IDEA League

Master of Science in Applied Geophysics

Research Thesis

Imaging of active layer characteristics through quasi-3D inversion of frequency-domain electromagnetic soundings

Marlene Thalhammer

August 5, 2022

Imaging of active layer characteristics through quasi-3D inversion of frequency-domain electromagnetic soundings

Master of Science Thesis

for the degree of Master of Science in Applied Geophysics

by

Marlene Thalhammer

August 5, 2022

IDEA LEAGUE
JOINT MASTER'S IN APPLIED GEOPHYSICS

TU Delft, The Netherlands
ETH Zurich, Switzerland
RWTH Aachen, Germany

Dated: *August 5, 2022*

Examiners:

Prof. Florian Wellmann

Dr. Dirk-Jan van Manen

Supervisors:

Dr. Florian Wagner

Dr. Sebastian Uhlemann

The Committee consists of examiners and supervisors.

Abstract

The active layer thickness has become an important indicator in climate change research as permafrost degradation has long been documented. The thawing of permafrost causes the release of greenhouse gases accelerating Arctic warming. Monitoring and quantifying spatial and temporal changes of the active layer are challenging but crucial for reliable climate projections. Geophysical methods offer a non-invasive investigation of electrical properties and their distribution in permafrost areas, revealing phase transitions from water to ice. Subsurface electrical resistivity images can be obtained through inversion of electromagnetic data, yet are inherently ambiguous because of the ill-posed nature of the inverse problem. Since regularization methods offer the possibility to stabilize the inversion, lateral and spatial constraints are incorporated in the inversion algorithm to produce quasi-2D and quasi-3D subsurface models. The developed methodology is evaluated based on synthetic data sets to determine suitable inversion parameters, which are subsequently applied to a field example from the Seward Peninsula, Alaska. Laterally constrained inversion methods based on a few-layer starting model succeed in resolving sharp interfaces in quasi-layered environments. In more complex settings minimum-structure models can retrieve accurate subsurface representations leveraging on vertical and horizontal smoothness constraints. Enforcing lateral and spatial consistency between neighboring soundings thereby yields a similar degree of model smoothness. The inverted field data confirms the conclusions drawn from the synthetic study, as meaningful three-layered models with regard to electrical resistivities are recovered, indicating resistive snow overlying the conductive active layer and highly resistive permafrost. However, the inversion results imply that the snow layer has a significant effect on the predicted model. The implemented constraints help in reducing the ambiguity of the models, but uncertainties introduced by limited data availability cannot be overcome. The potential of adopting spatial and lateral constraints to the inversion is shown, although it becomes evident that additional a priori information needs to be integrated in the objective function in order to comprehensively image the active layer.

Acknowledgements

My sincere gratitude goes to my principal supervisor Dr. Florian Wagner (RWTH Aachen University) for his valuable guidance and helpful suggestions throughout my Master thesis. I very much appreciate his willingness to share his programming expertise, which enabled me to vastly expand my coding skills.

I would like to express my deepest appreciation to my second supervisor Dr. Sebastian Uhlemann (Lawrence Berkeley National Laboratory, LBNL). He provided me with encouragement and constructive advice at all times. I am incredibly thankful for his great effort, which allowed me to join on-site at the LBNL in Berkeley, California. His passion for geophysics and enthusiasm for research have inspired me immensely.

Many thanks to Dr. Baptiste Dafflon (LBNL) for offering me the opportunity to contribute to the Next-Generation Ecosystem Experiments (NGEE) Arctic project. Furthermore, I would like to thank all colleagues from Berkeley Lab's Environmental Geophysics group for their warm welcome.

I would like to extend my thanks to Prof. Florian Wellmann (RWTH Aachen University) for taking over as my primary supervisor. I would also like to acknowledge the assistance of Dr. Denise Degen (RWTH Aachen University) with organizational matters.

Finally, I would like to thank my parents for their continuous support during my educational career and for fully encouraging my decision to study abroad during my Bachelor and Master degree. Additionally, many thanks to my brother Lukas, who has always served as a role model to me. Special thanks go to Yannick for his great patience and support.

RWTH Aachen University
August 5, 2022

Marlene Thalhammer

Table of Contents

List of Figures

List of Tables

Acronyms

cW, cWeight Constraint weight

DOI Depth of investigation

EMI Electromagnetic induction

ERT Electrical resistivity tomography

FDEM Frequency-domain electromagnetic

HCP Horizontal coplanar

LBNL Lawrence Berkeley National Laboratory

LCI Laterally constrained inversion

LIN Low induction number

MGS Minimum gradient support

MSE Mean squared error

NE Northeast

NGEE Next-Generation Ecosystem Experiments

SCI Spatially constrained inversion

SSIM Structural similarity

SW Southwest

TEM Time-domain electromagnetic

UAV Unmanned aerial vehicle

VCP Vertical coplanar

Chapter 1

Introduction

The relevance of permafrost degradation, enhancing global warming, is uncontested. Geophysical methods have the potential to improve the understanding of subsurface processes, yet challenges remain with regard to the inversion of geophysical data. This motivates the development of modern inversion algorithms in this Master thesis, and underlying objectives and methodology are elaborated in the following.

1-1 Motivation

In recent decades climate change was found to be especially pronounced in Arctic regions compared to the Northern Hemisphere as a whole. The associated surface air temperatures show an increasing trend about twice as fast as the rest of the world. This observation, known as a result of Arctic amplification, is a decisive factor in climate model projections over the next century [Miller et al., 2010; Serreze and Barry, 2011]. Permafrost degradation is considered one of the many influencing mechanisms further accelerating Arctic warming. Perennially frozen ground or permafrost is a widespread phenomenon in regions of high latitude and altitude [Van Huissteden, 2020]. Rising terrestrial temperatures and snow thicknesses respond to an extension of the seasonally thawing surface layer (i.e., the active layer), and consequently lead to a breakdown of organic carbon stored in frozen soils [Biskaborn et al., 2019; Jorgenson et al., 2006; Schuur et al., 2008]. The release of carbon-containing greenhouse gases sustainably alters the carbon cycle, thereby producing a positive feedback in the Arctic climate [Miller et al., 2010; Schuur et al., 2008]. Not only does permafrost degradation affect the atmosphere, but also impacts upon subsurface physical properties that influence man-made infrastructure, landscapes, and ecosystems. Changes in permafrost conditions, including but not limited to active layer thickening, can be linked to an increase in natural hazards, which induce damages to polar and mountainous infrastructure, posing a major threat to local communities [Hjort et al., 2022]. Abrupt permafrost thaw events caused by either temperature or precipitation extremes result in a rising magnitude and frequency of landslides [Harris, 2005; Niu et al., 2016].

Furthermore, climate-induced modifications of the permafrost distribution are expected to have adverse effects on hydrological processes related to the interaction between surface and subsurface water as well as on vegetation originating from variations in the physical soil foundation [Christensen et al., 2004; Uhlemann et al., 2021; Walvoord and Kurylyk, 2016].

Permafrost science and engineering are essential to foster development in Arctic regions and climate adaptation as permafrost continues to warm. Although it is agreed upon that permafrost degradation is long underway, the rate and extent remain uncertain [Slater and Lawrence, 2013]. A more holistic understanding of subsurface process interactions, ground temperature development, and hydrology is required [Uhlemann et al., 2021]. In addition, comprehensively quantifying and timing the permafrost carbon feedback is necessary in order to make confident climate projections [Schuur et al., 2015]. Currently, knowledge and data about permafrost distribution and structure at depth is sparse and data acquisition is challenging due to inaccessibility, sensitive ecosystems, and environmental conditions. However, exhaustive investigations of permafrost properties and processes, and development of models representing ecosystem-climate feedback are crucial to assess both the present situation and future trajectory of permafrost [Minsley et al., 2012; Uhlemann et al., 2021; Walvoord and Kurylyk, 2016].

Active layer thickness is most commonly determined to evaluate permafrost thaw [Walvoord and Kurylyk, 2016]. Data is usually obtained from conceptual models based on surface observations, borehole measurements, and geophysical surveys [Minsley et al., 2012]. Remote sensing, although only measuring surface properties, enables the study of subsurface permafrost variables (i.e., ground temperature, ice content, and thaw depth) using empirical and statistical methods. Point-scale measurements can be extrapolated and estimation of active layer thickness is possible, yet further research is necessary to overcome limitations that stem from sparse data and resolution incompatibility with discontinuous permafrost [Walvoord and Kurylyk, 2016; Westermann et al., 2015]. While boreholes give a valuable insight into the ground temperature evolution at different depths, drilling in ice-rich ground is expensive as well as time-consuming and only point-source information at discrete locations can be gained. Temperature sensors, however, cannot clearly distinguish between freezing and thawing processes in the subsurface. This is due to latent heat exchange, which takes place during phase change at temperatures close to 0 °C [Boaga et al., 2020; Kneisel et al., 2008; Mollaret et al., 2019]. In general, ice content is difficult to assess quantitatively relying on temperature data alone, as the presence of ice cannot be inferred with certainty from below-zero temperatures. Dry or saline permafrost is not necessarily frozen and freezing point depression has to be taken into account [Dobinski, 2011; Subcommittee, 1988].

Various geophysical tools including ground-based as well as airborne techniques have proven to be a suitable alternative to spatially delineate permafrost, active layer, and taliks [Walvoord and Kurylyk, 2016]. They allow for a rapid, indirect, and non-invasive investigation of subsurface physical properties and their distribution at low costs, thereby offering a high spatial and temporal resolution up to an investigation depth of a few tens of meters [Hauck, 2013; Kneisel et al., 2008]. Electrical resistivity tomography (ERT), electromagnetic induction (EMI), ground-penetrating radar as well as refraction seismics have been extensively applied for subsurface imaging and monitoring of permafrost areas both in two and three dimensions [Hauck, 2013; Walvoord and Kurylyk, 2016]. Even though

ERT has been successfully used in permafrost-related surveys, limitations arise in specific environments, where sufficient galvanic coupling between electrodes and ground is difficult to achieve. Seismic surveys enable a high-resolution 3D visualization of frozen subsurface characteristics [Hauck, 2013]. Electromagnetic methods, and specifically frequency-domain electromagnetics (FDEM), cover the need for efficient and easy-to-use tools that are appropriate for estimating the thickness of the active layer as well as identifying the ground ice distribution [Boaga et al., 2020; Kneisel et al., 2008].

The success of geophysical methods has been demonstrated in permafrost regions, due to strong contrasts that exist in the physical properties of permafrost and the surrounding non-cryotic (referring to the temperature of the permafrost, according to Subcommittee [1988]) material. The variation is mostly attributed to the ice and liquid water content in the pore space of sediments and can be detected by differences in electrical resistivity (or electrical conductivity respectively), seismic wave velocity, and dielectric permittivity [Kneisel et al., 2008; Walvoord and Kurylyk, 2016]. In particular electrical properties show a distinctive change occurring at the phase transition from an unfrozen to a frozen state. Electrical resistivity is also extremely susceptible to varying ground temperatures, especially below the freezing point [Mollaret et al., 2019; Uhlemann et al., 2021; Wu et al., 2017]. The measurement of electromagnetic fields relates to the electrical resistivity in the subsurface, and thus EMI tools have been widely employed for studies in periglacial environments. The high sensitivity to water and ice variations in the ground enables clear imaging of the active layer and discrimination from the underlying permafrost [Butler, 2005; Kneisel et al., 2008].

Structural interpretation of subsurface properties is achieved by mathematical inference of acquired geophysical data. An electrical resistivity subsurface model can be constructed, ideally revealing the characteristic permafrost composition [Kneisel et al., 2008; Siemon, 2009]. However, inversion of geophysical data, and in particular electromagnetic soundings, is challenging due to its non-uniqueness [Boaga et al., 2020; Hubbard et al., 2013; Oldenburg, 1990]. Not only is the electromagnetic inversion highly non-linear, data sparsity and limited data coverage contribute to the ambiguity of generated results [Kneisel et al., 2008; Knödel et al., 2007; Oldenburg, 1990]. The large discrepancy in electrical resistivity between frozen and unfrozen substrate further complicates the inversion and subsequent interpretation [Marescot et al., 2003]. Such ill-posed problems can be stabilized by regularization methods, which incorporate desired structures into the inversion [Deng and Tong, 2020; Hauck, 2013; Zhdanov, 2015]. The integration of a priori information, including available data from boreholes and complimentary geophysical surveys, makes it possible to generate more realistic resistivity models [Boaga et al., 2020; Hubbard et al., 2013; Knödel et al., 2007]. Modern inversion algorithms utilize joint inversion of different geophysical data sets in order to overcome the challenge of single-method imaging [Boaga et al., 2020; Mollaret et al., 2020; Wagner et al., 2019]. Even though Guillemoteau and Tronicke [2016] and Guillemoteau et al. [2017] have introduced new fast forward modeling approaches, full 2D and 3D electromagnetic inversion algorithms have not been commonly applied in large-scale environmental studies [Knödel et al., 2007]. Therefore, EMI processing is usually based on a layered half-space model (1D layered inversion) to avoid high computational costs while ensuring robustness [Davies et al., 2015; Guillemoteau et al., 2016; Saey et al., 2012; Siemon, 2009]. To find an adequate solution to the under-determined inversion problem, prior information in the form of smoothness constraints can be incorporated in the objective

function [Constable et al., 1987; Siemon, 2009]. A disadvantage might be that the assumption of smooth models restricts the characterization of sharp interfaces [Guillemoteau et al., 2022; Klose et al., 2022]. So far, few studies have addressed the intrinsic non-uniqueness of electromagnetic inversion by implementing lateral and spatial constraints. Auken and Christiansen [2004] and Auken et al. [2005b] present a 2D laterally constrained inversion (LCI) algorithm that guarantees lateral consistency between adjacent 1D profiles. Likewise, quasi-3D sections were generated by jointly inverting neighboring 1D models, while enforcing spatial coherency, termed a spatially constrained inversion (SCI) [Viezzoli et al., 2007, 2008]. Nevertheless, sharp boundaries between different geological formations cannot be reproduced according to the true subsurface model [Auken and Christiansen, 2004; Guillemoteau et al., 2022; Klose et al., 2022]. Recently, progress has been achieved by imposing regularization through minimum gradient support (MGS) stabilizers. The so-called focusing parameter proposes layered models, keeping the balance between sharp and smooth solutions [Klose et al., 2022; Vignoli et al., 2021]. Application with respect to FDEM soundings, however, is limited to the work by Klose et al. [2022], motivating further research focused on inversion strategies that deal with the non-unique nature of electromagnetic inversion, and at the same time accommodate the demand for exhaustive subsurface models.

1-2 Objectives

Monitoring of permafrost is essential for climate projections, subsequently showing the importance of active layer delineation. This emphasizes the need for non-invasive geophysical techniques that can produce accurate subsurface images and are applicable at large scales. This study intends to reduce the ambiguity inherent to the inversion of electromagnetic soundings by incorporating structural and spatial information in the inversion algorithm. Given that few methods exist that allow a stable, yet computationally efficient two-dimensional and three-dimensional inversion producing quasi-layered models, 1D forward solutions are employed to recover multi-dimensional subsurface models, leveraging on a smoothness-constrained regularization technique. Model parameters including layer thicknesses and resistivities are constrained to those of neighboring soundings. Simultaneously, the aim is to characterize sharp interfaces between subsurface layers. Structural decoupling of distinct layers can be accomplished by formulating a suitable constraint and weighting matrix, which can be incorporated in the objective function to be minimized. The project applies the inversion scheme to synthetic and field data sets, evaluating the benefits of laterally and spatially constrained inversion compared to conventional 1D inversion, based on both few-layer and Occam type inversion models.

1-3 Structure and methodology

The methodology for the Master thesis project is illustrated in Figure 1-1. The graph gives a simplified overview of the process steps taken in the research study, which are explained in more detail in the following.

Firstly, a state-of-the-art review was conducted and a brief summary of the theory including

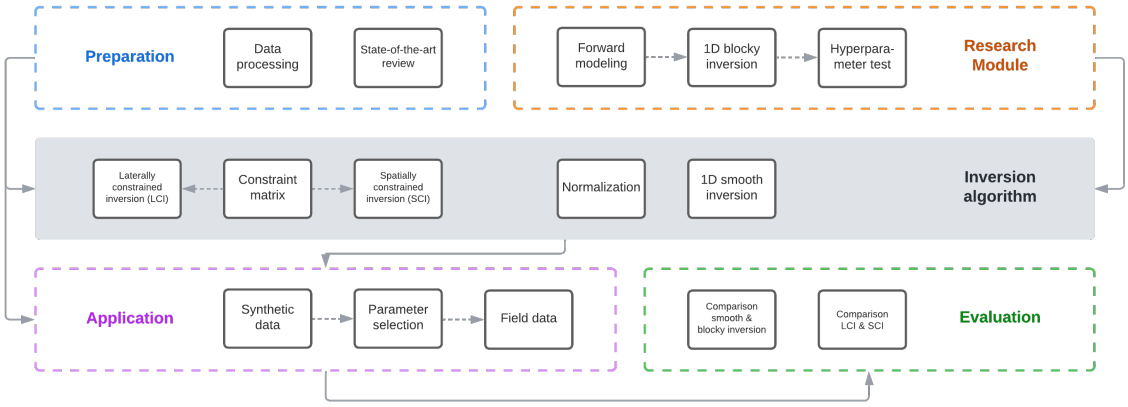


Figure 1-1: Flowchart representing the methodology and underlying structure of the Master thesis as well as the built-up on the previously conducted "Research Module" at the RWTH Aachen University by Thalhammer [2022].

the definition of permafrost, the basics of the frequency-domain electromagnetic (FDEM) method as well as the concept of inverse theory in geophysics is presented in Chapter 2. Furthermore, the implementation of lateral and spatial constraints in inversion algorithms is introduced. During the preparation phase (Figure 1-1: Preparation), the data sets acquired by the Lawrence Berkeley National Laboratory (LBNL) were processed and visualized in *ArcGIS*.

Chapter 3 forms the main part of the study and focuses on the development of quasi-2D and quasi-3D inversion methods accomplished through inclusion of lateral and spatial constraints (Figure 1-1: Inversion algorithm). This builds on the forward modeling scheme as well as the 1D blocky inversion technique introduced by Thalhammer [2022] in the course of the "Research Module" held at RWTH Aachen University (Figure 1-1: Research Module) and was further extended to a 1D smooth inversion.

The applicability of laterally and spatially constrained inversion (i.e., LCI and SCI) methods (Figure 1-1: Application) is demonstrated in Chapter 4 and 5. This is achieved through utilization of both synthetic data sets as well as field data from a permafrost monitoring site in Teller, Alaska. The parameter selection for the presented inversion models will be explained in detail.

Chapter 6 is dedicated towards generated inversion models based on an evaluation of used inversion methods for active layer imaging (Figure 1-1: Evaluation). Last but not least, Chapter 7 and 8 summarizes the extension of 1D algorithms towards more sophisticated inversion methods within the scope of this project and outlines topics for future research.

Chapter 2

Theory

This chapter introduces the theoretical background for imaging of permafrost soils utilizing geophysical methods. The composition of permafrost and the implications of permafrost degradation are explained, followed by a review of electromagnetic methods, in particular techniques operated in frequency-domain. Additionally, the concepts of laterally and spatially constrained inversion (i.e, LCI and SCI) are presented.

2-1 Permafrost

Permafrost or permanently frozen ground is soil or rock which remains at or below 0 °C for at least two consecutive years. Therefore, permafrost does not refer to areas that only seasonally experience below-zero temperatures [Dobinski, 2011; Van Everdingen et al., 1998; Van Huissteden, 2020]. Regardless of the temperature, the soil may not contain ice if the salt content of the pore water is high, because the salt causes freezing point depression, i.e., a decrease of the temperature at which ice forms [Dobinski, 2011; Van Huissteden, 2020]. Permafrost usually occurs in high-latitude regions, such as Alaska, as well as at high elevations (e.g., the Alps) and both in North America and Asia permafrost is particularly widespread. Depending on the geographical location, permafrost exhibits different spatial continuity and appears in continuous, discontinuous, sporadic, and isolated states (Figure 2-1). In addition, physical factors such as vegetation and soil conditions as well as topography and winter snow cover determine the presence of permafrost [Van Huissteden, 2020].

Figure 2-1 provides an exemplary overview of the typical composition of a permafrost soil. The top layer represents the active layer (i.e., the seasonal thaw layer), which is subject to annual thawing and freezing, underlain by permafrost [Dobinski, 2011; Van Everdingen et al., 1998]. The base of the active layer is usually defined as the permafrost table. The thickness of the active layer, also referred to as the thaw depth, varies up to more than a meter and is generally thicker in more southern permafrost areas. The permafrost base might range from a few meters up to more than a kilometer. Taliks are common features

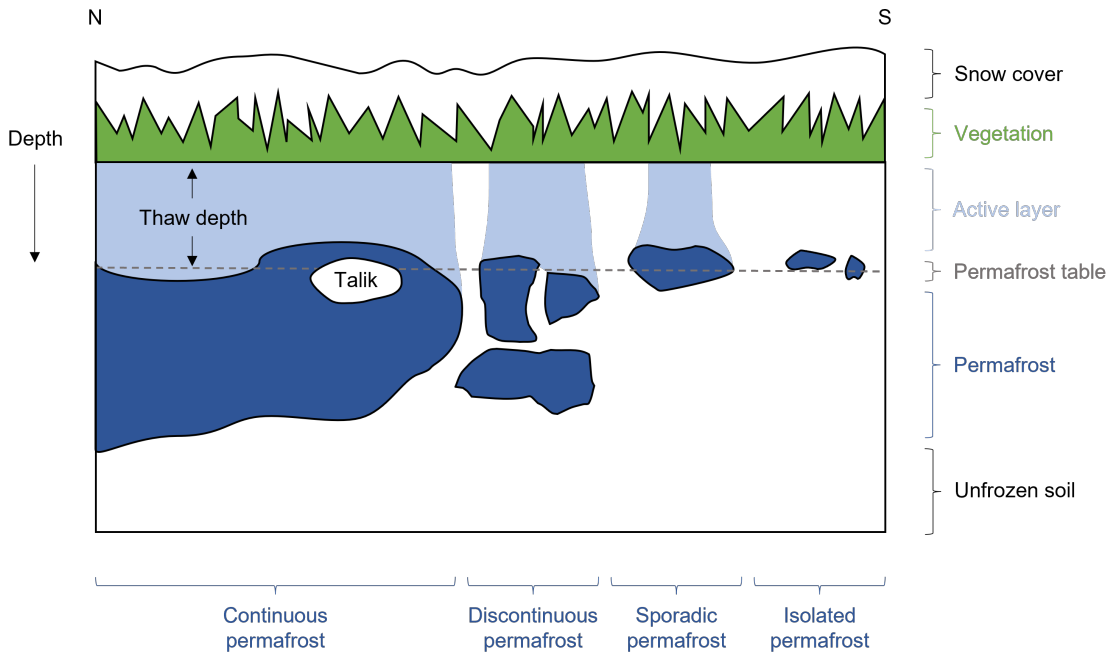


Figure 2-1: Cross section through continental permafrost illustrating differences in spatial continuity of active and permafrost layers (modified after Van Huissteden [2020]).

in permafrost environments, and constitute bodies of unfrozen rock or soil within frozen material [Van Huissteden, 2020].

A warming climate as well as the length of the thaw season are decisive factors driving permafrost thaw [Van Huissteden, 2020]. Sufficient warming will lead to the formation of taliks between active layer and permafrost and subsequently to a vertical extension of the active layer [Kane et al., 1991]. Regionally, the seasonal thaw depth is affected by additional variables such as the physical and thermal properties of the medium including heat capacity and heat permeability, vegetation cover, soil moisture content, albedo, as well as thickness and duration of snow cover [Dobinski, 2011; Hinzman et al., 2005; Wollschläger et al., 2010]. The snow cover acts as an insulator and minimizes heat loss from the ground to the atmosphere [Kane et al., 1991]. Since both annual air temperature and precipitation are expected to further increase in upcoming years, specifically in the winter season, the active layer thickness has become an important indicator in climate change research [Biskaborn et al., 2019; Van Huissteden, 2020].

Permafrost degradation both defines the increase of the active layer as well as the temperature growth within the permafrost itself due to a positive heat balance [Dobinski, 2011]. Most ecological, hydrological, biochemical, and pedogenical processes are confined to the active layer. Thus, permafrost degradation can be linked to the modification of subsurface ecosystem dynamics at local to regional scales [Kane et al., 1991; Walvoord and Kurylyk, 2016]. This includes alterations of the carbon and nutrient cycle of permafrost soils [Van Huissteden, 2020]. Permafrost carbon developed through accumulation of plant and animal remnants over tens of thousands of years, and permafrost thaw will expose

this organic carbon, enhancing microbial decomposition by soil fauna and bacteria [Schuur et al., 2015; Van Huissteden, 2020]. As a result, carbon-containing greenhouse gases such as methane and carbon dioxide are released into the atmosphere and the ocean [Schuur et al., 2015]. This mechanism is referred to as the permafrost carbon feedback and describes the acceleration of global warming due to permafrost thaw and subsequent emissions of greenhouse gases [Van Huissteden, 2020].

2-2 Frequency-domain electromagnetic method

Electromagnetic induction (EMI) tools are widely used to characterize geoelectrical features in near-surface layers. Electromagnetic methods are based on the measurement of electromagnetic fields, which relate to the electrical resistivity (or its reciprocal, electrical conductivity, respectively). Common techniques include frequency-domain electromagnetic (FDEM) and time-domain electromagnetic (TEM) systems. In the most simple form they are composed of two coils, a receiver and transmitter, whereby the latter induces electrical currents in the subsurface, and the other measures the subsurface response (Figure 2-2). In contrast to conventional electrical resistivity tomography (ERT) methods, electromagnetic techniques do not require galvanic coupling to the ground [Knödel et al., 2007].

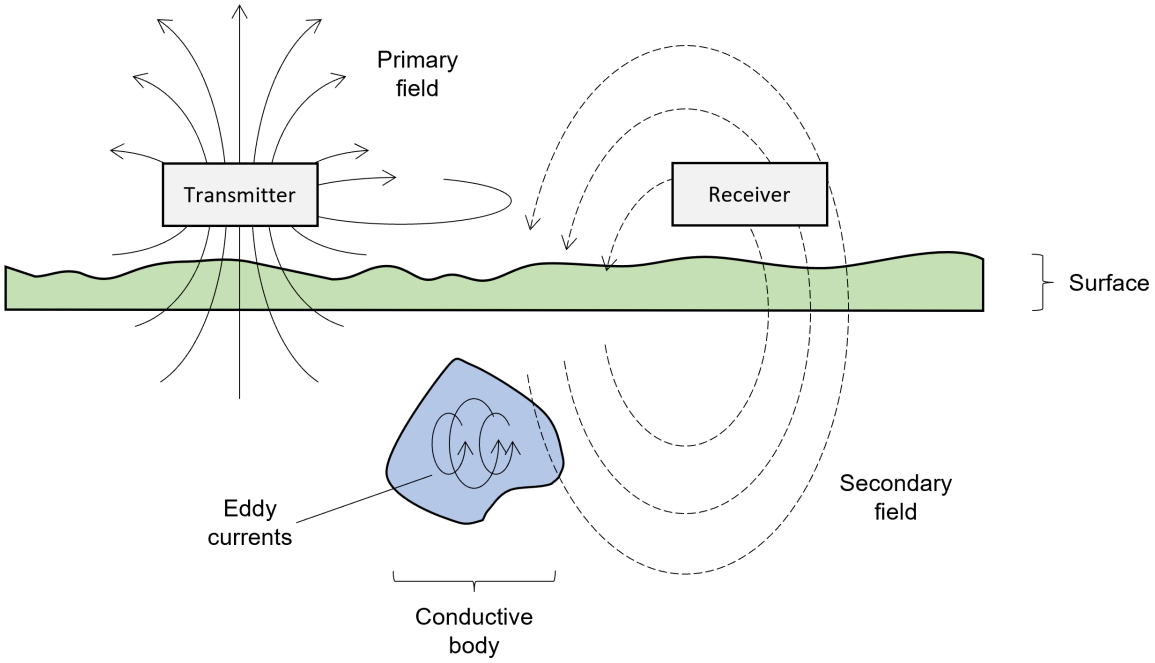


Figure 2-2: General principle of EMI methods (modified after Kearey et al. [2002] and Knödel et al. [2007]).

Depending on whether operated in the frequency or time-domain, primary electromagnetic fields are generated through alternating or terminating currents, respectively [Kneisel et al., 2008].

2-2-1 Mathematical background

The governing laws which relate the electromagnetic fields to physical properties are described by Maxwell's equations [West and Macnae, 1991]. According to Faraday's law, an oscillating primary electromagnetic field **B** creates eddy currents in conductive bodies, as displayed in Figure 2-2:

$$\text{Faraday's law} \qquad \nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}. \tag{2-1}$$

Here, **E** [V/m] describes the electric field intensity and **B** [Wb/m^2] the magnetic induction [Butler, 2005]. In turn, the resulting eddy currents **J** induce a secondary electromagnetic field **H**, which can be measured by one or more receiver loops at the surface [Kneisel et al., 2008]. The higher the electrical conductivity of subsurface features, the larger the eddy currents and consequently the greater the secondary field [Knödel et al., 2007]. The relationship between the current density **J** and the electric field **E** is given by Ohm's law:

$$\text{Ohm's law} \qquad \mathbf{J} = \sigma \mathbf{E}, \tag{2-2}$$

with σ [S/m] being the electrical conductivity and **J** [A/m^2] the current density [West and Macnae, 1991]. Ampere's law describes the secondary field **H**, which can be related to **J**, according to

$$\text{Ampere's law} \qquad \nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t}, \tag{2-3}$$

where **H** [A/m] is the magnetic field intensity, and **D** [C/m^2] is the electric displacement [Butler, 2005; Knödel et al., 2007]. The primary and secondary electromagnetic field are superimposed and detected by the receiver coil through the process of EMI. The response of combined fields differs in both phase and amplitude from the transmitted primary electromagnetic field and can be analyzed to obtain information about the subsurface geometry and geoelectrical properties [Kearey et al., 2002; Knödel et al., 2007].

