# ROBUST SHAPE OPTIMIZATION OF ELECTROMECHANICAL ENERGY CONVERTERS

Vom Fachbereich Mathematik
der Technischen Universität Darmstadt
zur Erlangung des akademischen Grades eines
Doktors der Naturwissenschaften (Dr. rer. nat.)
genehmigte

**Dissertation**

von

**Björn Polenz, M.Sc.**
aus Seligenstadt

| | |
|---|---|
| Referent: | Prof. Dr. Stefan Ulbrich |
| Korreferent: | Prof. Dr. Sebastian Schöps |
| Tag der Einreichung: | 05.07.2023 |
| Tag der mündlichen Prüfung: | 27.10.2023 |

Darmstadt 2023

**Robust Shape Optimization of Electromechanical Energy Converters**

Autor: Björn Polenz, geb. in Seligenstadt

Vom Fachbereich Mathematik der Technischen Universität Darmstadt zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften (Dr. rer. nat.) genehmigte Dissertation, Darmstadt.

Jahr der Veröffentlichung der Dissertation auf TUprints: 2024

Tag der mündlichen Prüfung: 27.10.2023

*To Romy*

# Acknowledgments

First and foremost, I would like to thank my supervisor Prof. Dr. Stefan Ulbrich. When I first met him in the bachelor seminar I felt his passion and joy for his research area, which inspired me to delve into the field of optimization. I thank him for giving me the opportunity to become part of the optimization research group, where there was always a great working atmosphere, the possibility to participate in conferences and fun out of work activities. I would also like to thank him for his support and helpful discussions especially during the last year.

I also want to thank Prof. Dr. Sebastian Schöps for acting as a referee for this thesis and the collaboration and fruitful discussions within the PASIROM project. I thank Prof. Dr. Herbert De Gersem for providing the Matlab-code for the simulation of an induction machine.

Furthermore, I would like to thank Prof. Dr. Marc Pfetsch and Prof. Dr. Robert Haller for taking the time and being members of my examination committee.

Gratefully I acknowledge the financial support of the Bundesministerium für Bildung und Forschung (BMBF) through PASIROM.

I thank my former colleagues for the fun times playing Doppelkopf or going Frühschoppen: Anna Walter, Johann Schmitt, Kristina Goschin, Paloma Schäfer-Aguilar and Philip Kolvenbach. Also I thank Isabel Jacob, Marcel Steinhardt and Theodor Komann for the good times in the office and at conferences. A special thanks goes to Elisabeth Diehl for the shared coffee breaks talking about so many (ir)relevant topics.

I would like to thank my family for their steady support: Thank you for always being there for me no matter what. I am fortunate having you.

Thank you Johanna for your love, your unconditional support and for being patient with me. I am more than happy to have met you here and to know you by my side. With you, my heart has found its true home, forever grateful, never alone.

# Zusammenfassung

Diese Arbeit beschäftigt sich mit der Simulation und Formoptimierung von elektromechanischen Energiewandlern unter Unsicherheit. Genauer wird eine Asynchronmaschine betrachtet, deren elektromagnetischen Felder durch die magnetoquasistatische Approximation der Maxwell Gleichungen beschrieben werden kann, welche mit Netzwerkgleichungen für den Käfigläufer und für den anregenden Dreiphasenstrom gekoppelt werden. Komplettiert wird das Zustandssystem durch eine Bewegungsgleichung, welche durch das elektromagnetische Drehmoment angeregt wird. Dies führt auf ein System von partiell differential-algebraischen Gleichungen.

Zur numerischen Lösung der Zustandsgleichung wird ein Finite Elemente Ansatz mit einem Zeitschrittverfahren verwendet.

Wir betrachten Unsicherheiten in Material und Geometrie der Maschine und verwenden einen Worst-Case-Ansatz, um diesen Unsicherheiten zu begegnen. Dies führt auf ein zweistufiges Optimierungsproblem. Da diese Probleme numerisch schwierig zu lösen sind, verwenden wir Approximationen bis zur zweiten Ordnung als Ersatzmodell. Insbesondere verwenden wir Taylormodelle in Kombination mit einer adaptiven Strategie zur Verbesserung der Approximationsgüte und ableitungsfreie Interpolationsmodelle, die ebenfalls iterativ verbessert werden können.

Die Diskretisierung von partiellen Differentialgleichungen führt auf Systeme mit vielen Freiheitsgraden. Zusätzlich erhöht die Betrachtung von Unsicherheiten in der Optimierung den Berechnungsaufwand. Um unsere Berechnungen zu beschleunigen, verwenden wir Techniken zur Modellreduktion.

