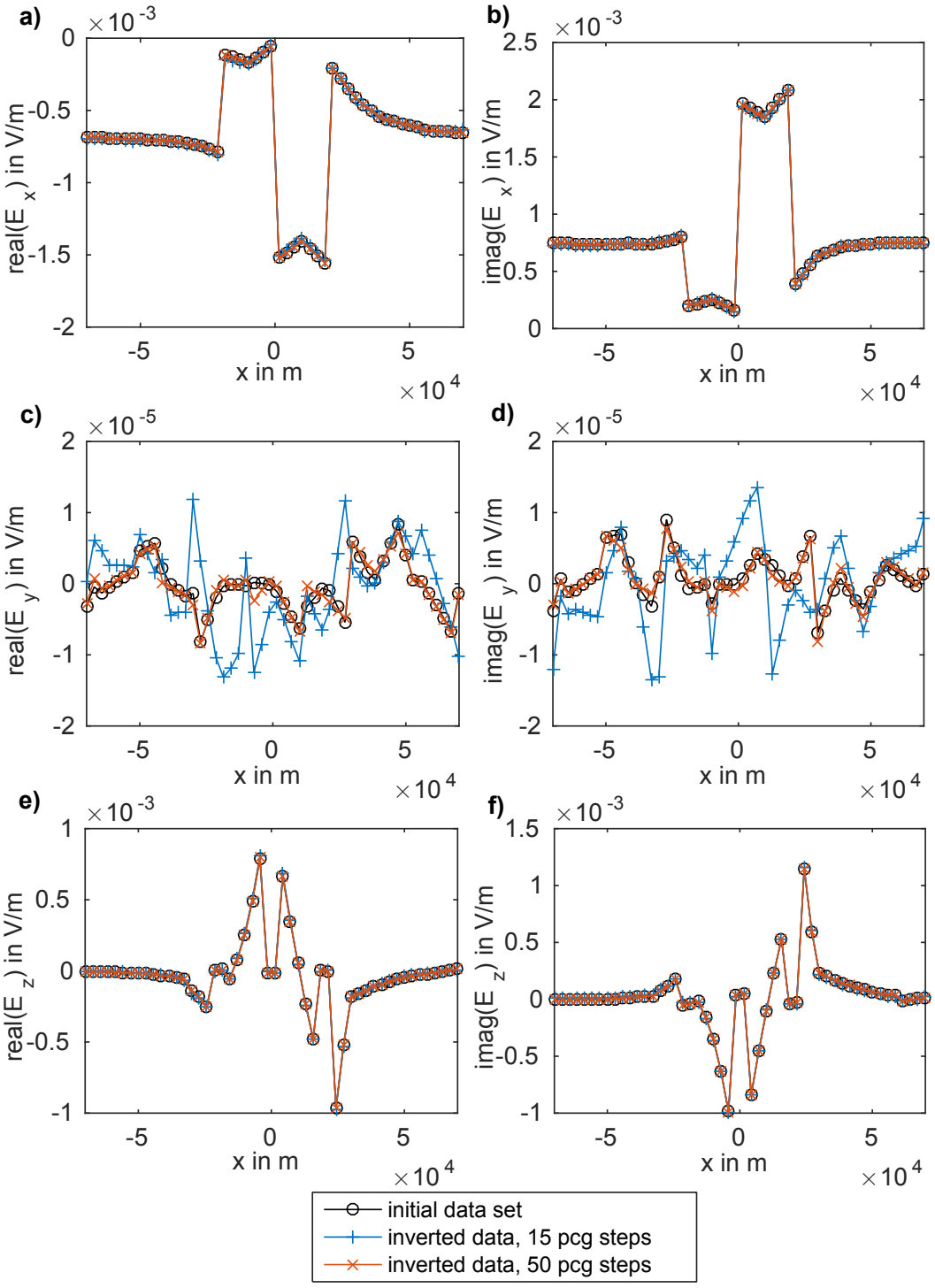

Fig. 3.8: Initial data set in comparison with the inversion result for real (left) and imaginary part (right) of the three electric field components and a different number of pcg steps.

3.2.1). In the electromagnetic inversion algorithm, we leave the parameters belonging to the air layer out. The solution of the least squares problem within the Gauss-Newton scheme is done by MATLAB's pcg with two different maximum number of steps (15 and 50) and an implicit calculation of the sensitivity matrix (see Section 3.3.1). Because we do not use a preconditioner, pcg is equal to the standard conjugate gradient method. The inversion algorithm is terminated with the damping criterion described in Section 2.3.6 or after a maximum iteration number of 25 Gauss-Newton steps. Figure 3.8 shows the real and imaginary part of measured and inverted data for all three components of the electric field. These results are the only ones which provide information to evaluate the quality of the inversion result for a real geophysical measurement. Most of the time we know nothing or only a few details about the conductivity distribution under the Earth's surface. Therefore, we can only solve the inverse problem by matching the inverted data set with the measured one. In Figure 3.8 we see that there is almost no difference between the measured data and the data for the inverted model parameters for the x- and z-component of the electric field. Even for a very small number of pcg steps, the approximation is sufficiently good. The inverted y-component differs a lot from the initial data set for 15 pcg steps. This component is almost zero or at least about two magnitudes smaller than the other components which leads to numerical inaccuracies. These are overcome when increasing the number of pcg steps which is shown in the curve with red crosses. A more detailed differentiation between more or fewer pcg steps can be achieved by

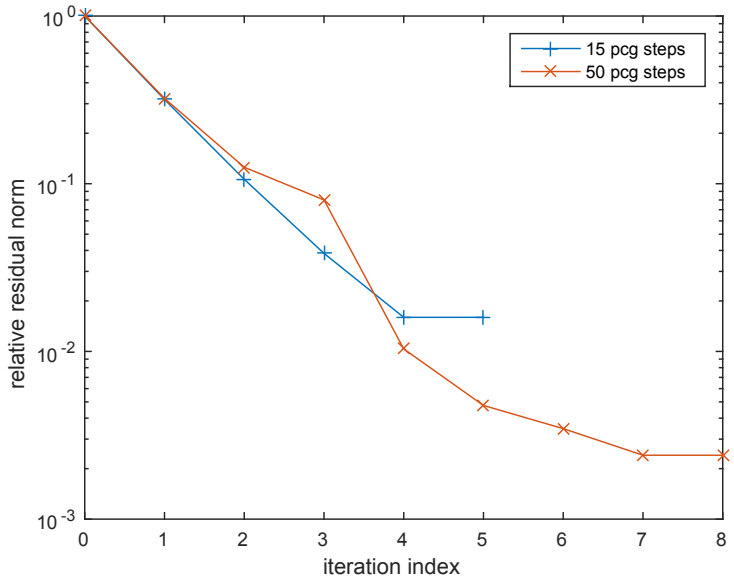

Fig. 3.9: Relative data residual norm for a different number of pcg steps.

considering the relative data residual norm in Figure 3.9 given by

$$\text{relative data residual norm} = \frac{||Q\boldsymbol{u}_i - \boldsymbol{b}||_2}{||Q\boldsymbol{u}_0 - \boldsymbol{b}||_2} = \frac{||\boldsymbol{r}_i||_2}{||\boldsymbol{r}_0||_2} \tag{3.39}$$

with the index 0 for the starting model and i for the result of the ith iteration step. Here, the approximation of all three components of the electric field has influence on the residual norm and makes a difference during the inversion steps. Obviously, because of the difficult reconstruction of the E_y-component, more pcg steps lead to a better result for the full electric field vector and improve the relative data residual norm by about one order of magnitude: For 15 pcg steps the inversions stops after five Gauss-Newton steps with a relative data residual norm of about $1.6 \cdot 10^{-2}$. For the higher number of pcg steps we reach a relative residual norm of $2.4 \cdot 10^{-3}$ after 8 Gauss-Newton iterations. Finally, we want to have a look at the reconstructed parameter model. For our synthetic model problem, we have the

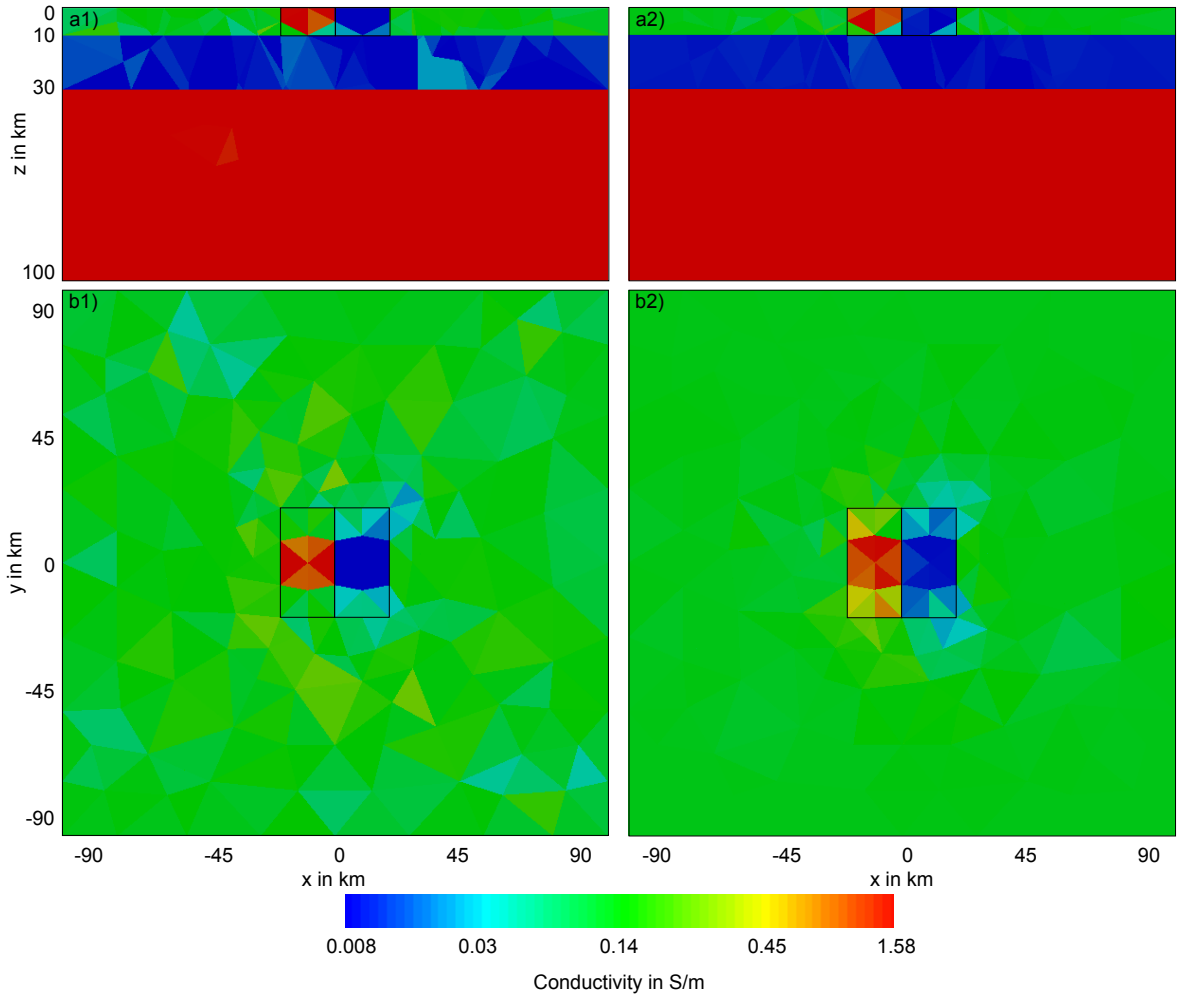

Fig. 3.10: 3D-2 COMMEMI model: Frontal slice (a) and topview (b) of inversion result for 15 (left) and 50 (right) pcg steps.

possibility to compare the result with the exact solution. Figure 3.10 shows a frontal slice and topview of the inversion result, again for 15 and 50 pcg steps. The exact dimension of the embedded blocks is given by the black contour. First of all, we see the good approximation of the layered background. Although we start our inversion with a homogeneous halfspace we arrive at the three layers. The embedded blocks can be seen in all four pictures. Especially for 50 pcg steps, the reconstruction of the blocks' lateral extension is better and the first and second layer is more homogeneous (lower figures). We might get a better result with additional receiver profiles along the faces of the blocks or data from a second polarization direction. The resolution in depth is better for the resistive block which originates in the skin effect: In the conductive block, the amplitude of the electric field decays much faster and therefore yields less information about the lower boundary. The conductivity values themselves are of correct order for the different bodies. The right block has the same conductivity as the second layer and the left block is a bit less conductive than the lowest layer which is the most conductive. The good quality of this result is not least because we use noiseless input data. Furthermore, we use the same mesh to produce the synthetic data set and to carry out the inversion: The conductivity structure is present in the mesh, which is a kind of regularization, and therefore the inversion result is forced to the given structure (*inverse crime*, see Section 2.3.7). Nevertheless, using this mesh both for modeling and inversion is sufficient for investigating the functionality and performance of the modeling and inversion code.

3.3.2.2 Inversion of electromagnetic impedances

Using three electric field components for the inversion process is not based on physics. A real MT data set for one polarization as we use it provides only the x-component of the electric field. In addition, the associated magnetic field component H_y can be measured which we can calculate using the measurement operator Q_H described in Section 3.2.3. The ratio of these components is called impedance Z and it is the quantity that is evaluated to extract the conductivity structure from measured data (Vozoff, 1988):

$$Z = \frac{E_x}{H_y}. \tag{3.40}$$

For a homogeneous halfspace, this ratio can be used to calculate the resistivity as follows:

$$\rho_{xy} = \frac{1}{\mu\omega} \left| \frac{E_x}{H_y} \right|^2 .$$

In case of inhomogeneities, this ratio depends on the frequency and is an apparent resistivity $\rho_{\mathsf{a}}(f)$. If we add the second polarization, the impedance becomes a tensor:

$$\begin{bmatrix} E_x \\ E_y \end{bmatrix} = \begin{bmatrix} Z_{xx} & Z_{xy} \\ Z_{yx} & Z_{yy} \end{bmatrix} \begin{bmatrix} H_x \\ H_y \end{bmatrix}.$$

For our model problem, only equation (3.40) is relevant. Using the impedance as input data set for our inversion algorithm requires a change in the calculation of the sensitivity matrix J which is the derivative of the data with respect to the model parameters. We derive the new matrix using equations (3.33), (3.34) and (3.40):

$$\begin{aligned} J = \frac{\partial Z}{\partial \boldsymbol{m}} &= \frac{\partial}{\partial \boldsymbol{m}} \frac{E_x}{H_y} \\ &= \frac{\frac{\partial E_x}{\partial \boldsymbol{m}} \cdot H_y - \frac{\partial H_y}{\partial \boldsymbol{m}} \cdot E_x}{H_y^2} \\ &= \frac{1}{H_y} \left(\frac{\partial E_x}{\partial \boldsymbol{m}} - Z \frac{\partial H_y}{\partial \boldsymbol{m}} \right) \\ &= \frac{1}{H_y} \left(\frac{\partial Q_E \boldsymbol{u}}{\partial \boldsymbol{m}} - Z \frac{\partial Q_H \boldsymbol{u}}{\partial \boldsymbol{m}} \right) \\ &= \frac{1}{H_y} \left(Q_E - Z Q_H \right) \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{m}}. \end{aligned}$$

The derivative $\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{m}}$ is the sensitivity matrix defined for the DC resistivity method in equation (2.50) and we just have a new measurement operator Q_Z:

$$\begin{aligned} J &= -\frac{1}{H_y} \left(Q_E - Z Q_H \right) A(\boldsymbol{m})^{-1} \left(\frac{\partial A(\boldsymbol{m})}{\partial \boldsymbol{m}} \times_2 \boldsymbol{u} \right) \\ &= -Q_Z A(\boldsymbol{m})^{-1} \left(\frac{\partial A(\boldsymbol{m})}{\partial \boldsymbol{m}} \times_2 \boldsymbol{u} \right). \end{aligned}$$

We only have to change our measurement operator to Q_Z in the inversion algorithm – particularly in the explicit or implicit calculation of the sensitivity matrix J. Now we want to use this realistic MT data type – the impedance – to invert data from the COMMEMI model. This means that we only use one component of the electric and magnetic field as input data which is less than for the inversion example in the previous section. We try to improve the resolution by increasing the number of `pcg` steps up to 150. Furthermore, to avoid the *inverse crime*, we add three percent random noise to the synthetic impedance data set:

$$\boldsymbol{b}_Z = \boldsymbol{b}_Z \cdot (1 + \mathsf{noise} \cdot \boldsymbol{r}),$$

where $\boldsymbol{r}$ is a vector of the same dimension as $\boldsymbol{b}_Z$ and contains normally distributed random numbers and the scalar factor noise is set to 0.03. For noisy data, we know from our experience that we have to increase the initial and smallest regularization parameter β_0 and β_{min}

to smoothen the result:

$$\beta_0 = 10^4$$
$$\beta_i = \max(\beta_{i-1}/10, 10^{-3}).$$

All other input parameters as well as the Dirichlet boundary conditions are the same as for the inversion of the electric field components given in Table 3.3. A further improvement of the inversion result might be achieved by adding at least a second frequency which provides information for another depth range. We invert the impedance data for the single frequency

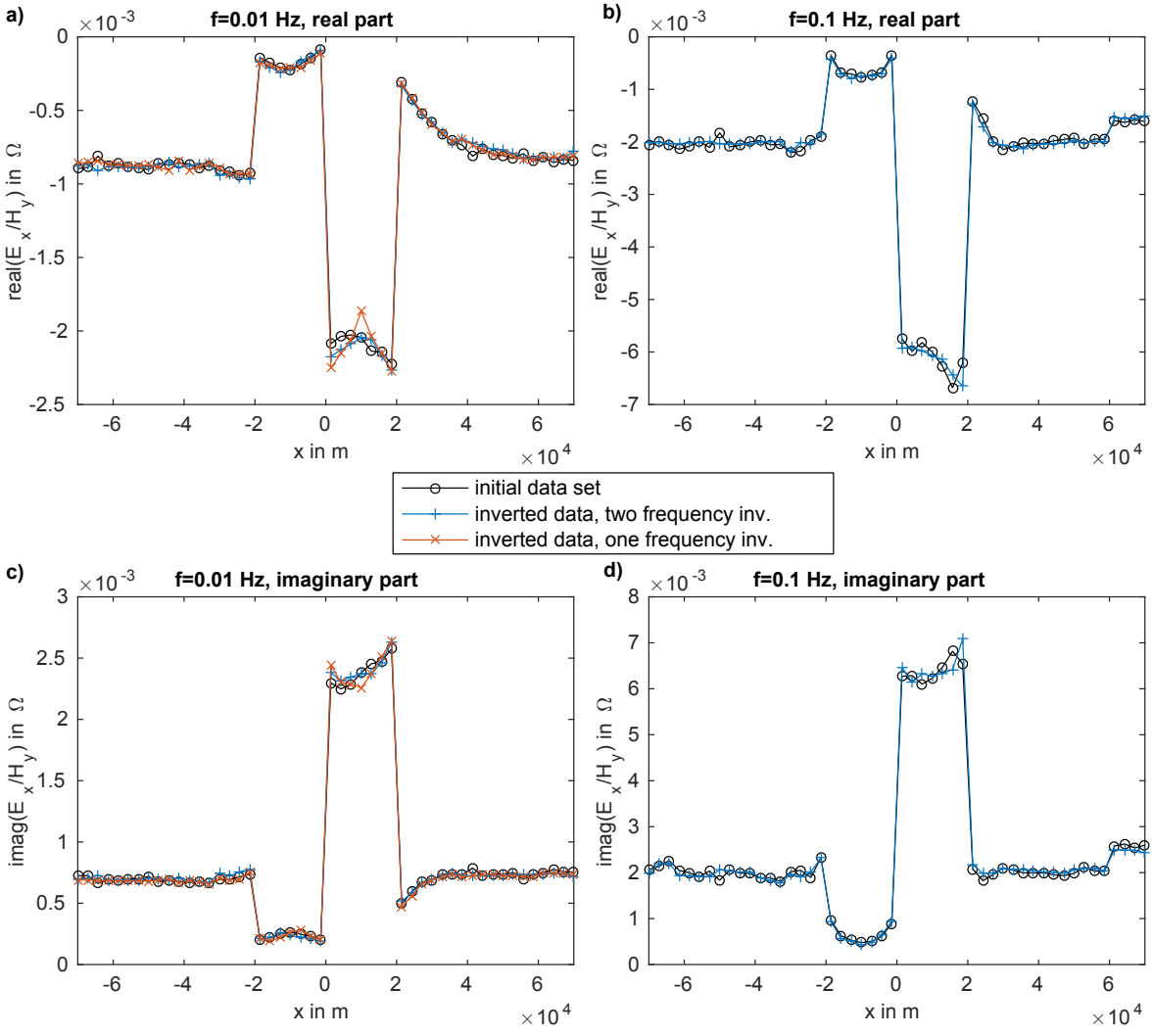

Fig. 3.11: Real (a and b) and imaginary (c and d) part of the initial noisy data set in comparison with the inversion result for one (a and c) and two frequencies (b and d).

of 0.01 Hz in the first case and add the data generated by a frequency of 0.1 Hz to the second inversion. We choose a larger frequency to improve the resolution of the two near-surface blocks. Figure 3.11 gives an impression of the noisy data set (black circles) for

both frequencies we used. Obviously the curves are not as smooth as they were for the previous data set without noise, but we can still see the sharp boundaries between the different conductivity structures. In comparison with the noiseless data set, the inversion stops a bit earlier. That is due to the smoothness regularization which is used for smoothing the inversion result and eliminating the noise which also improves the convergence of the Gauss-Newton scheme. The approximation of the measured data sets is sufficient and we do not see a difference in quality for the misfit between real and imaginary parts of the impedance. A complete match is not possible because we added data noise which is not based on the conductivity structures. If we compare the inversion for one and two frequencies, we see a slight difference in the data misfit for the resistive block which is located between $x = 0\,\text{km}$ and $x = 20\,\text{km}$. A data set for two different frequencies yields more information about the subsurface and especially the larger second frequency we used enables us to improve the near-surface resolution. Therefore, the blue curve is more consistent with the initial data set (black circles) especially in the region of the resistive block. The development of the relative data residual norm (equation (3.39)) during the inversion is shown in Figure 3.12. For one frequency, the inversion stops after six inversion steps with a relative residual norm of $1.75 \cdot 10^{-2}$ and it stops after seven steps with a relative residual norm of about $1.73 \cdot 10^{-2}$ for two frequencies.