Apart form the subsurface conductivity, the strength of the secondary electromagnetic field depends on the transmitter-receiver coil configuration as well as the operating frequency [Knödel et al., 2007]. The measured electromagnetic field is a complex quantity due to a phase shift between primary and secondary field caused by the induction process within the conducting earth. As a result, the complex valued-function is expressed through in-phase (real part) and quadrature (imaginary part) components [Kearey et al., 2002; Siemon, 2009]. Generally, the secondary field is determined in relative terms, for instance, in parts per million [ppm] against the primary field [Martinelli and Duplaá, 2008].

FDEM soundings are performed through varying frequency and/or coil spacing, also referred to as the coil separation [Knödel et al., 2007]. The coil orientation can be adjusted for both receiver and transmitter, whereby the most common configurations are vertical coplanar (VCP) and horizontal coplanar (HCP), as illustrated in Figure 2-3. The distance between the coils (i.e., coil separation) usually remains constant and the EMI system is moved

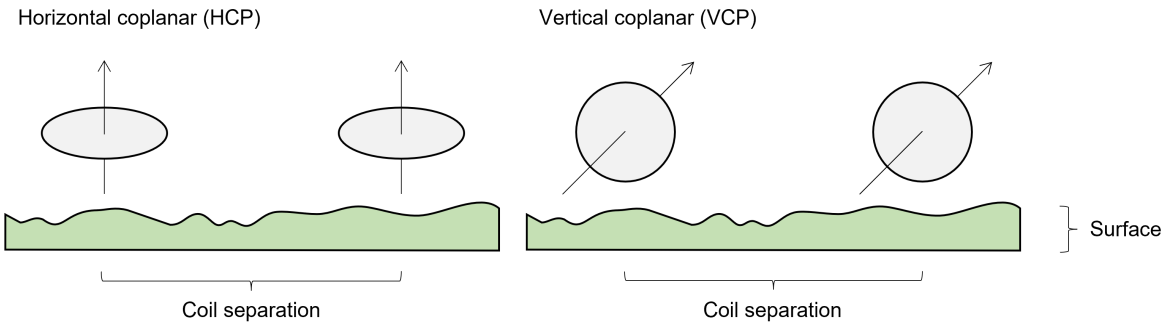


Figure 2-3: Typical coil-pair geometries between transmitter and receiver used in electromagnetic surveys (modified after Butler [2005] and Knödel et al. [2007]).

over the study area at an approximately steady height [Martinelli and Duplaá, 2008]. The investigation depth of the transmitted electromagnetic signals is controlled by the frequency and the coil separation. In general, generated eddy currents attenuate the electromagnetic field, negatively impacting the signal penetration [Knödel et al., 2007]. The depth of penetration, also referred to as the skin depth p [m], is given by

$$\text{Skin depth} \qquad p = \sqrt{\frac{2}{\sigma\omega\mu_0}} = 503\sqrt{\frac{1}{\sigma f}}, \tag{2-4}$$

with the angular frequency ω [rad/s], the magnetic permeability of vacuum μ_0 [H/m], and the frequency f [Hz] [Siemon, 2009]. According to Knödel et al. [2007], the penetration depth benefits from an increasing coil distance and a decreasing frequency. Moreover, HCP systems are known to have the greatest investigation depth, whereas VCP tools are most sensitive to near-surface layers.

The determination of ground conductivity is complicated, but can be simplified if the low induction number (LIN) principle holds. The LIN condition is met if the coil spacing is significantly smaller than the skin depth (cf. Equation 2-4). This can be accomplished by choosing a low enough operating frequency. Consequently, the signal response predominantly consists of the quadrature component and can be linearly linked to the conductivity of the ground [Knödel et al., 2007; McNeill, 1990]. The mean ground conductivity (i.e., apparent conductivity σ_a) can then be obtained directly, described by the following formula:

$$\text{Apparent conductivity} \qquad \sigma_a = \frac{4}{\omega\mu_0 s^2}\frac{\mathbf{H}_s}{\mathbf{H}_p}. \tag{2-5}$$

Thereby, $\mathbf{H}_p$ and $\mathbf{H}_s$ denote the amplitudes of primary and secondary electromagnetic fields, and s the coil spacing (i.e., coil separation) [Kearey et al., 2002]. As explained by Butler [2005], the apparent electrical conductivity equals the actual conductivity of the subsurface only in case of a homogeneous half-space. In fact, the apparent conductivity constitutes a weighted average over the investigated area. However, for heterogeneous environments the "true" electrical conductivity can be inferred through inversion methods, which will be expanded upon in Chapter 2-3 [Hauck et al., 2008].

2-2-2 Application

Electromagnetic methods, including FDEM tools, are widely used in environmental studies as they are based on the quantification of electrical conductivity of the ground [Butler, 2005; Knödel et al., 2007]. Generally, electrical resistivity variations can be attributed to differences in mineral composition, porosity, water saturation, salinity, as well as ice and clay content, as shown in Table 2-1 [Hauck et al., 2008]. In permafrost areas changes in electrical properties are particularly pronounced between frozen and unfrozen conditions [Butler, 2005; Wu et al., 2017]. Frozen substitutes are usually characterized by resistivities that are orders of magnitude larger than in unfrozen conditions (Table 2-1). Previous field and laboratory studies demonstrated that decreasing temperatures near the freezing point correspond to an increase in resistivity unbiased by the operating frequency [Hauck et al., 2008]. However, the actual resistivity values depend on the frequency and vary with different materials and unfrozen pore water content [Kneisel et al., 2008]. Special care must be taken with respect to distortions of the data caused for example by variable snow cover or electrical power lines [Hauck et al., 2008; Kneisel et al., 2008]. Nevertheless, electromagnetic methods distinguish well between ice and water, i.e., the active and permafrost layer, and are also easy to operate and highly efficient for investigations of the near-surface [Knödel et al., 2007]. Hence, electromagnetic tools have been continuously applied in permafrost research for many decades, both with regard to lowland arctic and submarine permafrost, but have also been increasingly utilized in mountainous regions in recent years [Hauck et al., 2008]. The usage ranges from FDEM [Daniels et al., 1976; Korhonen et al., 2009] to TEM methods [Harada et al., 2000; Hauck et al., 2001], including airborne electromagnetics [Madsen et al., 2022; Minsley et al., 2012]. As mentioned beforehand, no galvanic contact to the ground is required, which underlines the benefit of EMI on sealed terrains and high-resistive cover layers such as on a frozen surface or in arid areas [Knödel et al., 2007; Siemon, 2009]. Additionally, modern single-frequency, multi-configuration EMI devices with varying distances between the transmitter and several receiver coils enable an evaluation of the electrical conductivity at various depth levels [Bonsall et al., 2013; Saey et al., 2012].

Table 2-1: Typical ranges of electrical resistivity for common materials. The conversion of electrical resistivity ρ to electrical conductivity σ is carried out through the formula $\sigma = 1/\rho$, so that a conductivity of $\sigma = 1$ mS/m corresponds to a resistivity of $\rho = 1{,}000$ Ωm (modified after Hauck et al. [2008]).

Material	Range of resistivity ρ [Ωm]
Clay	1-100
Sand	100-5,000
Gravel	100-400
Groundwater	10-300
Frozen sediments, ground ice	1,000-10^6

2-3 Inversion

The main goal of a geophysical inversion is to define subsurface models that best describe indirect field observations. Measured signals from geophysical surveys are thereby used to infer unknown physical properties and processes, which relate to the structure of a medium of interest, termed an inverse problem [Kneisel et al., 2008; Zhdanov, 2015].

The model vector $\mathbf{m}$ holds the physical model parameters and is linked to the data vector $\mathbf{d}$ that is usually discrete in time and space, by the forward modeling equation [Butler, 2005], according to:

$$\text{Forward problem} \qquad \mathbf{d} = f(\mathbf{m}). \tag{2-6}$$

The forward modeling operator f typically constitutes a differential equation and needs to be calculated numerically, as is the case for EMI surveys governed by Maxwell's equations (cf. Chapter 2-2). Correspondingly, synthetic or predicted data $\mathbf{d}_{pred}$ can be produced by assuming a subsurface model and solving the forward problem [Kneisel et al., 2008]. On contrary, inverse modeling defines the estimation of model parameters $\mathbf{m}_{est}$ utilizing observed data $\mathbf{d}_{obs}$ (i.e., measured data) conditioned by physical laws [Zhdanov, 2015], given by:

$$\text{Inverse problem} \qquad \mathbf{m}_{est} = f^{-1}(\mathbf{d}_{obs}). \tag{2-7}$$

The operator f^{-1} describes the inverse transformation from data to model space. An optimal representation of the subsurface would be a model $\mathbf{m}$ (often referred to as true model) that exactly reconstructs the observations $\mathbf{d}_{obs}$ [Zhdanov, 2015]. The inversion of geophysical data, and in particular electromagnetic soundings, however, is usually non-unique, and thus challenging. Multiple models can be found that explain the data to a predefined error level, as the inversion problem is in most cases under-determined, i.e., the number of model parameters exceeds the number of data [Backus and Gilbert, 1967; Hubbard et al., 2013; Snieder and Trampert, 1999]. Non-uniqueness can stem from insufficient data as observation points in geophysical surveys are typically limited and besides, acquired data sets are contaminated by noise. Underlying mathematical relationships are often simplified, alongside an intrinsic physical non-uniqueness. Hence, inversion results are generally ambiguous and inaccurate [Kneisel et al., 2008; Knödel et al., 2007; Zhdanov, 2015]. The principal questions in geophysical inverse theory therefore deal with the existence, uniqueness, and stability of the solution [Zhdanov, 2015].

The non-linearity of the forward modeling equation associated with most geophysical problems further complicates the inversion of the forward operator f [Butler, 2005]. Highly non-linear problems typically require an iterative process to fit the model response (i.e., the predicted data $\mathbf{d}_{pred}$) to the observed data $\mathbf{d}_{obs}$ [Smith et al., 1999]. The procedure includes subsequent forward modeling and inversion until either the misfit is sufficiently minimized or the maximum number of iterations is reached [Knödel et al., 2007]. The inverse problem is solved by minimizing the objective function $\Phi(\mathbf{m})$ with respect to the model vector $\mathbf{m}$ [Butler, 2005]:

$$\text{Objective function} \qquad \Phi(\mathbf{m}) = \Phi_D(\mathbf{m}) + \lambda\Phi_M(\mathbf{m}) \rightarrow min. \tag{2-8}$$

The minimization problem is comprised of the following three components: data misfit $\Phi_D(\mathbf{m})$, model objective function $\Phi_M(\mathbf{m})$, and regularization parameter λ (also referred to as trade-off parameter). Φ_D describes the l_2-norm of the weighted residual between data $\mathbf{d}$ and model response $f(\mathbf{m})$ according to

$$\text{Data misfit} \qquad \Phi_D = \|\mathbf{W}_D(f(\mathbf{m}) - \mathbf{d})\|_2, \tag{2-9}$$

with the data weighting matrix $\mathbf{W}_\mathrm{D}$ containing the inverse of the data errors. Φ_M is expressed by

$$\text{Model objective function} \qquad \Phi_M = \|\mathbf{W}_M(\mathbf{m} - \mathbf{m}_r)\|_2, \tag{2-10}$$

composed of the model control matrix $\mathbf{W}_\mathrm{M}$ and the reference model $\mathbf{m}_\mathrm{r}$. The regularization parameter λ finds the relative balance between the fit ($\lambda \to 0$) and misfit ($\lambda \to \infty$) of the data [Butler, 2005; Zhdanov, 2015].

The qualitative measure of the agreement between observed and predicted data is often expressed as the error criterion χ^2

$$\text{Error criterion} \qquad \chi^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\mathbf{d}_{obs,i} - \mathbf{d}_{pred,i}}{\mathbf{e}_i}\right)^2, \tag{2-11}$$

where n represents the length of the data vector $\mathbf{d}$, $\mathbf{d}_i$ the i^{th} point of the measured and predicted data, respectively, and $\mathbf{e}_i$ the i^{th} point of the data error [Ruiz-Aguilar et al., 2018]. If $\chi^2 = 1$, an optimal fit within the given error margin is achieved if only Gaussian error is present [Costabel et al., 2016]. Different techniques are used to solve geophysical inverse problems, including, but not limited to, least squares and gradient-type methods [Zhdanov, 2015].

Geophysical inverse problems are inherently ill-posed, i.e., their solutions are characterized by either non-uniqueness or instability. This can be addressed by implementing regularization approaches, that exploit a priori geological and geophysical information [Zhdanov, 2015]. Thereby, the regularization parameter λ specifies the degree of regularization imposed on the problem [Hansen, 1992]. One methodology to select a suitable regularization parameter is the so-called L-curve criterion based on the analysis of the Tikhonov curve, as outlined in Chapter 3-3-2 [Butler, 2005]. The behavior of the curve indicates the trade-off between over-regularization (high λ) and under-regularization (small λ). On the one hand, over-regularized models are smooth, yet, they do not sufficiently describe the data. On the other hand, under-regularization produces rough models with artifacts as no a priori information is added to the minimization problem [Butler, 2005; Hansen, 1992; Zhdanov, 2015]. One possibility to regularize the inverse problem is the integration of constraints in the objective function, which is discussed extensively in the course of this project [Knödel et al., 2007; Tartaras and Beamish, 2006].

2-3-1 Conventional 1D inversion

The standard inversion framework is based on a 1D forward model, where each data point is inverted individually, termed a 1D inversion. Because of the intrinsic ill-posedness of geophysical problems, regularization is necessary to solve the 1D inversion. This is due to minor perturbations in the data causing solutions to be unstable, while the solution non-uniqueness enables many models to describe the data to the same level [Vignoli et al., 2017; Zhdanov, 2015]. In one-dimensionality these challenges have been addressed by enforcing smoothness (smooth inversion) or sharpness (blocky inversion) in the inversion.

A reference model (i.e., inversion starting model) with a given number of layers, which is either user-defined or automatically generated, is required to launch the inversion. Associated model parameters are updated in an iterative procedure until an adequate agreement between calculated (i.e. predicted data) and measured data is achieved, as explain in Chapter 2-3 [Sengpiel and Siemon, 2000; Siemon, 2009].

The Levenberg-Marquardt algorithm is a suitable technique to solve such non-linear least squares problems by combining the popular Gauss-Newton method with the gradient descent method, and is frequently used for both airborne and ground electromagnetic data. Local damping is applied by decreasing the regularization strength consecutively after each iteration by a chosen factor. Initially, the model is updated in small steps in the steepest-descent direction to avoid overshoot. With decreasing data misfit, the Levenberg-Marquardt algorithm approaches the Gauss-Newton method as the damping parameter is reduced [Gavin, 2019; Rücker et al., 2017; Sengpiel and Siemon, 2000; Siemon, 2009].

Smooth inversion

In a smoothness-constrained inversion problem the smoothest model is picked that fits the data. This is based on the principle by Occam that favours the "simplest" solution to the underlying inverse problem [Constable et al., 1987]. The smooth model is obtained using a l_2-norm in the optimization algorithm [Ellis and Oldenburg, 1994; Loke et al., 2003]. A large number of thin layers is inverted, while vertical constraints ensure smooth transitions in the model [Schamper et al., 2012]. Many authors stress that although the data is explained to a satisfactory degree, formation boundaries cannot be easily distinguished [Tartaras and Beamish, 2006; Vignoli et al., 2017]. Hence, in a subsurface where a gradual change between material properties is unlikely and distinct layers are expected, a smoothly varying distribution of model parameters is not necessarily the most effective representation. As a consequence, inversion algorithms have been implemented that favour a blocky rather than a smooth appearance.

Blocky inversion

A few-layer (multi-layer) inversion process, herein also referred to as blocky inversion, produces a layered subsurface model during the geophysical inversion. This builds on a l_2-norm optimization algorithm. Models are constructed that exhibit piece-wise constant model parameters separated by sharp boundaries [Ellis and Oldenburg, 1994; Loke et al., 2003]. Smooth

inversions only concern geophysical parameters, while blocky inversion algorithms solve for primary model parameters, such as resistivities and layer thicknesses [Schamper et al., 2012]. The basic assumption of the blocky inversion is that the subsurface is horizontally layered [Knödel et al., 2007].

Inversion of electromagnetic data

Conventional 1D inversion generally is computationally efficient, while offering flexibility when geological settings vary. Even though noisy data can be processed and filtered preceding the inversion process, standard 1D inversion is often insufficient to fit the data to a satisfactory level, as the electromagnetic inversion concerns an under-determined problem [Auken et al., 2005b; Siemon et al., 2009]. Subsurface resistivity patterns can be rather complex, and structures are non-unique when it comes to lateral consistency and interface sharpness [Klose et al., 2022; Sengpiel and Siemon, 2000]. To overcome aforementioned problems, additional constraints can be employed in order to stabilize the inversion, which will be outlined in the following chapters.

2-3-2 Laterally constrained inversion

A laterally constrained inversion (LCI) embodies a simultaneous inversion of adjacent sounding locations in order to retrieve quasi-layered subsurface representations. As first introduced by Auken and Christiansen [2004], the 1D-LCI scheme produces pseudo-2D models with smooth transitions between soundings relying on a set of laterally constrained 1D calculations.

The basic assumption supporting the LCI concept is that model parameters are smoothly varying along a profile [Auken et al., 2009]. Thus, LCI models are generated by stitching together 1D forward responses (Figure 2-4), not necessarily being sampled at equidistance [Siemon et al., 2009]. Lateral continuity is enforced by regularizing the inverse problem, i.e., constraining neighboring model parameters [Auken et al., 2005b]. The constraining formulation is rather flexible and both primary and secondary model parameters can be included in LCI algorithms. Primary model parameters refer to geophysical parameters such as electrical resistivities and layer thicknesses, as outlined in Figure 2-4. Contrary, secondary model parameters can be deduced from primary parameters through linear relationships (e.g., depths and elevations) [Auken et al., 2005b, 2015; Smith et al., 1999]. Constraints on depths typically promote horizontal layer boundaries, while constraints on thicknesses benefit continuous thicknesses in layers, specifically in the presence of faults [Auken and Christiansen, 2004]. In general, the resulting quasi-2D section is laterally smooth with sharp layer interfaces, as displayed in Figure 2-4.

Lateral constraints are basically considered a priori information, whereby changes between model parameters of adjacent observation points are kept small. The lesser the expected variation of a model parameter, the stronger the constraint [Auken et al., 2005b; Socco et al., 2009]. According to Auken and Christiansen [2004], any available a priori information can be readily incorporated in the inversion process, further addressing the non-uniqueness of the underlying numerical problem. Pre-existing knowledge of geological conditions can be used to influence nearby model parameters. Both a priori as well as information from

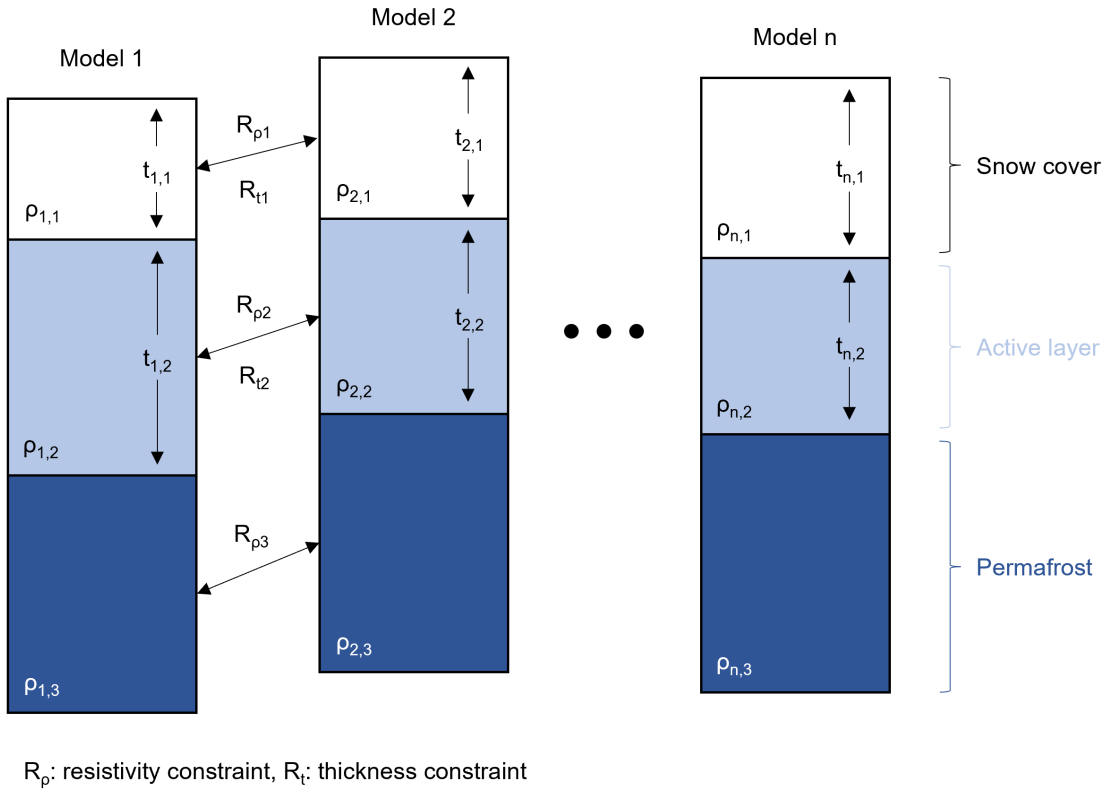


Figure 2-4: LCI model concept, where lateral constraints (denoted by R_ρ and R_t, respectively) connect primary model parameters, i.e., thicknesses t and resistivities ρ (modified after Auken et al. [2005b], Christiansen et al. [2007], and Fiandaca et al. [2012]).

well-resolved areas spread through lateral constraints to neighboring nodes along the profile (Figure 2-4). This implies that more data is available than model parameters, hence the number of unknowns for each model is significantly reduced. Consequently, LCI solves an over-determined problem, addressing the inherent ambiguity of inversion models [Auken and Christiansen, 2004; Auken et al., 2009; Socco et al., 2009].

The LCI algorithm provides a unique inversion process in which a 1D forward solution is directly applied to coincidentally obtain model parameters that built an exhaustive subsurface model. The corresponding forward problem is given by

$$\text{Forward problem} \qquad \delta\mathbf{d}_{obs} = \mathbf{G}\delta\mathbf{m}_{true} + \mathbf{e}_{obs}, \tag{2-12}$$

where $\mathbf{G}$ denotes the Jacobian matrix, $\mathbf{m}_{true}$ the true model, and $\mathbf{e}_{obs}$ the observational error [Auken and Christiansen, 2004]. The forward response and associated derivatives for the Jacobian matrix $\mathbf{G}$ are computed within the 1D formalism [Fiandaca et al., 2012]. The so-called roughening matrix $\mathbf{R}$ implements the lateral constraints, which are applied to the model vector $\mathbf{m}$ according to:

$$\text{Constrained forward solution} \qquad \mathbf{R}\delta\mathbf{m}_{true} = \delta\mathbf{r} + \mathbf{e}_r, \tag{2-13}$$

with $\mathbf{e}_r$ defining the error on the constraints and $\delta\mathbf{r}$ supporting the identity between parameters linked by constraints [Auken and Christiansen, 2004; Wisén and Christiansen, 2005]. By combining and rearranging Equations 2-12 and 2-13 the inversion problem can be written as

$$\text{Inversion problem} \qquad \begin{bmatrix}\mathbf{G}\\ \mathbf{R}\end{bmatrix} \cdot \delta\mathbf{m}_{true} = \begin{bmatrix}\delta\mathbf{d}_{obs}\\ \delta\mathbf{r}\end{bmatrix} + \begin{bmatrix}\delta\mathbf{e}_{obs}\\ \mathbf{e}_r\end{bmatrix}, \tag{2-14}$$

or summarized by

$$\text{Inversion problem} \qquad \mathbf{G\prime} \cdot \delta\mathbf{m}_{true} = \delta\mathbf{d\prime} + \mathbf{e\prime}. \tag{2-15}$$

Equation 2-16 displays the roughening matrix $\mathbf{R}$, which is a zero matrix, only introducing 1 and -1 values for constrained model parameters. Each matrix row enforces a constraint between two model parameters, specified in the respective columns [Tartaras and Beamish, 2006].

$$\text{Roughening matrix} \qquad \mathbf{R} = \begin{bmatrix} 1 & 0 & \dots & 0 & -1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & -1 & 0 & \dots & 0 & 0 \\ \vdots & & & & & \vdots & & & & \vdots \\ 0 & 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & -1 \end{bmatrix} \tag{2-16}$$

The strength, also referred to as variance, of the constraints is usually described in the covariance matrix $\mathbf{C}_R$ and carries nonzero entries in the same locations as $\mathbf{R}$ [Viezzoli et al., 2008]. The balance between over-fitting the data, or over-smoothing the model, respectively, lies in an appropriate choice of constraint weights (throughout this project referred to as "cWeights", or in short cW). In addition to lateral constraints, vertical constraints can be imposed on the inversion problem, yielding a minimum-structure model [Auken and Christiansen, 2004]. The entire data set including constraints is inverted at the same time, requiring one common objective function Q to be minimized [Wisén and Christiansen, 2005]:

$$\text{Objective function} \qquad Q = \left(\frac{1}{N+A}[(\delta\mathbf{d\prime}^T\mathbf{C\prime}^{-1}\delta\mathbf{d\prime})]\right)^{\frac{1}{2}}. \tag{2-17}$$

Thereby, A and N term the number of constraints and data, respectively. The quantity of output models depends on the number of 1D measurements utilized in the inversion algorithm [Wisén and Christiansen, 2005]. The inversion is performed iteratively, as generally practiced for non-linear inverse problems [Fiandaca et al., 2012]. A global Gauss-Newton scheme is appropriate to minimize the objective function Q [Costabel et al., 2016]. Other techniques exist such as the conjugate gradient methodology that can equally address the minimization problem [Tartaras and Beamish, 2006].

The generated inversion models are determined by the acquired data, and further influenced by underlying physics of the applied method, implemented constraints and possibly a priori information [Siemon et al., 2009]. Model parameters are controlled by the constraints, if they are poorly described by the data and vice versa [Christiansen et al., 2007]. The

smoothness term is weighted by a regularization parameter to fit the data to a predefined noise level [Costabel et al., 2016].

Many authors have proven a successful application of LCI algorithms, enhancing the resolution of inverted models. Accurate quasi-2D images can be recovered, efficiently suppressing noise and 3D effects. The LCI approach was found robust to different starting models, while showcasing a fast inversion of large data sets [Auken et al., 2005b, 2008; Costabel et al., 2016; Siemon et al., 2009; Tartaras and Beamish, 2006]. In environments, where abrupt lateral changes in the subsurface, and thus in geophysical properties are not suspected, 1D-LCI models provide meaningful results. Layered earth structures are favoured, leveraging on a first-order smoothness-constrained regularization approach [Costabel et al., 2016; Siemon et al., 2009; Wisén et al., 2005]. On the contrary, a 2D minimum-structure inversion, achieved by adding vertical constraints, generates models with spatially smooth transitions, complicating the reconstruction of formation boundaries. Nonetheless, minimum-structure solutions can be advantageous, when no background knowledge of the subsurface is available, or the subsurface is known to be not quasi-layered [Auken and Christiansen, 2004]. So far, LCI has been employed for both ground and airborne geophysical methods, however was mostly focused on resistivity [Auken et al., 2005b] and electromagnetic [Auken et al., 2005a; Schamper et al., 2012; Siemon et al., 2009] data sets.

2-3-3 Spatially constrained inversion

A spatially constrained inversion (SCI) provides the framework to produce quasi-3D models based on 1D forward solutions [Viezzoli et al., 2008]. Similarly to the LCI approach, model parameters are constrained to adjacent soundings in order to generate smooth models with sharp layer boundaries. The difference between LCI and SCI lies in the distribution of constrained observation points. Whereas LCI algorithms constrain 1D models along a profile, spatial constraints operate in all directions, i.e., both along and across profiles [Viezzoli et al., 2008].

