PADERBORN UNIVERSITY

The University for the Information Society

Increased Sensitivity in Parameter Identification Problems for Piezoelectrics

Von der Fakultät für Elektrotechnik, Informatik und Mathematik

der Universität Paderborn

zur Erlangung des akademischen Grades

DOKTOR DER NATURWISSENSCHAFTEN

- Dr. rer. nat. -

genehmigte Dissertation

von

Benjamin Jurgelucks

Tag der mündlichen Prüfung: 29.1.2019

Gutachter:
Prof. Dr. Andrea Walther
Univ.-Prof. Dipl.-Ing. Dr. Barbara Kaltenbacher
Jun.-Prof. Dr. Tom Lahmer

Betreuer:
Prof. Dr. Andrea Walther
Priv.-Doz. Dr. Kshitij Kulshreshtha

Acknowledgments

First and foremost, I would like to thank my first advisor Prof. Dr. Andrea Walther for her guidance and support, the many helpful conversations leading to new ideas and the patience she has had with me. I am very grateful for the opportunity she has given me and allowing me to follow the leads and ideas I had. I am thankful for the fruitful and stimulating research environment she has provided.

I would like to express my great appreciation to my second advisor Priv.-Doz. Dr. Kshitij Kulshreshtha for the many helpful discussions and advice on and around many issues that have arisen during this work.

Furthermore, I would like to thank Univ.-Prof. Dipl.-Ing. Dr. Barbara Kaltenbacher and Jun.-Prof. Dr. Tom Lahmer for their interest in my work and their willingness to be referees of this thesis.

I wish to express my gratitude to Prof. Dr.-Ing. Bernd Henning and the Measurement Engineering Group who provided advice on the physical behavior of piezoceramics and real measurement data.

I would like to thank Univ.-Prof. Dr. techn. Dr.-Ing. habil. Manfred Kaltenbacher who granted me access to the source code of the simulation tool CFS++.

I would also like to thank all past and present colleagues of the research group Mathematics and its Applications I have had the privilege to work with. I enjoyed and learned much from our long and fruitful conversations and I greatly appreciate the daily motivation. My thanks also extends to the past and present members of the Measurement Engineering Group which I have had the pleasure to work with.

Last but not least, I thank all proofreaders for their precision work.

Abstract

For the development of new piezoelectric devices computer simulations have become an essential part of the process. In order for the computer simulations to give correct predictions it is of grave importance to know the correct material parameter values for the material that is simulated. However, material parameter values provided by the manufacturer or by common characterization methods are error-prone with errors ranging up to 20%. Many of these methods are based on a solution of an inverse problem where simulations of the electrical impedance and measurements thereof are compared in an iterative process to find the correct values. A common complaint for these methods is that the sensitivity of impedance with respect to certain material parameters is low or zero.

In this thesis the possibility to increase the sensitivity of impedance with respect to material parameters using a triple-ring electrode setup is analyzed. Theoretical results concerning existence, uniqueness and regularity of solutions to the governing equations are given. The sensitivity is increased in an optimization problem where the optimization method is supplemented with accurate derivatives computed via Algorithmic Differentiation. Numerical results are presented. An optimized electrode geometry for a triple-ring electrode piezoceramic was developed in this thesis and is currently used for real physical measurements leading to novel material parameter characterization methods.

Keywords: Optimization, Piezoelectricity, Material Parameter Characterization, Sensitivity, Algorithmic Differentiation

Zusammenfassung

Für die Entwicklung neuer piezoelektrischer Geräte sind Computersimulationen unumgänglich geworden. Damit die Simulationen korrekte Vorhersagen treffen können ist es wichtig, dass diese mit den korrekten Materialparameterwerten des zu simulierenden Materials instanziiert werden. Materialparameterwerte werden üblicherweise durch den Hersteller oder durch neuere Methoden der Materialparametercharakterisierung quantisiert. Diese Werte sind häufig fehleranfällig mit Fehlern von bis zu 20%. Viele dieser Methoden basieren auf der Lösung eines sogenannten Inversen Problems bei dem Messungen und Simulationen der elektrischen Impedanz iterativ verglichen werden, um auf die ursächlichen korrekten Materialparameter zu schließen. Eines der Probleme dieses Ansatzes ist jedoch, dass die Sensitivität der Impedanz bezüglich einiger Materialparameter sehr klein oder gar Null beträgt.

In dieser Dissertation wird die Möglichkeit der Steigerung der Sensitivität bezüglich der Materialparameter unter Verwendung von Piezokeramiken mit drei Elektrodenringen erörtert. Theoretische Aussagen bezüglich der Existenz, Eindeutigkeit und Regularität von Lösungen von Partiellen Differentialgleichungen, die das Verhalten der Piezokeramik vorhersagen, werden gegeben. Die Sensitivität wird in einem Optimierungsproblem gesteigert. Verwendete Optimierungsverfahren werden durch präzise Ableitungen ergänzt welche mit Algorithmischem Differenzieren berechnet wurden. Dies wird auch anhand zahlreicher numerischer Resultate dargestellt. Schließlich wurde im Rahmen dieser Dissertation eine optimale Geometrie für Piezokeramiken mit drei Elektrodenringen ermittelt. Diese Piezokeramik wird zur Zeit für reale Messungen, die zu neuen Materialparametercharakterisierungsmethoden führen, genutzt.

Stichworte: Optimierung, Piezoelektrizität, Materialparameterbestimmung, Sensitivität, Algorithmisches Differenzieren

Contents

List of Figures

1

Introduction

1.1 Piezoelectric Applications

Piezoelectric devices are an essential component for a vast number of electrical appliances and have been hugely important for technological advancement in the past century. Multiple piezoelectric devices can be found in almost all households where the nearest piezoelectric device is often located only a couple of millimeters away from a person. Examples for such appliances using piezoelectrical devices are mobile telephones, headphones or hearing aides, electrical igniters or more critical medical devices such as heart rate monitors or drug delivery micropumps. The fundamental property of piezoelectric devices as used in these appliances is that they can transform electrical energy into kinetic energy and vice versa. This property will be further explored and explained in the upcoming chapters. In Fig. 1.1 some popular areas of application for piezoelectrics are given.

Industrial: Engine Knock Sensors, Pressure Sensors, Sonar Equipment, Diesel Fuel Injectors, Fast Response Solenoids, Optical Adjustment, Ultrasonic Cleaning, Ultrasonic Welding, Piezoelectric Motors, Stack Actuators, Stripe Actuators, Piezoelectric Relays, ...

Medical: Ultrasound Imaging, Ultrasonic Procedures, ...

Consumer Electronics: Piezoelectric Printers, Piezoelectric Speakers, Piezoelectric Buzzers, Piezoelectric Humidifiers, Electronic Toothbrushes, ...

Musical Applications: Instrument Pickups, Microphones, ...

Defense Applications: Micro Robotics, Course-changing Bullets, ...

Other Applications: Piezoelectric Igniters, Electricity Generation, Microelectronic Mechanical Systems (MEMS), Tennis Racquets, ...

Figure 1.1: Popular uses for piezoceramics according to [4].

1.2 Motivation and Existing Methods

When designing new piezoelectrical devices it is fundamental to be able to predict the behavior of the device. An increasingly important means of prediction is a computer simulation based on a simulation model. However, for the model to give valid predictions it is essential that the material parameters of the piezoelectric material are accurate or the device may behave differently than expected. These material parameters are usually supplied by the manufacturer of the material in data sheets and are oftentimes acquired by deploying variations of approaches based on the *1987 IEEE Standard On Piezoelectricity* [69]. However, for piezoelectric materials some material parameter data provided by manufacturers may be inaccurate with errors of up to 20% [54]. Hence, for the accurate simulation of new piezoelectrical devices it is important to also have an alternative source for these material parameters.

Let us assume we are handed a specimen of a piezoelectric ceramic e.g. by a manufacturing company or an entity aiming to utilize the ceramic in some fashion. Then, in order to predict the behavior of the ceramic in its desired area of application all relevant material parameters of this specimen are required. It is possible to directly measure some material parameters like the size or weight of the specimen, however, this is not possible for all parameters. Hence, if some material parameters cannot be directly measured then they must be reconstructed in a different manner. Some existing methods rely on determining the material parameters by solving a so-called inverse problem. In this inverse problem the aim is to fit a measured quantity (such as the frequency dependent electrical impedance curve) and a simulation of this quantity varying the material parameters. If the simulated and measured curves fit to a certain degree then it is assumed that the material parameters of the specimen are identical to the material parameters of the simulation. However, often this may not be the case.

We consider an example for a disk-shaped piezoelectric ceramic such as displayed in Fig. 1.2. By varying one material parameter value in a simulation we obtain two impedance curves as a result. These impedance curves are displayed in Fig. 1.3. These curves were obtained by two simulations of a piezoceramic with fully covering electrodes on top and bottom (see also Fig. 3.1 for exact setup). Furthermore, the only difference in the simulation configurations is that in the second simulation the

parameter ϵ_{11}^S was modified to be 10% larger than in the first simulation. We note that there is barely any difference in the impedance curve even though the material parameters are significantly different. This poses a serious problem when trying to identify material parameters by fitting curves: If the curves already fit with wrong parameter values then we cannot hope to retrieve the correct values or at least have little confidence in the result.

(a) Representation of typical disk-shaped piezoelectric ceramic.

(b) Representation of typical disk-shaped piezoelectric ceramic with front quarter cut out.
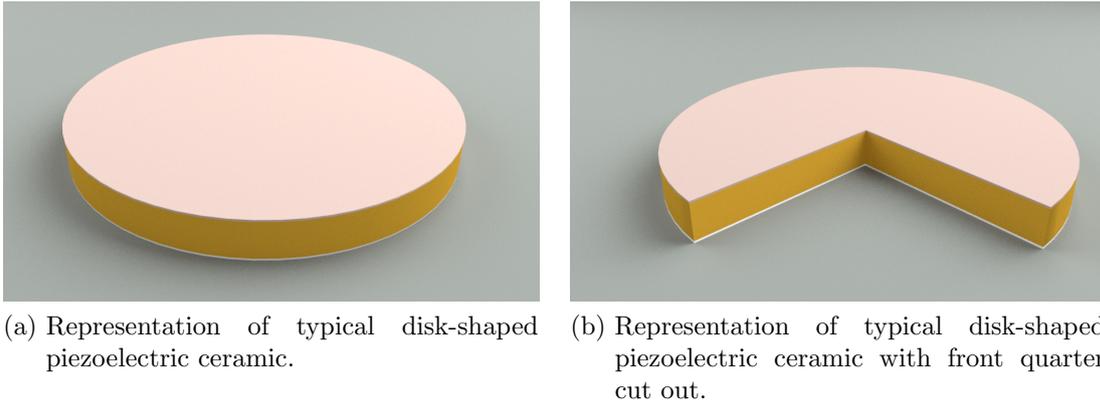
Figure 1.2: Piezoelectric ceramic (brown) with attached electrodes (metallic gray) fully covering top and bottom.

A different interpretation of this issue and a common complaint in the literature for such parameter identification methods in piezoelectricity is the fact that some material parameters display a sensitivity of the measurable quantity (such as the electrical impedance) which is small or close to zero (e.g. [63]). Most authors then exclude these material parameters from their material parameter characterization methods as they cannot be identified reliably.

In order to overcome this issue some authors have suggested methods to increase the sensitivity:

- Lahmer and coauthors have multiple publications in this area. In e.g. [40] the sensitivity is increased by selecting measurement frequencies which are used to compare the simulated and measured impedance curves as part of an optimization problem. Additional in [41] the sensitivity is increased by optimizing the excitation frequencies. These approaches use ideas from classical optimal experiment design. In [39] an efficient strategy to measure the sensitivity globally for many different material parameters is presented which could be
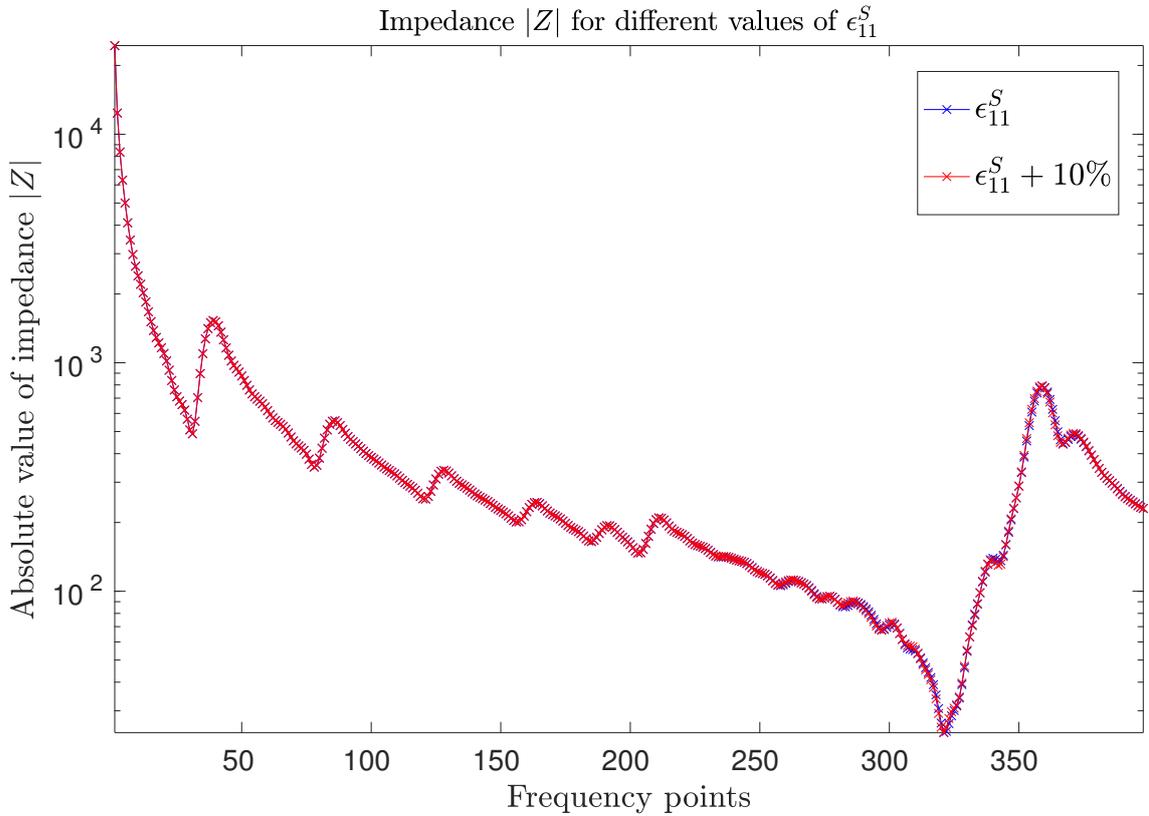
Figure 1.3: Simulation of impedance curves for different values of ϵ_{11}^S. It is very hard to visually find a difference in the two curves.

combined with different local sensitivity optimization strategies.

- Work in [45] and related articles use additional ultrasonic acoustical measurements to enhance the sensitivity.

- In [28] and related articles ten different specimens with varying diameter to thickness ratios and a method based on harmonic overtones was used.

- In [7] and related articles the method of moving asymptotes (MMA) was applied to determine the material parameters based on differences in electrical conductance and electrical resistance.

- Methods proposed by e.g. Rupitsch, Lerch, Ilg and Weiß in e.g. [61],[62] and related articles use multiple differently shaped piezoceramic specimens and additional measurements of the surface displacement as opposed to only using impedance measurements. Sensitivities based on additional information such

as the surface displacement are generally higher than those solely based on impedance measurements.

- The author also refers to Perez [54],[55] and the recent contribution of Rupitsch [61] for an overview of current developments in the area of material parameter characterizations of piezoceramics which are not necessarily directly related to increasing the sensitivity of the measurable quantity.

Some of the previous methods raise issues: Many of the approaches mentioned utilize multiple differently shaped specimens. However, this may lead to inconsistent data as the material parameters could vary from specimen to specimen. Furthermore, some of the approaches use additional surface displacement measurements. These measurements are usually made by laser vibrometers. However, as the surface displacements are comparatively small, measurements made by laser vibrometers may be wrongfully influenced by surrounding vibrations such as cars passing by. As some material parameters commonly have very low to zero influence small changes due to even small measurement errors could lead to incorrectly identified material parameters. Furthermore, the equipment required to make such additional measurements is often very expensive compared to equipment required for only impedance measurements. Thus, this method could only be applied by entities that can afford such equipment.

As a consequence, one goal of the joint research project *MoMeCha* is to identify the material parameters of piezoceramics using only a single specimen and only using the electrical impedance which can be measured comparatively cheaply. This is achieved by attaching a triple-ring electrode setup onto the disk-shaped piezoelectric ceramic as shown in section 3.2. Note that for the following it is only assumed that the ceramic is disk-shaped. However, we do not prescribe specific ratios to the geometry of the ceramic itself (e.g. ratio of height to width etc.). This is in contrast to e.g. [43] where the piezoceramics are exactly multiple specifically prescribed sizes and shapes. Furthermore, note that the assumption of disk-shaped ceramics is given here only to reduce computation time for solving the partial differential equations. Hence, in future it is planned to generalize the triple-ring electrode setup to a more general electrode geometry. In Fig. 1.4 a possible parametrization for the triple-ring setup is displayed.
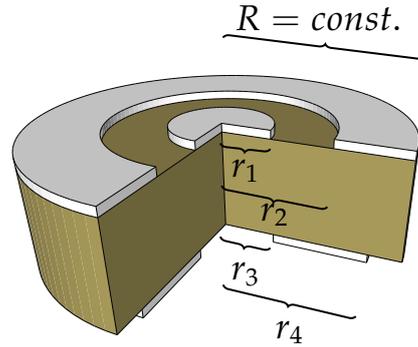
Figure 1.4: Parametrization of triple-ring electrodes.

1.3 Contributions of this Thesis

In this thesis the possibility of increased sensitivity in parameter identification problems for piezoelectrics, especially for triple-ring electrode piezoceramics, using only a single piezoceramic specimen and only impedance measurements is investigated. Some of the novel contributions of this thesis are:

- Revised, enhanced and novel theorems and proofs for existence, uniqueness and regularity of partial differential equations governing piezoceramics

- Computation of (mixed higher order) derivatives within the simulation tool CFS++ using the Algorithmic Differentiation tool ADOL-C

- Modeling and parametrization of triple-ring electrode piezoceramics

- Definition, computation, comparison and analysis of accuracy of sensitivity of impedance with respect to material parameters for triple-ring electrode piezoceramics

- Thorough optimization of sensitivity of impedance

- Proof of concept results for inverse problems in piezoelectrics using enhanced sensitivity

- A novel extension of the current triple-ring case which guarantees any prescribed sensitivity

Parts of this thesis have been previously published by the author of this thesis as the main or one of the main authors in the following [17, 29, 30, 37]. As part of this
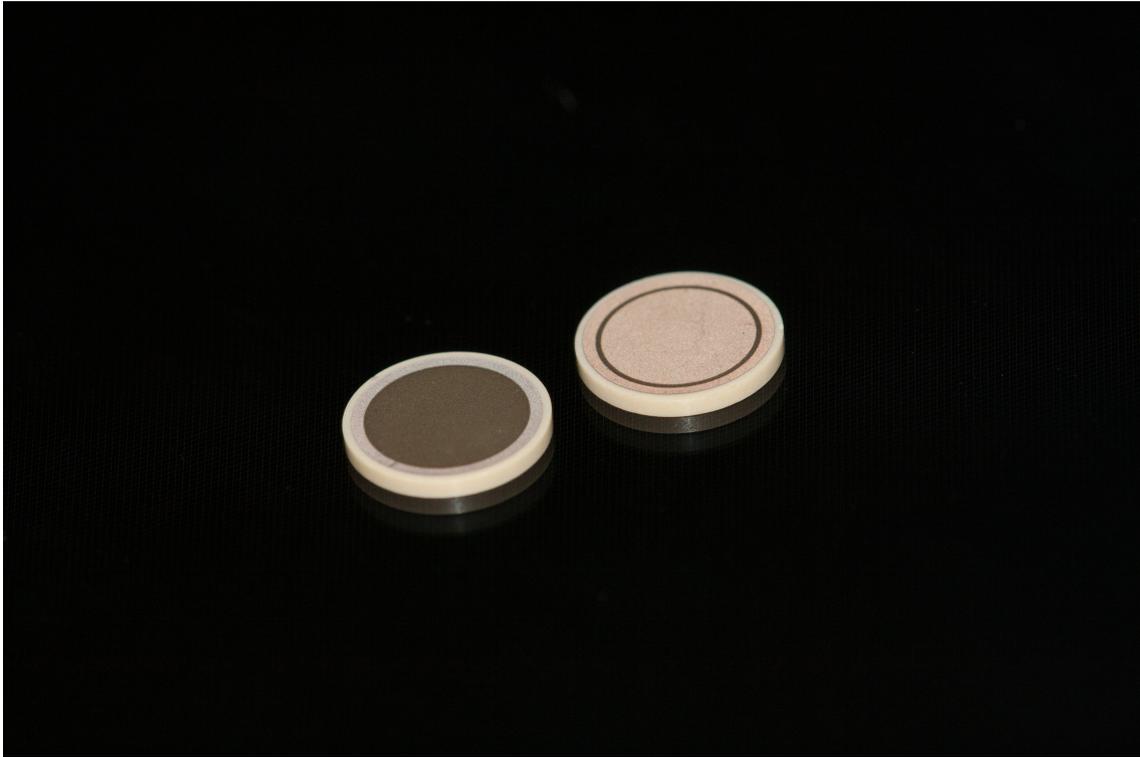
Figure 1.5: Triple-ring electrode geometry of piezoceramic developed in this thesis. Photograph by S. Olfert.

thesis a real triple-ring electrode piezoelectric ceramic with optimized sensitivity was designed which is now used for real measurements (see Fig. 1.5).

1.4 Structure of this Thesis

The thesis is structured as follows. In chapter 2 we will investigate the governing partial differential equations for piezoceramics. Therein, theoretical results concerning existence, uniqueness and regularity of solutions of the governing partial differential equations are presented. Furthermore, some aspects of finite element methods (FEM) for solving the partial differential equations (PDE) are given.

In chapter 3 the use of triple-ring electrode piezoceramics are motivated and discussed. A method to compute the overall impedance and the derivatives thereof is given. Moreover, a novel approach that can guarantee arbitrary sensitivity of

impedance for triple-ring electrode piezoceramics is stated.

Chapter 4 is focused on the sensitivity of impedance with respect to the material parameters. Here, the definition and computation of sensitivity are given. Additionally, interesting and unusual issues and solutions of computing the sensitivities are discussed.

Chapter 5 is dedicated to the optimization of sensitivity. Here solution methods and optimization results are given.

In chapter 6 solution methods for inverse problems in piezoelectrics are focused on. A proof of concept solution method for material parameter characterization of triple-ring electrode piezoelectric ceramics using the optimized sensitivity is discussed.

In the last chapter a conclusion and future research directions are stated.

1.5 Piezoelectrics From a Mathematical Point of View

This thesis was developed within the joint research project *MoMeCha*, funded by the German Research Foundation (DFG), between the research group *Mathematics and its Applications* of Prof. Dr. A. Walther and the *Measurement Engineering Group* of Prof. Dr.-Ing. B. Henning both at Paderborn University. Hence, this thesis is based upon prior work especially by C. Unverzagt (see e.g. [75]). Though this work has some similarities to work by Unverzagt, most notably the use and development of a triple-ring electrode geometry on a piezoceramic disk as well as the electrical network required to operate the piezoceramic, this work was developed and written from a mathematical point of view instead of an engineering viewpoint and is independent of work of Unverzagt.

Moreover, this work is independent with respect to most concepts, even as basic as the definition of sensitivity itself, and results gained as well as all code developed and used in this thesis unless stated otherwise.

2

Partial Differential Equations of Piezoelectricity

2.1 Physical Background, Notation, and Derivation

In this section the fundamental piezoelectric equations are derived from fundamental physical laws and the notation used is explained. These equations have been derived by numerous authors in numerous publications and are by no means novel. However, an introduction to the topic and an explanation of the notation are essential for a better understanding of later chapters.

As mentioned in the introduction, the piezoelectrical effect relates mechanical energy with electrical energy. However, it is also true that temperature plays an important role. For example, the pyroelectrical effect converts thermal energy into electrical energy. For the applications considered here there is little to no change in temperature. Hence, the pyroelectrical effect can be neglected for this thesis. Similarly, other temperature related effects such as the piezocaloric effect, the thermoelastic effect, thermal expansion, thermoelectricity, and the electrocaloric effect are neglected.

This is done so to keep the equations as simple as possible without too much loss of generality, and later on to keep the number of variables in the inverse problem as low as possible. However, the approach of this thesis could easily be extended to also incorporate temperature dependent material parameters. An overview over the related effects is given in the so-called Heckmann's diagram, see Fig. 2.1.

In this thesis we will not go into the history of piezoelectricity. However, a compelling article about the history of piezoelectricity can be found in [47].

The following is based on [34, 35, 38, 43, 69] and also recently [74].

Following [34], the mechanical and electrical quantities are related to each other by thermodynamic equilibrium. According to [60, p. 207], the first law of thermodynamics states 'that the time rate of change of the total energy is equal to the sum of the rate of work done by the external forces and the change of heat content per unit time'. Hence, with heat change $\mathrm{d}q$, change of total energy $\mathrm{d}U$ and work $\mathrm{d}w$ one gets

$$\mathrm{d}q = \mathrm{d}U - \mathrm{d}w \quad \Leftrightarrow \quad \mathrm{d}U = \mathrm{d}q + \mathrm{d}w.$$

For this thesis the temperature is assumed to be constant. Thus, $\mathrm{d}q = 0$ and thereby $\mathrm{d}U = \mathrm{d}w$ holds. As the temperature is constant, the work in piezoelectric solids is given by the mechanical work $\mathrm{d}w_{mech}$ and the electric work w_{elec}, hence

$$\mathrm{d}w = \mathrm{d}w_{mech} + \mathrm{d}w_{elec}.$$

The work is given [46, 34] by the sum of work per unit volume done by the stress σ_{ij} and the electric work caused by an electric field E_n

$$\mathrm{d}U = \mathrm{d}w_{mech} + \mathrm{d}w_{elec} = \sigma_{ij}\,\mathrm{d}S_{ij} + E_n\,\mathrm{d}D_n$$

where S is the mechanical strain and D the electric induction.

According to [34], in practice it is easier to work with σ and E than with S and D. Therefore, a Legendre transformation is performed to obtain the Gibbs free energy density G

$$G = U - \sigma_{ij}S_{ij} - E_nD_n$$

with the total differential [25, p. 11]

$$\mathrm{d}G = -S_{ij}\,\mathrm{d}\sigma_{ij} - D_n\,\mathrm{d}E_n.$$

Considering the electrical field E_n to be constant (and thus $\mathrm{d}E_n = 0$) gives the relationship

$$S_{ij} = -\left.\frac{\partial G}{\partial \sigma_{ij}}\right|_{E=c} \tag{2.1}$$
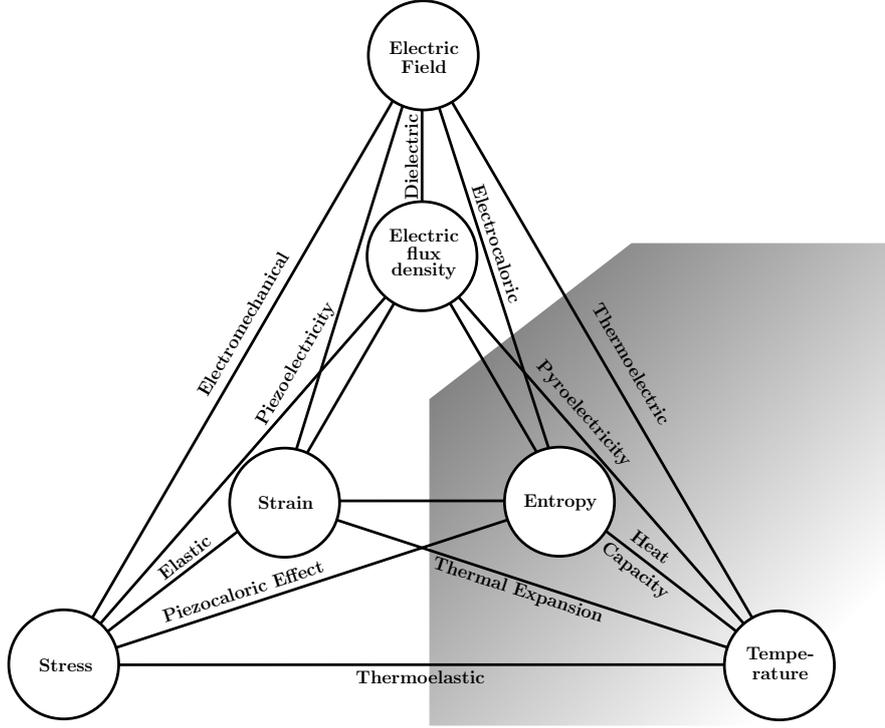
Figure 2.1: Heckmann's diagram showing the relations of physical quantities (cf. [72]). Adaptation based on [68]. As displayed in gray, due to low excitation loads we can neglect thermal effects in this work.

and conversely for constant σ_{ij} one obtains

$$D_n = - \left. \frac{\partial G}{\partial E_n} \right|_{\sigma=c} \tag{2.2}$$

where the notation $X|_{y=c}$ refers to the value of X while keeping the variable y constant. Following the argument in [38, p. 10, p. 108], a truncated Taylor series expansion of the functions $S_{ij}(\sigma_{kl}, E_k)$ and $D_i(\sigma_{kl}, E_k)$ gives

$$S_{ij} = S_{ij}(\sigma_{kl}^0, E_k^0) + \left. \frac{\partial S_{ij}}{\partial \sigma_{kl}} \right|_{E=c} (\sigma_{kl} - \sigma_{kl}^0) + \left. \frac{\partial S_{ij}}{\partial E_k} \right|_{\sigma=c} (E_k - E_k^0) \tag{2.3}$$

and analogously

$$D_n = D_n(\sigma_{kl}^0, E_k^0) + \left. \frac{\partial D_n}{\partial \sigma_{kl}} \right|_{E=c} (\sigma_{kl} - \sigma_{kl}^0) + \left. \frac{\partial D_n}{\partial E_k} \right|_{\sigma=c} (E_k - E_k^0). \tag{2.4}$$

Without loss of generality let $\sigma_{kl}^0 = 0$, $E_k^0 = 0$ and $S(\sigma_{kl}^0, E_k^0) = 0$, $D_n(\sigma_{kl}^0, E_k^0) = 0$. Now, Eq. (2.1) and Eq. (2.2) are substituted in the last equations in order to obtain

$$S_{ij} = \underbrace{\left.\frac{\partial^2 G}{\partial \sigma_{ij} \partial \sigma_{kl}}\right|_{E=c} \sigma_{kl}}_{\text{Hook's law}} + \underbrace{\left.\frac{\partial^2 G}{\partial E_k \partial \sigma_{ij}}\right|_{\sigma=c} E_k}_{\text{Inverse piezoelectric effect}},$$

$$D_n = -\underbrace{\left.\frac{\partial^2 G}{\partial E_n \partial \sigma_{kl}}\right|_{E=c} \sigma_{kl}}_{\text{Piezoelectric effect}} - \underbrace{\left.\frac{\partial^2 G}{\partial E_n \partial E_k}\right|_{\sigma=c} E_k}_{\text{Dielectric behavior}}.$$

These second order derivatives define the material tensors [34]: s_{ijkl}^E the mechanical compliance tensor given by Hook's law [50, p. 130] at a constant electric field intensity E

$$s_{ijkl}^E := \left.\frac{\partial^2 G}{\partial \sigma_{ij} \partial \sigma_{kl}}\right|_{E=c} = \left.\frac{\partial S_{ij}}{\partial \sigma_{kl}}\right|_{E=c},$$

ϵ_{nk}^σ the constitutive relation for dielectric materials at constant mechanical stress σ

$$\epsilon_{nk}^\sigma := -\left.\frac{\partial^2 G}{\partial E_n \partial E_k}\right|_{\sigma=c} = -\left.\frac{\partial D_n}{\partial E_k}\right|_{\sigma=c},$$

and the piezoelectric coupling between mechanical and electrical quantities

$$d_{nij} := -\left.\frac{\partial S_{ij}}{\partial E_n}\right|_{\sigma=c} = -\left.\frac{\partial^2 G}{\partial E_n \partial \sigma_{ij}}\right|_{\sigma=c} = -\left.\frac{\partial^2 G}{\partial E_n \partial \sigma_{ij}}\right|_{E=c} = -\left.\frac{\partial D_n}{\partial \sigma_{ij}}\right|_{E=c}$$

where subscripts indicate which entity is held constant. This gives the so-called d-form of the linear piezoelectric relations:

$$\begin{aligned} S_{ij} &= s_{ijkl}^E \sigma_{kl} + d_{kij} E_k \\ D_i &= d_{ikl} \sigma_{kl} + \epsilon_{ik}^\sigma E_k \end{aligned} \tag{2.5}$$

The occurring tensors can make the equations seem harder to handle than they actually are. In fact, the occurring tensors are symmetrical.

Thus this structure can be exploited by using Voigt notation [77] to reorganise the equations in matrix-vector form. In Voigt notation the indices of the tensors are reordered and combined using occurring symmetry, e.g. $\sigma_{ij} = \sigma_{ji}$, $i \in \{1, 2, 3\}$. In Table 2.1 the required substitutions are displayed. Hence, e.g. c_{1123}^E can be rewritten

Tensor indices ij	Replacement p
11	1
22	2
33	3
23 or 32	4
31 or 13	5
12 or 21	6

Table 2.1: Voigt Notation

as c_{14}^E.

The d-form can thus be reformulated using Voigt notation

$$S_p = s_{pq}^E \sigma_q + d_{kp} E_n$$
$$D_n = d_{nq} \sigma_q + \epsilon_{nk}^\sigma E_k$$

or more compact in matrix/vector notation where $\cdot^T$ denotes transposition

$$S = s^E \sigma + d^T E$$
$$D = d\sigma + \epsilon^\sigma E. \tag{2.6}$$

By rearranging terms in Eq. (2.6) this d-form can be transformed into a variety of different forms [25, p. 16], e.g. the h-form or the g-form, but most notably the e-form

$$\sigma = c^E S - e^T E \tag{2.7}$$
$$D = eS + \epsilon^S E \tag{2.8}$$

where the following relations [34, p. 378] were used for the transformation

$$s^E = \left(c^E\right)^{-1}, \quad d = e\left(c^E\right)^{-1}, \quad \epsilon^\sigma = \epsilon^S + e\left(c^E\right)^{-1} e^T.$$

The e-form description and its parameters c^E, e and ϵ^S are used for the remainder of this work.

Definition 2.1. *The real-valued material parameters c^E, e, ϵ^S are said to fullfill symmetry and non-negativity conditions, respectively, if*

$$c^E_{ijkl} = c^E_{jikl} = c^E_{klij}, \quad \epsilon^S_{ij} = \epsilon^S_{ji}, \qquad e_{ijk} = e_{jik}$$

and

$$\exists c_0, \epsilon_0 > 0 \quad \forall 1 \le i, j, k, l \le 3: \quad c^E_{ijkl} X_{ij} X_{kl} \ge c_0 X^2_{ij}, \quad \epsilon^S_{ij} y_i y_j \ge \epsilon_0 y^2_i$$

for tensorial quantities X and y where Einstein summation convention is used.

We will assume the material parameters are given in the following matrix form and with the following sparsity pattern

$$c^E := \begin{pmatrix} c_{11} & c_{12} & c_{13} & 0 & 0 & 0 \\ c_{12} & c_{11} & c_{13} & 0 & 0 & 0 \\ c_{13} & c_{13} & c_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & c_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2}(c_{11} - c_{12}) \end{pmatrix} \in \mathbb{R}^{6 \times 6}$$

$$\epsilon^S := \begin{pmatrix} \epsilon_{11} & 0 & 0 \\ 0 & \epsilon_{11} & 0 \\ 0 & 0 & \epsilon_{33} \end{pmatrix} \in \mathbb{R}^{3 \times 3}$$

$$e := \begin{pmatrix} 0 & 0 & 0 & 0 & e_{15} & 0 \\ 0 & 0 & 0 & e_{15} & 0 & 0 \\ e_{13} & e_{13} & e_{33} & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{6 \times 3}.$$

This assumption is said to be correct for many piezoelectric ceramics [34] and corresponds to the assumption of a transversely isotropic solid [11].

Remark 2.2. By the non-negativity conditions 2.1 it also follows that c^E and ϵ^S have (strictly) positive eigenvalues.

Following e.g. [43], using Newton's second law, Gauss' law and Faraday's law the e-form can be transformed into a partial differential equation: Let

$$\mathcal{B} := \begin{pmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \end{pmatrix}$$

with x, y, z referring to Cartesian coordinates. By linearized elasticity, the mechanical strain S can be computed as the symmetric gradient of the mechanical displacement $u \in \mathbb{R}^3$:

$$S = \begin{pmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \end{pmatrix} = \begin{pmatrix} S_{xx} \\ S_{yy} \\ S_{zz} \\ 2S_{yz} \\ 2S_{xz} \\ 2S_{xy} \end{pmatrix} = \mathcal{B}u$$

Newton's second law of motion for mechanical behavior [60] is given by

$$\mathcal{B}^T \sigma = \rho \frac{\partial u}{\partial t^2}. \tag{2.9}$$

Piezoelectric ceramics are insulating. Gauss' law [50] gives

$$\nabla \cdot D = 0. \tag{2.10}$$

By Faraday's law, the electrical field E is given by the negative gradient of the electrical potential ϕ, hence

$$E = -\nabla \phi. \tag{2.11}$$

These physical laws can be now processed in Eq. (2.7)

$$\sigma = c^E S - e^T E$$

$$\Rightarrow \mathcal{B}^T \sigma = \mathcal{B}^T \left(c^E S - e^T E \right) = \rho \frac{\partial u}{\partial t^2} = \rho \ddot{u}$$

$$\Rightarrow \mathcal{B}^T \left(c^E \mathcal{B} u + e^T \nabla \phi \right) = \rho \ddot{u}$$

and Eq. (2.8)

$$D = eS + \epsilon^S E$$

$$\Rightarrow 0 = \nabla \cdot D = \nabla \cdot \left(eS + \epsilon^S E \right)$$

$$\Leftrightarrow 0 = \nabla \cdot \left(e\mathcal{B} u - \epsilon^S \nabla \phi \right).$$

Consequently:

$$\rho \ddot{u} - \mathcal{B}^T \left(c^E \mathcal{B} u + e^T \nabla \phi \right) = 0$$
$$\nabla \cdot \left(e\mathcal{B} u - \epsilon^S \nabla \phi \right) = 0 \tag{2.12}$$

However, these equations are not yet suitable to describe real physical behavior. In real physical applications friction is expected. Hence, the amplitude of ϕ and u should decline in time. This is achieved by integrating a Rayleigh damping model [23],[64] to the first equation $\rho \ddot{u} - \mathcal{B}^T \left(c^E \mathcal{B} u + e^T \nabla \phi \right) = 0$ by adding two parameters α and β and one further term for each parameter to the equation. Thus, the equation is finally modified to

$$\rho \ddot{u} + \alpha \rho \dot{u} - \mathcal{B}^T \left(c^E \mathcal{B} u + \beta c^E \mathcal{B} \dot{u} + e^T \nabla \phi \right) = 0$$
$$\nabla \cdot \left(e\mathcal{B} u - \epsilon^S \nabla \phi \right) = 0. \tag{2.13}$$

Remark 2.3. Adding second, third or even higher order terms to the truncated Taylor series Eq. (2.3) and Eq. (2.4) gives a nonlinear piezoelectric model, see e.g. [38, p. 108]. In this case, instead of only being constants one can interpret the material parameters to be functions depending e.g. on strain S and the electric field E, see e.g. [67, 24]. The possibility that the material parameters could be functions instead of constants is of special significance for the instrumentation of Algorithmic

Differentiation into the simulation code CFS++ as will be pointed out in chapter 4. The extension of the work of this thesis to a generalized model is of interest for future undertakings.

2.2 Weak Formulation, Existence, Uniqueness, and Regularity of Solutions

The following considerations were developed in cooperation with V. Schulze at the research group 'Mathematics and its Applications'.

Before we can begin to solve any partial differential equation we must first establish an exact setup - the geometry Ω, the boundary $\partial\Omega$, the boundary conditions, and initial values of the partial differential equations in question. Furthermore, the notation of the remainder of this chapter is introduced.

Setup, Notation and Weak Formulation

We consider the case of a mechanically unclamped piezoceramic which is excited by prescribing a voltage on part of the boundary. Let $\Omega \subseteq \mathbb{R}^3$ be an open domain describing the piezoelectric ceramic and let $\partial\Omega =: \Gamma$ be the nonempty boundary of Ω. The boundary is divided into nonempty, disjunct, covering subsets of Γ (see also Fig. 2.2) which are assumed to have a positive 2D measure. Let Γ_e be the section of the boundary which is electrically excited, Γ_g the section of the boundary which is grounded, and $\Gamma_r = \Gamma \setminus (\Gamma_e \cup \Gamma_g)$ the remaining boundary section. For the reader's convenience the usual definitions of common function spaces which will be required later on are stated here. Let $d, k \in \mathbb{N}$ be integers and let α be a multiindex. Then
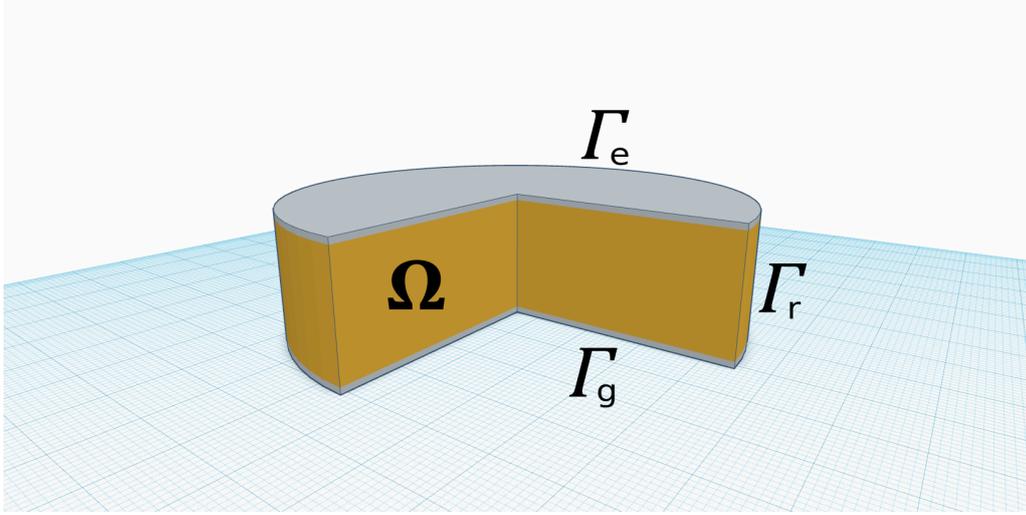
Figure 2.2: Domain and boundaries of a piezoceramic.

we define

$$C^k(\Omega) := \left\{ \sigma : \Omega \to \mathbb{R}^d : \sigma \text{ is } k\text{-times continuously differentiable} \right\}$$

$$L^2(\Omega) := \left\{ \sigma : \Omega \to \mathbb{R}^d : \quad \|\sigma\|_{L^2(\Omega)}^2 := \int_\Omega \sigma^T \sigma \, d\Omega < \infty \right\}$$

$$H^1(\Omega) := \left\{ \sigma : \Omega \to \mathbb{R} : \quad \|\sigma\|_{H^1(\Omega)}^2 := \|\sigma\|_{L^2(\Omega)}^2 + \|\nabla\sigma\|_{L^2(\Omega)}^2 < \infty \right\}$$

$$H_0^1(\Omega) := \left\{ \sigma \in H^1(\Omega) : \sigma|_\Gamma = 0 \text{ with } \|\sigma\|_{H_0^1(\Omega)} := \|\sigma\|_{H^1(\Omega)} \right\}$$

$$H_{0,\Gamma}^1(\Omega) := \left\{ \sigma_1 + \sigma_2 : \sigma_1 \in H_0^1(\Omega) \text{ and } \sigma_2 \in H^1(\Omega) \right\}$$

$$H_{\mathcal{B}}^1(\Omega) := \left\{ \sigma : \Omega \to \mathbb{R}^3 : \quad \|\sigma\|_{H_{\mathcal{B}}^1(\Omega)}^2 := \|\sigma\|_{L^2(\Omega)}^2 + \|\mathcal{B}\sigma\|_{L^2(\Omega)}^2 < \infty \right\}$$

$$H^{-1}(\Omega) := \Big\{ f \text{ continuous linear functional on } H_0^1(\Omega) :$$
$$\sup_{\|\sigma\|_{H_0^1(\Omega)} \leq 1} |\langle f, \sigma \rangle| < \infty \Big\}.$$

Let $\sigma : [0,T] \to X$ be Bochner-measurable. Then

$$L^2(0,T;X) := \left\{ \sigma : [0,T] \to X : \quad \int_{[0,T]} \|\sigma(t)\|_X^2 \, dt < \infty \right\}$$

$$L^\infty(0,T;X) := \left\{ \sigma : [0,T] \to X : \quad \underset{0 \le t \le T}{\operatorname{ess\,sup}} \|\sigma(t)\|_X < \infty \right\}$$

$$H^1(0,T;X) := \left\{ \sigma : [0,T] \to X : \quad \int_{[0,T]} \|\sigma(t)\|_X^2 + \|\dot\sigma(t)\|_X^2 \, dt < \infty \right\}$$

$$H^2(\Omega) := \left\{ \sigma : \Omega \to \mathbb{R}^3 : \right.$$

$$\left. \|\sigma\|_{H^2(\Omega)} := \left(\sum_{|\alpha| \le 2} \|D^{(\alpha)}\sigma\|_{L^2(\Omega)} \right)^{1/2} < \infty \right\}.$$

We denote derivatives with respect to time by the dot symbol e.g. $\dot\sigma$ and derivatives with respect to space by the nabla or $\mathcal{B}$ symbol e.g. $\nabla\sigma$ or $\mathcal{B}\sigma$. All derivatives in the above are understood in the distributional sense. In addition, the dual space of a Hilbert space X is denoted by X'. In particular, $H^{-1}(\Omega)$ denotes the dual space of $H_0^1(\Omega)$. Note that in order to simplify the notation superscripts indicating the dimension of $\mathbf{u}$ or $\mathcal{B}\mathbf{u}$, which are 3 and 6 respectively, are omitted. This is reasonable as the vectorial scalar product inside $\int_\Omega \sigma^T \sigma \, d\Omega$ always returns a scalar no matter what dimensions σ has.