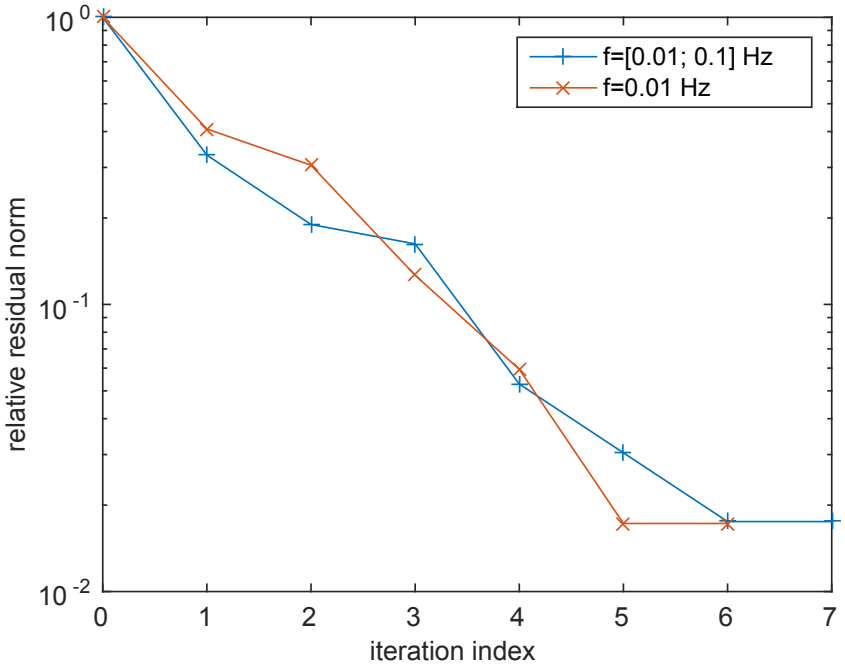

Fig. 3.12: Relative data residual norm for the inversion with one (red) and two (blue) frequencies.

Figure 3.13 shows the reconstructed conductivity model. First of all, if we compare the conductivity distribution for the inversion of three electric field components (Figure 3.10) with the result for the impedance inversion and a single frequency (Figures 3.13a1 and b1), we can see that the resolution in depth is almost equal for both results. The inversion with three electromagnetic field components provides a better impression of the dimension of the embedded blocks at the Earth's surface because it has access to more information for each receiver location than the impedance. Overall however we can achieve a good data approximation for the inversion of real MT data set and the conductivity structure is reconstructed as well. The figures on the right-hand side show the reconstructed parameter model for the impedance data for two frequencies. As stated before, with the second frequency we add information about the subsurface and improve the ratio of data values to unknown parameters. This

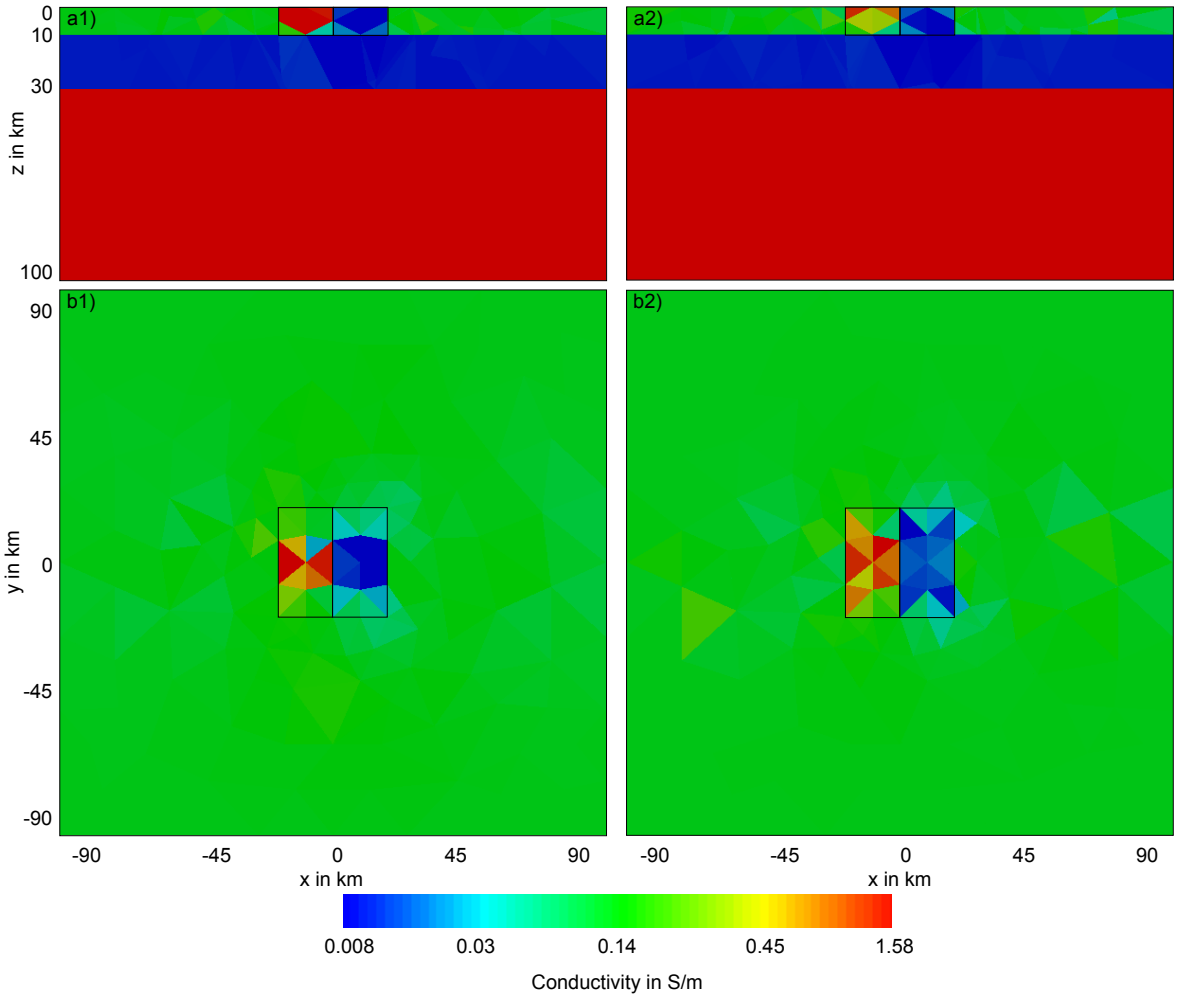

Fig. 3.13: 3D-2 COMMEMI model: Frontal slice (a) and topview (b) of inversion result for impedance data of one (left) and two frequencies (right).

leads to a better parameter model especially if we have a look at the resistive block colored in blue. The absolute conductivity value of the conductive block changes towards the true value, too.

A further improvement of the inversion results for one polarization direction might be achieved by adding a set of frequencies and weighting them amongst each other according to their depth range (skin depth, see Section 3.1). Additional parallel profiles with further input data will yield helpful information, too. Furthermore, we can examine the influence of a suitable preconditioner for the PCG method or add another regularization operator.

To achieve a standardization of our codes, we also implemented this problem in MATLAB. An overview on the structure of the electromagnetic code and implemented scripts and functions is given in Appendix A.5.2.

4 Joint inversion of DC resistivity and electromagnetic data

4.1 A sequential approach to joint inversion

The interpretation and inversion of geophysical data has been investigated for many decades and even the combined or joint inversion of different geophysical methods is not new, but there are significant differences in each scheme. In general, the term *joint inversion* stands for the combination of two or more different geophysical methods and sometimes different petrophysical parameters in one inversion approach to enhance the overall resolution power. Using joint inversion we can overcome some of the ambiguities and uncertainties which are intrinsic in each of the individual methods (Sasaki (1989), Raiche et al. (1985), Haber and Oldenburg (1997)). Each method is sensitive to a certain depth and parameter range and yields distinct parameter models of the Earth (Albouy et al., 2001). We can exploit their complementarity to get a better approximation of the measured data and a more precise image of the subsurface (Jegen et al., 2009). In some cases, it is more cost-effective to measure a restricted number of data sets of different methods than a large amount of data belonging to one method and there might be a structure which cannot be detected by any of the individual inversions but can be resolved using joint inversion (Vozoff and Jupp, 1975). The location and resolution of targets might be more precise.

There are several possibilities to implement a combined inversion using different data sets. Most of the literature, e.g. Meju (1996), Meju (2005) or Commer and Newman (2009), talk about a joint inversion where the minimization problem given in equation (2.37) is extended by adding a data residual norm for each additional geophysical method:

$$\Phi(\boldsymbol{m}) = \tfrac{1}{2}\alpha_1\|Q_1\boldsymbol{u}_1 - \boldsymbol{b}_1\|_2^2 + \tfrac{1}{2}\alpha_2\|Q_2\boldsymbol{u}_2 - \boldsymbol{b}_2\|_2^2 + \cdots + \tfrac{\beta}{2}\|W(\boldsymbol{m} - \boldsymbol{m}_{\text{ref}})\|^2 \to \min_{\boldsymbol{m}}$$

with the indices 1 and 2 for the first and second method. This additional data norm implies an extension of the sensitivity matrix, too. The scaling factors α_i, which can also be the norm of scaling matrices, are a first hint on the difficulties which arise from this approach. Now, the objective function is a sum of weighted misfit functionals and a stabilizing functional (Gribenko and Zhdanov, 2011). The different data sets and therefore the sensitivities have to

be balanced and there is a need for an appropriate weighting scheme using α_i. Otherwise we might fit only one data set and the joint models are dominated by the largest data set. The scaling factors or matrices might be chosen according to the data covariance matrices (Haber and Holtzman Gazit, 2013). Furthermore, we know from the previous chapters about the individual inversion of DC resistivity and EM data that the determination of the regularization parameter is very complex and there are no definitive rules about the right parameter for a certain inversion algorithm even for only one method. Therefore it is much more difficult to define this parameter and an appropriate smoothing regularization for different kinds of geophysical methods and data.

Within the following pages we want to introduce an alternative approach of joint inversion which almost eliminates all of these problems. In contrast to the combination of different geophysical methods and data sets in one objective function, we implement a sequential joint inversion approach. Raiche et al. (1985) propose to "sequentially use the result from one inversion to guide input to the other" in the introduction of their article, but they do not apply this approach. Another hint at sequential joint inversion is given in Commer and Newman (2009). They talk about different inversion strategies such as standalone, sequential and simultaneous inversion. Here, in the sequential inversion they use the result of the first method as starting model for the second method and note, that "the sequential approach is achieved at the lowest computational effort" and provides a better approximation of conductivities but is not as good as simultaneous joint inversion in reproducing the size of the anomaly.

We present a sequential approach to joint inversion as follows: The data set of the first method is inverted separately with its own regularization strategy and for example a homogeneous reference model equal to the background conductivity. Then, we use the final result of the first method ($\boldsymbol{m}_{\text{method }1}$) as reference model for the second method. The objective function for the minimization problem of the second geophysical method reads as follows:

$$\begin{aligned}\Phi_{\text{method }2}(\boldsymbol{m}) &= \tfrac{1}{2}\|Q\boldsymbol{u}-\boldsymbol{b}\|_2^2 + \tfrac{\beta}{2}||W(\boldsymbol{m}-\boldsymbol{m}_{\text{ref}})||^2 \\ &= \tfrac{1}{2}\|Q\boldsymbol{u}-\boldsymbol{b}\|_2^2 + \tfrac{\beta}{2}||W(\boldsymbol{m}-\boldsymbol{m}_{\text{method }1})||^2 \rightarrow \min_{\boldsymbol{m}}.\end{aligned}$$

The result of the second inversion includes information of both methods and can now be used as input for a third inversion and so on.

Because we implement a separate inversion for each geophysical method, we do not have to think about weighting the different data sets against each other. Just a weighting within one data set concerning for example depth of resolution can be useful. Furthermore, we have to find a suitable regularization strategy and parameter for only one method and do not have to include all data sets in one parameter regularization. The reference model is free for the first inversion where we usually use a homogeneous or layered background. In all following calculations we use the one we get as result of the previous inversion scheme. Another big advantage is the independence of the simulation software. It is possible to implement all the methods in a different language and just hand the final parameter vector over to another code. People from different working groups might work together more easily in this approach. The only intersection we have to think about is the inversion mesh which is the distribution of model parameters. This should be the same for all methods. Otherwise, we have to do a difficult interpolation of data which will worsen the inversion's result.

To show the simplicity and feasibility of the sequential joint inversion approach we apply it to two combinations of DC resistivity method data and an electromagnetic method below.

4.2 3D joint inversion of DC resistivity and VLF-R data

The sequential approach to joint inversion enables us to combine different data sets easily. Especially for a measuring area where we can access available data sets from former measurements and want to improve the available inversion result, we can add some new profiles for the second geophysical method and use the "old" result as reference model.

A DC resistivity measurement can last several hours even for one profile of a few hundred meters in length. If we realize a lack of data afterwards and a bad resolution of the subsurface, it would be expensive and time-consuming to measure additional DC data to improve the result, especially in difficult terrain. A good complement is the very low frequency-resistivity (VLF-R) method, which is a frequency domain electromagnetic technique that uses radio transmitters operating with low frequencies between 10 and 30 kHz and a transmission power of 100 to 1000 kW (Beamish (1998), Knödel et al. (2005d)). The sources are vertical antennas and primarily used in communication with submarines or long-range radio positioning. At sufficiently large distance from the source (in relation to the skin depth), the antenna acts as a vertical electric dipole with a vertical electric field component and the field can be considered

as a plane wave. Hence, there holds the curl-curl equation (3.7) which we can simulate and invert with the electromagnetic modeling and inversion code described in Chapter 3. The electromagnetic field spreads between the Earth's surface and the ionosphere which are approximately concentric spherical shells (Figure 4.1). The magnetic field vector oscillates in a

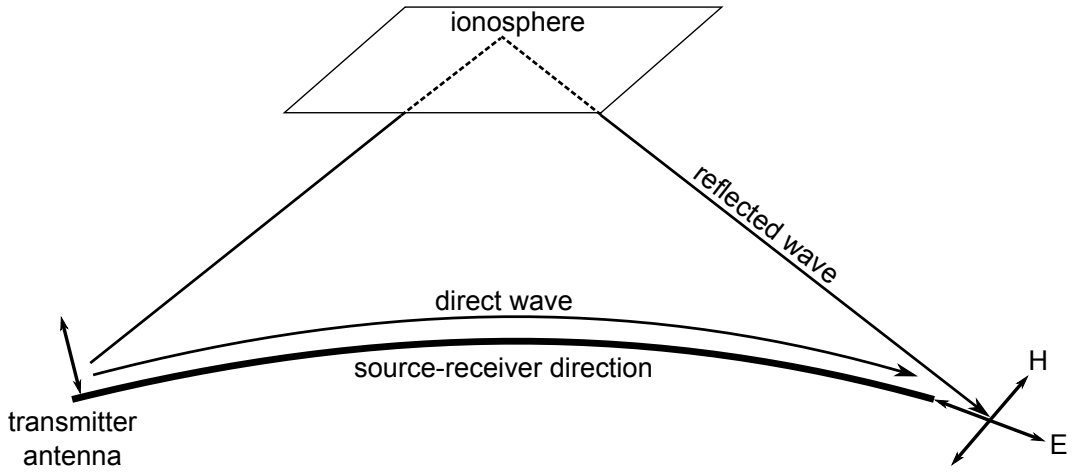

Fig. 4.1: Propagation of the electromagnetic wave and direction of the magnetic and induced electric field at the receiver location on the Earth's surface, adapted from Knödel et al. (ibid.). H is the magnetic field vector and E stands for the induced electric field vector in the subsurface.

horizontal direction which is parallel to Earth's surface (Knödel et al., 2005d). Arriving at the Earth's surface, the electromagnetic wave is partially reflected and partially transmitted into the subsurface. Because of the high conductivity contrast between the air and the conductive Earth, the transmitted part of the plane wave diffuses almost vertically into the Earth. The induced electric field is parallel to the source-receiver direction (Figure 4.1) and perpendicular to the magnetic field vector. Usually, a VLF-R measurement is carried out for one or two frequencies and source-receiver directions that are perpendicular to each other to induce perpendicular current systems and resolve the lateral boundaries of the embedded conductivity structures. Depending on the source-receiver direction, we measure the magnetic field components H_x and/or H_y using a coil oriented along the x- or y-direction and the electric field components E_x and/or E_y are measured with one pair of electrodes. The depth of investigation is guided by the skin depth given in equation (3.3) and rather small compared to an MT survey because the operating frequencies are high compared to the typical MT frequencies. For a homogeneous halfspace of 100 Ωm and a characteristic VLF-R frequency of 20 kHz, the skin depth is 35 m. Thus, the VLF-R method is used to investigate near-surface conductivity structures.

In the numerical experiments, we combine 3D DC resistivity and VLF-R data to improve the inversion result. We apply our joint inversion approach to the downscaled 3D-2 COMMEMI model (Zhdanov et al., 1997) with a vertical extent of 100 m. It was discretized into a relatively coarse mesh consisting of 2241 parameters. Figure 4.2a shows a plan view of the true model consisting of one conductive and one resistive block compared to the 0.1 $\frac{\mathrm{S}}{\mathrm{m}}$ background. The objective functions for both methods are given as follows:

$$\Phi(\boldsymbol{m}_{\mathrm{DC}}) = \|Q_1\boldsymbol{u}_1 - \boldsymbol{b}_1\|_2^2 + \tfrac{\beta_1}{2}\|W_1(\boldsymbol{m}_{\mathrm{DC}} - \boldsymbol{m}_{\mathrm{ref}})\|_2^2$$
$$\Phi(\boldsymbol{m}_{\text{VLF-R}}) = \|Q_2\boldsymbol{u}_2 - \boldsymbol{b}_2\|_2^2 + \tfrac{\beta_2}{2}\|W_2(\boldsymbol{m}_{\text{VLF-R}} - \boldsymbol{m}_{\mathrm{ref}})\|_2^2.$$

The individual inversion for the DC resistivity method was carried out using nine equidistant sources between $x/y = -30$ m and $x/y = 30$ m at the Earth's surface (Figure 4.2c, black dots) and 49 receivers inside the source configuration producing 441 apparent resistivity data points. We added three percent random noise. We use a homogeneous starting model of 0.1 $\frac{\mathrm{S}}{\mathrm{m}}$ and the layered background as reference model within our smoothness regularization. To find an optimal starting value for the regularization parameter β we use the ratio of the data residual and regularization norm as described in Section 2.3.7 and equation (2.57). According to our problem definition, we set $\beta_1 = 2.33$. To give more weight to the data residual with progressing inversion we decreased β by a factor of 10 in each Gauss-Newton step until it reached a value of 10^{-3}. A simple damping algorithm is applied (see Section 2.3.6). Figure 4.2c shows the individual DC resistivity result. Surface charges force the electric current to concentrate in a conductive body which yields a much better reconstruction of the conductive block compared to the resistive one.

For the VLF-R method, we used a transmitter frequency of 20 kHz and measured the components E_x and H_y at 50 receiver locations on a single profile at the Earth's surface ($z = y = 0$ m) between $x = -70$ m and $x = 70$ m. The synthetic data set includes three percent random noise as well. We use a resistive homogeneous starting model of 0.01 $\frac{\mathrm{S}}{\mathrm{m}}$ and again the layered reference model. The regularization parameter β was decreased by a factor of 10 in each step from 10^2 to 10^{-5}. In Figure 4.2b we see the individual VLF-R result. Due to the skin effect which describes the decay behavior of the electromagnetic field, the resolution of the resistive block is a bit better than for the conductive block. Understandably, with one profile in the center of the model, we are not able to get the exact boundaries of the embedded blocks. Data from the perpendicular polarization will improve this result.