The challenge lies in conceptualising a mesh that links the nodes to be constrained. The connections between measurement points must be repeatable, and should ideally adjust automatically to variations in the sampling density [Viezzoli et al., 2008]. Monteiro Santos et al. [2011] use a tridimensional mesh of hexahedral blocks. Alternatively, this can be achieved by imposing a Delaunay triangulation, which adapts to the spatial distribution of observations [Viezzoli et al., 2007; Von Papen et al., 2013]. As a result, triangles are small and frequent in dense sampled areas, and vice versa, displayed by an example in Figure 2-5. The number of links between points is chosen based on the density and distribution of the data [Viezzoli et al., 2007]. The points at the vertices of generated triangles constitute nearest neighbors. Subsequently, the constraints can be set to those neighboring points, simulated by the Delaunay triangulation. The information from model parameters then migrates horizontally through the Delaunay grid, influencing parameters in regions of low resolution, hence laterally smoothing the quasi-3D image, as shown in Figure 2-6. Both primary and secondary model parameters can be incorporated in the roughness matrix $\mathbf{R}$. It differs from the matrix used in LCI mainly in the entries (cf. Equation 2-16), since SCI influences offline soundings in addition to those along flight lines or profiles, respectively [Viezzoli et al., 2008]. Because

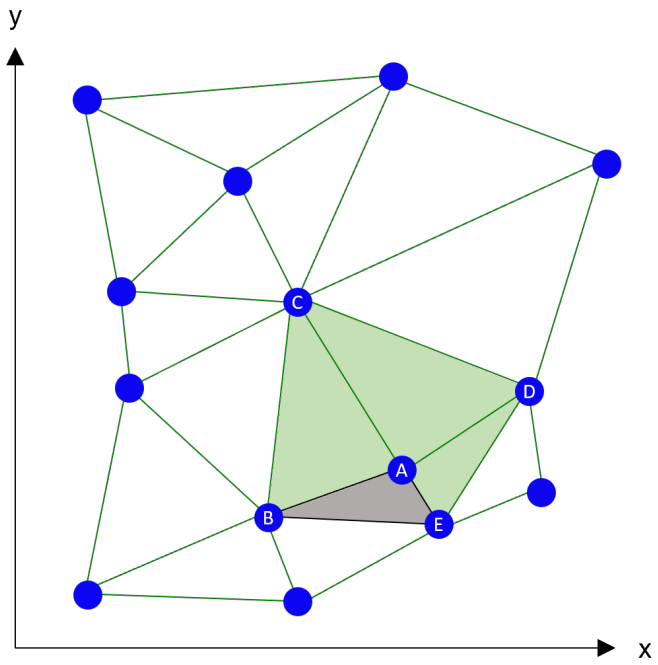


Figure 2-5: Delaunay triangulations of a randomly generated set of observation points on a plane, with primary model parameters (i.e., thicknesses t and resistivities ρ, as illustrated in Figure 2-4) of point A being constrained to model parameters of points B to E. The spatial constraints of triangle B-A-E are visualized in a three-dimensional view in Figure 2-6 (modified after Viezzoli et al. [2007], Viezzoli et al. [2008], and Viezzoli et al. [2009]).

one model parameter is constrained to all nearest neighbors defined by the mesh, the matrix takes on the following form:

$$\mathbf{R} = \begin{bmatrix} N^1 & 0 & \dots & 0 & -1 & 0 & \dots & 0 & -1 & 0 & \dots & 0 & 0 & 0 \\ 0 & N^2 & 0 & \dots & 0 & -1 & 0 & \dots & 0 & -1 & 0 & \dots & 0 & 0 \\ \vdots & & & & & & & \vdots & & & & \vdots & & \\ 0 & 0 & 0 & \dots & 0 & N^j & 0 & \dots & 0 & -1 & 0 & \dots & 0 & -1 \end{bmatrix}, \tag{2-18}$$

where $\mathbf{N}^j$ defines the number of nodes that the j^{th} model parameter is constrained to. The associated covariance matrix $\mathbf{C}_R$ carries the constraint weights (i.e., strength of the constraints), which can be calculated with the formula

$$\text{Constraint weight} \qquad C(d) = 1 + (A - 1)\left(\frac{d}{B}\right)^a, \tag{2-19}$$

as introduced by Viezzoli et al. [2007, 2008]. The weight C is thereby determined depending on the distance d between two constraining soundings. Furthermore, the average distance between observation points, termed reference distance B, the reference constraint value A, and the weighting factor a have to be taken into account. The reference constraint value A sets the weight of the constraints, if the distance between interconnected points is less

than the reference distance B. A weighting factor $a = 1$ implies that spatial constraints are linearly dependent on the distance. The formula ensures that spatial constraints are scaled according to the horizontal distance between model parameters, and thus attenuate with increasing separation, which corresponds to a reduced data density. Besides spatial constraints that exhibit spatially coherent models, vertical constraints can be incorporated between different layers [Von Papen et al., 2013].

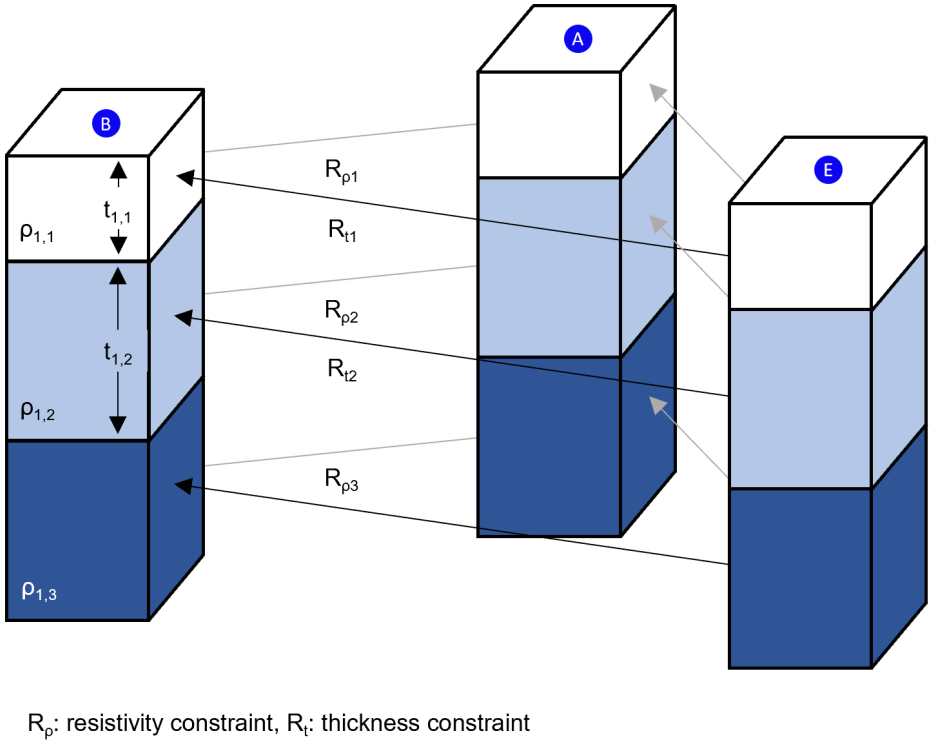


Figure 2-6: SCI model concept, where spatial constraints (denoted by R_ρ and R_t, respectively) connect primary model parameters, i.e., thicknesses t and resistivities ρ, on an exemplary Delaunay triangle, as showcased in Figure 2-5.

The mathematical formulation of the SCI setup resembles that of the 1D-LCI method by Auken and Christiansen [2004], introduced in Chapter 2-3-2. Generally, the framework can be readily implemented for any data set as long as a 1D forward solution is applicable [Viezzoli et al., 2007]. Since the inversion problem is based on a non-linear relationship between model response and parameters, an iterative procedure is necessary to minimize the common objective function. Data sets, models, and spatial constraints are inverted simultaneously, as outlined in Equation 2-17 [Monteiro Santos et al., 2011; Viezzoli et al., 2008].

The SCI approach has been previously used to derive pseudo-3D models of large survey areas [Ruiz-Aguilar et al., 2018; Viezzoli et al., 2007, 2008, 2009]. SCI provides a reasonable alternative to a full 3D inversion, maintaining computational efficiency while exploiting the entirety of available data sets [Ruiz-Aguilar et al., 2018; Vignoli et al., 2013]. The triangulated constraint mesh additionally offers the possibility to save computational costs by inverting data subsets in a parallel fashion [Viezzoli et al., 2007]. As LCI methods

only allow model parameters to vary minimally along parallel lines, subsurface features that are aligned perpendicular to the survey lines are not emphasized in pseudo-2D images. Thus, many authors stress that profile-oriented systems potentially highlight geological structures following the direction of data acquisition, i.e., the flight (or profile) direction in case of airborne geophysical techniques. This limitation can be overcome when using SCI [Viezzoli et al., 2007, 2008, 2009].

2-3-4 Assessment of inversion models

The agreement of inversion models, in particular quasi-3D models displaying complex structures, with the true model is difficult to be visually estimated. Quantitative assessment of the model misfit can be obtained by calculating the structural similarity index and the mean squared error, based on the open-source image processing library *scikit-image* [Van der Walt et al., 2014]. The reliability of models generated from field data sets can be evaluated by analyzing the depth of investigation.

Structural similarity

The structural similarity (SSIM) index is a measure for calculating the similarity between two images. Thereby, image degradation is assessed qualitatively as SSIM provides a perceived visual quality metric that considers structure, luminance, and contrast of respective images [Wang et al., 2004]. The SSIM index between signals $\mathbf{x}$ and $\mathbf{y}$ is given by

$$\text{Structural similarity index} \qquad SSIM(\mathbf{x},\mathbf{y}) = [l(\mathbf{x},\mathbf{y})]^{\alpha} \cdot [c(\mathbf{x},\mathbf{y})]^{\beta} \cdot [s(\mathbf{x},\mathbf{y})]^{\gamma}, \tag{2-20}$$

where l is the luminance comparison function, c is the contrast comparison function, s is the structure comparison function, and α, β, and γ are parameters used to adjust the importance of mentioned components. A $SSIM = 1$ indicates that the compared image is identical to the original image (i.e., true subsurface model). The method introduced by Wang et al. [2004] is classified as a full-reference approach, requiring that the original image is known and free of distortions. This implies that the SSIM coefficient cannot be used as an evaluation tool for a field data example.

Mean squared error

The mean squared error (MSE) between the inverted and true model constitutes a widely applied and simple full-reference quality metric, that estimates the strength of the error. Computationally, the MSE is implemented by averaging the squared differences between distorted and reference images, i.e., the inverted and true model in the context of geophysical inversion [Wang et al., 2004]. The MSE between signals $\mathbf{x}$ and $\mathbf{y}$, where N denotes the signal samples and x_i and y_i are the i^{th} samples in $\mathbf{x}$ and $\mathbf{y}$, respectively, is calculated according to [Wang and Bovik, 2009]:

$$\text{Mean squared error} \qquad MSE(\mathbf{x},\mathbf{y}) = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2. \tag{2-21}$$

The MSE does not evaluate the perceived similarity of compared images, yet can complement the quantitative assessment of inverted models [Wang et al., 2004].

Depth of investigation

The inversion of electromagnetic data is highly non-unique and the true depth of investigation (DOI) can usually not be inferred from the inversion results [Oldenburg and Li, 1999]. Thus, inverted models reproducing the measured electromagnetic data have a different validity depending on data noise and subsurface layer resistivities [Vest Christiansen and Auken, 2012].

To evaluate the credibility of generated inversion models from field data, a DOI analysis is useful. The DOI refers to the depth below which the geophysical data does not show any sensitivity to subsurface geophysical properties, crucial for interpretation purposes. The calculation of the DOI following the concept presented by Oldenburg and Li [1999] compares inversion results $\mathbf{m}_1$ and $\mathbf{m}_2$ from different reference models $\mathbf{m}_{1r}$ and $\mathbf{m}_{2r}$, respectively, according to:

$$\text{Depth of investigation index} \qquad R_{DOI}(x,z) = \frac{\mathbf{m}_1(x,z) - \mathbf{m}_2(x,z)}{\mathbf{m}_{1r} - \mathbf{m}_{2r}}. \tag{2-22}$$

Applied reference models $\mathbf{m}_{1r}$ and $\mathbf{m}_{2r}$ are defined as half-space models with constant background resistivities above and below the resistivity of the starting model used in the inversion algorithm. If the DOI index R_{DOI} approaches 0, corresponding areas are assumed to be highly reliable, whereas locations with values close to 1 show a poor data constraint on the model, and consequently low credibility.

Chapter 3

Inversion algorithm

The computational algorithm of this Master thesis project is based on the open-source library *pyGIMLi* (Geophysical Inversion & Modelling Library) [Rücker et al., 2017] within a Python environment. The class developed by Günther [2013] for modeling and inversion of FDEM data was previously extended to incorporate data from multiple coil spacings [Thalhammer, 2022], and further adapted to offer both smooth and blocky inversion methods (Figure 1-1). As part of this Master thesis, quasi-2D and quasi-3D modeling classes were built by applying lateral and spatial constraints (cf. Appendix A-1 and A-2), motivated by the possibility to accurately delineate the active layer thickness.

3-1 Overview of the code algorithm

A flowchart of the code structure for the synthetic and field data set is illustrated in Figure 3-1 and 3-2, respectively. In the synthetic case, the inversion procedure consists of the following steps (Figure 3-1):

1. the data input, including the setup of various synthetic subsurface models, characterized by varying snow thickness, varying snow resistivity, as well as a more complex 3D subsurface model, along with the definition of the survey parameters (cf. Chapter 4-1-1 and 4-2-1),
2. a numerical 1D forward modeling scheme for both a single-component (smooth) and a few-layer (blocky) inversion, with synthetic data generation from previously defined synthetic models (cf. Chapter 3-2),
3. an inversion framework utilizing either lateral or spatial constraints between 1D forward solutions to evaluate the degree of fit between synthetic and predicted data based on a constant starting model (cf. Chapter 3-3),
4. the implementation of a weighted roughness matrix (cf. Chapter 3-3-1) to regularize the inverse problem (cf. Chapter 3-3-2), which is normalized for the blocky inversion case to ensure stability (cf. Chapter 3-3-3),

5. the visualization of generated models (cf. Chapter 4) and their quality assessment by calculating structural similarity (SSIM) and mean squared error (MSE) coefficients (cf. Chapter 2-3-4).

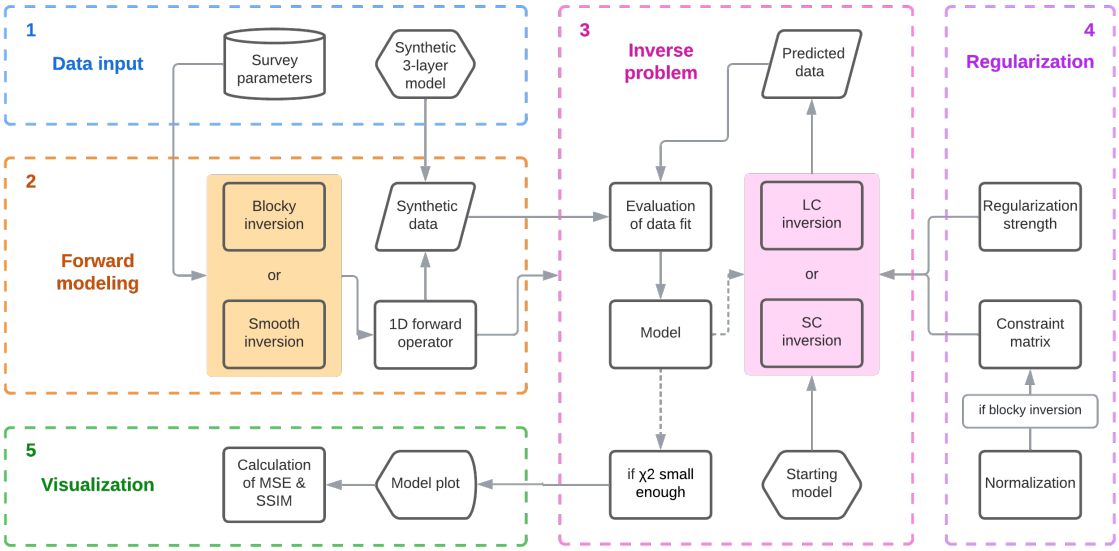


Figure 3-1: Flowchart representing the code structure for the synthetic data set based on the general concept of forward modeling and inverse problems. LCI and SCI were performed for 1D forward responses corresponding to a few-layer (i.e., blocky inversion) and smooth (i.e., smooth inversion) model, respectively.

The computational framework for the electromagnetic field data, as displayed in Figure 3-2, is comprised of:

1. the input of the acquired data from the field site in Teller, Alaska, as well as an explanation of acquisition setup and data processing (cf. Chapter 5-2 and 5-3),

2. a numerical 1D forward modeling focused on Occam's smooth inversion (cf. Chapter 3-2),

3. an inversion framework, which implements lateral and spatial constraints between 1D forward solutions to evaluate the degree of fit between measured and predicted data based on a constant starting model (cf. Chapter 3-3),

4. the formulation of a weighted roughness matrix (cf. Chapter 3-3-1) to regularize the inverse problem (cf. Chapter 3-3-2),

5. the visualization of inverted subsurface models and subsequent interpretation together with an assessment of the depth of investigation (cf. Chapter 2-3-4), consideration of measured snow layer thickness, and comparison to conventional 1D inversion models (cf. Chapter 5).

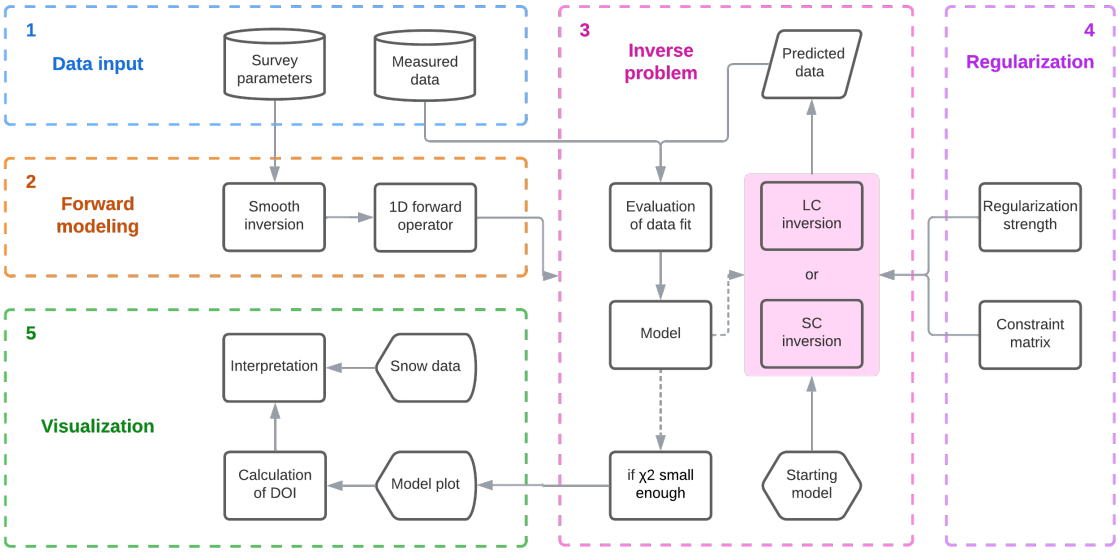


Figure 3-2: Flowchart representing the code structure for the field data set based on the general concept of forward modeling and inverse problems. LCI and SCI were performed for 1D forward responses corresponding to a smooth (i.e., smooth inversion) model.

3-2 Forward modeling

Frameworks for LCI and SCI require the application of a 1D forward solution. Secondary electromagnetic fields measured during electromagnetic experiments are complex, yet can be simplified if the LIN principle holds. However, for the underlying algorithm the solution is based on the derivations presented by Ward and Hohmann [1988], in which no approximations are made, and full Maxwell equations are exploited in the forward modeling scheme [Frederiksen et al., 2017]. According to Siemon [2009], the secondary magnetic field is expressed by a complex reflection factor and Bessel functions. During the forward response complex integrals need to be solved, which can be implemented with fast Hankel transformations. Two different forward operators are used in the regularized inversion methods, a blocky and a smooth model, respectively (cf. Chapter 2-3-1), as outlined in Figure 3-1 and 3-2.

3-3 Inversion

The inversion of electromagnetic data is non-linear, and thus an iterative procedure is required that applies a 1D forward modeling routine [Smith et al., 1999]. The inverse problem is solved by minimizing the objective function, as presented in Equation 2-8, composed of the data misfit term (cf. Equation 2-9) and the model objective function (cf. Equation 2-10), scaled by the regularization parameter λ. To include the lateral and spatial constraints, the model objective function, as introduced in Equation 2-10, has to be adapted:

$$\text{Model objective function} \qquad \Phi_M = \|\mathbf{R}\mathbf{C}_R(\mathbf{m} - \mathbf{m}_r)\|_2. \tag{3-1}$$

The roughness matrix $\mathbf{R}$ (Equations 2-16 and 2-18) holds the constraints, and corresponding covariance matrix $\mathbf{C}_R$ the respective constraint weights (cf. Chapter 2-3-2 and 2-3-3). The objective function is then expressed as

$$\text{Objective function} \qquad \Phi(\mathbf{m}) = \|\mathbf{W}_D(f(\mathbf{m}) - \mathbf{d})\|_2^2 + \lambda\|\mathbf{R}\mathbf{C}_R(\mathbf{m} - \mathbf{m}_r)\|_2^2 \rightarrow min \tag{3-2}$$

and minimized in a Gauss-Newton scheme with global regularization [Günther et al., 2006]. In the inversion algorithm, the model parameters $\mathbf{m}$, i.e., resistivities and thicknesses, are represented as logarithms to impose positivity and minimize non-linearity. Additionally, the resistivity values are kept in a pre-defined range using logarithmic barriers [Günther and Rücker, 2010]. The model vector $\mathbf{m}$ is updated iteratively in accordance with

$$\mathbf{m}^{k+1} = \mathbf{m}^k + \tau^k \Delta\mathbf{m}^k, \tag{3-3}$$

utilizing a line search parameter τ, whereby the superscript k denotes the iteration number. The respective Jacobian matrix $\mathbf{G}$ is dense and the iterative scheme is computationally expensive. As a consequence, $\mathbf{G}$ is not calculated explicitly, and the solution is approximated with a conjugate-gradient method [Günther et al., 2006; Günther and Rücker, 2010].

3-3-1 Constraint matrix

The regularization of the under-determined inverse problem was realized by incorporating lateral and spatial constraints. Suitable roughness matrices are defined and weighted according to the underlying 1D forward model. The introduction of blocky and smooth models, constrained both laterally and spatially, requires the formulation of different roughness operators. To ensure operational efficiency the constraints were computed as sparse matrices. While the smooth inversion only constraints one model parameter, the blocky inversion deals with primary model parameters, i.e., thicknesses and resistivities. A summary of the regularization methods applied in both cases is given in Table 3-1.

Table 3-1: Overview of the implemented constraint weights (cW) for model parameters associated with blocky and smooth inversion techniques for LCI and SCI methodologies. Blocky models are laterally smoothed, while smooth models showcase smoothing in both the lateral and vertical direction.

Inversion	Model parameter	Boundaries		Regularization
		Horizontal	Vertical	
Blocky	Thickness	$cW = 0$	Thickness cW*	Lateral smoothing
	Resistivity	$cW = 0$	Resistivity cW*	Lateral smoothing
Smooth	Resistivity	Horizontal cW	Vertical cW*	Vertical/lateral smoothing

*Distance-dependent within SCI framework

Laterally constrained inversion

In the few-layer model lateral smoothing is enforced across vertical mesh boundaries in both model parameter regions. Simultaneously, vertical smoothing is hindered by structurally decoupling known layer interfaces [Wagner and Uhlemann, 2021]. This is achieved by completely removing the smoothness constraints (i.e., horizontal cWeights) across horizontal boundaries, as illustrated in Figure 3-3. The weighted constraint matrix has the dimensions $b \times M$, where b denotes the number of inner mesh boundaries between model parameters, and M describes the number of model parameters. The matrix is computed separately for different model parameter regions, as illustrated by different colors in Figure 3-3. The variance, i.e., the constraint weight, imposed on the roughening matrices can take on various values, in particular for different model parameters, and is usually determined by trial and error (cf. Chapter 4).

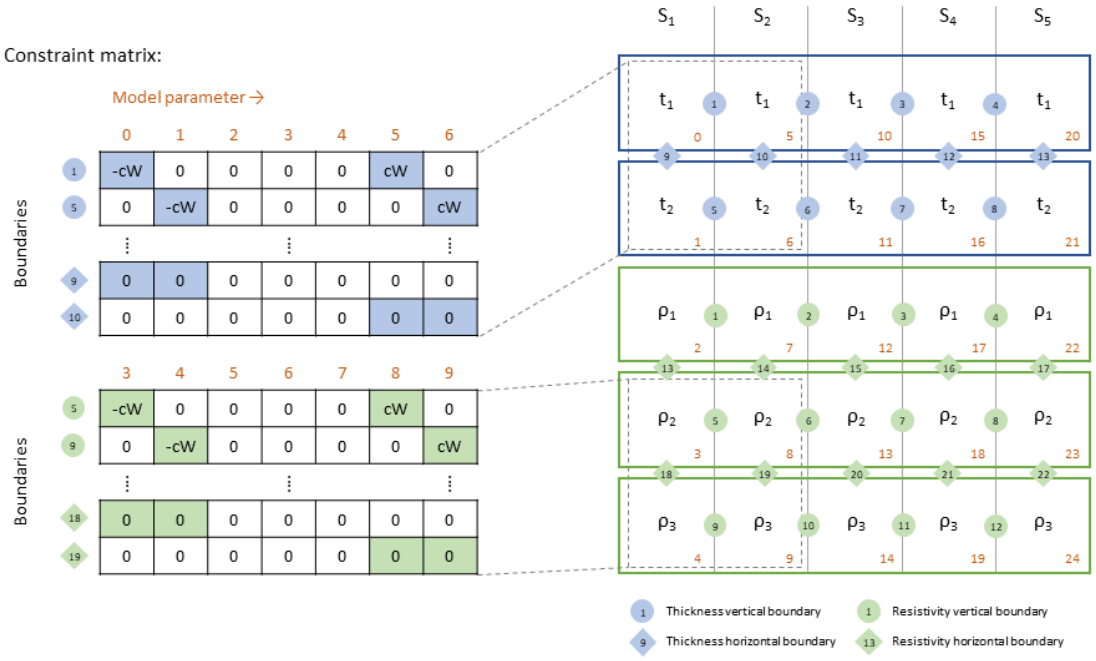


Figure 3-3: Three-layered blocky model based on five laterally constrained soundings S and associated weighted roughness matrices for subsets of the model parameter regions, i.e., thicknesses t and resistivities ρ, respectively. Vertical boundaries enforce lateral smoothing by choosing appropriate vertical constraint weights (cW). Horizontal boundaries are set to 0 to structurally decouple the corresponding interfaces.

The smooth inversion process constitutes an Occam type inversion. Model parameters are constrained both vertically and horizontally, resulting in a minimum-structure model [Auken and Christiansen, 2004]. A single constraint matrix is formulated to constraint resistivity values to those of adjacent soundings. Figure 3-4 displays the constraint matrix for a subset of six model parameters. A different constraint strength can be assigned to horizontal and vertical boundaries, or even to different subsets of the model. The effects of the variance on the inversion are analyzed in the synthetic study (cf. Chapter 4).

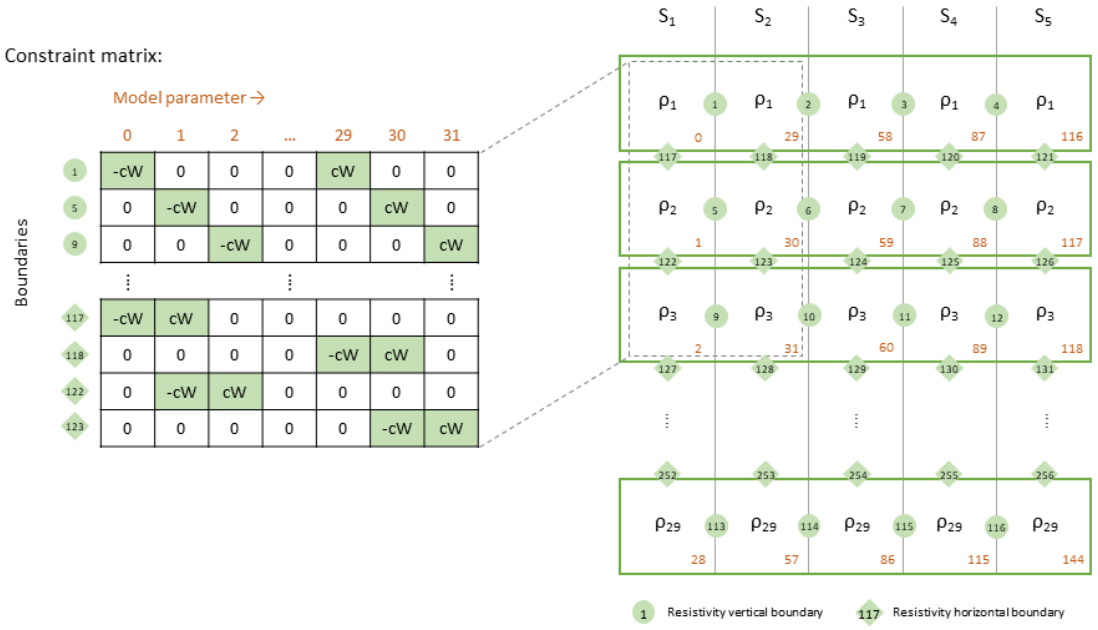


Figure 3-4: Thin-layered smooth model based on five laterally constrained soundings S and associated weighted roughness matrix for a subset of the model parameter region, i.e. resistivity ρ. Vertical boundaries and horizontal boundaries enforce lateral and vertical smoothing, respectively, by choosing appropriate vertical or horizontal constraint weights (cW).

Spatially constrained inversion

The SCI algorithm produces quasi-3D models by constraining 1D forward models in all directions, i.e., across and along profile lines. The setup of the mesh, as well as the constraint and covariance matrices are carried out as proposed by Viezzoli et al. [2008]. In a first step, connections between adjacent soundings are found by triangulating the measurement locations. The triangulation is performed with the 2D version of the Delaunay tessellation algorithm (cf. Chapter A-3), leveraging on the *Qhull* library [Barber et al., 1996]. Soundings to be constrained are determined based on vertices of the computed triangles (cf. Chapter 2-3-3). The horizontal boundaries are assembled according to the LCI approach in the roughness matrix, both for a smooth and blocky inversion (compare Figure 3-3 and 3-5). Therefore, the model parameters of each sounding are either vertically smoothed (smooth inversion) or structurally decoupled (blocky inversion), as explained in Table 3-1. The spatial constraints ensure lateral smoothing across vertical boundaries between adjacent model parameters, predefined by the Delaunay triangulation. Figure 3-5 gives an example of a sounding S_A that is tied to the other vertices of the triangle by constraints. The roughness matrix is implemented in a similar manner for Occam type 1D forward models.