Let $\vec{n} := (n_x, n_y, n_z)$ be the normal vector and

$$\mathcal{N} := \begin{pmatrix} n_x & 0 & 0 \\ 0 & n_y & 0 \\ 0 & 0 & n_z \\ 0 & n_z & n_y \\ n_z & 0 & n_x \\ n_y & n_x & 0 \end{pmatrix}.$$

The three dimensional transient linear piezoelectric equations with Rayleigh damping parameters $\alpha, \beta > 0$ chosen sufficiently large (so that the PDE is parabolic) and density $\rho > 0$ describing the mechanical displacement $\mathbf{u} \in \mathbb{R}^3$ and the electrical

potential $\phi \in \mathbb{R}$ for given boundary conditions are stated as:

$$
\begin{aligned}
\rho\ddot{\mathbf{u}}(t) + \alpha\rho\dot{\mathbf{u}}(t) - \mathcal{B}^T\left(c^E\mathcal{B}\mathbf{u}(t) + \beta c^E\mathcal{B}\dot{\mathbf{u}}(t) + e^T\nabla\phi(t)\right) &= 0 \text{ in } \Omega \times [0,T] \\
-\nabla \cdot \left(e\mathcal{B}\mathbf{u}(t) - \epsilon^S\nabla\phi(t)\right) &= 0 \text{ in } \Omega \times [0,T] \\
\phi(t) &= 0 \text{ on } \Gamma_g \times [0,T] \\
\phi(t) &= \phi^e(t) \text{ on } \Gamma_e \times [0,T] \\
\vec{n} \cdot \left(e\mathcal{B}\mathbf{u}(t) - \epsilon^S\nabla\phi(t)\right) &= 0 \text{ on } \Gamma_r \times [0,T] \\
\mathcal{N}^T\left(c^E\mathcal{B}\mathbf{u}(t) + \beta c^E\mathcal{B}\dot{\mathbf{u}}(t) + e^T\nabla\phi(t)\right) &= 0 \text{ on } \partial\Omega \times [0,T] \\
\mathbf{u}(0) &= \mathbf{u}_0 \\
\dot{\mathbf{u}}(0) &= \mathbf{u}_1
\end{aligned}
$$

The weak form of the equations above can be obtained [38] by testing with appropriate functions $v \in \mathbb{R}^3$ (for the first line) and $w \in \mathbb{R}$ (for the second line), integration by parts and using boundary conditions:

$$
\int_\Omega \left(\mathcal{B}^T\sigma\right)^T v \, d\Omega = -\int_\Omega \sigma^T\mathcal{B}v \, d\Omega + \int_{\partial\Omega}\left(\mathcal{N}^T\sigma\right)^T v \, d\Omega
$$

First, we use a *Dirichlet lift ansatz* to homogenize the Dirichlet boundary condition for $\phi(t)$: Let $t \in [0,T]$ and let $\chi \in H^1(\Omega)$ where $\chi|_{\Gamma_g} = 0$ and $\chi|_{\Gamma_e} = 1$. Such a χ exists if we assume that Ω is at least a Lipschitz domain. Let $\phi(t)$ consist of two parts $\phi(t) = \phi_0(t) + \phi_{\phi^e}(t)$ where $\phi_0(t) \in H_0^1(\Omega)$ and $\phi_{\phi^e}(t) \in H^1(\Omega)$. We then rewrite $\phi_{\phi^e}(t) = \phi^e(t)\chi$. We now set $\phi_0(t) := \phi(t) - \phi^e(t)\chi$. As $\phi^e(t)$ is a given value, $\phi^e(t)\chi$ can be taken out of the left hand side of the weak form and added to the right hand side. The weak form of the piezoelectric system is given by

$$
\begin{aligned}
&\forall t \in [0,T] \text{ a.e.}, \quad \forall(v,w) \in H_{\mathcal{B}}^1(\Omega) \times H_0^1(\Omega): \\
&\int_\Omega \rho\ddot{\mathbf{u}}^T v \, d\Omega + \alpha\int_\Omega \rho\dot{\mathbf{u}}^T v \, d\Omega + \int_\Omega \left(c^E\mathcal{B}\mathbf{u}\right)^T\mathcal{B}v \, d\Omega + \beta\int_\Omega\left(c^E\mathcal{B}\dot{\mathbf{u}}\right)^T\mathcal{B}v \, d\Omega \\
&+ \int_\Omega\left(e^T\nabla\phi_0\right)^T\mathcal{B}v \, d\Omega + \int_\Omega\left(e\mathcal{B}\mathbf{u}\right)^T\nabla w \, d\Omega - \int_\Omega\left(\epsilon^S\nabla\phi_0\right)^T\nabla w \, d\Omega \\
&= \phi^e\int_\Omega -(e^T\nabla\chi)^T\mathcal{B}v + (\epsilon^S\nabla\chi)^T\nabla w \, d\Omega.
\end{aligned} \tag{2.14}
$$

Note that in light of [15, Thm. 2 in section 5.9.2] it makes sense to demand $\mathbf{u}(0) = \mathbf{u}_0$ and $\dot{\mathbf{u}}(0) = \mathbf{u}_1$. See also the only remark in [15, section 7.2.1].

Existence, Uniqueness and Regularity of Solutions

Before we attempt to show existence, uniqueness and regularity of solutions some additional tools are required:

Lemma 2.4. *(Young inequality)*
Let $1 < p, q < \infty, \frac{1}{p} + \frac{1}{q} = 1$. Then for $a, b > 0$ the following inequality holds:

$$ab \leq \frac{a^p}{p} + \frac{b^q}{q}$$

Proof. See [15, Appendix B.2]. $\qquad\square$

Lemma 2.5. *(Hölder inequality)*
Let $1 \leq p, q \leq \infty, \frac{1}{p} + \frac{1}{q} = 1$. Then for $u \in L^p(\Omega), v \in L^q(\Omega)$ the following inequality holds:

$$\int_\Omega |uv| \, dx \leq \|u\|_{L^p(\Omega)} \|v\|_{L^q(\Omega)}$$

Proof. See [15, Appendix B.2]. $\qquad\square$

Remark 2.6. The latter two inequalities are especially true for $p = q = 2$. In that case the last inequality is also known as Cauchy–Schwarz (CS) inequality.

Lemma 2.7. *(Gronwall inequality, integral form)*

a) *Let $\eta : [0, T] \to \mathbb{R}^{\geq 0}$ be a non-negative, summable function on $[0, T]$, which satisfies for almost every t the integral inequality*

$$\eta(t) \leq C_1 \int_0^t \eta(s) \, ds + C_2$$

for constants $C_1, C_2 \geq 0$. Then

$$\eta(t) \leq C_2 \left(1 + C_1 t e^{C_1 t}\right)$$

for a.e. $0 \leq t \leq T$.

b) *In particular, if*

$$\eta(t) \leq C_1 \int_0^t \eta(s)ds$$

for a.e. $t \in [0,T]$, *then*

$$\eta(t) = 0 \ a.e.$$

Proof. See [15, Appendix B.2]. □

Lemma 2.8.
Let $u \in C^1([0,T])$ *and* $t \in [0,T]$. *Then the following holds:*

$$\|u(t)\|_{L^2(\Omega)}^2 \leq 2\|u(0)\|_{L^2(\Omega)}^2 + 2T \int_0^t \|\dot{u}(s)\|_{L^2(\Omega)}^2 \, ds.$$

Proof. This inequality can be seen via Young and Cauchy–Schwarz inequalities. □

Remark 2.9. (Sufficiently smooth boundary)
We say the boundary $\partial\Omega$ is sufficiently smooth if it permits application of the trace theorem (cf. [15]). Thus, a C^1-boundary is sufficient.

However, it is possible to utilize a variation of the trace theorem under less strict requirements (cf. [12]). We note that the boundary for our specific application (see Fig. 2.2) satisfies the special Lipschitz condition stated in Definition 5 of [12] and thus it appears that it can also be considered sufficiently smooth.

A proof for the following theorem was first given in [38]. The proof given there is also heavily oriented on work of [2] which itself is based on [49]. Here we present a proof with similar essential steps as in the other given proofs, but with more technical details and necessary exact descriptions. In order to overcome these issues a refined version is presented. The proof follows the usual guideline as seen for many partial differential equations (e.g. [15, p. 353]): We get existence and uniqueness of a weak solution by the usual procedure:

1. Discretization via Galerkin approximation of infinite dimensional function spaces

2. Energy estimates via Gronwall inequality in discretized space which provide finiteness of the discretized solution

3. Weak limit of discretized solution provides existence of a weak solution in infinite dimensional function space

4. Uniqueness of the solution is shown by applying estimates to the difference $w := w_1 - w_2$ of two solutions w_1 and w_2

Theorem 2.10.
Let $\Omega \subseteq \mathbb{R}^3$ be a bounded domain with sufficiently smooth boundary as specified in Remark 2.9 and let the real valued material parameters c^E, e and ϵ^S satisfy the non-negativity conditions specified in Def. 2.1. The Rayleigh coefficients α and β are assumed to be non-negative. Let $T > 0$ and $\rho > 0$.
Then there exists a $C > 0$ such that for any $\mathbf{u}_0 \in H^1_{\mathcal{B}}(\Omega), \mathbf{u}_1 \in L^2(\Omega)$ and $\phi^e \in H^1(0, T; H^{1/2}(\Gamma_e))$ there exists a unique solution

$$(\mathbf{u}, \phi) \in L^\infty(0, T; H^1_{\mathcal{B}}(\Omega)) \times L^\infty(0, T; H^1_{0,\Gamma}(\Omega)) \tag{2.15}$$

with

$$\dot{\mathbf{u}} \in L^\infty(0, T; L^2(\Omega)) \ \text{and} \ \ddot{\mathbf{u}} \in L^2(0, T; (H^1_{\mathcal{B}}(\Omega))') \tag{2.16}$$

to Eq. (2.14) satisfying the initial conditions

$$\mathbf{u}(0) = \mathbf{u}_0, \quad \dot{\mathbf{u}}(0) = \mathbf{u}_1 \ \text{on} \ \Omega$$

and the following estimate holds:

$$\begin{aligned}
&\|\mathbf{u}\|_{L^\infty(0,T;H^1_{\mathcal{B}}(\Omega))} + \|\dot{\mathbf{u}}\|_{L^\infty(0,T;L^2(\Omega))} + \|\ddot{\mathbf{u}}\|_{L^2(0,T;(H^1_{\mathcal{B}}(\Omega))')} + \|\phi\|_{L^\infty(0,T;H^1_{0,\Gamma}(\Omega))} \\
&\leq C \left(\|\mathbf{u}_0\|_{H^1_{\mathcal{B}}(\Omega)} + \|\mathbf{u}_1\|_{L^2(\Omega)} + \|\phi^e\|_{H^1(0,T;H^{1/2}(\Gamma_e))} \right).
\end{aligned} \tag{2.17}$$

Proof. Note that many concepts of this proof are taken from [15, chapter 7] and information regarding involved spaces can be found in [1].

In the following constants denoted by the letters C or $\tilde{C}$ are used. Unless explicitly specified otherwise we note that all these constants are positive $C_i > 0\,, i \geq 1$.

Weak solutions are functions $\mathbf{u}, \dot{\mathbf{u}}, \ddot{\mathbf{u}}$ and ϕ_0 as in (2.15) and (2.16) where $\phi = \phi_0 + \phi_{\phi_e}$ such that for almost all $t \in [0, T]$ for all $(v, w) \in H^1_{\mathcal{B}}(\Omega) \times H^1_0(\Omega)$ the following equation holds:

$$
\begin{aligned}
&\langle \rho \ddot{\mathbf{u}}(t), v \rangle + \alpha \left\langle \rho \dot{\mathbf{u}}(t), v \right\rangle + \left\langle c^E \mathcal{B} \mathbf{u}(t), \mathcal{B}v \right\rangle + \beta \left\langle c^E \mathcal{B} \dot{\mathbf{u}}(t), \mathcal{B}v \right\rangle \\
&+ \left\langle e^T \nabla \phi_0(t), \mathcal{B}v \right\rangle + \left\langle e\mathcal{B}\mathbf{u}(t), \nabla w \right\rangle - \left\langle \epsilon^S \nabla \phi_0(t), \nabla w \right\rangle \\
&= \langle f(t), v \rangle + \langle g(t), w \rangle
\end{aligned}
\tag{2.18}
$$

with

$$
\langle f(t), v \rangle := -\phi^e(t) \int_\Omega (e^T \nabla \chi)^T \mathcal{B}v \, d\Omega
$$

and

$$
\langle g(t), w \rangle := \phi^e(t) \int_\Omega (\epsilon^S \nabla \chi)^T \nabla w \, d\Omega.
$$

Note that by Riesz representation theorem there exists a unique representation for the latter functionals as an inner product, i.e. $\langle f, \cdot \rangle$ and $\langle g, \cdot \rangle$. As is common in the field of partial differential equation for convenience we will also use the same symbols f and g to refer to the Riesz-representative as well as the functionals $\langle f, \cdot \rangle$ and $\langle g, \cdot \rangle$. Furthermore, we remember that $\chi \in H^1(\Omega)$ and that ϵ^S, e are constant. The integrals $\int_\Omega (e^T \nabla \chi)^T \mathcal{B}v \, d\Omega$ and $\int_\Omega (\epsilon^S \nabla \chi)^T \nabla w \, d\Omega$ are finite, their values $c_1(\Omega), c_2(\Omega) < \infty$ depend e.g. on Ω but do not depend on t. Thus, by integrating this constant value over time we can estimate the Bochner-space norm of f by

$$
\|f\|_{H^1(0,T;(H^1_{\mathcal{B}}(\Omega))')} \leq c_1(\Omega) \|\phi^e\|_{H^1(0,T)}
$$

and analogously we get

$$
\|g\|_{H^1(0,T;H^{-1}(\Omega))} \leq c_2(\Omega) \|\phi^e\|_{H^1(0,T)}.
$$

Phase 1: Galerkin approximation

The weak form is tested with test functions $v_j \in H^1_{\mathcal{B}}(\Omega)$ and $w_j \in H^1_0(\Omega), \quad j \in \mathbb{N}$,

with

$$\mathbf{u}(t) \approx \mathbf{u}_m(t) = \sum_{j=1}^m u_m^j(t) v_j$$

and

$$\phi_0(t) \approx \phi_m(t) = \sum_{j=1}^m \phi_m^j(t) w_j,$$

where '$\approx$' is to be understood in the sense of an orthogonal projection in the appropriate spaces. The finite dimensional spaces spanned by the test functions are defined as

$$V_m := \mathrm{span}\{v_1, \ldots, v_m\} \quad \text{and} \quad W_m := \mathrm{span}\{w_1, \ldots, w_m\}.$$

Note that we can assume the dimensions $\dim(V_m) = \dim(W_m) = m$, for V_m in each vectorial component, as the test functions can be selected to be linearly independent. Furthermore, the functions can be chosen such that $\overline{\bigcup_{m=1}^\infty V_m} = H_{\mathcal{B}}^1(\Omega)$ and $\overline{\bigcup_{m=1}^\infty W_m} = H_0^1(\Omega)$.

Then via standard theory for ordinary differential equations (see e.g. [15] or [44]) for all $m \in \mathbb{N}$ and for all $(v_m, w_m) \in V_m \times W_m$ there exists a unique solution

$$(\mathbf{u}_m, \phi_m) \in C^2([0,T]; V_m) \times C([0,T]; W_m)$$

to the discretized version of Eq. (2.18) that fulfills the initial conditions $\mathbf{u}_m(0) = (\mathbf{u}_0)_m, \dot{\mathbf{u}}_m(0) = (\mathbf{u}_1)_m$. For more information on Sobolev spaces involving time see also [15, section 5.9.2].

Phase 2: Energy estimates
The aim of this phase is to use Gronwall inequality to show an energy estimate from which the finiteness of the finite dimensional solutions

$$(\mathbf{u}_m(t), \phi_m(t)) \in L^\infty(0,T; H_{\mathcal{B}}^1(\Omega)) \times L^\infty(0,T; H_0^1(\Omega))$$
$$\dot{\mathbf{u}}_m \in L^\infty(0,T; L^2(\Omega)), \quad \ddot{\mathbf{u}}_m \in L^2(0,T; (H_{\mathcal{B}}^1(\Omega))')$$

can be deduced: Let

$$\eta(t) := \left(\|\dot{\mathbf{u}}_m(t)\|_{L^2(\Omega)}^2 + \|\mathbf{u}_m(t)\|_{H_{\mathcal{B}}^1(\Omega)}^2 + \|\phi_m(t)\|_{H_0^1(\Omega)}^2 \right).$$

In order to use Gronwall inequality we must show that there are constants $p, q \geq 0$ such that $\eta(t) \leq p \int_0^t \eta(s)\, ds + q$ holds. If this condition is true, then it can be shown that

$$\|\dot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(t)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\phi_m(t)\|^2_{H^1_0(\Omega)}$$

$$\leq \left(1 + pte^{pt}\right) \left(\|\dot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(0)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\phi_m(0)\|^2_{H^1_0(\Omega)} \right. \tag{2.19}$$

$$\left. + \|f\|^2_{L^2(0,T;(H^1_{\mathcal{B}}(\Omega))')} + \|g\|^2_{H^1(0,T;H^{-1}(\Omega))} \right)$$

holds almost everywhere in $[0, T]$. Thus, this must also be true for the essential supremum over $0 \leq t \leq T$ and we will get finiteness in the $L^\infty(0, T; X)$ norm for the appropriate sub-spaces X. In order to show the requirement we consider the following:

First, the discretized version of the weak form Eq. (2.18) is supposed to hold for all test functions (v_m, w_m). Thus, it should also hold for $(\dot{\mathbf{u}}_m(t), 0)$:

$$\langle \rho\ddot{\mathbf{u}}_m(t), \dot{\mathbf{u}}_m(t) \rangle + \alpha \langle \rho\dot{\mathbf{u}}_m(t), \dot{\mathbf{u}}_m(t) \rangle + \langle c^E \mathcal{B}\mathbf{u}_m(t), \mathcal{B}\dot{\mathbf{u}}_m(t) \rangle$$

$$+ \beta \langle c^E \mathcal{B}\dot{\mathbf{u}}_m(t), \mathcal{B}\dot{\mathbf{u}}_m(t) \rangle + \langle e^T \nabla\phi_m(t), \mathcal{B}\dot{\mathbf{u}}_m(t) \rangle = \langle f(t), \dot{\mathbf{u}}_m(t) \rangle$$

By transposing the inner product and direct computation it is easy to see that one can swap the placement of constant scalars or matrices such as ρ, ϵ^S, c^E etc. (which are symmetric) in this bilinear form, e.g. the following holds:

$$\langle c^E \mathcal{B}\mathbf{u}_m(t), \mathcal{B}\dot{\mathbf{u}}_m(t) \rangle = \langle (c^E)^T \mathcal{B}\dot{\mathbf{u}}_m(t), \mathcal{B}\mathbf{u}_m(t) \rangle = \langle c^E \mathcal{B}\dot{\mathbf{u}}_m(t), \mathcal{B}\mathbf{u}_m(t) \rangle$$

Thus, by bilinearity of the inner product

$$2 \langle c^E \mathcal{B}\dot{\mathbf{u}}_m(t), \mathcal{B}\mathbf{u}_m(t) \rangle = \frac{d}{dt} \langle c^E \mathcal{B}\mathbf{u}_m(t), \mathcal{B}\mathbf{u}_m(t) \rangle .$$

Hence, the above equation simplifies to

$$\frac{1}{2}\frac{d}{dt} \left(\langle \rho\dot{\mathbf{u}}_m(t), \dot{\mathbf{u}}_m(t) \rangle + \langle c^E \mathcal{B}\mathbf{u}_m(t), \mathcal{B}\mathbf{u}_m(t) \rangle \right) + \alpha \langle \rho\dot{\mathbf{u}}_m(t), \dot{\mathbf{u}}_m(t) \rangle$$

$$+ \beta \langle c^E \mathcal{B}\dot{\mathbf{u}}_m(t), \mathcal{B}\dot{\mathbf{u}}_m(t) \rangle + \langle e^T \nabla\phi_m(t), \mathcal{B}\dot{\mathbf{u}}_m(t) \rangle = \langle f(t), \dot{\mathbf{u}}_m(t) \rangle . \tag{2.20}$$

Now, taking into account that the test functions v_m, w_m do not depend on time t

the weak form Eq. (2.18) is first differentiated with respect to t and it is taken into account that $\dot{v}_m, \dot{w}_m \equiv 0$.

Finally, the resulting equation is tested with $(0, \phi_m(t))$:

$$\langle e\mathcal{B}\dot{\mathbf{u}}_m(t), \nabla\phi_m(t)\rangle - \frac{1}{2}\frac{d}{dt}\left\langle \epsilon^S\nabla\phi_m(t), \nabla\phi_m(t)\right\rangle = \langle \dot{g}(t), \phi_m(t)\rangle \qquad (2.21)$$

A subtraction of Eq. (2.20) and Eq. (2.21) gives

$$\frac{1}{2}\frac{d}{dt}\left(\langle\rho\dot{\mathbf{u}}_m(t), \dot{\mathbf{u}}_m(t)\rangle + \left\langle c^E\mathcal{B}\mathbf{u}_m(t), \mathcal{B}\mathbf{u}_m(t)\right\rangle + \left\langle \epsilon^S\nabla\phi_m(t), \nabla\phi_m(t)\right\rangle\right)$$
$$+\alpha\langle\rho\dot{\mathbf{u}}_m(t), \dot{\mathbf{u}}_m(t)\rangle + \beta\left\langle c^E\mathcal{B}\dot{\mathbf{u}}_m(t), \mathcal{B}\dot{\mathbf{u}}_m(t)\right\rangle \qquad (2.22)$$
$$= \langle f(t), \dot{\mathbf{u}}_m(t)\rangle - \langle \dot{g}(t), \phi_m(t)\rangle$$

The last equation Eq. (2.22) is integrated with respect to t.

$$\mathcal{F}_l(t) := \left(\langle\rho\dot{\mathbf{u}}_m(t), \dot{\mathbf{u}}_m(t)\rangle + \left\langle c^E\mathcal{B}\mathbf{u}_m(t), \mathcal{B}\mathbf{u}_m(t)\right\rangle + \left\langle \epsilon^S\nabla\phi_m(t), \nabla\phi_m(t)\right\rangle\right)$$
$$+2\alpha\int_0^t \langle\rho\dot{\mathbf{u}}_m(s), \dot{\mathbf{u}}_m(s)\rangle\,ds + 2\beta\int_0^t \left\langle c^E\mathcal{B}\dot{\mathbf{u}}_m(s), \mathcal{B}\dot{\mathbf{u}}_m(s)\right\rangle\,ds$$
$$= \left(\langle\rho\dot{\mathbf{u}}_m(0), \dot{\mathbf{u}}_m(0)\rangle + \left\langle c^E\mathcal{B}\mathbf{u}_m(0), \mathcal{B}\mathbf{u}_m(0)\right\rangle + \left\langle \epsilon^S\nabla\phi_m(0), \nabla\phi_m(0)\right\rangle\right) \qquad (2.23)$$
$$+2\int_0^t \langle f(t), \dot{\mathbf{u}}_m(s)\rangle\,ds - 2\int_0^t \langle \dot{g}(t), \phi_m(s)\rangle\,ds =: \mathcal{F}_r(t)$$

For convenience the left hand side of Eq. (2.23) is referred to as $\mathcal{F}_l(t)$ and the right hand side is referred to as $\mathcal{F}_r(t)$. Hence, in short one can write

$$\mathcal{F}_l(t) = \mathcal{F}_r(t).$$

We now aim to use this equation to show that the requirements for Gronwall inequality are met. We start by showing that the left hand side $\mathcal{F}_l(t)$ has a lower bound. With $\lambda_{1,mech}$ the smallest eigenvalue of c^E (which is strictly positive) one

estimates

$$\int_\Omega (\mathcal{B}\mathbf{u}_m(t))^T c^E \mathcal{B}\mathbf{u}_m(t)\, d\Omega \geq \lambda_{1,mech} \int_\Omega (\mathcal{B}\mathbf{u}_m(t))^T \mathcal{B}\mathbf{u}_m(t)\, d\Omega$$
$$= \lambda_{1,mech} \|\mathcal{B}\mathbf{u}_m(t)\|^2_{L^2(\Omega)} \qquad (2.24)$$
$$= \lambda_{1,mech} \left(\|\mathbf{u}_m(t)\|^2_{H^1_{\mathcal{B}}(\Omega)} - \|\mathbf{u}_m(t)\|^2_{L^2(\Omega)} \right).$$

With $\lambda_{1,elec}$ the smallest eigenvalue of ϵ^S (which is strictly positive) one estimates

$$\int_\Omega (\nabla\phi_m(t))^T \epsilon^S \nabla\phi_m(t)\, d\Omega \geq \lambda_{1,elec} \int_\Omega (\nabla\phi_m(t))^T \nabla\phi_m(t)\, d\Omega$$
$$= \lambda_{1,elec} \|\nabla\phi_m(t)\|^2_{L^2(\Omega)}$$

From Poincaré inequality (see e.g. [66]) we obtain $c_1, c_2 \in \mathbb{R}$ such that

$$\int_\Omega (\nabla\phi_m(t))^T \epsilon^S \nabla\phi_m(t)\, d\Omega \geq \lambda_{1,elec} \|\nabla\phi_m(t)\|^2_{L^2(\Omega)}$$
$$= (1+c_2)c_1 \|\nabla\phi_m(t)\|^2_{L^2(\Omega)} = c_1 \left(\underbrace{c_2 \|\nabla\phi_m(t)\|^2_{L^2(\Omega)}}_{\geq \|\phi_m(t)\|^2_{L^2(\Omega)}} + \|\nabla\phi_m(t)\|^2_{L^2(\Omega)} \right) \qquad (2.25)$$
$$\geq C_{elec} \|\phi_m(t)\|^2_{H^1_0(\Omega)}.$$

By non-negativity of ρ, α, β and the two inequalities Eq. (2.24) and Eq. (2.25) one can now estimate

$$C_1 \left(\|\dot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(t)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\phi_m(t)\|^2_{H^1_0(\Omega)} - c_{mech} \|\mathbf{u}_m(t)\|^2_{L^2(\Omega)} \right) \leq \mathcal{F}_l(t) \quad (2.26)$$

with a positive constant $C_1 > 0$. Furthermore, by the inequalities Eq. (2.24) and Eq. (2.25) and Cauchy–Schwarz and Young inequalities the right hand side $\mathcal{F}_r(t)$

can be estimated from above with $c, \tilde{c} > 0$:

$$\mathcal{F}_r(t) = \underbrace{\langle \rho \dot{\mathbf{u}}_m(0), \dot{\mathbf{u}}_m(0) \rangle}_{=\rho \|\dot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)}} + \underbrace{\langle c^E \mathcal{B}\mathbf{u}_m(0), \mathcal{B}\mathbf{u}_m(0) \rangle}_{\leq c \|\mathbf{u}_m(0)\|^2_{H^1_\mathcal{B}(\Omega)}} + \underbrace{\langle \epsilon^S \nabla \phi_m(0), \nabla \phi_m(0) \rangle}_{\leq \tilde{c} \|\phi_m(0)\|^2_{H^1_0(\Omega)}}$$

$$+ 2 \int_0^t \langle f(s), \dot{\mathbf{u}}_m(s) \rangle \, ds - 2 \int_0^t \langle \dot{g}(s), \phi_m(s) \rangle \, ds$$

$$\leq \widehat{C_2} \left(\| \dot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(0)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(0)\|^2_{H^1_0(\Omega)} \right)$$

$$+ 2 \int_0^t |\langle f(s), \dot{\mathbf{u}}_m(s) \rangle| \, ds + 2 \int_0^t |\langle \dot{g}(s), \phi_m(s) \rangle| \, ds$$

$$\leq \widehat{C_2} \left(\| \dot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(0)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(0)\|^2_{H^1_0(\Omega)} \right)$$

$$+ \int_0^t \underbrace{\|\mathbf{u}_m(s)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\dot{\mathbf{u}}_m(s)\|^2_{L^2(\Omega)} + \|\phi_m(s)\|^2_{H^1_0(\Omega)}}_{\geq 0} \, ds$$

$$+ 2\|f\|^2_{L^2(0,T;(H^1_\mathcal{B}(\Omega))')} + 2\|g\|^2_{H^1(0,T;H^{-1}(\Omega))}$$

Hence, we get

$$\mathcal{F}_r(t) \leq C_2 \left(\|\dot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(0)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(0)\|^2_{H^1_0(\Omega)} \right)$$

$$+ \int_0^t \left(\|\dot{\mathbf{u}}_m(s)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(s)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(s)\|^2_{H^1_0(\Omega)} \right) ds \qquad (2.27)$$

$$+ \|f\|^2_{L^2(0,T;(H^1_\mathcal{B}(\Omega))')} + \|g\|^2_{H^1(0,T;H^{-1}(\Omega))}$$

with a positive constant $C_2 > 0$. As $\mathcal{F}_l(t) = \mathcal{F}_r(t)$ it is now clear that

$$C_1 \left(\|\dot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(t)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(t)\|^2_{H^1_0(\Omega)} - c_{mech} \|\mathbf{u}_m(t)\|^2_{L^2(\Omega)} \right)$$

$$\leq C_2 \left(\|\dot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(0)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(0)\|^2_{H^1_0(\Omega)} \right)$$

$$+ \int_0^t \left(\|\dot{\mathbf{u}}_m(s)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(s)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(s)\|^2_{H^1_0(\Omega)} \right) ds \qquad (2.28)$$

$$+ \|f\|^2_{L^2(0,T;(H^1_\mathcal{B}(\Omega))')} + \|g\|^2_{H^1(0,T;H^{-1}(\Omega))}.$$

Utilizing Lemma 2.8 we can remove $c_{mech} \|\mathbf{u}_m(t)\|^2_{L^2(\Omega)}$ from the left hand side of the

inequality to obtain:

$$
\begin{aligned}
& C_1 \left(\|\dot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(t)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(t)\|^2_{H^1_0(\Omega)} \right) \\
\leq\, & C_3 \left(\|\dot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(0)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(0)\|^2_{H^1_0(\Omega)} \right) \\
& + C_4 \int_0^t \left(\|\dot{\mathbf{u}}_m(s)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(s)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(s)\|^2_{H^1_0(\Omega)} \right) ds \\
& + \|f\|^2_{L^2(0,T;(H^1_\mathcal{B}(\Omega))')} + \|g\|^2_{H^1(0,T;H^{-1}(\Omega))}
\end{aligned}
\tag{2.29}
$$

where $C_4 > 0$ now also depends on the fixed value T.

Let

$$
\eta(t) := \|\dot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(t)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(t)\|^2_{H^1_0(\Omega)}
$$

and let

$$
\tilde{C}_2 := \frac{1}{C_1} \left(C_3 \eta(0) + \|f\|^2_{L^2(0,T;(H^1_\mathcal{B}(\Omega))')} + \|g\|^2_{H^1(0,T;H^{-1}(\Omega))} \right) \geq 0.
$$

Then the above inequality simplifies to

$$
\eta(t) \leq \frac{C_4}{C_1} \int_0^t \eta(s)\, ds + \tilde{C}_2.
$$

Hence, all requirements for Gronwall inequality have been shown to hold and it can now be safely applied and the result simplified to:

$$
\begin{aligned}
& \|\dot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(t)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(t)\|^2_{H^1_0(\Omega)} \\
\leq\, & \left(\frac{\tilde{C}_3}{C_1} + \frac{C_4 \tilde{C}_3}{C_1^2} t e^{\frac{C_4}{C_1} t} \right) \left(\|\dot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)} + \|\mathbf{u}_m(0)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\phi_m(0)\|^2_{H^1_0(\Omega)} \right. \\
& \left. + \|f\|^2_{L^2(0,T;(H^1_\mathcal{B}(\Omega))')} + \|g\|^2_{H^1(0,T;H^{-1}(\Omega))} \right)
\end{aligned}
\tag{2.30}
$$

holds almost everywhere in $[0, T]$.

We will return to this inequality shortly after considering the bilinear form

$$
A : H^1_0(\Omega) \times H^1_0(\Omega) \to \mathbb{R}, \quad A(\phi_m(t), w) := \left\langle \epsilon^S \nabla \phi_m(t), \nabla w \right\rangle
\tag{2.31}
$$

and the continuous linear functional on $H_0^1(\Omega)$ for a fixed $\mathbf{u}_m(t)$

$$b(w) := \langle e\mathcal{B}\mathbf{u}_m(t), \nabla w \rangle - \langle g(t), w \rangle$$

which together form the weak form Eq. (2.18) tested by $(0, w)$. This bilinear form A is coercive (inequality Eq. (2.25)) and continuous:

$$\left| \langle \epsilon^S \nabla \phi_m(t), \nabla w \rangle \right| \leq \lambda_{max} \left| \langle \nabla \phi_m(t), \nabla w \rangle \right| \leq \lambda_{max} \|\nabla \phi_m(t)\|_{L^2(\Omega)} \cdot \|\nabla w\|_{L^2(\Omega)}$$
$$\leq \lambda_{max} \|\phi_m(t)\|_{H_0^1(\Omega)} \cdot \|w\|_{H_0^1(\Omega)}$$

Hence, using Lax–Milgram lemma we get the estimate for $A(\phi_m(t), w) = b(w)$ $\forall w \in H_0^1(\Omega)$:

$$\begin{aligned}
\|\phi_m(t)\|_{H_0^1(\Omega)}^2 &\leq \tilde{M} \|b\|_{H^{-1}(\Omega)}^2 = \tilde{M} \sup_{\|w\|_{H_0^1(\Omega)} \leq 1} \|b(w)\|_{H_0^1(\Omega)}^2 \\
&= \tilde{M} \sup_{\|w\|_{H_0^1(\Omega)} \leq 1} \left| \langle e\mathcal{B}\mathbf{u}_m(t), \nabla w \rangle - \langle g(t), w \rangle \right|^2 \\
&\leq \tilde{M} \sup_{\|w\|_{H_0^1(\Omega)} \leq 1} \left(\left| \langle e\mathcal{B}\mathbf{u}_m(t), \nabla w \rangle \right| + \left| \langle g(t), w \rangle \right| \right)^2 \\
&\overset{Young}{\leq} \tilde{M} \sup_{\|w\|_{H_0^1(\Omega)} \leq 1} \left(2 \underbrace{\left| \langle e\mathcal{B}\mathbf{u}_m(t), \nabla w \rangle \right|^2}_{\overset{C.S.}{\leq} \|e\mathcal{B}\mathbf{u}_m(t)\|_{L^2(\Omega)}^2 \cdot \|w\|_{H_0^1(\Omega)}^2} + 2 \left| \langle g(t), w \rangle \right|^2 \right) \\
&\leq 2M(\|e\mathcal{B}\mathbf{u}_m(t)\|_{L^2(\Omega)}^2 + \|g(t)\|_{H^{-1}(\Omega)}^2)
\end{aligned}$$

(2.32)

Furthermore, in the case $t = 0$ we get

$$\|\phi_m(0)\|_{H_0^1(\Omega)}^2 \leq 2M \left(\|e\mathcal{B}\mathbf{u}_m(0)\|_{L^2(\Omega)}^2 + \|g(0)\|_{H^{-1}(\Omega)}^2 \right). \tag{2.33}$$

Hence, we obtain

$$\|\phi_m(0)\|_{H_0^1(\Omega)}^2 \leq C_5 \left(\|(\mathbf{u}_0)_m\|_{H_{\mathcal{B}}^1(\Omega)}^2 + \|\phi^e(0)\|_{H^{1/2}(\Gamma_e)}^2 \right).$$

Finally, from Gronwall inequality we can thus deduce

$$
\begin{aligned}
&\|\dot{\mathbf{u}}_m\|^2_{L^\infty(0,T;L^2(\Omega))} + \|\mathbf{u}_m\|^2_{L^\infty(0,T;H^1_{\mathcal{B}}(\Omega))} + \|\phi_m\|^2_{L^\infty(0,T;H^1_0(\Omega))} \\
&\leq C_6 \left(\|(\mathbf{u}_1)_m\|^2_{L^2(\Omega)} + \|(\mathbf{u}_0)_m\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\phi^e\|^2_{L^\infty(0,T;H^{1/2}(\Gamma_e))} \right).
\end{aligned}
\tag{2.34}
$$

Furthermore, now knowing that all these values are finite we can deduce from Eq. (2.23) that also

$$
\beta \|\mathcal{B}\dot{\mathbf{u}}_m\|_{L^2(0,T;L^2(\Omega))} < \infty.
\tag{2.35}
$$

It now remains to show that $\|\ddot{\mathbf{u}}_m\|_{L^2(0,T;(H^1_{\mathcal{B}}(\Omega))')}$ is finite. We follow the general guideline given in e.g. [15, p. 355]. Fix any $\tilde{v} \in H^1_{\mathcal{B}}(\Omega)$ with $\|\tilde{v}\|_{H^1_{\mathcal{B}}(\Omega)} \leq 1$ and $\tilde{v} := \tilde{v}^1 + \tilde{v}^2$ with $\tilde{v}^1 \in \mathrm{span}\{v_i\}_{i=1}^m$ and $\langle \tilde{v}^2, v_i \rangle = 0$ for all $1 \leq i \leq m$. Since $\{v_i\}_{i=0}^m$ can be assumed orthogonal in $H^1_{\mathcal{B}}(\Omega)$, $\|\tilde{v}^1\|_{H^1_{\mathcal{B}}(\Omega)} \leq \|\tilde{v}\|_{H^1_{\mathcal{B}}(\Omega)} \leq 1$. Now with $\mathbf{u}_m = \sum_{i=0}^m u_m^i(t)v_i$ the following holds almost everywhere in $[0,T]$:

$$
\begin{aligned}
\langle \ddot{\mathbf{u}}_m(t), \tilde{v} \rangle_{H^1_{\mathcal{B}}(\Omega)} &= \langle \ddot{\mathbf{u}}_m(t), \tilde{v} \rangle = \langle \ddot{\mathbf{u}}_m(t), \tilde{v}^1 \rangle \\
&= \langle f(t), \tilde{v}^1 \rangle - \langle c^E \mathcal{B}\mathbf{u}_m(t), \mathcal{B}\tilde{v}^1 \rangle - \langle e^T \nabla \phi_m(t), \mathcal{B}\tilde{v}^1 \rangle \\
&\quad - \alpha \langle \rho \dot{\mathbf{u}}_m(t), \tilde{v}^1 \rangle - \beta \langle c^E \mathcal{B}\dot{\mathbf{u}}_m(t), \mathcal{B}\tilde{v}^1 \rangle
\end{aligned}
$$

where the subscript $H^1_{\mathcal{B}}(\Omega)$ denotes the duality pairing between $(H^1_{\mathcal{B}}(\Omega))'$ and $H^1_{\mathcal{B}}(\Omega)$. Using Cauchy–Schwarz inequality we can deduce

$$
\begin{aligned}
&|\langle \ddot{\mathbf{u}}_m(t), \tilde{v} \rangle|_{H^1_{\mathcal{B}}(\Omega)} \\
&\leq C_7 \left(\|f(t)\|_{(H^1_{\mathcal{B}}(\Omega))'} + \|\mathbf{u}_m(t)\|_{H^1_{\mathcal{B}}(\Omega)} + \|\dot{\mathbf{u}}_m(t)\|_{L^2(\Omega)} + \|\mathcal{B}\dot{\mathbf{u}}_m(t)\|_{L^2(\Omega)} \right) \\
&\quad + |\langle e^T \nabla \phi_m(t), \mathcal{B}\tilde{v}^1 \rangle|.
\end{aligned}
\tag{2.36}
$$

Using Lax–Milgram lemma again on the form Eq. (2.31) for a general $t \in [0,T]$ we can further deduce with analogous arguments as in Eq. (2.32) that

$$
\left| \langle e^T \nabla \phi_m(t), \mathcal{B}\tilde{v}^1 \rangle \right|_{H^1_{\mathcal{B}}(\Omega)} \leq M \left(\|\mathcal{B}\mathbf{u}_m(t)\|^2_{L^2(\Omega)} + \|\phi^e(t)\|^2_{H^{1/2}(\Gamma_e)} \right).
$$

Thus by repetitive application of Young inequality we get for the norm

$$
\|\ddot{\mathbf{u}}_m(t)\|^2_{(H^1_{\mathcal{B}}(\Omega))'} = \sup_{\|v\|_{H^1_{\mathcal{B}}(\Omega)} \leq 1} |\langle \ddot{\mathbf{u}}_m(t), v\rangle|^2_{H^1_{\mathcal{B}}(\Omega)}
$$

$$
\leq C_8 \Big(\|f(t)\|^2_{(H^1_{\mathcal{B}}(\Omega))'} + \|\mathbf{u}_m(t)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\dot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)}
$$
$$
+ \|\mathcal{B}\dot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} + \|\mathcal{B}\mathbf{u}_m(t)\|^2_{L^2(\Omega)} + \|\phi^e(t)\|^2_{H^{1/2}(\Gamma_e)} \Big).
$$

Now we have finiteness for all components, hence we can finally integrate inequality Eq. (2.36) over $[0, T]$. We rearrange the terms and apply the estimates Eq. (2.34) and Eq. (2.35).

$$
\int_0^T \|\ddot{\mathbf{u}}_m(s)\|^2_{(H^1_{\mathcal{B}}(\Omega))'} ds
$$
$$
\leq C_9 \int_0^T \|f(s)\|^2_{(H^1_{\mathcal{B}}(\Omega))'} + \|\mathbf{u}_m(s)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\dot{\mathbf{u}}_m(s)\|^2_{L^2(\Omega)} + \|\mathcal{B}\dot{\mathbf{u}}_m(s)\|^2_{L^2(\Omega)} \tag{2.37}
$$
$$
+ \|\phi^e(s)\|^2_{H^{1/2}(\Gamma_e)} \, ds
$$
$$
\leq C_{10} \Big(\|(\mathbf{u}_0)_m\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|(\mathbf{u}_1)_m\|^2_{L^2(\Omega)} + \|\phi^e\|^2_{H^1(0,T;H^{1/2}(\Gamma_e))} \Big)
$$

Thus, it is now clear that

$$
(\mathbf{u}_m, \phi_m) \in L^\infty(0, T; H^1_{\mathcal{B}}(\Omega)) \times L^\infty(0, T; H^1_0(\Omega))
$$
$$
\dot{\mathbf{u}}_m \in L^\infty(0, T; L^2(\Omega)) \tag{2.38}
$$
$$
\ddot{\mathbf{u}}_m \in L^2(0, T; (H^1_{\mathcal{B}}(\Omega))').
$$

Phase 3: Weak limit

Following e.g. [15, p. 384], [66, p. 239] from the energy estimates Eq. (2.34) and Eq. (2.37) we get the boundedness of the sequence $(\mathbf{u}_m, \phi_m)_{m=1}^\infty$ in $L^\infty(0, T; H^1_{\mathcal{B}}(\Omega)) \times L^\infty(0, T; H^1_0(\Omega))$, the boundedness of $(\dot{\mathbf{u}}_m)_{m=1}^\infty$ in $L^\infty(0, T; L^2(\Omega))$ and the boundedness of $(\ddot{\mathbf{u}}_m)_{m=1}^\infty$ in $L^2(0, T; (H^1_{\mathcal{B}}(\Omega))')$. Thus there exist subsequences

$$
(\mathbf{u}_{m_l}, \phi_{m_l})_{l=1}^\infty \subseteq (\mathbf{u}_m, \phi_m)_{m=1}^\infty, \quad (\dot{\mathbf{u}}_{m_l})_{l=1}^\infty \subseteq (\dot{\mathbf{u}}_m)_{m=1}^\infty, \quad (\ddot{\mathbf{u}}_{m_l})_{l=1}^\infty \subseteq (\ddot{\mathbf{u}}_m)_{m=1}^\infty
$$

with

$$(\mathbf{u}, \phi) \in L^\infty(0, T; H^1_{\mathcal{B}}(\Omega)) \times L^\infty(0, T; H^1_{0,\Gamma}(\Omega)),$$

$$\dot{\mathbf{u}} \in L^\infty(0, T; L^2(\Omega))$$

$$\ddot{\mathbf{u}} \in L^2(0, T; (H^1_{\mathcal{B}}(\Omega))')$$

such that

$$\begin{aligned}
\mathbf{u}_{m_l} &\rightharpoonup \mathbf{u} \text{ weakly-* in } L^\infty(0, T; H^1_{\mathcal{B}}(\Omega)) \\
\phi_{m_l} &\rightharpoonup \phi_0 \text{ weakly-* in } L^\infty(0, T; H^1_0(\Omega)) \\
\dot{\mathbf{u}}_{m_l} &\rightharpoonup \dot{\mathbf{u}} \text{ weakly-* in } L^\infty(0, T; L^2(\Omega)) \\
\ddot{\mathbf{u}}_{m_l} &\rightharpoonup \ddot{\mathbf{u}} \text{ weakly in } L^2(0, T; (H^1_{\mathcal{B}}(\Omega))').
\end{aligned} \tag{2.39}$$

We now proceed to show that the weak limit is a solution of the weak form. Following [15, p. 384] we fix a $N \in \mathbb{N}$ and choose functions $v \in C^1(0, T; H^1_{\mathcal{B}}(\Omega))$ and $w \in C^1(0, T; H^1_0(\Omega))$ having the form

$$v(t) := \sum_{k=1}^N u^k_m(t) v_k, \quad w(t) := \sum_{k=1}^N \phi^k_m(t) w_k. \tag{2.40}$$

We choose $m \geq N$, multiply the discretized versions for each pair (v_k, w_k) of the weak form Eq. (2.18) with $(u^k_m(t), \phi^k_m(t))$, sum over $k = 1, \ldots, N$, integrate with respect to t and obtain

$$\begin{aligned}
\int_0^T \Big(&\int_\Omega \rho \ddot{\mathbf{u}}_m(s)^T v \, d\Omega + \alpha \int_\Omega \rho \dot{\mathbf{u}}_m(s)^T v \, d\Omega + \int_\Omega \left(c^E \mathcal{B} \mathbf{u}_m(s) \right)^T \mathcal{B} v \, d\Omega \\
&+ \beta \int_\Omega \left(c^E \mathcal{B} \dot{\mathbf{u}}_m(s) \right)^T \mathcal{B} v \, d\Omega + \int_\Omega \left(e^T \nabla \phi_m(s) \right)^T \mathcal{B} v \, d\Omega \\
&+ \int_\Omega (e \mathcal{B} \mathbf{u}_m(s))^T \nabla w \, d\Omega - \int_\Omega \left(\epsilon^S \nabla \phi_m(s) \right)^T \nabla w \, d\Omega \Big) \, ds \\
&= \int_0^T \langle f(s), v \rangle \, ds + \int_0^T \langle g(s), w \rangle \, ds.
\end{aligned} \tag{2.41}$$

Fixing $m = m_l$ and using Eq. (2.39) we obtain in the limit $m \to \infty$ along the

subsequence m_l

$$
\begin{aligned}
\int_0^T \Big(&\int_\Omega \rho \ddot{\mathbf{u}}(s)^T v \, d\Omega + \alpha \int_\Omega \rho \dot{\mathbf{u}}(s)^T v \, d\Omega + \int_\Omega \left(c^E \mathcal{B} \mathbf{u}(s) \right)^T \mathcal{B} v \, d\Omega \\
&+ \beta \int_\Omega \left(c^E \mathcal{B} \dot{\mathbf{u}}(s) \right)^T \mathcal{B} v \, d\Omega + \int_\Omega \left(e^T \nabla \phi_0(s) \right)^T \mathcal{B} v \, d\Omega \\
&+ \int_\Omega \left(e \mathcal{B} \mathbf{u}(s) \right)^T \nabla w \, d\Omega - \int_\Omega \left(\epsilon^S \nabla \phi_0(s) \right)^T \nabla w \, d\Omega \Big) \, ds \\
= &\int_0^T \langle f(s), v \rangle \, ds + \int_0^T \langle g(s), w \rangle \, ds.
\end{aligned}
\tag{2.42}
$$

Noting that all functions of form Eq. (2.40) are dense in the according spaces this equality holds for all functions $v \in L^2(0, T; H^1_{\mathcal{B}}(\Omega))$, $w \in L^2(0, T; H^1_0(\Omega))$. In particular it follows that also

$$
\begin{aligned}
&\int_\Omega \rho \ddot{\mathbf{u}}^T v \, d\Omega + \alpha \int_\Omega \rho \dot{\mathbf{u}}^T v \, d\Omega + \int_\Omega \left(c^E \mathcal{B} \mathbf{u} \right)^T \mathcal{B} v \, d\Omega + \beta \int_\Omega \left(c^E \mathcal{B} \dot{\mathbf{u}} \right)^T \mathcal{B} v \, d\Omega \\
&+ \int_\Omega \left(e^T \nabla \phi_0 \right)^T \mathcal{B} v \, d\Omega + \int_\Omega \left(e \mathcal{B} \mathbf{u} \right)^T \nabla w \, d\Omega - \int_\Omega \left(\epsilon^S \nabla \phi_0 \right)^T \nabla w \, d\Omega \\
&= \langle f, v \rangle + \langle g, w \rangle
\end{aligned}
\tag{2.43}
$$

almost everywhere $t \in [0, T]$ for all $v \in H^1_{\mathcal{B}}(\Omega)$ and $w \in H^1_0(\Omega)$.