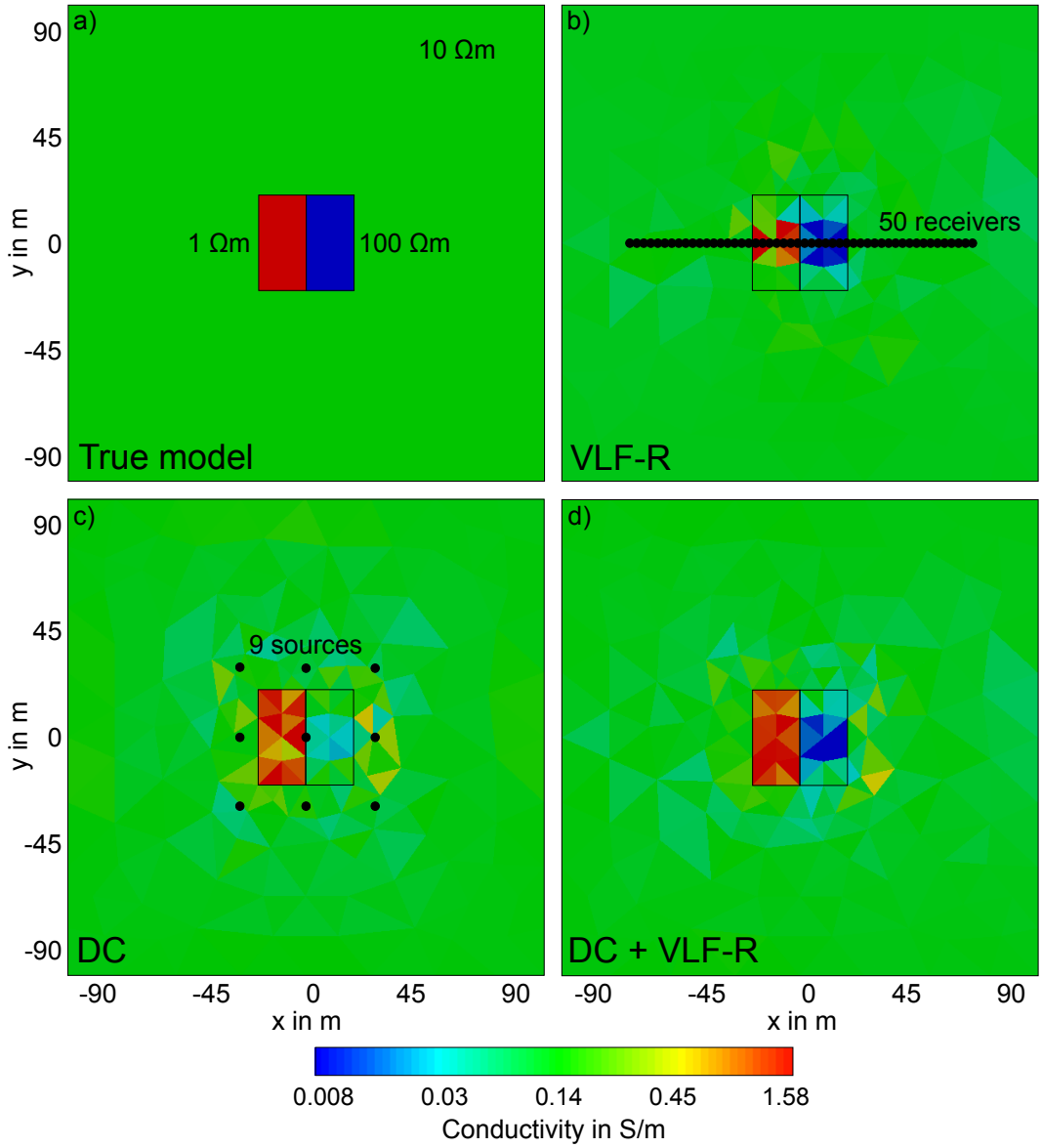

Fig. 4.2: Plan view of the downscaled 3D-2 COMMEMI model. a) The true model used to compute the synthetic data. b) and c) are the inversion results for the individual VLF-R and DC resistivity inversion, respectively. d) Using the DC resistivity solution as the reference model, we then inverted the VLF-R data to obtain a joint inversion result, which recovers the two anomalous bodies well.

The joint optimization problem is given as follows: Find the model parameters $\boldsymbol{m}_{\text{DC}}$ such that

$$\begin{aligned}\Phi(\boldsymbol{m}_{\text{DC}}) &= \|Q_1\boldsymbol{u}_1 - \boldsymbol{b}_1\|_2^2 + \tfrac{\beta_1}{2}\|W_1(\boldsymbol{m}_{\text{DC}} - \boldsymbol{m}_{\text{ref}})\|_2^2 \\ &= \|Q_1\boldsymbol{u}_1 - \boldsymbol{b}_1\|_2^2 + \tfrac{\beta_1}{2}\|W_1(\boldsymbol{m}_{\text{DC}} - \boldsymbol{m}_{\text{VLF-R}})\|_2^2\end{aligned}$$

is minimized. As described in the previous section, we use the result of one method as reference model for the second method to exploit the different sensitivity patterns. Here, we use the inverted VLF-R result as input for the DC inversion. We give a bit more weight to the

regularization norm and set the lower limit for β to be 10^{-2} in order to keep the result close to the VLF-R model. All other parameters stay the same as for the individual DC inversion. The iteration was terminated after 20 Gauss-Newton steps with a relative residual norm of $4.1 \cdot 10^{-2}$. The overall resolution of the joint result in Figure 4.2d is better than the simple addition of both individual results. We clearly see an improved image of the conductive block compared to the DC resistivity inversion as well as small improvements in the resistive block. Using the final result again as input for one of the individual inversions does not change the resulting conductivity model. It is also possible to run the sequential joint inversion the other way round: Using the DC result as reference model for the VLF-R inversion. Because there is no improvement in the final conductivity model, we leave this result out. A further sequential iteration also yields no improvement. Figure 4.3 shows the development of the relative data residual norm during the Gauss-Newton inversion for both individual data sets and both variants of the sequential joint inversion. In each case, the additional information in the reference model leads to an improved data fitting. A further step will be the improvement of the VLF-R result by adding a second data set based on the second polarization direction.

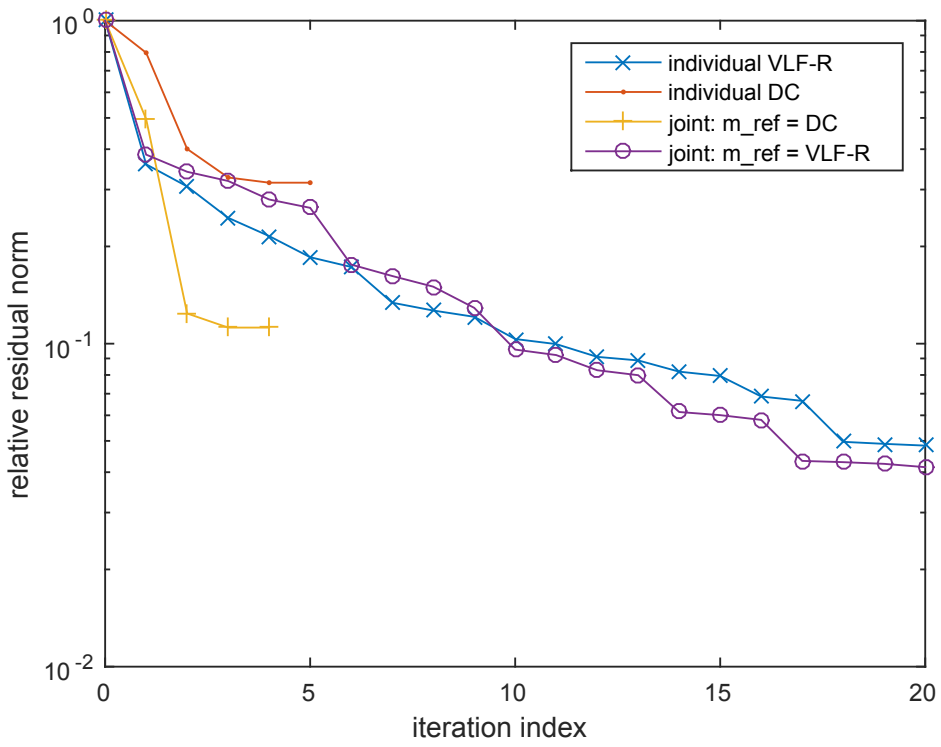

Fig. 4.3: Development of the relative data residual norm during the Gauss-Newton inversion for the individual data sets and the joint inversion approach.

4.3 2D joint inversion of DC resistivity and TEM data

We describe a 2D example from our Geotechnologien project *Three-dimensional Multi-Scale and Multi-Method Inversion to Determine the Electrical Conductivity Distribution of the Subsurface Using Parallel Computing Architectures* (Multi-EM) as presented in our science report from Ritter et al. (2014). Although we do not focus on 2D problems in this thesis, the DC-TEM joint inversion shows the functionality of the code as well as of the sequential joint inversion approach in two dimensions, too.

In contrast to frequency-domain electromagnetic methods, the time-domain or transient electromagnetic (TEM) method measures induced secondary fields when the primary field of the electromagnetic source is absent (Nabighian and Macnae (1988b), Knödel et al. (2005c)). Therefore, we do not have to perform the difficult separation of the measured field and the source field whose amplitude is usually larger. Generally, the transmitter is a large loop on the Earth's surface with a transmitting current I. This current is as abruptly turned off as feasible after all turn-on transients have vanished. The rapid power-down induces eddy currents in the subsurface and evokes a secondary field. The current system diffuses out- and downwards and dissipates because of ohmic losses. Its characteristic topology are so-called smoke rings (Nabighian and Macnae, 1988b). The propagation speed and dissipation of the corresponding field depends on the underlying structures and its conductivity. The depth of investigation for TEM methods is between several meters and about 200 meters. It is almost impossible to resolve targets which are a few meters below the surface for technical reasons: Because of the relatively fast propagation of the current system, the maximum current density has already passed into greater depths after a few microseconds before it is feasible to measure the dissipating secondary field. We are able to characterize the subsurface at a depth which is about one fourth of the depth of the maximum current density. Therefore, a combination with high frequency methods or DC resistivity surveys is highly recommended. For solving the TEM problem

$$\begin{aligned} \mu\sigma\partial_t \boldsymbol{e}(t) + \nabla \times \nabla \times \boldsymbol{e}(t) &= 0, & t &\in [0, \infty) \\ \boldsymbol{e}(t) &= \boldsymbol{e}_0(t), & t &= 0 \end{aligned}$$

with the electric field $\boldsymbol{e}(t)$, the magnetic permeability μ and the electrical conductivity σ, we apply the code described in Afanasjew et al. (2010) and Börner et al. (2015b). It uses Nédélec finite elements to discretize the spatial part of the curl-curl equation for the electric field and rational Krylov subspace techniques (Güttel, 2010) to reduce the numerical cost.

As said above, the numerical experiments are limited to two dimensions. However, the methodology applies to 3D scenarios in the same manner. To demonstrate the advantage of a joint TEM-DC approach and showcase the respective properties of the different methods, the model (Figure 4.4a) consists of two smaller bodies of high conductivity close to the surface and a larger structure of high resistivity buried at greater depth within a 100 Ωm background. First, the synthetic data for TEM and DC were independently calculated on different grids. The modeling domain extends from -6 km to 6 km in x-direction and down to 5.4 km in z-direction. The DC forward mesh consists of about 22 000 triangles and about 11 000 DOFs. The TEM modeling was carried out by Ralph-Uwe Börner on a mesh consisting of about 11 000 triangles. Our DC setup is a pole-pole configuration with four sources located at $\pm$400 m and $\pm$600 m on the Earth's surface and one borehole source at a horizontal distance of 200 m and a depth of 350 m. The 14 receivers are located along the surface between $\pm$600 m and yield 70 data points. The TEM receivers are located between -200 m and $+200$ m for two sources on the Earth's surface producing 465 data points. Then, the individual inversion processes were started for TEM and DC using a constant vector containing the logarithmized mean of the true parameter distribution for the starting model $\boldsymbol{m}_0$ and a homogeneous reference model equivalent to the background conductivity on a common inversion mesh with about 11000 parameters. This mesh differs from the mesh we used to generate the synthetic data sets. It represents a homogeneous halfspace and does not include the conductivity structures of the model given in Figure 4.4a. As for the 3D problem, we use a cooling approach for the different regularization parameters:

$$\beta_k = \max\left\{\frac{\beta_{k-1}}{10}, \beta_{\text{min}}\right\} \quad \text{with } k = 2, \ldots, n$$
$$\beta_1^{\text{DC}} \approx 10^{-1},\ \beta_1^{\text{TEM}} \approx 10^{1}, \quad \beta_{\text{min}} = 10^{-5}.$$

The objective functions are given as follows:

$$\Phi(\boldsymbol{m}_{\text{DC}}) = \|Q_1\boldsymbol{u}_1 - \boldsymbol{b}_1\|_2^2 + \tfrac{\beta_1}{2}\|W_1(\boldsymbol{m}_{\text{DC}} - \boldsymbol{m}_{\text{ref}})\|_2^2$$
$$\Phi(\boldsymbol{m}_{\text{TEM}}) = \|Q_2\boldsymbol{u}_2 - \boldsymbol{b}_2\|_2^2 + \tfrac{\beta_2}{2}\|W_2(\boldsymbol{m}_{\text{TEM}} - \boldsymbol{m}_{\text{ref}})\|_2^2.$$

Each function includes the respective residual norm with the individual measured data, the forward solution and the measurement operator as well as the regularization norm without any connection to the second geophysical method. The result of the Gauss-Newton scheme are the models $\boldsymbol{m}_{\text{DC}}$ and $\boldsymbol{m}_{\text{TEM}}$ displayed in Figures 4.4b and 4.4c, respectively. Each method yields an individual parameter model and has no information from the other method at that point. It is easy to see that the TEM configuration is not able to resolve the deeper

structure while being able to distinguish the two shallow bodies quite clearly. The DC inversion reconstructs the large structure at depth while failing to separate the smaller objects to our satisfaction. The joint optimization problem is given as follows: Find the model parameters $\boldsymbol{m}_{\text{DC}}$ such that

$$\begin{aligned}\Phi(\boldsymbol{m}_{\text{DC}}) &= \|Q_1\boldsymbol{u}_1 - \boldsymbol{b}_1\|_2^2 + \tfrac{\beta_1}{2}\|W_1(\boldsymbol{m}_{\text{DC}} - \boldsymbol{m}_{\text{ref}})\|_2^2 \\ &= \|Q_1\boldsymbol{u}_1 - \boldsymbol{b}_1\|_2^2 + \tfrac{\beta_1}{2}\|W_1(\boldsymbol{m}_{\text{DC}} - \boldsymbol{m}_{\text{TEM}})\|_2^2\end{aligned}$$

is minimized. Again, we use the result of one method as reference model for the second method to exploit the different sensitivity patterns. Figure 4.4d shows the DC inversion result

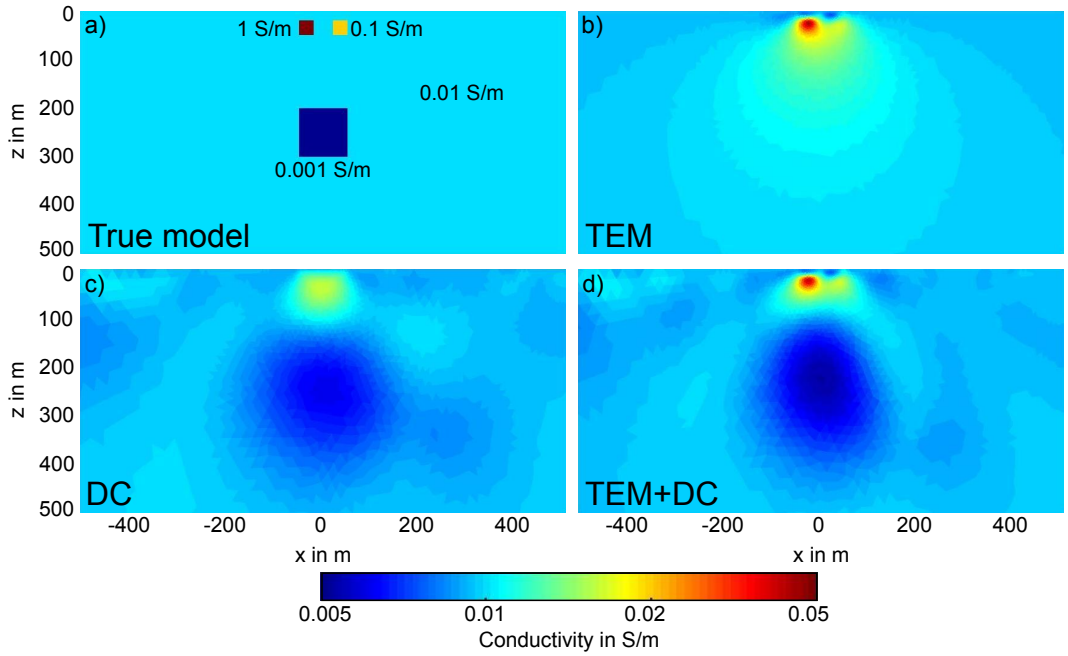

Fig. 4.4: a) The true 2D model used to compute the synthetic data. Figures b) and c) show the inversion results for the individual inversion of only TEM or DC data, respectively. d) Using the TEM solution as the reference model, we then inverted the DC resistivity data to obtain a joint inversion result, which recovers the three anomalous bodies well. All figures are cut out from the complete modeling domain.

using the same configuration as described above and the TEM solution 4.4b as the reference model. We increase the lower limit of the regularization parameter by one decade to keep the resulting model close to the TEM result. The inversion stops after four Gauss-Newton steps in the damping algorithm with a relative data residual norm of about 10^{-2}. Combining the individual resolution properties of these two methods yields a better image of the two conductive blocks as well as of the resistive body at greater depth.

It is also possible to use the result of the DC resistivity inversion as reference model for the TEM inversion:

$$\begin{aligned}\Phi(\boldsymbol{m}_{\mathsf{TEM}}) &= \|Q_2\boldsymbol{u}_2 - \boldsymbol{b}_2\|_2^2 + \tfrac{\beta_2}{2}\|W_2(\boldsymbol{m}_{\mathsf{TEM}} - \boldsymbol{m}_{\mathsf{ref}})\|_2^2 \\ &= \|Q_2\boldsymbol{u}_2 - \boldsymbol{b}_2\|_2^2 + \tfrac{\beta}{2}\|W_2(\boldsymbol{m}_{\mathsf{TEM}} - \boldsymbol{m}_{\mathsf{DC}})\|_2^2\end{aligned}$$

Because there is almost no difference to the joint result given in Figure 4.4d we do not show an additional figure of these results here. Another inversion using the joint result as reference model for the DC or TEM inversion also yields no improvement.

5 Conclusions and future work

In this thesis we developed new DC resistivity and EM forward modeling and inversion codes based on linear and quadratic finite elements. The basic principles of both scalar nodal Lagrange elements and vector-valued Nédélec elements are stated in a short and precise manner in order to enable others in the scientific community to understand the principles and make use of it as well as work on the code to improve and enhance it for regular use. Finite elements enable us to model and invert most types of electromagnetic data with only small changes to the code. The finite element modules are independent of the surrounding forward problem and of the geophysical method we implemented and can be used and enhanced in connection with other PDEs. Within the electromagnetic modeling and inversion code we only have to make small changes in the right-hand side of the linear system of equations which contains the source term to model almost all kinds of electromagnetic data. We tested the forward modeling codes using known analytical solutions and convergence studies to avoid programming mistakes and ensure a correct assembly of the system matrices as well as to prove the expected convergence rate of the finite element approximation. Elements of higher order lead to an improved modeling result for both the DC as well as the electromagnetic modeling code. The codes can be used to develop an optimal setting for a measurement in a realistic geological setting as shown in our paper Börner et al. (2015a).

For the inversion algorithms, we utilize a simple Gauss-Newton approach which yields a slightly different formulation in case of complex field quantities which we have to handle for all electromagnetic methods. The presented implicit strategy for the calculation of the sensitivity matrix leads to an improved performance of the code and a considerable reduction of the required memory. We investigated a smoothness regularization in a mixed finite element method with Raviart-Thomas elements. This enables us to handle the piecewise constant conductivity distribution and get sufficiently smooth inversion results as well as to add known information within a reference model. We successfully inverted synthetic DC resistivity and MT data sets from two different 3D models. The added noise overcomes the influence of the *inverse crime*. An appropriate starting value for the regularization parameter, which is the ratio of data residual and regularization norm, as well as the cooling approach yield an optimal development of the ratio between data fitting and a smooth model.

We finally introduced a sequential approach to joint inversion and successfully improved the reconstructed conductivity model by adding at least a second DC resistivity or EM data set to the inversion algorithm. The usage of the result of the first geophysical method as the reference model for the second method overcomes several problems concerning data weighting and scaling factors which occur when combining more than one data set in one objective function.

Although we present a detailed explanation of the development of our codes, there are a lot of additional features to implement in order to improve the results and enhance the functionality. All our codes have been implemented in MATLAB for reasons of clarity and comprehensibility and to create a consistent code basis within our working group. In order to improve the performance, we suggest to work with MATLAB's Parallel Computing Toolbox. The DC resistivity code is able to work with topography on the Earth's surface as we implemented an inhomogeneous Neumann boundary condition. We did preliminary tests with simple surface structures which are not included in this thesis because there should be more detailed studies to ensure the correct implementation. To be able to work with topography is indispensable in view of the inversion of a real geophysical data set. The data we worked with in this thesis are generated from synthetic conductivity models in order to validate the implementation. Now, an application to real field data is the next step. Furthermore, the electromagnetic code contains only one polarization direction which is essential for the simulation and inversion of a VLF-R data set. To obtain multifunctional software, we have to add the second polarization direction and think about other electromagnetic source types concerning the right-hand side of the PDE. In addition, the secondary field approach would reduce the computational cost for a passive electromagnetic method and overcome low convergence rates in case of source singularities. The smoothness regularization yields a good method to solve ill-posed problems and get a smooth model. In order to emphasize the sharp boundaries of embedded conductivity structures, we propose to implement a total variation regularization strategy (Hansen, 2010) and compare the results with the ones presented here.