As described in Chapter 2-3-3, the constraint weights are defined in the covariance matrix $\mathbf{C}_R$, and are distance-dependent to adequately reflect the distribution of soundings. The strength of the constraint is calculated based on the Euclidean distance between nearest neighbors, as

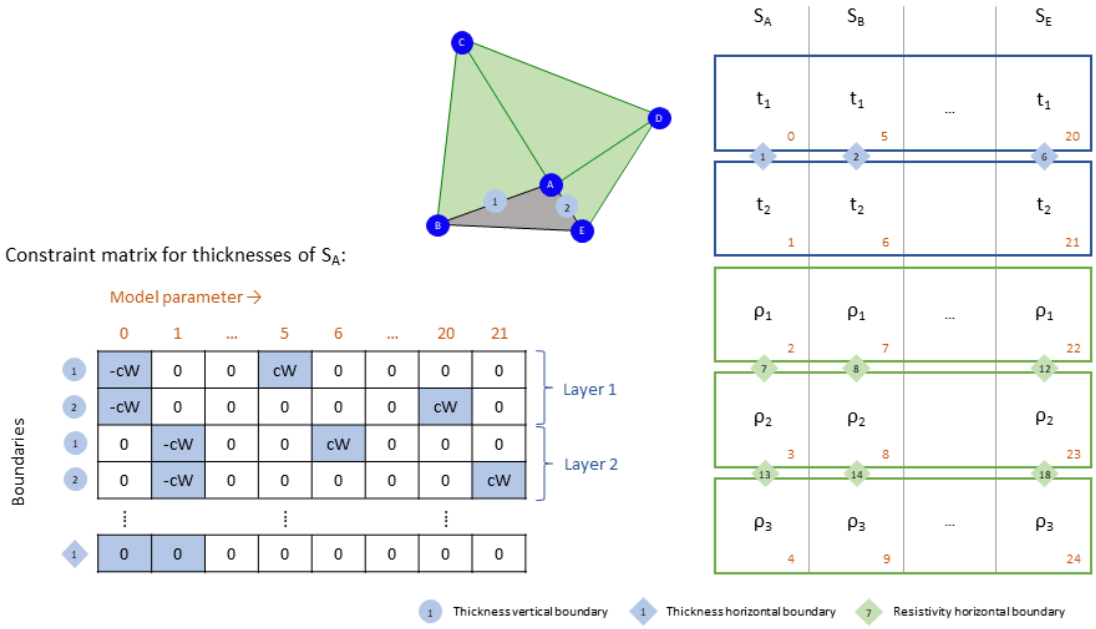


Figure 3-5: Three-layered blocky model based on five spatially constrained soundings S connected by Delaunay triangulation and associated weighted roughness matrix for sounding S_A in the triangle B-A-E, focused on the thickness model parameter t. Vertical boundaries enforce lateral smoothing between all neighboring soundings by choosing appropriate vertical constraint weights (cW). Horizontal boundaries are set to 0 to structurally decouple the corresponding interfaces.

described in Equation 2-19. The impacts of the spatial constraints with respect to the parameters used in the formula, i.e., reference constraint value A and exponent a, are outlined in Chapter 4.

3-3-2 Regularization strength

Varying the regularization strength leads to different solutions of the inverse problem, and among other factors determines the convergence behavior during the minimization of the objective function. This explains why the regularization parameter needs to be chosen carefully. The maximum curvature of the L-curve suggests the optimal regularization parameter that offers a compromise between data misfit and model structure and generates a smooth model that explains the data within the error bounds (cf. Chapter 2-3).

In the synthetic study, the L-curve framework was applied to visually estimate suitable regularization parameters. This was proven to be successful within the LCI scheme, in particular for single-component inversions, as shown in Figure 3-6b. For few-layer inversion models it was difficult to assess the regularization parameter based on the L-curve criterion (Figure 3-6a). Thus, a hyperparameter test was conducted to ensure an adequate choice of regularization strength. The regularization parameter for the SCI as well as the field data

set was deduced from the L-curves computed within the LCI algorithm based on the same constraint weights.

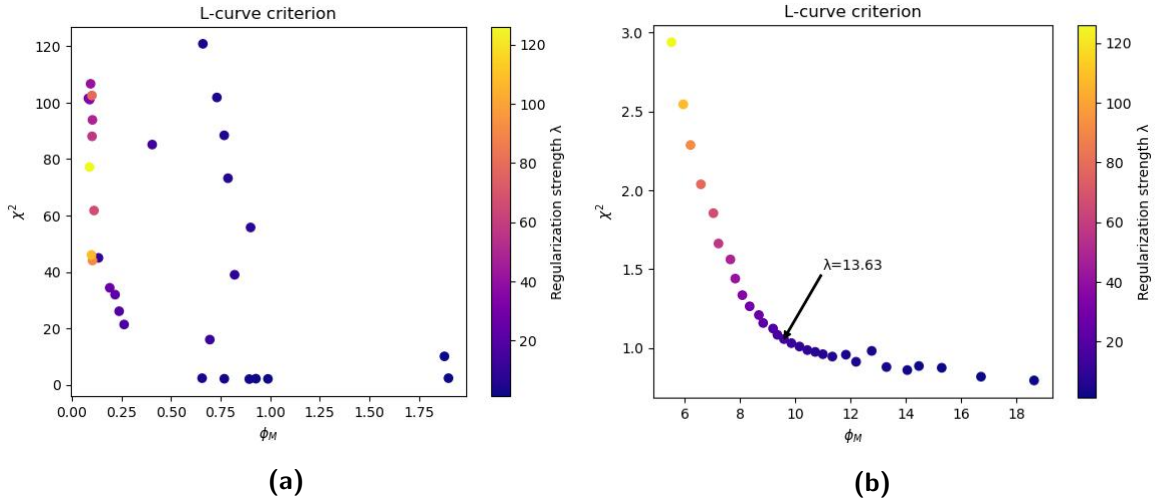


Figure 3-6: L-curve criterion for a laterally constrained blocky (a) and smooth (b) inversion framework applied to a synthetic permafrost subsurface model with varying snow thickness (a) or resistivity (b) by adapting the regularization parameter λ within a logarithmic interval of [0.1, 2.1]. The optimal regularization parameter was found at the point of maximum curvature for $\lambda = 13.63$ by visual inspection (b).

3-3-3 Normalization

The developed quasi-2D and quasi-3D inversion algorithms are constrained by implementing a weighted constraint matrix, which is further influenced by the chosen regularization parameter. In the blocky case, the lateral smoothing is controlled by vertical constraint weights for both model parameters, i.e., thicknesses and resistivities. In contrast, in the smooth case, resistivities are not only constrained laterally, but also vertically by introducing lateral constraint weights (Table 3-1).

In the synthetic study it was discovered that the regularization strength applied varies strongly with the choice of constraint weights, even if the ratio of implemented weights as well as the regularization parameter remain unchanged. The observation was made by analyzing the model objective function Φ_M prior to the inversions process. As shown in Figure 3-7, the model objective function Φ_M yields a different value depending on the chosen constraint weights. The showcased example is based on a three-layered blocky inversion model stitching together 11 soundings, which were constrained laterally. The associated roughening matrix consists of 50 rows corresponding to 20 and 30 constraints between thicknesses and resistivities, respectively. Figure 3-7 and complementary plots in Appendix A (Figure B-1) demonstrate that the model objective function Φ_M takes on higher values if the strength of the constraints increases. Additionally, a higher number of boundaries b between model parameters, linked to the size of the respective part of the constraint matrix, indicates a rise of Φ_M.

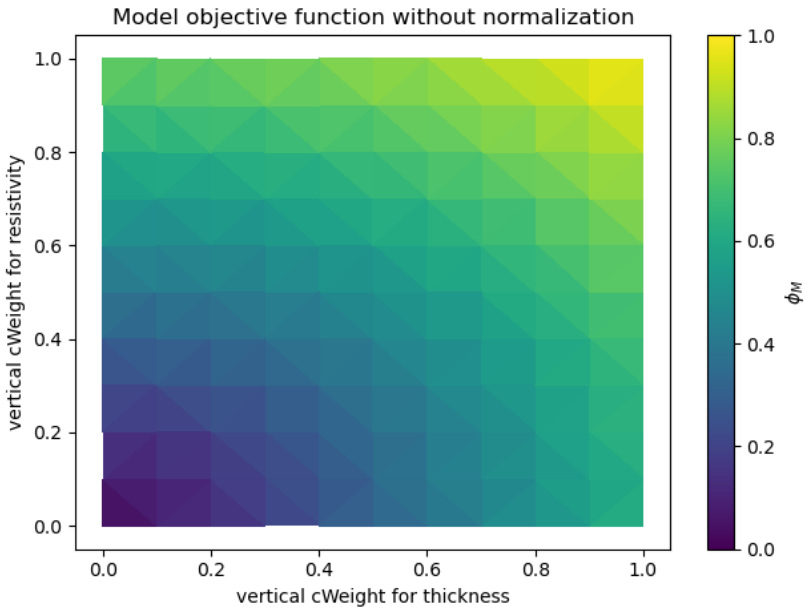


Figure 3-7: Model objective function Φ_M as a function of constraint weights for both model parameters (i.e., thicknesses and resistivities) based on a three-layered blocky model with 11 laterally constrained soundings. Φ_M values are displayed relative to the highest Φ_M when corresponding cWeights = 1.

A normalization procedure was designed to account for the aforementioned effect in order to ensure an equal regularization if the ratio between applied constraint weights is not modified. As a consequence, the amount of regularization is solely subject to the regularization strength. The normalization is implemented with regard to the ratio and value of the used constraint weights as well as the number of constraints between two model parameters, i.e., the number of inner mesh boundaries b between model parameters. The model objective function Φ_M is calculated based on a linear relationship and outlined in Table B-1 in Appendix B. The respective values are used to approximate the model objective function for any combination of constraint weights employed in the inversion algorithm. Depending on the ratio between used constraint weights six different cases were formulated to compute the normalization factor (referred to as norm), which is directly applied to the weighted constraint matrix (Appendix B: Table B-2).

The normalization factor was tested for synthetic data sets in 2D and applied to all blocky models. For smooth models, the normalization is redundant, due to inherent stability of minimum-structure models, leveraging on the l_2-norm. The approach will be further discussed in Chapter 6.

Chapter 4

Application to synthetic data

To validate the developed LCI and SCI algorithm, a synthetic case study was conducted. The synthetic data generation is explained hereafter and applied to demonstrate the capability of regularized inversion procedures for solving the under-determined electromagnetic inverse problem. The effect of different constraint weights on LCI models is outlined, while the SCI study is focused on the evaluation of distance-dependent constraint weights. Moreover, generated inversion results are compared to conventional 1D inversions with respect to the true subsurface model. Quantitative measures are used to study the quality of inverted models. Finally, the results shown here are used to decide on suitable inversion parameters and constraint weights for the field data example in Chapter 5.

4-1 Laterally constrained inversion

Synthetic quasi-2D models were produced by implementing lateral constraints for a smooth as well as a blocky model. As pointed out in Figure 3-1, synthetic subsurface models are required to generate data for the inversion scheme. Hereafter, the synthetic model setup is explained and corresponding LCI results are presented based on two different subsurface models.

4-1-1 Synthetic model setup and data generation

The synthetic model of the subsurface, referred to as the true model, consists of three layers and is representative for a typical permafrost soil during the winter season (Figure 2-1). The unfrozen active layer is covered by snow and underlain by permafrost. Previous permafrost studies in Alaska provide reasonable estimates for associated layer dimensions and resistivities, or conductivities, respectively [Douglas et al., 2021; Hubbard et al., 2013; Minsley et al., 2012; Uhlemann et al., 2021].

Two synthetic permafrost models were built in order to generate data for the inversion process (Figure 4-1). The computed models are characterized by an inhomogeneous first snow layer with respect to its thickness (a) or resistivity (b). They reflect conditions in nature, as changes in electrical properties of snow are linked to the inherent water content, and snow thickness varies depending on the topography and vegetation distribution [Glen and Paren, 1975]. The objective of the synthetic study is to analyze the suitability of the developed inversion approaches to recover subsurface variations from FDEM data, typical of Arctic environments.

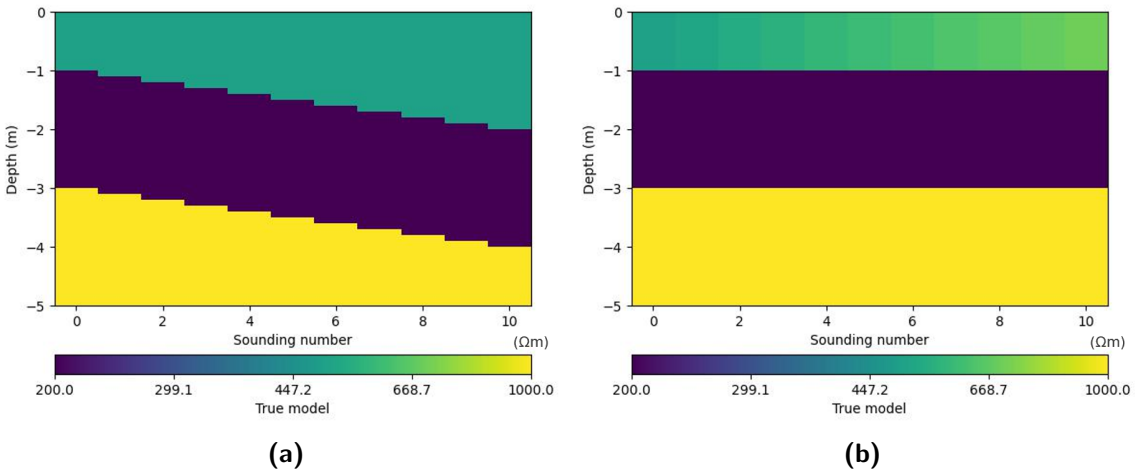


Figure 4-1: Synthetic 2D three-layered permafrost models (referred to as true models) consisting of a snow layer with varying thickness (a) or varying resistivity (b), an active layer, and the permafrost layer for a profile of 11 soundings. Corresponding parameter values are summarized in Table 4-1.

The chosen parameters for the subsurface models illustrated in Figure 4-1 are specified in Table 4-1. The inhomogeneous snow layers are marked by a varying thickness in the range from 1 to 2 m (Figure 4-1a) and resistivity in the range from 500 to 700 Ωm (Figure 4-1b), respectively. The conductive active layer has a resistivity $\rho = 200$ Ωm, whereas the permafrost soil is highly resistive with $\rho = 1{,}000$ Ωm. A noise level of 2.15 % was added to the data.

4-1-2 Inversion parameters

In the case of a few-layer inversion in which the model objective function is minimized for both model parameter regions, a homogeneous starting model with thicknesses and resistivities equal to 1.25 m and 1.25 Ωm, respectively, and a three-layer parameterization were chosen, based on a hyperparameter study conducted by Thalhammer [2022]. For the smooth inversion, i.e., the Occam type inversion, a resistivity of 580 Ωm was used for the reference model and was selected by trial and error. A suitable regularization strength was found using an L-curve approach and hyperparameter testing as outlined in Chapter 3-3-2. The 1D inversion was performed with the Levenberg-Marquardt technique for a Marquardt factor of 0.9 and a regularization strength λ that yields a $\chi^2 \leq 1$ [Thalhammer, 2022].

Table 4-1: Model parameters of the synthetic 2D three-layered permafrost model consisting of a snow layer with varying thickness (a) or resistivity (b), respectively, an active layer, and the permafrost, as shown in Figures 4-1a and 4-1b.

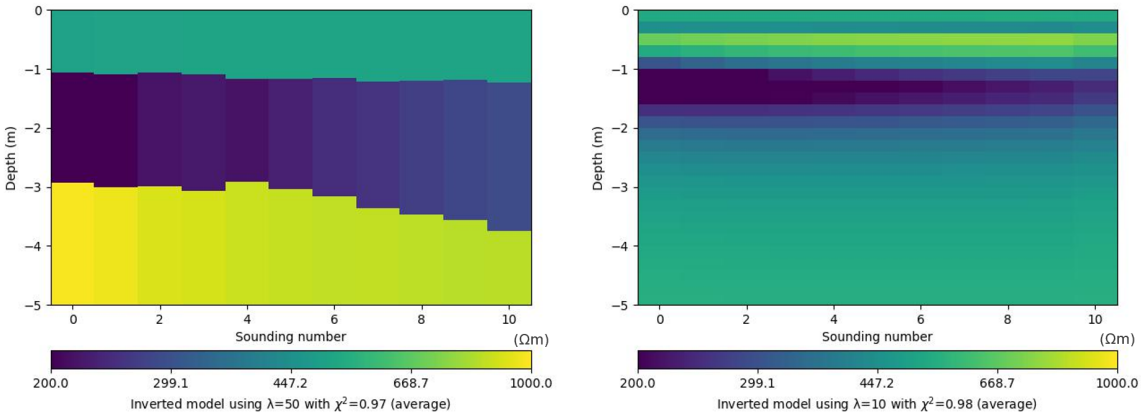
	Depth [m]	Resistivity ρ [Ωm]
(a) Model with varying snow thickness		
Snow layer	1-2	500
Active layer	2	200
Permafrost	-	1,000
(b) Model with varying snow resistivity		
Snow layer	1	500-700
Active layer	2	200
Permafrost	-	1,000

4-1-3 Results

In the following, the benefit of the regularized LCI is demonstrated in comparison with 1D inversion models generated with the Levenberg-Marquardt algorithm. Furthermore, the versatility of lateral constraints is illustrated using blocky and smooth models. The examples show the influence of different constraint weights on the inverted model.

Model with varying snow thickness

The inversion models are presented hereafter with respect to the true model characterized by a changing snow layer thickness (Figure 4-1a and Table 4-1). Figure 4-2 displays the inverted resistivity models for individually inverted soundings that are stitched together in one image for a blocky (a) and smooth (b) model. The active layer is detected in both models, yet the true dimensions cannot be inferred accurately at all observation points. While the blocky inversion model (a) indicates that a thin snow layer allows to recover the active layer thickness, a thicker snow layer leads to an overestimation of the active layer both with respect to resistivity (∼300 compared to 200 Ωm) and vertical extent (2.7 compared to 2 m). On contrary, the smooth inversion (b) produces a very smooth underground model that cannot mirror the range of the true subsurface electrical properties. Nevertheless, the generated model suggests a thicker snow layer in the right part of the profile, whereas the blocky model images a constant thickness of 1 m throughout the profile.



(a) **(b)**

Figure 4-2: Stitched-together 1D inversion results for a profile of 11 soundings plotted for a blocky (a) and a smooth model (b) based on a synthetic subsurface model with varying snow thickness, shown in Figure 4-1a.

Figure 4-3 compares the LCI models for different constraint weights based on a blocky model, normalized as explained in Chapter 3-3-3. It is clear that only a few combinations of constraints weights can retrieve the features of the true model. The inversion converges to a reasonably low χ^2 value of 2.56 if the strength of constraints for thicknesses is set to 0. Consequently, the image is produced by laterally smoothing the resistivity model parameters to those of neighboring soundings with sharp layer boundaries. Those two models almost perfectly resemble the synthetic permafrost model.

The LCI utilizing a smooth model provides images that have similar features compared to the synthetic model as characterized by a small χ^2 (Figure 4-4). The different inversion constraints clearly influence the resulting subsurface models. While a horizontal cWeight of 0 does not vertically smooth the image, a value of 1 produces models with comparably small vertical resistivity variation. A horizontal cWeight of 0.5 showcases the change in snow thickness, yet underestimates the depth of the active layer in the right part of the profile ($\sim$0.8 compared to 2 m). For the models with more vertical structure (if horizontal $cW < 1$) an artifact can be observed close to the surface at a depth of 0.2 m. Additionally, an increasing vertical cWeight visibly smooths the images in lateral direction.

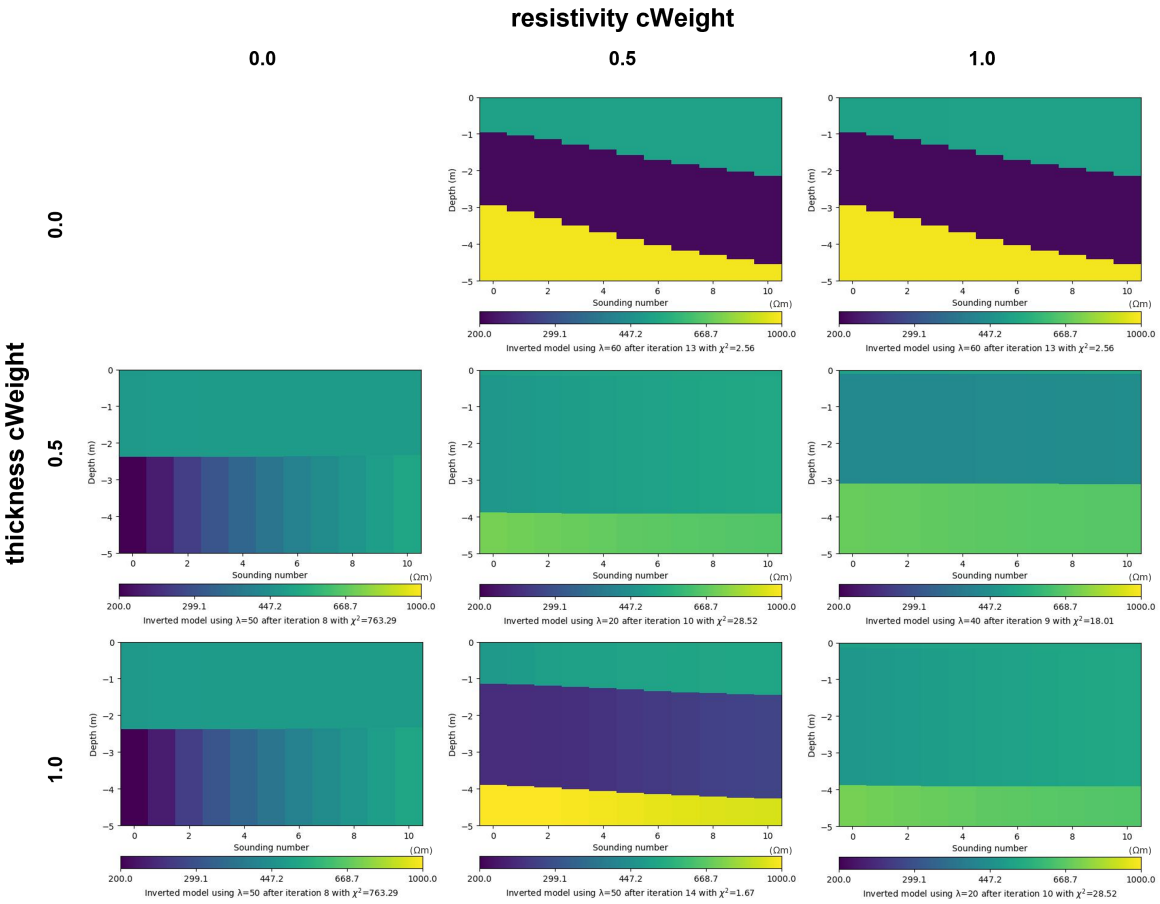


Figure 4-3: Normalized LCI results for varying constraint weights for model parameter regions (i.e., thicknesses and resistivities) which yield the smallest χ^2 for an optimal λ based on a synthetic subsurface model with varying snow thickness, shown in Figure 4-1a.

Model with varying snow resistivity

The second synthetic example exhibits varying electrical properties for the first layer (Figure 4-1b and Table 4-1). The conventional 1D inversion (i.e., no lateral or spatial constraints) constructs reasonable subsurface models by stitching together the single-site inversions with an average $\chi^2 \leq 1$ (Figure 4-5). Figure 4-5a and 4-5b illustrate the response of the blocky and smooth model, respectively. The active layer is recovered, yet not delineated precisely, when compared to the true model. The resistivity variations are accurately mapped in the blocky model (a), and transitions between electrical properties are similarly shown in the smooth model (b). A high snow resistivity causes an underestimation of the extent of the active layer (1.5 compared to 2 m) (a), or rather an overestimation of the snow layer itself (∼1.2 compared to 1 m) (b). Additionally, the smooth model is unable to retrieve the true resistivity of the permafrost layer, which is underestimated considerably (∼500 compared to 1,000 Ωm).

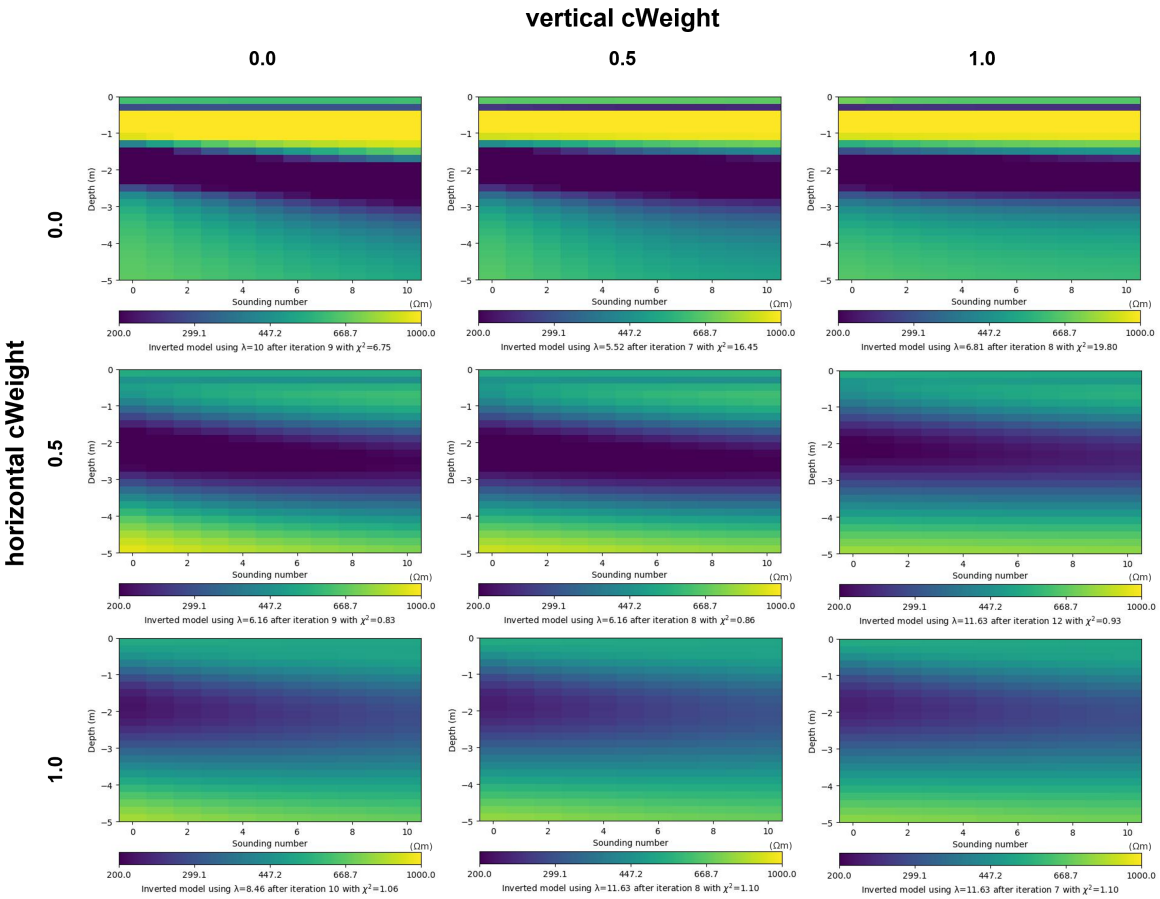


Figure 4-4: LCI results for varying constraint weights across vertical and horizontal boundaries which yield the the smallest χ^2 for an optimal λ based on a synthetic subsurface model with varying snow thickness, shown in Figure 4-1a.

Figure 4-6 gives an overview of normalized inversion models with respect to a constant few-layer starting model. All combinations of horizontal constraint weights generate a model that mimics the true subsurface model to a great extent. Hence, corresponding χ^2 values do not deviate much from 1. While all figures reproduce the resistivity variation, the thickness of the snow layer is more ambiguous. The consistency of the layer is resolved well, at least if the thickness cWeight is greater than 0, however the thickness is overestimated for all combinations of constraint weights ($\sim$1.4 compared to 1 m). Even though the electrical properties of the active layer are constant as is given for the initial model ($\rho = 200$ Ωm), the vertical dimension varies for the final inversion models ($\leq$ 1.5 compared to 2 m).