Following [38] and [15, section 7.2.2 Thm. 3, p. 385] we confirm that the initial conditions are also met. Choose any function $(v, 0)$ with $v \in C^2([0, T]; H^1_{\mathcal{B}}(\Omega))$ and $v(T) = \dot{v}(T) = 0$. By integrating by parts twice with respect to t of Eq. (2.41) we get

$$
\begin{aligned}
\int_0^T \Big(&\int_\Omega \rho \mathbf{u}_m(t)^T \ddot{v} \, d\Omega - \alpha \int_\Omega \rho \mathbf{u}_m(t)^T \dot{v} \, d\Omega + \int_\Omega \left(c^E \mathcal{B} \mathbf{u}_m(t) \right)^T \mathcal{B} v \, d\Omega \\
&- \beta \int_\Omega \left(c^E \mathcal{B} \mathbf{u}_m(t) \right)^T \mathcal{B} \dot{v} \, d\Omega + \int_\Omega \left(e^T \nabla \phi_m(t) \right)^T \mathcal{B} v \, d\Omega \Big) \, dt \\
= &\int_0^T \langle f(t), v \rangle \, dt - \langle \rho \mathbf{u}_m(0), \dot{v}(0) \rangle + \langle \rho \dot{\mathbf{u}}_m(0), v(0) \rangle \\
&+ \alpha \langle \rho \mathbf{u}_m(0), v(0) \rangle + \beta \langle c^E \mathcal{B} \mathbf{u}_m(0), \mathcal{B} v(0) \rangle
\end{aligned}
\tag{2.44}
$$

and analogously using Eq. (2.42) we get

$$
\begin{aligned}
\int_0^T &\left(\int_\Omega \rho \mathbf{u}(t)^T \ddot{v} \, d\Omega - \alpha \int_\Omega \rho \mathbf{u}(t)^T \dot{v} \, d\Omega + \int_\Omega \left(c^E \mathcal{B} \mathbf{u}(t) \right)^T \mathcal{B} v \, d\Omega \right. \\
&\left. - \beta \int_\Omega \left(c^E \mathcal{B} \mathbf{u}(t) \right)^T \mathcal{B} \dot{v} \, d\Omega + \int_\Omega \left(e^T \nabla \phi_0(t) \right)^T \mathcal{B} v \, d\Omega \right) dt \\
= \int_0^T &\langle f(t), v \rangle \, dt - \langle \rho \mathbf{u}(0), \dot{v}(0) \rangle + \langle \rho \dot{\mathbf{u}}(0), v(0) \rangle \\
&+ \alpha \langle \rho \mathbf{u}(0), v(0) \rangle + \beta \langle c^E \mathcal{B} \mathbf{u}(0), \mathcal{B} v(0) \rangle .
\end{aligned}
\tag{2.45}
$$

For Eq. (2.44) we set $m = m_l$ and recall Eq. (2.39) to deduce

$$
\begin{aligned}
\int_0^T &\left(\int_\Omega \rho \mathbf{u}(t)^T \ddot{v} \, d\Omega - \alpha \int_\Omega \rho \mathbf{u}(t)^T \dot{v} \, d\Omega + \int_\Omega \left(c^E \mathcal{B} \mathbf{u}(t) \right)^T \mathcal{B} v \, d\Omega \right. \\
&\left. - \beta \int_\Omega \left(c^E \mathcal{B} \mathbf{u}(t) \right)^T \mathcal{B} \dot{v} \, d\Omega + \int_\Omega \left(e^T \nabla \phi_0(t) \right)^T \mathcal{B} v \, d\Omega \right) dt \\
= \int_0^T &\langle f(t), v \rangle \, dt - \langle \rho \mathbf{u}_0, \dot{v}(0) \rangle + \langle \rho \mathbf{u}_1, v(0) \rangle \\
&+ \alpha \langle \rho \mathbf{u}_0, v(0) \rangle + \beta \langle c^E \mathcal{B} \mathbf{u}_0, \mathcal{B} v(0) \rangle .
\end{aligned}
\tag{2.46}
$$

By equating coefficients of Eq. (2.45) and Eq. (2.46) (set either $v(0)$ or $\dot{v}(0)$ to zero) we conclude $\mathbf{u}(0) = \mathbf{u}_0$ and $\dot{\mathbf{u}}(0) = \mathbf{u}_1$.

Phase 4: Uniqueness

Following e.g. [15, p. 385] it suffices to show that the only weak solution with

$$
f \equiv 0, \ g \equiv 0, \ \phi^e \equiv 0, \ \mathbf{u}_0 = \mathbf{u}_1 \equiv 0
$$

is

$$
\mathbf{u} \equiv 0, \ \phi \equiv 0.
$$

Notice that by property Eq. (2.35) $\|\mathcal{B} \dot{\mathbf{u}}_m\|_{L^2(0,T;L^2(\Omega))}$ is finite. Hence, the remark in [15, remark below Thm 4, section 7.2.2 c), p. 385] does not apply to our case and we can continue in the fashion of [15, Thm. 4, section 7.1.2c), p. 358] instead. Passing to limits, we substitute $v = \mathbf{u}$ and $w = \phi_0$ in the original weak form. This is not prohibited as by property Eq. (2.35) all components exist also in the limit. Hence,

we can deduce that the following non-discretized inequality holds

$$
C_1 \left(\|\dot{\mathbf{u}}(t)\|^2_{L^2(\Omega)} + \|\mathbf{u}(t)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\phi_0(t)\|^2_{H^1_0(\Omega)} \right)
$$
$$
\leq C_3 \left(\|\dot{\mathbf{u}}(0)\|^2_{L^2(\Omega)} + \|\mathbf{u}(0)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\phi_0(0)\|^2_{H^1_0(\Omega)} \right)
$$
$$
+ C_4 \int_0^t \left(\|\dot{\mathbf{u}}(s)\|^2_{L^2(\Omega)} + \|\mathbf{u}(s)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\phi_0(s)\|^2_{H^1_0(\Omega)} \right) ds
$$
$$
+ \|f\|^2_{L^2(0,T;(H^1_{\mathcal{B}}(\Omega))')} + \|g\|^2_{H^1(0,T;H^{-1}(\Omega))}.
$$

In the case $t = 0$ we get from Eq. (2.33) that $\|\phi(0)\|^2_{H^1_0(\Omega)} = 0$. Hence, we now note that

$$
\tilde{C}_2 = \frac{1}{C_1} \left(C_3 \eta(0) + \|f\|^2_{L^2(0,T;(H^1_{\mathcal{B}}(\Omega))')} + \|g\|^2_{H^1(0,T;H^{-1}(\Omega))} \right) = 0.
$$

Finally, we can apply the second part of Gronwall inequality to conclude that

$$
\eta(t) = \|\dot{\mathbf{u}}(t)\|^2_{L^2(\Omega)} + \|\mathbf{u}(t)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\phi_0(t)\|^2_{H^1_0(\Omega)} = 0 \quad \text{a.e. } t \in [0,T].
$$

Thus, the only solution can be the trivial solution. $\qquad\square$

Now that we know unique solutions to the weak form exist we can investigate what conditions are necessary for higher regularity. The following theorem is inspired by Thm. 5, chapter 7.2 in [15]. The proof uses ideas from [44] adapted for additional Rayleigh damping.

Theorem 2.11. *Let all requirements of Thm. 2.10 hold. If additionally $\mathbf{u}_0 \in H^2(\Omega)$, $\mathbf{u}_1 \in H^1(\Omega)$, $\beta\mathbf{u}_1 \in H^2(\Omega)$, $\phi^e \in H^2(0,T;H^{1/2}(\Gamma_e))$, then*

$$
\mathbf{u} \in L^\infty(0,T;H^1_{\mathcal{B}}(\Omega)), \quad \dot{\mathbf{u}} \in L^\infty(0,T;H^1_{\mathcal{B}}(\Omega)), \qquad \ddot{\mathbf{u}} \in L^\infty(0,T;L^2(\Omega))
$$
$$
\phi \in L^\infty(0,T;H^1_{0,\Gamma}(\Omega)), \quad \dot{\phi} \in L^\infty(0,T;H^1_{0,\Gamma}(\Omega)). \tag{2.47}
$$

Proof. We differentiate the weak form Eq. (2.18) once with respect to time t and test the result first with $(\ddot{\mathbf{u}}_m(t), 0)$ to obtain

$$
\frac{1}{2}\frac{d}{dt} \left(\langle \rho\ddot{\mathbf{u}}_m(t), \ddot{\mathbf{u}}_m(t) \rangle + \langle c\mathcal{B}\dot{\mathbf{u}}_m(t), \mathcal{B}\dot{\mathbf{u}}_m(t) \rangle \right) + \alpha \langle \rho\ddot{\mathbf{u}}_m(t), \ddot{\mathbf{u}}_m(t) \rangle
$$
$$
+ \beta \left\langle c^E\mathcal{B}\ddot{\mathbf{u}}_m(t), \mathcal{B}\ddot{\mathbf{u}}_m(t) \right\rangle + \left\langle e\nabla\dot{\phi}_m(t), \mathcal{B}\ddot{\mathbf{u}}_m(t) \right\rangle = \left\langle \dot{f}(t), \ddot{\mathbf{u}}_m(t) \right\rangle.
$$

We differentiate the weak form Eq. (2.18) twice with respect to time t and test the result first with $(0, \dot{\phi}_m(t))$ to obtain

$$\left\langle e\mathcal{B}\ddot{\mathbf{u}}_m(t), \nabla\dot{\phi}_m(t) \right\rangle - \frac{1}{2}\frac{d}{dt}\left\langle \epsilon^S \nabla\dot{\phi}_m(t), \nabla\dot{\phi}_m(t) \right\rangle = \left\langle \ddot{g}(t), \dot{\phi}_m(t) \right\rangle.$$

Analogously to the proof of Thm. 2.10 we subtract these two results and integrate with respect to t to obtain in analogy to Eq. (2.23)

$$\begin{aligned}
&\left(\langle \rho\ddot{\mathbf{u}}_m(t), \ddot{\mathbf{u}}_m(t)\rangle + \left\langle c^E\mathcal{B}\dot{\mathbf{u}}_m(t), \mathcal{B}\dot{\mathbf{u}}_m(t) \right\rangle + \left\langle \epsilon^S\nabla\dot{\phi}_m(t), \nabla\dot{\phi}_m(t) \right\rangle \right) \\
&+2\alpha \int_0^t \langle \rho\ddot{\mathbf{u}}_m(s), \ddot{\mathbf{u}}_m(s)\rangle \, ds + 2\beta \int_0^t \left\langle c^E\mathcal{B}\ddot{\mathbf{u}}_m(s), \mathcal{B}\ddot{\mathbf{u}}_m(s) \right\rangle \, ds = \\
&\left(\langle \rho\ddot{\mathbf{u}}_m(0), \ddot{\mathbf{u}}_m(0)\rangle + \left\langle c^E\mathcal{B}\dot{\mathbf{u}}_m(0), \mathcal{B}\dot{\mathbf{u}}_m(0) \right\rangle + \left\langle \epsilon^S\nabla\dot{\phi}_m(0), \nabla\dot{\phi}_m(0) \right\rangle \right) \\
&+2\int_0^t \left\langle \dot{f}(s), \dot{\mathbf{u}}_m(s) \right\rangle \, ds - 2\int_0^t \left\langle \ddot{g}(s), \dot{\phi}_m(s) \right\rangle \, ds
\end{aligned} \tag{2.48}$$

or, again, abbreviated as $\mathcal{F}_l = \mathcal{F}_r$. Analogously to inequality Eq. (2.29) we then can obtain

$$\begin{aligned}
&C_1 \left(\|\ddot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} + \|\dot{\mathbf{u}}_m(t)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\dot{\phi}_m(t)\|^2_{H^1_0(\Omega)} \right) - c_{mech}\|\dot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} \\
&\leq C_3 \left(\|\ddot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)} + \|\dot{\mathbf{u}}_m(0)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\dot{\phi}_m(0)\|^2_{H^1_0(\Omega)} \right) \\
&+C_4 \int_0^t \left(\|\ddot{\mathbf{u}}_m(s)\|^2_{L^2(\Omega)} + \|\dot{\mathbf{u}}_m(s)\|^2_{H^1_\mathcal{B}(\Omega)} + \|\dot{\phi}_m(s)\|^2_{H^1_0(\Omega)} \right) \, ds \\
&+\|f\|^2_{H^1(0,T;(H^1_\mathcal{B}(\Omega))')} + \|g\|^2_{H^2(0,T;H^{-1}(\Omega))}
\end{aligned} \tag{2.49}$$

for some $C_1, C_3, C_4 > 0$. Note that by differentiating the weak form which we then test by $(0, \phi(t))$ we additionally obtain a bilinear form similar to Eq. (2.31) and can analogously deduce with Lax–Milgram lemma that

$$\|\dot{\phi}_m(0)\|^2_{H^1_0(\Omega)} \leq 2M \left(\|e\mathcal{B}\dot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)} + \|\dot{g}(0)\|^2_{H^{-1}(\Omega)} \right).$$

This is only possible because of the added requirement of increased regularity of $\dot{\mathbf{u}}(0)$ and $\dot{g}$. Furthermore, by the additional requirements on $\mathbf{u}_m(0) \in H^2(\Omega)$ we also obtain (estimating the H^2 norm by the norm of the Laplacian, see e.g. [70])

$$\|\phi_m(0)\|_{H^2(\Omega)} \leq C \left(\|\mathbf{u}_m(0)\|_{H^2(\Omega)} + \|g(0)\|_{H^{-1}(\Omega)} \right).$$

In order to utilize Gronwall lemma we are left to show finiteness of $\|\ddot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)}$. Notice that by the increased regularity of $\mathbf{u}_m(0) \in H^2(\Omega)$ and $\phi_m(0) \in H^2(\Omega)$ the weak solution is also a *strong* solution (not quite a *classical* solution but solves the classical equations in $t = 0$ almost everywhere, see e.g. [42, section 2.3 and 3.5]). Thus, by evaluating the strong system in $t = 0$ using the initial data and previously deduced inequalities we obtain

$$
\begin{aligned}
\|\rho\ddot{\mathbf{u}}_m(0)\|_{L^2(\Omega)} &= \|\alpha\rho\dot{\mathbf{u}}_m(0) - \mathcal{B}^T\left(c^E\mathcal{B}\mathbf{u}_m(0) + \beta c^E\mathcal{B}\dot{\mathbf{u}}_m(0) + e^T\nabla\phi_m(0)\right) \\
&\quad + f(0)\|_{L^2(\Omega)} \\
&\leq \|\alpha\rho\dot{\mathbf{u}}_m(0)\|_{L^2(\Omega)} + \|\mathcal{B}^T c^E\mathcal{B}\mathbf{u}_m(0)\|_{L^2(\Omega)} \\
&\quad + \|\mathcal{B}^T\beta c^E\mathcal{B}\dot{\mathbf{u}}_m(0)\|_{L^2(\Omega)} + \|\mathcal{B}^T e^T\nabla\phi_m(0)\|_{L^2(\Omega)} + \|f(0)\|_{L^2(\Omega)}.
\end{aligned}
$$

Note that this f is given by the Dirichlet Lift ansatz for the strong system. Therefor we choose $\chi \in H^2(\Omega)$ where $\chi|_{\Gamma_g} = 0$ and $\chi|_{\Gamma_e} = 1$. With this requirement the right hand sight of the above inequality is bounded independently of m.

Now that all components are finite, analogously to inequality Eq. (2.30) with

$$
\eta(t) := \|\ddot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} + \|\dot{\mathbf{u}}_m(t)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\dot{\phi}_m(t)\|^2_{H^1_0(\Omega)}
$$

we can apply Gronwall lemma to obtain

$$
\begin{aligned}
&\|\ddot{\mathbf{u}}_m(t)\|^2_{L^2(\Omega)} + \|\dot{\mathbf{u}}_m(t)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\dot{\phi}_m(t)\|^2_{H^1_0(\Omega)} \\
&\leq \left(\frac{\tilde{C}_3}{C_1} + \frac{C_4\tilde{C}_3}{C_1^2}te^{\frac{C_4}{C_1}t}\right)\left(\|\ddot{\mathbf{u}}_m(0)\|^2_{L^2(\Omega)} + \|\dot{\mathbf{u}}_m(0)\|^2_{H^1_{\mathcal{B}}(\Omega)} + \|\dot{\phi}_m(0)\|^2_{H^1_0(\Omega)}\right. \\
&\quad\left. + \|f\|^2_{H^1(0,T;(H^1_{\mathcal{B}}(\Omega))')} + \|g\|^2_{H^2(0,T;H^{-1}(\Omega))}\right)
\end{aligned}
$$

holds almost everywhere in $[0, T]$.

Using results from Thm. 2.10 it is now clear that

$$
\mathbf{u} \in L^\infty(0, T; H^1_{\mathcal{B}}(\Omega)), \quad \dot{\mathbf{u}} \in L^\infty(0, T; H^1_{\mathcal{B}}(\Omega)), \quad \ddot{\mathbf{u}} \in L^\infty(0, T; L^2(\Omega))
$$

$$
\phi \in L^\infty(0, T; H^1_{0,\Gamma}(\Omega)), \quad \dot{\phi} \in L^\infty(0, T; H^1_{0,\Gamma}(\Omega)).
$$

$\square$

Remark 2.12. Intuitively one may think that in order to achieve $\ddot{\mathbf{u}} \in L^\infty(0, T; L^2(\Omega))$ it is only required that $\dot{\mathbf{u}}(0) = \mathbf{u}_1 \in H^1(\Omega)$ instead of $\beta\mathbf{u}_1 \in H^2(\Omega)$ (or more precisely $\|\mathcal{B}^T \beta c^E \mathcal{B} \mathbf{u}_1\|_{L^2(\Omega)} < \infty$). However, this is not the case. The condition $\beta\mathbf{u}_1 \in H^2(\Omega)$ is required to show that $\|\ddot{\mathbf{u}}_m(0)\|_{L^2(\Omega)}$ is finite. This is essential in order to apply Gronwall inequality.

Remark 2.13. In e.g. [15, p. 390, Eq. (59)] a $H^2(\Omega)$ regularity for $\mathbf{u}$ is achieved by selecting the test functions for $\mathbf{u}$ to be the complete eigenfunction sequence of $-\Delta\mathbf{u}$ which, indirectly, allows an estimation of $\|\mathbf{u}\|_{H^2(\Omega)}$. A similar argument should also be possible for $\mathcal{B}^T\mathcal{B}$ (or more precisely the operator that works on the solution vector $\binom{\mathbf{u}}{\phi}$ and contains $\mathcal{B}^T\mathcal{B}$).

This would directly increase the regularity of ϕ so that not only $\mathbf{u} \in L^\infty(0, T; H^2(\Omega))$ but also $\phi \in L^\infty(0, T; H^2(\Omega) \cap H^1_{0,\Gamma}(\Omega))$.

However, this argumentation was not followed here. Note that the here occurring differential operators are slightly different from the Laplacian. Hence, this leads to rather unpleasant changes (due to the now very technical arguments and spaces). In that case, it would be possible to reduce the regularity requirements, however, at the cost of a more technical proof.

Corollary 2.14. *Let all requirements of Thm. 2.10 hold and let $\alpha, \beta > 0$ strictly. If additionally there exists a $t_0 \in \mathbb{R}$, $t_0 \geq 0$ such that $\phi^e(t) = 0$ for $t \geq t_0$, then*

$$\|\dot{\mathbf{u}}_m(t)\|_{L^2(\Omega)} \to 0, \quad \|\mathcal{B}\mathbf{u}_m(t)\|_{L^2(\Omega)} \to 0, \quad \|\phi_m(t)\|_{H^1_0(\Omega)} \to 0$$

and

$$\|\mathbf{u}_m(t)\|_{L^2(\Omega)} \to c \in \mathbb{R}$$

for $t \to \infty$.

Proof. The right hand side $\mathcal{F}_r(t)$ of the energy balance Eq. (2.23) is constant for $t \geq t_0$ as no new energy is given into the system starting from time t_0, i.e. $\mathcal{F}_r(t) = c_1 \in \mathbb{R}^{\geq 0}$ for $t \geq t_0$. Let

$$\gamma(t) := 2\alpha \int_0^t \langle \rho\dot{\mathbf{u}}_m(s), \dot{\mathbf{u}}_m(s) \rangle \, ds + 2\beta \int_0^t \langle c^E \mathcal{B}\dot{\mathbf{u}}_m(s), \mathcal{B}\dot{\mathbf{u}}_m(s) \rangle \, ds,$$

and let

$$\tilde{\eta}(t) := \langle \rho \dot{\mathbf{u}}_m(t), \dot{\mathbf{u}}_m(t) \rangle + \langle c^E \mathcal{B} \mathbf{u}_m(t), \mathcal{B} \mathbf{u}_m(t) \rangle + \langle \epsilon^S \nabla \phi_m(t), \nabla \phi_m(t) \rangle.$$

Then Eq. (2.23) implies that $\tilde{\eta}(t) + \gamma(t) = c_1$ for $t \geq t_0$. As $\gamma(t)$ is monotonically increasing, it follows that $\tilde{\eta}(t)$ is monotonically decreasing. Both $\tilde{\eta}(t), \gamma(t)$ are bounded below and above by zero and c_1, respectively. Hence, $\tilde{\eta}(t)$ and $\gamma(t)$ must converge. From this we can already deduce the following.

- Note that all summands of $\tilde{\eta}(t)$ and $\gamma(t)$ are non-negative. Thus, the non-negative integrands in $\gamma(t)$ must converge to zero as otherwise $\gamma(t)$ would be unbounded. Especially, we get $\langle \rho \dot{\mathbf{u}}_m(t), \dot{\mathbf{u}}_m(t) \rangle \to 0$.

- From the last point we note that ρ is a positive constant. This implies that $\dot{\mathbf{u}}_m(t) \to 0$ and therefore $\mathcal{B} \dot{\mathbf{u}}_m(t) \to 0$. Hence, $\langle c^E \mathcal{B} \mathbf{u}_m(t), \mathcal{B} \mathbf{u}_m(t) \rangle$ must converge to some unknown limit.

- We are aware that $\tilde{\eta}(t)$ converges. As all other summands of $\tilde{\eta}(t)$ converge the last remaining summand $\langle \epsilon^S \nabla \phi_m(t), \nabla \phi_m(t) \rangle$ must also converge to some unknown limit.

Thus, all summands in $\tilde{\eta}(t)$ converge to currently unknown values for $t \to \infty$. In order to identify these values we test the weak form Eq. (2.18) first with $(\mathbf{u}_m(t), 0)$ and get

$$\underbrace{\langle \rho \ddot{\mathbf{u}}_m(t), \mathbf{u}_m(t) \rangle}_{\to 0} + \alpha \underbrace{\langle \rho \dot{\mathbf{u}}_m(t), \mathbf{u}_m(t) \rangle}_{\to 0} + \langle c^E \mathcal{B} \mathbf{u}_m(t), \mathcal{B} \mathbf{u}_m(t) \rangle$$
$$+ \beta \underbrace{\langle c^E \mathcal{B} \dot{\mathbf{u}}_m(t), \mathcal{B} \mathbf{u}_m(t) \rangle}_{\to 0} + \langle e^T \nabla \phi_m(t), \mathcal{B} \mathbf{u}_m(t) \rangle = \underbrace{\langle f(t), \mathbf{u}_m(t) \rangle}_{=0} \tag{2.50}$$

to obtain

$$\lim_{t \to \infty} \langle e^T \nabla \phi_m(t), \mathcal{B} \mathbf{u}_m(t) \rangle = \lim_{t \to \infty} - \langle c^E \mathcal{B} \mathbf{u}_m(t), \mathcal{B} \mathbf{u}_m(t) \rangle \leq 0. \tag{2.51}$$

We now test a second time with $(0, \phi_m(t))$

$$\langle e \mathcal{B} \mathbf{u}_m(t), \nabla \phi_m(t) \rangle - \langle \epsilon^S \nabla \phi_m(t), \nabla \phi_m(t) \rangle = \underbrace{\langle g(t), \phi_m(t) \rangle}_{=0}$$

to obtain

$$\lim_{t\to\infty} \langle e\mathcal{B}\mathbf{u}_m(t), \nabla\phi_m(t)\rangle = \lim_{t\to\infty} \langle \epsilon^S\nabla\phi_m(t), \nabla\phi_m(t)\rangle \geq 0. \qquad (2.52)$$

Combining Eq. (2.51) and Eq. (2.52) we get

$$\lim_{t\to\infty} \langle \epsilon^S\nabla\phi_m(t), \nabla\phi_m(t)\rangle = 0 \text{ and } \lim_{t\to\infty} \langle c^E\mathcal{B}\mathbf{u}_m(t), \mathcal{B}\mathbf{u}_m(t)\rangle = 0.$$

From Eq. (2.32) we can deduce with the knowledge that $\|\mathcal{B}\mathbf{u}_m(t)\|_{L^2(\Omega)} \to 0$ that also $\|\phi_m(t)\|_{H_0^1(\Omega)} \to 0$. Thus, we have shown that all summands of $\tilde{\eta}(t)$ converge to zero.

Lastly, as the time and space derivative of $\mathbf{u}_m(t)$ converge to zero, we finally obtain $\|\mathbf{u}_m(t)\|_{L^2(\Omega)} \to c \in \mathbb{R}$ for $t \to \infty$. $\qquad\square$

2.3 Special Cases and Variations

Depending on the application at hand, oftentimes it is advantageous or necessary to solve a more specialized version of the partial differential equation. In this section the rotationally symmetric case, which can be used to simulate many disk-shaped piezoceramics, as well as the time-harmonic formulation of the partial differential equation are given.

As there are many different areas of application there are many other special cases which are not considered here. Hence, especially tailored theory would be required for these cases. For an example the author refers to [18, 19] where piezoelectric composites are considered which cannot be handled by the simpler model given here.

2.3.1 Rotationally Symmetric Case

In the case of disk-shaped piezoceramics with rotational symmetrical electrodes a full 3D simulation with coordinates (x, y, z) can be avoided by transforming into cylindrical coordinates (r, z). Following the notation of [35, p. 38] the differential

operators are defined as

$$\mathcal{B} := \begin{pmatrix} \frac{\partial}{\partial r} & 0 \\ \frac{1}{r} & 0 \\ 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial z} & \frac{\partial}{\partial r} \end{pmatrix}$$

and

$$\nabla := \begin{pmatrix} \frac{\partial}{\partial r} \\ \frac{\partial}{\partial z} \end{pmatrix}$$

where the material tensors are then reduced to

$$c^E = \begin{pmatrix} c_{11}^E & c_{12}^E & c_{13}^E & 0 \\ c_{12}^E & c_{11}^E & c_{13}^E & 0 \\ c_{13}^E & c_{13}^E & c_{33}^E & 0 \\ 0 & 0 & 0 & c_{44}^E \end{pmatrix},$$

$$e = \begin{pmatrix} 0 & 0 & 0 & e_{15} \\ e_{31} & e_{31} & e_{33} & 0 \end{pmatrix},$$

$$\epsilon^S = \begin{pmatrix} \epsilon_{11}^S & 0 \\ 0 & \epsilon_{33}^S \end{pmatrix}.$$

More details of the transformation can be found in [71]. Note that all ten material parameters of the full 3D simulation are still present in the rotational symmetric case. However, in the rotational symmetric case not all material parameters have the same potential of achieving a high sensitivity as influence of some parameters is severely limited, see also [31] or [38, p. 19].

2.3.2 Harmonic Formulation

In some situations it is advantageous from a numerical point of view to deal with a time-harmonic formulation of the partial differential equation Eq. (2.12) which, for example, would not require a time-stepping scheme to solve. The variant given here has been adapted from [34, 38].

The harmonic formulation can be deduced from the transient formulation Eq. (2.12) over the time domain $t \in [0, T]$ by applying a Fourier-transform resulting in a partial

differential equation in frequency domain $\omega \in \mathcal{F}$:

$$-\rho\omega^2\hat{u} - \mathcal{B}^T(\tilde{c}^E\mathcal{B}\hat{u} + \tilde{e}^T\nabla\hat{\phi}) = 0$$
$$-\nabla \cdot (\tilde{e}\mathcal{B}\hat{u} - \tilde{\epsilon}^S\nabla\hat{\phi}) = 0$$

$$(2.53)$$

Here, the mechanical displacement $\hat{u}$ and electrical potential $\hat{\phi}$ now depend on the (angular) frequency $\omega = 2\pi f$ instead of time t. Furthermore, by shifting into frequency domain, $\hat{u}$ and $\hat{\phi}$ as well as the material parameters in $\tilde{c}^E, \tilde{e}$ and $\tilde{\epsilon}^S$ are now complex-valued. These complex-valued parameters now, however, allow a generalization of the Rayleigh damping model. The conventional Rayleigh damping can be introduced by setting (e.g. [38])

$$\tilde{c}^E = \frac{1 + j\beta_0}{1 - j\alpha_0}c^E, \quad \tilde{e} = \frac{e}{1 - j\alpha_0}, \quad \tilde{\epsilon}^S = \frac{\epsilon^S}{1 - j\alpha_0}$$

with c^E, e and ϵ^S being the real valued material parameters as in Eq. (2.13) and $j = \sqrt{-1}$ the imaginary unit and considering the damping to now be a function of frequency

$$\alpha(\omega) := \alpha_0\omega, \quad \beta(\omega) := \frac{\beta_0}{\omega}$$

with $\alpha_0, \beta_0 \in \mathbb{R}^{\geq 0}$ the Rayleigh coefficients from the transient case. As some numerical packages cannot deal with complex valued partial differential equations the real and imaginary parts of the material parameters are stated here separately:

$$\tilde{c}^E = \frac{1 - \alpha_0\beta_0}{1 + \alpha_0^2}c^E + j\frac{\alpha_0 + \beta_0}{1 + \alpha_0^2}c^E, \quad \tilde{e} = \frac{e}{1 + \alpha_0^2} + j\frac{\alpha_0 e}{1 + \alpha_0^2}, \quad \tilde{\epsilon}^S = \frac{\epsilon^S}{1 + \alpha_0^2} + j\frac{\alpha_0\epsilon^S}{1 + \alpha_0^2}$$

Analogously, the partial differential equation Eq. (2.53) can also be stated in terms of real and imaginary parts. Due to the lengthy details these equations are given in Appendix A.2.

Existence and uniqueness of weak solutions of Eq. (2.53) have been discussed in [31].

2.4 Finite Element Method (FEM) for the Piezoelectric Equations

For numerical packages like *FEniCS* [3] the weak form Eq. (2.18) can be directly used for the finite element formulation. Otherwise, from the weak formulation Eq. (2.18) a finite element scheme can be deduced, see e.g. [34, 35, 38, 43, 71, 82].

Let the following matrices be $\mathbb{R}^{m \times m}$, where

$$
\begin{aligned}
M_{uu} &:= \left[\langle \rho v_j, v_k \rangle \right]_{kj}, & 1 \le j, k \le m \\
K_{\phi\phi} &:= \left[\langle \epsilon^S \nabla w_j, \nabla w_k \rangle \right]_{kj}, & 1 \le j, k \le m \\
K_{uu} &:= \left[\langle c^E \mathcal{B} v_j, \mathcal{B} v_k \rangle \right]_{kj}, & 1 \le j, k \le m \\
K_{u\phi} &:= \left[\langle e^T \nabla w_j, \mathcal{B} v_k \rangle \right]_{kj}, & 1 \le j, k \le m \\
K_{\phi u} &:= \left[\langle e \mathcal{B} v_j, \nabla w_k \rangle \right]_{kj}, & 1 \le j, k \le m.
\end{aligned}
$$

Then ϕ_m can be given by the electrostatic second part of the equation

$$
\phi_m := (K_{\phi\phi})^{-1} (\mathbf{g}_m - K_{\phi u} \mathbf{u}_m)
$$

and with $\mathbb{R}^m$ vectors given as

$$
\begin{aligned}
\mathbf{f}_m &:= \left[f, v_k \right]_k, & 1 \le k \le m \\
\mathbf{g}_m &:= \left[g, w_k \right]_k, & 1 \le k \le m
\end{aligned}
$$

we obtain

$$
\ddot{\mathbf{u}}_m + \alpha \dot{\mathbf{u}}_m + \beta M_{uu}^{-1} K_{uu} \dot{\mathbf{u}}_m + M_{uu}^{-1} \left(K_{uu} + K_{u\phi} K_{\phi\phi}^{-1} K_{\phi u} \right) \mathbf{u}_m = M_{uu}^{-1} \left(\mathbf{f}_m + K_{\phi\phi}^{-1} \mathbf{g}_m \right).
$$

Disregarding damping terms (a variation including damping terms can be found in e.g. [34, p. 395]) this can be restated in a convenient and compact matrix-vector notation [38]

$$
M \ddot{\underline{u}} + K \underline{u} = F
$$

with

$$M := \begin{pmatrix} M_{uu} & 0 \\ 0 & 0 \end{pmatrix}, \quad K := \begin{pmatrix} K_{uu} & K_{u\phi} \\ K_{\phi u} & -K_{\phi\phi} \end{pmatrix}$$

where

$$\underline{\mathbf{u}} := \begin{pmatrix} \mathbf{u}_m \\ \phi_m \end{pmatrix} \text{ and } F := \begin{pmatrix} f_m \\ g_m \end{pmatrix}$$

Analogously, for the harmonic case the following compact matrix-vector notation can be given:

$$\begin{pmatrix} -\omega^2 M_{uu} + K_{uu} & K_{u\phi} \\ K_{\phi u} & -K_{\phi\phi} \end{pmatrix} \begin{pmatrix} \hat{\mathbf{u}}_m \\ \hat{\phi}_m \end{pmatrix} = \begin{pmatrix} \hat{f}_m \\ \hat{g}_m \end{pmatrix}.$$

2.5 Practical Aspects of Solutions and CFS++

In this section we will take a closer look at the aspects of practical solving of these partial differential equations. For this, we will look into common and practical boundary conditions as well as the software *CFS++* used in this thesis.

2.5.1 CFS++

The research simulation software (see [33]) **C**oupled **F**ield **S**imulation (CFS++) written in C++ is hosted by Prof. Dr. Manfred Kaltenbacher at the Technical University of Vienna. It is mainly developed and maintained by research groups in Vienna and Erlangen-Nuremberg. The simulation capabilities of CFS++ specifically include, but are not limited to, advanced handling of piezoelectric systems (such as handling hysteresis effects not considered here). This does not come as a surprise as many authors in and around the involved research groups have worked on or are strongly related to piezoelectrics. Though CFS++ is not open-source it can in some sense be considered 'source-accessible' for academic purposes.

Thus, the main motivation behind choosing this software package is that its sources can be altered to feature the capability of computing derivatives using Algorithmic Differentiation tools like *ADOL-C* (which is the subject of chapter 4). This would otherwise not be possible as most alternative software packages capable of handling piezoelectrics are commercial and thus usually not source-accessible.

2.5.2 Types of Excitation and other Practical Aspects

Previously in this chapter we have only considered the excitation of the piezoceramic by means of Dirichlet boundary conditions of the potential $\phi(t) = \phi^e(t)$ on Γ_e and $\phi(t) = 0$ on Γ_g for the excited electrode Γ_e and grounded electrode Γ_g. This is certainly the most important and most used form of excitation. However, in practice there are many other possibilities in which a piezoceramic can be excited which can be reflected in conditions set to the partial differential equations. An overview over possible variations is given e.g. in [82, p. 25-26]. In the practical part of this thesis, unless specified otherwise, a charge pulse $Q(t)$ is used as a means of excitation (similar to [34, p. 407], see also [74, section 4.3]). For an arbitrary charge Q this can be achieved [82, p. 25] by applying a so-called 'weak constraint'

$$\int_{\Gamma_e} \left(e\mathcal{B}\mathbf{u} - \epsilon^S \nabla \phi \right) \cdot n \, d\Gamma_e = Q$$

for a prescribed charge Q (which may depend on time t or frequency ω). As the electrodes are conductive (and thus the charge is distributed equally on the loaded electrode) this specifies exactly one degree of freedom similarly to setting strong Dirichlet boundary conditions. Although the author of this thesis has not found documentation related to how this is achieved in CFS++ this could be integrated into the weak formulation and the solution method of such e.g. by means of additional Lagrange multipliers or by utilizing a Nitsche method e.g. [52].

Many other practical aspects of simulations with and around CFS++ and piezoceramics can be found in [34].

3

Simulation and Measurement of Impedance of Piezoceramics

One idea which arose as a result of the joint project of the research group *Mathematics and its Applications* and the *Measurement Engineering Group* was to increase the sensitivity of impedance with respect to material parameters by deploying a triple-ring electrode setup on the piezoceramic specimen. The aim of this chapter is to derive the fundamental equations for the electrical circuit which is needed to measure a combined impedance curve for this setup. The network equations presented here have been adapted from [74].

In real-world applications piezoelectric ceramics come in many shapes and sizes. The ceramics are usually geometrically processed in such a way that the geometry fits the specific requirements for the application at hand. In this thesis we focus on disk-shaped ceramics. However, the methodology can easily be extended or adapted to deal with different shapes.

3.1 Electrical Impedance for Piezoelectric Ceramics with Fully Covering Electrodes

In electrical circuits the resistance R describes the ratio of voltage V across and current I through an object using direct current (DC). Hence, Ohm's law states

$$R = \frac{V}{I}.$$

This concept can be extended to electrical circuits driven by alternating current (AC). When alternating current is used, the sinusoidal voltage V and current I can be given by their amplitude and phase e.g.

$$V(t) = v_{max} \cdot \cos(\omega t + \varphi_v)$$

where $v_{max} \in \mathbb{R}^+$ is the amplitude, $\omega \in \mathbb{R}^+$ is the angular frequency and φ_v is the phase shift. For computations with these values it is convenient to use a complex representation, e.g.

$$V(t) = \text{Re}\left(v_{max} \cdot e^{j(\omega t + \varphi_v)}\right)$$

with $j := (-1)^{\frac{1}{2}}$ the imaginary unit.

The *electrical impedance* of a component is defined as the ratio of the complex voltage and complex current

$$Z = \frac{v_{max} \cdot e^{j(\omega t + \varphi_v)}}{i_{max} \cdot e^{j(\omega t + \varphi_i)}} = \frac{v_{max}}{i_{max}} e^{j(\varphi_v - \varphi_i)}.$$

Hence, the electrical impedance $Z \in \mathbb{C}$ describes the complex valued resistance when alternating current is applied.