The validation of the sequential joint inversion approach included some simple tests with a CSEM modeling and inversion code developed in our working group by Christoph Schwarzbach (Schwarzbach, 2009) and Feiyan Wang (Wang, 2016). We did not state the preliminary and fragmentary results in this thesis. Further studies will emphasize how simple a joint inversion with codes of different programming languages is when using the presented sequential approach. Only the parameter vector and the inversion mesh has to be interchanged and

converted into the appropriate format. Finally, the comparison of our sequential joint inversion approach with the usual method to combine more than one data set in one objective function will lead to a better understanding of the advantages of both approaches and improve the interpretation of geophysical data in many cases.

Acknowledgements

First and foremost, I would like to express my deep gratitude to my supervisor Professor Klaus Spitzer for giving me the opportunity to start my PhD studies in the Multi-EM research project, for his patient guidance and enthusiastic encouragement during all highs and lows of my scientific work and for the useful critique at every step of my way.

I am very grateful to Professor Oliver G. Ernst for opening the door to the world of finite elements and patiently explaining all mathematical problems as often as I asked for it. I would like to thank him and Professor Michael Eiermann for their valuable and constructive support and for their willingness to give their time so generously whenever I needed advice. I thank Professor Yuguo Li for reviewing this thesis.

Special thanks should be given to Felix Eckhofer, my colleague in the mathematics department and fellow team member during the development of the DC resistivity code, for working with me as a natural scientist on mathematical problems in a pleasant and constructive way. I thank him for countless hours of help with IT problems and for the good teamwork when preparing conference talks and posters. I would also like to extend my thanks to Saskia Stein who kindly offered me her ideas and concepts of a Nédélec element discretization. And I give thanks to Dr. Ralph-Uwe Börner for many productive discussions and for offering his implementation of Wait's algorithm and to all PhD students of our electromagnetic working group for the good teamwork, help and cheery distraction when needed.

I would like to thank the German Ministry of Education and Research (BMBF) who funded the Multi-EM project within the GEOTECHNOLOGIEN Programme, grant 03G0746A,B.

Finally, I would like to thank the understanding and encouraging support provided by my family and friends during my PhD studies.

Bibliography

Afanasjew, M., R.-U. Börner, M. Eiermann, O. G. Ernst, S. Güttel, and K. Spitzer (2010). "2D Time Domain TEM Simulation Using Finite Elements, an Exact Boundary Condition, and Krylov Subspace Methods, Extended Abstract". In: *Proceedings: 20th International Workshop on Electromagnetic Induction in the Earth, Giza, Egypt* (cit. on p. 100).

Albouy, Y., P. Andrieux, G. Rakotondrasoa, M. Ritz, M. Descloitres, J.-L. Join, and E. Rasolomanana (2001). "Mapping Coastal Aquifers by Joint Inversion of DC and TEM Soundings-Three Case Histories". In: *Ground Water* 39.1, pp. 87–97. DOI: 10.1111/j.1745-6584.2001.tb00354.x. URL: http://dx.doi.org/10.1111/j.1745-6584.2001.tb00354.x (cit. on p. 93).

Arnold, D. N., R. S. Falk, and R. Winther (2010). "Finite element exterior calculus: from Hodge theory to numerical stability". In: *Bull. Amer. Math. Soc.(NS)* 47.2, pp. 281–354 (cit. on p. 43).

Aster, R. C., B. Borchers, and C. H. Thurber (2013). "Chapter Four - Tikhonov Regularization". In: *Parameter Estimation and Inverse Problems (Second Edition)*. Ed. by Richard C. Aster, Brian Borchers, and Clifford H. Thurber. Second Edition. Boston: Academic Press, pp. 93–127. DOI: http://dx.doi.org/10.1016/B978-0-12-385048-5.00004-5. URL: http://www.sciencedirect.com/science/article/pii/B9780123850485000045 (cit. on p. 41).

Avdeev, D. and A. Avdeeva (2009). "3D magnetotelluric inversion using a limited-memory quasi-Newton optimization". In: *Geophysics* 74.3, F45–F57. DOI: 10.1190/1.3114023. eprint: http://geophysics.geoscienceworld.org/content/74/3/F45.full.pdf. URL: http://geophysics.geoscienceworld.org/content/74/3/F45 (cit. on p. 55).

Beamish, D. (1998). "Three-dimensional modelling of VLF data". In: *Journal of Applied Geophysics* 39.2, pp. 63–76 (cit. on p. 95).

Bing, Z. and S. A. Greenhalgh (2001). "Finite element three-dimensional direct current resistivity modelling: accuracy and efficiency considerations". In: *Geophysical Journal International* 145.3, pp. 679–688. DOI: 10.1046/j.0956-540x.2001.01412.x. URL: http://dx.doi.org/10.1046/j.0956-540x.2001.01412.x (cit. on p. 5).

Boyle, J., M. D. Mihajlović, and J. A. Scott (2007). *HSL MI20: an efficient AMG preconditioner*. Tech. rep. RAL-TR-2007-021 (cit. on p. 27).

– (2010). "HSL_MI20: An efficient AMG preconditioner for finite element problems in 3D". In: *International Journal for Numerical Methods in Engineering* 82.1, pp. 64–98. DOI: 10.1002/nme.2758. URL: http://dx.doi.org/10.1002/nme.2758 (cit. on p. 27).

Braess, D. (2003). *Finite Elemente - Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie*. Berlin, Heidelberg, New York: Springer-Verlag, pp. I–XIV, 1–302 (cit. on pp. 13, 43).

Brezzi, F. and M. Fortin (1991). *Mixed and Hybrid Finite Element Methods*. New York, NY, USA: Springer-Verlag New York, Inc. (cit. on pp. 42 sq.).

Börner, J. H., F. Wang, J. Weißflog, M. Bär, I. Görz, and K. Spitzer (2015a). "Multi-method virtual electromagnetic experiments for developing suitable monitoring designs: A fictitious CO_2 sequestration scenario in Northern Germany". In: *Geophysical Prospecting* 63.6, pp. 1430–1449. DOI: 10.1111/1365-2478.12299. URL: http://dx.doi.org/10.1111/1365-2478.12299 (cit. on pp. 27, 105).

Börner, R.-U., O. G. Ernst, and S. Güttel (2015b). "Three-dimensional transient electromagnetic modelling using Rational Krylov methods". In: *Geophysical Journal International* 202.3, pp. 2025–2043. DOI: 10.1093/gji/ggv224. eprint: http://gji.oxfordjournals.org/content/202/3/2025.full.pdf+html. URL: http://gji.oxfordjournals.org/content/202/3/2025.abstract (cit. on p. 100).

Cagniard, L. (1953). "Basic theory of the magneto-telluric method of geophysical prospecting". In: *Geophysics* 18.3, pp. 605–635. DOI: 10.1190/1.1437915. eprint: http://geophysics.geoscienceworld.org/content/18/3/605.full.pdf. URL: http://geophysics.geoscienceworld.org/content/18/3/605 (cit. on p. 55).

Coggon, J. H. (1971). "Electromagnetic and electrical modelling by the finite element method". In: *Geophysics* 36, pp. 132–155 (cit. on pp. 5, 11).

Commer, M. and G. A. Newman (2009). "Three-dimensional controlled-source electromagnetic and magnetotelluric joint inversion". In: *Geophysical Journal International* 178.3, pp. 1305–1316. DOI: 10.1111/j.1365-246X.2009.04216.x. URL: http://dx.doi.org/10.1111/j.1365-246X.2009.04216.x (cit. on pp. 93 sq.).

Dey, A. and H. F. Morrison (1979). "Resistivity modeling for arbitrarily shaped threedimensional structures". In: *Geophysics* 44.4, pp. 753–780. DOI: 10.1190/1.1440975. eprint: http://dx.doi.org/10.1190/1.1440975. URL: http://dx.doi.org/10.1190/1.1440975 (cit. on p. 5).

Dieter, K., N. R. Paterson, and F. S. Grant (1969). "Ip and resistivity type curves for three-dimensional bodies". In: *Geophysics* 34.4, pp. 615–632. DOI: 10.1190/1.1440035. eprint: http://geophysics.geoscienceworld.org/content/34/4/615.full.pdf. URL: http://geophysics.geoscienceworld.org/content/34/4/615 (cit. on p. 5).

Ellis, R. G. and D. W. Oldenburg (1994). "The pole-pole 3-D Dc-resistivity inverse problem: a conjugate gradient approach". In: *Geophysical Journal International* 119.1, pp. 187–194. DOI: 10.1111/j.1365-246X.1994.tb00921.x. URL: http://dx.doi.org/10.1111/j.1365-246X.1994.tb00921.x (cit. on p. 5).

Engl, H. W., M. Hanke, and A. Neubauer (2000). *Regularization of Inverse Problems*. Vol. 375. Mathematics and Its Applications. p. 31. Springer Netherlands (cit. on p. 41).

Farquharson, C. G. and J. A. Craven (2009). "Three-dimensional inversion of magnetotelluric data for mineral exploration: An example from the McArthur River uranium deposit, Saskatchewan, Canada". In: *Journal of Applied Geophysics* 68.4, pp. 450 –458. DOI: http://dx.doi.org/10.1016/j.jappgeo.2008.02.002. URL: http://www.sciencedirect.com/science/article/pii/S0926985108000657 (cit. on p. 55).

Flathe, H. (08/1974). "Geologisches Jahrbuch, Reihe E, Heft 2". In: Stuttgart, Germany: Schweizerbart Science Publishers. Chap. Aspekte zur Bewertung des Informationsinhaltes geoelektrischer Sondierungskurven bei der Prospektion auf Grundwasser, pp. 111–121. URL: http://www.schweizerbart.de//publications/detail/isbn/9783510960835/Geologisches_Jahrbuch_Reihe_E_Heft (cit. on p. 7).

Geselowitz, D. B. (1971). "An Application of Electrocardiographic Lead Theory to Impedance Plethysmography". In: *Biomedical Engineering, IEEE Transactions on* BME-18.1, pp. 38–41. DOI: 10.1109/TBME.1971.4502787 (cit. on p. 7).

Gopalakrishnan, J., L. E. García-Castillo, and L. F. Demkowicz (2005). "Nédélec spaces in affine coordinates". In: *Computers & Mathematics with Applications* 49.7–8, pp. 1285 –1294. DOI: http://dx.doi.org/10.1016/j.camwa.2004.02.012. URL: http://www.sciencedirect.com/science/article/pii/S0898122105001380 (cit. on p. 65).

Gribenko, A. V. and M. S. Zhdanov (2011). "Joint 3D Inversion of Marine CSEM And MT Data". In: *SEG Technical Program Expanded Abstracts 2011*. Society of Exploration Geophysicists. Chap. 116, pp. 574–578. DOI: 10.1190/1.2792486. eprint: http://library.seg.org/doi/pdf/10.1190/1.2792486. URL: http://library.seg.org/doi/abs/10.1190/1.2792486 (cit. on p. 93).

Günther, T., C. Rücker, and K. Spitzer (2006). "Three-dimensional modelling and inversion of dc resistivity data incorporating topography – II. Inversion". In: *Geophysical Journal International* 166.2, pp. 506–517. DOI: 10.1111/j.1365-246X.2006.03011.x. URL: http://dx.doi.org/10.1111/j.1365-246X.2006.03011.x (cit. on p. 5).

Güttel, S. (2010). "Rational Krylov methods for operator functions". PhD thesis. TU Bergakademie Freiberg. URL: http://nbn-resolving.de/urn:nbn:de:bsz:105-qucosa-27645 (cit. on p. 100).

Haber, E. (2005). "Quasi-Newton methods for large-scale electromagnetic inverse problems". In: *Inverse Problems* 21.1, pp. 305–323. URL: http://stacks.iop.org/0266-5611/21/i=1/a=019 (cit. on p. 55).

Haber, E. and M. Holtzman Gazit (2013). "Model Fusion and Joint Inversion". English. In: *Surveys in Geophysics* 34.5, pp. 675–695. DOI: 10.1007/s10712-013-9232-4. URL: http://dx.doi.org/10.1007/s10712-013-9232-4 (cit. on p. 94).

Haber, E. and D. W. Oldenburg (1997). "Joint inversion: a structural approach". In: *Inverse Problems* 13.1, pp. 63–77. URL: http://stacks.iop.org/0266-5611/13/i=1/a=006 (cit. on p. 93).

Hansen, P. C. (2010). *Discrete Inverse Problems: Insight and Algorithms*. Fundamentals of Algorithms. Philadelphia: Society for Industrial and Applied Mathematics. URL: http://books.google.de/books?id=r-uK2bzAUrUC (cit. on pp. 47, 106).

Heise, W., T. G. Caldwell, H. M. Bibby, and S. C. Bannister (05/2008). "Three-dimensional modelling of magnetotelluric data from the Rotokawa geothermal field, Taupo Volcanic Zone, New Zealand". In: *Geophysical Journal International* 173, pp. 740–750. DOI: 10.1111/j.1365-246X.2008.03737.x (cit. on p. 55).

Hestenes, M. R. and E. Stiefel (1952). "Methods of conjugate gradients for solving linear systems". In: *Journal of research of the National Bureau of Standards* 49, pp. 409–436 (cit. on pp. 5, 36).

Hohmann, G. W. (1975). "Three-dimensional induced polarization and electromagnetic modeling". In: *Geophysics* 40.2, pp. 309–324. DOI: 10.1190/1.1440527. eprint: http://geophysics.geoscienceworld.org/content/40/2/309.full.pdf. URL: http://geophysics.geoscienceworld.org/content/40/2/309 (cit. on p. 5).

Jegen, M. D., R. W. Hobbs, P. Tarits, and A. Chave (2009). "Joint inversion of marine magnetotelluric and gravity data incorporating seismic constraints: Preliminary results of sub-basalt imaging off the Faroe Shelf". In: *Earth and Planetary Science Letters* 282.1–4, pp. 47 –55. DOI: http://dx.doi.org/10.1016/j.epsl.2009.02.018. URL: http://www.sciencedirect.com/science/article/pii/S0012821X09001332 (cit. on p. 93).

Kaipio, J. and E. Somersalo (2007). "Statistical inverse problems: Discretization, model reduction and inverse crimes". In: *Journal of Computational and Applied Mathematics* 198.2. Special Issue: Applied Computational Inverse Problems, pp. 493 –504. DOI: http://dx.doi.org/10.1016/j.cam.2005.09.027. URL: http://www.sciencedirect.com/science/article/pii/S0377042705007296 (cit. on p. 49).

Keller, G. V. and F. C. Frischknecht (1966). "Galvanic Resistivity Methods". In: Electrical Methods in Geophysical Prospecting. Friedr. Vieweg and Sohn Verlag, Braunschweig. Chap. III, pp. 90–196 (cit. on pp. 8 sq.).

Knödel, K., H. Krummel, and G. Lange (2005a). "Elektromagnetische Zweispulen-Systeme". In: *Geophysik*. 2. überarbeitete Auflage. Handbuch zur Erkundung des Untergrundes von Deponien und Altlasten / BGR, Bundesanstalt für Geowissenschaften und Rohstoffe, Bd. 3. Online-Ressource : v.: digital. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg. Chap. 5.7, pp. 241–312. URL: http://dx.doi.org/10.1007/b137869 (cit. on p. 57).

– (2005b). "Gleichstromgeoelektrik". In: *Geophysik*. 2. überarbeitete Auflage. Handbuch zur Erkundung des Untergrundes von Deponien und Altlasten / BGR, Bundesanstalt für Geowissenschaften und Rohstoffe, Bd. 3. Online-Ressource : v.: digital. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg. Chap. 5.3, pp. 128–173. URL: http://dx.doi.org/10.1007/b137869 (cit. on pp. 6 sq., 9).

– (2005c). "Transientelektromagnetik". In: *Geophysik*. 2. überarbeitete Auflage. Handbuch zur Erkundung des Untergrundes von Deponien und Altlasten / BGR, Bundesanstalt für Geowissenschaften und Rohstoffe, Bd. 3. Online-Ressource : v.: digital. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg. Chap. 5.8, pp. 313–334. URL: http://dx.doi.org/10.1007/b137869 (cit. on p. 100).

– (2005d). "VLF, VLF-R und Radiomagnetotellurik". In: *Geophysik*. 2. überarbeitete Auflage. Handbuch zur Erkundung des Untergrundes von Deponien und Altlasten / BGR, Bundesanstalt für Geowissenschaften und Rohstoffe, Bd. 3. Online-Ressource : v.: digital. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg. Chap. 5.9, pp. 335–361. URL: http://dx.doi.org/10.1007/b137869 (cit. on pp. 95 sq.).

Li, Y. and K. Spitzer (2002). "Three-dimensional DC resistivity forward modelling using finite elements in comparison with finite-difference solutions". In: *Geophysical Journal International* 151.3, pp. 924–934. DOI: 10.1046/j.1365-246X.2002.01819.x. eprint: http://gji.oxfordjournals.org/content/151/3/924.full.pdf+html. URL: http://gji.oxfordjournals.org/content/151/3/924.abstract (cit. on p. 5).

Lowry, A., M. B. Allen, and P. N. Shive (1989). "Singularity removal; a refinement of resistivity modeling techniques". In: *Geophysics* 54.6, pp. 766–774. DOI: 10.1190/1.1442704. eprint: http://geophysics.geoscienceworld.org/content/54/6/766.full.pdf+html. URL: http://geophysics.geoscienceworld.org/content/54/6/766.abstract (cit. on pp. 5, 11).

Mackie, R. L. and T. R. Madden (1993). "Three-dimensional magnetotelluric inversion using conjugate gradients". In: *Geophysical Journal International* 115.1, pp. 215–229. DOI: 10.1111/j.1365-246X.1993.tb05600.x. URL: http://dx.doi.org/10.1111/j.1365-246X.1993.tb05600.x (cit. on pp. 39, 55).

Meju, M. A. (1996). "Joint inversion of TEM and distorted MT soundings; some effective practical considerations". In: *Geophysics* 61.1, pp. 56–65. DOI: 10.1190/1.1443956. eprint: http://geophysics.geoscienceworld.org/content/61/1/56.full.pdf+html. URL: http://geophysics.geoscienceworld.org/content/61/1/56.abstract (cit. on p. 93).

– (2005). "Simple relative space–time scaling of electrical and electromagnetic depth sounding arrays: implications for electrical static shift removal and joint DC-TEM data inversion with the most-squares criterion". In: *Geophysical Prospecting* 53.4, pp. 463–479. DOI: 10.1111/j.1365-2478.2005.00483.x. URL: http://dx.doi.org/10.1111/j.1365-2478.2005.00483.x (cit. on p. 93).

Monk, P. (2003). *Finite element methods for Maxwell's equations*. Oxford: Oxford University Press (cit. on pp. 13, 56, 60, 62 sq., 65, 72, 74).

Mueller, J. and S. Siltanen (2012). *Linear and Nonlinear Inverse Problems with Practical Applications*. Philadelphia, PA: Society for Industrial and Applied Mathematics. DOI: 10.1137/1.9781611972344. eprint: http://epubs.siam.org/doi/pdf/10.1137/1.9781611972344. URL: http://epubs.siam.org/doi/abs/10.1137/1.9781611972344 (cit. on p. 49).

Mufti, I. R. (1976). "Finite-difference modeling for arbitrarily shaped two-dimensional structures". In: *Geophysics* 41.1, pp. 62–78. DOI: 10.1190/1.1440608. eprint: http://dx.doi.org/10.1190/1.1440608. URL: http://dx.doi.org/10.1190/1.1440608 (cit. on p. 5).

Møller, I., K. I. Sørensen, and E. Auken (2006). "BurVal Working Group: Groundwater Resources in Buried Valleys. A Challenge for Geosciences". In: ed. by Reinhard Kirsch, Hanna-Maria Rumpel, Wolfgang Scheer, and Helga Wiederhold. Hannover: Leibniz Institute for Applied Geosciences (GGA-Institut). Chap. Geoelectrical methods, pp. 77–88. URL: http://www.liag-hannover.de/de/fsp/gws-hydro/strukturerkundung-und-parameterermittlung/burval/buch.html (cit. on pp. 7 sq.).

Nabighian, M. N. and J. C. Macnae (1988a). "Physics of the electromagnetic induction exploration method". In: *Electromagnetic Methods in Applied Geophysics*. Society of Exploration Geophysicists. Chap. 1, pp. 5–46. URL: https://books.google.de/books?id=Ec3eYTppHikC (cit. on pp. 56 sq.).

– (1988b). "Time Domain Electromagnetic Prospecting Methods". In: *Electromagnetic Methods in Applied Geophysics*. Society of Exploration Geophysicists. Chap. 6, pp. 427–520. URL: https://books.google.de/books?id=Ec3eYTppHikC (cit. on p. 100).

Nédélec, J.-C. (1980). "Mixed finite elements in $\mathbb{R}^3$". English. In: *Numerische Mathematik* 35.3, pp. 315–341. DOI: 10.1007/BF01396415. URL: http://dx.doi.org/10.1007/BF01396415 (cit. on p. 60).