The same synthetic model was inverted for a 1D Occam type inversion algorithm constrained laterally. The resulting models which are smoothed vertically and laterally to a different degree are plotted in Figure 4-7. The algorithm produces reasonable subsurface representations in most cases, except if no regularization is applied in vertical direction, i.e., across horizontal boundaries. If the horizontal cWeight is greater than 0 the active layer is well determined, but

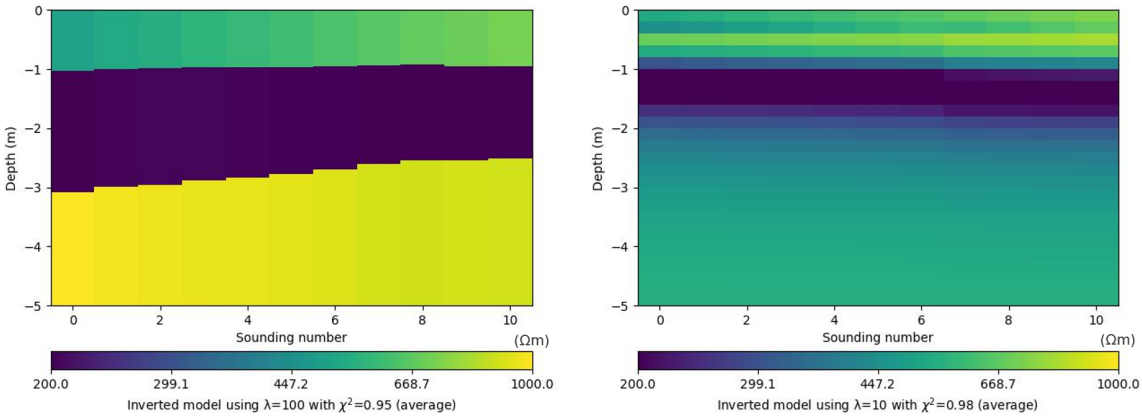


Figure 4-5: Stitched-together 1D inversion results for a profile of 11 soundings plotted for a blocky (a) and a smooth model (b) based on a synthetic subsurface model with varying snow resistivity, shown in Figure 4-1b.

also the variation in snow resistivity is clearly recovered. A horizontal cWeight of 1 creates very smooth subsurface models in which sharp boundaries are difficult to locate. In general, it can be observed, that vertical smoothing undermines the effect of vertical constraint weights for the given subsurface model. The difference between the strength of weights in lateral direction (i.e., vertical cWeights) is hard to perceive.

Deduction of inversion parameters

The illustrated LCI models (Figures 4-3, 4-4, 4-6, and 4-7) show large discrepancies depending on the chosen combination of constraint weights. Not only is the visual difference between inversion models significant, but also does the computed error criterion χ^2 imply that the selection of constraint weights has considerable importance. With respect to the models that incorporate a smooth response (Figure 4-4 and 4-7), it is visible, that vertical smoothing based on the l_2-norm is necessary, yet should be kept at a medium level (horizontal $cW = 0.5$) to avoid over-smoothing. As a consequence, the SCI study presented hereafter, is carried out with a horizontal cWeight of 0.5 in order to analyze the impact of lateral constraint based on the distance between nearest soundings (cf. Equation 2-19). A regularization parameter of $\lambda = 10$ is chosen as it is close to the point of maximum curvature taking into account all tested constraint weights.

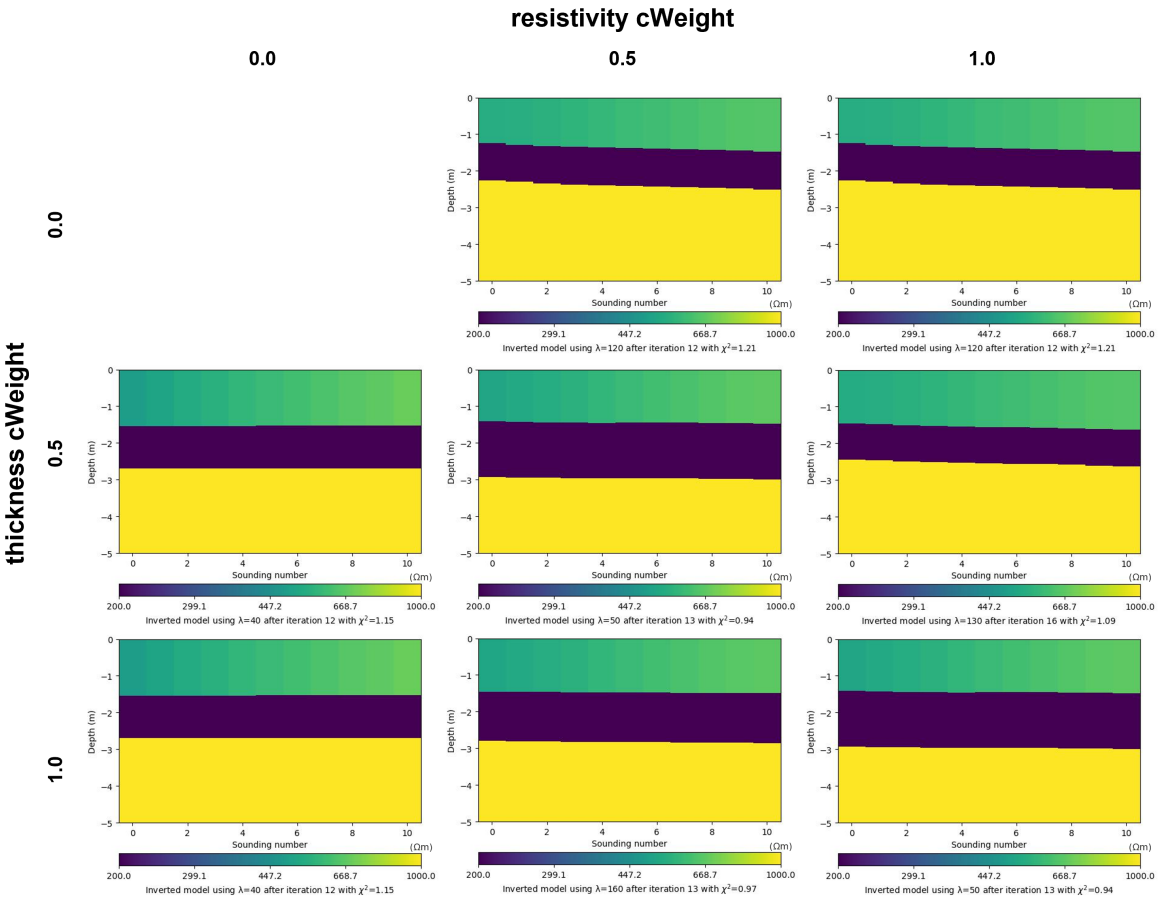


Figure 4-6: Normalized LCI results for varying constraint weights for model parameter regions (i.e., thicknesses and resistivities) which yield the smallest χ^2 for an optimal λ based on a synthetic subsurface model with varying snow resistivity, shown in Figure 4-1b.

4-2 Spatially constrained inversion

Synthetic quasi-3D models were generated based on 1D forward responses regularized with spatial constraints. Synthetic data was created by building a 3D subsurface model for a set of observation points. The exact workflow for the SCI algorithm is shown in Figure 3-1. Initially, the SCI scheme was performed for both a blocky as well as a smooth model. However, the inversion for two parameters did not yield satisfactory results, which will be addressed in Chapter 6. In the following, the model setup and subsequent data generation are described and results of the smooth inversion are illustrated.

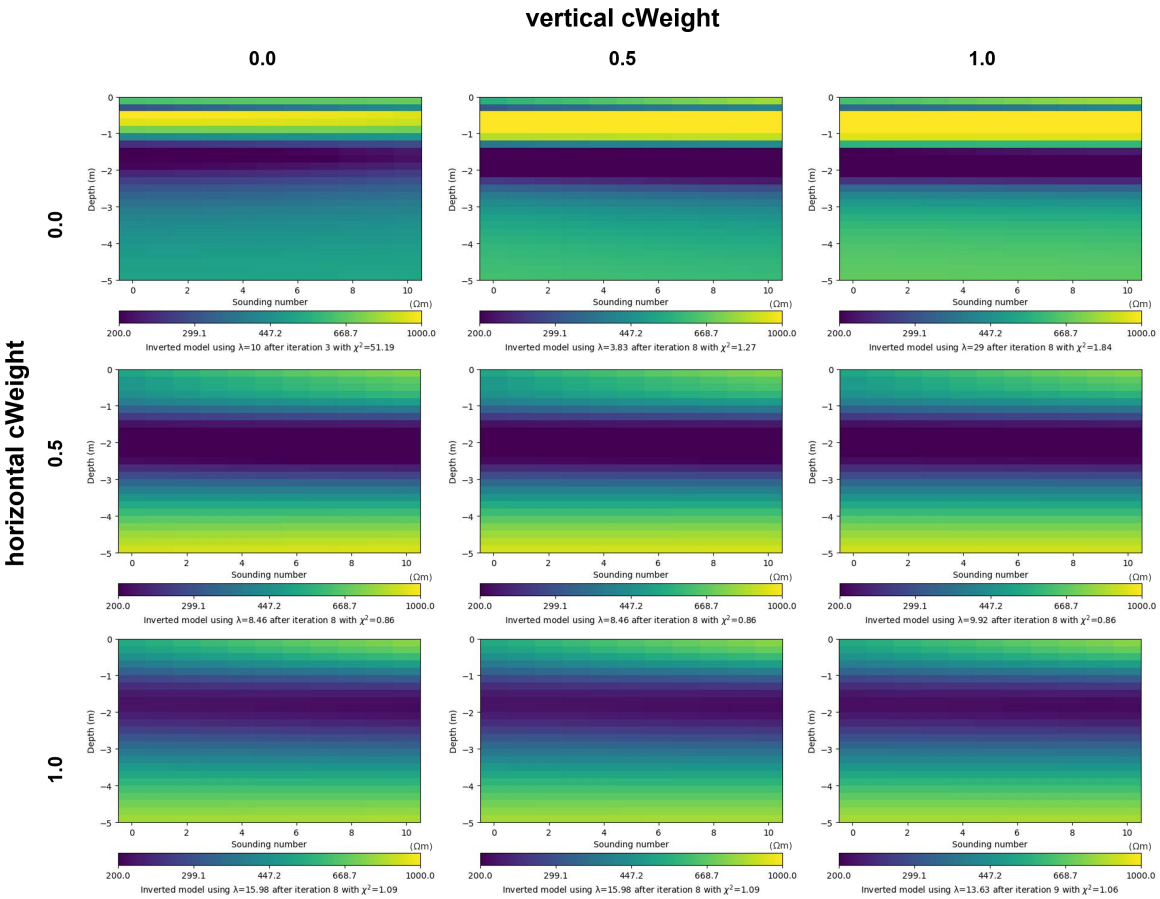
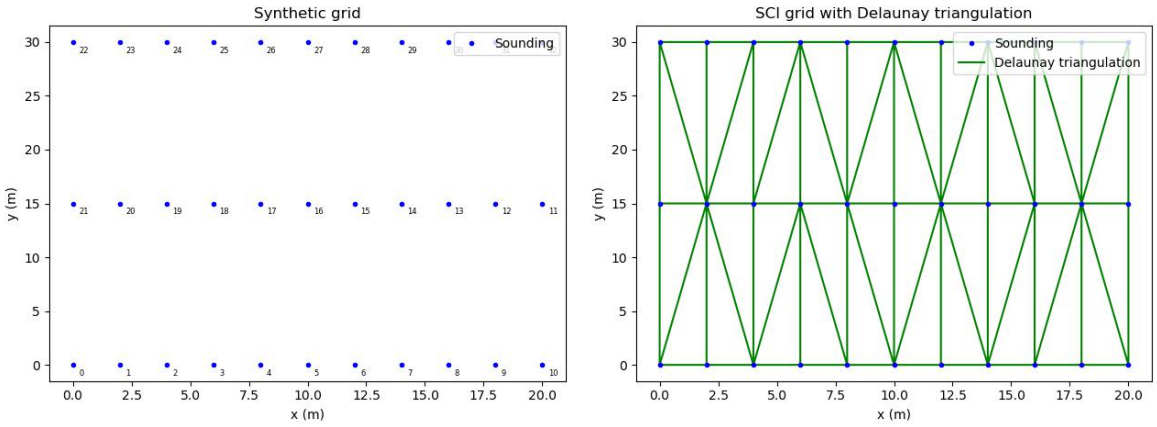


Figure 4-7: LCI results for varying constraint weights across vertical and horizontal boundaries which yield the the smallest χ^2 for an optimal λ based on a synthetic subsurface model with varying snow resistivity, shown in Figure 4-1b.

4-2-1 Synthetic model setup and data generation

For the synthetic SCI study a 2D (i.e., x and y location) mesh of sounding stations was set up prior to the data inversion. The mesh is required to construct a 3D subsurface model for generating a suitable data set, but also for providing locations for the soundings to be constrained based on the distribution of Delaunay triangles (Figure 4-8b). The corresponding grid is displayed in Figure 4-8a.

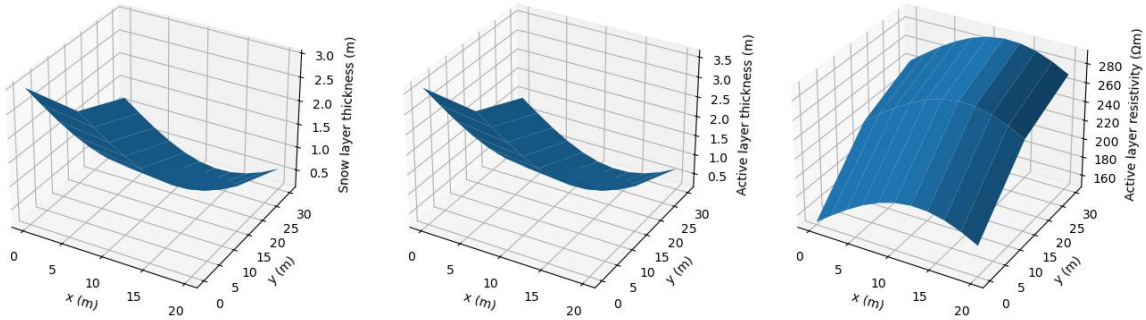
The synthetic 3D underground model, referred to as the true model, is composed of three layers typical for a permafrost soil (cf. Chapter 2-1 and 4-1-1). With respect to the grid of observation points, covering an area of 20 x 30 m, subsurface layers were modelled as expected on a hillside [Douglas et al., 2021; Hubbard et al., 2013; Minsley et al., 2012; Uhlemann et al., 2021]. The presumed topography resembles the survey area in Alaska, where at one point the snow thickness was measured in a range from 2.5 to 0.5 m over a distance of 20 m (Figure 4-9a). The structure (i.e., thickness and resistivity) of the active layer was deduced



(a) **(b)**

Figure 4-8: Synthetically derived 2D grid of observation points aligned in 3 three profile lines (a), and connections between nearest neighbors based on the Delaunay triangulation algorithm (b).

from the estimated variation of the snow layer, with a thicker active layer below a thicker snowpack (Figure 4-9 and 4-10). This relationship is based on the knowledge that the snow layer limits the heat transfer, and thus protects the surface from winter temperatures [Kane et al., 1991]. The resistivities of the snow and permafrost are kept constant at $\rho = 500$ Ωm and $\rho = 1{,}000$ Ωm, respectively. The synthetic field values for the SCI study were calculated in the forward routine and a relative noise of 2.15 % was employed.



(a) **(b)** **(c)**

Figure 4-9: Computed parameter distribution for a synthetic 3D model representative for a permafrost soil on a hillside in Alaska. The space curves provide the thickness of both snow layer (a) and active layer (b), and the resistivity of the active layer (c). The corresponding subsurface model (i.e., the true model) is shown in Figure 4-10.

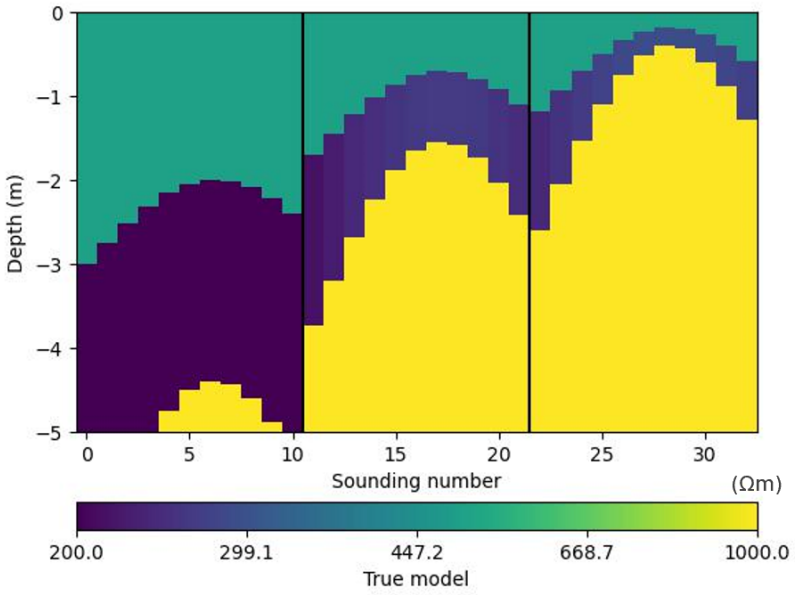


Figure 4-10: Synthetic 3D three-layered permafrost model (referred to as true model), displayed in 2D, consisting of a snow layer with varying thickness, an active layer with varying thickness and resistivity, as shown in Figure 4-9, and the permafrost layer. The model is comprised of 33 stitched-together soundings aligned in three profile lines (separated by black lines), displayed in Figure 4-8.

4-2-2 Inversion parameters

The inversion parameters (i.e., horizontal constraint weight and regularization parameter λ) are applied, as explained in Chapter 4-1-3, in order to analyze the effect of the reference constraint value A and the exponent a for the distance-dependent weighting of constraints, as part of Equation 2-19.

4-2-3 Results

Quasi-3D models are obtained leveraging on 1D forward operators that are smoothed vertically according to the Occam formalism. The inversion algorithm utilizes spatial constraints to ensure spatially coherent models. The objective of testing the 3D formalism is to validate the dispersion of a priori information in all directions to reconstruct the synthetic 3D subsurface model, but also to derive a suitable parameter combination for the field data example (cf. Chapter 5).

Figure 4-11 gives an overview of the quasi-3D inversion models, whereby different profile lines are separated by black vertical lines (Figure 4-8). In general, it can be observed that the output of the SCI algorithm is reliable for any choice of parameters, i.e., reference constraint value A and exponent a. If the reference constraint value is lower than 1.25 the inversion converges to a reasonably low χ^2 (i.e., $1.10 \leq \chi^2 \leq 1.73$). According to Viezzoli et al. [2008], the exponent a defines how the constraints decrease with distance. While an exponent a of 1.0 results in a relative smooth transition between different profile lines, higher

exponents minimally constrain soundings that are not in close proximity. The reference constraint value A determines the strength of constraints for soundings that are in immediate vicinity, i.e., closer than the average distance between two connected points (cf. Chapter 2-3-3). High reference constraints will consequently promote inline lateral smoothing, given that the inline spacing is smaller than the line spacing. If the reference constraint value is greater than 0.5, and thus exceeds the horizontal constraint weight fixed at 0.5, spatial coherency is more promoted than vertical smoothing.

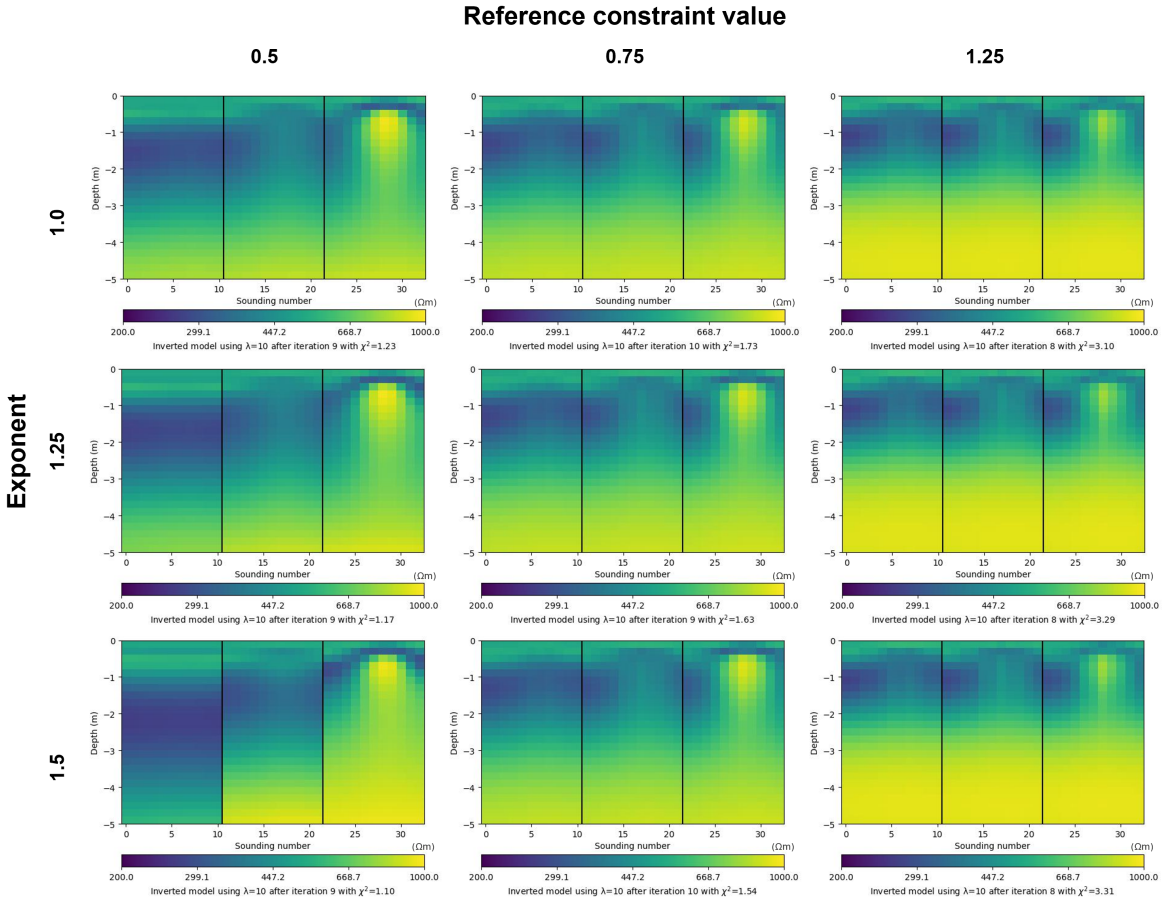


Figure 4-11: SCI results for different distance-dependent vertical constraint weights (cf. Table 3-1) with a horizontal $cW = 0.5$ based on a synthetic three-layered subsurface model (Figure 4-10) (profile lines are separated by black lines).

Deduction of inversion parameters

The "best" inversion model, i.e., the inversion model that resembles the true subsurface model the most, is determined by evaluating the data misfit and model similarity, both by visual inspection as well as by quantitative measures. With respect to Equation 2-19, the best model was found for $a = 1.5$ and $A = 0.5$ (Figure 4-11). The assumption is based on the respective χ^2 value that is inferior to those of other models. This is further confirmed by a high SSIM index of 64 % (Table 4-2) and a low MSE of 7.8 % (Table 4-3) compared to the results of other quasi-3D models. The inversion parameters used are subsequently applied in the field data example (cf. Chapter 5).

Table 4-2: SSIM index [%] (cf. Chapter 2-3-4) for the true subsurface model (Figure 4-10) compared to SCI models (Figure 4-11).

Exponent a	Reference constraint value A		
	0.5	0.75	1.25
1.0	61.9	61.3	60.9
1.25	62.3	61.5	61.1
1.5	64.0	61.9	61.1

Table 4-3: MSE [%] (cf. Chapter 2-3-4) between the true subsurface model (Figure 4-10) and SCI models (Figure 4-11).

Exponent a	Reference constraint value A		
	0.5	0.75	1.25
1.0	10.9	10.6	10.2
1.25	10.3	10.5	10.2
1.5	7.8	10.6	10.2

Comparison of inversion algorithms

The "best" inversion model is compared to a conventional 1D and LCI result and the results are plotted for each profile line (Figure 4-12). Whereas the subsurface section generated with vertically smoothed 1D inversions cannot provide a clear image of the active layer, a structural similarity to the true model exists. No lateral smoothing is applied, hence the transitions between sounding stations are rather sharp. On the contrary, the quasi-2D model (i.e., LCI) can image both the dimensions of the permafrost soil as well as the distribution of electrical properties reasonably well. On the one hand, the extent of the active layer and the inherent electrical resistivity can be inferred from the LCI model, but on the other hand also the permafrost layer, in particular in sounding interval $[22, 32]$, is recovered. However,

a thicker snow layer in the sounding interval $[0, 10]$ indicates that the sensitivity at depth is limited. A slight artifact is shown at a depth of 0.2 m. In contrast, the SCI model is smoothed both laterally and vertically. Compared to the LCI results, the smoothing in the resistivity image is more pronounced and the interface boundaries are difficult to identify. In the first sounding interval $[0, 10]$, the result suggests an active layer closer to the surface, both with respect to the true model and the LCI section. Similarly, the active layer structure is blurred out in interval $[21, 11]$. The electrical resistivity values are generally represented accurately. At the same depth as before, an artifact is recognizable.

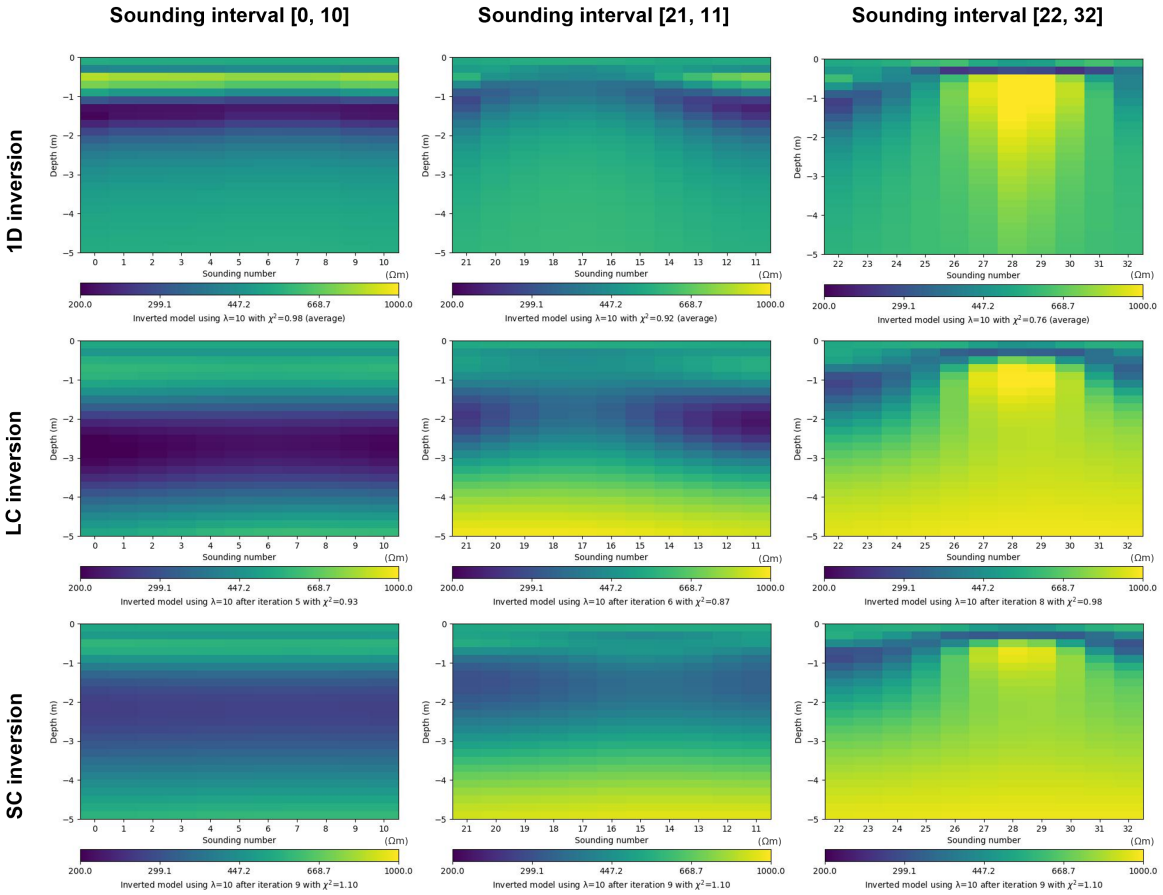


Figure 4-12: Comparison of 1D inversion (Levenberg-Marquardt algorithm), LCI, and SCI result based on a synthetic three-layered subsurface model (Figure 4-10) (each profile line plotted separately).

Chapter 5

Application to field data from Teller, Alaska

The potential of using a quasi-3D modeling approach for imaging of active layer characteristics is demonstrated based on a case study from Teller, Alaska. The monitoring site is introduced hereafter, which is part of the Next-Generation Ecosystem Experiments (NGEE) Arctic project, concerned with the understanding of Arctic system processes and feedbacks to climate. A FDEM data set was inverted to evaluate the benefit of lateral and spatial smoothness constraints in comparison with a conventional 1D inversion technique. The results of the DOI analysis are displayed to assess the credibility of generated inversion models.

5-1 Permafrost monitoring site

The permafrost monitoring site is located on the Seward Peninsula (64.73°N, 165.94°W), approximately 40 km northwest of Nome, Alaska (Figure 5-1). The data was acquired in the Southern part of the Seward peninsula, which is an area of discontinuous permafrost that is generally more sensitive to climate warming [Uhlemann et al., 2021]. Permafrost at the Seward Peninsula is known to be comparably warm, with temperatures usually between -2 and -1 $°C$, but recent deep temperature measurements indicate even warmer temperatures. These highlight the critical state of this environment, which appears to be at a tipping point to loosing its permafrost.

As part of a multi-method monitoring program conducted under the U.S. Department of Energy funded NGEE Arctic project, FDEM soundings were recorded within a watershed at the base of an Arctic hillslope. At the same time, snow thickness was estimated based on repeated UAV (unmanned aerial vehicle, i.e., drone) photogrammetric surveys, which were carried out to derive winter and summer digital elevation models, the difference of which indicating the snow thickness. These estimates were calibrated utilizing direct snow thickness measurements obtained with tile probes. Since permafrost at the site exhibits

different spatial continuity, with locations of near surface permafrost, taliks, and areas of no permafrost, the active layer thickness also shows considerable variability, and can be as shallow as 0.5 to 0.75 m, but usually extends to 1.5 m or deeper.