For measuring and simulating the electrical impedance of a piezoceramic specimen it is common to fully cover the top and bottom of the (non-conductive) piezoelectric ceramic with conducting material (electrodes). The electrodes are then electrically excited by an AC current source. In Fig. 3.1 such a setup is displayed.

In order to compute the electrical impedance from the solutions of the PDE (2.14) only values derived from the mechanical displacement u and electrical potential ϕ can be used.

One possibility which will be used here has been suggested in e.g. [34, p. 406]. It makes use of a prescribed electrical charge $q(t)$ on the excited electrode as described in section 2.5.2. Hence, given a prescribed charge $q(t)$, the flow of charge $\dot{q}$ (the time derivative of charge q) is the electrical current I at the electrode. Thus, the electrical impedance can be computed as the quotient

$$Z(t) = \frac{V}{I} = \frac{\phi^e(t)}{\dot{q}(t)}$$
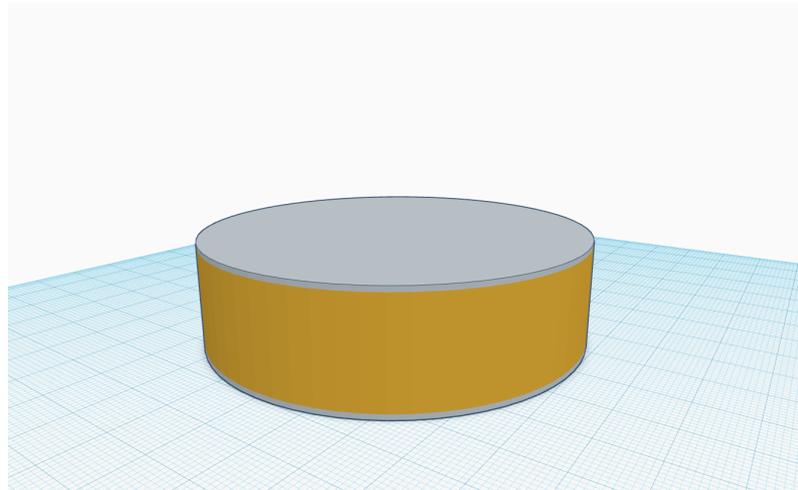
(a) 3D representation of a piezoceramic and electrodes
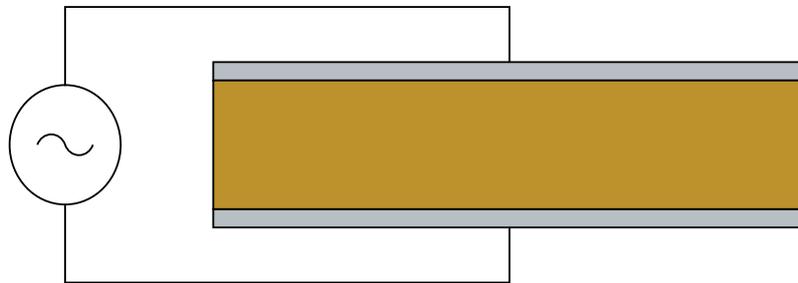
(b) Applied electrical network

Figure 3.1: Piezoelectric ceramic (brown) with attached electrodes (gray) fully covering top and bottom.

where $\phi^e(t)$ is the computed potential solution of the partial differential equation at the charged electrode Γ_e. As the circuit is AC driven we are interested in how the impedance behaves dependent on frequency instead of time. Hence, these signals are converted into frequency domain by applying a Fourier transform $\widehat{\cdot}$. We obtain

$$Z(\omega) = \frac{\widehat{\phi}(\omega)}{\widehat{\dot{q}}(\omega)} = \frac{\widehat{\phi}(\omega)}{j\omega\widehat{q}(\omega)}.$$

Note that the mechanical displacement u is not needed here for the impedance computation but would be required (for computing the charge) if a Dirichlet boundary condition of the potential was used instead of a charge load.

As the impedance is a complex value, it is often convenient and/or necessary for further usage to keep only the (real-valued) absolute value $|Z(\omega)|$ and to disregard the phase of the impedance. Taking this into account, the piezoceramic disk resonantes at specific frequencies which can be seen in the frequency dependent impedance curve. Hence, the impedance curve, or more precisely the absolute value of the impedance curve, has a very distinct appearance where resonance and antiresonance appear in characteristic pairs at certain frequencies (see Fig. 3.2). For example, according to e.g. [22] for a disk-shaped piezoceramic (as seen in Fig. 3.1) the so-called thickness modes f_n, $n \in \mathbb{N}$ can be estimated to be at

$$f_n = n\frac{c}{2T}, \quad n \in \mathbb{N}$$

where T is the thickness of the ceramic disk and c is the speed of sound depending on the material used. More detailed estimates on the location and types resonances can be found e.g. in [22].

3.2 Triple-ring Electrodes

A common issue with the identification of material parameters of piezoelectric ceramics using the impedance curve is that the sensitivity of impedance with respect to some material parameters is very low and may be close to zero.

In order to explain this behavior consider the field lines of the electrical field for
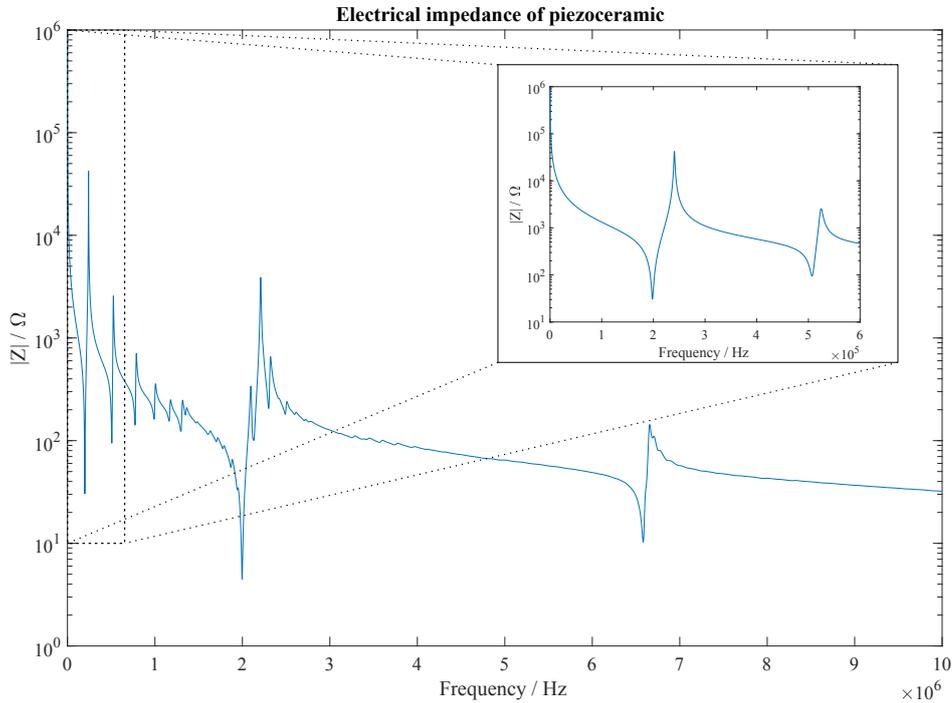
Figure 3.2: Typical impedance curve for a piezoelectric ceramic. Real measurement data provided by N. Feldmann of the Measurement Engineering Group.

a piezoceramic disk with electrodes fully covering top and bottom of the disk. In Fig. 3.3 the field lines are shown.

Notice that the field lines only point in the direction top to bottom or bottom to top (depending on which electrode has been charged). In this case the material parameters describing the behavior of the electrical field in horizontal direction do not have any influence on the field lines. Hence, the electrical impedance of the specimen does not change if these specific parameters had different values. However, for parameter identification via the impedance curve this has the consequence that these parameters cannot be reconstructed by such measurements. The behavior of the impedance curve for such material parameters can also be stated alternatively: the sensitivity of impedance with respect to these material parameters is zero.

Motivated by this behavior the authors of [76], [37] developed the idea to circumvent the homogenous field lines of the electrical field by deploying a triple-ring electrode setup on the piezoceramic. This setup generates non-homogenous field lines that now
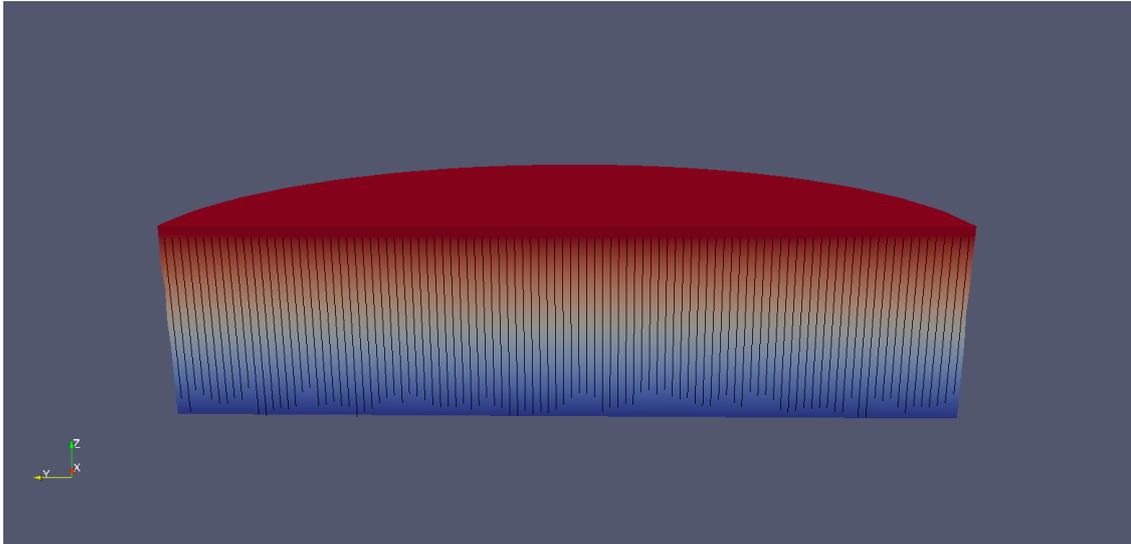
(a) Electrical field in a piezoceramic.
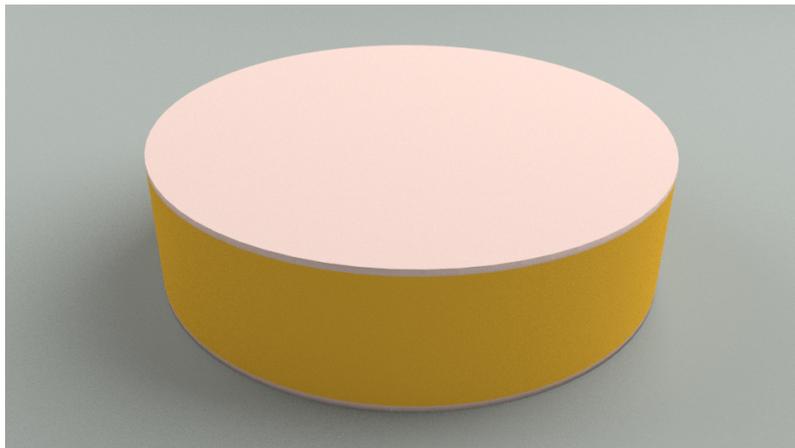
(b) 3D representation of a piezoceramic.

Figure 3.3: Electrical field lines for a usual piezoceramic with fully covering electrodes on top and bottom.

point in multiple directions instead of only one. This way, the electrical impedance of the specimen now is actively influenced by material parameters that work in other than the vertical direction.

As the piezoceramic has three electrode rings (as opposed to two full faced electrodes covering top and bottom), the calculation of impedance of the whole device requires an intermediate step: We electrically excite via pulse charge $q(t)$ one of the
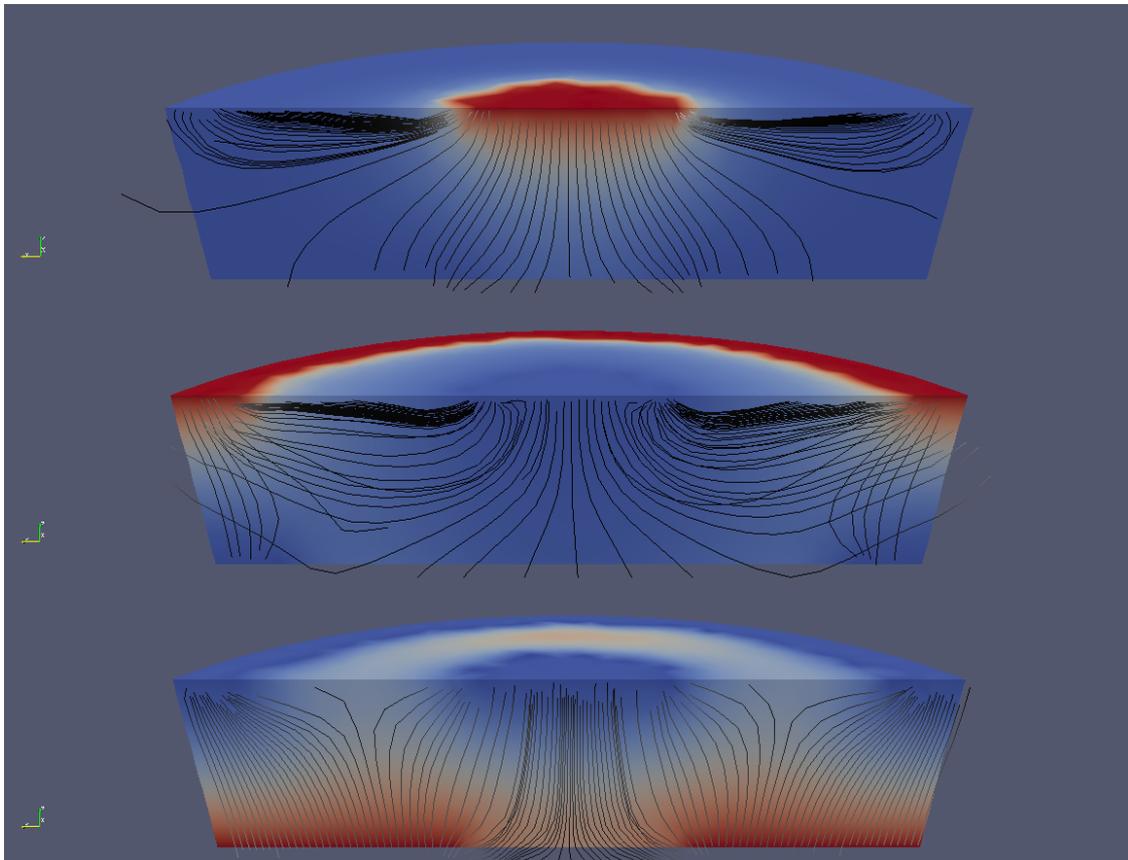
(a) Electrical field in a piezoceramic.
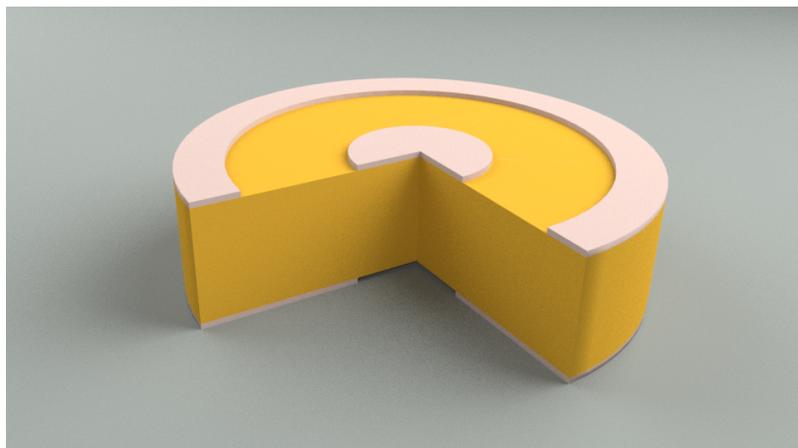
(b) 3D representation of a piezoceramic and electrodes.

Figure 3.4: Electrical field lines for the three triple-ring setup cases.

three electrodes and ground another electrode by imposing an appropriate boundary condition to the partial differential equation. We calculate the impedance between the electrode where the charge was imposed and the single remaining ungrounded electrode by dividing the electric potential at the rings ϕ by the current $\dot{q}$ flowing at the rings. An additional Fourier transform $\widehat{\cdot}$ gives the impedances

$$Z_i(\omega) = \frac{\widehat{\phi_i}(\omega)}{\widehat{\dot{q}}(\omega)} \in \mathbb{C}. \tag{3.1}$$

This is done by using each possible combination of electrically excited and grounded electrodes, hence $i \in \{1, 2, 3\}$. Furthermore, as there are three possible combinations we also need to solve Eq. (2.14) three times separately with according boundary conditions.

3.3 Equations of Network and Derivatives

By applying an additional network of electrical components (see Fig. 3.5) to the three electrodes and using a delta-star conversion (or solving Kirchhoff's laws e.g. [9]) these three impedances are finally combined to a total impedance Z. This allows for a single measurement to be used for comparison with the simulation results. Resistance and capacitance of the resistors and capacitors used in the electrical network have been pre-optimized in prior work by Unverzagt [75]. This was done to further increase the sensitivity of the impedance measurement. However, this step could/should also be repeated after an optimal ring geometry has been determined.

Given the previously mentioned three impedances Z_i, $i \in \{1, 2, 3\}$, obtained by either solving the PDE (2.14) with different boundary conditions and using the formula Eq. (3.1) or by direct physical measurements we now aim to compute the overall impedance. The reader may also consult Unverzagt [75] for more details on the very similar procedure.

We first consider a delta-star or delta-Y transformation (see, e.g., [9]) of only the three inner impedances

$$Z_a^\Delta := \frac{2}{\frac{1}{Z_1} - \frac{1}{Z_2} + \frac{1}{Z_3}}, \quad Z_b^\Delta := \frac{2}{\frac{1}{Z_1} + \frac{1}{Z_2} - \frac{1}{Z_3}}, \quad Z_c^\Delta := \frac{2}{-\frac{1}{Z_1} + \frac{1}{Z_2} + \frac{1}{Z_3}}.$$
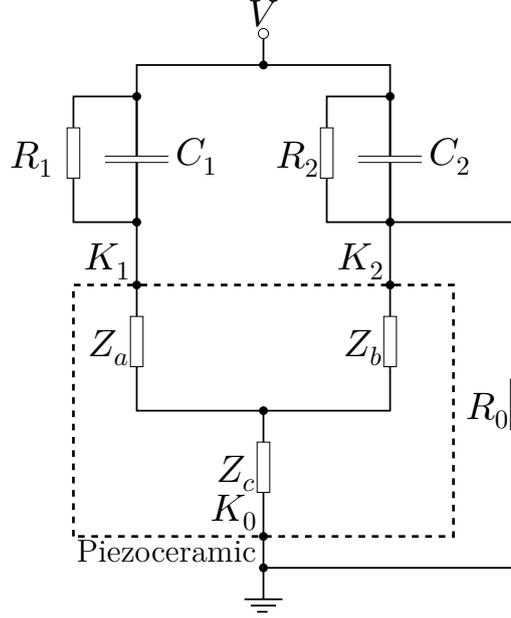
Figure 3.5: Network of electrical components for triple-ring electrodes. Adaptation from Unverzagt [75].

The resulting star-circuit values are given by

$$Z_a^Y := \frac{Z_b^\Delta \cdot Z_c^\Delta}{Z_a^\Delta + Z_b^\Delta + Z_c^\Delta}, \quad Z_b^Y := \frac{Z_a^\Delta \cdot Z_c^\Delta}{Z_a^\Delta + Z_b^\Delta + Z_c^\Delta}, \quad Z_c^Y := \frac{Z_a^\Delta \cdot Z_b^\Delta}{Z_a^\Delta + Z_b^\Delta + Z_c^\Delta}.$$

Now, we also add the external resistor R_0

$$\tilde{Z}_a^Y := \frac{Z_b^\Delta \cdot Z_c^\Delta \cdot R_0}{Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta + R_0 \cdot (Z_a^\Delta + Z_b^\Delta + Z_c^\Delta)}$$

$$\tilde{Z}_b^Y := \frac{Z_a^\Delta \cdot Z_b^\Delta \cdot Z_c^\Delta + Z_a^\Delta \cdot Z_c^\Delta \cdot R_0}{Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta + R_0 \cdot (Z_a^\Delta + Z_b^\Delta + Z_c^\Delta)}$$

$$\tilde{Z}_c^Y := \frac{Z_a^\Delta \cdot Z_b^\Delta \cdot R_0}{Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta + R_0 \cdot (Z_a^\Delta + Z_b^\Delta + Z_c^\Delta)}.$$

And lastly, we add the resistors and capacitors R_1, R_2, C_1, C_2

$$Z_{v1} := \frac{R_1}{1 + 2\pi j f \cdot R_1 \cdot C_1}, \quad Z_{v2} := \frac{R_2}{1 + 2\pi j f \cdot R_2 \cdot C_2}, \quad Z_{vp} := \frac{1}{\frac{1}{\tilde{Z}_b^Y + Z_{v1}} + \frac{1}{\tilde{Z}_c^Y + Z_{v2}}}.$$

Hence, the overall impedance can be computed as

$$Z_{ges} := \tilde{Z}_a^Y + Z_{vp}.$$

Remark 3.1. The derivatives of these network equations are required for gradient based optimization. As these network computations are done outside the scope of the FEM tool the derivatives are not automatically computed. The first derivative was computed by hand and is given in Appendix A.1. It was not possible to compute the second derivative by hand. Even more: the derivative computed by a popular computer algebra software produced a huge formula with more than 4 million elemental commands which would require an unpractical amount of runtime to evaluate (i.e. infeasible for all practical purposes). As analytical derivatives increase exponentially in length this behavior does not come as a surprise. The author then applied the Algorithmic Differentiation tool *ADiMat* [6] to the first derivative computed by hand and was able to retrieve the short (relative to the computer algebra solution) and correct second derivatives. As the derivative formulas for the second derivatives are still fairly lengthy they are not given in this thesis.

In Algorithm 1 the complete algorithm to evaluate the overall impedance for a triple-ring electrode setup is displayed.

Algorithm 1: Compute overall impedance

    Input: material parameters p, radii geometry r, equidistant discrete time
           points $T_n = \{0, \Delta t, \ldots, n\Delta t\}$
    Output: overall impedance Z

1   q(t) = Compute excitation pulse charge for $t \in T_n$ as in [34, p. 406]
2   Compute current $I(t) = \frac{\partial q(t)}{\partial t}$
3 $\{\omega_0, \ldots, \omega_n\}$ = compute frequency points from discrete time steps T_n
4   Compute Fourier transform $\hat{I}(\omega) = \mathtt{fft}(I(t))$
5   **for** $i = 1, \ldots, 3$ **do**
6 $\phi_i(t)$ = solve PDE case for given i, p, r, T_n
7   Compute Fourier transform $\hat{\phi}_i(\omega) = \mathtt{fft}(\phi_i(t))$
8 $Z_i = \hat{\phi}_i(\omega)/\hat{I}(\omega)$
9   **end**
10 Z = compute the overall impedance as in section 3.3 for Z_1, Z_2, Z_3
11   Return Z

3.4 A Novel Extension

While in the process of writing this thesis the author had an idea for a novel extension of how the single impedance measurements in the triple-ring electrode case can be used. This approach allows directly prescribing any sensitivity. As this is slightly outside of the set scope of this thesis only the basic idea and some basic results are presented. However, in future the author plans to investigate the following phenomenon further. The author is not aware of any similar approaches at least in the field of piezoelectricity.

Because the impedance computation via a network is motivated by real physical measurements it is a 'natural choice' as the measurable quantity of the inverse problem. One of the main benefits of this choice is that the resulting impedance curve and the influence of the parameters may be interpreted (e.g. it is well known that some material parameters mainly influence specific resonance frequencies). Moreover, it is expected that this type of influence is valid even for varying material parameters (i.e. the specific influence of some of the material parameters is expected to be valid globally instead of only locally). As solution methods for inverse problems are usually very problem specific, this can help while designing a concrete inverse scheme.

By sacrificing this problem knowledge, it is, in fact, possible to achieve *any* prescribed sensitivity for a given set of material parameters. However, this prescribed sensitivity then may hold only locally.

Consider the case where we have a more general electrode geometry. This especially could mean applying more than three electrodes, i.e. $n \in \mathbb{N}$ electrodes, on the piezoceramic. However, the simple delta-Y conversion then cannot be applied. For more than three components there is no conversion formula for a star circuit with n components to a fully-connected circuit with n nodes. Hence, in this case a more general approach e.g. by using Kirchhoff's laws on each node could make it possible to compute the overall impedance and it's derivatives. Note that in this case the resulting linear system of equations generated by Kirchhoff's laws is underdetermined and further conditions determined by the exact setup are required.

In this context it seems wise to define a simpler combination of single impedance

measurements as the measurable quantity for the inverse problem instead of the over-all impedance. A simple alternative could be the weighted sum of single impedance measurements

$$Z(\omega) := \gamma_1(\omega)Z_1(\omega) + \cdots + \gamma_m(\omega)Z_m(\omega) = \sum_{1 \leq i \leq m} \gamma_i(\omega)Z_i(\omega)$$

for a fixed frequency ω, fixed weights $\gamma_i \in \mathbb{R}$ and m different impedance measurements $Z_1, \ldots, Z_m$. For convenience we will also call this weighted sum *impedance* though it is not technically correct. Let us assume for the moment we use 5 rings (as $\max_{0 \leq k \leq 4} \binom{4}{k} < 10$) and obtain 10 measurements curves Z_i. Then, the derivative of Z with respect to the material parameters is given by

$$\frac{\partial Z}{\partial p_j} = \sum_{1 \leq i \leq 10} \frac{\partial Z_i}{\partial p_j} \gamma_i.$$

Using matrix-vector notation this can be rewritten in terms of a matrix vector product with $A_{ij} := \frac{\partial Z_i}{\partial p_j}$ and $\gamma := [\gamma_i]_{1 \leq j \leq 10}$. We obtain

$$A\gamma = \left[\frac{\partial Z}{\partial p_j} \right]_{1 \leq j \leq 10}.$$

As we are using a multiple ring electrode setup based on this thesis none of the derivatives are zero, linear dependent or badly scaled in relation to eachother and we can safely assume that A is a full rank matrix with relatively low condition number. In this case we can prescribe arbitrary values for the derivatives e.g. $b = (b_1(\omega), \ldots, b_{10}(\omega))^T$ by instead solving the linear system of equations for γ:

$$\underbrace{\begin{pmatrix} \frac{\partial Z_1}{\partial p_1} & \frac{\partial Z_1}{\partial p_2} & \cdots & \frac{\partial Z_1}{\partial p_{10}} \\ \frac{\partial Z_2}{\partial p_1} & \frac{\partial Z_2}{\partial p_2} & \cdots & \frac{\partial Z_2}{\partial p_{10}} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial Z_{10}}{\partial p_1} & \frac{\partial Z_{10}}{\partial p_2} & \cdots & \frac{\partial Z_{10}}{\partial p_{10}} \end{pmatrix}}_{=A} \underbrace{\begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_{10} \end{pmatrix}}_{=\gamma} = \underbrace{\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{10} \end{pmatrix}}_{=b}$$

In fact: in order to create a full rank matrix A we do not necessarily require 5 electrodes. We can also generate 10 linear independent measurements from a triple

ring setup by dividing the measurements we already have into 10 segments. There are many possible ways to achieve this. One option will be given here. In order to avoid confusion we will denote the 10 final measurement curves by $\tilde{Z}_1, \ldots, \tilde{Z}_{10}$.

First, the amount of measurements can be doubled by combining measurements of neighboring discrete frequency points ω_i, ω_{i+1}. We obtain from the discrete frequency points $\{\omega_0, \ldots, \omega_{2N}\}$ the set $\{\tilde{\omega}_0, \ldots, \tilde{\omega}_N\}$ (calling the new domain the *combined frequency domain*).

Second, as the impedance measurements are complex valued we can separate some of the single impedance measurements into their real and imaginary parts:

$$Z_1 = \operatorname{Re} Z_1 + j \operatorname{Im} Z_1, \quad Z_2 = \operatorname{Re} Z_2 + j \operatorname{Im} Z_2, \quad Z_3 = \operatorname{Re} Z_3 + j \operatorname{Im} Z_3$$

Thus we obtain for $0 \leq i \leq N$ the following 10 measurements curves

$$\begin{aligned}
\tilde{Z}_1(\tilde{\omega}_i) &= \operatorname{Re} Z_1(\omega_{2i}), & \tilde{Z}_2(\tilde{\omega}_i) &= \operatorname{Im} Z_1(\omega_{2i}), & \tilde{Z}_3(\tilde{\omega}_i) &= \operatorname{Re} Z_2(\omega_{2i}), \\
\tilde{Z}_4(\tilde{\omega}_i) &= \operatorname{Im} Z_2(\omega_{2i}), & \tilde{Z}_5(\tilde{\omega}_i) &= \|Z_3(\omega_{2i})\|, \\
\tilde{Z}_6(\tilde{\omega}_i) &= \operatorname{Re} Z_1(\omega_{2i+1}), & \tilde{Z}_7(\tilde{\omega}_i) &= \operatorname{Im} Z_1(\omega_{2i+1}), & \tilde{Z}_8(\tilde{\omega}_i) &= \operatorname{Re} Z_2(\omega_{2i+1}), \\
\tilde{Z}_9(\tilde{\omega}_i) &= \operatorname{Im} Z_2(\omega_{2i+1}), & \tilde{Z}_{10}(\tilde{\omega}_i) &= \|Z_3(\omega_{2i+1})\|.
\end{aligned}$$

Using this setup we can prescribe any arbitrary values for the derivatives.

In Fig. 3.6a the derivative with respect to c_{13} was chosen to be a sine curve, the other derivatives have integer values ranging from 0 to 9. This example was selected to show that any arbitrary sensitivity can be reached.

In Fig. 3.6b for each point $\tilde{\omega}_i$ it was chosen to have only one derivative with the value of 1, all other values are zero. After 20 frequency points the parameter with derivative 1 is replaced by a different one. This example was selected to demonstrate that it is possible that each parameter can be assigned a fixed bandwidth where no other parameter shows any sensitivity.

Obviously this approach can also be used to set all the derivatives of the impedance Z with respect to the material parameters to 1.0. Hence, each parameter would have the same influence on the new impedance.

However, currently not enough data has been gathered to show if and how material

parameter identification methods can benefit from this approach.

Remark 3.2. Under the assumption of an impedance function that is at least once continuously differentiable with respect to the material parameters the derivatives of the new overall impedance function with respect to material parameters are at least continuous. The author is unaware of any cases where the derivatives were not at least continuous. Thus, this assumption seems to be reasonable.

Furthermore, under this assumption it can be guaranteed that the derivatives locally do not change too much from the prescribed values when evaluating for different material parameters. This may or may not be true globally and may vary for different prescribed values. For future research it may be worthwhile to identify variations of this method which guarantee a desired level of sensitivity for all parameters simultaneously for a large subset of all possible material parameters. One possible approach could be combining this idea with work from e.g. [39].

Remark 3.3. This idea is based on the fact that the matrix A containing the derivatives is a full rank matrix with a relatively low condition number. This is only possible because the triple-ring electrode geometry developed in this thesis already provides optimized sensitivities that are not too small and not badly scaled in relation to eachother. Otherwise, if the matrix A had a large condition number (e.g. a nearly singular matrix) small measurement errors could be amplified by this procedure. Hence, this idea is a novel extension firmly based on the current work rather than an entirely different approach.

However, the author does not see any reason why this idea would not work with different approaches to increased sensitivity such as using multiple differently shaped ceramics or using additional surface displacement measurements. Furthoremore, it seems that this idea could also be used for different inverse problems that need not be related to piezoelectrics where low sensitivity also is a prevailing issue.
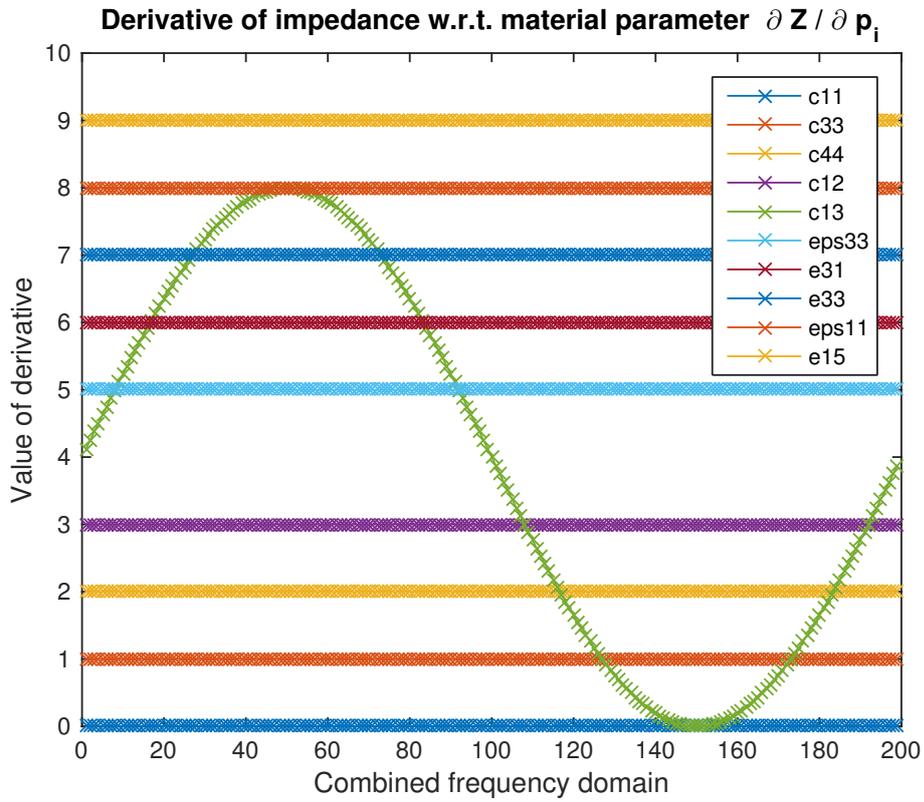
(a) Example 1
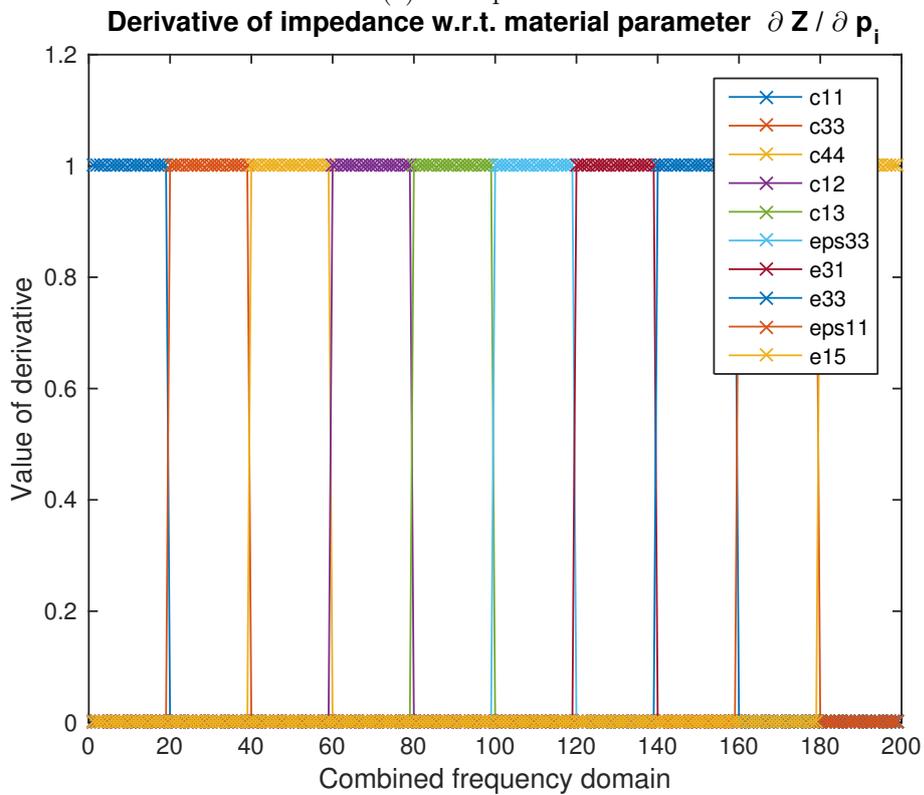
(b) Example 2

Figure 3.6: Different examples for prescribed derivatives.

4

Computing Sensitivity

The aim of this chapter is to define and compute the sensitivity of impedance with respect to material parameters. These sensitivities predominantly consist of derivatives that need to be computed. Hence, one essential technique used and developed on for this thesis is Algorithmic Differentiation (AD). Parts of this chapter have been published in [29].

4.1 Algorithmic Differentiation

Algorithmic Differentiation or Automatic Differentiation is a technique which enables the computation of derivatives of 'arbitrary' computer programs. AD is neither symbolic differentiation nor numerical differentiation (in the sense of finite difference schemes). Some main advantages of using AD over other differentiation techniques are:

- The computed derivatives are provable accurate up to machine precision under almost all practical circumstances.

- Low provable runtime and memory consumption. Derivatives can be computed in a small multiple of the runtime of one function evaluation. For example for scalar valued functions this constant $\leq 5/2$ is independent of the number of variables (inputs) the function has.

A comprehensive introduction as well as advanced topics on AD can be found in e.g. [21]. As there are already numerous publications dealing with the basic concepts of Algorithmic Differentiation there is no need to provide yet another introduction here. Thus, for this thesis it is assumed that the reader is already familiar with the basic concepts.

4.2 Instrumentation of ADOL-C in CFS++

Parts of the following sections containing more detailed information have been published by the author in [29].

For this work the AD software tool ADOL-C (Algorithmic Differentiation via Operator Overloading in C++) was applied. ADOL-C (see e.g. [80, 81]) is developed and maintained at the research group *Mathematics and its Applications* of Prof. Dr. Andrea Walther. In order to apply ADOL-C some preparations of the target code are necessary. The aim of this section is to outline some of the unusual challenges which arose while adapting CFS++ for derivative computation. Due to the technical nature of the details we will focus here only on string operation related issues. Furthermore, as it is central to some important decisions made in this thesis some issues regarding the choice of a so-called *traceless* AD mode, instead of a *traced* AD mode, are given in Appendix A.3. However, much more details are given in [29].

ADOL-C introduces the data type `adouble` which manages and keeps track of derivatives. The data type is used by replacing the usual `double` data type by an `adouble` in the target code. Then, in the target code if an operation is called of which the derivative is to be computed the `adouble` data type prevents that the usual operation is called. Instead an `adouble` operation with the same name is called which also takes care of the derivative computation. The concept of implicitly replacing an original operation by a second operation with the same name is called *operator overloading*. Operator overloading based AD tools are one of two major classes of AD tools. The other major class uses a technique called *source transformation* where the target code is analyzed and augmented by the tool.

For an example some C++ code, say the function `double f( double a, double b)` is considered. It is necessary to change the data type of all variables directly involved and relevant to the derivative computation (this is referred to as an *active variable*) to the data type `adouble`. Hence one could overload the function `double f(double a, double b)` to `adouble f(adouble a, double b)` assuming that only `a` is directly involved and `b` is not. We refer to the selective change of data type at exactly the locations where a change is required for correct derivative computation as *type-insertion*.

Conversely, a 'brute-force' approach is to define the data type `double` to in fact be an `adouble` in all parts of the C++ code. This is usually done by including code similar to the line `#typedef adouble Double` in the main include file and using `Double` instead of `double` throughout all the code. Thus, this approach is also known as the *typedef approach*.

Type-insertion clearly results in a faster program code: The custom `adouble` data type has larger overhead compared to the `double` data type. If an `adouble` is used where it is not needed the resulting code will not run at the optimal speed. However, in large and complicated code bases it is usually very demanding to implement operator overloading based AD with type-insertion as the type change of interface parameters (i.e. type change in method signatures) will render the interface unusable (signature mismatch) without also adapting the callers interface parameters. In large and complicated code bases this manual adaptation might be necessary for nearly every method, object property, temporary variable, etc.

In the latest implementation of ADOL-C into CFS++ we used a mix of both approaches: We used the *typedef approach* as a general guideline and type-insertion for handling more specific implementation issues of which one important aspect for CFS++ will be discussed in the next subsection.

The changes made for this thesis in the local CFS++ implementation of the author have been made available to anyone with access to CFS++. Furthermore, an enhanced version which is not limited to piezoelectric problems has already been or will shortly be added to the main development branch of CFS++ so that this adaptation can be utilized by anyone with access to CFS++ for their simulation.

Arithmetic Expressions with Strings

To allow non-constant, i.e., temperature dependent material parameters, among other entities, are incorporated into CFS++ via scalar, vectorial or tensorial functions called `CoefFunction` defined on the computational domain. As an example, consider the following method declaration:

```
static PtrCoefFct
Generate( MathParser * mp, Global::ComplexPart type, UInt numRows, UInt numCols,
    const StdVector<std::string>& realVal, const StdVector<std::string>& imagVal
    = StdVector<std::string>() );
```

which returns a `PtrCoefFct`, a pointer to a `CoefFunction`. This method has the following usage (marked gray, line 6):

```
1   if (xpr.GetDimType() == CoefFunction::TENSOR )
2   {
3       StdVector<std::string> real, imag;
4       UInt numRows, numCols;
5       xpr.GetTensorXpr( numRows, numCols, real, imag );
6       ret = Generate( mp, part, numRows, numCols, real, imag );
7       [...]
8   }
```

Note that the method depends on a `MathParser mp` and vectors of `std::string` for the real and imaginary parts `real` and `imag` of the `CoefFunction` (marked gray, line 3). It is emphasized that the material parameters are handled as `CoefFunction` and thus mostly as strings and not via `double` or `complex<double>`. The strings containing the material data, but also other computations, are then also combined into larger strings via composition. Hence, large `string` formulas can be created by this process. Consider the following example where the unary operation `sqrt` is performed on a `CoefFunction` (see lines marked in gray):

```
1   void CoefXpr::ApplyUnaryFunc( std::string& retReal, const std::string& argReal,
        OpType op ) {
2       StdVector<std::string> args;
3       switch( op )
4       {
5       [...]
6       case OP_SQRT:
7         args = "sqrt(", B(argReal), ")";
8         retReal = B(args.Serialize('␣'));
9         break;
10      [...]
11      }
12  }
```

where

```
1   inline std::string B(const std::string& xpr ) {
2     return "(" + xpr + ")";
3   }
```

is a helper function returning the same string surrounded by brackets. Hence, let `argReal` be `"123456.0"` which, for the sake of this example, could be one of the material parameters considered. Then `retReal` would be returned as the string `"sqrt( (123456.0) )"`. A subsequent reapplication would thus lead to the string

`"sqrt( (sqrt( (123456.0) )) )"`. Whatever formula was written inside `argReal` would be surrounded by the operation `"sqrt( )"`. These generated string formulas, which can also contain variables (see below), are then parsed and evaluated by the `MathParser`. Although in our case only constant material parameters are used, these values are obligatorily evaluated by this `MathParser mp` - a wrapper for the mathematical string parser external library `MuParser` [5].

MuParser is a bitcode generating parser for mathematical formulas. According to [5], the speed of frequent evaluations of a specific formula with varying variable values can be significantly increased by saving and reusing constant parts of generated bitcode for the evaluation to disk thus avoiding future overhead for multiple similar formula evaluations.

Since ADOL-C only operates on numerical types, currently, both operations in the `string` data type and bitcode cannot be differentiated in CFS++. An approach to differentiate `string` formulas in MuParser is presented in the next section. However, due to the requirement of an additional external package as well as the consent of the main CFS++ developers, this has not yet been implemented in CFS++.

Instead, in order to have matching interface data types (i.e. matching method signatures in `adouble` data type) in CFS++, MuParser is compiled using the `adouble` data type despite custom data types not officially supported by MuParser. We verified that this does not change the primal results. Furthermore, this also has no noticeable impact on runtime as there is only a relatively small number of string formulas parsed compared to the main code. Despite the `adouble` data type it does not, however, have the functionality to compute derivatives using AD. Hence, the use of MuParser breaks the differentiation chain at any point it is used.

Fortunately, for these specific cases of sensitivity evaluation for piezoelectric transducers we were able to identify special locations in CFS++ where we could find workarounds for these issues: Special case conditionals circumvent these issues and custom values are set to correct the values computed by CFS++. However, this only works for this one specific case of sensitivity evaluation and does not replace general case solutions.

4.3 Computing Derivatives in MuParser Applications

In this subsection a method to handle MuParser usage under certain assumptions is presented. In order to achieve this, no changes to the MuParser package are required. Note that even if the MuParser issues are fully dealt with, this does not imply that all string related issues within CFS++ have been resolved. However, string operations unrelated to MuParser (such as used in `CoefFunction`) could probably be dealt with in a similar fashion.