Newman, G. A. and D. L. Alumbaugh (2000). "Three-dimensional magnetotelluric inversion using non-linear conjugate gradients". In: *Geophysical Journal International* 140.2, pp. 410–424. DOI: 10.1046/j.1365-246x.2000.00007.x. eprint: http://gji.oxfordjournals.org/content/140/2/410.full.pdf+html. URL: http://gji.oxfordjournals.org/content/140/2/410.abstract (cit. on p. 55).

Newman, G. A. and M. G. Hoversten (2000). "Solution strategies for two- and three-dimensional electromagnetic inverse problems". In: *Inverse Problems* 16.5, p. 1357. URL: http://stacks.iop.org/0266-5611/16/i=5/a=314 (cit. on p. 49).

Nocedal, J. and S. Wright (2006). "Least-Squares Problems". In: *Numerical Optimization*. Second. Springer Series in Operations Research and Financial Engineering. New York, NY: Springer. Chap. 10 (cit. on p. 33).

Oberai, A., M. Malhotra, and P. M. Pinsky (1998). "On the implementation of the Dirichlet-to-Neumann radiation condition for iterative solution of the Helmholtz equation". In: *Applied Numerical Mathematics* 27.4. Special Issue on Absorbing Boundary Conditions, pp. 443–464. DOI: http://dx.doi.org/10.1016/S0168-9274(98)00024-5. URL: http://www.sciencedirect.com/science/article/pii/S0168927498000245 (cit. on p. 10).

Paige, C. C. and M. A. Saunders (1982). "LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares." In: *ACM Trans. Math. Softw.* 8.1, pp. 43–71. URL: http://dblp.uni-trier.de/db/journals/toms/toms8.html#PaigeS82 (cit. on pp. 5, 36).

Park, S. K. and G. P. Van (1991). "Inversion of pole-pole data for 3-D resistivity structure beneath arrays of electrodes". In: *Geophysics* 56.7, pp. 951–960. DOI: 10.1190/1.1443128. eprint: http://geophysics.geoscienceworld.org/content/56/7/951.full.pdf. URL: http://geophysics.geoscienceworld.org/content/56/7/951 (cit. on p. 5).

Patro, P. K. and G. D. Egbert (2008). "Regional conductivity structure of Cascadia: Preliminary results from 3D inversion of USArray transportable array magnetotelluric data". In: *Geophysical Research Letters* 35.20. L20311, n/a–n/a. DOI: 10.1029/2008GL035326. URL: http://dx.doi.org/10.1029/2008GL035326 (cit. on p. 55).

Pridmore, D. F., G. W. Hohmann, S. H. Ward, and W. R. Sill (1981). "An investigation of finite-element modeling for electrical and electromagnetic data in three dimensions". In: *Geophysics* 46.7, pp. 1009–1024. DOI: 10.1190/1.1441239. eprint: http://geophysics.geoscienceworld.org/content/46/7/1009.full.pdf. URL: http://geophysics.geoscienceworld.org/content/46/7/1009 (cit. on p. 5).

Raiche, A. P., D. L. B. Jupp, H. Rutter, and K. Vozoff (1985). "The joint use of coincident loop transient electromagnetic and Schlumberger sounding to resolve layered structures". In: *Geophysics* 50.10, pp. 1618–1627. DOI: 10.1190/1.1441851. eprint: http://geophysics.geoscienceworld.org/content/50/10/1618.full.pdf+html. URL: http://geophysics.geoscienceworld.org/content/50/10/1618.abstract (cit. on pp. 93 sq.).

Ritter, O., K. Spitzer, M. Afanasjew, M. Becken, R.-U. Börner, F. Eckhofer, M. Eiermann, O. G. Ernst, A. Grayver, J. Klump, N. Meqbel, C. Nittinger, J. Thaler, U. Weckmann, and J. Weißflog (2014). "Tomography of the Earth's Crust: From Geophysical Sounding to Real-Time Monitoring: GEOTECHNOLOGIEN Science Report No. 21". In: ed. by M. Weber and U. Münch. Cham: Springer International Publishing. Chap. Three-Dimensional Multi-Scale and Multi-Method Inversion to Determine the Electrical Conductivity Distribution of the Subsurface (Multi-EM), pp. 83–93. DOI: 10.1007/978-3-319-04205-3_5. URL: http://dx.doi.org/10.1007/978-3-319-04205-3_5 (cit. on p. 100).

Roy, A. and A. Apparao (1971). "Depth Of Investigation In Direct Current Methods". In: *Geophysics* 36.5, pp. 943–959. DOI: 10.1190/1.1440226. eprint: http://geophysics.geoscienceworld.org/content/36/5/943.full.pdf+html. URL: http://geophysics.geoscienceworld.org/content/36/5/943.abstract (cit. on pp. 7, 9, 25, 30).

Rücker, C., T. Günther, and K. Spitzer (2006). "Three-dimensional modelling and inversion of dc resistivity data incorporating topography – I. Modelling". In: *Geophysical Journal International* 166.2, pp. 495–505. DOI: 10.1111/j.1365-246X.2006.03010.x. URL: http://dx.doi.org/10.1111/j.1365-246X.2006.03010.x (cit. on pp. 5, 16, 36).

Saad, Y. (2003). *Iterative Methods for sparse linear systems*. 2nd. SIAM. Chap. 6 (cit. on p. 34).

Sasaki, Y. (1989). "Two-dimensional joint inversion of magnetotelluric and dipole-dipole resistivity data". In: *Geophysics* 54.2, pp. 254–262. DOI: 10.1190/1.1442649 (cit. on pp. 5, 93).

– (1994). "3-D resistivity inversion using the finite-element method". In: *Geophysics* 59.12, pp. 1839–1848. DOI: 10.1190/1.1443571. eprint: http://geophysics.geoscience\-world.org/content/59/12/1839.full.pdf. URL: http://geophysics.geoscienceworld.org/content/59/12/1839 (cit. on p. 5).

– (2004). "Three-dimensional inversion of static-shifted magnetotelluric data". In: *Earth, Planets and Space* 56.2, pp. 239–248. DOI: 10.1186/BF03353406. URL: http://dx.doi.org/10.1186/BF03353406 (cit. on p. 55).

Schmucker, U. and P. Weidelt (1975). "Electromagnetic induction in the earth: Lecture Notes". Aarhus University (unpublished) (cit. on pp. 55, 59).

Schwarzbach, C. (2009). "Stability of finite element solutions to Maxwell's equations in frequency domain". PhD thesis. TU Bergakademie Freiberg (cit. on pp. 106, 131).

Schwarzbach, C. and E. Haber (2013). "Finite element based inversion for time-harmonic electromagnetic problems". In: *Geophysical Journal International* 193.2, pp. 615–634. DOI: 10.1093/gji/ggt006. eprint: http://gji.oxfordjournals.org/content/193/2/615.full.pdf+html. URL: http://gji.oxfordjournals.org/content/193/2/615.abstract (cit. on p. 42).

Schwetlick, H. (1979). *Numerische Lösung nichtlinearer Gleichungen*. Auch: R. Oldenbourg Verlag, München-Wien, 1979. Berlin: Deutscher Verlag der Wissenschaften (cit. on pp. 33, 46).

Siripunvaraporn, W. and G. Egbert (2007). "Data space conjugate gradient inversion for 2-D magnetotelluric data". In: *Geophysical Journal International* 170.3, pp. 986–994. DOI: 10.1111/j.1365-246X.2007.03478.x. URL: http://dx.doi.org/10.1111/j.1365-246X.2007.03478.x (cit. on pp. 39, 55).

Siripunvaraporn, W. and W. Sarakorn (2011). "An efficient data space conjugate gradient Occam's method for three-dimensional magnetotelluric inversion". In: *Geophysical Journal International* 186.2, pp. 567–579. DOI: 10.1111/j.1365-246X.2011.05079.x. URL: http://dx.doi.org/10.1111/j.1365-246X.2011.05079.x (cit. on p. 55).

Siripunvaraporn, W., G. Egbert, Y. Lenbury, and M. Uyeshima (2005). "Three-dimensional magnetotelluric inversion: data-space method". In: *Physics of the Earth and Planetary Interiors* 150.1–3. Electromagnetic Induction in the EarthElectromagnetic Induction in the Earth, pp. 3 –14. DOI: http://dx.doi.org/10.1016/j.pepi.2004.08.023. URL: http://www.sciencedirect.com/science/article/pii/S0031920104003590 (cit. on p. 55).

Spitzer, K. (1995). "A 3-D finite-difference algorithm for DC resistivity modelling using conjugate gradient methods". In: *Geophysical Journal International* 123.3, pp. 903–914. DOI: 10.1111/j.1365-246X.1995.tb06897.x. URL: http://dx.doi.org/10.1111/j.1365-246X.1995.tb06897.x (cit. on p. 5).

Strang, G. (1986). "Optimization". In: Introduction to Applied Mathematics. Massachussetts: Wellesley-Cambridge Press. Chap. 8, pp. 665–734 (cit. on p. 42).

Vozoff, K. (1988). "The Magnetotelluric Method". In: *Electromagnetic Methods in Applied Geophysics*. Society of Exploration Geophysicists. Chap. 8, pp. 641–712. DOI: 10.1190/1.9781560802686.ch8. eprint: http://library.seg.org/doi/pdf/10.1190/1.9781560802686.ch8. URL: http://library.seg.org/doi/abs/10.1190/1.9781560802686.ch8 (cit. on p. 87).

– (1990). "Magnetotellurics: Principles and practice". In: *Proceedings of the Indian Academy of Sciences - Earth and Planetary Sciences* 99.4, pp. 441–471. DOI: 10.1007/BF02840313. URL: http://dx.doi.org/10.1007/BF02840313 (cit. on p. 55).

Vozoff, K. and D. L. B. Jupp (1975). "Joint Inversion of Geophysical Data". In: *Geophysical Journal of the Royal Astronomical Society* 42.3, pp. 977–991. DOI: 10.1111/j.1365-246X.1975.tb06462.x. URL: http://dx.doi.org/10.1111/j.1365-246X.1975.tb06462.x (cit. on p. 93).

Wait, J. R. (1953). "Propagation of radio waves over a stratified ground". In: *Geophysics* 18.2, pp. 416–422. DOI: 10.1190/1.1437893. eprint: http://dx.doi.org/10.1190/1.1437893. URL: http://dx.doi.org/10.1190/1.1437893 (cit. on pp. 59, 79, 131).

Wang, F. (2016). "Isotropic and anisotropic three-dimensional inversion of frequency-domain controlled-source electromagnetic data using finite element techniques". PhD thesis. TU Bergakademie Freiberg (cit. on p. 106).

Ward, S. H. and G. W. Hohmann (1988). "Electromagnetic Theory for Geophysical Applications". In: *Electromagnetic Methods in Applied Geophysics*. Ed. by Misac N. Nabighian. Society of Exploration Geophysicists. Chap. 4, pp. 130–311. DOI: 10.1190/1.9781560802631.ch4. eprint: http://library.seg.org/doi/pdf/10.1190/1.9781560802631.ch4. URL: http://library.seg.org/doi/abs/10.1190/1.9781560802631.ch4 (cit. on pp. 59, 131 sq.).

Weißflog, J., F. Eckhofer, R.-U. Börner, M. Eiermann, O. G. Ernst, and K. Spitzer (2013). "Regularization on unstructured grids". eng. In: *Protokoll über das 25. Schmucker-Weidelt-Kolloquium für Elektromagnetische Tiefenforschung*, pp. 21–25 (cit. on p. 27).

Whitney, H. (1957). *Geometric integration theory*. Princeton Mathematical Series 21. Princeton University Press, pp. xv+387 (cit. on pp. 66, 127).

Wolters, C. H., H. Köstler, C. Möller, J. Härdtlein, L. Grasedyck, and W. Hackbusch (2007). "Numerical Mathematics of the Subtraction Method for the Modeling of a Current Dipole in EEG Source Reconstruction Using Finite Element Head Models." In: *SIAM J. Scientific Computing* 30.1, pp. 24–45. URL: http://dblp.uni-trier.de/db/journals/siamsc/siamsc30.html#WoltersKMHGH07 (cit. on p. 16).

Xiong, Z. (1992). "Electromagnetic modeling of 3-D structures by the method of system iteration using integral equations". In: *Geophysics* 57.12, pp. 1556–1561. DOI: 10.1190/1.1443223. eprint: http://geophysics.geoscienceworld.org/content/57/12/1556.full.pdf+html. URL: http://geophysics.geoscienceworld.org/content/57/12/1556.abstract (cit. on pp. 55, 80).

Yi, M.-J., J.-H. Kim, and S.-H. Chung (2003). "Enhancing the resolving power of least-squares inversion with active constraint balancing". In: *Geophysics* 68.3, pp. 931–941. DOI: 10.1190/1.1581045. eprint: http://geophysics.geoscienceworld.org/content/68/3/931.full.pdf. URL: http://geophysics.geoscienceworld.org/content/68/3/931 (cit. on p. 5).

Yuan, Y., J. Qiang, J. Tang, Z. Ren, and X. Xiao (2016). "2.5D direct-current resistivity forward modelling and inversion by finite-element–infinite-element coupled method". In: *Geophysical Prospecting* 64.3, pp. 767–779. DOI: 10.1111/1365-2478.12298. URL: http://dx.doi.org/10.1111/1365-2478.12298 (cit. on p. 5).

Zhang, J., R. L. Mackie, and T. R. Madden (1995). "3-D resistivity forward modeling and inversion using conjugate gradients". In: *Geophysics* 60.5, pp. 1313–1325. DOI: 10.1190/1.1443868. eprint: http://geophysics.geoscienceworld.org/content/60/5/1313.full.pdf. URL: http://geophysics.geoscienceworld.org/content/60/5/1313 (cit. on p. 5).

Zhao, S. and M. J. Yedlin (1996). "Some refinements on the finitedifference method for 3-D dc resistivity modeling". In: *Geophysics* 61.5, pp. 1301–1307. DOI: 10.1190/1.1444053. eprint: http://dx.doi.org/10.1190/1.1444053. URL: http://dx.doi.org/10.1190/1.1444053 (cit. on p. 5).

Zhdanov, M. S. and G. V. Keller (1994). *The geoelectrical methods in geophysical exploration*. English. Includes bibliographical references and index. Amsterdam ; New York : Elsevier (cit. on pp. 6, 8 sq.).

Zhdanov, M. S., I. M. Varentsov, J. T. Weaver, N. G. Golubev, and V. A. Krylov (1997). "Methods for modelling electromagnetic fields Results from COMMEMI—the international project on the comparison of modelling methods for electromagnetic induction". In: *Journal of*

Applied Geophysics 37.3–4. Methods for modelling electromagnetic fields, pp. 133 –271. DOI: http://dx.doi.org/10.1016/S0926-9851(97)00013-X. URL: http://www.sciencedirect.com/science/article/pii/S092698519700013X (cit. on pp. 78, 80, 97).

Zhdanov, M. S., S. Fang, and G. Hursán (2000). "Electromagnetic inversion using quasi-linear approximation". In: *Geophysics* 65.5, pp. 1501–1513. DOI: 10.1190/1.1444839. eprint: http://geophysics.geoscienceworld.org/content/65/5/1501.full.pdf. URL: http://geophysics.geoscienceworld.org/content/65/5/1501 (cit. on p. 55).

Zhou, B. and T. Dahlin (2003). "Properties and Effects of Measurement Errors on 2D Resistivity Imaging Surveys". In: *Near Surface Geophysics* 1:3, pp. 105–117 (cit. on p. 7).

Zienkiewicz, O. C., R. L. Taylor, and J. Z. Zhu (2005). *The finite element method: its basis and fundamentals*. Oxford: Butterworth-Heinemann (cit. on pp. 13, 16, 63).

A Appendix

In this appendix, we give additional information for the finite element approximation of our DC resistivity and electromagnetic problems. For the DC resistivity code, we explain the assembly of matrices and systems of equations for linear elements in 3D in Chapter 2. The appendix contains basis functions, gradients and other information for 2D and quadratic elements. For the electromagnetic code, we explain the calculation of the Nédélec moments. Furthermore, we state differentiation rules for (complex) vector-valued functions which are needed to derive the Gauss-Newton scheme in Section 3.3.1 and we explain Wait's algorithm. Finally, the structure of the implemented codes is outlined.

A.1 Lagrange elements – additional information

A.1.1 Linear elements in 2D

In comparison with the reference tetrahedron and the basis functions given in equation (2.30) for a 3D element, the overall assembly process for 2D problems is almost equal. To improve the understanding of the given DC resistivity code, we just want to state some features which are different to that in 3D. First of all, the reference element $\hat{K}$ is a unit triangle given by

$$\hat{K} = \{(\xi, \eta) \in \mathbb{R}^2 \,:\, 0 \leq \xi \leq 1, 0 \leq \eta \leq 1 - \xi\}.$$

For the transformation between the reference triangle $\hat{K}$ and an arbitrary element K in the modeling domain, there holds:

$$F_K \,:\, \hat{K} \to K, \quad \hat{K} \ni \boldsymbol{\xi} \mapsto \boldsymbol{x} \in K, \quad \boldsymbol{x} = F_K(\boldsymbol{\xi}) = B_K \boldsymbol{\xi} + \boldsymbol{b}_K$$

with

$$\begin{bmatrix} x \\ y \end{bmatrix} = \underbrace{\begin{bmatrix} x_1 - x_3 & x_2 - x_3 \\ y_1 - y_3 & y_2 - y_3 \end{bmatrix}}_{B_K} \begin{bmatrix} \xi \\ \eta \end{bmatrix} + \underbrace{\begin{bmatrix} x_3 \\ y_3 \end{bmatrix}}_{\boldsymbol{b}_K}.$$

The local basis functions in the reference element $\hat{K}$ in 2D are given by

$$\hat{\phi}_1(\xi, \eta) = \xi$$

$$\hat{\phi}_2(\xi, \eta) = \eta$$

$$\hat{\phi}_3(\xi, \eta) = 1 - \xi - \eta$$

and their gradients are

$$\begin{bmatrix} \hat{\nabla}\phi_1 & \hat{\nabla}\phi_2 & \hat{\nabla}\phi_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix}.$$

A.1.2 Quadratic elements

A.1.2.1 Basis functions and gradients

For quadratic Lagrange elements there are additional DOFs on the midpoints of all edges of the element shown in Figure A.1. For the unit triangle, the quadratic basis functions are

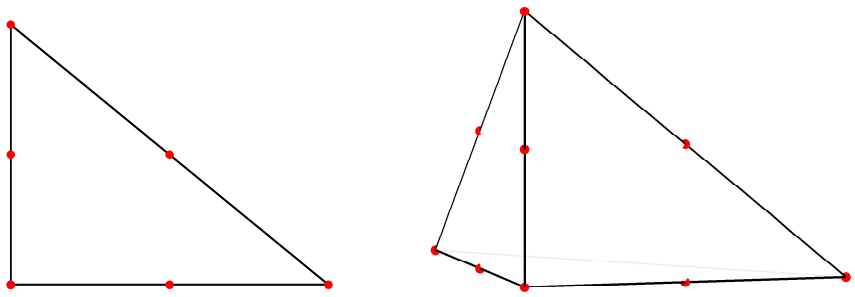

Fig. A.1: Degrees of freedom (red) for quadratic Lagrange elements in two (left) and three (right) dimensions.

$$\lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} = \begin{bmatrix} \xi \\ \eta \\ 1-\xi-\eta \end{bmatrix}, \quad \nabla\lambda = \begin{bmatrix} \nabla\lambda_1 & \nabla\lambda_2 & \nabla\lambda_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix},$$

$$\begin{aligned} \hat{\phi}_1(\xi,\eta) &= \lambda_3(2\lambda_3 - 1) & \hat{\phi}_2(\xi,\eta) &= 4\lambda_1\lambda_3 \\ \hat{\phi}_3(\xi,\eta) &= \lambda_1(2\lambda_1 - 1) & \hat{\phi}_4(\xi,\eta) &= 4\lambda_2\lambda_3 \\ \hat{\phi}_5(\xi,\eta) &= 4\lambda_1\lambda_2 & \hat{\phi}_6(\xi,\eta) &= \lambda_2(2\lambda_2 - 1). \end{aligned}$$

Their gradients are given by

$$\begin{aligned} \hat{\nabla}\phi_1(\xi,\eta) &= (4\lambda_3 - 1)\nabla\lambda_3 & \hat{\nabla}\phi_2(\xi,\eta) &= 4(\lambda_1\nabla\lambda_3 + \lambda_3\nabla\lambda_1) \\ \hat{\nabla}\phi_3(\xi,\eta) &= (4\lambda_1 - 1)\nabla\lambda_1 & \hat{\nabla}\phi_4(\xi,\eta) &= 4(\lambda_2\nabla\lambda_3 + \lambda_3\nabla\lambda_2) \\ \hat{\nabla}\phi_5(\xi,\eta) &= 4(\lambda_1\nabla\lambda_2 + \lambda_2\nabla\lambda_1) & \hat{\nabla}\phi_6(\xi,\eta) &= (4\lambda_2 - 1)\nabla\lambda_2. \end{aligned}$$

In 3D, we get the following quadratic basis functions for the reference tetrahedron:

$$\lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} \xi \\ \eta \\ \zeta \\ 1-\xi-\eta-\zeta \end{bmatrix}, \quad \nabla\lambda = \begin{bmatrix} \nabla\lambda_1 & \nabla\lambda_2 & \nabla\lambda_3 & \nabla\lambda_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix},$$

$$\begin{aligned} \hat{\phi}_1(\xi,\eta) &= \lambda_3(2\lambda_3-1) & \hat{\phi}_2(\xi,\eta) &= 4\lambda_1\lambda_3 \\ \hat{\phi}_3(\xi,\eta) &= \lambda_1(2\lambda_1-1) & \hat{\phi}_4(\xi,\eta) &= 4\lambda_2\lambda_3 \\ \hat{\phi}_5(\xi,\eta) &= 4\lambda_1\lambda_2 & \hat{\phi}_6(\xi,\eta) &= \lambda_2(2\lambda_2-1) \\ \hat{\phi}_7(\xi,\eta) &= 4\lambda_3\lambda_4 & \hat{\phi}_8(\xi,\eta) &= 4\lambda_1\lambda_4 \\ \hat{\phi}_9(\xi,\eta) &= 4\lambda_2\lambda_4 & \hat{\phi}_{10}(\xi,\eta) &= \lambda_4(2\lambda_4-1). \end{aligned}$$

The corresponding gradients are

$$\begin{aligned} \hat{\nabla}\phi_1(\xi,\eta) &= (4\lambda_3-1)\nabla\lambda_3 & \hat{\nabla}\phi_2(\xi,\eta) &= 4(\lambda_1\nabla\lambda_3+\lambda_3\nabla\lambda_1) \\ \hat{\nabla}\phi_3(\xi,\eta) &= (4\lambda_1-1)\nabla\lambda_1 & \hat{\nabla}\phi_4(\xi,\eta) &= 4(\lambda_2\nabla\lambda_3+\lambda_3\nabla\lambda_2) \\ \hat{\nabla}\phi_5(\xi,\eta) &= 4(\lambda_1\nabla\lambda_2+\lambda_2\nabla\lambda_1) & \hat{\nabla}\phi_6(\xi,\eta) &= (4\lambda_2-1)\nabla\lambda_2 \\ \hat{\nabla}\phi_7(\xi,\eta) &= 4(\lambda_4\nabla\lambda_3+\lambda_3\nabla\lambda_4) & \hat{\nabla}\phi_8(\xi,\eta) &= 4(\lambda_4\nabla\lambda_1+\lambda_1\nabla\lambda_4) \\ \hat{\nabla}\phi_9(\xi,\eta) &= 4(\lambda_4\nabla\lambda_2+\lambda_2\nabla\lambda_4) & \hat{\nabla}\phi_{10}(\xi,\eta) &= (4\lambda_4-1)\nabla\lambda_4. \end{aligned}$$

A.1.2.2 Local numbering of DOFs

To understand the quadratic version of the code we have to explain an additional feature used within the assembly process, especially in the function `fe_assemble_stiffness`. The global numbers of the nodes describing one element are given in the matrix `mesh.tetra` (in 2D: `mesh.tri`). For linear elements, this matrix only contains the vertices of the element. In the quadratic case, the additional DOFs for the ith element, which are located at the midpoints of the edges of this element, are contained in the ith column of the matrix `mesh.tetra` (in 2D: `mesh.tri`), too. Therefore, the size of this matrix changes from $4\times n_{\text{elements}}$ (in 2D: $3\times n_{\text{elements}}$) for linear elements to $10 \times n_{\text{elements}}$ (in 2D: $6 \times n_{\text{elements}}$) for quadratic elements.