This work deals with the simulation and shape optimization of electromechanical energy converters under uncertainty. More precisely, an asynchronous machine is considered, whose electromagnetic fields can be described by the magnetoquasistatic approximation of Maxwell's equations, which are coupled with network equations for the rotor cage and for the exciting three-phase current. The state system is

completed by an equation of motion which is excited by the torque. This leads to a system of partial differential algebraic equations.

A finite element approach with a time-stepping method is used to numerically solve the equation numerically.

We consider uncertainties in the material and geometry of the machine and use a worst-case approach to address these uncertainties. This leads to a bi-level structured optimization problem. Since these problems are difficult to solve numerically, we use approximations up to second order as surrogate models. In particular, we use Taylor models in combination with an adaptive strategy to improve the approximation quality and derivative-free interpolation models that can also be improved iteratively.

Both the problem formulation and the consideration of uncertainty in the optimization lead to a high computational cost. To speed up our computations, we apply model dimension reduction techniques.

# Contents

CHAPTER 1

# Introduction

## 1.1 Motivation

Nowadays, electric motors can be found in many areas. For example, they are installed in everyday appliances such as kitchen appliances, vacuum cleaners or hair dryers and tools such as drills or electric saws. They are also used in means of transport: trains are often equipped with electric motors and increasingly cars and bicycles. In order to design the right motor for each product, prototypes have to be developed and tested. The production of classical physical prototypes is time-, money- and resource-intensive. With increasing computing power, the development of physical prototypes is shifting towards digital prototypes. With the help of these digital prototypes, not only the design can be developed, but, through the mathematical description of the underlying physical properties, often including partial differential equations, the simulations and optimizations can be performed on the computer. Multi-physical phenomena can be simulated, different objectives can be investigated and thus products can be developed for a customer individually.

## 1.2 Related Work

As the mathematical description of an induction machine involves many physical laws, one can find existence and uniqueness results which neglect different effects:

In Langer et al. [4] a proof of the existence for the 3D eddy current equation with a nonlinear relation between the magnetic field and the magnetic induction in the conducting domain and a linear relation in the non-conducting region using

a Schur complement can be found. Nicaise and Tröltzsch [63] examine the same equation with a linear material relation but coupled with an equation accounting for magnetic induction in a given region where they transform the equations into an integro-differential system and prove existence and uniqueness using a theorem by Showalter on degenerate parabolic equations. In a recent paper [23] by Chill et al. the well-posedness of the regularity of solutions to the same system as in [63] but with non-linear material behavior and weaker conditions on the domain is investigated, where gradient systems on Hilbert spaces and $\epsilon$-gradients are used in the analysis. An investigation of the magnetoquasistatic approximation of Maxwell's equations involving rotation of the inner domain in 2D with linear material can be found in [20]. Optimization with partial differential equations is an active field of research [47, 80].

The spatial discretization method used in this thesis is the finite element method, see [24, 16]. However, recently many authors make us of isogeometric analysis [48], which uses non-rational B-splines to describe the geometry, which has the advantage, that the airgap, which is a circle, can be represented precisely [58, 14].

Model order reduction for affinely parametrized equations or domains as it is used in this thesis is well studied. In [72] Rozza et al. examine the reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic partial differential equations. Grepl and Patera [43] use the same reduced basis approach, deducing a posteriori error bounds for parabolic equations. The reduced basis approach uses a greedy selection procedure of solution snapshots to generate a reduced order model. Another widely used approach to construct a reduced order model is the proper orthogonal decomposition (POD) method, where the solution snapshots are used to compute basis vectors which minimize the squared distance between the snapshots and their projection onto the low dimension space spanned by the basis vectors. A good survey on POD can be found in [82]. POD was applied for example in the simulation and optimization of permanent magnet synchronous machines, see [56, 15] and in several other engineering applications [13, 44, 55]. If the underlying equation involves nonlinearities, POD is not efficient, since the nonlinearities have to be evaluated in full dimension. To overcome this ineffeciency Chaturantabut and Sorensen [21, 22] proposed the discrete empirical interpolation method (DEIM). Its idea is to approximate the image of the nonlinearity by a linear subspace and then use this linear subspace to interpolate the full nonlinearity by the evaluation of a subset of its components. Drmac and Saibaba [38] examined the selection procedure of the components to evaluate by a strong rank revealing QR decomposition to lower the DEIM error bound. Applications of DEIM can be found in [55] for the simulation of lithium ion batteries and in [50] for the numerical solution of nonlinear magneto-quasistatic equations.