As the assumptions discussed below have currently not been implemented in CFS++ we will instead work, for the reader's convenience, on a slightly simplified version of the default example given on the MuParser homepage:

```cpp
#include <iostream>
#include "muParser.h"

// Function callback
double MySqr(double a_fVal)
{
  return a_fVal*a_fVal;
}

// main program
int main(int argc, char* argv[])
{
  using namespace mu;

  try
  {
    double fVal = 1;
    Parser p;
    p.DefineVar("a", &fVal);
    p.DefineFun("MySqr", MySqr);
    p.SetExpr("MySqr(a)");

    for (std::size_t a=0; a<100; ++a)
    {
      fVal = a;  // Change value of variable a
      std::cout << p.Eval() << std::endl;
    }
  }
  catch (Parser::exception_type &e)
  {
    std::cout << e.GetMsg() << std::endl;
  }
  return 0;
}
```

This program computes the squares of the numbers $0, \ldots, 99$ by means of MuParser calls. The functionality of MuParser is the same as mentioned in the last section: We register expressions containing preregistered functions and variables with the call to `SetExpr(string)` (see lines 19 to 21). Variables in MuParser, once registered, automatically change their values if their counterpart in the original code changes (by using pointers). This has the effect that registered functions can be called with different arguments without explicitly calling the function in the original code (see also line 25 and line 26, note that in line 26 no function argument is provided).

In order to keep the MuParser package unaltered for the derivative computation process all function calls and variables of MuParser must be given in `double` data type. For simplicity we will now use the function `MySqr` from above as an example. We can abide by this condition by defining a double function `double dMySqr` which has the purpose of computing the derivative of the function while having only double arguments. In fact, any function registered to MuParser of which derivatives are required can have the following very general form:

```
1  double dMySqr(double a_fVal, double adval)
2  {
3    adtl::adouble tmp;
4    tmp.setValue(a_fVal);
5    tmp.setADValue(0,adval);
6    adtl::adouble ret;
7    ret=ADMySqr(tmp);
8    return ret.getADValue(0);
9  }
```

Inside of this function a temporary `adouble` is created which calls the AD variant `ADMySqr` of the original function `MySqr`:

```
1  adtl::adouble ADMySqr(adtl::adouble a_fVal)
2  {
3    return a_fVal*a_fVal;
4  }
```

Hence, if the seed value `adval` is known to MuParser when calling the function `MySqr` then the derivative of `MySqr` could thus easily be computed. However, notifying the seed to MuParser can be achieved by registering a second variable e.g. `dx` for each variable `x` at its registration.

Now that we *could* easily compute derivatives of functions registered in MuParser

this leaves us with the issue of actually calling the function `dMySqr`. In MuParser this is done by evaluating the expression set in `SetExpr(string)`. Hence, the assumption we have to make is that a second expression containing the analytical derivative of the first expression is provided. This second expression can contain abstract references to derivatives of functions contained in the first expression. For example a legitimate second expression to the first expression `"f*g"` can be the very abstract expression `"df*g + f*dg"` as all the functions `f,g,df,dg` have been previously registered to MuParser in the methodology from above. This assumption is reasonable as many software packages, including any popular computer algebra system or, for example, the freely available package *SymPy* [51], can easily compute these abstract derivatives at runtime as long as the expressions are not too long. Note that these software packages do not need any knowledge of the specifics of the functions `f` or `df`. In fact, as the length of formulas for analytical derivatives increases exponentially the author recommends keeping the expressions provided to these packages as short and simple as possible.

The required changes to the original code are simple and could easily be automated for a larger number of functions.

4.4 Higher Order Differentiation

For the use of gradient-based optimization methods at least the first derivatives of the objective function with respect to the optimization variables are needed. In our case (see next chapter) this means we also need the second order mixed derivatives of impedance with respect to material parameters and geometry variables. However, at the beginning of this work this could not be handled by ADOL-C in the traceless variant (which we are limited to in this thesis, see Appendix A.3). Hence, some extensions to preexisting code [36] was necessary. In this section we will briefly discuss Taylor coefficient arithmetic and the application to computing (mixed) higher order derivatives.

4.4.1 Taylor Coefficient Arithmetic

The following is based upon [21]. One possible way to reduce runtime when computing derivatives compared to full symbolic differentiation is to use the concept of Taylor coefficient arithmetic. We start with Taylor polynomials S and T of order $N \in \mathbb{N}$. Let $x, a \in \mathbb{R}$ as well as $f, g : \mathbb{R} \to \mathbb{R}$ smooth and let $t := (x - a)$, then we choose S, T with

$$S(t) := \sum_{k=0}^{N} \frac{f^{(k)}(a)}{k!} t^k = f(a) + f^{(1)}(a)t + \frac{f^{(2)}(a)}{2!} t^2 + \cdots + \frac{f^{(N)}(a)}{N!} t^N$$

$$T(t) := \sum_{k=0}^{N} \frac{g^{(k)}(a)}{k!} t^k = g(a) + g^{(1)}(a)t + \frac{g^{(2)}(a)}{2!} t^2 + \cdots + \frac{g^{(N)}(a)}{N!} t^N.$$

Following the notation of [21, chapter 13] the n-th coefficient denoted by a subscript

$$S_n := \frac{f^{(n)}(a)}{n!}, \quad 0 \leq n \leq N \tag{4.1}$$

is called a *Taylor coefficient.*

Then for the sum of S and T we have

$$(S + T)(t) = \sum_{k=0}^{N} \frac{f^{(k)}(a) + g^{(k)}(a)}{k!} t^k.$$

We notice that

$$\left. \frac{\partial f}{\partial x} \right|_{x=a} = f'(a) = S_1$$

and, consequently,

$$\left. \frac{\partial (f + g)}{\partial x} \right|_{x=a} = f'(a) + g'(a) = S_1 + T_1.$$

More generally by comparing the coefficients we note for $0 \leq n \leq N$ the following rule for additions

$$\left. \frac{\partial^n (f + g)}{\partial x^n} \right|_{x=a} = n! \, (S_n + T_n)$$

or more compact

$$(S + T)_n = S_n + T_n.$$

For a more interesting example we furthermore consider the product $f \cdot g$ and obtain using the general Leibniz rule

$$\frac{\partial^n (f \cdot g)(x)}{\partial x^n} = \sum_{k=0}^n \binom{n}{k} \frac{\partial^{(n-k)} f}{\partial x^{(n-k)}}(x) \cdot \frac{\partial^k g}{\partial x^k}(x)$$

and thus evaluated at $x = a$ we get the following rule for products

$$\frac{\partial^n (f \cdot g)(a)}{\partial x^n} = \sum_{k=0}^n \frac{n!}{k! \, (n-k)!} \underbrace{\frac{\partial^{(n-k)} f}{\partial x^{(n-k)}}(a)}_{=(n-k)! \cdot S_{n-k}} \cdot \underbrace{\frac{\partial^k g}{\partial x^k}(a)}_{k! \cdot T_k}$$

$$= n! \left(\sum_{k=0}^n S_{n-k} T_k \right).$$

Or, again, more compact

$$(S \cdot T)_n = \sum_{k=0}^n S_{n-k} T_k. \tag{4.2}$$

There are three main points to take notice of:

- By adding more rules for arithmetic operations of Taylor polynomials (see e.g. [21, Table 13.1, Table 13.2]) it is similarly easy to compute higher order derivatives of other arithmetic operations (e.g. `+,-,*,/,log,exp,sin,cos,...`).

- The functions f, g from above are chosen arbitrarily. Hence, they can also be chosen to be compositions of the form $h(x) = g(f(x))$. Thus, any combination of arithmetic operations can be handled in the same manner.

- In order to make use of the rules some initial Taylor coefficients are assumed to be given. These initial coefficients are usually referred to as *seeds*.

In fact, we can make direct use of the seeds for our computation: By defining the Taylor coefficients of the differentiation variable (i.e. the Taylor coefficients of the identity function $f(x) = x$) prior to computation we specify *which* derivatives we choose to compute.

Example 4.1. In this short example we aim to use the method from above to compute the first and second derivatives of the function

$$f(x) = x \cdot \exp(x)$$

for $x = 2.0$. First, we note that the analytical derivatives of this function are given by

$$f^{(n)}(x) = (x + n) \cdot \exp(x).$$

We continue by creating a partition of f into its *elemental operations*

$$z_1(x) := x, \quad z_2 := \exp(x), \quad z_3(x) := z_1(x) \cdot z_2(x) \quad \Rightarrow f(x) = z_3(x).$$

In order to reduce confusion the following notation is introduced for $0 \leq i \leq n = 2$:

- Let A_i denote the Taylor coefficients of x (and thus also z_1),
- let B_i denote the Taylor coefficients of z_2 and
- let C_i denote the Taylor coefficients of z_3.

For our computation we must specify all values A_i for $0 \leq i \leq n$: Let

$$A_i := \begin{cases} 2.0 & \text{for } i = 0 \\ 1.0 & \text{for } i = 1 \\ 0 & \text{else} \end{cases}$$

which can be interpreted as directly setting the derivative $\frac{\partial x}{\partial x} = 1.0$.

The Taylor coefficient arithmetic rule for multiplication is given in Eq. (4.2) and the rule for the application of the exponential function is given by [21]

$$(\exp S)_n := \frac{1}{n} \sum_{k=1}^{n} k \cdot (\exp S)_{n-k} \cdot S_k.$$

Then by direct and successive application of the rules we obtain

$$A = \begin{pmatrix} 2.0 \\ 1 \\ 0 \end{pmatrix} , \; B = \begin{pmatrix} \exp(2.0) \\ \exp(2.0) \\ \frac{1}{2}\exp(2.0) \end{pmatrix} , \; C = \begin{pmatrix} 1.3 \cdot \exp(2.0) \\ (2.0 + 1)\exp(2.0) \\ \frac{1}{2}(2.0 + 2)\exp(2.0) \end{pmatrix} .$$

Finally, we obtain the correct derivatives by remembering that Taylor coefficients are derivatives divided by the factorial of their index (see Eq. (4.1)).

Hence, in a computer program it is possible to construct a data type e.g. `adouble_ho` that, instead of running the usual operations (like `+,-,*,/,log,exp,...`) evaluates the Taylor coefficient arithmetic version of those operations. If this data type is used throughout the code and the user has initially specified the desired seed values the computer program can easily compute the desired derivatives.

Remark 4.2.

Note that one main benefit of this method is a reduced amount of required operations (and thus runtime). This is especially true when the above is used to compute adjoints. However, in this thesis we are limited to the use of only the traceless variant of ADOL-C (and thus only the forward mode, see also section A.3). Hence, faster versions involving adjoints cannot be utilized in this thesis. For more detailed analysis of runtime the author refers to [21, chapter 13].

4.4.2 Higher Order Mixed Derivatives

Using the Taylor coefficient method above it is also possible to compute higher order derivatives with respect to multiple variables. However, setting seed values for this case is not as straightforward as before. A minimal introduction to this topic is given here. However, for a more detailed version the reader may consult [21] or [79].

Let us assume we have two mulitdimensional Taylor polynomials and an analytical function $f : \mathbb{R}^n \to \mathbb{R}^m$ where, for simplicity, $n = 2$, $m = 1$, $d \in \mathbb{N}$ the desired degree

of differentiation and a small $\epsilon > 0$ such that

$$S : \,]-\epsilon, \epsilon[\,\to \mathbb{R}^n, \quad S(t) := \sum_{k=0}^{d} S_k t^k, \quad S_k \in \mathbb{R}^n$$

$$U : \,]-\epsilon, \epsilon[\,\to \mathbb{R}^m, \quad U(t) := \sum_{k=0}^{d} U_k t^k, \quad U_k \in \mathbb{R}^m$$

with the local representation

$$U(t) = f(S(t)).$$

Then we have e.g. $U_0 = U(0) = f(S(0)) = f(S_0)$. Moreover, we get by the chain rule for the first few coefficients of U (see [21, p. 303]):

$$U_0 = f(S_0)$$
$$U_1 = f'(S_0)S_1$$
$$U_2 = f'(S_0)S_2 + \frac{1}{2}f''(S_0)S_1 S_1$$
$$U_3 = f'(S_0)S_3 + f''(S_0)S_1 S_2 + \frac{1}{6}f'''(S_0)S_1 S_1 S_1$$

$$\dots$$

Bear in mind that when using vectorial valued functions this representation is actually a misuse of notation [21, p. 303]. This is useful so high readability can be maintained of now tensorial quantities.

In order to calculate the second derivative of f with respect to two variables, e.g. x_1, x_2, we select U_3 as this is the first term which contains two different coefficients S_1, S_2. Let H_f denote the Hessian of f. Then the second order mixed derivative could be computed e.g. via $\frac{\partial^2 f}{\partial x_1 \partial x_2} = \left(\begin{smallmatrix}1\\0\end{smallmatrix}\right)^T H_f \left(\begin{smallmatrix}0\\1\end{smallmatrix}\right) = S_1^T H_f S_2 = f''(S_0)S_1 S_2$

However, if we compute U_3 by setting $S_1 = \left(\begin{smallmatrix}1\\0\end{smallmatrix}\right), S_2 = \left(\begin{smallmatrix}0\\1\end{smallmatrix}\right)$ and $S_3 = 0$ and store the result in a temporary variable τ_1 then τ_1 additionally contains the term $\frac{1}{6}f'''(S_0)S_1 S_1 S_1$ instead of only $f''(S_0)S_1 S_2$ (as $S_3 = 0$ the first term vanishes).

Thus, the required derivative can be obtained by subtracting $\frac{1}{6}f'''(S_0)S_1 S_1 S_1$ from τ_1. This can be achieved by recomputing U_3 this time setting $S_1 = \left(\begin{smallmatrix}1\\0\end{smallmatrix}\right), S_2 = S_3 = 0$ and storing the result into the temporary variable τ_2. Then, the desired mixed

derivative is the result of $\tau = \tau_1 - \tau_2$.

Remark 4.3. Note that this procedure requires two function evaluations with Taylor polynomials of order 3. However, in e.g. [20] a method that computes second order mixed derivatives by computations of order 2 was suggested (requiring three function evaluations of order 2, see equation (8) in [20, p. 1123]). For this thesis this method is not utilized for several reasons:

First, the second function evaluation on the last page stored in τ_2 also computes $U_1 = f'(S_0)S_1$. This is calculated as a byproduct of the computation of U_3 which indirectly requires the computation of the terms U_2, U_1 and U_0 (analogously to the computation in section 4.4.1). For the computation of the second order derivative of the overall impedance with respect to a material parameter p and a ring radius r the first derivatives $\frac{\partial Z_i}{\partial p}, \frac{\partial Z_i}{\partial r}$ as well as the second order mixed derivatives $\frac{\partial^2 Z_i}{\partial p \partial r}$ for $i = 1, 2, 3$ are required seperatly, see Remark 3.1. Thus, this byproduct can be directly used to obtain $\frac{\partial Z_i}{\partial r}$ while computing $\frac{\partial^2 Z_i}{\partial p \partial r}$.

Second, the current implementation of the method is based on a trace evaluation that is currently not feasible for the sensitivity optimization problem here. Thus, further adaptation of its implementation to allow application to the problems here would be necessary.

As a result it seems questionable that the method of [20] would give a faster overall runtime for all required derivatives in this specific case and that the development time necessary for reimplementation is probably better invested in different aspects of this thesis.

4.4.3 Increasing Accuracy of Very High Derivatives

While implementing and testing a (traceless) higher order derivative variant of ADOL-C the author noticed that higher order derivatives could be computed correctly only up to a specific degree and that the error would rapidly increase with the degree. To explain this behavior we note the following.

To retrieve the i-th derivative of a function f, we need to multiply the i-th coefficient by $i!$

$$S_i := \frac{f^{(i)}(a)}{i!} \Leftrightarrow f^{(i)}(a) = i! \cdot S_i.$$

However, for high values of i the value of $i!$ increases extremely fast. This can lead to numerical accuracy issues.

Example 4.4. Let $f(x) = x \cdot \exp(x)$.
Then

$$f(2) = 2\exp(2) \approx 14.77$$

and with

$$f^{(n)}(x) = (n + x) \cdot \exp(x)$$

we get

$$f^{(170)}(2) = (170 + 2) \cdot \exp(2) \approx 1270.91.$$

Note that these values are within the usual range of `double` variables. However, in order to retrieve the n-th derivative from the n-th Taylor coefficient we need to compute and multiply by $n!$. In this case we have $170! \approx 7.257416 \cdot 10^{306}$ which is barely inside the range of values of a `double` (the maximal value for a usual IEEE 754 64-bit `double` is $\approx 1.8 \cdot 10^{308}$). In order to obtain the correct value of ≈ 1270.91 the corresponding coefficient must be of the magnitude 10^{-303}. The author's faith in the correctness of the result of that computation is quite limited. We conclude that the numerical accuracy of very high order derivatives is severely limited by the multiplication with the factorial of the degree of differentiation.

In order to achieve a smaller error for very high derivatives the usual arithmetic rules used by the AD community were adapted with focus on numerical accuracy. The reader may note that the following considerations are not required for the optimization of sensitivity of piezoelectric ceramics. The following was adapted together with Nora Irene Ghartey who was a summer student at the research group *Mathematics and its Applications* in July 2017. Do note that some of the arithmetic rules are not novel (see e.g. [59, p. 40]). However, the author has no knowledge of any considerations of these rules in the context of increased numerical accuracy of very high derivatives.

In order to eliminate the final multiplication with the factorial (of the degree of the derivative) we try to include this operation stepwise during the evaluation of the Taylor coefficients. Thus, in contrast to the common usage in the AD community

we now directly define

$$S_i := f^{(i)}(a).$$

We now aim to include the missing multiplication stepwise by instead adding the binomial coefficient $\binom{n}{k}$ as an additional factor back into the summation in each Taylor arithmetic rule.

For illustration we continue the example Ex. 4.4 where multiplication and application of the exponential function to Taylor polynomials are considered. For the multiplication of two Taylor polynomials by the general Leibniz rule we directly obtain for an $n \leq d$ with d the degree of the polynomials that

$$(S \cdot T)_n(t) = \sum_{k=0}^{n} \binom{n}{k} S^{(n-k)}(t) T^{(k)}(t)$$

For the exponential function we recursively get

$$e_n := (\exp(T))_n(t) = \sum_{k=1}^{n} \binom{n-1}{k-1} T^{(k)}(t) e_{n-k}(t).$$

Analogously, the binomial coefficient can be integrated into the summation for all other operations.

We can now analyze what improvement we expect: We note that e.g. for an even $n = 2m$ we have $\text{argmax}_{0 \leq k \leq n} \binom{n}{k} = m$. Hence,

$$\max_{0 \leq k \leq n} \binom{n}{k} = \binom{n}{m} = \frac{n!}{m!\,(n-m)!} = \frac{n!}{(m!)^2}.$$

For the ratio we can compute

$$\frac{\binom{n}{m}}{n!} = \frac{1}{(m!)^2} \ll 1.$$

This implies that these intermediate binomial coefficients are very much smaller than $n!$ which is the factor we previously were forced to compute and multiply with.

Example 4.5. For $n = 1000$ we have $\max_{0 \leq k \leq n} \binom{n}{k} \approx 2.70288 \cdot 10^{299}$. Using the adapted formulas we get $f^{(1000)}(2) \approx 7403.83$ which is the correct value of the 1000-th derivative.

Hence, it is now possible to practically compute much higher derivatives within a C++ code (here the maximum is 1028). We can now compute $1028/171 \approx 6.01$ times more derivatives than before. The accuracy is, in any case, still limited by the numeric value of

$$\frac{n!}{\lfloor \frac{n}{2} \rfloor!^2}.$$

Remark 4.6. In [21, p. 302] it is directly noted that "the binomial weights occurring in Leibniz's formula for the derivatives of a product are avoided, and overflow is a little less likely" when the k-th Taylor coefficient is rescaled with the factor $k!$. In the simple examples considered here, however, substantial error was introduced into the Taylor coefficients due to underflow, which also get rescaled by $k!$.

4.5 Definition and Computation of Sensitivity of Impedance

Now that all preliminaries have been dealt with, we can finally go ahead and first define, then compute and finally optimize the sensitivity of impedance with respect to material parameters. In the following sections we will discuss interesting aspects which arose during this process and give the optimization results. Parts of the following sections have been published by the author of this thesis in [29].

As a first step we must define what exactly we mean by the term *sensitivity*. There have been various approaches to this in the context of piezoceramics which we will discuss later on. In this work we will focus on a local sensitivity definition based on derivatives of the measurable quantity.

4.5.1 Definition of Sensitivity

We start by assuming the complex valued impedance Z to be a given function

$$Z : \mathcal{F} \times \mathbf{P} \times \mathcal{R} \to \mathbb{C}$$

depending on frequency $\omega \in \mathcal{F}$, the material parameters $p = [p_1, \ldots, p_{10}] \in \mathbf{P} \subseteq \mathbb{R}^{10}$ as well as the ring radii $r \in \mathcal{R}$, for the set of feasible electrode configurations

$\mathcal{R}$ representing the ring geometry and defining the boundary regions on Ω. This impedance can be computed e.g. by methods derived in section 3.3.

We define the *partial sensitivity* of impedance with respect to a specific material parameter $p_i, i \in \{1, \ldots, 10\}$ as the L^2-norm of the curve

$$\frac{\partial Z(\cdot; p, r)}{\partial p_i} \in \{\mathcal{F} \to \mathbb{C}\} \tag{4.3}$$

and denote

$$\|\nabla_{p_i} Z(\cdot; p, r)\| := \left\|\frac{\partial Z(\cdot; p, r)}{\partial p_i}\right\|_{L^2(\mathcal{F})} = \left(\int_{\mathcal{F}} \left\|\frac{\partial Z(\omega; p, r)}{\partial p_i}\right\|_2^2 d\omega\right)^{\frac{1}{2}}. \tag{4.4}$$

Here, the last integral can be computed by discretizing the frequency domain $\mathcal{F}$ and applying a numerical quadrature scheme (usually Newton-Cotes formulas, e.g. trapezoidal rule, Simpson's rule or polcherima rule, see e.g. [10]). As an example, in Fig. 4.1 the complex valued and frequency dependent partial derivative of the impedance Z with respect to the material parameter ϵ_{11} is displayed.

Because of the form of these curves they have often been refered to as 'bumblebee curves'.
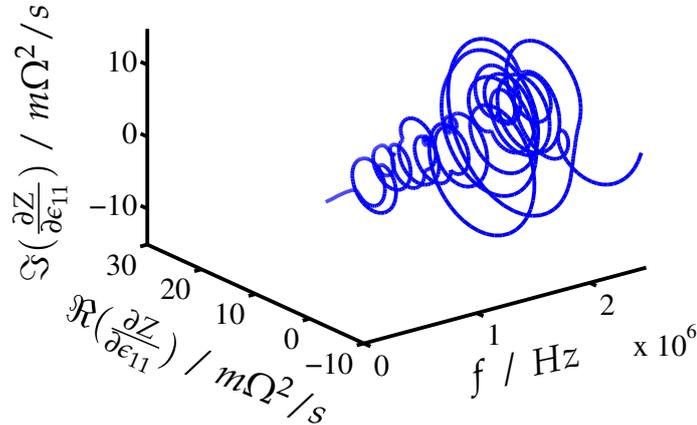

Figure 4.1: Derivative of impedance with respect to the material parameter ϵ_{11}.

At this point it is important to note that these partial sensitivities have very different numerical values for the different parameters p_i. This is partly because some SI units may be much too largely defined for the context they are used in. For example

distances are measured in meters, however the occurring mechanical displacements operate on a much smaller scale. Hence, in order to avoid numerical errors and make the partial sensitivities comparable they need to be scaled to a similar magnitude. This is partly achieved by setting the `adouble` seed values of the material parameters in CFS++ to the magnitude of the specific material parameters, instead of choosing the default value 1. For example using ADOL-C the seed values can be set by using code similar to the following.

```
double eps_11_scale=1e-09;
material_param_eps11.setADValue(&eps_11_scale);
```

This has the benefit that not only have the partial sensitivities received some scaling but also we expect the derivatives to be numerically more stable while they are computed. Alternatively, and mathematically equivalently one could divide the resulting partial sensitivities by the magnitude of the specific material parameters and scale them in this way if Algorithmic Differentiation is not applied. Do note that this may have an impact on numerical accuracy.

Note that even by scaling with the magnitude of the specific material parameter numerical values the partial sensitivities still have very different magnitudes. The partial sensitivities for the case of an optimized triple-ring setup are shown in Fig. 4.2.
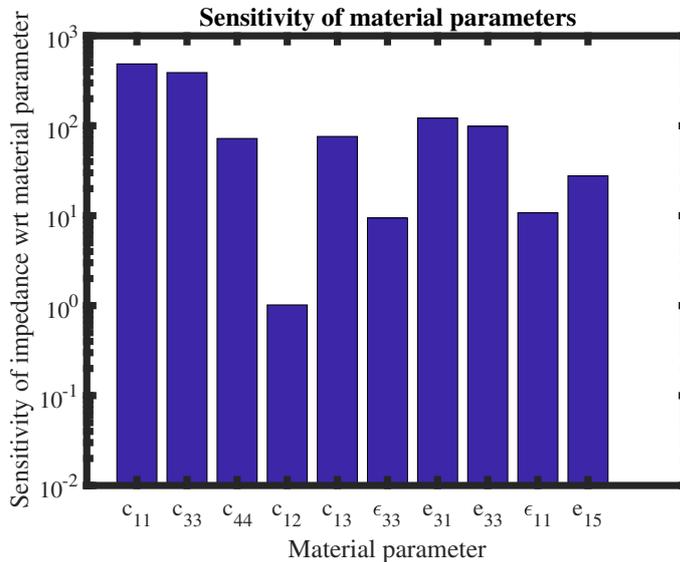

Figure 4.2: Pre-scaled partial sensitivity for different material parameters.

For the identification of all material parameters as part of an inverse problem all

sensitivities (of the overall impedance) would ideally globally have a sensitivity of approximately 1.0. However, this does not seem to be possible using this specific type of triple-ring setup, at least when using the physically motivated definition of impedance. One reason for this behavior is that the sensitivities also depend on the overall geometry of the ceramic itself and not only on the electrode geometry placed onto the ceramic. The overall geometry of the ceramic is fixed and cannot be modified as it would usually be provided by the manufacturer. Hence, in this work it is the goal to make best use of the sensitivities that can be gained by any given ceramic geometry. Motivated by this fact we furthermore introduce a weight w_i for each material parameter p_i. These weights aim to make sure that those material parameters, which have a low sensitivity associated with, are prioritized in the optimization over those material parameters that have a high sensitivity. These weights could then easily be adapted for a given ceramic geometry or also be specifically adapted to ignore material parameters which are not of interest for a specific case.

Hence, the *overall sensitivity* is defined by the (weighted) Euclidean norm

$$J_1(r) := \left\| WS \left[\| \nabla_{p_i} Z(\cdot; p, r) \| \right]_{i=1,\dots,N_p} \right\|_2$$

with $\left[\| \nabla_{p_i} Z(\cdot; p, r) \| \right]_{i=1,\dots,N_p} \in \mathbb{R}^{N_p}$ the vector containing all partial sensitivities, the diagonal weight matrix $W \in \mathbb{R}^{N_p \times N_p}$ which in the easiest case is the unit matrix and a diagonal scaling matrix $S \in \mathbb{R}^{N_p \times N_p}$ which scales the very different contributions of each parameter to a common level. These matrices are stated here separately to emphasize the importance of scaling in this context. However, they can be unified in a single diagonal matrix instead. For future reference we also note $J_1(r)$ as the objective function for optimization which we seek to maximize with $r \in \mathcal{R}$ the set of feasible electrode configurations.

4.5.2 Comparison to Alternative Sensitivity Definitions

Some alternative definitions have also been suggested in this context:

First, do note the strong similarity between the definition given here and classical methods in the context of optimal experiment design. Especially note that Eq. (4.4)

is a different interpretation of the Fisher information matrix (see e.g. [58]). One distinction from classical methods is that for the Fisher information matrix the frequency points are permitted to be chosen arbitrarily. In light of the L^2-norm defined here, however, it makes sense to choose these in the context of error theory for numerical quadrature schemes. As the definitions are slightly different e.g. in the operation order but also in other aspects the method suggested here could be interpreted only similar to a classical optimal design method. For future research it may be worthwhile to unify both definitions in order to benefit from the rich research results on classical optimal experiment design. This includes a more statistical interpretation of the sensitivities incorporating noise distribution assumptions which is not explicitly handled in this thesis.

Second, there is the concept of global sensitivity. The sensitivity defined here is based on a local interpretation of sensitivity. This means that increasing the sensitivity only does so for this specific set of material parameters. An optimization does not necessarily cover all other values for the material parameters but one hopes that it also simultaneously increases for all other values of the material parameters (which is in general true only locally). In [39] a method is suggested to evaluate the local sensitivities for many different values of the material parameters. In light of the inverse problem where the material parameters are the unknowns it seems a very good idea to also account for changing material parameter values. However, note that this comes at the cost of a huge amount of function evaluations and thus huge computational cost for each evaluation of the global sensitivity. For this work, where each sensitivity evaluation can run from a couple of seconds to multiple hours (depending on the refinement of the computational grid, see [29]) this idea seems good but not practical in the context of this thesis.

Third, there are methods motivated by other regions of research. For example in the Ph.D. thesis of Unverzagt [75] a cross-correlation method is used to define a quantitative sensitivity measure. This method seems to be partially motivated by an approach from signal theory. In this thesis a comparison to this method is not given. However, the optimized electrode geometry of the piezoceramic developed in [75] is used as an initial point to some of the optimization presented later on.

4.6 Accuracy of Derivatives

Parts of this section have been published in [29], especially section 4 of that article. The accuracy of the derivatives is of central importance for the optimization process: the cost function nearly entirely consists of derivatives that need to be calculated. We have validated the derivatives of CFS++ produced by ADOL-C by comparing them directly with derivatives produced by applying a finite difference (FD) scheme. This shows good accuracy. The post-processing steps (i.e. network computations and I/O, see Fig. 5.3) written in MATLAB[TM] [48] have been differentiated symbolically. An independent comparison with finite differences also shows good accuracy.

One major problem in the application of finite differences is choosing a step width. Numerical error theory of finite differences (see e.g. [53, section 9.1] or [21]) states that step widths too large generally provide a poor approximation. Conversely, an approximation with a step width too small accumulates large round-off errors and also provides a poor approximation. The optimal step width is not known a-priori (without a reference value this is also true a-posteriori) and depends on the evaluation point. However, it is also fairly simple to construct functions (e.g. $f(x) = \sin(1/x)$ for $x > 0$ small or $f(x) = x\sin(x)$ for large x) where for some points x no step width h can be identified where the error is sufficiently small (see e.g. magenta coloured line in Fig. 4.3).

For our application it is, thus, also important to note that the optimal step width depends on the concrete electrode configuration used.

Let

$$\nabla_{p_i}^{ad} z(\omega_{discr}; p, r) \approx \frac{\partial Z(\omega_{discr}; p, r)}{\partial p_i}$$

be the partial sensitivity computed using Algorithmic Differentiation at discrete frequency points ω_{discr}. This is an approximation as FEM simulation tools only compute an approximate solution to a partial differential equation. Analogously, for a given small $h > 0$ the approximation to the partial sensitivity gained by applying (first order) finite differences to the CFS++ simulation is defined as

$$\nabla_{p_{i,h}}^{fd} z(\omega_{discr}; p, r) := \frac{Z(\omega_{discr}; p_{i,h}, r) - Z(\omega_{discr}; p, r)}{h} \approx \frac{\partial Z(\omega_{discr}; p, r)}{\partial p_i}$$
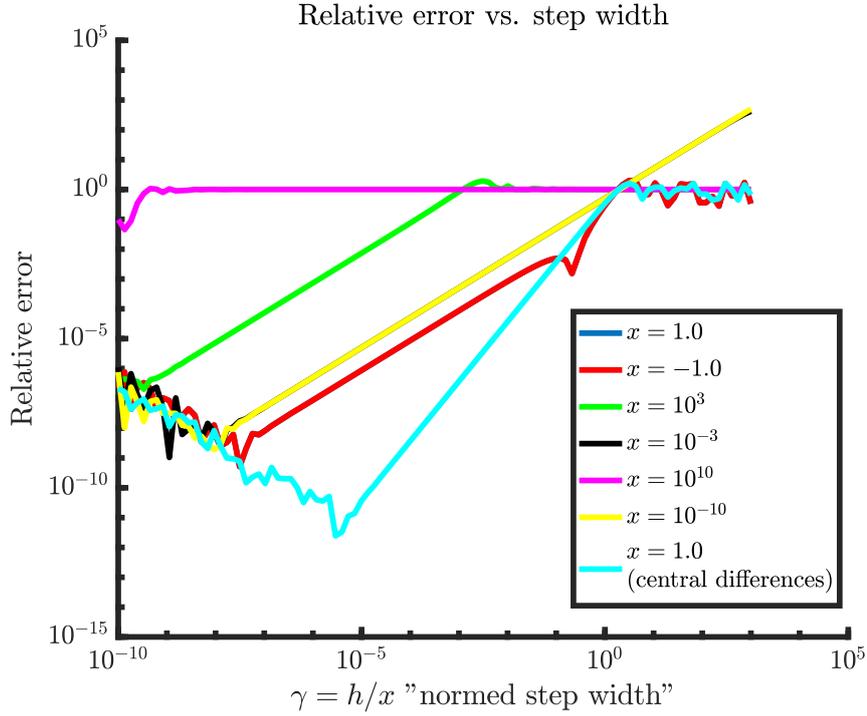
Figure 4.3: Relative error of finite differences using different step widths h and different points of evaluation x for the function $f(x) = x \cdot \sin x$.

with $p_{i,h} := [p_1, \ldots, p_i + h, \ldots, p_{10}]$.

In Fig. 4.4 we compare the relative difference[1]

$$\text{rel\_diff}(h, x) := \frac{\nabla^{fd}_{p_{i,h}} z(x) - \nabla^{ad}_{p_i} z(x)}{\nabla^{ad}_{p_i} z(x)}$$

of sensitivity for four different electrode configurations x calculated by AD and FD for a single material parameter by varying the step width. We show this for different feasible electrode configurations $x \in \mathcal{R}$. First, we focus on the blue and green lines indicating the relative differences of AD and FD with respect to i) a default electrode configuration (reference configuration we also physically possess for measurement purposes, 'x' markers, blue line) and ii) one where we set the first radii r_1 to 3.3 mm (square markers, green line). In the first case choosing a normed step width size of $\gamma = \frac{x+h}{x} - 1 = \frac{h}{x} = 5.995 \cdot 10^{-5}$ gives a small relative difference

---

[1]As AD methods usually yield derivatives with working accuracy [21] we may carefully refer to the relative differences as relative errors.
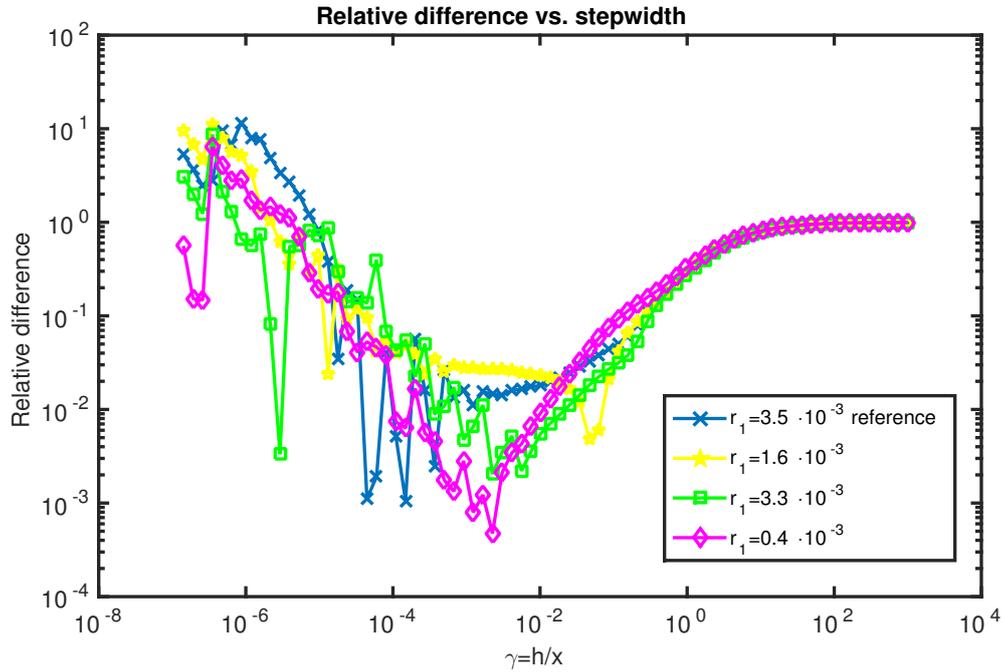
Figure 4.4: Relative differences ('error') of finite differences vs. Algorithmic Differentiation over step width.

of $0.1916\,\%$. In the second case this situation is now very different. The relative difference is now $39.76\,\%$.

Similar discrepancies can also be found for the other configurations given ('*' markers, yellow line, $r_1 = 1.6\,\mathrm{mm}$; diamond markers, magenta line, $r_1 = 0.4\,\mathrm{mm}$). As each of the datapoints shown takes approx. one hour to compute (cf. [29, Tab. 4]) we do not show more configurations. However, it is our experience that it is very challenging to find an appropriate differentiation step width for one specific configuration and even more so to identify a step width sufficient for any arbitrary electrode configuration provided by an optimizer.

This is very troubling for the optimization of sensitivity with the electrode configuration as optimization variables via finite differences: The relative differences are of the same magnitude as some of the achieved gains by optimization in [37]. Even more: the goal of the optimization is to increase overall sensitivity. This may also be interpreted as increasing a measure of nonlinearity. In this case we expect finite differences generally to yield increasingly poorer approximations.

In the second plot Fig. 4.5 we see the effect finite differences have on the cost function for a possible optimization. For Fig. 4.5 we fix the step width and now plot the (unscaled) sensitivity of impedance with respect to ϵ_{11} computed either by AD (blue line) or by finite differences (red line) on the y-axis. For the x-axis we vary the first radii of the configuration from $0.15\,\text{mm}$ to $3.8\,\text{mm}$. We note that the blue line (AD) is very smooth and the red line (FD) has many zig-zag features, however, they seem to describe some common function. While it is possible to use the red curve (FD) for optimization purposes due to the zig-zagging we expect optimizers to only find very locally optimal points. On the other side with the blue line (AD), we expect an optimizer to not have too many difficulties to even find a globally optimal point. We tested this hypothesis on this small example with only one variable radius and found it to be precise. In the next section we will see that this also has a significant effect on the optimization.
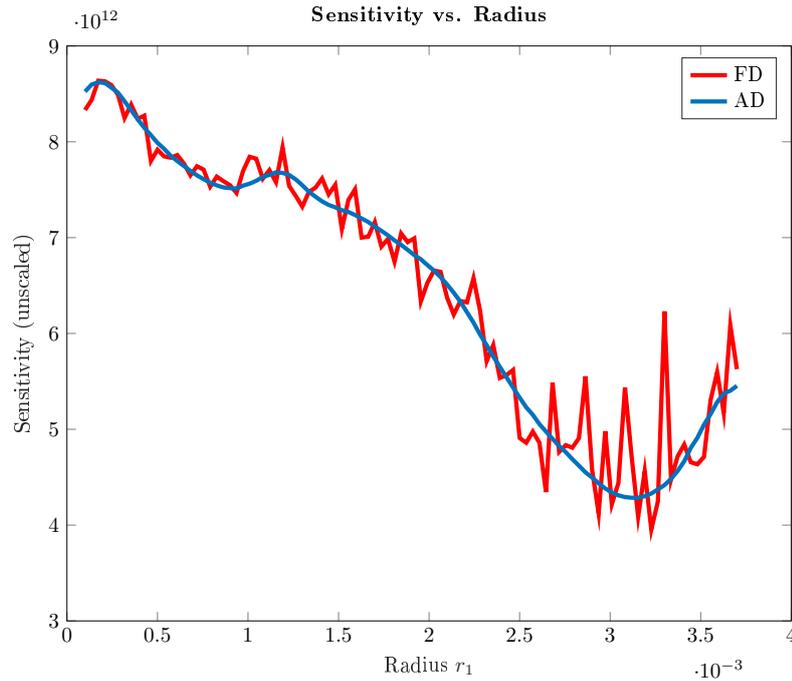

Figure 4.5: Sensitivity of impedance with respect to ϵ_{11} for the first radii r_1 between minimally and maximally feasible values.

Naturally, the author also experimented with a higher order finite differencing scheme to increase the accuracy. As each FEM solution is runtime intensive this

comes at the cost of an overall much higher runtime. Nevertheless, one still faces the issue of choosing a differencing step width h suitable not only for one specific electrode configuration but all electrode configurations so we have low confidence in the quality of the results computed this way. Practical experiments were conducted in [37] to test this and the author was unable to find any substantial improvement on the behavior of the optimizer and the resulting solution.

As a consequence the author has even less confidence in mixed second order derivatives $\frac{\partial^2 Z}{\partial p \partial r}$ computed solely by finite differences which would be required for gradient based optimization.

Remark 4.7. The author did verify correctness of second order mixed derivatives computed by the higher order traceless variant which was adapted for this thesis (discussed in section 4.4). The author found that the results of the previous analysis presented in this section are analogously valid for second order derivatives. Thus, the verification results are omitted for brevity.

5

Optimization of Sensitivity

5.1 Optimization Problem and General Procedure

First, we need to establish the optimization problem itself. As mentioned earlier, the objective function we aim to increase is

$$J_1(r) := \left\| W S \left[\left\| \nabla_{p_i} Z(\cdot; p, r) \right\| \right]_{i=1,\ldots,N_p} \right\|_2$$

where we are permitted to modify the ring radii parametrized by r. The parametrization used for the later parts of this thesis on is shown in Fig. 5.1. However, the choice
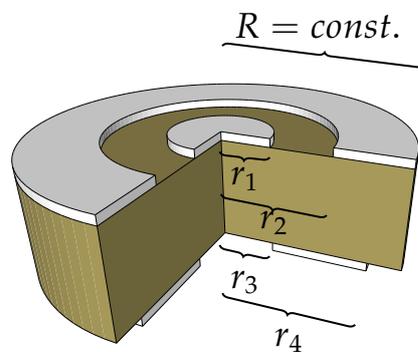
Figure 5.1: Parametrization of triple-ring electrodes.

of parametrization leads to some restrictions on what kind of triple-ring electrode can be described by the specific parametrization. For instance, using this specific parametrization the top outer electrode always ends at the outer perimeter of the piezoceramic - it is not possible to have a gap between the outer perimeter of the ceramic and the electrode. Though in principle it would not be hard to add an

extra variable that parametrizes this gap, however, the extra variable would significantly increase the runtime of the optimization procedure. Hence, the choice of parametrization is a trade-off between the importance of a general description of all possible triple-ring electrodes and the amount of parameters needed to do so.

Furthermore, a very important aspect in the choice of parametrization is also the question of scalability of parameters - in other words: Can we prevent choosing a parametrization where large changes of the parameters have a small impact on the objective function near the optimum? It is common for many algorithms to not gracefully handle parameters that are nonlinear and badly scaled in combination. An example of this behavior is the Rosenbrock-function (see also remarks in e.g. [73, p. 27-28], [53, p. 26-28]) which is used as a popular benchmark for many optimization algorithms as it incorporates a badly scaled region near the optimum (large changes in some of the optimization variables have only a small impact on the objective function).

In a first attempt of parametrization we did not consider this scalability issue and subsequently did not gain any satisfying results from the optimization. Results are shown in section 5.3.1.

In summary, the aspects of practical runtime, generalization and scalability went into consideration for the choice of the parametrization developed in this thesis. This leads to a parametrization using four variables $r = (r_1, r_2, r_3, r_4) \in \mathbb{R}^4$ as shown in Fig. 5.1. This selection was developed exclusively for this thesis by the author and was not used in other work e.g. [74].

Now that a parametrization has been chosen we must now establish which configurations r are deemed feasible: First, for obvious physical reasons the ring radii cannot be negative. Furthermore, in the resulting configuration the electrodes may not overlap - there must always be a gap otherwise we do not have a triple-ring setup. For physically building these configurations we furthermore face some technical restrictions: A laser with a fixed width is used to create the gaps. Hence, the gap also must have a minimal width of the same size as the laser. Second, for physically affixing the electrodes in an experimental measurement setup similar to [75] the electrode rings must have a minimal width.