We generated most of our meshes for the examples described in this thesis and implemented in the functions `fe_create_mesh` and `fe_load_mesh` using *COMSOL Multiphysics*. This software does not simply add the additional (quadratic) DOFs to the element matrix below the

vertices of the element but uses a consistent local numbering for each element. Figures A.2 and A.3 show this numbering for triangles and tetrahedra. In some cases, for example to

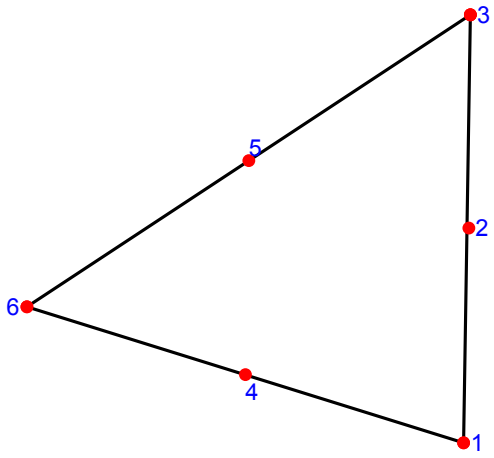

Fig. A.2: Numbered DOFs for quadratic Lagrange elements in two dimensions.

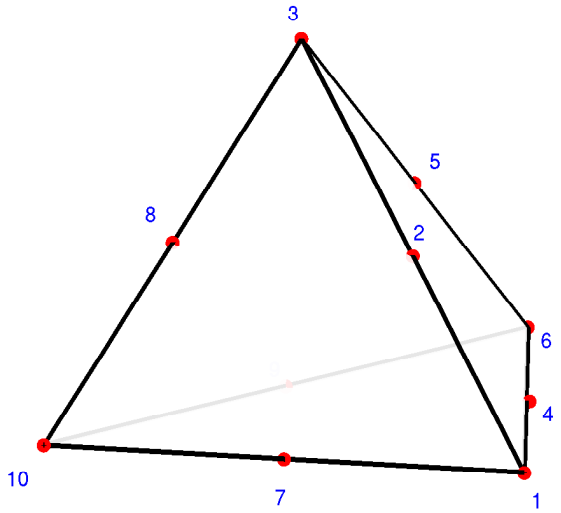

Fig. A.3: Numbered DOFs for quadratic Lagrange elements in three dimensions (Node 9 is located at the midpoint of the back edge).

calculate the transformation matrix B_K (equation (2.31)), it is not only important to know the DOFs of a tetrahedron but especially which of these nodes are the vertices of the element. Therefore, we introduce a variable called `points` which contains the local indices of all vertices in a consistent numbering. Considering the figures the variable points is given by `points = [3; 6; 1]` for triangles and `points = [3; 6; 1; 10]` for tetrahedra.

For the generation of new meshes, which are not implemented in our mesh functions `fe_create_mesh` and `fe_load_mesh`, it is recommended to adapt the local numbering of the DOFs to the one described above or to adjust the variable `points` in the following m-files:

- `fe_assemble_stiffness.m`
- `fe_get_stiffness_robin.m`
- `fe_get_rhs.m`
- `fe_get_rhs_bc.m`
- `fe_assemble_observation.m`
- `gaussNewtonR.m`.

A.2 Nédélec elements – additional information

As for linear elements, we refer to Whitney (1957) in the quadratic case on how to determine a basis of R_2 on the reference element and use barycentric coordinates to define 20 basis functions:

$$\begin{aligned}
\hat{\boldsymbol{\phi}}_1 &= \lambda_1\boldsymbol{\theta}_{12}, & \hat{\boldsymbol{\phi}}_2 &= \lambda_2\boldsymbol{\theta}_{12}, & \hat{\boldsymbol{\phi}}_3 &= \lambda_1\boldsymbol{\theta}_{13}, & \hat{\boldsymbol{\phi}}_4 &= \lambda_3\boldsymbol{\theta}_{13},\\
\hat{\boldsymbol{\phi}}_5 &= \lambda_1\boldsymbol{\theta}_{14}, & \hat{\boldsymbol{\phi}}_6 &= \lambda_4\boldsymbol{\theta}_{14}, & \hat{\boldsymbol{\phi}}_7 &= \lambda_2\boldsymbol{\theta}_{23}, & \hat{\boldsymbol{\phi}}_8 &= \lambda_3\boldsymbol{\theta}_{23},\\
\hat{\boldsymbol{\phi}}_9 &= \lambda_2\boldsymbol{\theta}_{24}, & \hat{\boldsymbol{\phi}}_{10} &= \lambda_4\boldsymbol{\theta}_{24}, & \hat{\boldsymbol{\phi}}_{11} &= \lambda_3\boldsymbol{\theta}_{34}, & \hat{\boldsymbol{\phi}}_{12} &= \lambda_4\boldsymbol{\theta}_{34},\\
\hat{\boldsymbol{\phi}}_{13} &= \lambda_3\boldsymbol{\theta}_{12}, & \hat{\boldsymbol{\phi}}_{14} &= \lambda_2\boldsymbol{\theta}_{13}, & \hat{\boldsymbol{\phi}}_{15} &= \lambda_4\boldsymbol{\theta}_{12}, & \hat{\boldsymbol{\phi}}_{16} &= \lambda_2\boldsymbol{\theta}_{14},\\
\hat{\boldsymbol{\phi}}_{17} &= \lambda_4\boldsymbol{\theta}_{13}, & \hat{\boldsymbol{\phi}}_{18} &= \lambda_3\boldsymbol{\theta}_{14}, & \hat{\boldsymbol{\phi}}_{19} &= \lambda_4\boldsymbol{\theta}_{23}, & \hat{\boldsymbol{\phi}}_{20} &= \lambda_3\boldsymbol{\theta}_{24},
\end{aligned}$$

where the basis functions $\lambda_i\boldsymbol{\theta}_{ij}$ and $\lambda_j\boldsymbol{\theta}_{ij}$ are associated with the edge defined by the vertices $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$. On this edge, the edge moments of these two functions are equal to one and they vanish on the other five edges. The basis functions $\lambda_k\boldsymbol{\theta}_{ij}$ with $k \neq i, j$ are associated with the face defined by the vertices $\boldsymbol{v}_i$, $\boldsymbol{v}_j$ and $\boldsymbol{v}_k$. This definition leads to twelve basis functions belonging to the tetrahedron's edges:

$$\boldsymbol{\phi}_1 = \begin{bmatrix} -xy \\ x^2 \\ 0 \end{bmatrix}, \quad \boldsymbol{\phi}_2 = \begin{bmatrix} -y^2 \\ xy \\ 0 \end{bmatrix}, \quad \boldsymbol{\phi}_3 = \begin{bmatrix} -xz \\ 0 \\ x^2 \end{bmatrix}, \quad \boldsymbol{\phi}_4 = \begin{bmatrix} -z^2 \\ 0 \\ xz \end{bmatrix}, \quad \boldsymbol{\phi}_5 = \begin{bmatrix} -x + xy + xz \\ -x^2 \\ -x^2 \end{bmatrix},$$

$$\boldsymbol{\phi}_6 = \begin{bmatrix} (-1+y+z)(1-x-y-z) \\ -x+x^2+xy+xz \\ -x+x^2+xy+xz \end{bmatrix}, \quad \boldsymbol{\phi}_7 = \begin{bmatrix} 0 \\ -yz \\ y^2 \end{bmatrix}, \quad \boldsymbol{\phi}_8 = \begin{bmatrix} 0 \\ -z^2 \\ yz \end{bmatrix},$$

$$\boldsymbol{\phi}_9 = \begin{bmatrix} -y^2 \\ y(x+z-1) \\ -y^2 \end{bmatrix}, \quad \boldsymbol{\phi}_{10} = \begin{bmatrix} -y(1-x-y-z) \\ (x+z-1)(1-x-y-z) \\ -y(1-x-y-z) \end{bmatrix},$$

$$\boldsymbol{\phi}_{11} = \begin{bmatrix} -z^2 \\ -z^2 \\ xz+yz-z \end{bmatrix}, \quad \boldsymbol{\phi}_{12} = \begin{bmatrix} -z+xz+yz+z^2 \\ -z+xz+yz+z^2 \\ (x+y-1)(1-x-y-z) \end{bmatrix}.$$

The last eight basis functions correspond to the faces of the tetrahedron:

$$\boldsymbol{\phi}_{13} = \begin{bmatrix} -yz \\ xz \\ 0 \end{bmatrix}, \quad \boldsymbol{\phi}_{14} = \begin{bmatrix} -yz \\ 0 \\ xy \end{bmatrix}, \quad \boldsymbol{\phi}_{15} = \begin{bmatrix} -y+xy+yz+y^2 \\ x-xy-xz-x^2 \\ 0 \end{bmatrix}, \quad \boldsymbol{\phi}_{16} = \begin{bmatrix} -y+yz+y^2 \\ -xy \\ -xy \end{bmatrix},$$

$$\boldsymbol{\phi}_{17} = \begin{bmatrix} -z+xz+yz+z^2 \\ 0 \\ x-xy-xz-x^2 \end{bmatrix}, \quad \boldsymbol{\phi}_{18} = \begin{bmatrix} -z+yz+z^2 \\ -xz \\ -xz \end{bmatrix},$$

$$\boldsymbol{\phi}_{19} = \begin{bmatrix} 0 \\ -z+xz+yz+z^2 \\ y-xy-yz-y^2 \end{bmatrix}, \quad \boldsymbol{\phi}_{20} = \begin{bmatrix} -yz \\ -z+xz+z^2 \\ -yz \end{bmatrix}.$$

In contrast to first-order elements, the face and edge moments for elements of order $k = 2$ with the associated polynomial space R_2 are defined as follows:

$$M_e(\hat{\boldsymbol{u}}) := \left\{ \int_{\hat{e}} \hat{\boldsymbol{u}} \cdot \hat{\boldsymbol{\tau}} \hat{q} \, d\hat{s}, \quad \text{for all } \hat{q} \in P_1(\hat{e}) \text{ for each edge } \hat{e} \text{ of } \hat{K} \right\},$$

$$M_f(\hat{\boldsymbol{u}}) := \left\{ \frac{1}{|\hat{f}|} \int_{\hat{f}} \hat{\boldsymbol{u}} \cdot \hat{\boldsymbol{q}} \, d\hat{A}, \quad \text{for each face } \hat{f} \text{ of } \hat{K}, \hat{\boldsymbol{q}} \in \mathbb{R}^3, \hat{\boldsymbol{q}} \cdot \hat{\boldsymbol{n}} = 0 \right\}.$$

Again, we have to find a set of bases of the polynomial spaces to determine the moments for quadratic elements. For the face moments we choose two constant vector test polynomials $\hat{\boldsymbol{q}}_1$ and $\hat{\boldsymbol{q}}_2$. An appropriate choice are the non-normalized edge vectors, spanning the considered face according to the definition of the faces in equation (3.23). For the DOFs on the edges of an element, which are called edge moments, we have to find a basis for the scalar polynomial space $P_1(\hat{e})$. Consider an arbitrary edge of the reference element $\hat{K}$ (equations (3.22) to (3.24)) defined by two vertices $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$. The parametrization of the edge is given by

$$\boldsymbol{\xi}(s) = \hat{\boldsymbol{v}}_i + s(\hat{\boldsymbol{v}}_j - \hat{\boldsymbol{v}}_i), \qquad s \in [0,1].$$

In terms of the curve parameter s, the barycentric coordinates associated with the two vertices read as follows:

$$\boldsymbol{v}_i: \quad \lambda_i(s) = 1 - s \tag{A.1}$$

$$\boldsymbol{v}_j: \quad \lambda_j(s) = s \tag{A.2}$$

and each coordinate is equal to one on the associated vertex and equal to zero on the other vertex. The edge basis functions are defined by their Whitney forms

$$\boldsymbol{\theta}_{ij} := \lambda_i \nabla\lambda_j - \lambda_j \nabla\lambda_i, \qquad 1 \le i < j \le 6,$$

with the pair $\lambda_i\boldsymbol{\theta}_{ij}$ and $\lambda_j\boldsymbol{\theta}_{ij}$. If we evaluate the edge moments for the basis functions ϕ, there should be only one non-vanishing moment:

$$\int_{\hat{e}} (\lambda_i \boldsymbol{\theta}_{ij} \cdot \hat{\boldsymbol{\tau}})\, \hat{q}_1(s)\, ds = 1, \qquad \int_{\hat{e}} (\lambda_j \boldsymbol{\theta}_{ij} \cdot \hat{\boldsymbol{\tau}})\, \hat{q}_1(s)\, ds = 0,$$

$$\int_{\hat{e}} (\lambda_i \boldsymbol{\theta}_{ij} \cdot \hat{\boldsymbol{\tau}})\, \hat{q}_2(s)\, ds = 0, \qquad \int_{\hat{e}} (\lambda_j \boldsymbol{\theta}_{ij} \cdot \hat{\boldsymbol{\tau}})\, \hat{q}_2(s)\, ds = 1.$$

$\boldsymbol{\theta}_{ij}$ represents the linear basis functions $\hat{\phi}_i (i = 1, \dots 6)$ given in equation (3.27) and the product $\boldsymbol{\theta}_{ij} \cdot \boldsymbol{\tau}$ is equal to the reciprocal edge length:

$$\hat{\boldsymbol{\phi}}_1 \cdot \hat{\boldsymbol{\tau}}_1|_{\hat{e}_1} \equiv \frac{1}{\sqrt{2}}, \qquad \hat{\boldsymbol{\phi}}_2 \cdot \hat{\boldsymbol{\tau}}_2|_{\hat{e}_2} \equiv \frac{1}{\sqrt{2}}, \qquad \hat{\boldsymbol{\phi}}_3 \cdot \hat{\boldsymbol{\tau}}_3|_{\hat{e}_3} \equiv 1,$$

$$\hat{\boldsymbol{\phi}}_4 \cdot \hat{\boldsymbol{\tau}}_4|_{\hat{e}_4} \equiv \frac{1}{\sqrt{2}}, \qquad \hat{\boldsymbol{\phi}}_5 \cdot \hat{\boldsymbol{\tau}}_5|_{\hat{e}_5} \equiv 1, \qquad \hat{\boldsymbol{\phi}}_6 \cdot \hat{\boldsymbol{\tau}}_6|_{\hat{e}_6} \equiv 1.$$

These factors cancel with the differential of the curve parametrization which results in the simplified integrals

$$\int_{\hat{e}} \lambda_i\, \hat{q}_1(s)\, ds = 1, \qquad \int_{\hat{e}} \lambda_j\, \hat{q}_1(s)\, ds = 0,$$

$$\int_{\hat{e}} \lambda_i\, \hat{q}_2(s)\, ds = 0, \qquad \int_{\hat{e}} \lambda_j\, \hat{q}_2(s)\, ds = 1.$$

In the following, we show the detailed derivation of the coefficients a_1 and b_1 for an arbitrary linear polynomial $\hat{q}_1(s) = a_1 + b_1 s$. Inserting these polynomials and the barycentric coordinates given in equations (A.1) and (A.2) the integrals can be evaluated as follows:

$$\int_0^1 (1-s)\cdot(a_1+b_1 s)\,ds = 1, \qquad \int_0^1 s\cdot(a_1+b_1 s)\,ds = 0,$$

$$\int_0^1 a_1 + (b_1-a_1)s - b_1 s^2\,ds = 1, \qquad \int_0^1 a_1 s + b_1 s^2\,ds = 0,$$

$$\left[a_1 s + \frac{(b_1-a_1)}{2}s^2 - \frac{b_1}{3}s^3\right]_0^1 = 1, \qquad \left[\frac{a_1}{2}s^2 + \frac{b_1}{3}s^3\right]_0^1 = 0,$$

$$a_1 + \frac{(b_1-a_1)}{2} - \frac{b_1}{3} = 1, \qquad \frac{a_1}{2} + \frac{b_1}{3} = 0,$$

$$a_1 = 2 - \frac{b_1}{3}, \qquad b_1 = -\frac{3a_1}{2}.$$

Inserting one of these two equations into the other finally yields the coefficients $a_1 = 4$ and $b_1 = -6$ and the first polynomial $\hat{q}_1(s) = 4 - 6s$. The second polynomial $\hat{q}_2(s) = -2 + 6s$ can be derived in the same way.