There are different approaches to dealing with uncertainties. In this thesis, we follow the worst-case minimization approach. In this approach, the constraint functions must be satisfied, regardless of the realization of the uncertain variables. The first works where the robust counterpart for parameterized optimization problems with a linear objective function and convex feasible set were introduced and investigated are by Ben-Tal and Nemirovski [7, 8]. In these papers the term robust optimization is used and coined due to the investigation of the robust counterpart. Dealing with nonlinear functions, Diehl, Bock and Kostina [37] examined the use of linearization in the uncertain parameter, which has to lie inside a ball, described by a scaled Hölder norm. This approach has the benefit that the worst-case can be computed analytically. Sichau and Ulbrich [78] investigated a nonlinear PDE constrained problem, where the objective and constraint functions are replaced by quadratic approximations and the state equation by a linear approximation. This leads to quadratic inner problems, which cannot be solved analytically, but the optimality conditions are known and added as constraints to the optimization problem, which results in a mathematical program with complementarity constraints. In the recent works of Lass and Ulbrich [56] and Kolvenbach, Lass and Ulbrich [52] also the curvature of the state equation is taken into account and an algorithm to iteratively move the expansion point of the Taylor model to improve its approximation of the worst-case is presented. Kolvenbach [51] and Kolvenbach, Lass and Ulbrich [52] also examine second order models but do not add the optimality conditions to the optimization problem, but rewrite it with so called reduced worst-case functions, which evaluate the worst-case. These worst-case functions are non-smooth which has to be taken into account in their analysis and the solution of the robust optimization problems.

Another approach to bi-level optimization where no derivative information is needed is examined by Conn and Vicente in [28]. They use a quadratic interpolation model as a surrogate model. The interpolation model uses less interpolation points than needed to uniquely define it, by requiring that it has minimal curvature, a technique described by Powell [67] which lowers the amout of possibly expensive function evaluations.

## 1.3 Outline of the thesis

In this work, we study the simulation and shape optimization of electrical machines. More precisely, we consider asynchronous machines, which are also called induction machines. In contrast to permanent magnet synchronous machines, these machines

do not have magnets in the rotor and thus, in the first place, no magnetic field in the rotor. Instead, they have a so-called rotor cage (also called squirrel cage), which contains short-circuited conductive bars. Due to the position of three windings in the stator and the use of a three-phase current, a rotating magnetic field is formed when the currents are applied to the windings. As the rotating field moves across the short-circuited bars, a voltage and a current is induced in the rotor bars, which in turn generates a magnetic field by magnetic induction, which is then pulled along by the external rotating magnetic field. If the rotor rotates at the same speed as the external rotating magnetic field, the rotor bars no longer experience a changing magnetic field and the currents induced in the rotor bars become smaller and the rotor rotates slower. Since the rotor must therefore always rotate slower than the externally rotating magnetic field, the motor is called asynchronous machine.

In Chapter 3 the physical basics for the description of the behavior of the asynchronous machine are laid and a proof for the existence of a solution of the derived state equation is given.

In the fourth chapter, methods are described with which the simulation and shape optimization of an asynchronous machine can be carried out on the computer. The chapter begins with the description of the discretization in space by the finite element method using a domain decomposition method, where the domain is subdivided into rotor and stator, which are then coupled by interpolation. In our optimization, we want to optimize the width and height of the rotor bars. To perform the transformation of the domain for different designs, we use an affine parametrization, which allows us to compute quantities, such as stiffness and mass matrix on a reference domain and then use a parameter dependent coefficient to assemble the quantities for a specific design. As some of the material properties of the motor depend on the magnitude of the underlying magnetic flux density, we describe how we treat the nonlinearity. At the end of the chapter, we describe the time stepping scheme we are using and how we compute adjoint state and sensitivities, which are used in the optimization to calculate descent directions.

Since the discretization of partial differential-algebraic equations leads to a system with many degrees of freedom, methods for model order reduction are discussed in the fifth chapter. We focus on the proper orthogonal decomposition method (POD) and how we use it in the simulation and optimization of the asynchronous machine. Since the POD method is not efficient in the presence of nonlinearities, we describe the use of the discrete empirical interpolation method for the simulation of the induction machine to shorten the simulation time.

There are many approaches to treat uncertainties in mathematical models. For example, uncertainties can be treated stochastically by their expected value and or variance, assuming knowledge of the underlying probability distribution of the

uncertainty. In other approaches, the probability distribution itself is part of the uncertainty. In this thesis, we use so-called worst-case optimization, which is described in the sixth chapter. In worst-case optimization, we assume, that the uncertainty lies inside a known ellipsoidal set. The objective is then to minimize the worst possible case under the requirement that the constraint functions are also fulfilled in their worst possible case. This is the most conservative way of dealing with uncertainty. However, in some application it is necessary that an engine is still rotating in the worst case and not only in the expected value.

In the seventh chapter, the optimization problem we are solving numerically is introduced, which states the minimization of the Joule losses in the rotor bars under the constraint, that a given torque is preserved and the results of the application of the methods described in the previous chapters are presented.