Thus, we can describe the feasible triple-ring electrode setup parametrized as in

Fig. 5.1 by the set

$$\mathcal{R} := \{r = (r_1, r_2, r_3, r_4) \in \mathbb{R}^4 : Ar \leq b\}$$

with

$$A := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix} \in \mathbb{R}^{10 \times 4}, \quad b := \begin{bmatrix} 3.7 \cdot 10^{-3} \\ 4 \cdot 10^{-3} \\ 5 \cdot 10^{-3} \\ 5 \cdot 10^{-3} \\ -0.5 \cdot 10^{-3} \\ 0 \\ -0.15 \cdot 10^{-3} \\ 0 \\ -0.3 \cdot 10^{-3} \\ -1 \cdot 10^{-3} \end{bmatrix} \in \mathbb{R}^{10}.$$

This gives us 10 linear constraints for 4 variables. Each variable r_i is bounded from below and above by a box constraint and furthermore r_1 and r_2 as well as r_3 and r_4 share a linear constraint, respectively, assuring that there is no overlap for the top electrodes and a minimal width for the bottom electrode. Because of the mild linear constraints coupling only two variables in each constraint we can easily show a graphical representation of the 4 dimensional feasible domain $\mathcal{R}$ (see Fig. 5.2). Here, each small blue line indicates a single constraint and the interior of the two green triangles indicates the feasible domain. Finally we can state the full optimization problem:

$$\min_{r \in \mathbb{R}^4} -J_1(r) \quad \text{s.t.} \quad Ar \leq b \tag{5.1}$$

The optimization of sensitivity is not straightforward from an implementation point of view due to many file I/O operations and overall long runtime of simulations. In fact, much time was dedicated to set up an environment in which sensitivity optimization problems can be handled as efficiently as possible. In Fig. 5.3 a flowchart of the general procedure is displayed. More information and details e.g. concerning runtime and memory consumption but also more technical details on the implementation of the scheme as displayed in Fig. 5.3 are given in [29].
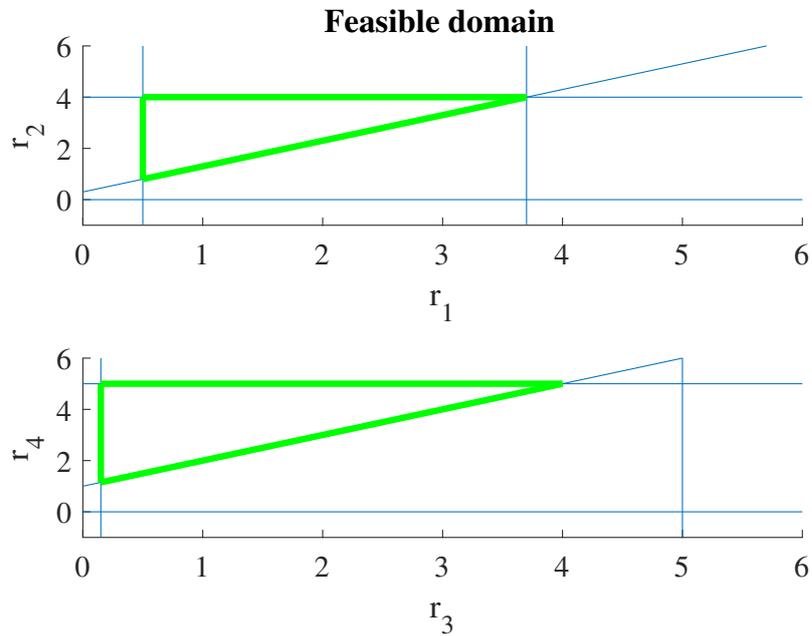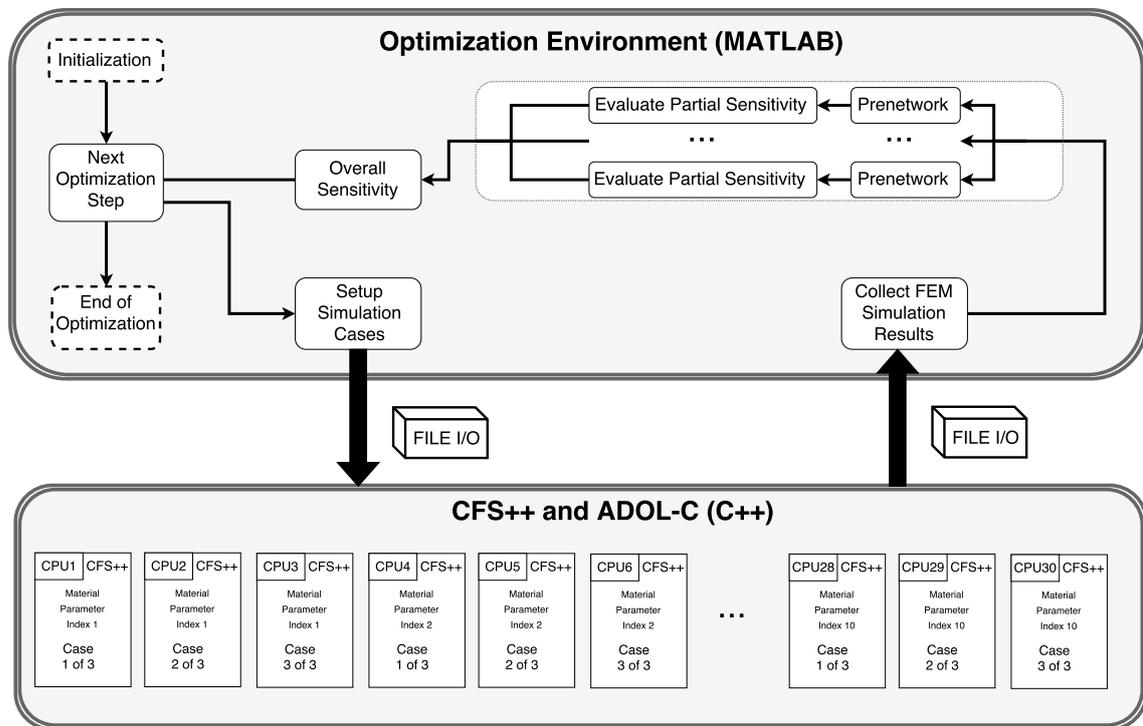
Figure 5.2: Feasible domain for ring radii.

Figure 5.3: Flowchart of optimization procedure.

5.2 Computational Mesh Settings

Before we can go into the details of the optimization methods used and results gained we need to take a moment to affirm that the computational mesh is adequate for simulation and optimization. This is usually achieved by demonstrating that the simulation results converge for decreasing mesh granularity. As the start and end points of the electrodes move for different ring electrode configurations, regeneration of the mesh (with negligible runtime) is important especially when computing derivatives with respect to the ring radii. Thus, it makes sense to also verify convergence for different electrode settings. In Fig. 5.4 this behavior is shown to be true for various ring electrode settings. Furthermore, it can also be observed that the sensitivity computed on meshes that are too coarse do not seem to be smooth and thus unsuitable for optimization (e.g. mesh size 0.07 mm). As increasing the number of vertices in the mesh drastically increases the runtime of the computations a trade-off between runtime and accuracy must be made. For results presented in section 5.3.2.1 a mesh with element size 0.01 mm is used. In order to thoroughly test the results obtained by `IPOPT` by testing more initial electrode configurations than before (shown in section 5.3.2.2) a coarser mesh with mesh element size 0.03 mm was selected as a trade-off with full knowledge that a coarser mesh could affect accuracy. Runtimes resulting from the mesh selection are given in [29].

Mesh element sizes, typical values for the number of vertices and elements (which vary slightly for different electrode geometries) for mesh element sizes between 0.06 mm and 0.01 mm are given in Tab. 5.1. In Fig. 5.5 the meshes used for this

Mesh element size in mm	0.06	0.05	0.04	0.03	0.02	0.01
Vertices	1729	2494	3958	7230	16008	64428
Elements	1832	2618	4112	8356	16315	65040

Table 5.1: Mesh information.

thesis are displayed. Note that due to the radial symmetry of the piezoceramic a simple rectangular geometry in cylindrical coordinates can be chosen as the domain.
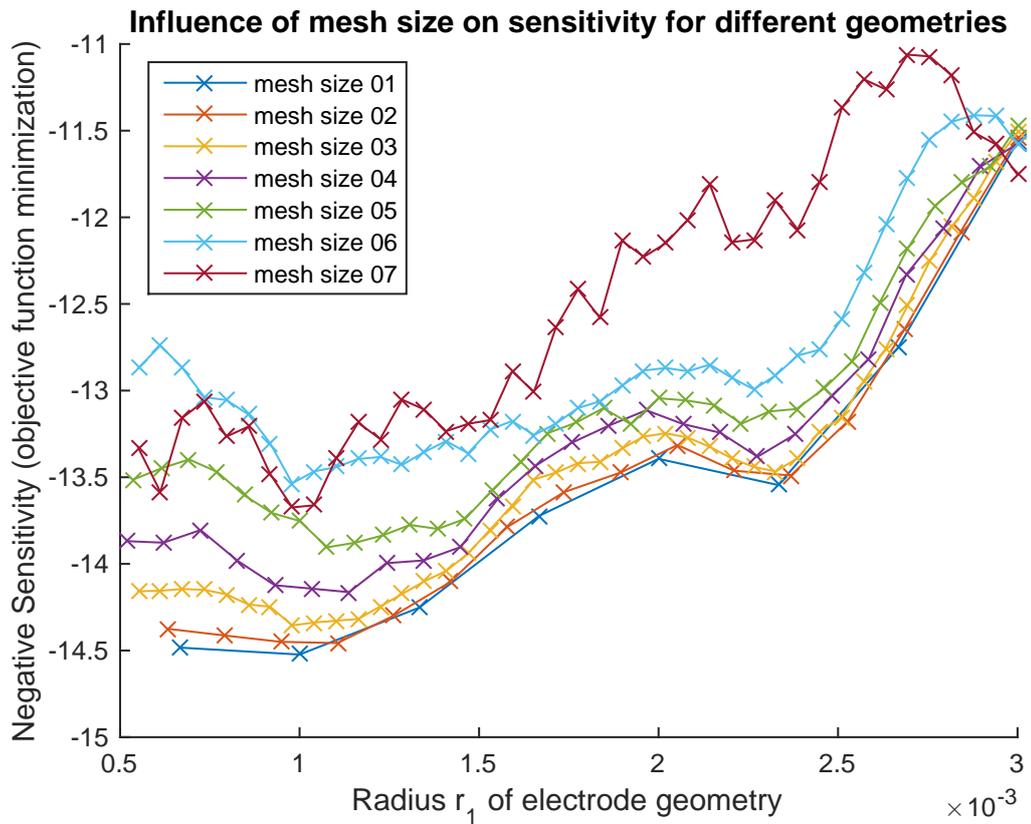
Figure 5.4: Mesh dependency of sensitivity evaluated for different electrode configurations.

(a) Mesh size 0.06 mm

(b) Mesh size 0.05 mm

(c) Mesh size 0.04 mm

(d) Mesh size 0.03 mm

(e) Mesh size 0.02 mm

(f) Mesh size 0.01 mm

(g) Mesh of full domain, mesh size 0.06 mm

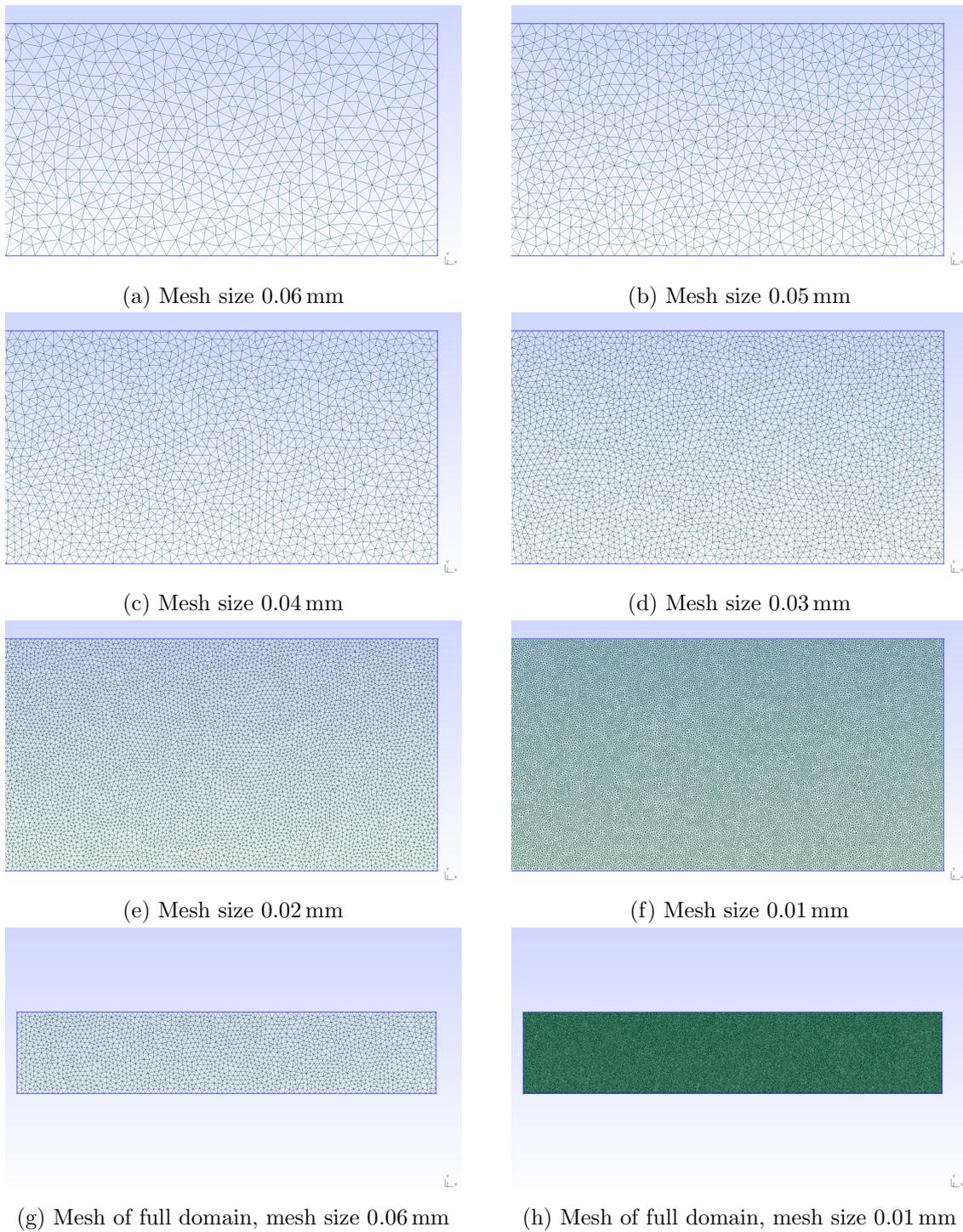(h) Mesh of full domain, mesh size 0.01 mm

Figure 5.5: Meshes used in this thesis.

5.3 Optimization Methods and Results

Practically solving the optimization problem (5.1) proved to be a very challenging undertaking and work leading to a solution procedure is one of the major drivers of this thesis. There have been two major approaches in this thesis to solve this problem: derivative free optimization and gradient based optimization.

5.3.1 Derivative Free Optimization

A first attempt to solve this problem was made as part of [37]. Initially the runtime for each objective function evaluation was approximately 30 hours. Hence, any global optimization strategy based on cheap function evaluations (such as evolutionary algorithms like genetic optimization, e.g. [27], or machine learning techniques) is infeasible due to the expected high runtime. Furthermore, in the setting of [37] any gradient-based optimization strategy was also infeasible due to the fact that in [37] only finite differences were available in order to compute the sensitivities. Gradient based optimization would have required second order mixed derivatives computed by finite differences which cannot be considered reliable for this work, as demonstrated in the last chapter. Hence, we experimented with various gradient-free optimization algorithms.

In [37] the simulations were carried out using a closed source commercial simulation tool and not CFS++. Note that due to licensing restrictions with the simulation software and computation server the author is unable to recompute the results in this exact setting and thus only limited data on the optimization results can be provided here.

First, as a standard derivative-free approach the well-known Nelder-Mead method [26, p. 453] was applied to the problem. However, as the Nelder-Mead method does not allow for constraints the constrained optimization problem Eq. (5.1) was reformulated into a slightly different unconstrained optimization problem:

$$\min_{[x_1, x_2, x_3, x_4] \in \mathbb{R}^4} -J_1(r(x))$$

where

$$r(x) := [r_1, r_2, r_3, r_4]$$

and

$$r_2 := \left(0.5 + \frac{\arctan(x_2)}{\pi} \right) \cdot (R - \epsilon_1 - \epsilon_2) + \epsilon_1 + \epsilon_2$$

$$r_4 := \left(0.5 + \frac{\arctan(x_4)}{\pi} \right) \cdot (R - \epsilon_1 - \epsilon_2) + \epsilon_1 + \epsilon_2$$

$$r_1 := \left(0.5 + \frac{\arctan(x_1)}{\pi} \right) \cdot (r_2 - \epsilon_1 - \epsilon_2) + \epsilon_1 + \epsilon_2$$

$$r_3 := \left(0.5 + \frac{\arctan(x_3)}{\pi} \right) \cdot (r_4 - \epsilon_1 - \epsilon_2) + \epsilon_1 + \epsilon_2$$

The constant R denotes the radius of the entire ceramic and the constants ϵ_1, ϵ_2 guarantee a minimal gap between the electrodes. It is important to observe that this reparametrization always yields a feasible electrode configuration. This is of special relevance as the simulation tool cannot handle any unphysical electrode configurations. Note that this parametrization does not scale well near the boundary of feasibility as $\arctan(x)$ converges only very slowly to $\pm\frac{\pi}{2}$ for $x \to \pm\infty$. Ultimately, the results of the optimization using the Nelder-Mead method were not satisfying.

As a next approach it was decided to investigate other well known derivative-free optimization methods that can handle constraints. For this purpose the well known optimization routines of M. J. D. Powell (available at [56]) were chosen.

From the suite of the optimizers provided there, the optimization methods NEWUOA (unconstrained optimization), BOBYQA (box constraints) and LINCOA (linear constraints), all written in FORTRAN, were chosen for further investigation. These optimizers were first tested on benchmark problems and the satisfying results concerning the benchmark problems also observed in [8] are confirmed here. After these promising results it was decided to further investigate LINCOA [57] for the purposes of [37]. This comes with the benefit that the newer ring parametrization as depicted in Fig. 5.1 can be applied.

In Tab. 5.2 some of the optimization results obtained in [37] are given. The objective function was configured with a uniform weight matrix $W = diag(1, \ldots, 1)$ and the initial electrode configurations are given for the case in (a): $r = [0.3, 0.6, 4.4, 4.7]$, for the case in (b): $r = [2.0, 2.3, 0.3, 4.7]$ and finally for the case in (c): $r =$

$[0.3, 4.68, 2.05, 2.35]$. Note that a slightly different objective function and constraints are used in [37]. Hence, the achieved objective function values are not given here.

Param.	Gain ratio	SROR*	Param.	Gain ratio	SROR*
c_{11}	1.2434	0.0849	c_{11}	0.9293	0.2234
c_{33}	1.4205	0.1292	c_{33}	0.8531	0.3203
c_{44}	5.3202	0.3028	c_{44}	1.1749	1.2426
c_{12}	1.0125	0.2835	c_{12}	0.5684	1.1232
c_{13}	1.0978	0.0855	c_{13}	0.8933	0.2433
ϵ_{33}	1.3629	0.3185	ϵ_{33}	0.6659	0.7925
e_{31}	1.1556	0.0947	e_{31}	1.0677	0.1759
e_{33}	1.8522	0.1520	e_{33}	0.9212	0.2585
ϵ_{11}	2.8385	0.2002	ϵ_{11}	1.2864	0.5104
e_{15}	3.2481	0.3084	e_{15}	1.4586	0.9625

(a) Initial configuration:
$r = (0.3,\ 0.6,\ 4.4,\ 4.7)$mm

(b) Initial configuration:
$r = (2.0,\ 2.3,\ 0.3,\ 4.7)$mm

Param.	Gain ratio	SROR*
c_{11}	1.1830	0.0068
c_{33}	2.3508	0.0271
c_{44}	4.3539	0.2084
c_{12}	2.6384	0.0748
c_{13}	1.3414	0.0073
ϵ_{33}	2.5290	0.0877
e_{31}	1.2714	0.0109
e_{33}	2.4383	0.0705
ϵ_{11}	2.0128	0.1069
e_{15}	1.8070	0.0977

(c) Initial configuration:
$r = (0.3,\ 4.68,\ 2.05,\ 2.35)$mm

Table 5.2: Optimization results using derivative free method LINCOA. The abbreviation *SROR* stands for *sensitivity ratio to optimized reference*. More details available in [37].

It can be seen that the partial sensitivities are significantly increased by the optimization (up to a factor of 5.3). However, the sensitivity ratio to optimized reference (SROR*) indicates that the overall sensitivity results after optimization are very low compared to a reference configuration ($r = [3.5, 3.8, 2.05, 3.55]\,$mm) developed in [75]. Note that the resulting optimized electrode configurations vary in each optimization case presented in [37]. This indicates that there are many only locally optimal configurations. This observation is consistent to results of section 4.6.

5.3.2 Gradient Based Optimization

Motivated by the unsatisfying optimization results in [37] Algorithmic Differentiation was implemented into the simulation code CFS++. By comparing the sensitivities (see Fig. 4.5) computed either via finite differences or Algorithmic Differentiation it is noted that the many locally optimal points can be explained by the lack of accuracy of finite differences schemes for this specific function. Furthermore, we also note that the objective function computed via Algorithmic Differentiation seems much smoother and thus much easier to optimize.

This conclusion leads to reconsider gradient-based optimization methods for this optimization problem. When applicable, gradient-based methods are preferable to derivative-free methods for many optimization problems. In general they usually have a pre-existing convergence theory with high convergence rates and thus a lower expected runtime in comparison to many derivative-free optimization methods.

In the following, the results of the two gradient based optimizers `fmincon` and `IPOPT` are presented in section 5.3.2.1 and section 5.3.2.2, respectively. In order to independently verify the findings of section 4.6 in the case of `fmincon` (section 5.3.2.1) results where both the objective function and the gradient of the objective function are computed using the best possible settings for finite differences known to the author for this case are given. In contrast, in this case we will also give results where the objective function is computed using Algorithmic Differentiation. This will show much improved results. In any case the gradient of the objective function is computed using finite differences.

In the case of `IPOPT` (section 5.3.2.2) the objective function is computed using Algorithmic Differentiation and the gradient of the objective function is computed either by finite differences or by Algorithmic Differentiation (second order mixed derivatives).

Two test cases are considered: First, the optimization of sensitivity with respect to only ϵ_{11}. Thus the weight matrix only has a single nonzero entry for ϵ_{11}. This test case is interesting as ϵ_{11} typically is a critical parameter with low sensitivity. Second, the optimization of sensitivity with respect to all material parameters (the weight matrix is now the unit matrix). An overview of the different settings and optimization cases is given in Tab. 5.3 and Tab. 5.4.

Method	FD	AD
Objective function method	✓*	✓
Gradient	✓	✗
General test cases	2	
Different starting points for single parameter test case	2	
Different starting points for all parameter test case	1	
Mesh element size	0.01 mm	
Runtime per function evaluation	approx. 4 hours	

Table 5.3: Overview of optimization tests and settings for `fmincon` (section 5.3.2.1). *FD for objective function only in single parameter case.*

| Method	FD	AD
Objective function method	✗	✓
Gradient	✓	✓
General test cases	2	
Different starting points for single parameter test case	10	
Different starting points for all parameter test case	10	
Mesh element size	0.03 mm	
Runtime per function evaluation	approx. 15 minutes	

Table 5.4: Overview of optimization tests and settings for `IPOPT` (section 5.3.2.2).

5.3.2.1 Optimization Results Gained by `fmincon`

The optimization method `fmincon` is an optimizer provided inside the software tool MATLAB$^{\text{TM}}$. It can handle constrained optimization. The objective function is computed at first using finite differences and in a second run using Algorithmic Differentiation. The gradient of the objective function is computed by finite differences. Parts of this subsection have been published in [29].

**Optimization ith Respect to the Single Parameter ϵ_{11}**

In the following figures we show the maximization of sensitivity of impedance with respect to ϵ_{11}. We will show two optimization cases and compare the optima computed for both cases either with the help of FD or AD, respectively. In the first case we will start from the reference configuration $[3.5, 3.8, 2.05, 3.55]$ mm, in the second case we will start from a different 'custom' configuration $[0.7, 3, 0.5, 4]$ mm chosen randomly as a configuration in the interior of the feasible domain. In Fig. 5.6 the
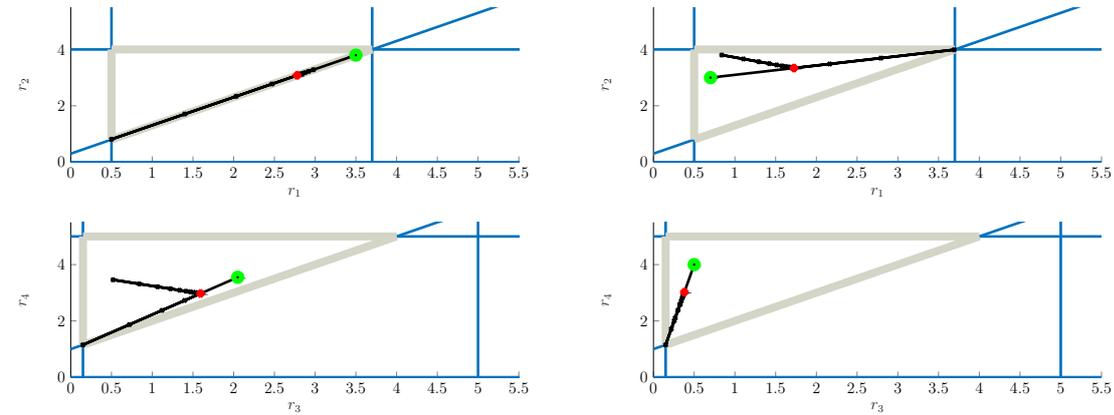
(a) Progress of electrode radii evaluation in optimization using the default reference electrode configuration.

(b) Progress of electrode radii evaluation in optimization using a changed electrode configuration.
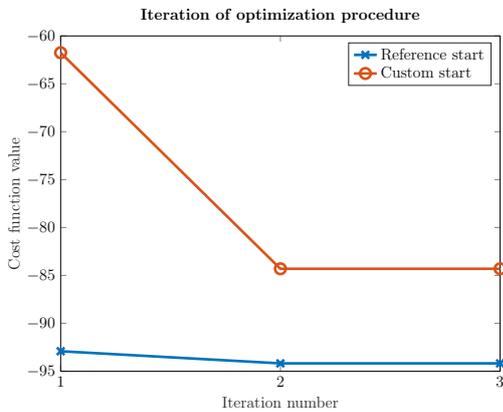
(c) Iteration of optimization procedure.
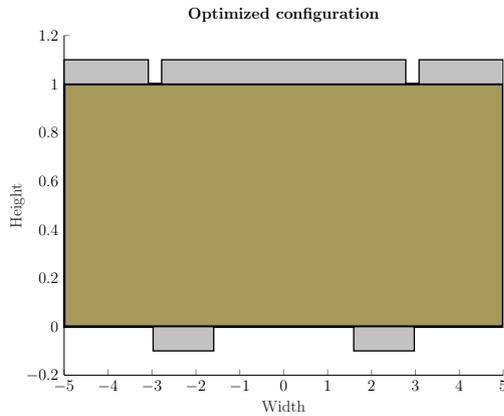
(d) Best optimized configuration $r := [2.78, 3.082, 1.594, 2.974]$ mm found for the two FD-based optimization cases presented.

Figure 5.6: Optimization via FD using two different starting points.

optimization using finite differences is shown. There, small black '+' symbols denote configurations that were evaluated, a large circular green and a smaller 'x' red marker indicate the start point and endpoint of the optimization, respectively. We provide the best possible conditions for this optimization: We use a fine mesh with element size 0.01 mm and also the best possible step width known to us $\gamma = 10^{-3}$ based on Fig. 4.4 and experience we made with the many optimization cases we have already run. In Fig. 5.6a we start the optimization with the reference configuration and an initial cost function value of -92.92. The optimization halts with

the final configuration $[2.78, 3.082, 1.594, 2.974]$ mm and the optimized cost function value of -94.19. This gives an increase of sensitivity by a factor of 1.0137. In Fig. 5.6b we start the optimization with the 'custom' configuration and an initial cost function value of -61.76. The optimization halts with the final configuration $[1.716, 3.344, 0.378, 3.009]$ mm and the optimized cost function value of -84.31. This gives an increase of sensitivity by a factor of 1.37, however, with a lower sensitivity than our already known reference configuration. A representation of the electrode configuration optimized using finite differences can be found in Figure 5.6d.

Now, we consider the same two optimization cases, however, using AD instead of FD (Figure. 5.7). In Fig. 5.7a we start the optimization with the reference configuration and a slightly different initial cost function value of -94.0 which is due to the different computation of sensitivity (AD instead of FD). The optimization halts with the final configuration $[0.5, 3.963, 3.99, 4.99]$ mm and the optimized cost function value of -162.3. This gives an increase of sensitivity by a factor of 1.73. In Fig. 5.7b we start the optimization with the 'custom' configuration and the initial cost function value of -61.77 which, again, is slightly different to the cost function computed via FD. However, this time the optimization returns nearly exactly the same optimized configuration $[0.5, 3.977, 3.99, 4.99]$ mm and the optimized cost function value of -158.8. This gives an increase of sensitivity by a factor of 2.57. In Figure 5.6c and Figure 5.7c we see the corresponding values of the cost function at each iteration step for both simulation cases computed by FD and AD respectively. We note that the start configurations in both optimization cases give nearly the same values respectively for AD and FD. A representation of the electrode configuration optimized using AD can be found in Figure 5.7d.

In conclusion, for both optimization cases the AD variant gets optimization results with a higher sensitivity value than the FD variant which even seem to be independent of the choice of the starting point.
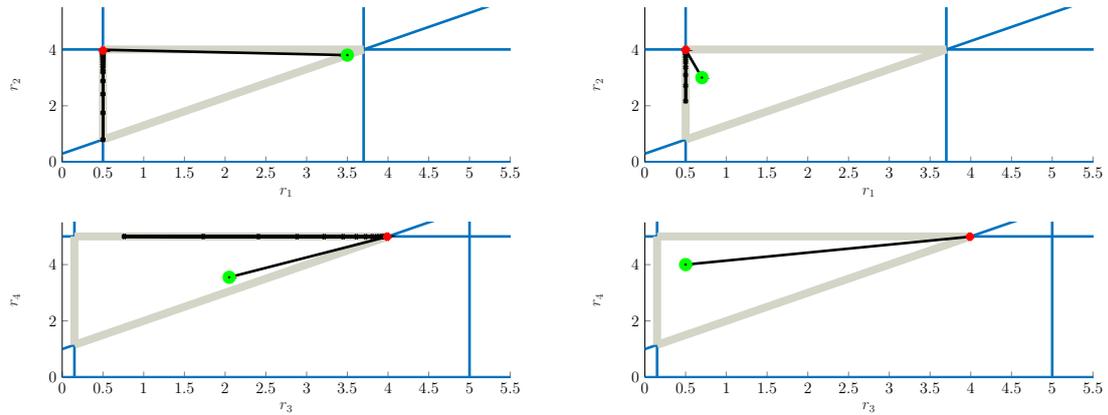
(a) Progress of electrode radii evaluation in optimization using the default reference electrode configuration. The optimizer `fmincon` finds an optimal configuration in $[0.5, 3.963, 3.99, 4.99]$ and searches the r_2 and r_3 boundary axis for better solutions.

(b) Progress of electrode radii evaluation in optimization using a changed electrode configuration. Similarly to Fig. 5.7a the optimizer also finds an optimal configuration in $[0.5, 3.977, 3.99, 4.99]$ and searches the r_2 boundary axis for better solutions.
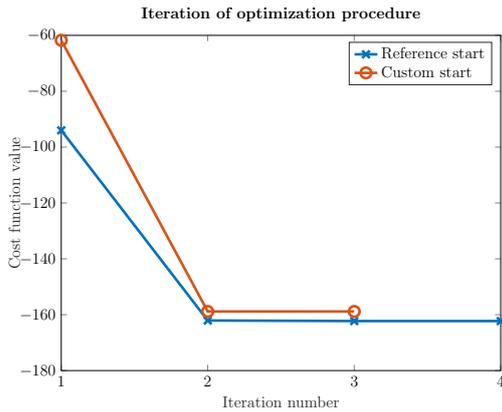
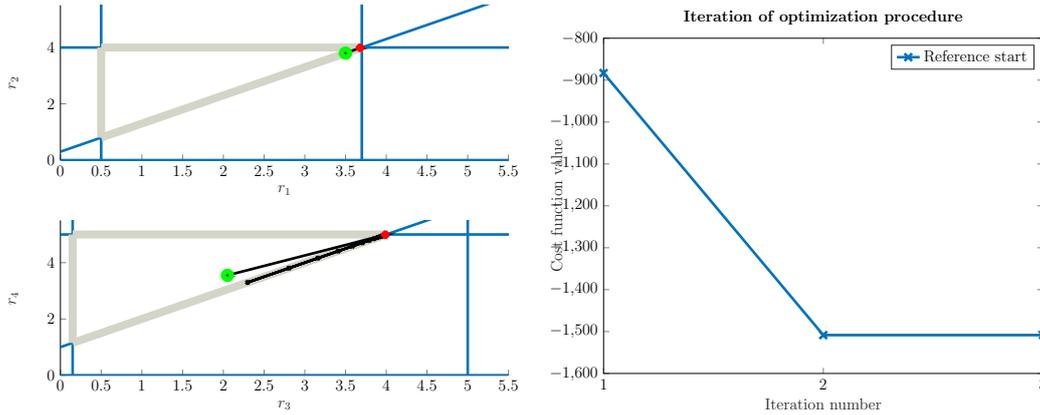(c) Iteration of optimization procedure.
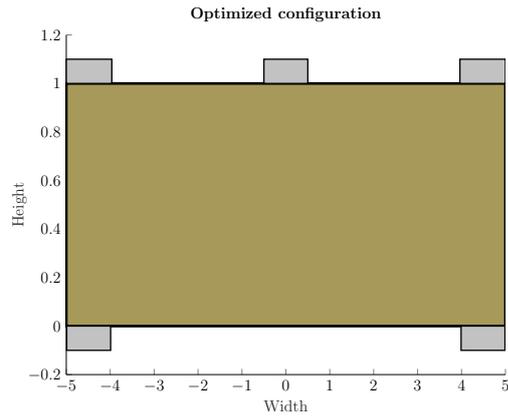
(d) Best optimized configuration $r := [0.5, 3.963, 3.99, 4.99]$ mm found for the two AD-based optimization cases presented.

Figure 5.7: Optimization via AD using two different starting points. Markings analogous to Figure 5.6.

Optimization with Respect to All Parameters

Here we show the optimization of sensitivity with respect to all material parameters. We scale the partial sensitivities by the magnitude of the corresponding material parameters and use the identity matrix as weight matrix, thus treating each material parameter equally. In Fig. 5.8 we see the progress of electrode radii evaluation in the

optimization process and the values of the cost function at each iteration. The initial cost function value at the reference starting point is -883.13 which is optimized to -1501.96. This resembles an increase in sensitivity by a factor of 1.7. The optimized configuration is $[3.68, 3.99, 3.97, 4.97]$ mm.

(a) Progress of electrode radii evaluation in optimization using the default reference electrode configuration.

(b) Iteration of optimization procedure.

Figure 5.8: Optimization for all material parameters via AD using the reference starting point. Markings analogously to Figure 5.6 and Figure 5.7.

5.3.2.2 Optimization Results Gained by `IPOPT`

The optimization method IPOPT [78] is an optimizer based on an interior point method written in C++ and can be called from within MATLAB$^{\text{TM}}$ via a separately available interface. It can directly handle constrained optimization. The objective function is computed using Algorithmic Differentiation. The gradient of the objective function is computed either by finite differences or by Algorithmic Differentiation. Here, one explicit goal is to test more initial configurations than before which allows us to broaden and verify previous results. However, for brevity not all details can be given. Hence, only the iterates and other interesting values are given here explicitly.

Starting Points for the Optimization

Ten different electrode geometries were selected as starting points for the optimiza-

tion. Nine of the ten geometries were selected in the interior of the feasible domain but close to locations where multiple constraints meet, i.e. corners of the constraint triangles. The tenth geometry was chosen to be approximately central of the domain. A graphical representation is given in Fig. 5.9.
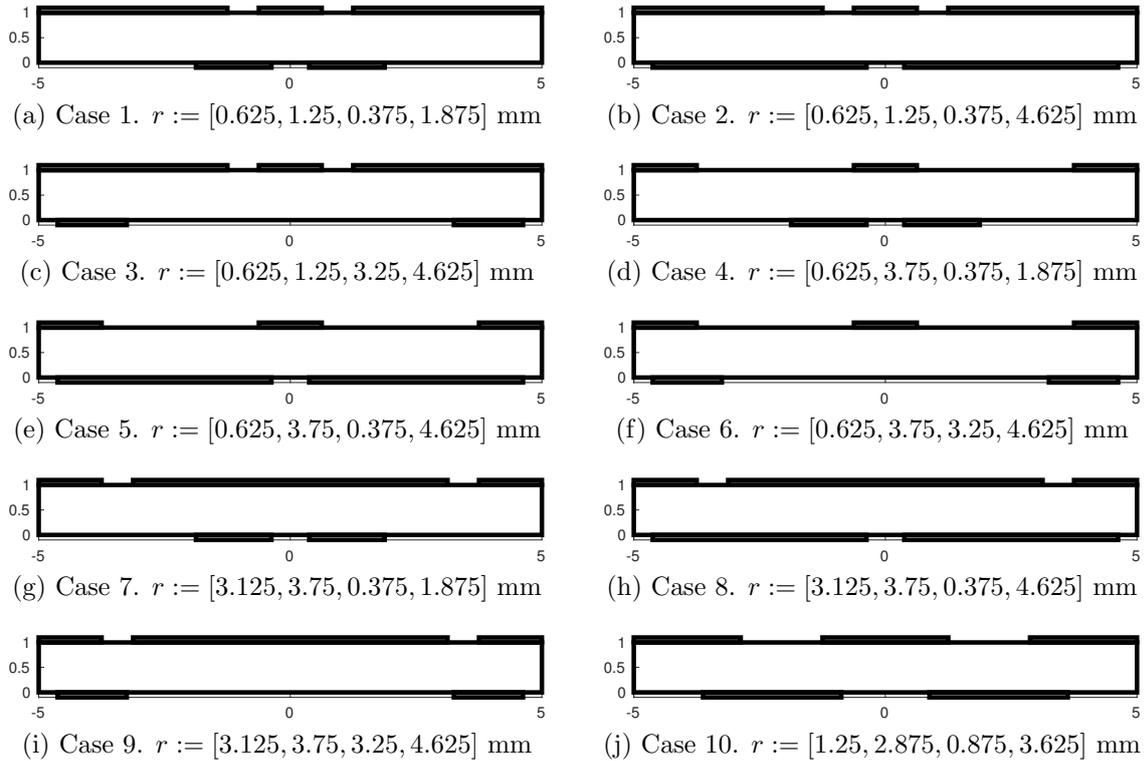

(a) Case 1. $r := [0.625, 1.25, 0.375, 1.875]$ mm

(b) Case 2. $r := [0.625, 1.25, 0.375, 4.625]$ mm

(c) Case 3. $r := [0.625, 1.25, 3.25, 4.625]$ mm

(d) Case 4. $r := [0.625, 3.75, 0.375, 1.875]$ mm

(e) Case 5. $r := [0.625, 3.75, 0.375, 4.625]$ mm

(f) Case 6. $r := [0.625, 3.75, 3.25, 4.625]$ mm

(g) Case 7. $r := [3.125, 3.75, 0.375, 1.875]$ mm

(h) Case 8. $r := [3.125, 3.75, 0.375, 4.625]$ mm

(i) Case 9. $r := [3.125, 3.75, 3.25, 4.625]$ mm

(j) Case 10. $r := [1.25, 2.875, 0.875, 3.625]$ mm

Figure 5.9: Starting geometries for optimization.

Optimization with Respect to the Single Parameter ϵ_{11}

Optimization results for an objective function incorporating only the sensitivity of ϵ_{11} are shown in Fig. 5.10 and Fig. 5.11. In all cases except for case 7 using Algorithmic Differentiation to compute the gradient of the objective function (second order derivatives) gives much better optimization results than using finite differences.

The best objective function value was obtained in case 1 with a value of -164.48 using the configuration $r := [0.5, 3.99, 4.00, 4.99]$ mm. This is (nearly) exactly the same objective function value and configuration as achieved by `fmincon` in Fig. 5.7. The second best objective function value was gained in case 2 with a value of -161.90 using the configuration $r := [0.83, 3.97, 4.0, 4.99]$ mm.

In both cases the sensitivity was approximately quadrupled when compared with the starting configuration.

Optimization with Respect to All Parameters

Optimization results for an objective function incorporating the sensitivity with respect to all parameters are shown in Fig. 5.12 and Fig. 5.13. In all cases except for case 3 and case 9 using Algorithmic Differentiation to compute the gradient of the objective function gives much better optimization results than using finite differences. However, regardless of how the gradient was computed we note that the iterates for this optimization are not as smooth as we would expect from benchmark problems.

The best objective function value was obtained in case 1 with a value of -1643.67 using the configuration $r := [0.5, 3.99, 3.99, 4.99]$ mm. Interestingly, this is (nearly) exactly the same configuration as in the case for the single material parameters ϵ_{11}.
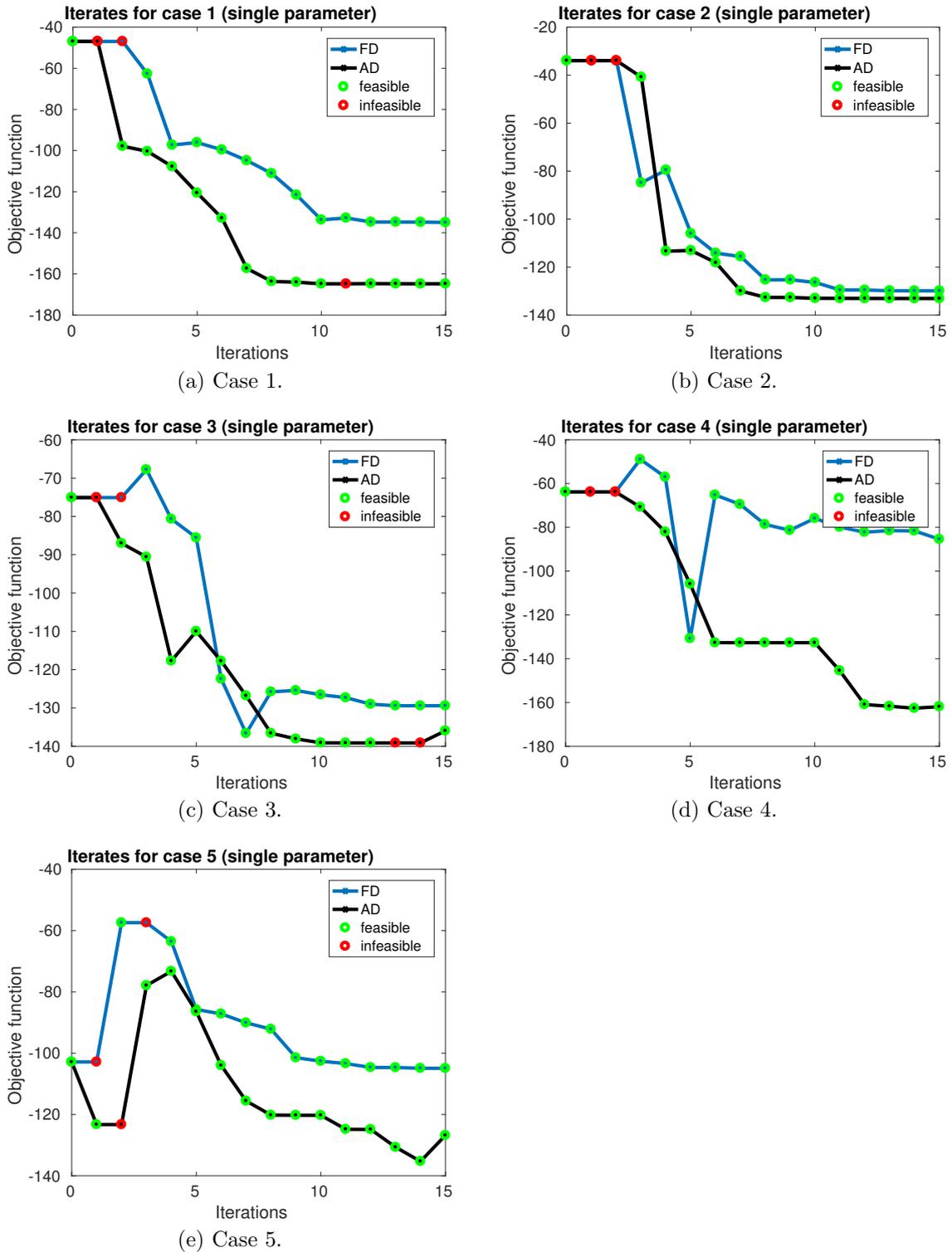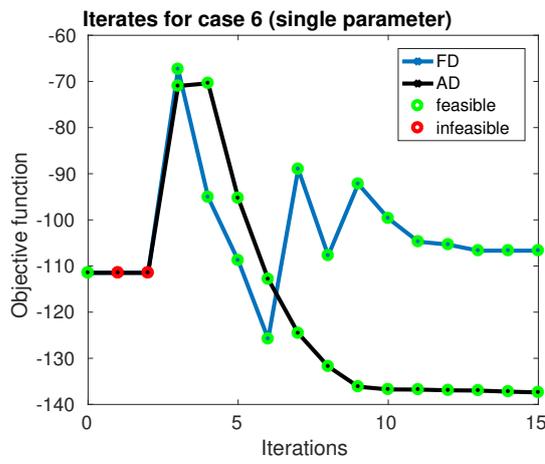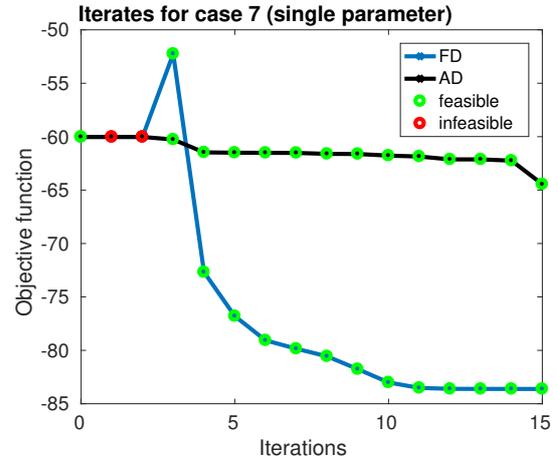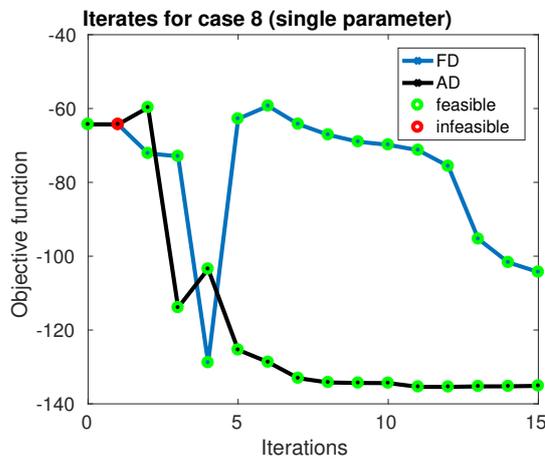
(a) Case 1.