A.3 Differentiation of vector-valued functions

For a given function $g : \mathbb{R}^d \to \mathbb{R}$ and the vectors $\boldsymbol{x}, \boldsymbol{x}_0 \in \mathbb{R}^d$, the gradient of g has the following properties:

- For $(\gamma \in \mathbb{R}) : \nabla(\gamma g)(\boldsymbol{x}_0) = \gamma \nabla(g)(\boldsymbol{x}_0)$
- $\nabla(g+h)(\boldsymbol{x}_0) = \nabla g(\boldsymbol{x}_0) + \nabla h(\boldsymbol{x}_0)$
- For $(\gamma \in \mathbb{R}) : g(\boldsymbol{x}) = \gamma \Rightarrow \nabla g(\boldsymbol{x}_0) = \boldsymbol{0} \in \mathbb{R}^d$
- For $\boldsymbol{b} \in \mathbb{R}^d : g(\boldsymbol{x}) = \boldsymbol{b}^T\boldsymbol{x} = \boldsymbol{x}^T\boldsymbol{b} \Rightarrow \nabla g(\boldsymbol{x}_0) = \boldsymbol{b}$
- For $\boldsymbol{c} \in \mathbb{C}^d : g(\boldsymbol{x}) = \boldsymbol{c}^H\boldsymbol{x} \Rightarrow \nabla g(\boldsymbol{x}_0) = \overline{\boldsymbol{c}}$
- For $\boldsymbol{c} \in \mathbb{C}^d : g(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{c} \Rightarrow \nabla g(\boldsymbol{x}_0) = \boldsymbol{c}$
- $g(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{x} \Rightarrow \nabla g(\boldsymbol{x}_0) = 2\boldsymbol{x}_0$
- For $A \in \mathbb{R}^{d\times d} : g(\boldsymbol{x}) = \boldsymbol{x}^T A\boldsymbol{x} \Rightarrow \nabla g(\boldsymbol{x}_0) = A\boldsymbol{x}_0 + A^T\boldsymbol{x}_0$

- For $A \in \mathbb{C}^{d\times d}$: $g(\boldsymbol{x}) = \boldsymbol{x}^T A\boldsymbol{x} \Rightarrow \nabla g(\boldsymbol{x}_0) = A\boldsymbol{x}_0 + \overline{A}\boldsymbol{x}_0 = 2\mathsf{Re}\,(A)\boldsymbol{x}_0$

- For $A \in \mathbb{R}^{m\times d}$ and $\boldsymbol{b} \in \mathbb{R}^m$: $g(\boldsymbol{x}) = ||\boldsymbol{b} - A\boldsymbol{x}||^2$
  $g(\boldsymbol{x}) = (\boldsymbol{b} - A\boldsymbol{x})^T(\boldsymbol{b} - A\boldsymbol{x}) = \boldsymbol{b}^T\boldsymbol{b} - \underbrace{\boldsymbol{b}^T A}_{\boldsymbol{c}^T}\boldsymbol{x} - \boldsymbol{x}^T\underbrace{A^T\boldsymbol{b}}_{\boldsymbol{c}} + \boldsymbol{x}^T A^T A\boldsymbol{x}$
  $\nabla g(\boldsymbol{x}_0) = \boldsymbol{0} - A^T\boldsymbol{b} - A^T\boldsymbol{b} + A^T A\boldsymbol{x}_0 + A^T A\boldsymbol{x}_0 = 2A^T A\boldsymbol{x}_0 - 2A^T\boldsymbol{b}$
  $\nabla g(\boldsymbol{x}_0) = 2A^T(A\boldsymbol{x} - \boldsymbol{b})$

- For $A \in \mathbb{C}^{m\times d}$ and $\boldsymbol{b} \in \mathbb{C}^m$: $g(\boldsymbol{x}) = ||\boldsymbol{b} - A\boldsymbol{x}||^2$
  $g(\boldsymbol{x}) = (\boldsymbol{b} - A\boldsymbol{x})^H(\boldsymbol{b} - A\boldsymbol{x}) = \boldsymbol{b}^H\boldsymbol{b} - \underbrace{\boldsymbol{b}^H A}_{\boldsymbol{c}^T}\boldsymbol{x} - \boldsymbol{x}^T\underbrace{A^H\boldsymbol{b}}_{\boldsymbol{c}} + \boldsymbol{x}^T A^H A\boldsymbol{x}$
  $\nabla g(\boldsymbol{x}_0) = \boldsymbol{0} - A^T\overline{\boldsymbol{b}} - A^H\boldsymbol{b} + A^H A\boldsymbol{x}_0 + A^T\overline{A}\boldsymbol{x}_0 = -\overline{A^H\boldsymbol{b}} - A^H\boldsymbol{b} + A^H A\boldsymbol{x}_0 + \overline{A^H A}\boldsymbol{x}_0$
  $\nabla g(\boldsymbol{x}_0) = 2\,\mathsf{Re}(A^H A)\boldsymbol{x}_0 - 2\,\mathsf{Re}(A^H\boldsymbol{b}) = 2\,\mathsf{Re}(A^H(A\boldsymbol{x}_0 - \boldsymbol{b})).$

A.4 Plane waves in horizontally layered conductivity structures

As given in section 3.2.1, we calculate appropriate boundary values for the propagation of a plane wave in a layered halfspace with the function `getE1dMT` according to Wait's algorithm (Wait, 1953).

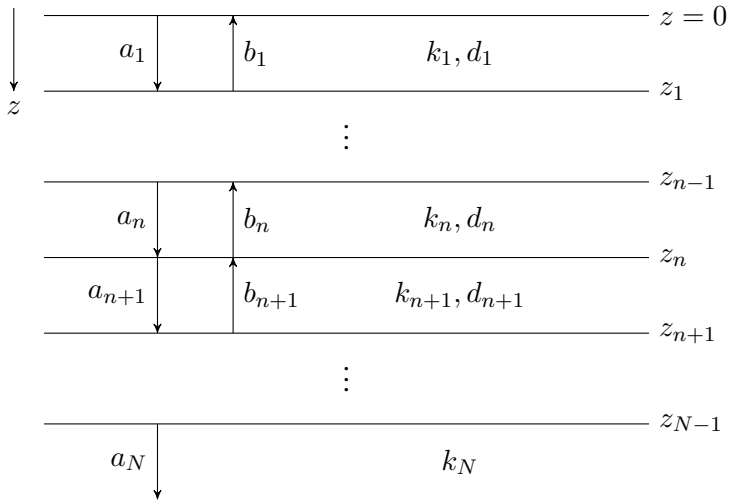

Fig. A.4: The layered halfspace and corresponding parameters.

Here, we refer to Ward and Hohmann (1988) and Schwarzbach (2009) and consider a layered halfspace consisting of $n = 1, \ldots, N$ layers with the corresponding physical parameters σ_n and μ, the complex wave number k_n with $k_n^2 = -i\omega\mu\sigma_n$ and the layer thickness d_n belonging to the nth layer between z_{n-1} and z_n (Figure A.4). In this layer, the x-component of the electric field and the y-component of the magnetic field can be described according to

Faraday's law (3.5):

$$E_x^n(z) = Z_n \left(a_n e^{-ik_n(z-z_n)} + b_n e^{ik_n(z-z_n)} \right) \tag{A.3}$$

$$H_y^n(z) = -\frac{1}{i\omega\mu} \frac{\partial E_x^n(z)}{\partial z} = a_n e^{-ik_n(z-z_n)} - b_n e^{ik_n(z-z_n)}, \tag{A.4}$$

with the amplitudes a_n and b_n of the wave propagating in positive or negative z-direction, respectively and the intrinsic impedance $Z_n = \frac{\omega\mu}{k_n}$ of the nth layer (Ward and Hohmann, 1988). Because the layer below z_{N-1} is a halfspace which extends to infinity, there is no reflected wave traveling upwards and $b_n \equiv 0$ for $z \geq z_{N-1}$.

We describe the electromagnetic impedance in section 3.3.2.2 and especially in equation (3.40). According to that, we calculate the apparent impedance $\hat{Z}_n$ in the nth layer as follows:

$$\hat{Z}_n = \frac{E_x^n(z)}{H_y^n(z)} = Z_n \frac{a_n e^{-ik_n(z-z_n)} + b_n e^{ik_n(z-z_n)}}{a_n e^{-ik_n(z-z_n)} - b_n e^{ik_n(z-z_n)}}. \tag{A.5}$$

The impedance of the lowermost infinite halfspace is then given by $Z_N = \frac{\omega\mu}{k_N}$. In the following, we want to determine the impedance $\hat{Z}_n$ from the impedance $\hat{Z}_{n+1}$ of the layer which lies beneath the nth layer. At $z = z_n$ equation (A.5) yields

$$\hat{Z}_n = \frac{E_x^n(z_n)}{H_y^n(z_n)} = Z_n \frac{a_n + b_n}{a_n - b_n} = Z_n \frac{1 + r_n}{1 - r_n}. \tag{A.6}$$

The reflection coefficient

$$r_n = \frac{b_n}{a_n}$$

describes the ratio of the amplitudes of the up- and downward traveling wave at the layer interface $z = z_n$. Similarly, we can derive the impedance $\hat{Z}_{n-1}$ on top of the nth layer as follows: At $z = z_{n-1}$, the continuity of the tangential components of the electric and magnetic field yields

$$E_x^{n-1} = E_x^n$$

$$H_y^{n-1} = H_y^n$$

and therefore, the apparent impedance $\hat{Z}_{n-1}$ along $z = z_{n-1}$ can be calculated according to equation (A.5):

$$\hat{Z}_{n-1} = Z_n \frac{a_n e^{-ik_n(z_{n-1}-z_n)} + b_n e^{ik_n(z_{n-1}-z_n)}}{a_n e^{-ik_n(z_{n-1}-z_n)} - b_n e^{ik_n(z_{n-1}-z_n)}}$$

$$\hat{Z}_{n-1} = Z_n \frac{1 + r_n e^{-2ik_n d_n}}{1 - r_n e^{-2ik_n d_n}},$$

with the layer thickness $d_n = z_n - z_{n-1}$. Furthermore, we introduce the coefficient $\alpha = ik_n d_n$ and rearrange this equation to calculate the reflection coefficients r_n:

$$\begin{aligned}
\hat{Z}_{n-1}\left(1 - r_n e^{-2\alpha}\right) &= Z_n\left(1 + r_n e^{-2\alpha}\right) \\
\left(Z_n + \hat{Z}_{n-1}\right) e^{-2\alpha} r_n &= \hat{Z}_{n-1} - Z_n \\
r_n &= \frac{\hat{Z}_{n-1} - Z_n}{\hat{Z}_{n-1} + Z_n} e^{2\alpha}. \qquad \text{(A.7)}
\end{aligned}$$

We insert this equation into (A.6), expand the fraction with $e^{-\alpha}$ and get

$$\begin{aligned}
\hat{Z}_n &= Z_n \frac{1 + \frac{\hat{Z}_{n-1} - Z_n}{\hat{Z}_{n-1} + Z_n} e^{2\alpha}}{1 - \frac{\hat{Z}_{n-1} - Z_n}{\hat{Z}_{n-1} + Z_n} e^{2\alpha}} \\
\hat{Z}_n &= Z_n \frac{-Z_n(e^{\alpha} - e^{-\alpha}) + \hat{Z}_{n-1}(e^{\alpha} + e^{-\alpha})}{Z_n(e^{\alpha} + e^{-\alpha}) - \hat{Z}_{n-1}(e^{\alpha} - e^{-\alpha})}.
\end{aligned}$$

The hyperbolic functions $\sinh(x) = \frac{e^x - e^{-x}}{2}$, $\cosh(x) = \frac{e^x + e^{-x}}{2}$ and $\tanh(x) = \frac{\sinh(x)}{\cosh(x)}$ yield

$$\begin{aligned}
\hat{Z}_n &= Z_n \frac{-Z_n \sinh(\alpha) + \hat{Z}_{n-1} \cosh(\alpha)}{Z_n \cosh(\alpha) - \hat{Z}_{n-1} \sinh(\alpha)} \\
\hat{Z}_n &= Z_n \frac{\hat{Z}_{n-1} - Z_n \tanh(\alpha)}{Z_n - \hat{Z}_{n-1} \tanh(\alpha)}.
\end{aligned}$$

We rearrange this equation to obtain a recursion formula for $\hat{Z}_{n-1}$:

$$\begin{aligned}
\hat{Z}_n\left(Z_n - \hat{Z}_{n-1}\tanh(\alpha)\right) &= Z_n\left(\hat{Z}_{n-1} - Z_n \tanh(\alpha)\right) \\
Z_n\left(\hat{Z}_n - Z_n \tanh(\alpha)\right) &= \hat{Z}_{n-1}\left(Z_n + \hat{Z}_n \tanh(\alpha)\right) \\
\hat{Z}_{n-1} &= Z_n \frac{\hat{Z}_n + Z_n \tanh(\alpha)}{Z_n + \hat{Z}_n \tanh(\alpha)}.
\end{aligned}$$

Now, the impedances can be calculated recursively, starting from the halfspace impedance $Z_N = \frac{\omega\mu}{k_N}$ to finally get the reflection coefficients in equation (A.7). To calculate the electric and magnetic field values as given in equations (A.3) and (A.4) we need to know at least the amplitude a_n. We apply the continuity condition for the electric field at $z = z_n$ to calculate a_{n+1} from the previous amplitude a_n and the corresponding reflection coefficients:

$$\begin{aligned}
E_x^n(z = z_n) &= E_x^{n+1}(z = z_n) \\
a_n + b_n &= a_{n+1} e^{ik_{n+1}d_{n+1}} + b_{n+1} e^{-ik_{n+1}d_{n+1}} \\
a_n(1 + r_n) &= a_{n+1}(e^{ik_{n+1}d_{n+1}} + r_{n+1} e^{-ik_{n+1}d_{n+1}}) \\
a_{n+1} &= \frac{a_n(1 + r_n)}{(e^{ik_{n+1}d_{n+1}} + r_{n+1} e^{-ik_{n+1}d_{n+1}})},
\end{aligned}$$

with $d_{n+1} = z_{n+1} - z_n$. For an incident electric field $E_0 = 1\,\text{V/m}$, there holds $a_0 = 1$ and

$$a_1 = \frac{(1 + r_0)}{(e^{ik_1 d_1} + r_1 e^{-ik_1 d_1})}.$$

We are now able to compute the electric and magnetic field components in each layer.

A.5 Architecture of the programs

A.5.1 Program structure of the DC resistivity code

A.5.1.1 Overview

Chapter 2 explained the algorithms and mathematical principles implemented in the DC resistivity code. Within this section we want to show its structure, how to use it for the inversion of DC resistivity data and which function accesses which subroutines. The script `DCinversion.m`

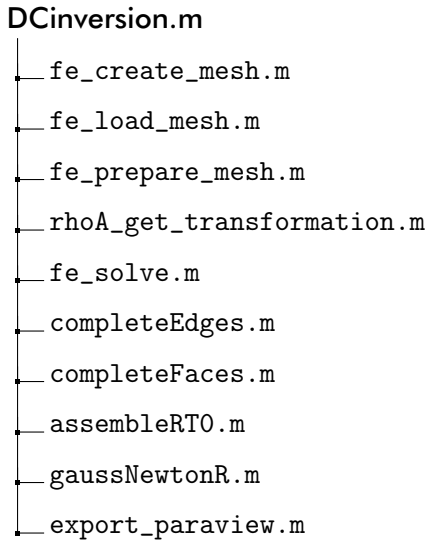

Fig. A.5: Directory tree of the DC resistivity code.

is the driver file for a DC resistivity inversion. Here, we have to set the main variables, define the used synthetic model or include real data. The directory tree in Figure A.5 shows the setup including all relevant m-files.

The first four variables in the driver determine the dimension of the model problem, the degree of the Lagrange elements, the used boundary condition, the use of secondary or total field approach and decide whether we perform an individual or joint inversion. After that, the following tasks are performed:

- Generation of forward and inversion mesh and assembly of system matrices,
- Definition of sources and receivers,
- Generation of a synthetic data set,
- Assembly of regularization matrices,

- Definition of reference and starting model,
- Application of a Gauss-Newton scheme,
- Plotting and saving result.

A.5.1.2 Mesh generation and assembly of system matrices

The directory tree A.6 shows a part of the full program which is used to generate or load meshes and assemble the associated system matrices as well as the measurement operator. Within `DCinversion.m` we can choose the model we want to use by setting the parameters `model_name` and `invmodel_name` as well as the associated sources and receivers.

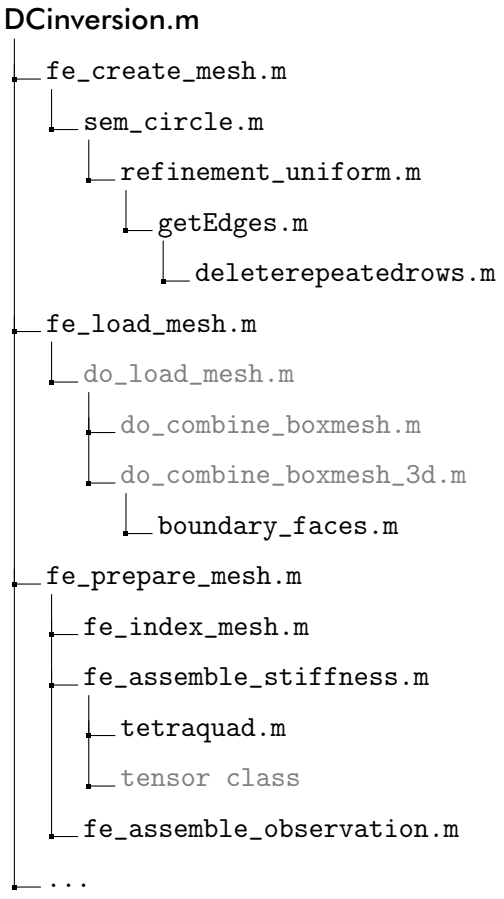

Fig. A.6: Directory tree for mesh generation and assembly.

The function `fe_create_mesh` provides different models in two and three dimensions. When using this function, a new mesh is generated from scratch using the parameter `hmax` given in `DCinversion.m` to determine the maximum element diameter. To use these models we have to start MATLAB together with COMSOL because all implemented models are based

on COMSOL commands. In contrast, the function `fe_load_mesh` loads static meshes. The element size cannot be changed for these. The subroutines and local functions are used to get boundary nodes and faces. Furthermore, we have to add certain features to the mesh to enable its usage within forward calculations. Within `fe_prepare_mesh.m` we project the initial conductivity vector `m_0`, which contains unique conductivity values of the chosen model, onto all tetrahedra using the mesh property `mesh.regio`:

$$\texttt{m0} = \texttt{m0}(\texttt{mesh.regio}).$$

We separately run `fe_prepare_mesh` for the forward and inversion mesh. After generating or loading a mesh, the structure array `mesh` contains all information about the discretized modeling area. Its properties are

- `tetra/tri`: matrix containing the mapping between DOFs and elements, `tetra`: $4 \times n_{\text{elements}}$ (2D, `tri`: $3 \times n_{\text{elements}}$),
- `nodes`: matrix containing the coordinates of all vertices (all DOFs for quadratic elements), $3 \times n_{\text{vtx}}$ (2D: $2 \times n_{\text{vtx}}$),
- `regio`: vector of indices containing the mapping between elements and conductivities, $1 \times n_{\text{elements}}$,
- `diric_dofs`: vector of DOFs belonging to the Dirichlet boundary, empty matrix if there is no Dirichlet boundary, $n_{\text{Diric. DOFs}} \times 1$,
- `robin_bc`: matrix containing boundary objects (faces in 3D, edges in 2D) belonging to the Robin boundary, $3 \times n_{\text{Robin faces}}$ (2D: $2 \times n_{\text{Robin edges}}$), empty matrix if there is no Robin boundary,
- `neum`: matrix containing boundary objects belonging to the Neumann boundary, empty matrix if there is no Neumann boundary, $3 \times n_{\text{Neumann faces}}$ (2D: $2 \times n_{\text{Neumann edges}}$),
- `degree`: degree of finite element basis functions (1 for linear or 2 for quadratic),
- `dim`: dimension of modeled problem,
- `I`: matrix to exclude Dirichlet nodes: $u_{\text{inner nodes}} = u(I)$,
- `E`: matrix to insert Dirichlet nodes: $u = E \cdot u_{\text{inner nodes}}$,

- `dofmap`: vector to map DOFs onto the grid nodes of the COMSOL mesh. In COMSOL, the global numbering of grid nodes and DOFs is not necessarily the same,

- `raw`: output of COMSOL, finite element structure array containing the initial COMSOL mesh structure.

All future meshes should contain these properties in order to guarantee compatibility with the code. The last two items are only required if the user creates a COMSOL mesh and wants to apply COMSOL functions, e. g. to plot the potential field using the command `postplot`.

After mesh generation, the mesh information can be used to assemble the system matrices. The function `fe_assemble_stiffness` is used to assemble a tensor which only contains geometric aspects in a three-way-tensor as given in Section 2.3.3. To get the full system or stiffness matrix we use (and explain) the function `fe_get_stiffness` later on and multiply each slice i of the tensor with the corresponding conductivity $\sigma_i = e^{m_i}$. We use the tensor class implemented by Martin Afanasjew.

Finally, we assemble the observation operator Q as described in Section 2.2.4 within `fe_assemble_observation.m`.

A.5.1.3 Forward calculation and generation of a synthetic data set

The function `fe_solve` (Figure A.7) realizes the assembly and solution of the forward problem which is the calculation of synthetic DC resistivity data. As described in the previous section we use the assembled tensor to build the stiffness matrix within `fe_get_stiffness.m` by multiplication of the slices with the corresponding conductivities. Furthermore, using the function `fe_get_stiffness_robin` we assemble a matrix `A_rob` which contains the Robin boundary condition and is just added to the system matrix before we solve the forward problem. We build up the right-hand sides for total or secondary field approach using the function `fe_get_rhs` in the first case and equation (2.27) in the second case and add the Robin condition to the right-hand side, too. If we choose to use a Dirichlet boundary condition, the calculation of the matrix `A_rob` is omitted. The choice between different boundary conditions is made at the beginning of `DCinversion.m` as described in Appendix A.5.1.1.

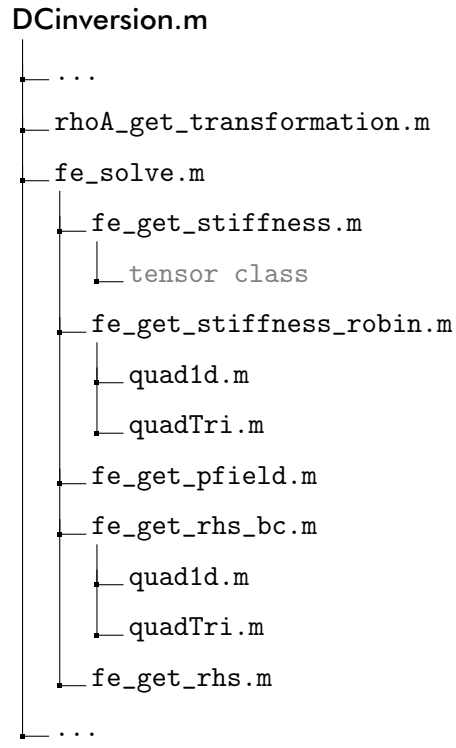

Fig. A.7: Directory tree for forward solution.