CHAPTER 2

# Mathematical Background

**Definition 2.1.** *A measurable function $f : \Omega \to \mathbb{R}$, for $\Omega \subset \mathbb{R}^n$ open, is called locally integrable if for all $K \subset \Omega$ compact,*

$$\int_K |f(x)| \, \mathrm{d}x < \infty.$$

*The set of locally integrable functions on $\Omega$ is a vector space denoted by $\mathcal{L}^1_{loc}(\Omega)$. With $L^1_{loc}(\Omega)$ we denote the equivalence classes we obtain, when we identify two function $f, g \in \mathcal{L}^1_{loc}(\Omega)$, if $f = g$ almost everywhere.*

**Definition 2.2.** *Let $\Omega \subset \mathbb{R}^n$ be an open subset and $f \in L^1_{loc}(\Omega)$ a locally integrable function. If there exists a function $g \in L^1_{loc}(\Omega)$ for which*

$$\int_\Omega g\phi \, \mathrm{d}x = (-1)^{|\alpha|} \int_\Omega f D^\alpha \phi \, \mathrm{d}x,$$

*holds for all $\phi \in C^\infty_c(\Omega)$, we call $D^\alpha f := g$ the $\alpha$-th weak partial derivative of $f$.*

In the definition $C^\infty_c(\Omega)$ is the set of infinitely differentiable functions with compact support in $\Omega$ and the multi-index $\alpha = (\alpha_1, \ldots, \alpha_N) \in (\mathbb{N} \cup \{0\})^N$ should be understood in the way that

$$|\alpha| := \sum_{i=1}^n \alpha_i, \;\; D^\alpha f := \frac{\partial^\alpha f}{\partial^{\alpha_1}_{x_1} \cdots \partial^{\alpha_n}_{x_n}}.$$

**Definition 2.3.** *For an open and bounded domain $\Omega \subset \mathbb{R}^n$ we say that it has $C^{k,\beta}$-boundary, where $k \in \mathbb{N}$ and $\beta \in [0, 1]$, if for all $x \in \partial\Omega$ there exists $r > 0$,*

*$l \in \{1, \dots, n\}$, $\sigma \in \{-1, +1\}$, and a function $\gamma \in C^{k,\beta}(\mathbb{R}^{n-1})$, such that*

$$\Omega \cap B(c, r) = \{y \in B(x, r) : \sigma y_l < \gamma(y_1, \dots, y_{l-1}, y_{l+1}, \dots, y_n)\}.$$

*Here, $B(x, r)$ denotes the open ball with midpoint $x$ and radius $r$. For the special case $k = 0$ and $\beta = 1$, we also say the domain has Lipschitz-boundary.*

In the definition we have the space

$$C^{k,\beta}(\bar{\Omega}) := \left\{ f \in C^k(\bar{\Omega}) : \sup \left\{ \frac{|D^\alpha f(x) - D^\alpha f(y)|}{\|x - y\|^\beta} : x, y \in \bar{\Omega}, x \neq y \right\} < \infty \text{ for } |\alpha| = k \right\}.$$

**Definition 2.4.** *We call the set of functions $f \in L^p(\Omega)$ on open sets $\Omega \subset \mathbb{R}^n$ Sobolev space $W^{m,p}(\Omega)$ if the weak derivative exists up to order $m$*

$$W^{m,p} := \{f \in L^p(\Omega) : D^\alpha f \in L^p(\Omega), \forall |\alpha| \leq m\}.$$

*Equipped with norms*

$$\|f\|_{W^{m,p}(\Omega)} := \left( \sum_{|\alpha| \leq m} \|D^\alpha f\|^p_{L^p(\Omega)} \right)^{1/p} \quad \text{for } 1 \leq p < \infty \text{ and}$$

$$\|f\|_{W^{m,\infty}(\Omega)} := \max_{|\alpha| \leq m} \|D^\alpha f\|_{L^\infty(\Omega)},$$

*the Sobolev spaces define Banach spaces.*

For the case $p = 2$ the Sobolev spaces are Hilbert spaces and we denote $H^m(\Omega) := W^{m,2}(\Omega)$ and the inner product is defined as

$$(f, g)_{H^m(\Omega)} := \sum_{|\alpha| \leq m} (D^\alpha f, D^\alpha g)_{L^2(\Omega)}.$$

**Theorem 2.5** (Trace operator)**.**
*Let $\Omega \subset \mathbb{R}^n$ open and bounded with Lipschitz-boundary and $1 \leq p \leq \infty$. Then there exists exactly one linear bounded operator*

$$S : W^{1,p}(\Omega) \to L^p(\Omega) \quad (\textit{trace operator})$$

*such that*

$$Su = u|_{\partial\Omega} \text{ for } u \in W^{1,p} \cap C^0(\overline{\Omega}).$$