(b) Case 2.

(c) Case 3.

(d) Case 4.

(e) Case 5.

Figure 5.10: Iterates for `IPOPT` optimization for single parameter objective function. Cases 1-5.

(a) Case 6.

(b) Case 7.

(c) Case 8.

(d) Case 9.

(e) Case 10.

Figure 5.11: Iterates for `IPOPT` optimization for single parameter objective function. Cases 6-10.

(a) Case 1.

(b) Case 2.

(c) Case 3.

(d) Case 4.

(e) Case 5.

Figure 5.12: Iterates for `IPOPT` optimization for all parameter objective function. Cases 1-5.
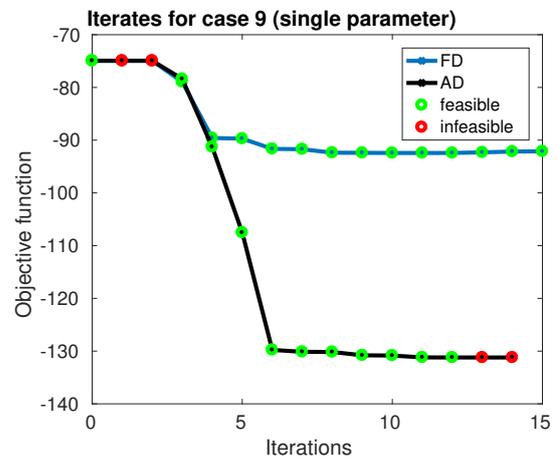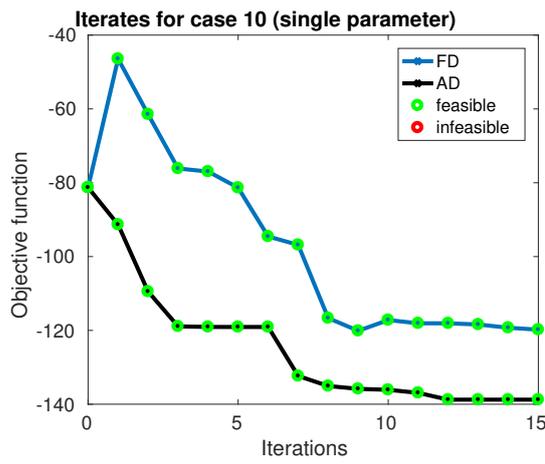
(a) Case 6.

(b) Case 7.

(c) Case 8.

(d) Case 9.

(e) Case 10.
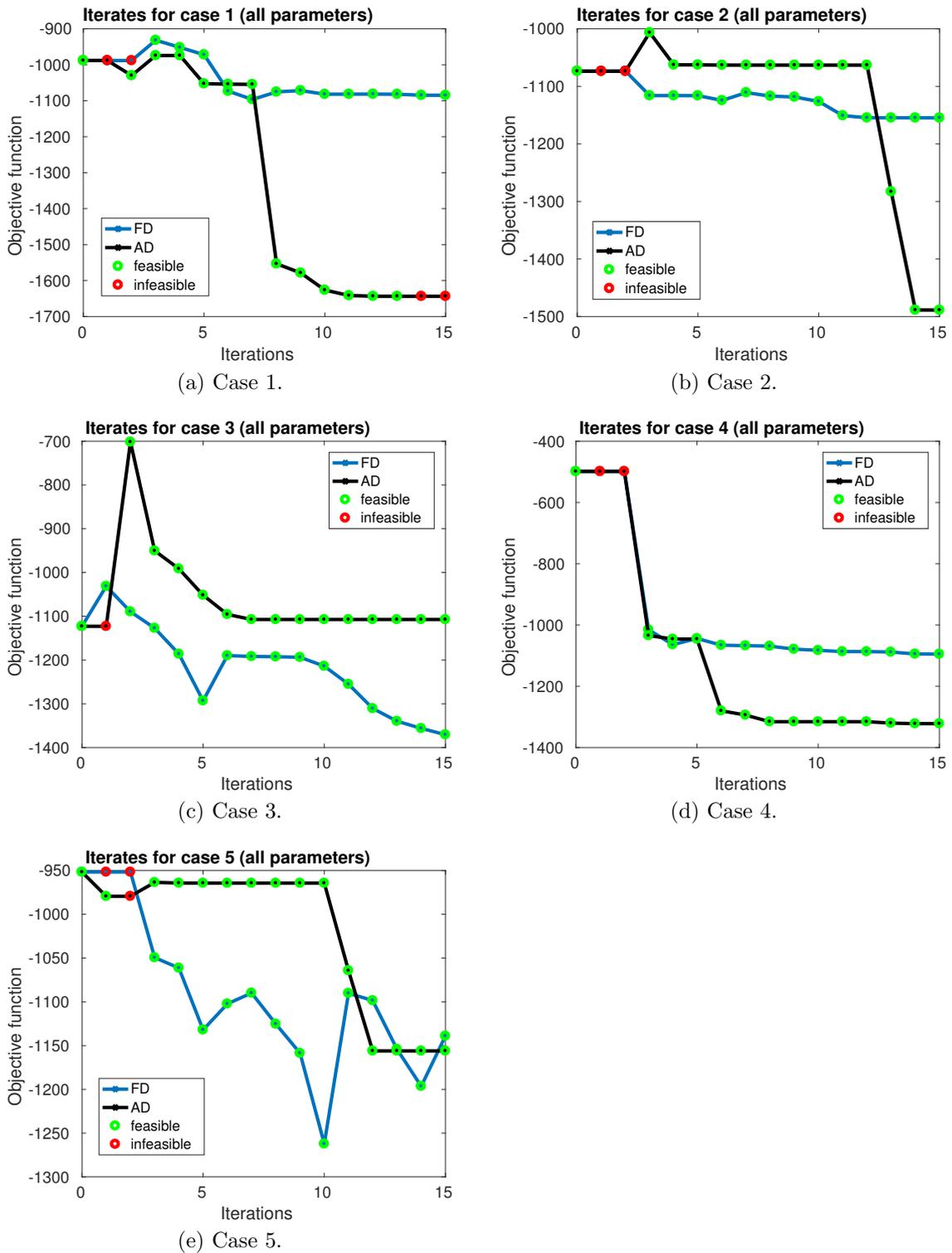
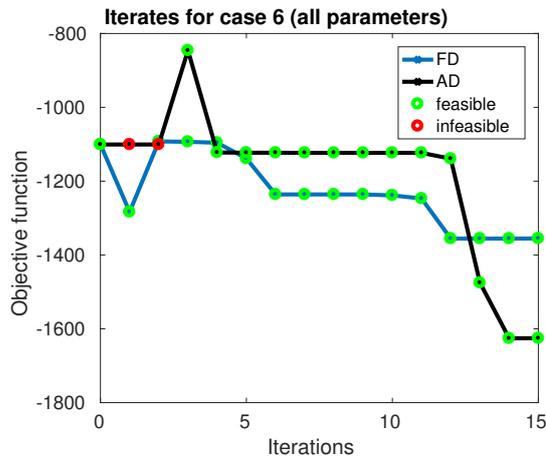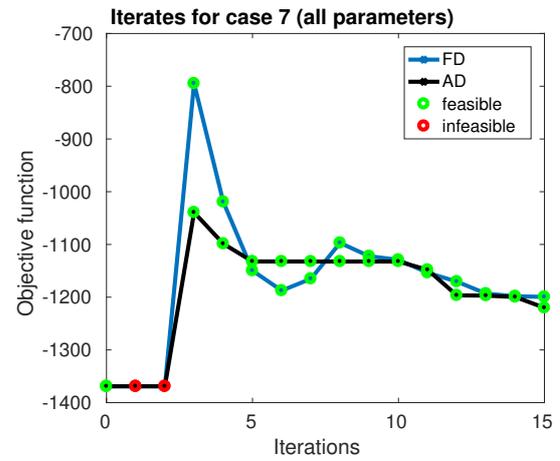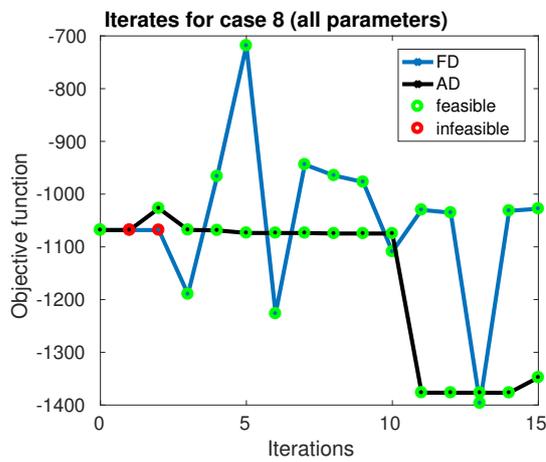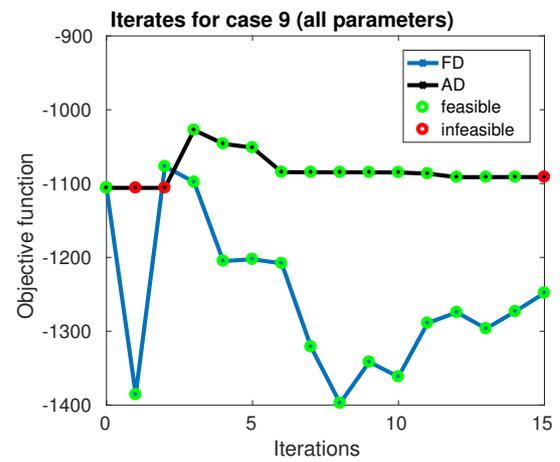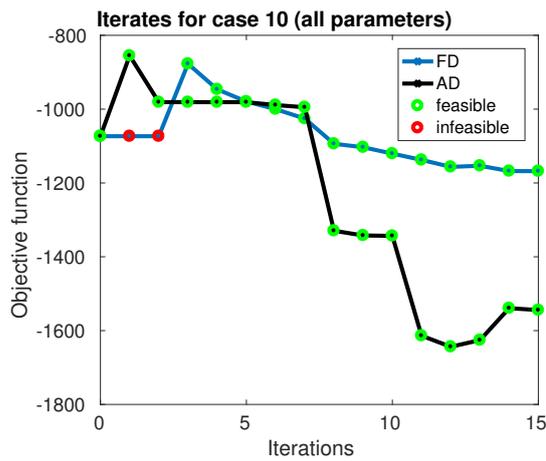Figure 5.13: Iterates for `IPOPT` optimization for all parameter objective function. Cases 6-10.

5.3.3 Discussion

Before we focus on the improvement of the objective function values for the different cases we should take a moment to discuss the unexpected behavior of some of the iterates. When examining the results gained by `IPOPT` we notice that there are few cases where using finite differences for the computation of the second derivatives provides better results than when using Algorithmic Differentiation (using a first order FD scheme applied to an objective function computed by Algorithmic Differentiation). Some additional review and debugging shows that the likely cause of this behavior is that the function is almost entirely a smooth function, but not everywhere. While the objective function does look very smooth from a macro perspective there are some exceptions where the function has very small (micro-)areas where the function slightly but suddenly drops in comparison to values surrounding this area. As finite differences have a regularizing effect, this drop in function value is not noticed by the FD scheme. In contrast, Algorithmic Differentiation gives the exact derivatives of the objective function. Hence, in these (micro-)areas the derivatives computed by AD can significantly vary from values computed by FD. Thus, the behavior of results gained by AD are in line with expectations.

Furthermore, we notice that `IPOPT` sometimes requests function evaluations outside of the feasible domain. The author verified that all settings especially concerning the definition of constraints were correct. As mentioned above, the author also verified that the derivatives of the objective function are also exactly as expected. Thus, it seems likely that this behavior is also caused by the tiny areas mentioned above.

A possible explanation for the existence of such areas could be that the mesh may have been selected too coarse. In Fig. 5.4 it is shown that the sensitivity curves converge for decreasing mesh granularity for various electrode configurations. However, in some locations the convergence is faster than in others. Hence, it seems plausible that deviations in the gradients for different mesh granularities may also vary locally and consequently converge at different paces. This was a known risk and part of the trade-off between runtime and accuracy. This explanation is consistent with the observation that none of the cases run with `fmincon`, where a very fine mesh was selected, had similar issues.

As simulations with a finer mesh require much more runtime (optimization cases

run with `fmincon` had a runtime of approx. two weeks each) further parallelization of the code would be necessary. This is currently not possible with the simulation code used here in the exact case considered.

Now that the behavior has been addressed we can focus the achieved improvements which were gained nevertheless. First, it is reassuring that the best configurations obtained by both optimization methods `fmincon` and `IPOPT` for the single parameter case are given by the configuration $r_1 = 0.5$, $r_2 = 3.99$, $r_3 = 3.99$, $r_4 = 4.99$ even while using a different set of starting points. By using many different starting points this indicates that this configuration is likely to be one of the best, or perhaps the best configuration possible. Due to the fact that configurations, which yield close to the best values obtained, only vary in the first variable it seems that configurations along the values of $r_2 = 3.99$, $r_3 = 3.99$, $r_4 = 4.99$ generally give very good results.

When optimizing the sensitivity with respect to all material parameters the best results obtained by the optimizers vary. Here, `IPOPT` finds the objective function value -1643.67 with the configuration $r_1 = 0.5$, $r_2 = 3.99$, $r_3 = 3.99$, $r_4 = 4.99$ which is approx. 10% larger in magnitude than the best value -1501.96 obtained by `fmincon`. Without testing more starting configurations than before it is likely that this configuration may have been missed. Consequently, this gain in sensitivitity was only feasible because of the trade-off mentioned earlier.

Furthermore, it is interesting that the best configuration for the single parameter and complete parameter cases give nearly the same result. The author hypothesizes that there must be a comprehensive physical explanation why exactly this configuration gives the best results. The author, however, cannot give an explanation.

Moreover, these results presented in here indicate that using AD instead of FD for gradient calculation and `IPOPT` instead of the standard approach `fmincon` have been beneficial for this specific optimization: For optimization cases regarding only one material parameter it confirms the previous results obtained via standard approaches. In almost all optimization cases considered here better results were obtained when additionally AD was used for second order derivatives. In cases where AD was not used at least for the computation of the objective function the achieved optimization results were not as good in comparisson.

6

Inverse Problems in Piezoelectricity

In the last chapters much effort was put into the ability to reliably increase the sensitivity of impedance with respect to the material parameters e.g. selectively for specific material parameters. In this chapter we want to reap the benefits of the work and apply this to the problem of identifying material parameters of piezoelectric ceramics. It is not the goal of this chapter nor of this thesis to present a fully functional and reliable inverse scheme to compute all material parameters using only one single piezoceramic. This would require much more effort and more specialized handling such as special estimators for initial values and more specialized methods etc. An example for this is given in [17] where the triple-ring electrode geometry developed in this thesis is used for material parameter identification based on real measurements. However, the author of this thesis may well conduct future research on this topic more thoroughly from a mathematical and algorithmic point of view. Instead this chapter should be interpreted more as a 'proof of concept': The goal is to demonstrate that the optimization of sensitivity has had a positive impact and that it now is at least easier to solve these still very demanding inverse problems than before, especially when using only a single piezoceramic and only impedance measurements.

As mentioned in the introduction, in the 1987 *IEEE Standard on Piezoelectricity* [69] there are some methods described in what manner the material parameters of piezoelectrics could and should be retrieved. However, following this and similar methods there may be a very significant error of up to 20% in the reconstructed material parameters [54]. This large error is unacceptable if a computer simulation is to successfully predict the correct physical behavior of devices containing piezoceramics. Hence, work that can help decrease the error of estimated material parameters is directly related to the ability to design and construct better devices

involving piezoceramics. As piezoceramics are a very widespread technical component, in principle, a better method to determine the correct material parameters would have a large impact.

As there is no direct way of measuring some of the material parameters a common method is to reformulate the identification problem as a so-called inverse problem and solve the latter.

6.1 Ill-posed and Inverse Problems

We begin by explaining what an inverse problem is and what properties of common inverse problems are. In order to distinguish inverse problems from other problems we introduce the notion of a *direct problem*. Given cause, the task of a direct problem is to compute the effect the cause has had. We can interpret this more mathematically by assuming a cause $x \in X$. The task is now to find the effect $f(x)$ a given cause x has had. Or simpler: given an $x \in X$ evaluate $f(x) = y \in Y$ where it is common to assume Hilbert spaces $(X, \|\cdot\|_X), (Y, \|\cdot\|_Y)$.

Conversely, the *inverse problem* is to identify the cause x from a given effect $y = f(x)$ also illustrated in Fig. 6.1. This problem has some straightforward properties:
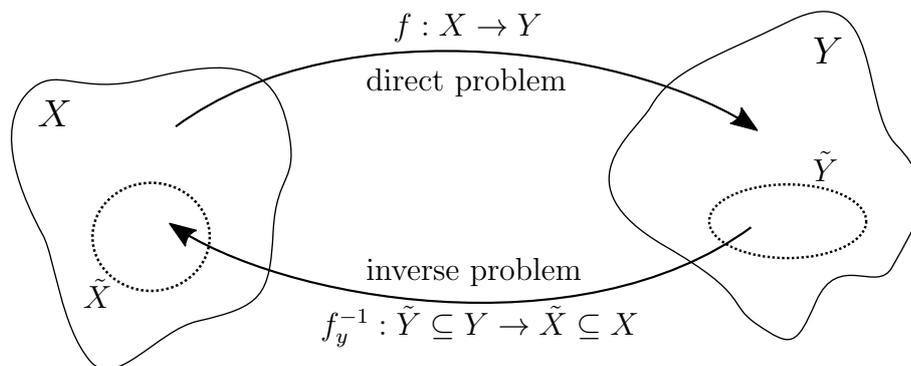

Figure 6.1: Direct and inverse problem.

- Obviously, if the function f is surjective then for every $y \in Y$ there exists at least one $x \in X$ such that we get $y = f(x)$. In other words, if f is surjective then the inverse problem has a solution.

- On the other hand, if f is injective then each cause leads to a different effect $\forall x_1, x_2 \in X : x_1 \neq x_2 \Rightarrow f(x_1) \neq f(x_2)$. If f is not injective, then multiple causes can have the same effect. In other words, if f is injective then solutions to the inverse problem are unique.

- Lastly, the solution depends continuously on the data, i.e. f^{-1} is continuous (also referred to as *stability*).

If for an inverse problem there does not exist a stable and unique solution then the problem is called an *ill-posed problem* in the sense of Hadamard. In fact, many real physical problems are ill-posed problems. The most prominent reason is that real measurement data is never perfectly accurate and always holds some quantity of measurement uncertainty (e.g. Heisenberg's uncertainty principle or round-off errors when using measurement data on a computer).

6.2 Solution Methods for Inverse Problems in Piezoelectricity

First, a reminder of the goal of the inverse problem is stated: Let us assume we have measured the frequency dependent impedance curve for a triple-ring electrode piezoceramic, in order to keep to the usual notation of the literature we will call the measurement y_d.

We now aim to reconstruct the material parameters p. Then, this can formally be interpreted as finding $p \in \mathbb{R}^{10}$ such that the error between the measurement and the simulation

$$E(p) := \|Z(\omega, p) - y_d(\omega)\|^2$$

is minimal for the simulated impedance curve $Z(p)$. As usual, perfectly accurate measurement data y_d cannot be guaranteed. In Fig. 1.3 it is also demonstrated that, using a fully-covering electrode setup, a small (or sometimes even zero) error E can be achieved for incorrect material parameters. If we combine the latter two facts we can deduce that it is very likely that we are dealing with an ill-posed problem.

There are a variation of tools we can apply in order to deal with specific aspects of ill-posed problems: If, due to low or zero sensitivity of some material parameters, we

do not have an injective function we can try and increase the sensitivity. This was the goal of this thesis. In Fig. 6.2 we can now see that this situation has changed.

However, this does not deal with the fact that the function still may not be continuous or surjective. The usual procedure in this case is to transform the original problem into a well-posed problem by using *regularization* techniques e.g. [14, 32].

One category of regularization methods (so-called *Tikhonov methods*) is based on adding some variation of a penalty term e.g. $\gamma \|p - p_0\|^2$ to the original problem, e.g.

$$\min_{p \in \mathbb{R}^{10}} E(p) := \|Z(\omega, p) - y_d(\omega)\|^2 + \gamma \|p - p_0\|^2.$$

This penalty term can be chosen in such a way that the problem now has much better theoretical or practical properties. Depending on the choice of the penalty term different regularization methods arise. However, this category of methods can also be combined with other categories such as using nonlinear iterative methods and stopping the process early, or by using coarse discretization.

Some of the classical methods include [14, 32]

- Nonlinear Landweber iteration
- Newton type methods such as the Levenberg–Marquardt method or iteratively regularized Gauss–Newton methods
- Multilevel methods

In the area of piezoelectrics there exists a vast amount of publications relating to material parameter identification methods. Some of the main works from a mathematical perspective are given in [31, 38] and references therein. An overview of the current state of the art for material parameter identification methods for piezoceramics is given in the recent [61].

Remark 6.1. Clearly one could try to use 'normal' optimization algorithms to minimize the error. However, due to the nature of ill-posed problems this usually converges to an incorrect local minimum, fails to converge or is impractical due to excessive runtime. This is due to the fact that the unregularized minimization problem $\min_{p \in \mathbb{R}^{10}} E(p)$ for $\gamma = 0$ suffers from the same ill-posedness as the original problem $Z(p) = y_d$.
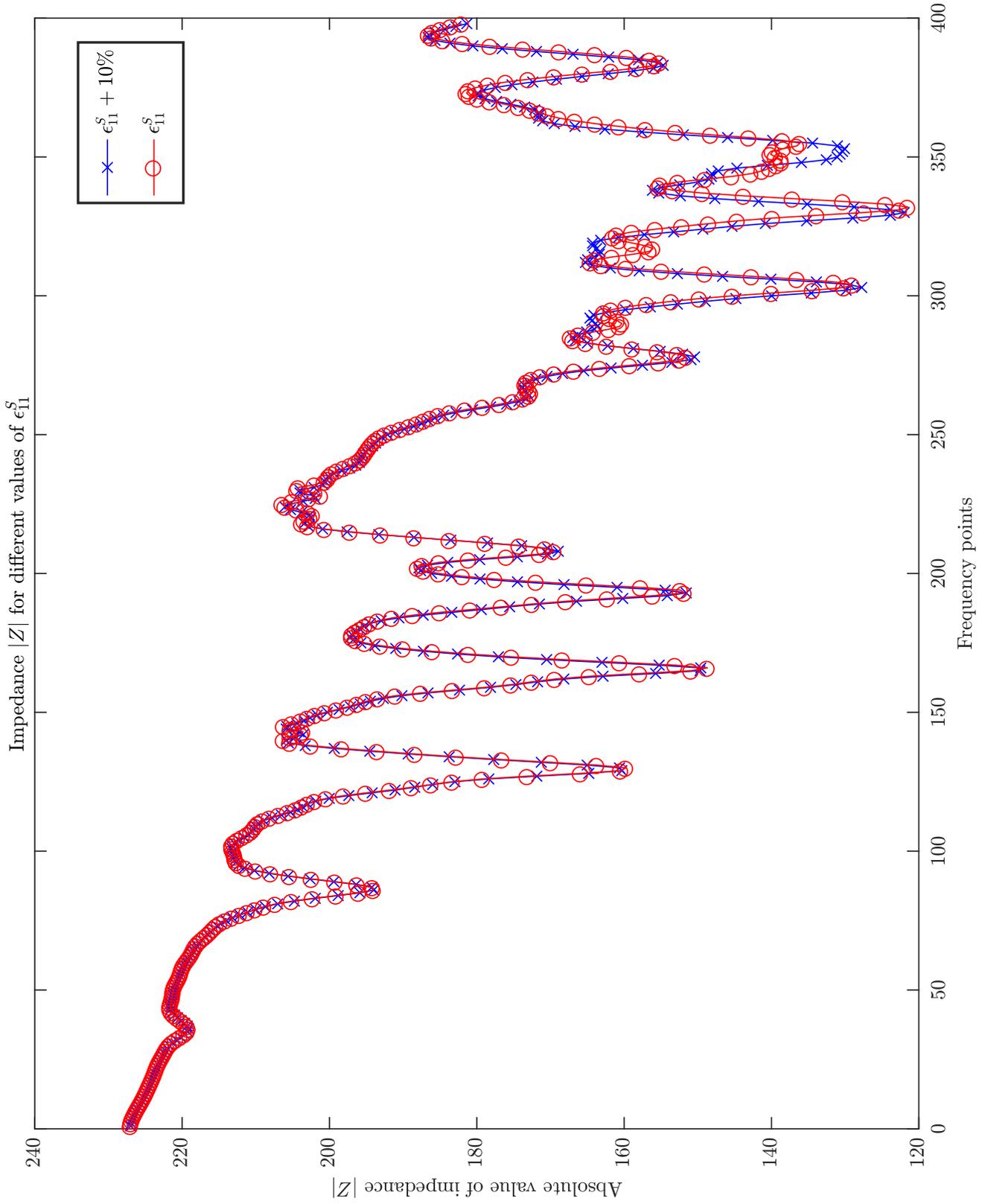
Figure 6.2: Impedance curves for different values of ϵ_{11} after sensitivity optimization.

6.3 Results

In this section we will test some available solution algorithms for inverse problems but also give some practical results for our own approaches. The methods used here are
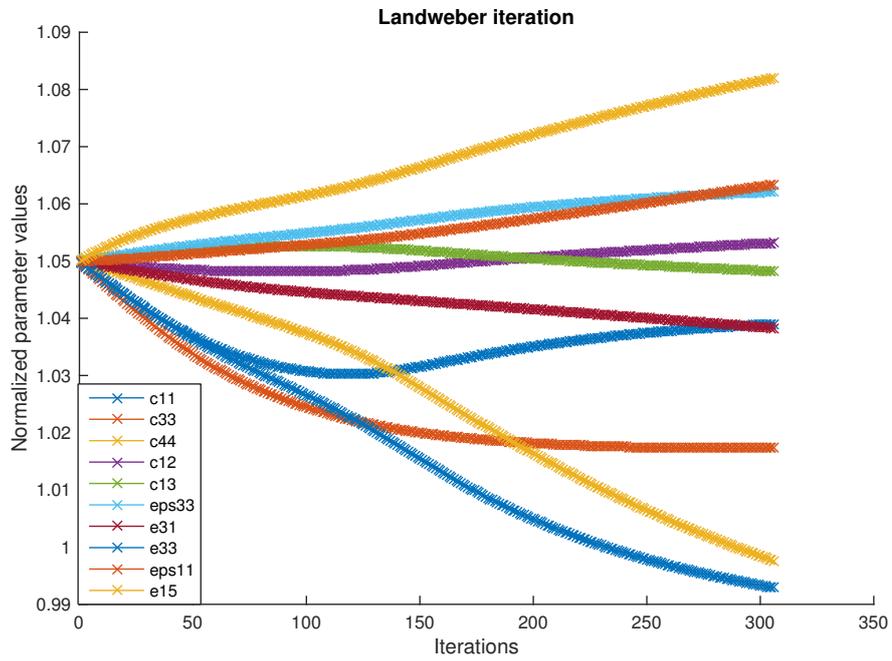
1. Landweber iteration,

2. Iteratively regularized Gauss-Newton method,

3. Iteratively regularized Gauss-Newton method applied to subsets of material parameters and

4. Block coordinate descent iteratively regularized Gauss-Newton method applied to all material parameters.

Due to the fact that we only want to provide a proof of concept here we will not go into details of the methods but instead outline interesting aspects. The reader may refer to [14], [32] or [38].

As a test setup a measurement of the impedance curve was generated via simulation. In order to overcome the so-called *inverse crime* a much finer computational mesh for the generation of the artificial measurement of the impedance curve is used than is used for the simulations. For convenience of the reader the material parameters are all normalized. Hence, ideally we hope that all curves converge to the value 1. In all test cases a 5% deviation for all parameters from the correct parameters was chosen as a start point.

Landweber Iteration

We start with the Landweber iteration which is known to converge in a very stable fashion under some assumptions. Do note that the convergence rate can be agonizingly slow. In Fig. 6.3 the results for the Landweber iteration are shown. In the first plot the development of the material parameters are shown. In the second plot the error E is displayed. We note that the error steadily decreases. However, after 300 iterations the Landweber method has not yet fully converged and the material parameters seem to diverge from the correct solution.

(a) Landweber iteration - development of parameters.
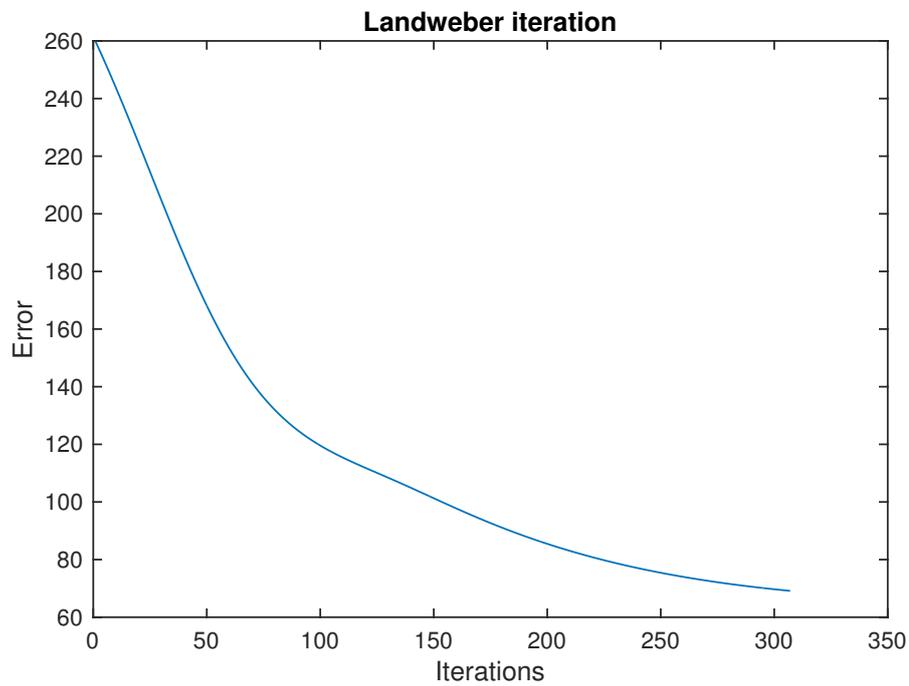
(b) Landweber iteration - development of error.

Figure 6.3: Landweber iteration.

Iteratively Regularized Gauss-Newton Method

By deploying a Newton type method we can now increase the convergence rate significantly.
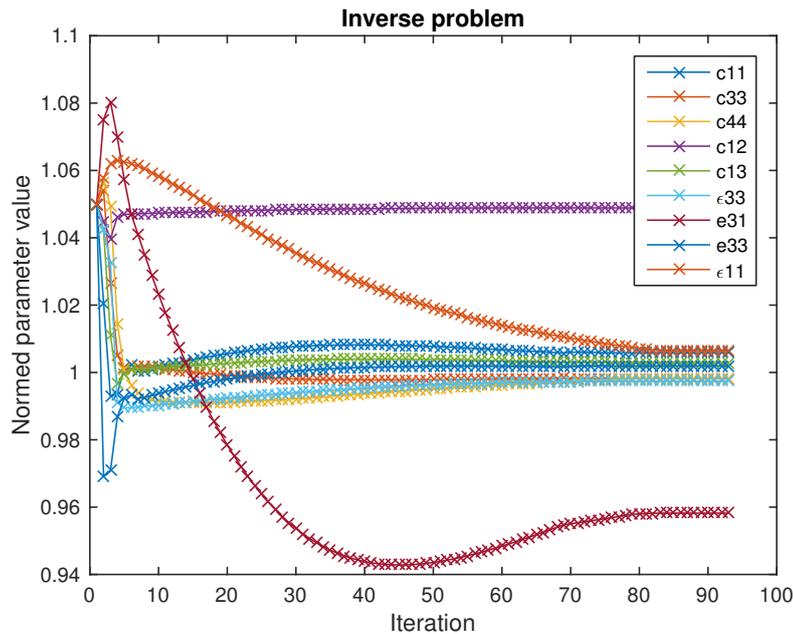
Figure 6.4: Gauss-Newton method for inverse problem.

In Fig. 6.4 we see that all parameters seem to have converged after approx. 100 iterations. However, we note that we did not find the correct material parameters.

Iteratively Regularized Gauss-Newton Method Applied to Subsets of Material Parameters

By analyzing the last case we noticed that some material parameters with comparatively low sensitivity are 'dominated' by other material parameters with high sensitivity. Hence, in the next test, see Fig. 6.5, we try to reconstruct these two

groups separately and independently assuming the other group already has the correct values.

We note that each group can be quickly (under 10 iterations) and correctly reconstructed.

Block Coordinate Descent Iteratively Regularized Gauss-Newton Method Applied to All Material Parameters

The former test motivates us to partition material parameters with similar sensitivities into groups and apply the iteratively regularized Gauss-Newton method on these separately while holding the other (possibly incorrect) parameters constant. This procedure, see Fig. 6.6, is repeated until the resulting error is sufficiently low. As we are applying our solution methods onto variable blocks this method is formally a block coordinate descent method with an iteratively regularized Gauss-Newton method as the submethod.

We notice that all parameters seem to converge to the correct material parameter values but more iteration steps ultimately would be necessary. Finally, in Fig. 6.7 the measured and reconstructed impedance curve are shown. We note that we have a very good match between these two curves.

The author notes that he is unaware of any similar results gained by other authors using only a single piezoceramic and only impedance measurements. Hence, it is concluded that it seems feasible to gain all correct material parameter values with this or more specialized methods. However, there is definitely room for further improvement especially concerning convergence rates.
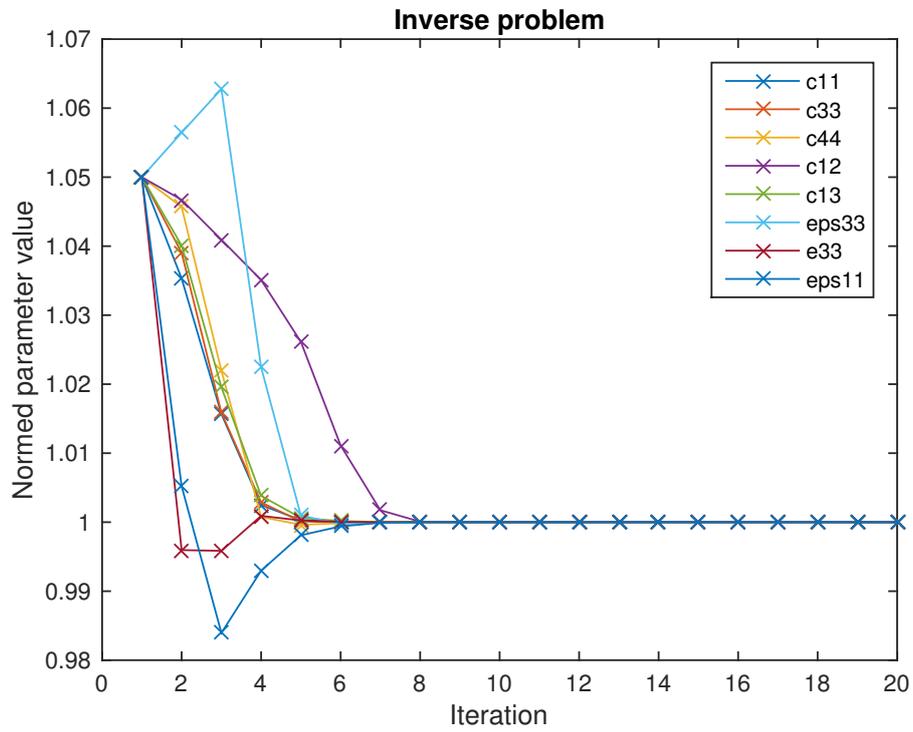
(a) Restriction to 8 out of 10 parameters.
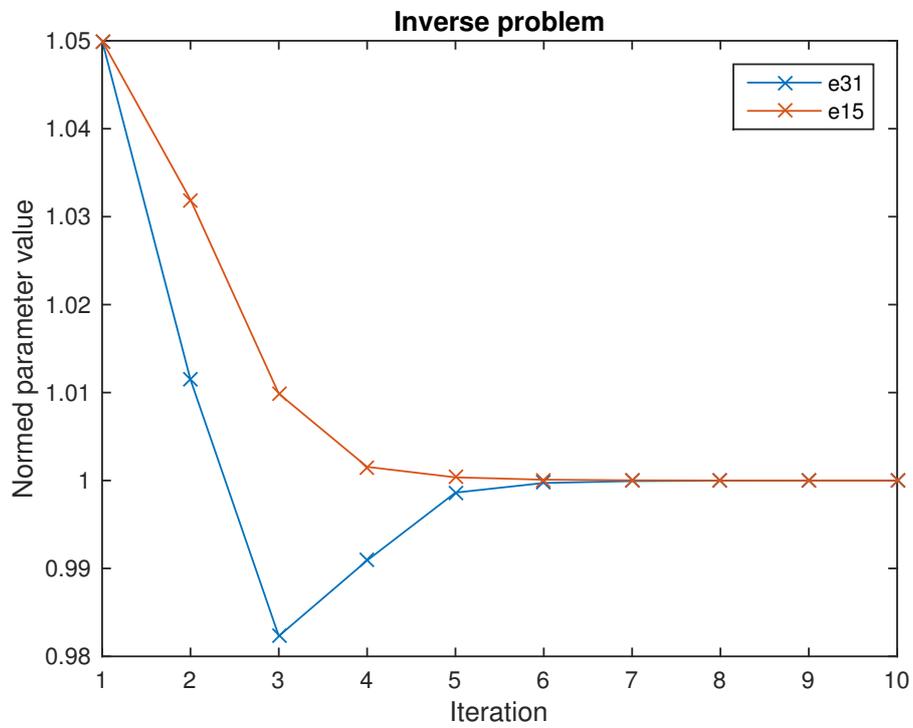
(b) Restriction to the remaining 2 out of 10 parameters.

Figure 6.5: Separated reconstruction.
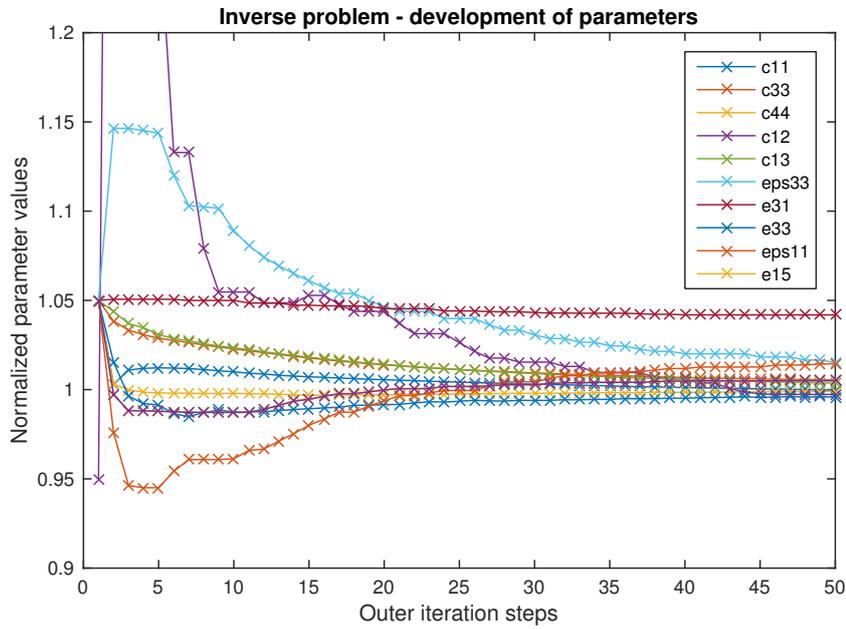
(a) Block coordinate descent iteratively regularized Gauss-Newton - development of parameters.
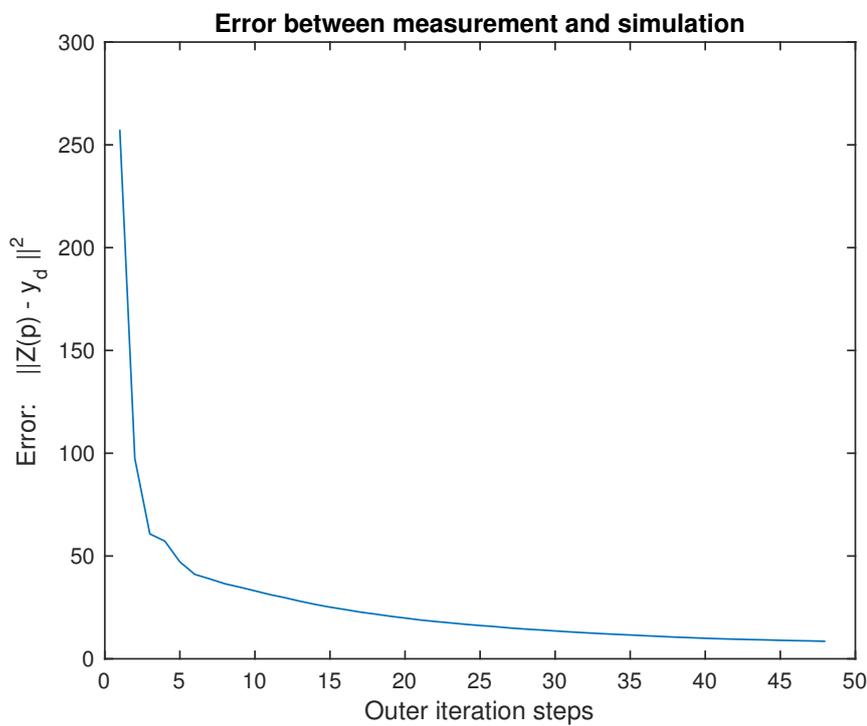
(b) Block coordinate descent iteratively regularized Gauss-Newton - development of error.