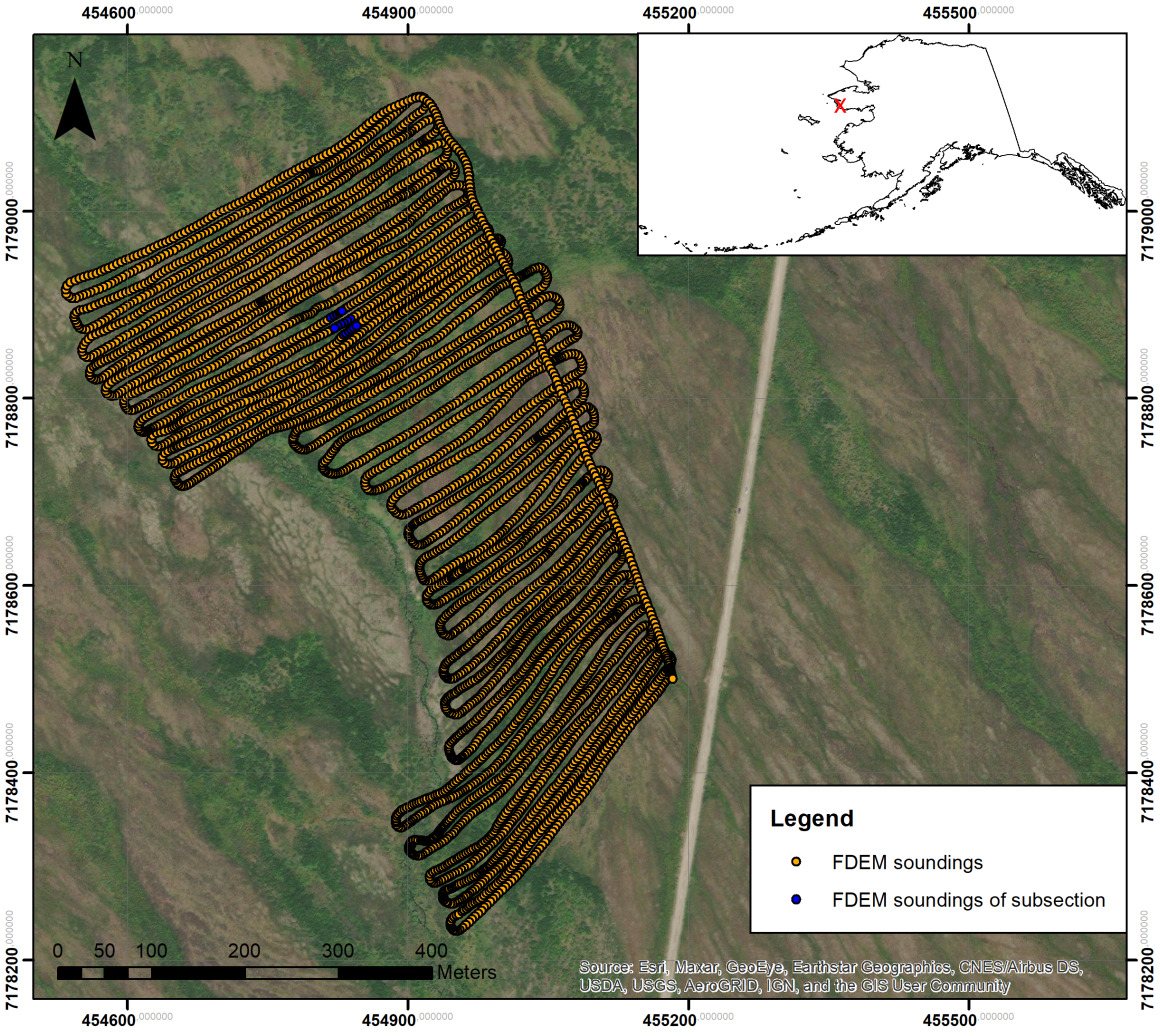


Figure 5-1: Overview map of the survey area in Alaska, US showing the measurement locations of the FDEM data set and the subsection utilized in the inversion case study.

5-2 Field data acquisition

The survey was carried out during the winter season in March 2019. The FDEM soundings were acquired with a sampling distance of approximately 1.7 m along NE to SW profiles spaced with an average distance of 14 m covering an area of 700 x 400 m (Figure 5-1). A subset of the electromagnetic data was selected for the inversion to ensure computational efficiency. The subset is located at the hillslope, where differences in snow thickness were measured and variations in subsurface electrical properties are expected (Figure 5-1).

A single-frequency, multi-configuration CMD Mini explorer was used in combination with a snowmobile to obtain the data. The device consists of one transmitter and 6 receivers with coil spacings of 0.2 m, 0.33 m, 0.5 m, 0.72 m, 1.03 m, and 1.5 m, operating at a frequency of 24,740 Hz. The data set was acquired in the HCP mode (cf. Chapter 2-2-1) and contained 10,743 soundings. Generally, the data set is of good quality, yet the measurements from the coil spacing $s = 0.2$ m produced negative results and were therefore excluded in the inversion process.

5-3 Data processing

The EMI sensor provides in-phase [ppt] and apparent electrical conductivity [mS/m] values for all coil spacings (cf. Chapter 2-2-1). Prior to the inversion, the data was processed to fit the requirements of the *pyGIMLi* library [Rücker et al., 2017]. The existing FDEM forward operator [Rücker et al., 2017] generates in-phase and quadrature values relative to the free air solution. Thus, the acquired apparent conductivities were converted to quadrature ratios using a linear calibration method. The in-phase response produces relative values and is not calibrated to directly obtain magnetic susceptibility [Bonsall et al., 2013]. However, the measured in-phase components turned out to have a large uncertainty, and hence large noise values were assigned to them in the inversion process.

5-4 Spatially constrained inversion

Quasi-3D models of the subsurface electrical resistivity distribution were generated for a subsection of the data set (Figure 5-1) and subsequently compared to results from stitched-together independent 1D inversion of individual soundings as well as LCI. Data from three profile lines were taken into account, amounting to a total of 36 observation points, with 5 in-phase and quadrature measurements, respectively. The corresponding grid is visualized in Figure 5-2a. The constraints between nearest neighbors were implemented with the Delaunay triangulation algorithm, as shown in Figure 5-2b (cf. Chapter 2-3-3).

5-4-1 Inversion parameters

Various combinations of inversion parameters have been tested and evaluated during the synthetic case study (cf. Chapter 4-1-3 and 4-2-3). Suitable parameters for the LCI and SCI of the field data set were deduced, in addition to the parameters already previously determined for the 1D inversion within a hyperparameter test by Thalhammer [2022]. The usage of the "best" inversion parameters that minimize the objective function for the 1D inversion, LCI, and SCI, respectively, ensures comparability between models generated from different inversion algorithms in order to analyze the benefit of lateral and spatial constraints. Table 5-1 summarizes the parameters selected, which are subsequently applied to the Alaska case study.

As outlined in Table 5-1, the inversions of the FDEM soundings were carried out based on a homogeneous starting model with an electrical resistivity of 580 Ωm, which was

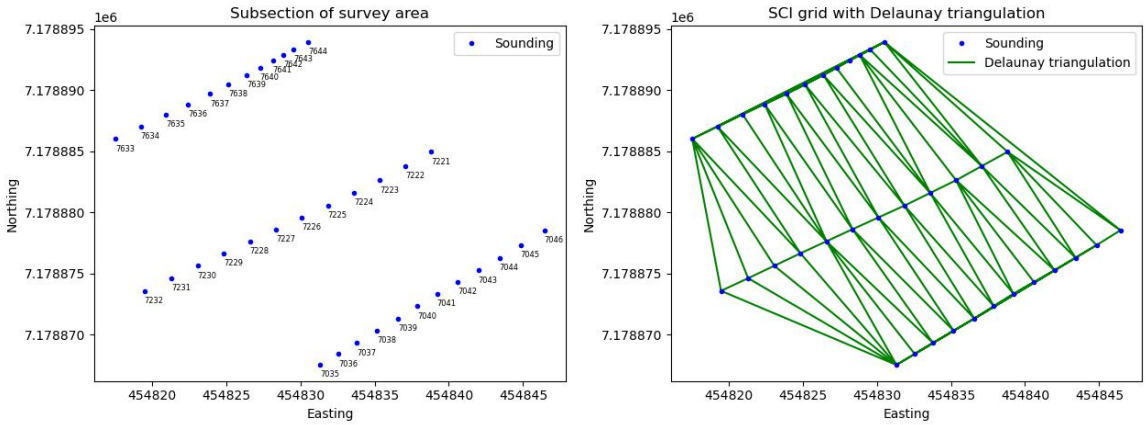


Figure 5-2: 2D grid of 36 observations points aligned in 3 profile lines (a), and connections between nearest neighbors based on the Delaunay triangulation algorithm (b) for a subsection of the survey area in Alaska (Figure 5-1).

obtained by trial and error. The reliability of the in-phase component was considered by setting the noise level to a value of 80 %, while quadrature measurements were assigned a 2.15 % error margin. For the horizontal and vertical constraint weights in case of a LCI, a value of 0.5 was chosen, while spatial constraints for the SCI method were calculated according to Equation 2-19 with an exponent $a = 1.5$ and a reference constraint value $A = 0.5$.

Table 5-1: Inversion parameters used in the field data example for the conventional 1D inversion, LCI, and SCI.

Inversion parameter	Inversion type	Value
Levenberg-Marquardt parameter	1D	0.9
Regularization strength λ	1D, LCI, SCI	10
Starting model [Ωm]	1D, LCI, SCI	580
Noise for quadrature component [%]	1D, LCI, SCI	2.15
Noise for in-phase component [%]	1D, LCI, SCI	80
Horizontal cWeight	LCI, SCI	0.5
Vertical cWeight	LCI	0.5
Exponent a (cf. Equation 2-19)	SCI	1.5
Reference constraint value A (cf. Equation 2-19)	SCI	0.5

5-4-2 Results

In the following, the inversion results based on conventional 1D inversion, LCI, and SCI methods are illustrated and explained. The selection of inversion parameters enables a successful inversion for all techniques as the error criterion χ^2 converges to values close to 1. Figure 5-3 gives an overview of the inverted subsurface models plotted for each profile as well as the measured snow thickness in meters.

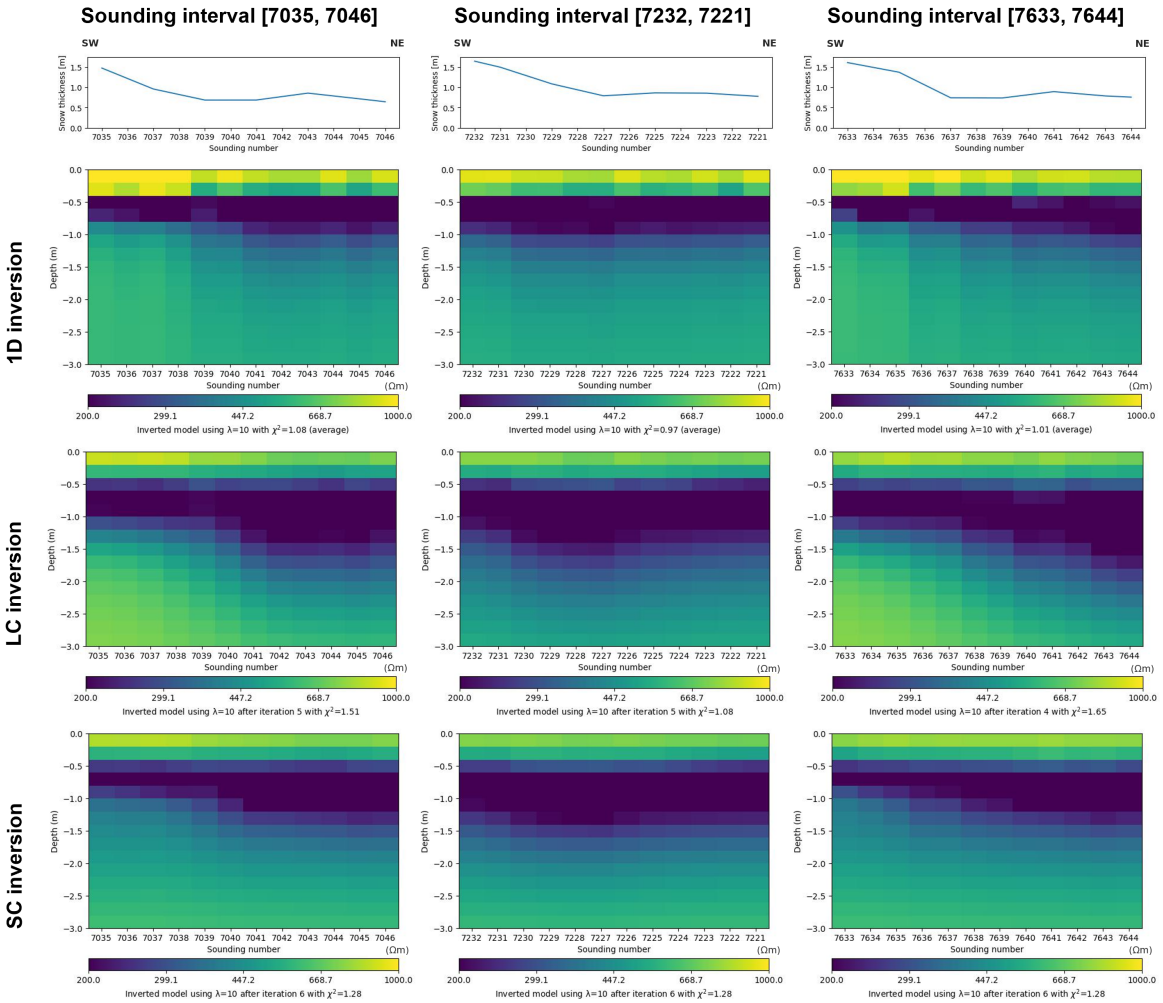


Figure 5-3: Comparison of 1D inversion (Levenberg-Marquardt algorithm), LCI, and SCI result of the electromagnetic field data (each profile line plotted separately) with measured snow thickness.

A comparison of the 1D inversion, LCI, and SCI indicates a structural similarity between inverted models for the respective profile lines, yet differences in layer dimensions as well as smoothness exist. The SCI image has a very smooth appearance where no sharp resistivity changes are apparent. Likewise, the transitions between laterally constrained soundings are smoothed out, whereby the conventional 1D approach does not give a consistent representation of the underground.

In general, it can be observed, that all inversion models exhibit a three-layered image of the subsurface, regardless of the inversion technique. The inverted models are composed of a high-resistive cover layer (ranging from 700 to 1,000 Ωm), a very conductive middle layer (200 Ωm), and a bottom layer characterized by an intermediate resistivity (ranging from 300 to 700 Ωm). Whereas the first layer is described by a constant thickness throughout all profiles, the second layer shows a variation in the vertical extent for different profile lines. A more detailed description of the three layers is presented in the following.

First layer

The layer closest to the surface constitutes the snow layer as the data set was obtained during the winter season. The measured snow thickness is plotted separately for all profile lines. A downward trend from SW to NE with a slight peek on the right side of the profile can be recognized for all three sounding intervals. The thickness of the snow layer ranges from approximately 1.5 to 0.7 m in SW and NE direction, respectively.

Throughout all inversion profiles, the electrical resistivity of the cover layer reveals a similar trend, as it is decreasing from SW to NE from approximately 1,000 to 700 Ωm, depending on the inversion type. Whereas the 1D inversion models show large resistivity variations until a depth of 0.4 m, LCI and SCI provide a much smoother transition between neighboring soundings. This is the case for both inline soundings, as well as across profile lines for spatially constrained models. Subsequently, the differences in electrical resistivity are lowest for SCI models. However, the snow layer appears to have a constant thickness of approximately 0.5 m, regardless of location and inversion algorithm, although not confirmed by the obtained snow thickness measurements. A clear separation from the high-conductive underlying layer (i.e., the second layer) is visible, based on the resistivity distribution.

Second layer

The second layer, which usually defines the active layer in a permafrost environment, is characterized by a constant electrical resistivity of about 200 Ωm, however slight fluctuations are noticeable for single soundings in the conventional 1D inversion. Apart from that, the second layer exhibits varying vertical dimensions, not only along the profile, but also across profile lines as well as for different inversion techniques. The sounding intervals [7035, 7046] and [7633, 7644] display an increasing thickness of the high-conductive second layer towards the NE of the data subset, especially pronounced for LCI and SCI models. The second interval [7232, 7221] indicates an opposite trend, as the vertical extent is highest in SW direction, but also shows a more steady layer thickness along the line. The stitched-together 1D sounding profile generally suggests a relatively constant second layer with a thickness in the range of 0.4 to 0.6 m. The LCI and SCI sections provide more variation, and LCI images the second layer with a maximum thickness of 1 m within sounding interval [7633, 7644] at the most NE section. In contrast, SCI models retrieve a high-conductive layer that is 0.2 to 0.8 m thick.

Third layer

The third layer is described by a relatively smooth electrical resistivity distribution. The resistivity values vary from 400 to 600 Ωm, and in particular LCI and 1D inversion show a wide span of resistivities. The SCI models do not showcase a resistivity contrast within the third layer, but display a constant resistivity value of approximately 500 Ωm.

5-4-3 Assessment

In order to determine the reliability of the generated subsurface models, a DOI analysis was carried out. The DOI index enables the definition of a cutoff value $R_{DOI} = 0.2$, as proposed by Oldenburg and Li [1999], for the interpretation of the permafrost images. Figure 5-4 suggests, that the data is very sensitive to uppermost layers, yet the interface at 0.5 m should be considered carefully. Naturally, the DOI increases with depth, however deeper layers at the left part of the subset seem to be more reliable, at least for sounding intervals [7035, 7046] and [7633, 7644]. For the second profile line (i.e, sounding interval [7232, 7221]) the depth sensitivity is more limited and approaches 0.2 at a depth of 3 m along the whole profile line.

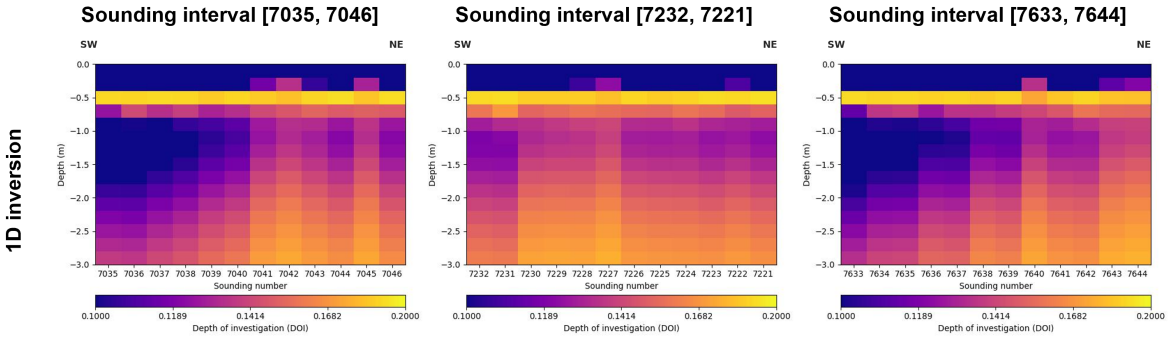


Figure 5-4: DOI index computed from 1D inversion models having starting models with $\rho =$ 5,800 Ωm and $\rho = 58$ Ωm, respectively. The cutoff value is set at $R_{DOI} = 0.2$.

Chapter 6

Discussion

Lateral and spatial coherency between soundings has been enforced to decrease the ambiguity of 1D electromagnetic inversion models. The aim of this thesis was to create an accurate subsurface image within a permafrost environment to delineate the active layer as an essential indicator of climate change.

The synthetic study in Chapter 4 based on three different subsurface models shows that inversion models implemented with lateral and spatial constraints provide a more consistent representation of the true model compared to a conventional 1D inversion. This is presented both for a blocky as well as a smooth subsurface model, yet the results suggest that a careful selection of constraint weights is essential as the blocky inversion has been found less reliable when the subsurface is not characterized by a layered structure. The field data inversion in Chapter 5 successfully models a high-conductive layer, i.e., the assumed active layer, although a thick snow layer seems to influence the retrieved subsurface image. The key findings of this thesis will be discussed in detail hereafter.

Inversion of synthetic data

The synthetic examples demonstrate that the LCI method certainly outperforms conventional 1D inversion techniques (Figures 4-2, 4-3, 4-4, 4-5, 4-6, and 4-7). The blocky inversion not only allows a clear location of subsurface interfaces, but additionally enables a reconstruction of electrical properties with high precision. This implies that the expected subsurface structure is met with a correct parametrization, which for this study was achieved by defining a suitable inversion starting model. Figure 4-3 further illustrates that the constraint weights bear great impact on the final results and that the method is prone to errors if the assumption of a layered subsurface is not met [Knödel et al., 2007]. The corresponding objective function could only be minimized sufficiently if implemented constraints do not consider thickness model parameters, but rather act on resistivity model parameters. These findings confirm the statements made by Vignoli et al. [2017], outlining that few-layer inversion methods tend to create artifacts in particular when the geological structure is unexpected and complex.

Vignoli et al. [2017] further claim that the choice of discretization and regularization is significant. From Figure 4-6 it can be concluded that an equal constraint weight on both model parameters yields accurate results, whereas dominant resistivity constraints lead to an overestimation of the snow thickness and therefore layering is promoted to a small extent.

Even though the effectiveness of laterally constrained blocky inversion models is limited in specific environments, the smooth inversion provides a reliable alternative. The choice of constraint weights seems straightforward, as both Figure 4-4 and 4-7 indicate that a medium horizontal variance is beneficial for producing a smooth yet fitting model, independent of the implemented degree of vertical smoothing. The results corroborate the flexibility of LCI based on a 1D Occam type inversion in different contexts [Guillemoteau et al., 2016].

The synthetic 3D permafrost model shows that even complex subsurface structures can be reconstructed using a SCI method that relies on a smooth forward model (Figure 4-11). The output is plausible for a wide range of parameters as introduced in Equation 2-19, which is relevant for spatially constraining nearest neighbors based on the distance between soundings. This is in accordance with observations made by Viezzoli et al. [2008]. From the inverted models it can be inferred that the reference constraint value should not exceed the vertical constraint weights with the objective to maintain the inherent 1D smoothing regularization. Quantitative methods, i.e., the calculation of MSE and SSIM index, further support the choice of parameters. Nevertheless, Figure 4-12 suggests that SCI greatly enhances spatial smoothing, even if structural differences between profile lines are severe. One possible explanation is that the distance between profile lines is overemphasized in the SCI formalism in relation to the distance between consecutive soundings. Subsequently, it can be deduced that SCI not necessarily exceeds the performance of LCI, in addition to being computationally more expensive. In general, it can be concluded that the error criterion χ^2 gives a good indication of the success of constrained inversion methods.

Normalization

The normalization procedure, introduced in Chapter 3-3-3, manages to ensure an equal regularization strength independent of the choice of constraint weights, producing inversion models close to the true model (Figures 4-3 and 4-6). The developed method is the first known implementation of such a normalization technique, successfully integrated for blocky inversion models. Despite these promising results, some questions remain with regard to the formalism of the normalization function as well as the application to smooth inversion models. The normalization factor is applied to the weighted constraint matrix to keep the model objective function Φ_M constant for different examples, yet the effect of the amplitude of Φ_M on the minimization problem is unclear and exceeds the scope of this Master thesis. Figure 6-1 compares the LCI results with (b) and without (a) normalization applied. The objective function of the normalized model is minimized to a desired level, but the artifact close to the surface is much more emphasized. Additionally, more iterations as well as a higher regularization strength are necessary to solve the inversion problem. This outlines the difficulties of the normalization which should be addressed in future studies.

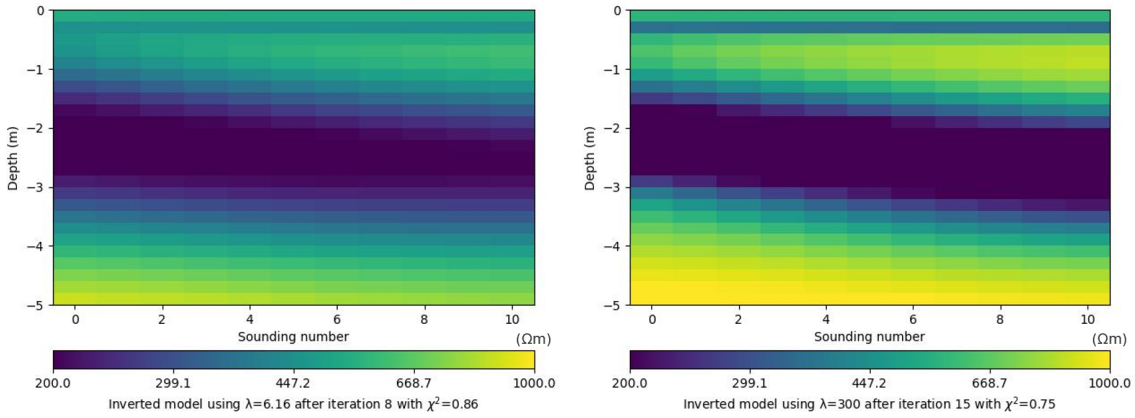


Figure 6-1: Comparison between not normalized (a) and normalized (b) LCI result for constraint weights of 0.5 across vertical and horizontal boundaries based on a synthetic subsurface model with varying snow thickness, shown in 4-1a.

Inversion of field data

The developed SCI presents its benefits compared to LCI and in particular conventional 1D inversion when applying it to a real field data example (Figure 5-3). Although all used inversion methods produce meaningful subsurface models, SCI succeeds in limiting large resistivity variations within layers. Quasi-3D models additionally showcase spatially more coherent layer dimensions, evident by the vertical extent of the second layer that is more aligned both along and across profile lines. On the one hand, this could lead to a misinterpretation of the true thickness as profile lines are spaced by more than 10 m, and therefore are not necessarily structurally related. On the other hand, models that are only laterally constrained could overly promote features such as the layer thickness based on the result of one sounding, e.g., the sounding at the end of a profile.

Figure 5-3 clearly illustrates the inability of the inversion procedure to accurately reconstruct the snow cover. Although the snow layer thickness measurements show a decrease from SW to NE, the retrieved models visualize a constant snow layer. The discrepancy is confirmed by the DOI index indicating a lack of sensitivity at a depth of 0.5 m (Figure 5-4). While a higher measured snow thickness corresponds with higher snow resistivities, the extent of the high-conductive layer, i.e., assumed to be the active layer, is minimal. One possible explanation for this result is that the measured signals show no sensitivity to the active layer, which consequently is underestimated. The generated inversion models further link a thin snow cover with smaller snow resistivities. The reason is not clear, yet this effect can already be observed when analyzing the raw data set, i.e., the apparent conductivity values. A thick snow layer relates to a high apparent resistivity and vice versa. Nevertheless, these findings are in agreement with those obtained by Hauck et al. [2008] who state that a difference in snow resistivity is not necessarily associated with actual resistivity variations but rather attributable to a heterogeneous snow layer thickness. This is in particular the case for dry snow, correlating with cold snow temperatures, and thus high snow resistivities [Glen and Paren, 1975; Hauck et al., 2008]. Another explanation might be that varying

electrical properties in snow indicate an inherent temperature difference according to Glen and Paren [1975]. A previous 1D synthetic study additionally observed an underestimation of the thickness of the active layer in response to high snow resistivities [Thalhammer, 2022]. From this follows that there is abundant room for determining the impact of the snow cover on the success of an electromagnetic inversion, as suggested by Kneisel et al. [2008], which should be more extensively studied with a recently obtained data set from the summer season.

While the smooth inversion obtains meaningful results of the subsurface, the few-layer inversion could not converge to a reasonably low χ^2 value, as illustrated by an example in Figure 6-2. However, this is not surprising as the electromagnetic inversion problem is highly under-determined and a three-layer model is rather complicated to retrieve. The in-phase components of the acquired data set are not quantitatively robust, and furthermore, the data gained from the smallest coil spacing was found to be flawed. The importance of regularization in order to produce a stable solution is again confirmed, as has been long discussed in the literature [Engl et al., 1996; Zhdanov, 2015]. This can be acknowledged in future studies by utilizing more frequencies or coil spacings during the data acquisition.

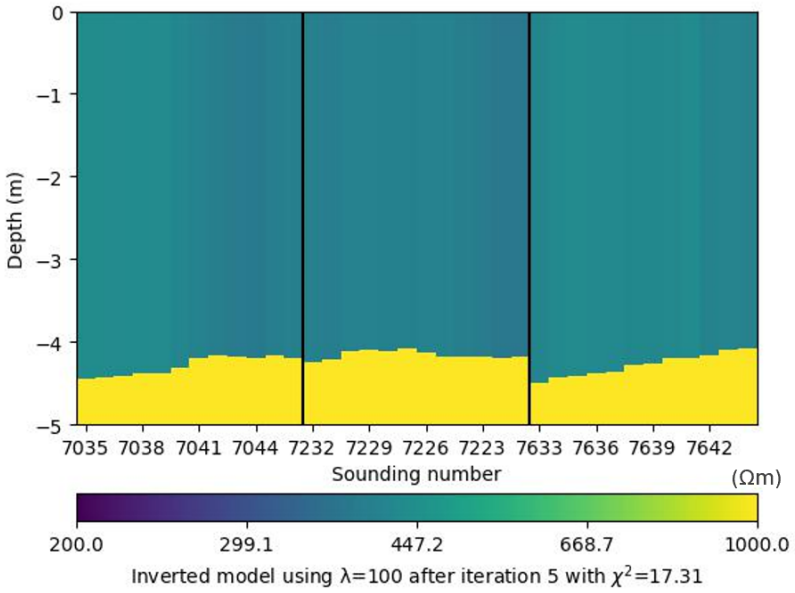


Figure 6-2: Exemplary SCI result of the electromagnetic field data with thickness parameters constrained twice as much as resistivity parameters, with respect to the distance, as outlined in Equation 2-19, calculated according to Table 5-1 (profile lines are separated by black lines).