*Proof.* See for example [1]. □

**Lemma 2.6** (Hölder's inequality)**.**
*Let $p, p' \in [1, \infty]$ with $\frac{1}{p} + \frac{1}{p'} = 1$. For $f \in L^p(\Omega)$ and $g \in L^{p'}(\Omega)$, we have $fg \in L^1(\Omega)$ and*

$$\|fg\|_{L^1(\Omega)} \leq \|f\|_{L^p} \|g\|_{L^{p'}}.$$

*Proof.* See for example [1]. □

**Theorem 2.7** (Young's inequality)**.**
*If $a \geq 0$ and $b \geq 0$ are nonnegative real numbers and if $p > 1$ and $q > 1$ with $\frac{1}{p} + \frac{1}{q} = 1$, then*

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

*Equality holds if and only if $a^p = b^q$.*

**Definition 2.8.** *The space $L(X, \mathbb{R})$ of continuous and linear functionals on a normed space $X$ is called the dual space of $X$ and denoted by $X^*$. We use the notation $\langle v, u \rangle_{X^*,X} = v(u)$ for $v \in X^*$ and $u \in X$.*

The space $X^*$ endowed with the norm $\|v\|_{X^*} = \sup_{\|u\|_X = 1} |\langle v, u \rangle_{X^*,X}|$ is a Banach space, see [47].

**Definition 2.9.** *Let $H_1$ and $H_2$ be Hilbert spaces and $T \in L(H_1, H_2)$. Then the adjoint operator of $T$ in a Hilbert space sense is denoted by $T^*$ and defined by*

$$(T(x), y)_{H_2} = (x, T^*(y))_{H_1} \ \forall x \in H_1, y \in H_2.$$

**Theorem 2.10** (Variation of constants)**.**
*Let the ordinary differential equation (ODE)*

$$u'(t) = au(t) + b(t),\ t \in I := [t_0, t_0 + T],$$

*with $a \in \mathbb{R}$ and the continuous function $b : I \to \mathbb{R}$. The solution to the ODE has the form (given the initial condition $u(t_0) = u_0$)*

$$u(t) = e^{a(t-t_0)} u_0 + \int_{t_0}^{t} e^{a(t-\tau)} b(\tau)\, \mathrm{d}\tau.$$

*Proof.* See for example [68]. □

**Theorem 2.11** (Picard/Lindelöf)**.**

*Let $X$ be a Banach space and let $Q \subseteq \mathbb{R} \times X$ for given $t_0 \in \mathbb{R}$ and $p_0 \in X$, $a, b > 0$ be defined as*

$$Q := \{(t, y) \in \mathbb{R} \times X : |t - t_0| \leq a, \|y - p_0\|_X \leq b\}.$$

*Let the function $f : Q \to X$ be continuous in $Q$ and Lipschitz-continuous wrt. to the second variable. Then there exists a unique continuous solution of*

$$\begin{aligned} x'(t) &= f(t, x(t)), \\ x(t_0) &= p_0, \end{aligned}$$

*in the intervall $[t_0 - c, t_0 + c]$ with $c := \min(a, \frac{b}{\max_{(t,y)\in Q} \|f(t,y)\|_X})$.*

*Proof.* See for example [73]. □

**Theorem 2.12** (Implicit function theorem)**.**
*Let $X, Y$ and $Z$ be Banach spaces. Let the mapping $f : X \times Y \to Z$ be continuously Fréchet differentiable. If $(x_0, y_0) \in X \times Y$, $f(x_0, y_0) = 0$, and $y \mapsto Df(x_0, y_0)(0, y)$ is a Banach space isomorphism from $Y$ onto $Z$, then there exist neighbourhoods $U$ of $x_0$ and $V$ of $y_0$ and a Fréchet differentiable function $g : U \to V$ such that $f(x, g(x)) = 0$ and $f(x, y) = 0$ if and only if $y = g(x)$, for all $(x, y) \in U \times V$.*

**Theorem 2.13** (Elliptic regularity)**.**
*Assume $U \subset \mathbb{R}^n$ is a bounded, open set. Let the differential operator*

$$Lu = -\sum_{i,j=1}^{n} (a^{ij}(x)u_{x_i})_{x_j} + \sum_{i=1}^{n} b^i(x)u_{x_i} + c(x)u$$