Figure 6.6: Block coordinate descent iteratively regularized Gauss-Newton.
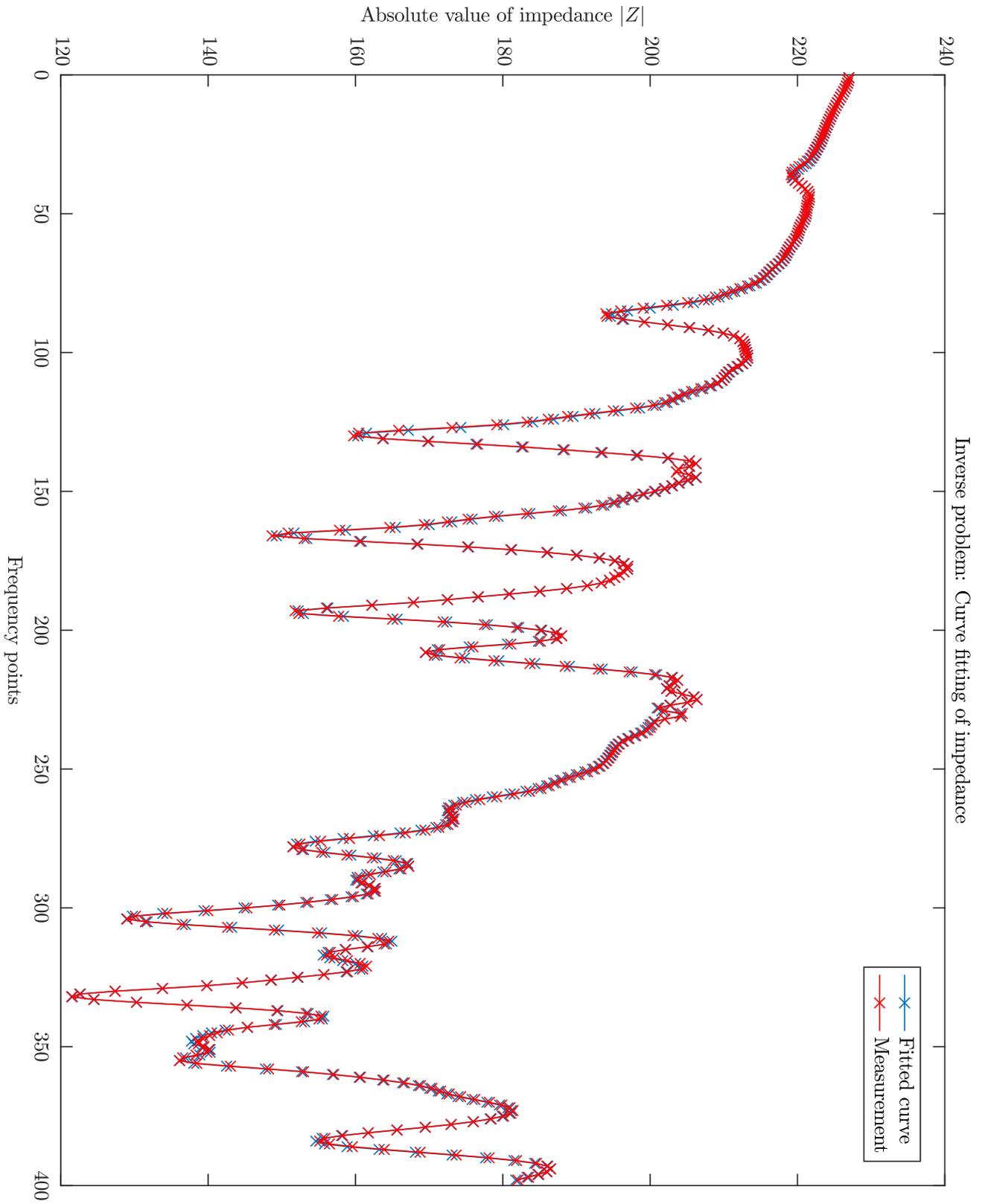
Figure 6.7.: Fitting of artificially measured and simulated impedance.

7

Conclusion

7.1 Summary

In this thesis the possibility of increased sensitivity in parameter identification problems for piezoelectrics, especially for triple-ring electrode piezoceramics, using only a single piezoceramic specimen and only impedance measurements was investigated. Partial differential equations governing the piezoceramics were derived, existence, uniqueness, and regularity results are stated and proved. Furthermore, the impedance of triple-ring electrodes was defined and computed. Sensitivities of the impedance were then defined, computed, analyzed and accuracy thereof was increased. Then, by optimizing the ring electrode geometry the sensitivity of impedance with respect to all material parameters was significantly increased. This lead to a proof of concept parameter identification method for triple-ring electrode piezoceramics that shows that the parameter identification problem is now easier to solve. Additionally, a novel extension based on the optimized electrode geometry was developed which allows directly prescribing arbitrary derivatives for the measurable quantity of the inverse problem.

7.2 Future Research Directions

Based on this thesis many more possible research directions open:

First, the work in this thesis is based on triple-ring electrode geometries parametrized by four variables. In future this could be extended to incorporate more general electrode geometries. This could be achieved by applying shape and/or topology optimization.

Related to this, one possible improvement could be the use of the software package FEniCS instead of CFS++. This is especially interesting as the newest version of FEniCS already incorporates shape optimization packages without the tedious requirement of manipulating the piezoelectric equations for shape gradient calculus. The other benefit is that this software already has the capability to compute derivatives with the AD package dolfin-adjoint [16]. This should make it possible to compute adjoints in reverse mode AD which was not possible in this thesis due to reasons presented in Appendix A.3. Furthermore, this software package was designed to run in the context of high performance computing (HPC). This is not the case for CFS++ where parallelization of the code cannot be achieved easily (see also [29]). Some effort by the author and his colleagues has been put into rebuilding the current simulation and optimization capabilities for the software package FEniCS. However, currently some additional work is still required.

The author is also excited about the novel approach presented in section 3.4 which allows manipulating the impedance in such a way that it is possible to directly set *any* arbitrary sensitivity desired. This has the potential to greatly improve the quality of material parameter identification methods in piezoelectrics. As this approach came up while writing this thesis it has not yet been fully researched. Hence, investigations on how this can be exploited are ongoing.

Furthermore, the work presented here was done in a simplified setting of linear piezoelectric equations excluding thermal effects. Hence, in future this could be expanded to incorporate nonlinear effects and/or other piezoelectric materials with different crystal structures such as quartz. As common piezoceramics currently contain lead (e.g. PZT) this is of direct interest as the EU is restricting the use of lead in appliances. Currently, some piezoelectrics are exempt from the restrictions (by e.g. REACH Article 58 (2)). However, this may change in future if and when other piezoelectric materials have been developed which can replace current lead-based piezoceramics.

Bibliography

[1] R. A. Adams and J. J. F. Fournier. *Sobolev Spaces*. Pure and Applied Mathematics. Elsevier Science, 2003.

[2] M. Akamatsu and G. Nakamura. Well-posedness of initial-boundary value problems for piezoelectric equations. *Applicable Analysis*, 81(1):129–141, 2002.

[3] M. S. Alnæs, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M. E. Rognes, and G. N. Wells. The FEniCS Project Version 1.5. *Archive of Numerical Software*, 3(100), 2015.

[4] APC International. The top uses of piezoelectricity in everyday applications. https://www.americanpiezo.com/blog/top-uses-of-piezoelectricity-in-everyday-applications/, 2015. [Online; accessed 3-September-2018].

[5] I. Berg. muparser - Fast Math Parser Library. http://beltoforion.de/article.php?a=muparser. [Online; accessed 3-September-2018].

[6] C. H. Bischof, H. M. Bücker, B. Lang, A. Rasch, and A. Vehreschild. Combining source transformation and operator overloading techniques to compute derivatives for MATLAB programs. In *Proceedings of the Second IEEE International Workshop on Source Code Analysis and Manipulation (SCAM 2002)*, pages 65–72, Los Alamitos, CA, USA, 2002. IEEE Computer Society.

[7] F. Buiochi, C. Y. Kiyono, N. Perez, J. C. Adamowski, and E. C. N. Silva. Efficient algorithm using a broadband approach to determine the complex constants of piezoelectric ceramics. *Physics Procedia*, 70:143 – 146, 2015. Proceedings of the 2015 ICU International Congress on Ultrasonics, Metz, France.

[8] P. D. Conejo, E. W. Karas, and L. G. Pedroso. A trust-region derivative-free algorithm for constrained optimization. *Optimization Methods and Software*, 30(6):1126–1145, 2015.

[9] V. Del Toro. *Electrical engineering fundamentals*. Prentice Hall, 1972.

[10] P. Deuflhard and A. Hohmann. *Numerische Mathematik I: eine algorithmisch orientierte Einführung*. Walter de Gruyter, 1993.

[11] P. Dineva, D. Gross, R. Müller, and T. Rangelov. Piezoelectric materials. In *Dynamic Fracture of Piezoelectric Materials*, pages 7–32. Springer, 2014.

[12] Z. Ding. A proof of the trace theorem of Sobolev spaces on Lipschitz domains. *Proceedings of the American Mathematical Society*, 124(2):591–600, 1996.

[13] O. Ebel. Analyse komplexer harmonischer Differentialgleichungen zur FE-Simulation des elektromechanischen Verhaltens einer piezoelektrischen Keramik. Master's thesis, Paderborn University, 2017.

[14] H. W. Engl, M. Hanke-Bourgeois, and A. Neubauer. *Regularization of Inverse Problems*. Mathematics and Its Applications. Springer Netherlands, 1996.

[15] L. C. Evans. *Partial Differential Equations*. Graduate studies in mathematics. American Mathematical Society, 1998.

[16] P. E. Farrell, D. A. Ham, S. W. Funke, and M. E. Rognes. Automated derivation of the adjoint of high-level transient finite element programs. *SIAM Journal on Scientific Computing*, 35(4):C369–C393, 2013.

[17] N. Feldmann, B. Jurgelucks, L. Claes, V. Schulze, B. Henning, and A. Walther. An inverse approach to the characterisation of material parameters of piezoelectric discs with triple-ring-electrodes. *tm-Technisches Messen*, Accepted.

[18] W. Geis. *Piezoelectricity in composites: Modelling, analysis and simulation.* PhD thesis, University Stuttgart, 2007.

[19] W. Geis, G. Mishuris, and A. M. Sändig. Piezoelectricity in multilayer actuators. Modelling and analysis in two and three dimensions. Technical report, University of Stuttgart, 2003.

[20] A. Griewank, J. Utke, and A. Walther. Evaluating higher derivative tensors by forward propagation of univariate Taylor series. *Mathematics of Computation*, 69(231):1117–1130, 2000.

[21] A. Griewank and A. Walther. *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*. Society for Industrial Mathematics, 2nd edition, 2008.

[22] N. Guo. *The vibration characteristics of piezoelectric discs*. PhD thesis, University of London, 1989.

[23] J. F. Hall. Problems encountered from the use (or misuse) of Rayleigh damping. *Earthquake Engineering & Structural Dynamics*, 35(5):525–545, 2006.

[24] T. Hegewald. *Modellierung des nichtlinearen Verhaltens piezokeramischer Aktoren.* PhD thesis, University of Erlangen-Nuremberg, 2008.

[25] T. Ikeda. *Fundamentals of piezoelectricity.* Oxford University Press, 1996.

[26] F. Jarre and J. Stoer. *Optimierung.* Springer-Lehrbuch. Springer Berlin Heidelberg, 2013.

[27] G. Jones. Genetic and evolutionary algorithms. *Encyclopedia of Computational Chemistry*, 2:1127–1136, 1998.

[28] U. G. Jonsson, B. M. Andersson, and O. A. Lindahl. A FEM-based method using harmonic overtones to determine the effective elastic, dielectric, and piezoelectric parameters of freely vibrating thick piezoelectric disks. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 60(1):243–255, January 2013.

[29] B. Jurgelucks, L. Claes, A. Walther, and B. Henning. Optimization of triple-ring electrodes on piezoceramic transducers using algorithmic differentiation. *Optimization Methods and Software*, 33(4-6):868–888, 2018.

[30] B. Jurgelucks, N. Feldmann, L. Claes, B. Henning, and A. Walther. Material parameter determination of a piezoelectric disc with triple-ring-electrodes for increased sensitivity. *Proceedings of Meetings on Acoustics*, 32(1):030010, 2017.

[31] B. Kaltenbacher, T. Lahmer, M. Mohr, and M. Kaltenbacher. PDE based determination of piezoelectric material tensors. *European Journal of Applied Mathematics*, 17(4):383–416, 2006.

[32] B. Kaltenbacher, A. Neubauer, and O. Scherzer. *Iterative regularization methods for nonlinear ill-posed problems.* Walter de Gruyter, 2008.

[33] M. Kaltenbacher. Advanced simulation tool for the design of sensors and actuators. *Procedia Engineering*, 5:597 – 600, 2010. Eurosensor XXIV Conference.

[34] M. Kaltenbacher. *Numerical simulation of mechatronic sensors and actuators: finite elements for computational multiphysics.* Springer Berlin Heidelberg, 2015.

[35] J. Kocbach. *Finite element modeling of ultrasonic piezoelectric transducers.* PhD thesis, Department of Physics, University of Bergen, 2000.

[36] A. Koniaeva and E. Lorenzen. Berechnung von höheren Ableitungen und ihre Anwendung im Kontext der Renormierungsgruppenflussgleichung im Quark-Meson Modell. Bachelor's thesis, Paderborn University, 2011.

[37] K. Kulshreshtha, B. Jurgelucks, F. Bause, J. Rautenberg, and C. Unverzagt. Increasing the sensitivity of electrical impedance to piezoelectric material parameters with non-uniform electrical excitation. *Journal of Sensors and Sensor Systems*, 4:217–227, 2015.

[38] T. Lahmer. *Forward and inverse problems in piezoelectricity.* PhD thesis, University of Erlangen-Nuremberg, 2008.

[39] T. Lahmer, J. Ilg, and R. Lerch. Variance-based sensitivity analyses of piezoelectric models. *Computer Modeling in Engineering & Sciences*, 106(2):105–126, 2015.

[40] T. Lahmer, B. Kaltenbacher, and V. Schulz. Optimal measurement selection for piezoelectric material tensor identification. *Inverse Problems in Science and Engineering*, 16(3):369–387, 2008.

[41] T. Lahmer and E. Rafajlowicz. On the optimality of harmonic excitation as input signals for the characterization of parameters in coupled piezoelectric and poroelastic problems. *Mechanical Systems and Signal Processing*, 90:399 – 418, 2017.

[42] S. Larsson and V. Thomee. *Partial differential equations with numerical methods*, volume 45. Springer Berlin Heidelberg, 2008.

[43] R. Lerch. Simulation of piezoelectric devices by two- and three-dimensional finite elements. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 37(3):233–247, 1990.

[44] G. Leugering, A. A. Novotny, G. P. Menzala, and J. Sokołowski. Shape sensitivity analysis of a quasi-electrostatic piezoelectric system in multilayered media. *Mathematical Methods in the Applied Sciences*, 33(17):2118–2131, 2010.

[45] S. Li, L. Zheng, W. Jiang, R. Sahul, V. Gopalan, and W. Cao. Characterization of full set material constants of piezoelectric materials based on ultrasonic method and inverse impedance spectroscopy using only one sample. *Journal of applied physics*, 114(10):104505, 2013.

[46] W. P. Mason. *Piezoelectric crystals and their application to ultrasonics.* Van Nostrand, 1950.

[47] W. P. Mason. Piezoelectricity, its history and applications. *The Journal of the Acoustical Society of America*, 70(6):1561–1566, 1981.

[48] MATLAB. *version 9.1.0 (R2016b).* The MathWorks Inc., Natick, Massachusetts, USA, 2016.

[49] V. N. Melnik. Existence and Uniqueness Theorems of the Generalized Solution for a Class of Non-Stationary Problems of Coupled Electroelasticity. *Soviet Math. (Iz. VUZ)*, 35:24–32, 1991.

[50] D. Meschede. *Gerthsen Physik.* Springer-Verlag, 2015.

[51] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry, A. R. Terrel, Š. Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman, and A. Scopatz. SymPy: symbolic computing in Python. *PeerJ Computer Science*, 3:e103, January 2017.

[52] J. Nitsche. Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 36(1):9–15, Jul 1971.

[53] J. Nocedal and S. Wright. *Numerical Optimization.* Springer Series in Operations Research and Financial Engineering. Springer New York, 2006.

[54] N. Perez, M. A. B. Andrade, F. Buiochi, and J. C. Adamowski. Identification of elastic, dielectric, and piezoelectric constants in piezoceramic disks. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 57(12), 2010.

[55] N. Perez, F. Buiochi, M. A. B. Andrade, and J. C. Adamowski. Numerical characterization of piezoceramics using resonance curves. *Materials*, 9(2):71, 2016.

[56] M. J. D. Powell. Software by Professor M.J.D. Powell. http://mat.uc.pt/%7Ezhang/software.html. [Online; accessed 10-June-2018].

[57] M. J. D. Powell. On fast trust region methods for quadratic models with linear constraints. *Mathematical Programming Computation*, 7(3):237–267, 2015.

[58] F. Pukelsheim. *Optimal Design of Experiments*. Wiley Series in Probability and Statistics. Wiley, 1993.

[59] L. Rall. *Automatic Differentiation: Techniques and Applications*. Lecture Notes in Computer Science. Springer Berlin, 1981.

[60] J. N. Reddy. *An introduction to continuum mechanics*. Cambridge University Press, 2013.

[61] S. J. Rupitsch. *Characterization of Sensor and Actuator Materials*, pages 127–193. Springer Berlin Heidelberg, Berlin, Heidelberg, 2019.

[62] S. J. Rupitsch and J. Ilg. Complete characterization of piezoceramic materials by means of two block-shaped test samples. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 62(7):1403–1413, July 2015.

[63] S. J. Rupitsch, A. Sutor, J. Ilg, and R. Lerch. Identification procedure for real and imaginary material parameters of piezoceramic materials. In *2010 IEEE International Ultrasonics Symposium*, pages 1214–1217. IEEE, 2010.

[64] M. K. Samal, P. Seshu, and B. K. Dutta. An analytical formulation in 3d domain for the nonlinear response of piezoelectric slabs under weak electric fields. *International Journal of Solids and Structures*, 44(14):4656 – 4672, 2007.

[65] V. Schulze. Analyse und FEM-basierte Simulation instationärer piezoelektrischer Differentialgleichungen. Master's thesis, Paderborn University, 2017.

[66] B. Schweizer. *Partielle Differentialgleichungen: Eine anwendungsorientierte Einführung*. Springer-Verlag, 2013.

[67] R. Simkovics. *Nichtlineares piezoelektrisches Finite-Elemente-Verfahren zur Modellierung piezokeramischer Aktoren.* PhD thesis, University of Erlangen-Nuremberg, 2002.

[68] R. C. Smith. *Smart material systems: model development.* SIAM, 2005.

[69] Standards Committee of the IEEE Ultrasonics, Ferroelectrics, and Frequency Control Society. IEEE Standard on Piezoelectricity, 1987. ANSI/IEEE Standard 176-1987, 1988.

[70] E. M. Stein. *Singular integrals and differentiability properties of functions (PMS-30).* Princeton University Press, 2016.

[71] P. Steinhorst. *Anwendung adaptiver FEM für piezoelektrische und spezielle mechanische Probleme.* PhD thesis, Technical University of Chemnitz, 2009.

[72] J. Tichỳ, J. Erhart, E. Kittinger, and J. Prívratská. *Fundamentals of Piezoelectric Sensorics: Mechanical, Dielectric, and Thermodynamical Properties of Piezoelectric Materials.* Springer Berlin Heidelberg, 2010.

[73] M. Ulbrich and S. Ulbrich. *Nichtlineare Optimierung.* Springer-Verlag, 2012.

[74] C. Unverzagt. *Sensitivitätssteigerung durch Elektrodenmodifikation für die Materialparameterbestimmung von Piezokeramiken.* PhD thesis, Paderborn University, 2018.

[75] C. Unverzagt, J. Rautenberg, and B. Henning. Sensitivitätssteigerung bei der inversen Materialparameterbestimmung für Piezokeramiken. *tm-Technisches Messen*, 82(2):102–109, 2015.

[76] C. Unverzagt, J. Rautenberg, B. Henning, and K. Kulshreshtha. Modified electrode shape for the improved determination of piezoelectric material parameters. In Gan Woon Siong, Lim Siak Piang, and Khoo Boo Cheong, editors, *Proceedings of the 2013 International Congress on Ultrasonics*, pages 758–763, Singapore, 2013. Research Publishing.

[77] W. Voigt. *Lehrbuch der Kristallphysik (mit Ausschluß der Kristalloptik).* Teubner, 1910.

[78] A. Wächter and L. T. Biegler. On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming. *Mathematical Programming*, 106(1):25–57, 2006.

[79] S. Walter. *Structured higher-order algorithmic differentiation in the forward and reverse mode with application in optimum experimental design.* PhD thesis, Humboldt University of Berlin, Mathematisch-Naturwissenschaftliche Fakultät II, 2012.

[80] A. Walther and A. Griewank. ADOL-C project homepage. https://projects.coin-or.org/ADOL-C. [Online; accessed 3-September-2018].

[81] A. Walther and A. Griewank. Getting started with ADOL-C. In U. Naumann and O. Schenk, editors, *Combinatorial scientific computing*, Chapman & Hall/CRC Computational Science, pages 181–202. Taylor & Francis, 2012.

[82] F. Wein. *Topology Optimization of Smart Piezoelectric Transducers.* PhD thesis, University of Erlangen-Nuremberg, 2011.

A Appendix

A.1 Derivatives of Network Equations

Now, let $d(\cdot)$ denote the partial derivative $\frac{\partial(\cdot)}{\partial\xi}$ for a quantity $(\cdot)$ and for a dependent variable ξ. Then by using the chain rule we can also compute the following formal derivatives:

$$dZ_a^\Delta = \frac{-2\cdot\left(\frac{-dZ_1}{Z_1^2} + \frac{dZ_2}{Z_2^2} - \frac{dZ_3}{Z_3^2}\right)}{\left(\frac{1}{Z_1} - \frac{1}{Z_2} + \frac{1}{Z_3}\right)^2}$$

$$dZ_b^\Delta = \frac{-2\cdot\left(\frac{-dZ_1}{Z_1^2} - \frac{dZ_2}{Z_2^2} + \frac{dZ_3}{Z_3^2}\right)}{\left(\frac{1}{Z_1} + \frac{1}{Z_2} - \frac{1}{Z_3}\right)^2}$$

$$dZ_c^\Delta = \frac{-2\cdot\left(\frac{dZ_1}{Z_1^2} - \frac{dZ_2}{Z_2^2} - \frac{dZ_3}{Z_3^2}\right)}{\left(\frac{-1}{Z_1} + \frac{1}{Z_2} + \frac{1}{Z_3}\right)^2}$$

$$dZ_a^Y = \frac{Z_b^\Delta\cdot(Z_a^\Delta\cdot dZ_c^\Delta - Z_c^\Delta\cdot dZ_a^\Delta) + Z_c^\Delta\cdot(Z_a^\Delta + Z_c^\Delta)\cdot dZ_b^\Delta + (Z_b^\Delta)^2\cdot dZ_c^\Delta}{(Z_a^\Delta + Z_b^\Delta + Z_c^\Delta)^2}$$

$$dZ_b^Y = \frac{Z_b^\Delta\cdot(Z_c^\Delta\cdot dZ_a^\Delta + Z_a^\Delta\cdot dZ_c^\Delta) + (Z_c^\Delta)^2\cdot dZ_a^\Delta - Z_a^\Delta\cdot Z_c^\Delta\cdot dZ_b^\Delta + (Z_a^\Delta)^2\cdot dZ_c^\Delta}{(Z_a^\Delta + Z_b^\Delta + Z_c^\Delta)^2}$$

$$dZ_c^Y = \frac{Z_b^\Delta \cdot (Z_c^\Delta \cdot dZ_a^\Delta - Z_a^\Delta \cdot dZ_c^\Delta) + (Z_b^\Delta)^2 \cdot dZ_a^\Delta + Z_a^\Delta \cdot (Z_a^\Delta + Z_c^\Delta) \cdot dZ_b^\Delta}{(Z_a^\Delta + Z_b^\Delta + Z_c^\Delta)^2}$$

Let $dZ_{abc} := dZ_a^\Delta + dZ_b^\Delta + dZ_c^\Delta$, $Z_{abc} := Z_a^\Delta + Z_b^\Delta + Z_c^\Delta$ and $PZ_i^\Delta := Z_a^\Delta \cdot Z_b^\Delta \cdot Z_c^\Delta$

$$
\begin{aligned}
d\tilde{Z}_a^Y = &- \frac{R_0 \cdot Z_b^\Delta \cdot Z_c^\Delta \cdot (R_0 \cdot dZ_{abc} + Z_b^\Delta \cdot dZ_a^\Delta + Z_a^\Delta \cdot dZ_b^\Delta + Z_c^\Delta \cdot dZ_b^\Delta + Z_b^\Delta \cdot dZ_c^\Delta)}{(R_0 \cdot Z_{abc} + Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta)^2} \\
&+ \frac{R_0 \cdot Z_c^\Delta \cdot dZ_b^\Delta}{R_0 \cdot Z_{abc} + Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta} \\
&+ \frac{R_0 \cdot Z_b^\Delta \cdot dZ_c^\Delta}{R_0 \cdot Z_{abc} + Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta}
\end{aligned}
$$

$$
\begin{aligned}
d\tilde{Z}_b^Y = &\frac{Z_b^\Delta \cdot Z_c^\Delta \cdot dZ_a^\Delta + R_0 \cdot Z_c^\Delta \cdot dZ_a^\Delta + Z_a^\Delta \cdot Z_c^\Delta \cdot dZ_b^\Delta + Z_a^\Delta \cdot Z_b^\Delta \cdot dZ_c^\Delta + R_0 \cdot Z_a^\Delta \cdot dZ_c^\Delta}{R_0 \cdot Z_{abc} + Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta} \\
&- \frac{(PZ_i^\Delta + R_0 \cdot Z_a^\Delta \cdot Z_c^\Delta) \cdot (R_0 \cdot dZ_{abc} + Z_b^\Delta \cdot dZ_a^\Delta + Z_a^\Delta \cdot dZ_b^\Delta + Z_c^\Delta \cdot dZ_b^\Delta + Z_b^\Delta \cdot dZ_c^\Delta)}{(R_0 \cdot Z_{abc} + Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta)^2}
\end{aligned}
$$

$$
\begin{aligned}
d\tilde{Z}_c^Y = &\frac{-R_0 \cdot Z_a^\Delta \cdot Z_b^\Delta \cdot (R_0 \cdot dZ_{abc} + Z_b^\Delta \cdot dZ_a^\Delta + Z_a^\Delta \cdot dZ_b^\Delta + Z_c^\Delta \cdot dZ_b^\Delta + Z_b^\Delta \cdot dZ_c^\Delta)}{(R_0 \cdot Z_{abc} + Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta)^2} \\
&+ \frac{R_0 \cdot Z_b^\Delta \cdot dZ_a^\Delta}{R_0 \cdot Z_{abc} + Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta} \\
&+ \frac{R_0 \cdot Z_a^\Delta \cdot dZ_b^\Delta}{R_0 \cdot Z_{abc} + Z_a^\Delta \cdot Z_b^\Delta + Z_b^\Delta \cdot Z_c^\Delta}
\end{aligned}
$$

$$dZ_{vp} = \frac{d\tilde{Z}_b^Y \cdot (\tilde{Z}_c^Y + Z_{v2})^2 + (\tilde{Z}_b^Y + Z_{v1})^2 \cdot d\tilde{Z}_c^Y}{(\tilde{Z}_b^Y + \tilde{Z}_c^Y + Z_{v1} + Z_{v2})^2}$$

$$dZ_{ges} = d\tilde{Z}_a^Y + dZ_{vp};$$

Remark A.1. The variable of which respect the derivative should be computed is intentionally left as general as possible. Usually, this variable will be one of the material parameters. However, for gradient based optimization the variable could also be a variable describing the electrode geometry.

A.2 Prerequisites for a Comparison to FEniCS

FEniCS [3] is an open-source finite element tool for python which allows handling and solving PDEs with *relative* ease. Moreover, with dolfin-adjoint [16] a built-in AD tool, it is also possible to compute derivatives in forward and reverse mode (i.e. tangents and adjoints) whereas in this thesis only tangents can be computed, see Appendix A.3. A comparison and verification of the simulation results of a piezoceramic obtained by FEniCS using dolfin-adjoint and CFS++ using ADOL-C is interesting and worthwhile. In the following section some important prerequisites for a comparison are given.

A.2.1 Real and Imaginary Equations

The weak formulation of the piezoelectric PDE in time domain can be solved numerically by using time-stepping schemes such as the Crank-Nicolson or Newmark method. The implementation of these time-stepping methods in FEniCS have been investigated in [65]. Current implementations in FEniCS for the transient formulation are based on collaboration between V. Schulze, N. Feldmann, and the author of this thesis and results will be part of future publications. Note that some additional work is still required especially with regard to use of dolfin-adjoint analogously to issues raised in A.3.

However, the time-harmonic formulation requires some more intermediate steps before it can be implemented. The time-harmonic reformulation comes at the cost that the partial differential equation is now complex-valued and needs to be solved as such. In FEniCS it is currently not possible to directly handle complex-valued equations, hence Eq. (2.53) needs to be separately stated and solved in both real and imaginary parts while only using real valued quantities

$$\mathrm{Re}\left(-\rho\omega^2\hat{u} - \mathcal{B}^T(\tilde{c}^E\mathcal{B}\hat{u} + \tilde{e}^T\nabla\hat{\phi})\right) = 0 \text{ in } \Omega$$

$$\mathrm{Im}\left(-\rho\omega^2\hat{u} - \mathcal{B}^T(\tilde{c}^E\mathcal{B}\hat{u} + \tilde{e}^T\nabla\hat{\phi})\right) = 0 \text{ in } \Omega$$

$$\mathrm{Re}\left(-\nabla\cdot(\tilde{e}\mathcal{B}\hat{u} - \tilde{\epsilon}^S\nabla\hat{\phi})\right) = 0 \text{ in } \Omega$$

$$\mathrm{Im}\left(-\nabla\cdot(\tilde{e}\mathcal{B}\hat{u} - \tilde{\epsilon}^S\nabla\hat{\phi})\right) = 0 \text{ in } \Omega$$

where $\hat{u} = \hat{u}_r + j\hat{u}_i$ and $\hat{\phi} = \hat{\phi}_r + j\hat{\phi}_i$ designate the real and complex parts of $\hat{u}$ and $\hat{\phi}$. In order to achieve this, these equations need to be separated into their real and imaginary parts.

Using the notation $\langle u, v \rangle := \int_\Omega u^T \overline{v} \, d\Omega$ the weak form of Eq. (2.53) (see [38])

$$\int_\Omega \left(-\rho\omega^2 u^T \overline{v} + (\tilde{c}^E \mathcal{B} u)^T \overline{\mathcal{B}v} + (\tilde{e}^T \nabla\phi)^T \overline{\mathcal{B}v} + (\tilde{e}\mathcal{B}u)^T \overline{\nabla\eta} - (\tilde{\epsilon}^S \nabla\phi)^T \overline{\nabla\eta} \right) d\Omega = 0$$

with test functions v and η can be abbreviated as

$$W := \left\langle -\rho\omega^2 u, v \right\rangle + \left\langle \tilde{c}^E \mathcal{B}u, \mathcal{B}v \right\rangle + \left\langle \tilde{e}^T \nabla\phi, \mathcal{B}v \right\rangle + \left\langle \tilde{e}\mathcal{B}u, \nabla\eta \right\rangle - \left\langle \tilde{\epsilon}^S \nabla\phi, \nabla\eta \right\rangle .$$

We now expand all summands by their real and imaginary parts. For two complex quantities $w = w_r + jw_i$, $y = y_r + jy_i$ we have

$$\langle w, y \rangle = w^T \overline{y} = \langle w_r, y_r \rangle + \langle w_i, y_i \rangle + j \left(\langle w_i, y_r \rangle - \langle w_r, y_i \rangle \right).$$

Hence, we get

$$\text{Re} \left\langle -\rho\omega^2 u, v \right\rangle = \left\langle -\rho\omega^2 u_r, v_r \right\rangle + \left\langle -\rho\omega^2 u_i, v_i \right\rangle$$

$$\text{Im} \left\langle -\rho\omega^2 u, v \right\rangle = \left\langle -\rho\omega^2 u_i, v_r \right\rangle - \left\langle -\rho\omega^2 u_r, v_i \right\rangle$$

For three complex quantities $c = c_r + jc_i$, $w = w_r + jw_i$, $y = y_r + jy_i$ we have

$$\text{Re} \left\langle cw, y \right\rangle = \text{Re} \left((cw)^T \overline{y} \right) = c_r w_r y_r + c_r w_i y_i + c_i w_r y_i - c_i w_i y_r$$

$$\text{Im} \left\langle cw, y \right\rangle = \text{Im} \left((cw)^T \overline{y} \right) = -c_r w_r y_i + c_r w_i y_r + c_i w_r y_r + c_i w_i y_i$$

Analogously we get for the remaining summands e.g.

$$\text{Re} \left\langle \tilde{c}^E \mathcal{B}u, \mathcal{B}v \right\rangle = \left\langle \tilde{c}_r^E \mathcal{B}u_r, \mathcal{B}v_r \right\rangle + \left\langle \tilde{c}_r^E \mathcal{B}u_i, \mathcal{B}v_i \right\rangle + \left\langle \tilde{c}_i^E \mathcal{B}u_r, \mathcal{B}v_i \right\rangle - \left\langle \tilde{c}_i^E \mathcal{B}u_i, \mathcal{B}v_r \right\rangle$$

$$\text{Im} \left\langle \tilde{c}^E \mathcal{B}u, \mathcal{B}v \right\rangle = -\left\langle \tilde{c}_r^E \mathcal{B}u_r, \mathcal{B}v_i \right\rangle + \left\langle \tilde{c}_r^E \mathcal{B}u_i, \mathcal{B}v_r \right\rangle + \left\langle \tilde{c}_i^E \mathcal{B}u_r, \mathcal{B}v_r \right\rangle + \left\langle \tilde{c}_i^E \mathcal{B}u_i, \mathcal{B}v_i \right\rangle$$

Hence, the weak form W can be rewritten as

$$W = W_r + jW_i$$

with

$$W_r := \left\langle -\rho\omega^2 u_r, v_r \right\rangle + \left\langle -\rho\omega^2 u_i, v_i \right\rangle$$
$$+ \left\langle \tilde{c}_r^E \mathcal{B}u_r, \mathcal{B}v_r \right\rangle + \left\langle \tilde{c}_r^E \mathcal{B}u_i, \mathcal{B}v_i \right\rangle + \left\langle \tilde{c}_i^E \mathcal{B}u_r, \mathcal{B}v_i \right\rangle - \left\langle \tilde{c}_i^E \mathcal{B}u_i, \mathcal{B}v_r \right\rangle$$
$$+ \left\langle \tilde{e}_r^T \nabla\phi_r, \mathcal{B}v_r \right\rangle + \left\langle \tilde{e}_r^T \nabla\phi_i, \mathcal{B}v_i \right\rangle + \left\langle \tilde{e}_i^T \nabla\phi_r, \mathcal{B}v_i \right\rangle - \left\langle \tilde{e}_i^T \nabla\phi_i, \mathcal{B}v_r \right\rangle$$
$$+ \left\langle \tilde{e}_r \mathcal{B}u_r, \nabla\eta_r \right\rangle + \left\langle \tilde{e}_r \mathcal{B}u_i, \nabla\eta_i \right\rangle + \left\langle \tilde{e}_i \mathcal{B}u_r, \nabla\eta_i \right\rangle - \left\langle \tilde{e}_i \mathcal{B}u_i, \nabla\eta_r \right\rangle$$
$$- \left\langle \tilde{\epsilon}_r^S \nabla\phi_r, \nabla\eta_r \right\rangle - \left\langle \tilde{\epsilon}_r^S \nabla\phi_i, \nabla\eta_i \right\rangle - \left\langle \tilde{\epsilon}_i^S \nabla\phi_r, \nabla\eta_i \right\rangle + \left\langle \tilde{\epsilon}_i^S \nabla\phi_i, \nabla\eta_r \right\rangle$$

$$W_i := \left\langle -\rho\omega^2 u_i, v_r \right\rangle - \left\langle -\rho\omega^2 u_r, v_i \right\rangle$$
$$- \left\langle \tilde{c}_r^E \mathcal{B}u_r, \mathcal{B}v_i \right\rangle + \left\langle \tilde{c}_r^E \mathcal{B}u_i, \mathcal{B}v_r \right\rangle + \left\langle \tilde{c}_i^E \mathcal{B}u_r, \mathcal{B}v_r \right\rangle + \left\langle \tilde{c}_i^E \mathcal{B}u_i, \mathcal{B}v_i \right\rangle$$
$$- \left\langle \tilde{e}_r^T \nabla\phi_r, \mathcal{B}v_i \right\rangle + \left\langle \tilde{e}_r^T \nabla\phi_i, \mathcal{B}v_r \right\rangle + \left\langle \tilde{e}_i^T \nabla\phi_r, \mathcal{B}v_r \right\rangle + \left\langle \tilde{e}_i^T \nabla\phi_i, \mathcal{B}v_i \right\rangle$$
$$- \left\langle \tilde{e}_r \mathcal{B}u_r, \nabla\eta_i \right\rangle + \left\langle \tilde{e}_r \mathcal{B}u_i, \nabla\eta_r \right\rangle + \left\langle \tilde{e}_i \mathcal{B}u_r, \nabla\eta_r \right\rangle + \left\langle \tilde{e}_i \mathcal{B}u_i, \nabla\eta_i \right\rangle$$
$$+ \left\langle \tilde{\epsilon}_r^S \nabla\phi_r, \nabla\eta_i \right\rangle - \left\langle \tilde{\epsilon}_r^S \nabla\phi_i, \nabla\eta_r \right\rangle - \left\langle \tilde{\epsilon}_i^S \nabla\phi_r, \nabla\eta_r \right\rangle - \left\langle \tilde{\epsilon}_i^S \nabla\phi_i, \nabla\eta_i \right\rangle$$

Some further work is now required to break down terms involving the differential operator $\mathcal{B}$ into elemental statements using only basic algebraic operations. However, as the resulting terms are very lengthy (approx. 2500 characters for each the real and imaginary parts) a computer algebra system was used. Alternatively, in [13] the differential operator was defined as a basic operator for the differential equation. In this case the separation $W = W_r + jW_i$ with W_r, W_i as above is sufficient for implementation of the variational problem into FEniCS.

Remark A.2. The time-harmonic formulation allows the evaluation of impedance at discrete frequencies. Whereas the transient formulation only permits evaluations on a predetermined frequency bandwidth. Hence, for the solution of the inverse problem it could be advantageous to utilize this harmonic formulation instead of the transient one. Note that the ten complex-valued material parameters could still be transformed back into ten real valued parameters. Thus, in this case there may not be more computational effort required to solve the inverse problem than when utilizing a transient formulation. However, this case is not considered in this thesis and will be part of future investigation.

A.3 ADOL-C Traced Versus Traceless Mode

The author first used ADOL-C's traced forward mode for derivative calculation. The trace process with `trace_on` was started simultaneously with setting the first material parameter as dependent (as opposed to setting this in a different location i.e. the main class) and stopped with `trace_off` after the PDE has been solved for the last time step. However, a first test was canceled after about 10 h runtime and a 400 GB trace file as only very few time steps of the tracing process were completed. The tracing process is clearly influenced by available memory and memory consumption can only increase even more for the vast amount of remaining to be computed time steps. We then experimented with only tracing single time steps and reusing the created trace but ultimately could not yet find a suitable solution. Consider the following main time stepping loop for this simulation case: Firstly, note that the

```
1   for (actTimeStep_ = startStep; actTimeStep_ <= endStep_;
2        actTimeStep_ += 1, count++)
3      {
4        mathParser_->SetValue( MathParser::GLOB_HANDLER, "t", actTime_ );
5        mathParser_->SetValue( MathParser::GLOB_HANDLER, "dt", dt );
6        mathParser_->SetValue( MathParser::GLOB_HANDLER, "step", actTimeStep_ );
7        [...]
8        ptPDE_->GetSolveStep()->PreStepTrans();
9        ptPDE_->GetSolveStep()->SolveStepTrans();
10       ptPDE_->GetSolveStep()->PostStepTrans();
11       [...]
12     }
```

current time step is set as a variable for later usage in `string` formulas parsed by the `MathParser` also mentioned earlier. Secondly, in the lower part we notice the solution step methods `PreStepTrans()`, `SolveStepTrans()` and `PostStepTrans()` marked in gray. We point out that they are all invoked by the object `ptPDE_` containing information of the partial differential equation. As there are many possible partial differential equations which could be considered for this time stepping scheme and thus be solved by CFS++, `ptPDE_` is a base class object - it's real object type is only known at runtime. Therefore these methods are (non-callable) pure virtual methods in the base class, they have their callable implementation in their specialization only known at runtime. Moreover, we also note that all these methods do not take any arguments. To trace single time steps and/or reuse prerecorded traces we face the following issues:

- For tracing single steps we need to set the dependents again. However, in the first time step the material parameters are included in many computations of the initialization phase. These costly initializations, i.e., the LDL-factorization of the main system matrix would have to be repeated for every time step. Hence, further adjustments to decrease the cost of the initialization phase would be necessary.

- Reusing a previously recorded trace for a single time step is in principle a great idea: We record the operations of solution process on the trace and reevaluate this for a different input value - in our case for a different time step. However, this input value needs to be set in such a way that all objects depending on this value also are updated when a new input value is set. Unfortunately, none of the solution step methods mentioned above take any arguments. The current time step is stored implicitly in various other objects in the containing class and its related objects, i.e. `ptPDE_` or in the `Mathparser` object. Hence, there may be many objects which are influenced by the current time step which then also need to be updated.

It is possible to solve this issue by updating all variables influenced by the updated time step. However, as there is an unknown amount of those variables it was decided to save development time and use ADOL-C's traceless mode instead.

A.4 MuParser Applications

A full example C++ file which demonstrates a solution to handling MuParser applications is given:

```cpp
#include <iostream>
#include "muParser.h"
#include "adolc/adtl.h"

// Both variants in double and adouble are required!
adtl::adouble ADMySqr(adtl::adouble a_fVal)
{
  return a_fVal*a_fVal;
}
  double MySqr(double a_fVal)
{
  return (ADMySqr(adtl::adouble(a_fVal))).getValue();
```

```
13     // We can call the adouble variant! It is not necessary to convert all
14     // dependencies of the MySqr method to double. The typedef approach
15     // is still feasible.
16   }
17   // If we could not have called this method in adouble datatype variant
18   // from the double variant then we would have needed to convert all
19   // dependencies to double.
20   // In a typedef approach this can be extremely demanding as any
21   // dependency can be/is part of the original codebase.
22   // This would have given a huge amount of datatype mismatches in
23   // all parts of the codebase.
24   //
25   // We just need to define the double variant caller for the AD method:
26   double dMySqr(double a_fVal,double adval)
27   {
28     adtl::adouble tmp;  // define temporary adouble
29     tmp.setValue(a_fVal);
30     tmp.setADValue(0,adval);
31     adtl::adouble ret;
32     ret=ADMySqr(tmp);  // call the AD version of the method
33     return ret.getADValue(0);
34   }
35
36   int main()
37   {
38   using namespace mu;
39
40   try
41   {
42     adtl::adouble adtest=1.0;
43     adtest.setADValue(0,1.0);
44
45     // Get the pointer of the primal and derivative values.
46     // Note: Not part of standard ADOL-C 2.6.3.
47     double* tmp=adtest.getValuePtr();
48     double* tmp2=adtest.getADPtr(0);
49
50     // Make sure everything is setup correctly.
51     std::cout << "My test adouble is: "<< adtest << std::endl;
52
53     // And initialise:
54     adtl::adouble test=1.0;
55     double test2;
56     test2=(double)test;
57
58     double* fVal=tmp;
59     double* dfVal=tmp2;
60
61     Parser p;
62     p.DefineVar("a",fVal);
63     p.DefineFun("MySqr", MySqr);
```

```
64    p.SetExpr("MySqr(a)");
65
66    // Every time a parser registers an entity, then also register a second
          'derivative' version for AD purposes:
67    Parser dp;
68    dp.DefineVar("a",fVal);
69    dp.DefineVar("da",dfVal);
70
71    dp.DefineFun("dMySqr", dMySqr);
72    dp.SetExpr("dMySqr(a,da)"); // <-------- This critical string needs to be
          provided externally.
73
74    // Print the solution.
75    for (std::size_t a=0; a<10;++a)
76    {
77    std::cout << "Primal:␣␣" ;
78    std::cout << p.Eval()  << std::endl;
79    std::cout << "Tangent␣␣" ;
80    std::cout << dp.Eval() << std::endl;
81    }
82 }
83 catch (Parser::exception_type &e)
84 {
85    std::cout << e.GetMsg() << std::endl;
86 }
87 return 0;
88 }
89
```