With regard to generated results and derived conclusions it has to be considered that the results are based on small data subsections. The data set is extensive and an unconstrained conventional 1D inversion takes approximately 30 min. The inversion is naturally computationally expensive if constraints are implemented, hence it was not possible to carry out a LCI and SCI of the full data set, because the inversion would use 1.39 and 2.78 TiB of memory, respectively. Nevertheless, this should be addressed by a subsequent inversion of data subsets, as introduced by Viezzoli et al. [2008, 2009].

Chapter 7

Conclusion

Developing more sophisticated, computationally efficient inversion techniques is of great interest, not only for environmental studies, but generally part of ongoing research [Guillemoteau et al., 2017; Klose et al., 2022]. Geophysical inversion, however, is challenging due to the non-linearity inherent to most forward modeling equations and their solution instability or non-uniqueness. The integration of structural and spatial information helps to obtain reliable subsurface models, the delineation of sharp layer interfaces is limited nevertheless.

Quasi-3D inversion outperforms conventional 1D inversion techniques for imaging of permafrost soil. The benefit of regularization methods for electromagnetic inverse problems was evaluated by incorporating lateral and spatial constraints in the inversion algorithm with the objective to characterize properties of the active layer. Synthetic data sets derived from idealized subsurface models as well as a field data set were used to show the applicability of the developed inversion code. Quasi-2D models were produced by laterally smoothing model parameters of adjacent soundings, i.e., a laterally constrained inversion (LCI). LCI models based on a blocky model allow an almost perfect reconstruction of quasi-layered formations and an accurate identification of layer boundaries. In the case of more complex subsurface structures, smooth LCI models (i.e, Occam type 1D forward responses that are additionally constrained in horizontal direction) provide minimum-structure models, which offer meaningful subsurface representations. Intermediate constraint weights ($cW = 0.5$) were found to be appropriate for various subsurface settings, while at the same time preventing over-smoothing of the generated inversion model. Similarly, spatially constrained quasi-3D inversions (i.e., SCI) established with a Delaunay triangulation succeed in investigating permafrost environments. The associated distance-dependent constraint function is relatively robust for a wide range of parameters. Ideally, implemented lateral smoothness constraints connecting adjacent observation points should not exceed the intrinsic 1D smoothness constraints across horizontal boundaries. Furthermore, an exponential relationship between sounding distance and constraint weight is advantageous. SCI models seem to overemphasize a structural similarity, and thus the variance between linked soundings across profile lines. Subsequently, the choice between LCI and SCI needs careful consideration as computed inversion models do not give a clear indication if enforcing spatial coherency benefits the

inversion models in comparison to a quasi-2D inversion procedure. The field data study is able to confirm the selection of utilized inversion parameters and the potential of regularized inversion methods for permafrost monitoring. Even though the inversion results cannot replicate the snow layer thickness in accordance with obtained measurements, the vast influence of the snow presence on retrieved electrical properties and layer thicknesses is obvious.

The ambiguity of models inverted from FDEM soundings was successfully addressed, yet uncertainties remain if data is sparse and of low quality. The forward operator simultaneously inverts data from different coil spacings, while a priori information is added to the algorithm by laterally or spatially constraining model parameters of neighboring nodes. Integrating geological knowledge in the objective function or expanding the single-method imaging to a joint inversion approach could stabilize the inversion, leaving room for further research advances.

Chapter 8

Outlook

While the implemented inversion algorithm regularized with lateral and spatial constraints provides meaningful subsurface models in areas of permafrost, some questions and challenges for future studies remain. The imaging of a three-layered subsurface section was possible, yet the incorporation of a priori information could further constrain the intrinsically complex electromagnetic inverse problem. This can be achieved by tuning the constraint weights according to available a priori information, as well as by extending the objective function to be minimized. The integration of the measured snow thickness, for example, would help to address regions of low sensitivity.

The development of more exhaustive subsurface models can for instance be accomplished by inverting for secondary model parameters, as well as utilizing more extensive data sets. This would allow to incorporate knowledge about the topography, and thus elevation data in the inversion algorithm. Alternatively, the inversion could solve for depth parameters, rather than thicknesses, which according to Auken and Christiansen [2004] promotes layered subsurface structures. The computational efficiency is limited when applying large data sets, however the data inversion could be performed on subsets suitable for a parallel computation [Viezzoli et al., 2008, 2009].

This thesis further emphasizes the impact of the snow layer on the inversion performance which has not previously been discussed in detail in literature. A better understanding of snow properties, including water content and temperature, and their influence on the electromagnetic response could help to reduce uncertainties with respect to the interpretation of generated inversion models. The comparison of models inverted with data obtained during summer and winter, respectively, has the potential to improve the assessment of the snow cover effect on the sensitivity of electromagnetic signals.

Bibliography

Auken, E., Christiansen, A., Jacobsen, L., and Sørensen, K. (2005a). Laterally constrained 1d-inversion of 3d tem data. In *Symposium on the Application of Geophysics to Engineering and Environmental Problems 2005*, pages 514–518. Society of Exploration Geophysicists.

Auken, E., Christiansen, A., Viezzoli, A., Fitzpatrick, A., Cahill, K., Munday, T., and Berens, V. (2009). Investigation on the groundwater resources of the south eyre peninsula, south australia, determined from laterally constrained inversion of tempest data. *ASEG Extended Abstracts*, 2009(1):1–5.

Auken, E. and Christiansen, A. V. (2004). Layered and laterally constrained 2d inversion of resistivity data. *Geophysics*, 69(3):752–761.

Auken, E., Christiansen, A. V., Jacobsen, B. H., Foged, N., and Sørensen, K. I. (2005b). Piecewise 1d laterally constrained inversion of resistivity data. *Geophysical Prospecting*, 53(4):497–506.

Auken, E., Christiansen, A. V., Jacobsen, L. H., and Sørensen, K. I. (2008). A resolution study of buried valleys using laterally constrained inversion of tem data. *Journal of applied geophysics*, 65(1):10–20.

Auken, E., Christiansen, A. V., Kirkegaard, C., Fiandaca, G., Schamper, C., Behroozmand, A. A., Binley, A., Nielsen, E., Effersø, F., Christensen, N. B., et al. (2015). An overview of a highly versatile forward and stable inverse algorithm for airborne, ground-based and borehole electromagnetic and electric data. *Exploration Geophysics*, 46(3):223–235.

Backus, G. E. and Gilbert, J. F. (1967). Numerical applications of a formalism for geophysical inverse problems. *Geophysical Journal International*, 13(1-3):247–276.

Barber, C. B., Dobkin, D. P., and Huhdanpaa, H. (1996). The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4):469–483.

Biskaborn, B. K., Smith, S. L., Noetzli, J., Matthes, H., Vieira, G., Streletskiy, D. A., Schoeneich, P., Romanovsky, V. E., Lewkowicz, A. G., Abramov, A., et al. (2019). Permafrost is warming at a global scale. *Nature communications*, 10(1):1–11.

Boaga, J., Phillips, M., Noetzli, J., Haberkorn, A., Kenner, R., and Bast, A. (2020). A comparison of frequency domain electro-magnetometry, electrical resistivity tomography and borehole temperatures to assess the presence of ice in a rock glacier. *Frontiers in Earth Science*, page 593.

Bonsall, J., Fry, R., Gaffney, C., Armit, I., Beck, A., and Gaffney, V. (2013). Assessment of the cmd mini-explorer, a new low-frequency multi-coil electromagnetic device, for archaeological investigations. *Archaeological Prospection*, 20(3):219–231.

Butler, D. K. (2005). *Near-surface geophysics*. Society of Exploration Geophysicists.

Christensen, T. R., Johansson, T., Åkerman, H. J., Mastepanov, M., Malmer, N., Friborg, T., Crill, P., and Svensson, B. H. (2004). Thawing sub-arctic permafrost: Effects on vegetation and methane emissions. *Geophysical research letters*, 31(4).

Christiansen, A. V., Auken, E., Foged, N., and Sørensen, K. I. (2007). Mutually and laterally constrained inversion of cves and tem data: a case study. *Near Surface Geophysics*, 5(2):115–123.

Constable, S. C., Parker, R. L., and Constable, C. G. (1987). Occam's inversion: A practical algorithm for generating smooth models from electromagnetic sounding data. *Geophysics*, 52(3):289–300.

Costabel, S., Günther, T., Dlugosch, R., and Müller-Petke, M. (2016). Torus-nuclear magnetic resonance: Quasicontinuous airborne magnetic resonance profiling by using a helium-filled balloon. *Geophysics*, 81(4):119–129.

Daniels, J. J., Keller, G. V., and Jacobson, J. (1976). Computer-assisted interpretation of electromagnetic soundings over a permafrost section. *Geophysics*, 41(4):752–765.

Davies, G., Huang, J., Monteiro Santos, F. A., and Triantafilis, J. (2015). Modeling coastal salinity in quasi 2d and 3d using a dualem-421 and inversion software. *Groundwater*, 53(3):424–431.

Deng, X. and Tong, X. (2020). 2d inversion of frequency-domain electromagnetic data generated by line current source. In *Journal of Physics: Conference Series*, volume 1624. IOP Publishing.

Dobinski, W. (2011). Permafrost. *Earth-Science Reviews*, 108(3-4):158–169.

Douglas, T. A., Hiemstra, C. A., Anderson, J. E., Barbato, R. A., Bjella, K. L., Deeb, E. J., Gelvin, A. B., Nelsen, P. E., Newman, S. D., Saari, S. P., et al. (2021). Recent degradation of interior alaska permafrost mapped with ground surveys, geophysics, deep drilling, and repeat airborne lidar. *The Cryosphere*, 15(8):3555–3575.

Ellis, R. and Oldenburg, D. (1994). Applied geophysical inversion. *Geophysical Journal International*, 116(1):5–11.

Engl, H. W., Hanke, M., and Neubauer, A. (1996). *Regularization of inverse problems*, volume 375. Springer Science & Business Media.

Fiandaca, G., Auken, E., Christiansen, A. V., and Gazoty, A. (2012). Time-domain-induced polarization: Full-decay forward modeling and 1d laterally constrained inversion of cole-cole parameters. *Geophysics*, 77(3):213–225.

Frederiksen, R. R., Christiansen, A. V., Christensen, S., and Rasmussen, K. R. (2017). A direct comparison of emi data and borehole data on a 1000 ha data set. *Geoderma*, 303:188–195.

Gavin, H. P. (2019). The levenberg-marquardt algorithm for nonlinear least squares curve-fitting problems. *Department of Civil and Environmental Engineering, Duke University*, 19.

Glen, J. and Paren, J. (1975). The electrical properties of snow and ice. *Journal of Glaciology*, 15(73):15–38.

Guillemoteau, J., Christensen, N. B., Jacobsen, B. H., and Tronicke, J. (2017). Fast 3d multichannel deconvolution of electromagnetic induction loop-loop apparent conductivity data sets acquired at low induction numbers. *Geophysics*, 82(6):357–369.

Guillemoteau, J., Simon, F.-X., Lück, E., and Tronicke, J. (2016). 1d sequential inversion of portable multi-configuration electromagnetic induction data. *Near Surface Geophysics*, 14(5):423–432.

Guillemoteau, J. and Tronicke, J. (2016). Evaluation of a rapid hybrid spectral-spatial domain 3d forward-modeling approach for loop-loop electromagnetic induction quadrature data acquired in low-induction-number environments. *Geophysics*, 81(6):447–458.

Guillemoteau, J., Vignoli, G., Barreto, J., and Sauvin, G. (2022). Sparse laterally constrained inversion of surface-wave dispersion curves via minimum gradient support regularization. *Geophysics*, 87(3):281–289.

Günther, T. (2013). On inversion of frequency domain electromagnetic data in salt water problems-sensitivity and resolution. In *Near Surface Geoscience 2013-19th EAGE European Meeting of Environmental and Engineering Geophysics*. European Association of Geoscientists & Engineers.

Günther, T. and Rücker, C. (2010). Advanced inversion strategies using a new geophysical inversion and modelling library. In *23rd EEGS Symposium on the Application of Geophysics to Engineering and Environmental Problems*. European Association of Geoscientists & Engineers.

Günther, T., Rücker, C., and Spitzer, K. (2006). Three-dimensional modelling and inversion of dc resistivity data incorporating topography—ii. inversion. *Geophysical Journal International*, 166(2):506–517.

Hansen, P. C. (1992). Analysis of discrete ill-posed problems by means of the l-curve. *SIAM review*, 34(4):561–580.

Harada, K., Wada, K., and Fukuda, M. (2000). Permafrost mapping by transient electromagnetic method. *Permafrost and Periglacial Processes*, 11(1):71–84.

Harris, C. (2005). Climate change, mountain permafrost degradation and geotechnical hazard. In *Global change and mountain regions*, pages 215–224. Springer.

Hauck, C. (2013). New concepts in geophysical surveying and data interpretation for permafrost terrain. *Permafrost and Periglacial Processes*, 24(2):131–137.

Hauck, C., Guglielmin, M., Isaksen, K., and Vonder Mühll, D. (2001). Applicability of frequency-domain and time-domain electromagnetic methods for mountain permafrost studies. *Permafrost and Periglacial Processes*, 12(1):39–52.

Hauck, C., Kneisel, C., et al. (2008). *Applied geophysics in periglacial environments*. Cambridge University Press.

Hinzman, L. D., Bettez, N. D., Bolton, W. R., Chapin, F. S., Dyurgerov, M. B., Fastie, C. L., Griffith, B., Hollister, R. D., Hope, A., Huntington, H. P., et al. (2005). Evidence and implications of recent climate change in northern alaska and other arctic regions. *Climatic change*, 72(3):251–298.

Hjort, J., Streletskiy, D., Doré, G., Wu, Q., Bjella, K., and Luoto, M. (2022). Impacts of permafrost degradation on infrastructure. *Nature Reviews Earth & Environment*, 3(1):24–38.

Hubbard, S. S., Gangodagamage, C., Dafflon, B., Wainwright, H., Peterson, J., Gusmeroli, A., Ulrich, C., Wu, Y., Wilson, C., Rowland, J., et al. (2013). Quantifying and relating land-surface and subsurface variability in permafrost environments using lidar and surface geophysical datasets. *Hydrogeology Journal*, 21(1):149–169.

Jorgenson, M. T., Shur, Y. L., and Pullman, E. R. (2006). Abrupt increase in permafrost degradation in arctic alaska. *Geophysical Research Letters*, 33(2).

Kane, D. L., Hinzman, L. D., and Zarling, J. P. (1991). Thermal response of the active layer to climatic warming in a permafrost environment. *Cold Regions Science and Technology*, 19(2):111–122.

Kearey, P., Brooks, M., and Hill, I. (2002). *An introduction to geophysical exploration*, volume 4. John Wiley & Sons.

Klose, T., Guillemoteau, J., Vignoli, G., and Tronicke, J. (2022). Laterally constrained inversion (lci) of multi-configuration emi data with tunable sharpness. *Journal of Applied Geophysics*, 196.

Kneisel, C., Hauck, C., Fortier, R., and Moorman, B. (2008). Advances in geophysical methods for permafrost investigations. *Permafrost and periglacial processes*, 19(2):157–178.

Knödel, K., Lange, G., and Voigt, H.-J. (2007). *Environmental geology: handbook of field methods and case studies*. Springer Science & Business Media.

Korhonen, K., Ruskeeniemi, T., Paananen, M., and Lehtimäki, J. (2009). Frequency domain electromagnetic soundings of canadian deep permafrost. *Geophysica*, 45(1-2):77–92.

Loke, M. H., Acworth, I., and Dahlin, T. (2003). A comparison of smooth and blocky inversion methods in 2d electrical imaging surveys. *Exploration geophysics*, 34(3):182–187.

Madsen, L. M., Bording, T., Grombacher, D., Foged, N., Foley, N., Dugan, H. A., Doran, P. T., Mikucki, J., Tulaczyk, S., and Auken, E. (2022). Comparison of ground-based and airborne transient electromagnetic methods for mapping glacial and permafrost environments: Cases from mcmurdo dry valleys, antarctica. *Cold Regions Science and Technology*, 199.

Marescot, L., Loke, M., Chapellier, D., Delaloye, R., Lambiel, C., and Reynard, E. (2003). Assessing reliability of 2d resistivity imaging in mountain permafrost studies using the depth of investigation index method. *Near Surface Geophysics*, 1(2):57–67.

Martinelli, P. and Duplaá, M. C. (2008). Laterally filtered 1d inversions of small-loop, frequency-domain emi data from a chemical waste site. *Geophysics*, 73(4):143–149.

McNeill, J. D. (1990). Use of electromagnetic methods for groundwater studies. In *Geotechnical an Environmental Geophysics: Volume I: Review and Tutorial*, pages 191–218. Society of Exploration Geophysicists.

Miller, G. H., Alley, R. B., Brigham-Grette, J., Fitzpatrick, J. J., Polyak, L., Serreze, M. C., and White, J. W. (2010). Arctic amplification: can the past constrain the future? *Quaternary Science Reviews*, 29(15-16):1779–1790.

Minsley, B. J., Abraham, J. D., Smith, B. D., Cannia, J. C., Voss, C. I., Jorgenson, M. T., Walvoord, M. A., Wylie, B. K., Anderson, L., Ball, L. B., et al. (2012). Airborne electromagnetic imaging of discontinuous permafrost. *Geophysical Research Letters*, 39(2).

Mollaret, C., Hilbich, C., Pellet, C., Flores-Orozco, A., Delaloye, R., and Hauck, C. (2019). Mountain permafrost degradation documented through a network of permanent electrical resistivity tomography sites. *The Cryosphere*, 13(10):2557–2578.

Mollaret, C., Wagner, F. M., Hilbich, C., Scapozza, C., and Hauck, C. (2020). Petrophysical joint inversion applied to alpine permafrost field sites to image subsurface ice, water, air, and rock contents. *Frontiers in Earth Science*, 8:85.

Monteiro Santos, F. A., Triantafilis, J., and Bruzgulis, K. (2011). A spatially constrained 1d inversion algorithm for quasi-3d conductivity imaging: Application to dualem-421 data collected in a riverine plain. *Geophysics*, 76(2):43–53.

Niu, F., Luo, J., Lin, Z., Fang, J., and Liu, M. (2016). Thaw-induced slope failures and stability analyses in permafrost regions of the qinghai-tibet plateau, china. *Landslides*, 13(1):55–65.

Oldenburg, D. (1990). Inversion of electromagnetic data: An overview of new techniques. *Surveys in Geophysics*, 11(2):231–270.

Oldenburg, D. W. and Li, Y. (1999). Estimating depth of investigation in dc resistivity and ip surveys. *Geophysics*, 64(2):403–416.

Rücker, C., Günther, T., and Wagner, F. M. (2017). pyGIMLi: An open-source library for modelling and inversion in geophysics. *Computers and Geosciences*, 109:106–123.

Ruiz-Aguilar, D., Tezkan, B., and Arango-Galván, C. (2018). Exploration of the aquifer of san felipe geothermal area (mexico) by spatially constrained inversion of transient electromagnetic data. *Journal of Environmental and Engineering Geophysics*, 23(2):197–209.

Saey, T., De Smedt, P., Islam, M. M., Meerschman, E., Van De Vijver, E., Lehouck, A., and Van Meirvenne, M. (2012). Depth slicing of multi-receiver emi measurements to enhance the delineation of contrasting subsoil features. *Geoderma*, 189:514–521.

Schamper, C., Rejiba, F., and Guérin, R. (2012). 1d single-site and laterally constrained inversion of multifrequency and multicomponent ground-based electromagnetic induction data—application to the investigation of a near-surface clayey overburden. *Geophysics*, 77(4):19–35.

Schuur, E. A., Bockheim, J., Canadell, J. G., Euskirchen, E., Field, C. B., Goryachkin, S. V., Hagemann, S., Kuhry, P., Lafleur, P. M., Lee, H., et al. (2008). Vulnerability of permafrost carbon to climate change: Implications for the global carbon cycle. *BioScience*, 58(8):701–714.

Schuur, E. A., McGuire, A. D., Schädel, C., Grosse, G., Harden, J. W., Hayes, D. J., Hugelius, G., Koven, C. D., Kuhry, P., Lawrence, D. M., et al. (2015). Climate change and the permafrost carbon feedback. *Nature*, 520(7546):171–179.

Sengpiel, K.-P. and Siemon, B. (2000). Advanced inversion methods for airborne electromagnetic exploration. *Geophysics*, 65(6):1983–1992.

Serreze, M. C. and Barry, R. G. (2011). Processes and impacts of arctic amplification: A research synthesis. *Global and planetary change*, 77(1-2):85–96.

Siemon, B. (2009). Electromagnetic methods–frequency domain. In *Groundwater geophysics*, pages 155–178. Springer.

Siemon, B., Auken, E., and Christiansen, A. V. (2009). Laterally constrained inversion of helicopter-borne frequency-domain electromagnetic data. *Journal of Applied Geophysics*, 67(3):259–268.

Slater, A. G. and Lawrence, D. M. (2013). Diagnosing present and future permafrost from climate models. *Journal of Climate*, 26(15):5608–5623.

Smith, T., Hoversten, M., Gasperikova, E., and Morrison, F. (1999). Sharp boundary inversion of 2d magnetotelluric data. *Geophysical Prospecting*, 47(4):469–486.

Snieder, R. and Trampert, J. (1999). Inverse problems in geophysics. In *Wavefield inversion*, pages 119–190. Springer.

Socco, L. V., Boiero, D., Foti, S., and Wisén, R. (2009). Laterally constrained inversion of ground roll from seismic reflection records. *Geophysics*, 74(6):35–45.

Subcommittee, N. R. C. C. P. (1988). *Glossary of permafrost and related ground-ice terms*. National Research Council Canada.

Tartaras, E. and Beamish, D. (2006). Laterally constrained inversion of fixed-wing frequency-domain aem data. In *Near Surface 2006-12th EAGE European Meeting of Environmental and Engineering Geophysics*, pages cp–14. European Association of Geoscientists & Engineers.

Thalhammer, M. (2022). Hyperparameter optimization of 1d fdem inversions for permafrost imaging. *RWTH Aachen University, Research Module*, 54.PV00033. Unpublished.

Uhlemann, S., Dafflon, B., Peterson, J., Ulrich, C., Shirley, I., Michail, S., and Hubbard, S. S. (2021). Geophysical monitoring shows that spatial heterogeneity in thermohydrological dynamics reshapes a transitional permafrost system. *Geophysical Research Letters*, 48(6).

Van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., and Yu, T. (2014). scikit-image: image processing in python. *PeerJ*, 2.

Van Everdingen, R. O., Association, I. P., et al. (1998). *Multi-language glossary of permafrost and related ground-ice terms in chinese, english, french, german...* Arctic Inst. of North America University of Calgary.

Van Huissteden, J. (2020). *Thawing permafrost*. Springer.

Vest Christiansen, A. and Auken, E. (2012). A global measure for depth of investigation. *Geophysics*, 77(4):171–177.

Viezzoli, A., Auken, E., and Munday, T. (2009). Spatially constrained inversion for quasi 3d modelling of airborne electromagnetic data–an application for environmental assessment in the lower murray region of south australia. *Exploration Geophysics*, 40(2):173–183.

Viezzoli, A., Christiansen, A. V., Auken, E., and Sørensen, K. (2007). Spatially constrained inversion for quasi 3-d modelling of aem data. *ASEG Extended Abstracts*, 2007(1):1–4.

Viezzoli, A., Christiansen, A. V., Auken, E., and Sørensen, K. (2008). Quasi-3d modeling of airborne tem data by spatially constrained inversion. *Geophysics*, 73(3):105–113.

Vignoli, G., Fiandaca, G., Christiansen, A. V., Kirkegaard, C., and Auken, E. (2013). Sharp spatially constrained inversion (ssci) with applications to transient electromagnetic data. *Geophysical Prospecting*.

Vignoli, G., Guillemoteau, J., Barreto, J., and Rossi, M. (2021). Reconstruction, with tunable sparsity levels, of shear wave velocity profiles from surface wave data. *Geophysical Journal International*, 225(3):1935–1951.

Vignoli, G., Sapia, V., Menghini, A., and Viezzoli, A. (2017). Examples of improved inversion of different airborne electromagnetic datasets via sharp regularization. *Journal of Environmental and Engineering Geophysics*, 22(1):51–61.

Von Papen, M., Tezkan, B., and Israil, M. (2013). Characterization of an aquifer in roorkee, india using the spatially constrained inversion of in-loop tem data. *Near Surface Geophysics*, 11(1):85–94.

Wagner, F., Mollaret, C., Günther, T., Kemna, A., and Hauck, C. (2019). Quantitative imaging of water, ice and air in permafrost systems through petrophysical joint inversion of seismic refraction and electrical resistivity data. *Geophysical Journal International*, 219(3):1866–1875.

Wagner, F. M. and Uhlemann, S. (2021). An overview of multimethod imaging approaches in environmental geophysics. *Advances in Geophysics*, 62:1–72.

Walvoord, M. A. and Kurylyk, B. L. (2016). Hydrologic impacts of thawing permafrost—a review. *Vadose Zone Journal*, 15(6).

Wang, Z. and Bovik, A. C. (2009). Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE signal processing magazine*, 26(1):98–117.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612.

Ward, S. H. and Hohmann, G. W. (1988). Electromagnetic theory for geophysical applications. In *Electromagnetic Methods in Applied Geophysics: Voume 1, Theory*, pages 130–311. Society of Exploration Geophysicists.

West, G. and Macnae, J. (1991). Physics of the electromagnetic induction exploration method. In *Electromagnetic methods in applied geophysics: Volume 2, Application, Parts A and B*, pages 5–46. Society of Exploration Geophysicists.

Westermann, S., Duguay, C. R., Grosse, G., and Kääb, A. (2015). 13 remote sensing of permafrost and frozen ground.

Wisén, R., Auken, E., and Dahlin, T. (2005). Combination of 1d laterally constrained inversion and 2d smooth inversion of resistivity data with a priori data from boreholes. *Near Surface Geophysics*, 3(2):71–79.

Wisén, R. and Christiansen, A. V. (2005). Laterally and mutually constrained inversion of surface wave seismic data and resistivity data. *Journal of Environmental and Engineering Geophysics*, 10(3):251–262.

Wollschläger, U., Gerhards, H., Yu, Q., and Roth, K. (2010). Multi-channel ground-penetrating radar to explore spatial variations in thaw depth and moisture content in the active layer of a permafrost site. *The Cryosphere*, 4(3):269–283.

Wu, Y., Nakagawa, S., Kneafsey, T. J., Dafflon, B., and Hubbard, S. (2017). Electrical and seismic response of saline permafrost soil during freeze-thaw transition. *Journal of Applied Geophysics*, 146:16–26.

Zhdanov, M. S. (2015). *Inverse theory and applications in geophysics*, volume 36. Elsevier.

Appendix A

Inversion algorithm

This appendix contains the most important parts of the developed code for quasi-2D (Chapter A-1) and quasi-3D (Chapter A-2) modeling, respectively. Furthermore, is the implemented Delaunay triangulation algorithm presented in Chapter A-3.