*be given. Assume*

$$a^{ij} \in C^1(U), b^i, c \in L^\infty(U) \ (i, j = 1, \ldots, n)$$

*and*

$$f \in L^2(U).$$

*Suppose furthermore that $u \in H^1(U)$ is a weak solution of the elliptic PDE*

$$Lu = f \quad \text{in } U.$$

*Then*

$$u \in H^2_{loc}(U)$$

*and for each open subset $V \subset U$ with $\overline{V} \subset U$ compact, we have the estimate*

$$\|u\|_{H^2(V)} \leq C(\|f\|_{L^2(U)} + \|u\|_{L^2(U)}),$$

*the constant $C$ depending only on $V$, $U$, and the coefficients of $L$.*

*Proof.* See for example [41, 76] □

**Theorem 2.14** (Singular value decomposition)**.**
*Let $M \in \mathbb{R}^{n \times m}$ be a matrix with rank $r$. The singular value decomposition of $M$ is a factorization*

$$M = U\Sigma V^T,$$

*where $U$ is an $m \times m$ orthogonal matrix with columns $u_1, \ldots, u_m$, $\Sigma$ is an $m \times n$ rectangular diagonal matrix with non-negative real number on the diagonal, $V$ is an $n \times n$ orthogonal matrix with columns $v_1, \ldots, v_n$. The diagonal entries $\sigma_i = \Sigma_{ii}$ of $\Sigma$ are uniquely determined by $M$ are known as the singular values of $M$. The SVD always exists, but is not unique. The singular value decomposition can be written as $M = \sum_{i=1}^{r} \sigma_i u_i v_i^T$.*

**Definition 2.15** (Locally Lipschitz continuity)**.**
*A function $f : \mathbb{R}^n \to \mathbb{R}$ is called locally Lipschitz continuous, if for every bounded subset $B$ of $\mathbb{R}^n$ there exists a constant $K > 0$ such that*

$$|f(x_1) - f(x_2)| \leq K\|x_1 - x_2\|$$

*for all points $x_1$ and $x_2$ in $B$.*

CHAPTER 3

# Physical problem

In this chapter, we describe the model we are using to simulate the behavior of an asynchronous machine. Starting with the description of the electromagnetic fields, via network equations to the equation of motion, torque and boundary conditions. In contrast to a permanent magnet synchronous machine, the asynchronous machine, which is also called induction machine, does not require magnets in the rotor, but a so called squirrel cage which is a short-circuited bar network and will be described in detail in this chapter.

## 3.1 Maxwell's Equation

The set of Maxwell's is the given by:

$$\begin{aligned} \nabla \cdot \vec{B} &= 0 && \text{Gauss's law for magnetism} \\ \nabla \cdot \vec{D} &= \rho && \text{Gauss's law} \\ \nabla \times \vec{H} &= \frac{\partial \vec{D}}{\partial t} + \vec{J} && \text{Ampère's law} \\ \nabla \times \vec{E} &= -\frac{\partial \vec{B}}{\partial t} && \text{Faraday's law of induction} \end{aligned} \tag{3.1}$$

In 1864 Maxwell was the first to write down all the equations which are known today as Maxwell's equations. In his paper [57], however, there were 20 equations, because they were not written down in vector notation. Maxwell added the displacement currents to Faraday's Law. In 1891, Oliver Heaviside published a paper in which Maxwell's equations were written in vector notation, i.e. essentially in the form we

know today. These equations describe the interaction between eletric and magnetic fields via the involved quantities:

$$\begin{aligned} &\vec{B} && \text{the magnetic flux density,} \\ &\vec{D} && \text{the displacement field,} \\ &\rho && \text{the density of free electric charges,} \\ &\vec{H} && \text{the magnetic field strength,} \\ &\vec{J} && \text{the current density,} \\ &\vec{E} && \text{the electric field strength.} \end{aligned} \tag{3.2}$$

All quantities depend on space and time and the ones with a vector-arrow on top are three-dimensional vector fields, meaning $\rho$ is the only scalar quantity. These quantities follow the material relations

$$\begin{aligned} \vec{D} &= \epsilon \vec{E}, \\ \vec{B} &= \mu \vec{H}, \\ \vec{J} &= \sigma \vec{E}, \end{aligned} \tag{3.3}$$

with the electric permittivity $\epsilon$, the magnetic permeability $\mu$, which is related to the magnetic reluctivity $\nu$ by $\mu^{-1} = \nu$ and the electric conductivity $\sigma$. In general, these materials can be complicated, since in anisotropic material they behave differently in different directions, for some materials, they depend on the magnitutde of the underlying fields and therefore are nonlinear functions, in some materials hysteresis effects occur and further, they might depend on temperature.

In this work, we assume, that the materials we are using are simply constant scalars for different materials. But in our numerics we will examine nonlinear material behavior in the reluctivity $\nu$.