For the secondary field approach we have to set up the primary potential $\boldsymbol{u}_\mathrm{p}$ which is calculated within `fe_get_pfield.m` using the analytical solution for a homogeneous halfspace given in equations (2.4). After that, the linear system of equations is solved for the total or secondary potential using MATLAB's `mldivide` or a multigrid method (see Section 2.2.6). For the secondary field approach, we have to add the primary potential to the solution of the system to obtain the total potential. The resulting vector is the output parameter of `fe_solve`. In Section 2.1 and especially equation (2.1) we explained the connection between the electric potential u and the apparent resistivity ρ_a. The outputs of the forward modeling code are potentials, but real DC resistivity data are apparent resistivities. Hence, we have to convert the potentials using the geometric factor k as described in Section 2.2.4.2. The function `rhoA_get_transformation` implements this conversion and outputs a matrix `D_rho` which can be multiplied by the data or potential vector to obtain apparent resistivities.

A.5.1.4 Assembly of regularization matrices

The directory tree in Figure A.8 shows the functions which are used to assemble the regularization matrices M and D which have been described in Section 2.3.5. Because our regularization strategy is based on RT elements where the DOFs are located on edges in the 2D

case and on faces in 3D, we need to add the associated edges and faces to the given mesh information. This is realized by the functions `completeEdges` and `completeFaces` which use the given element table `mesh.tri` or `mesh.tetra` to define all edges or faces. The actual

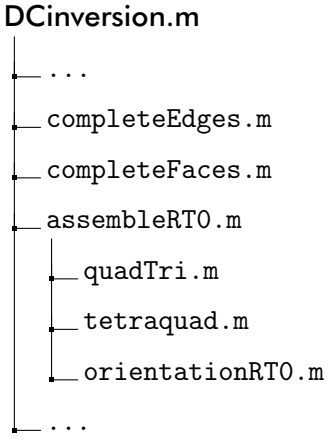

Fig. A.8: Directory tree for regularization operator.

assembly of the regularization matrices is implemented in `assembleRT0.m` and is quite similar to the assembly routine of the stiffness matrix. The additional function `orientationRT0` ensures the correct orientation of neighboring faces and edges (see Section 2.3.5 and especially Figure 2.16). We consider each face (or edge in 2D) twice and have to keep in mind that, because of the global numbering of the DOFs and the edges and faces, the direction of rotation is different between neighboring faces. Therefore, we have to change the sign of the outer basis function to get a conforming discretization.

A.5.1.5 The Gauss-Newton scheme

The actual inversion algorithm is implemented in `gaussNewtonR` and the directory tree in Figure A.9 shows the subroutines that are called inside the function. Here, we have to set the most important parameters concerning the inversion:

- `maxit`: maximum number of Gauss-Newton iterations,
- `tol`: termination tolerance for the absolute residual norm of the Gauss-Newton iteration,
- `solve_method`: solution method of least squares problem (MATLAB's `mldivide`, PARDISO, PCG or LSQR, see Section 2.3.2),

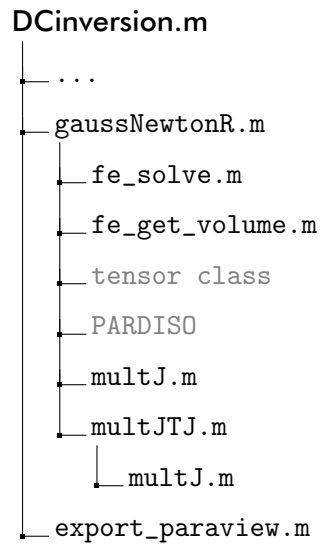

Fig. A.9: Directory tree for the Gauss-Newton scheme.

- `maxit_iter`: maximum number of iterations for iterative solver,
- `tol_iter`: relative target residual for iterative solver,
- `beta`: initial regularization parameter,
- `betamin`: lower bound for the regularization parameter.

We calculate all cell volumes within the function `fe_get_volume` in order to possibly add a weighted identity matrix to the regularization operator. The solution of the least squares problem is then executed in a while-loop, where we use a simple damping algorithm in order to control the step size of the update of the model parameter vector (see Section 2.3.6).

A.5.1.6 Some additional script files and functions

The script file `export_paraview.m` can be used to create a vtu-file to display the 3D inversion result in the vtk-based open-source software ParaView. It needs a mesh structure containing the array `mesh.tetra` of size $4 \times n_{\text{elements}}$ which contains the four nodes belonging to the ith tetrahedron in the ith column and the array `mesh.nodes` of size $3 \times n_{\text{vtx}}$ containing the coordinates of all vertices. Furthermore, the parameter vector has to be in the variable `m_k`. One can visualize just the final result or the development of the parameter distribution within the inversion algorithm. Then, the result of the ith Gauss-Newton step has to be in the ith column of the $n_{\text{parameters}} \times n_{\text{iteration steps + 1}}$ matrix `m_k`. Additionally, we can add e. g.

the starting or reference model for comparison using `d_cell.m_ref = log(m_ref)`. The export routine is used at the end of `DCinversion.m` to save and display the 3D result. In two dimensions, we display the parameter distribution using the MATLAB function `trisurf`.

The script file `plot_sounding_curve.m` can be used to plot analytical and modeled sounding curves for a pole-pole configuration and a model with two concentric semicircles in 2D and two concentric hemispheres in 3D. The profile line where the data is calculated is defined in the vector $\boldsymbol{x} = (x, y, z)$ and initially given by 50 logarithmically spaced points at the Earth's surface. Using the analytical solutions given in Section 2.2.6 we are able to validate our modeling results.

The function `inspect_mesh` is able to visualize boundary nodes and elements in order to check the correct setup of the tables describing the boundary of the mesh. The input is a mesh structure containing the properties `mesh.neum` for the boundary triangles (or edges in 2D) belonging to the Neumann boundary, `mesh.robin_bc` for the boundary triangles belonging to the Robin boundary and `mesh.diric_dofs` for all Dirichlet nodes.

A.5.2 Program structure of the electromagnetic code

A.5.2.1 Overview

Chapter 3 explained the mathematical and physical background of the electromagnetic modeling and inversion code. To simplify its usage, we give a short introduction to the structure and the main functions. The directory tree in Figure A.10 shows the setup of the driver file

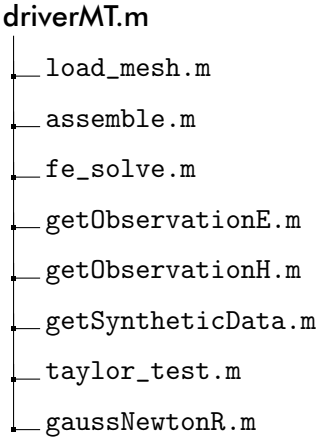

Fig. A.10: Directory tree of the electromagnetic code.

`driverMT.m` including all relevant m-files. The main parts are 1. the provision of a mesh and the related parameters, 2. the assembly of the system matrix, 3. the solution of the forward problem, 4. the assembly of the measurement operators, 5. the calculation of a synthetic data set 6. the test of the first derivatives and finally, 7. the inversion algorithm. In the first lines of `driverMT.m` we set up the input parameters which are for example the used geophysical method, the order of Nédélec elements, the frequency range and others. We can differentiate between one consistent mesh for observed synthetic data and the inversion algorithm or – which is the only choice to avoid the *inverse crime* – two different meshes. Furthermore, we define the profiles and receiver locations as well as the percentage of the noise.

A.5.2.2 Loading and preparing a mesh

We have a small directory `mt/mesh` containing some meshes for a homogeneous unit cube with and without air and for the 3D-2 COMMEMI model including different refinements. If one wants to add his own mesh, then they have to keep in mind the needed fields of the mesh structure (Figure A.11):

- the element array `tetToVtx` $\in \mathbb{R}^{4\times n_{\text{elements}}}$ which matches the tetrahedra with the associated global vertices,

- the point array `vtx` $\in \mathbb{R}^{3\times n_{\text{vertices}}}$ which contains the coordinates of all vertices,

- and the mapping array `regio` $\in \mathbb{R}^{1\times n_{\text{elements}}}$ which maps each element with the corresponding conductivity area.

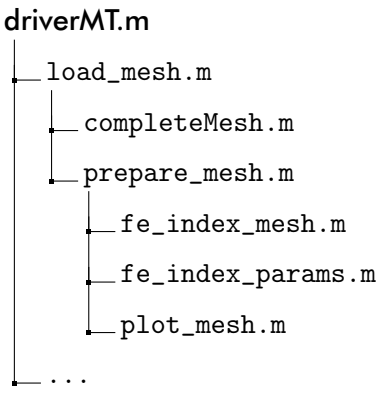

Fig. A.11: Directory tree for mesh preparation.

We have to apply the functions `completeMesh` and `prepare_mesh` to these tables to add further fields describing the edges, faces and boundaries of the mesh and to assemble matrices which are able to exclude Dirichlet nodes from the system matrix of the Galerkin system and the right-hand side or to exclude the air layer in the inversion scheme. Within `prepare_mesh` the conductivity values for the areas defined in the variable `mesh.regio` are set. The main properties of the structure `mesh` are described in Appendix A.5.1.2.

A.5.2.3 Assembly of system matrices

The directory tree A.12 shows the structure of the matrix assembly as described in Section 3.2.2. The matrices for the Galerkin system

$$(C + i\omega\mu_0 M(\boldsymbol{\sigma}))\boldsymbol{u} = \mathbf{0}$$

are assembled within the function `assemble`. First of all, we define quadrature nodes and calculate the Nédélec basis functions and their curls on these nodes. The stiffness matrix C is independent of the parameters. Therefore, we assemble this matrix once. On the other hand, the mass matrix depends on $\boldsymbol{\sigma}$. Therefore, we use the tensor class implemented by Martin Afanasjew to store the derivative of $M(\boldsymbol{\sigma})$ with respect to sigma and multiply it by the

parameter vector if we need the full matrix (see Section 3.2.2). The function `getAffineMap` yields the transformation matrix B_K and the vector $\boldsymbol{b}_K$ to transform between the reference element and an arbitrary tetrahedron.

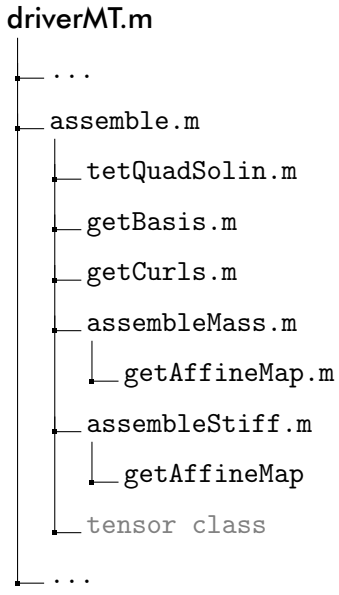

Fig. A.12: Directory tree for the assembly of matrices.

A.5.2.4 Forward problem

The solution of the forward problem is done within the function `fe_solve` as shown in Figure A.13. We hand over the current parameter vector and the tensor to build up the mass matrix and finally the full system matrix A. Before solving the system, we remove the DOFs belonging to Dirichlet boundaries. This is done by calculating the analytical moments of the Nédélec basis functions on the associated Dirichlet edges and faces. Finally, we are able to solve the forward problem and decide between two direct solvers which are MATLAB's `mldivide` or PARDISO (see Section 2.3.1). The output parameters are the full system matrix and the solution of the Galerkin system, both with and without Dirichlet values.

A.5.2.5 Measurement operators and synthetic data

The previous section described the calculation of the forward solution $\boldsymbol{u}$ which contains no physical field values in the case of Nédélec elements, but the DOFs or the moments on faces and edges, respectively. To calculate the electric or magnetic fields, especially on some selected receiver locations, we have to multiply these moments with the associated basis func-

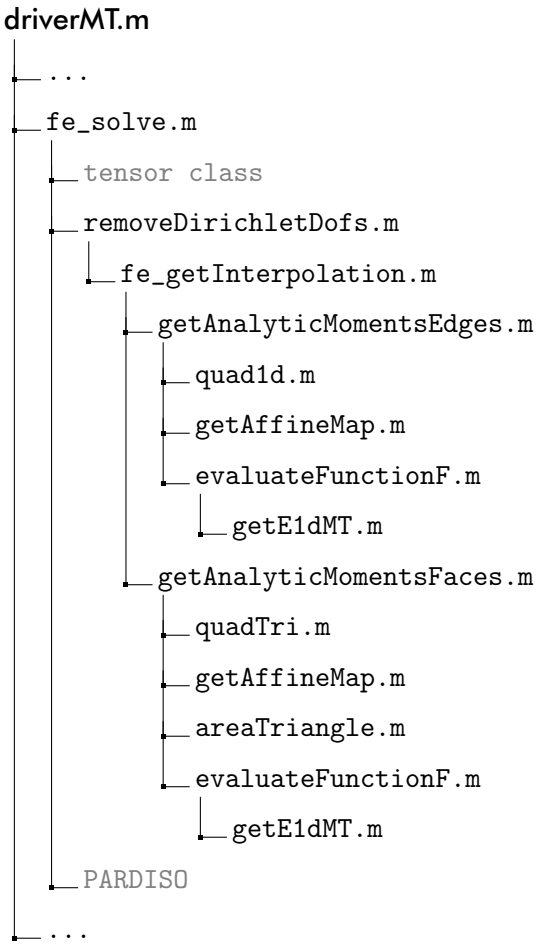

Fig. A.13: Directory tree for the solution of the forward problem.

tions or their curls for the magnetic field as described in Section 3.2.3 and Figure A.14. Within `driverMT.m` we set an input parameter `physics` to decide whether the measured data are for example electric field components, impedances or other. We have two different functions `getObservationE` and `getObservationH` to calculate the measurement operators Q_E and Q_H. Within the function for the magnetic field, we can decide between two methods which are the assembly of the curl of the electric field $\boldsymbol{E}$ or the integration over a small coil. We then hand over the measurement operators and the solution $\boldsymbol{u}$ to the function `getSyntheticData` to calculate the synthetic data set. Immediately after this function call, we are able to add some noise to these data within `driverMT.m`.

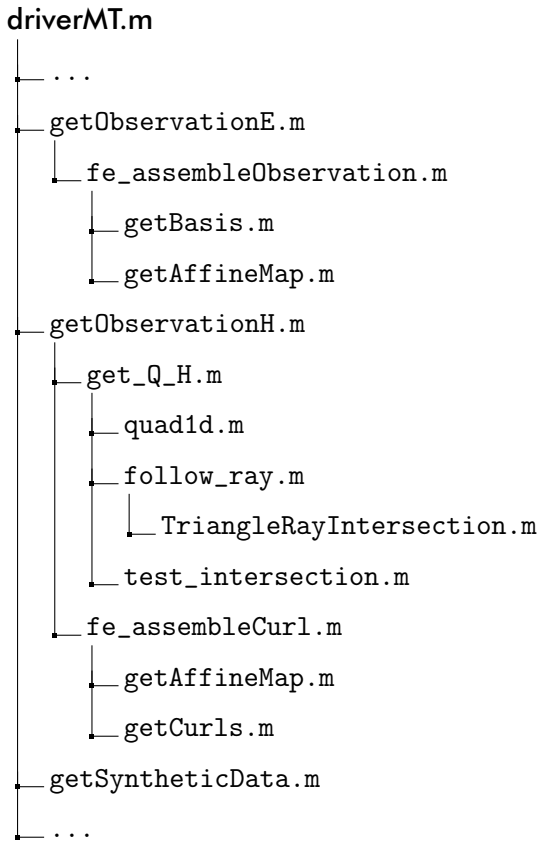

Fig. A.14: Directory tree for the assembly of the measurement operators.

A.5.2.6 Validation of the derivatives

The most important quantities within the forward and inversion algorithm are the system matrix A, which is built up from its derivative tensor dA, and the sensitivity or Jacobian matrix J which is the derivative of the data with respect to the parameters. Therefore, we ensure the correct assembly by doing a derivative test within the function given in Figure A.16. If we consider the taylor series

$$f(x) = f(x_0) + f'(x_0)\Delta x + \mathcal{O}(\Delta x^2) \quad \text{and} \quad \Delta x = x - x_0$$

then, the difference

$$f(x) - f(x_0) - f'(x_0)\Delta x$$

decays with $\mathcal{O}(\Delta x^2)$ and if we keep Δx constant and multiply it by a small scalar value h, the resulting difference converges to zero with h^2. Figure A.15 shows an exemplary test result for the system and sensitivity matrix for the 3D-2 COMMEMI model (see Section 3.2.5.3) with impedance data and a relatively coarse mesh of 3 687 elements and 24 516 DOFs for quadratic elements. We see the quadratic convergence of the derivatives and the linear convergence of the data norm.

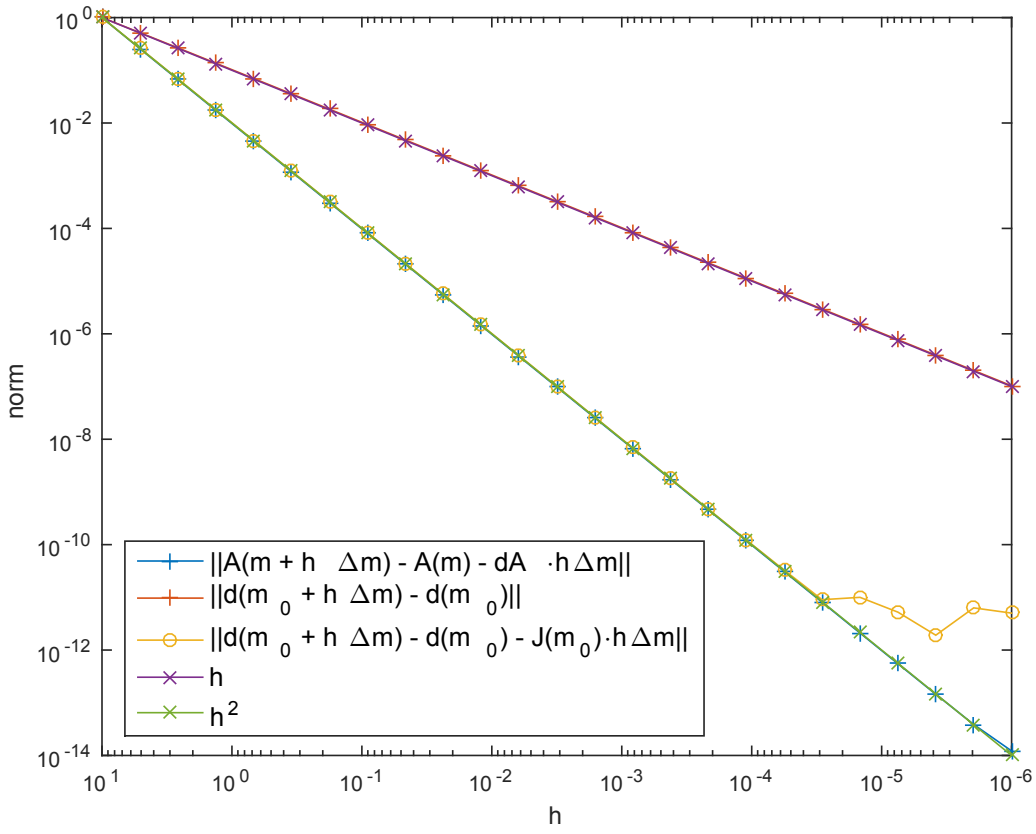

Fig. A.15: Taylor test for the derivative of the system matrix A and the sensitivity matrix.

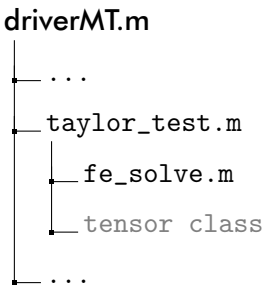

Fig. A.16: Directory tree for the taylor test.

A.5.2.7 The Gauss-Newton scheme

Finally, we show the structure of the inversion algorithm in Figure A.17. In the first lines of the function `gaussNewtonR` we set the input parameters, which are for example the maximum number of Gauss-Newton steps as well as the maximum number of steps and the tolerance for the iterative solver. Afterwards, the operator for the smoothness regularization (see Section

2.3.5) is assembled. We implemented a regularization matrix

$$W = \begin{bmatrix} k_1 W_{\mathrm{SR}} \\ k_2 I \end{bmatrix}$$

consisting of two parts: The smoothness regularization and the volume-weighted identity matrix. We can decide to use one of these matrices or both in combination with the weighting scalars k_1 and k_2. Within the Gauss-Newton iteration, we are able to solve the normal

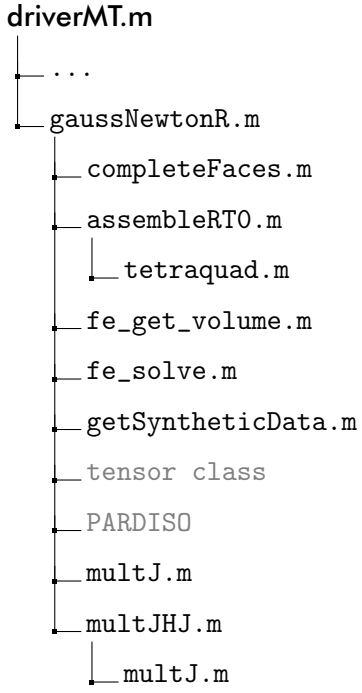

Fig. A.17: Directory tree for the Gauss-Newton scheme.

equations directly with MATLAB's `mldivide` or PARDISO. Furthermore, the iterative Krylov subspace method PCG (see Section 2.3.2) is implemented and uses an implicit calculation of the sensitivity matrix J. After solving the normal equations, we apply a simple damping algorithm as described in Section 2.3.6 and set the output parameters to be the inverted data set, the residual norm, the resulting parameter model and the number of Gauss-Newton steps.