A-1 Laterally constrained inversion class

```
from __future__ import annotations

import pygimli as pg
import numpy as np

class LCIModelling(pg.frameworks.LCModelling):
    """Laterally constrained inversion (LCI) class.

        Parameters
            -----------
            smooth inversion : smooth or blocky (few-layer) inversion; default: blocky inversion
            normalized inversion : default: not normalized
            fop : 1D forward operator
        """

    def __init__(self, smooth_inversion, normalized_inversion, fop, **kwargs):
        super().__init__(fop, **kwargs)

        self.smooth_inversion = smooth_inversion
        self.normalized_inversion = normalized_inversion

        if self.smooth_inversion == True:
            print("Laterally constrained smooth inversion")
        else:
            print("Laterally constrained blocky inversion")

        if self.normalized_inversion == True:
            print("Inversion is normalized")
        else:
            print("Inversion is not normalized")

    def initJacobian(self, dataVals, nLayers):
        """Initiate Jacobian matrix.

        Parameters
        ----------
        dataVals : list of 1D pyGIMLi data vectors (nSounding x Data per sounding) [list]
        nLayers : number of layers or smooth inversion respectively [int]
        """

        if self.smooth_inversion == True:
            nPar = 0 # number of model parameter regions +1
        else:
```

```
            nPar = 1 # number of model parameter regions +1

        nSoundings = len(dataVals) # number of soundings

        self.createParametrization(nSoundings, nLayers=nLayers, nPar=nPar)

        if self._jac is not None:
            self._jac.clear()
        else:
            self._jac = pg.matrix.BlockMatrix()

        self.fops1D = []
        nData = 0

        for i in range(nSoundings):
            kwargs = {}
            for key, val in self._fopKwargs.items():
                if hasattr(val, '__iter__'):
                    if len(val) == nSoundings:
                        kwargs[key] = val[i]
                    else:
                        kwargs[key] = [val]
                else:
                    kwargs[key] = val

            f = None
            if isinstance(self._fopTemplate, pg.frameworks.Modelling):
                f = self._fopTemplate
            elif issubclass(self._fopTemplate, pg.frameworks.Modelling):
                f = self._fopTemplate(**kwargs)
            else:
                f = type(self._fopTemplate)(self.verbose, **kwargs)

            f.setMultiThreadJacobian(self._parPerSounding)

            self._fops1D.append(f)

            nID = self._jac.addMatrix(f.jacobian())

            self._jac.addMatrixEntry(nID, nData, self._parPerSounding * i)
            print(self._jac)
            nData += len(dataVals[i])

        self._jac.recalcMatrixSize()
        print("Jacobian size:", self._jac.rows(), self._jac.cols(), "Amount of data points:", nData)
        self.setJacobian(self._jac)

    def response(self, par):
        """Cut together forward responses of all soundings."""

        mods = np.asarray(par).reshape(self._nSoundings, self._parPerSounding)

        resp = pg.Vector(0)
        for i in range(self._nSoundings):
            r = self._fops1D[i].response(mods[i])
            resp = pg.cat(resp, r)

        return resp

    def constraint_matrix(self, dataVals, nLayers=None, thk=None):
        """Create constraint matrix.

        Parameters
        ----------
        dataVals : list of 1D pyGIMLi data vectors (nSounding x Data per sounding) [list]
        nLayers : number of layers (for blocky inversion) [int]
        thk : starting model (for smooth model) [np.ndarray]
        """

        nSoundings = len(dataVals) # number of soundings

        if self.smooth_inversion == True:
            """Constraint matrix for smooth model."""

            par = len(thk) - 1 # number of model parameters

            boundaries_hor = (par - 1) * nSoundings # horizontal inner mesh boundaries
            boundaries_ver = par * (nSoundings - 1) # vertical inner mesh boundaries

            CM_np = np.zeros((boundaries_hor + boundaries_ver, (len(thk) - 1) * nSoundings))
            print("Constraint matrix size:", CM_np.shape)

            h = -np.eye(1, 2) + np.eye(1, 2, k=1)  # for horizontal boundaries
            v = -np.eye(1, len(thk)) + np.eye(1, len(thk), k=par)  # for vertical boundaries

            for ii in range(par):  # vertical boundaries
                for i in range(nSoundings - 1):
                    j = i * (par - 1)
                    CM_np[i + ii * (nSoundings - 1), j + i + ii: j + i + ii + v.shape[1]] = v
```

```
            for ii in range(par - 1):  # horizontal boundaries
                for i in range(nSoundings):
                    j = i * par
                    CM_np[boundaries_ver + ii + i * (par - 1), ii + j: ii + j + h.shape[1]] = h

        else:
            """Constraint matrix for blocky model."""

            boundaries_ver_thk = (nLayers - 1) * (nSoundings - 1) # vertical inner mesh boundaries
                for thicknesses
            boundaries_hor_thk = (nLayers - 2) * nSoundings # horizontal inner mesh boundaries for
                thicknesses

            boundaries_ver_res = nLayers * (nSoundings - 1) # vertical inner mesh boundaries for
                resistivities
            boundaries_hor_res = (nLayers - 1) * nSoundings # horizontal inner mesh boundaries for
                resistivities

            CM_thk_np = np.zeros((boundaries_hor_thk + boundaries_ver_thk, (nLayers * 2 - 1) *
                nSoundings))
            CM_res_np = np.zeros((boundaries_hor_res + boundaries_ver_res, (nLayers * 2 - 1) *
                nSoundings))

            h = -np.eye(1, 2) + np.eye(1, 2, k=1)  # for horizontal boundaries
            v = -np.eye(1, nLayers * 2) + np.eye(1, nLayers * 2, k=(nLayers * 2 - 1))  # for
                vertical boundaries

            # region 1: thicknesses: vertical and horizontal boundaries

            for ii in range(nLayers - 1):  # vertical boundaries
                for i in range(nSoundings - 1):
                    j = i * (nLayers * 2 - 2)
                    CM_thk_np[i + ii * (nSoundings - 1), j + i + ii: j + i + ii + v.shape[1]] = v

            for ii in range(nLayers - 2):  # horizontal boundaries
                for i in range(nSoundings):
                    j = i * (nLayers * 2 - 1)
                    CM_thk_np[boundaries_ver_thk + i + ii * nSoundings, j: j + h.shape[1]] = h

            # region 2: resistivities: vertical and horizontal boundaries

            for ii in range(nLayers):  # vertical boundaries
                for i in range(nSoundings - 1):
                    j = i * (nLayers * 2 - 2)
                    CM_res_np[i + ii * (nSoundings - 1),
                    j + i + (nLayers - 1) + ii: j + i + (nLayers - 1) + ii + v.shape[1]] = v

            for ii in range(nLayers - 1):  # horizontal boundaries
                for i in range(nSoundings):
                    j = i * (nLayers * 2 - 1)
                    CM_res_np[boundaries_ver_res + i + ii * nSoundings,
                    j + (nLayers - 1) + ii: j + (nLayers - 1) + ii + h.shape[1]] = h

            CM_np = np.concatenate((CM_thk_np, CM_res_np))

        print("Size constraint matrix:", np.shape(CM_np))

        self.CM = pg.utils.toSparseMatrix(CM_np)  # convert to sparse pg matrix

        return self.CM

    def createWeight(self, dataVals, cWeight_1, cWeight_2, nLayers=None, thk=None):
        """Create constraint weights (cWeights).
        Smooth model : vertical and horizontal constraint weights
        Blocky model : vertical constraint weights for both model parameter regions

        Parameters
        ----------
        dataVals : list of 1D pyGIMLi data vectors [list]
        cWeight_1 : vertical constraint weight (smooth model) or thickness constraint weight (blocky
            model) [float]
        cWeight_2 : horizontal constraint weight (smooth model) or resistivity constraint weight (
            blocky model) [float]
        nLayers : number of layers (for blocky inversion) [int]
        thk : starting model (for smooth model) [np.ndarray]
        """

        nSoundings = len(dataVals) # number of soundings

        if self.smooth_inversion == True:
            """Constraint weights for smooth model."""

            par = len(thk) - 1 # number of model parameters
            cWeight_ver = cWeight_1
            cWeight_hor = cWeight_2

            boundaries_ver = par * (nSoundings - 1)
            boundaries_hor = (par - 1) * nSoundings

            res_ver = pg.Vector(boundaries_ver, cWeight_ver)
```

```
            res_hor = pg.Vector(boundaries_hor, cWeight_hor)

            """Normalization"""
            if self.normalized_inversion == True:
                self.cWeight = pg.cat(res_ver, res_hor) / self.norm * (boundaries_hor+boundaries_ver
                    )
            else:
                self.cWeight = pg.cat(res_ver, res_hor)

        else:
            """Constraint weights for blocky model."""
            cWeight_ver_thk = cWeight_1
            cWeight_ver_res = cWeight_2

            boundaries_ver_thk = (nLayers - 1) * (nSoundings - 1)
            boundaries_hor_thk = (nLayers - 2) * nSoundings

            boundaries_ver_res = nLayers * (nSoundings - 1)
            boundaries_hor_res = (nLayers - 1) * nSoundings

            thk_ver = pg.Vector(boundaries_ver_thk, cWeight_ver_thk)
            thk_hor = pg.Vector(boundaries_hor_thk, 0)
            cWeight_thk = pg.cat(thk_ver, thk_hor)

            res_ver = pg.Vector(boundaries_ver_res, cWeight_ver_res)
            res_hor = pg.Vector(boundaries_hor_res, 0)

            cWeight_res = pg.cat(res_ver, res_hor)

            """Normalization"""
            if self.normalized_inversion == True:
                self.cWeight = pg.cat(cWeight_thk, cWeight_res) / self.norm * (boundaries_ver_thk+
                    boundaries_ver_res)
            else:
                self.cWeight = pg.cat(cWeight_thk, cWeight_res)

        print("Constraint cWeight:", self.cWeight)

    def createConstraints(self):
        """Create weighted constraint matrix."""

        self._CW = pg.matrix.LMultRMatrix(self.CM, self.cWeight)  # , verbose=True)
        self.setConstraints(self._CW)

    def normalization(self, dataVals, cWeight_1, cWeight_2, phiM_norm, nLayers=None, thk=None):
        """Create normalization factor.

        Parameters
        ----------
        dataVals : list of 1D pyGIMLi data vectors [list]
        cWeight_1 : vertical constraint weight (smooth model) or thickness constraint weight (blocky
            model) [float]
        cWeight_2 : horizontal constraint weight (smooth model) or resistivity constraint weight (
            blocky model) [float]
        phiM_norm : model objective function calculated by multiplying inversion starting model with
            constraint matrix for 4 cases (cWeight_1 and cWeight_2 set to 0 and 1, respectively) [
            list[float]]
        nLayers : number of layers (for blocky inversion) [int]
        thk : starting model (for smooth model) [np.ndarray]
        """

        if self.smooth_inversion == True:
            """Normalization factor for smooth model."""

            cWeight_ver = cWeight_1
            cWeight_hor = cWeight_2

            cWeight_ratio_hor = cWeight_hor / (cWeight_ver + cWeight_hor)
            cWeight_ratio_ver = cWeight_ver / (cWeight_ver + cWeight_hor)

            if cWeight_hor == 0 and cWeight_ver == 0:
                print("cWeights are zero")
                slope = phiM_norm[0]  # if both cWeights = 0
                self.norm = 0
                raise ValueError("At least one cWeight should be non-zero!")

            elif cWeight_hor == cWeight_ver and cWeight_ver != 0:
                print("cWeights are equal")
                slope = phiM_norm[3]  # if both cWeights = 1
                self.norm = slope * cWeight_hor

            elif cWeight_hor == 0 and cWeight_ver != 0:
                print("only vertical cWeight")
                slope = phiM_norm[1]  # if cWeight_hor = 0
                self.norm = slope * cWeight_ver

            elif cWeight_ver == 0 and cWeight_hor != 0:
                print("only horizontal cWeight")
                slope = phiM_norm[2]  # if cWeight_ver = 0
                self.norm = slope * cWeight_hor
```

```
            elif cWeight_ver < cWeight_hor:
                print("horizontal cWeight is dominant")
                slope_ver = phiM_norm[1]  # if cWeight_hor = 0
                slope_hor = phiM_norm[2]  # if cWeight_ver = 0
                self.norm = slope_hor * cWeight_hor + slope_ver * cWeight_ver * cWeight_ratio_ver

            else: # cWeight_ver > cWeight_hor
                print("vertical cWeight is dominant")
                slope_ver = phiM_norm[1]  # if cWeight_hor = 0
                slope_hor = phiM_norm[2]  # if cWeight_ver = 0
                self.norm = slope_ver * cWeight_ver + slope_hor * cWeight_hor * cWeight_ratio_hor

        else:
            """Normalization factor for blocky model."""

            cWeight_ver_thk = cWeight_1
            cWeight_ver_res = cWeight_2

            cWeight_ratio_thk = cWeight_ver_thk / (cWeight_ver_res + cWeight_ver_thk)
            cWeight_ratio_res = cWeight_ver_res / (cWeight_ver_res + cWeight_ver_thk)

            if cWeight_ver_thk == 0 and cWeight_ver_res == 0:
                print("cWeights are zero")
                slope = phiM_norm[0]  # if both cWeights = 0
                self.norm = 0
                raise ValueError("At least one cWeight should be non-zero!")

            elif cWeight_ver_thk == cWeight_ver_res and cWeight_ver_thk != 0:
                print("cWeights are equal")
                slope = phiM_norm[3]  # if both cWeights = 1
                self.norm = slope * cWeight_ver_thk

            elif cWeight_ver_res == 0 and cWeight_ver_thk != 0:
                print("only cWeight for thickness")
                slope = phiM_norm[2]  # if cWeight_res = 0
                self.norm = slope * cWeight_ver_thk

            elif cWeight_ver_thk == 0 and cWeight_ver_res != 0:
                print("only cWeight for resistivity")
                slope = phiM_norm[1]  # if cWeight_thk = 0
                self.norm = slope * cWeight_ver_res

            elif cWeight_ver_thk < cWeight_ver_res:
                print("cWeight for resistivity is dominant")
                slope_res = phiM_norm[1]  # if cWeight_thk = 0
                slope_thk = phiM_norm[2]  # if cWeight_res = 0
                self.norm = slope_res * cWeight_ver_res + slope_thk * cWeight_ver_thk *
                    cWeight_ratio_thk

            else:  # cWeight_ver_thk > cWeight_ver_res
                print("cWeight for thickness is dominant")
                slope_res = phiM_norm[1]  # if cWeight_thk = 0
                slope_thk = phiM_norm[2]  # if cWeight_res = 0
                self.norm = slope_thk * cWeight_ver_thk + slope_res * cWeight_ver_res *
                    cWeight_ratio_res

            print("Normalization", self.norm)

    def drawModel(self, ax, model, **kwargs):
        if self.smooth_inversion == True:
            valPerSounding = self._parPerSounding * 2 - 1

            mods = np.asarray(model).reshape(self._nSoundings, valPerSounding)
            pg.viewer.mpl.showStitchedModels(mods, ax=ax, useMesh=True, x=self.soundingPos, **kwargs
                )
        else:
            mods = np.asarray(model).reshape(self._nSoundings, self._parPerSounding)
            pg.viewer.mpl.showStitchedModels(mods, ax=ax, useMesh=True, x=self.soundingPos, **kwargs
                )
```

A-2 Spatially constrained inversion class

```
from __future__ import annotations

import pygimli as pg
import numpy as np


import ClassLCI
"""Import laterally constrained inversion (LCI) class."""

class SCIModelling(ClassLCI.LCIModelling):
    """Spatially constrained inversion (SCI) class.

    Parameters
```

```
    -----------
    smooth inversion : smooth or blocky (few-layer) inversion; default: blocky inversion
    normalized inversion : default: not normalized
    fop : 1D forward operator
"""

def __init__(self, smooth_inversion, normalized_inversion, fop, **kwargs):
    super().__init__(smooth_inversion, normalized_inversion, fop, **kwargs)

    self.smooth_inversion = smooth_inversion
    self.normalized_inversion = normalized_inversion

    if self.smooth_inversion == True:
        print("Spatially constrained smooth inversion")
    else:
        print("Spatially constrained blocky inversion")

    if self.normalized_inversion == True:
        print("Inversion is normalized")
    else:
        print("Inversion is not normalized")


def constraint_matrix(self, dataVals, neighbors, nLayers=None, thk=None):
    """Create constraint matrix.

    Parameters
    -----------
    dataVals : list of 1D pyGIMLi data vectors (nSounding x Data per sounding) [list]
    neighbors : indeces of direct neighbors [list]
    nLayers : number of layers (for blocky inversion) [int]
    thk : starting model (for smooth model) [np.ndarray]
    """

    nSoundings = len(dataVals) # number of soundings

    if self.smooth_inversion == True:
        """Constraint matrix for smooth model."""

        par = len(thk) - 1 # number of model parameters

        boundaries_hor = (par - 1) * nSoundings # horizontal inner mesh boundaries
        boundaries_ver = 0 # vertical inner mesh boundaries

        for i in range(len(neighbors)):
            amount_neighbors = len(neighbors[i])
            boundaries = par * amount_neighbors
            boundaries_ver += boundaries

        CM_np = np.zeros((boundaries_hor + boundaries_ver, par * nSoundings))
        print("Constraint matrix size:", CM_np.shape)

        """Vertical boundaries"""

        neighbors_ver = 0

        for iii in range(len(neighbors)):
            neighbors_direct = neighbors[iii] # neighbors of one sounding
            for ii in range(len(neighbors_direct)):
                neighbor_position = neighbors_direct[ii] # indeces of neighbors
                neighbor_distance = neighbor_position - iii # index difference to nearest
                    neighbor
                for i in range(par):
                    v = -np.eye(1, len(thk) + (neighbor_distance - 1) * par) + np.eye(1, len(thk
                        ) + (neighbor_distance - 1) * par, k=par * neighbor_distance)
                    CM_np[neighbors_ver * par + i, i + iii * par: i + iii * par + v.shape[1]] =
                        v
                neighbors_ver += 1

        """Horizontal boundaries"""

        h = -np.eye(1, 2) + np.eye(1, 2, k=1)

        for ii in range(par - 1):
            for i in range(nSoundings):
                j = i * par
                CM_np[boundaries_ver + ii + i * (par - 1), ii + j: ii + j + h.shape[1]] = h #
                    constraint matrix

    else:
        """Constraint matrix for blocky model."""

        par = nLayers * 2 - 1 # number of model parameters

        """Horizontal boundaries for thicknesses (thk) and resistivities (res)"""

        boundaries_hor_thk = (nLayers - 2) * nSoundings
        boundaries_hor_res = (nLayers - 1) * nSoundings

        """Vertical boundaries for thicknesses (thk) and resistivities (res)"""
```

```
            boundaries_ver_thk = 0
            boundaries_ver_res = 0

            for i in range(len(neighbors)):
                amount_neighbors = len(neighbors[i])
                boundaries_thk = (nLayers - 1) * amount_neighbors
                boundaries_res = nLayers * amount_neighbors
                boundaries_ver_thk += boundaries_thk
                boundaries_ver_res += boundaries_res

            CM_thk_np = np.zeros((boundaries_hor_thk + boundaries_ver_thk, par * nSoundings))

            CM_res_np = np.zeros((boundaries_hor_res + boundaries_ver_res, par * nSoundings))

            """Vertical boundaries"""

            """Model parameter: thickness"""

            neighbors_ver = 0

            for iii in range(len(neighbors)):
                neighbors_direct = neighbors[iii] # neighbors of one sounding
                for ii in range(len(neighbors_direct)):
                    neighbor_position = neighbors_direct[ii] # indeces of neighbors
                    neighbor_distance = neighbor_position - iii # index difference to nearest
                        neighbor
                    for i in range(nLayers - 1):
                        v = -np.eye(1, nLayers * 2 + (neighbor_distance - 1) * par) + np.eye(1,
                            nLayers * 2 + (
                                neighbor_distance - 1) * par, k=par * neighbor_distance)
                        CM_thk_np[neighbors_ver * (nLayers - 1) + i, i + iii * par: i + iii * par +
                            v.shape[1]] = v
                    neighbors_ver += 1

            """Model parameter: resistivity"""

            neighbors_ver = 0

            for iii in range(len(neighbors)):
                neighbors_direct = neighbors[iii] # neighbors of one sounding
                for ii in range(len(neighbors_direct)):
                    neighbor_position = neighbors_direct[ii] # indeces of neighbors
                    neighbor_distance = neighbor_position - iii # index difference to nearest
                        neighbor
                    for i in range(nLayers):
                        v = -np.eye(1, nLayers * 2 + (neighbor_distance - 1) * par) + np.eye(1,
                            nLayers * 2 + (neighbor_distance - 1) * par, k=par * neighbor_distance)
                        CM_res_np[neighbors_ver * nLayers + i, i + iii * par: i + iii * par + v.
                            shape[1]] = v
                    neighbors_ver += 1

            """Horizontal boundaries"""

            h = -np.eye(1, 2) + np.eye(1, 2, k=1)

            """Model parameter: thickness"""

            for ii in range(nLayers - 2):
                for i in range(nSoundings):
                    CM_thk_np[boundaries_ver_thk + i + ii * nSoundings, i * par: i * par + h.shape
                        [1]] = h

            """Model parameter: resistivity"""

            for ii in range(nLayers - 1):
                for i in range(nSoundings):
                    CM_res_np[boundaries_ver_res + i + ii * nSoundings,
                    i * par + (nLayers - 1) + ii: i * par + (nLayers - 1) + ii + h.shape[1]] = h

            CM_np = np.vstack((CM_thk_np, CM_res_np))

        """Convert to sparse pyGIMLi matrix"""
        self.CM = pg.utils.toSparseMatrix(CM_np)

        return self.CM


    def distance_weights(self, distances, ref_distance, ref_constraint, exp_distance):
        """Create distance-dependent constraint weights [Viezzoli et al., 2007].

        Parameters
        -----------
        distances : average distance between nearest neighbors [list]
        ref_distance : Euclidian distances to nearest neighbors for all soundings [list]
        ref_constraint : reference constraint value A [float]
        exp_distance : exponent a [float]
        """

        self.constraint_strengths = []
```

```
        for distance in distances:
            constraint_strength = 1 + (ref_constraint - 1) * (distance / ref_distance) **
                exp_distance
            self.constraint_strengths.append(constraint_strength)

        print("Distance weights:", self.constraint_strengths)

        return self.constraint_strengths

    def createWeight(self, dataVals, neighbors, cWeight_1, cWeight_2, nLayers=None, thk=None):
        """Create constraint weights (cWeights).
        Smooth model : vertical and horizontal constraint weights
        Blocky model : vertical constraint weights for both model parameter regions

        Parameters
        ----------
        dataVals : list of 1D pyGIMLi data vectors [list]
        neighbors : indeces of direct neighbors [list]
        cWeight_1 : thickness constraint weight (blocky model) [float]
        cWeight_2 : horizontal constraint weight (smooth model) or resistivity constraint weight (
            blocky model) [float]
        nLayers : number of layers (for blocky inversion) [int]
        thk : starting model (for smooth model) [np.ndarray]
        """

        nSoundings = len(dataVals) # number of soundings

        if self.smooth_inversion == True:
            """Constraint weights for smooth model."""

            par = len(thk) - 1 # number of model parameters
            cWeight_hor = cWeight_2

            boundaries_hor = (par - 1) * nSoundings
            boundaries_ver = 0

            for i in range(len(neighbors)):
                amount_neighbors = len(neighbors[i])
                boundaries = par * amount_neighbors
                boundaries_ver += boundaries

            res_ver = pg.Vector(boundaries_ver, 0)

            for i in range(len(self.constraint_strengths)):
                for ii in range(par):
                    res_ver[i * par + ii] = self.constraint_strengths[i]

            res_hor = pg.Vector(boundaries_hor, cWeight_hor)
            self.cWeight = pg.cat(res_ver, res_hor)

        else:
            """Constraint weights for blocky model."""

            cWeight_ver_thk = cWeight_1
            cWeight_ver_res = cWeight_2

            boundaries_hor_thk = (nLayers - 2) * nSoundings
            boundaries_hor_res = (nLayers - 1) * nSoundings

            boundaries_ver_thk = 0
            boundaries_ver_res = 0

            for i in range(len(neighbors)):
                amount_neighbors = len(neighbors[i])
                boundaries_thk = (nLayers - 1) * amount_neighbors
                boundaries_res = nLayers * amount_neighbors
                boundaries_ver_thk += boundaries_thk
                boundaries_ver_res += boundaries_res

            thk_ver = pg.Vector(boundaries_ver_thk, 0)
            res_ver = pg.Vector(boundaries_ver_res, 0)

            for i in range(len(self.constraint_strengths)):
                for ii in range(nLayers - 1):
                    thk_ver[i * (nLayers - 1) + ii] = self.constraint_strengths[i] * cWeight_ver_thk
                for iii in range(nLayers):
                    res_ver[i * nLayers + iii] = self.constraint_strengths[i] * cWeight_ver_res

            thk_hor = pg.Vector(boundaries_hor_thk, 0)
            cWeight_thk = pg.cat(thk_ver, thk_hor)

            res_hor = pg.Vector(boundaries_hor_res, 0)
            cWeight_res = pg.cat(res_ver, res_hor)

            self.cWeight = pg.cat(cWeight_thk, cWeight_res)
```

A-3 Delaunay triangulation

```
from __future__ import annotations

import numpy as np

from scipy.spatial import Delaunay, distance

def generate_grid(
        lateral_distance: float = None,
        lateral_soundings: int = None,
        vertical_distance: float = None,
        vertical_soundings: int = None
):
    """Delaunay triangulation algorithm.
        Objective: generate 2D grid (x,y locations) of observation points and impose a Delaunay
            triangulation

        Parameters
        ----------
        lateral_distance : length of grid in x-direction
        lateral_soundings : number of observation points in x-direction [int]
        vertical_distance : length of grid in y-direction [float]
        vertical_soundings : number of observation points in y-direction [int]
        """

    soundings_lateral = np.linspace(0, lateral_distance, lateral_soundings)
    soundings_vertical = np.linspace(0, vertical_distance, vertical_soundings)

    """Generate a meshgrid."""

    x, y = np.meshgrid(soundings_lateral, soundings_vertical)

    """Generate an array of coordinate pairs."""

    grid_pairs = np.empty((np.size(x), 2))

    for i in range(x.shape[0]):
        for j in range(x.shape[1]):
            grid_pairs[j + i * x.shape[1], :] = x[i,j], y[i,j]

    """Impose Delaunay triangulation."""

    tri = Delaunay(grid_pairs)

    return tri, grid_pairs

def find_neighbors(
        tri: scipy.spatial = None,
        grid_pairs: np.ndarray = None
):
    """Delaunay triangulation algorithm.
        Objective: find the nearest neighbors of observation points and their respective Euclidian
            distance based on the Delaunay triangulation

        Parameters
        ----------
        tri : Delaunay object [scipy.spatial]
        grid_pairs : x,y coordinate data of observation points (nSoundings x 2) [np.ndarray]
        """

    simplex_neighbors = {}
    for pair in range(grid_pairs.shape[0]):
        simplex_neighbors[pair] = []

    for simplex in tri.simplices:
        simplex_neighbors[simplex[0]] += [simplex[1],simplex[2]]
        simplex_neighbors[simplex[1]] += [simplex[2],simplex[0]]
        simplex_neighbors[simplex[2]] += [simplex[0],simplex[1]]

    neighbors_list = []

    """Find neighbors of an observation point within a Delunay triangle."""

    for i in range(len(simplex_neighbors)):
        simplex_neighbor = simplex_neighbors[i]
        neighbors = []
        for neighbor_value in simplex_neighbor:
            if neighbor_value not in neighbors and neighbor_value > i: # exclude all neighbours with
                boundaries already in list
                neighbors.append(neighbor_value)
        neighbors_list.append(neighbors)

    """Calculate distance between neighboring and adjacent points."""

    neighbors_distance_list = []

    for i in range(len(neighbors_list)):
        neighbors_direct = neighbors_list[i]
```

```
        for ii in range(len(neighbors_direct)):
            neighbor_position = neighbors_direct[ii]
            neighbor_distance = distance.euclidean(grid_pairs[i], grid_pairs[neighbor_position])
            neighbors_distance_list.append(neighbor_distance)
                                                                                                    89
    neighbors_adjacent_distance = sum(neighbors_distance_list) / len(neighbors_distance_list)

    return neighbors_list, neighbors_distance_list, neighbors_adjacent_distance
```

Appendix B

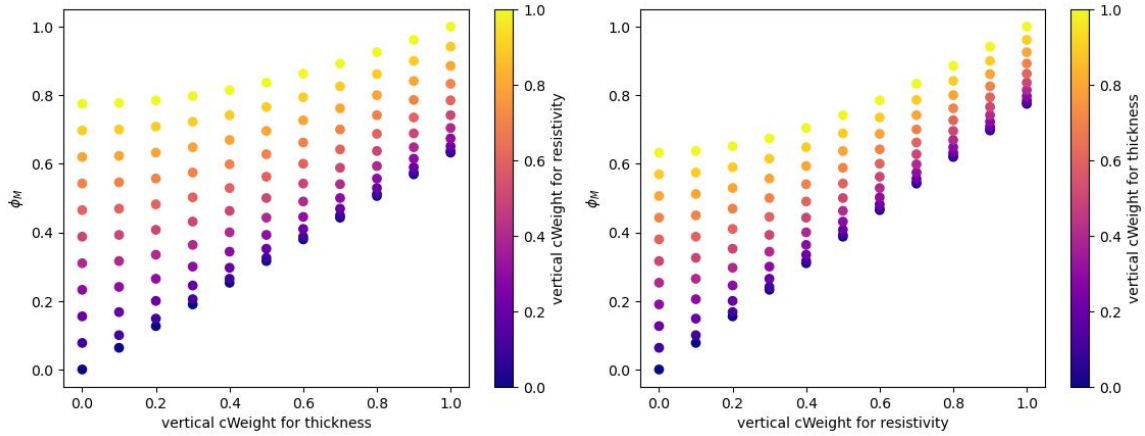
Normalization

Table B-1: Computation of the model objective function Φ_M for three different combinations of constraint weights (cW), which is incorporated in the formula of the normalization factor (cf. Table B-2) applied to the weighted constraint matrix.

Model objective function Φ_M	cWeight 1 (cW_1)	cWeight 2 (cW_2)
$\Phi_{M,1}$	$\neq 0$	$= 0$
$\Phi_{M,2}$	$= 0$	$\neq 0$
$\Phi_{M,1,2}$	$= 1$	$= 1$

Table B-2: Formula for the normalization factor (in short norm) defined for six different ratios of the constraint weights cW_1 and cW_2 utilizing the computed model objective function Φ_M and the total number of inner mesh boundaries b between model parameters.

Case	Formula for normalization
$cW_1 = 0$, $cW_2 = 0$	not normalized
$cW_1 = cW_2$, $cW_1 \neq 0$	$\text{norm} = (\Phi_{M,1,2} * cW_1)/b$
$cW_1 = 0$, $cW_2 \neq 0$	$\text{norm} = (\Phi_{M,2} * cW_2)/b$
$cW_1 \neq 0$, $cW_2 = 0$	$\text{norm} = (\Phi_{M,1} * cW_1)/b$
$cW_1 > cW_2$	$\text{norm} = (\Phi_{M,1} * cW_1 + \Phi_{M,2} * cW_2 * cW_{2,rel})/b$
$cW_1 < cW_2$	$\text{norm} = (\Phi_{M,2} * cW_2 + \Phi_{M,1} * cW_1 * cW_{1,rel})/b$



(a) **(b)**

Figure B-1: Model objective function Φ_M with respect to constraint weights (i.e., cWeights) for both model parameters (i.e., (a) thicknesses and (b) resistivities) based on a three-layered blocky model with 11 laterally constrained soundings. Φ_M values are displayed relative to the highest Φ_M when corresponding cWeights $= 1$.