### 3.1.1 Nonlinear Material Properties

In ferromagnetic magnetic material, the relation between the magnetic field strength $\vec{H}$ and the magnetic flux density $\vec{B}$ depends on the magnitude of the underlying field

$$\vec{H} = \nu(|\vec{B}|)\vec{B}.$$

The relation can be modelled by a B-H-curve

$$f : \mathbb{R}_0^+ \to \mathbb{R}_0^+ : |\vec{H}| \mapsto |\vec{B}| = f(|H|)$$

which we assume satisfies the following properties:

**Assumption 3.1** ([42], Assumption 1)**.** *Let $f : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ be a $B-H$-curve. Then the following holds:*

1. *$f$ is continuously differentiable on $\mathbb{R}_0^+$,*
2. *$f(0) = 0$,*
3. *$f'(s) \geq \mu_0$ for all $s \geq 0$,*
4. *$\lim_{s\to\infty} f'(s) = \mu_0$.*

Here $\mu_0$ is the magnetic permeability in vacuum. The reluctivity $\nu$ is related to $f$ via

$$\nu(s) = \frac{f^{-1}(s)}{s}.$$

If $f$ fulfills Assumption (3.1), then there exists a $\nu_l > 0$ and it holds

$$\nu_l \leq \nu(s) \leq \nu_0,$$

where $\nu_0 = \frac{1}{\mu_0}$ is the magnetic relcuctivity of vacuum, see [66].

### 3.1.2 Magnetoquasistatics

Maxwell's equations describe general electromagnetic phenomena. When considering special applications, some of the quantities in (3.2) may be negligibly small compared to the other quantities, so that not all quantities and equations need to be considered to describe a behavior accurately enough. Depending on which equations are neglected, the resulting systems have different behaviors. For example, in a permanent magnet synchronous machine (PMSM) the magnetic field in the rotor is already given by the magnets and does not have to be built up by induced currents first, i.e. eddy currents can be neglected in the model of a PMSM and it is sufficient to consider the so-called magnetostatic approximation of Maxwell's equations, which leads to an elliptic equation, while the full set of Maxwell's equations leads to hyperbolic PDEs.

In applications with slowly varying electromagnetic fields and where the electric

energy is low, compared to time dependent power losses and magnetic energy, i.e.

$$\left|\frac{\partial \vec{D}}{\partial t}\right| \ll \left|\vec{J}\right|,$$

the displacements currents can be neglected [53]. In this case one ends up with the *magnetoquasistatic approximation* of Maxwell's equations

$$\begin{aligned} \nabla \cdot \vec{B} &= 0, \\ \nabla \cdot \vec{D} &= \rho, \\ \nabla \times \vec{H} &= \vec{J}, \\ \nabla \times \vec{E} &= -\frac{\partial \vec{B}}{\partial t}. \end{aligned} \tag{3.4}$$

To combine the equations into one, potentials can be introduced, which guarantee the fulfillment of some of the equations. Introducing the magnetic vector potential (MVP) $\vec{A}$ and choosing $\vec{B} = \nabla \times \vec{A}$ leads to

$$\vec{B} = \nabla \times \vec{A} \Rightarrow \nabla \cdot \vec{B} = 0$$

meaning that the divergence-free condition on the magnetic flux density $\vec{B}$ is automatically fulfilled, since the divergence of a rotation always vanishes. Further we can introduce the electric scalar potential $q$ to obtain

$$\vec{E} = -\frac{\partial \vec{A}}{\partial t} - \nabla q \Rightarrow \nabla \times \vec{E} = -\nabla \times \frac{\partial \vec{A}}{\partial t}.$$

Using the material relations

$$\nu \vec{B} = \vec{H} \text{ and } \vec{J} = \sigma \vec{E}$$

one can combine the equations above and obtain the parabolic-elliptic partial differential equation (PDE)

$$\nabla \times (\nu \nabla \times \vec{A}) + \sigma \frac{\partial \vec{A}}{\partial t} = -\sigma \nabla q. \tag{3.5}$$

This is a degenerate parabolic equation, since in materials where the conductivity $\sigma$ is zero, the time derivative vanishes and one has an elliptic equation in this region.

This model is also known as *eddy current model*, as the varying magnetic field term $\sigma \frac{\partial \vec{A}}{\partial t}$ is responsible for the induction of eddy currents in regions with nonzero conductivity $\sigma$. The current density $\vec{J} = \sigma \vec{E} = -\sigma(\frac{\partial \vec{A}}{\partial t} - \nabla q)$ is split into eddy

current density and source current density.

## 3.2 Reduction from 3D to 2D

We call the domain of the machine $V_{3D}$ which is a subset of $\mathbb{R}^3$. If we assume that the material and the geometry of the machine is invariant in the $z$-direction and that the currents $\vec{J}$ also only flow in $z$-direction, then we can derive a $2D$ model on the cross-section of the machine, see [66]. We denote the cross-section by $\Omega \subset \mathbb{R}^2$.

$$\vec{J}(x,y,z,t) = \begin{pmatrix} 0 \\ 0 \\ J_z(x,y,t) \end{pmatrix} \text{ and } \vec{H}(x,y,z,t) = \begin{pmatrix} H_x(x,y,t) \\ H_y(x,y,t) \\ 0 \end{pmatrix}$$

By the material law $\vec{B} = \nu\vec{H}$ we also have

$$\vec{B}(x,y,z,t) = B_x(x,y,t)\vec{e}_x + B_y(x,y,t)\vec{e}_y.$$

Since we have chosen $\vec{B} = \nabla \times \vec{A}$, we get

$$\nabla \times \vec{B} = \nu \begin{pmatrix} \partial_y B_z - \partial_z B_y \\ \partial_z B_x - \partial_x B_z \\ \partial_x B_y - \partial_y B_x \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ J_z \end{pmatrix} = \vec{J}$$

For the curl-curl-part we get (since the material is isotropic)

$$\nabla \times (\nu\nabla \times \vec{A}) = \begin{pmatrix} \partial_z \nu \partial_x A_z \\ \partial_z \nu \partial_y A_z \\ -\partial_x \nu \partial_x A_z - \partial_y \nu \partial_y A_z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -\nabla \cdot (\nu \nabla A z) \end{pmatrix}$$

such that we can write the equation (3.5) in 2D planar case as

$$\sigma \frac{\partial A_z}{\partial t} - \nabla \cdot (\nu \nabla A_z) = J_{s,z} = -\sigma \partial_z q. \tag{3.6}$$

Note, that the Columb gauche $\nabla \cdot \vec{A} = 0$, which is used to guarantee a unique solution $\vec{A}$ to the eddy current model in $3D$ is fulfilled since $A_z$ does not depend on $z$.

## 3.3 Domain decomposition and boundary conditions

We will use a domain decomposition in the continuous model and in the discrete model of the machine to account for the rotation of the inner part. We split the whole domain $\bar{\Omega} = \bar{\Omega}_1 \cup \bar{\Omega}_2$ into the rotor domain $\Omega_1$ and the stator domain $\Omega_2$. We then get an equation for the rotor and for the stator

$$\sigma_i \frac{\partial A_{z,i}}{\partial t} - \nabla \cdot (\nu_i \nabla A_{z,i}) = J_{s,z,i} \text{ in } \Omega_i, \ i = 1, 2.$$

In order to couple the two equations, we supply them with two coupling conditions, which depend on the rotation $\theta(t)$ of the rotor.

$$\begin{aligned} A_{z,1}(r_{-\theta(t)}x, t) - A_{z,2}(x, t) = 0 \quad \Gamma\times]0, T[, \\ \nu_1(r_{-\theta(t)}x)\frac{\partial A_{z,1}}{\partial n}(r_{-\theta(t)}x, t) - \nu_2(x)\frac{\partial A_{z,2}}{\partial n}(x, t) = 0 \quad \Gamma\times]0, T[. \end{aligned}$$

Where $\Gamma := \bar{\Omega}_1 \cap \bar{\Omega}_2$ is called airgap or interface. In the conditions

$$r_{\theta(t)} := \begin{pmatrix} \cos(\theta(t)) & -\sin(\theta(t)) \\ \sin(\theta(t)) & \cos(\theta(t)) \end{pmatrix} \tag{3.7}$$

describes a counterclockwise rotation (we assume that $\Omega$ is a disk with center 0). $r_{-\theta(t)}$ then describes a clockwise rotation. The first condition imposes the continuity of the magnetic vector potential and the second one, the continuity of the tangential component of the magnetic field strength. In the derivatives $\frac{\partial}{\partial n} = n \cdot \nabla$ the normal direction is with respect to the unit normal vector pointing from rotor to stator. Since the material in the airgap is air, we have $\nu_1(x) = \nu_2(y) = \nu_{air}$ for all $x, y \in \Gamma$.

On the outer boundary, we have homogeneous Dirchlet boundary conditions

$$\begin{aligned} A_{z,1}|_{\bar{\Omega}_1 \setminus \{\Omega_1 \cup \Gamma\}} = 0, \\ A_{z,2}|_{\bar{\Omega}_2 \setminus \{\Omega_2 \cup \Gamma\}} = 0. \end{aligned}$$

## 3.4 Network in Rotor

In the rotor of an induction machine a short-circuited rotor bat network, which is known as *squirrel cage*, is located. The bars of the squirrel cage are made out of conductive material for example copper or aluminium and the bars are short-circuited by rings at the front and the back of the machine, such that when the bars

experience a varying magnetic field a current is induced, which can be explained by Faraday's law. We want to describe the circuit of the squirrel cage by the modified nodal analysis and couple it with the eddy current model (3.6), since in the $2D$ eddy current model only the cross-section of the machine and the rotor bars is modelled and the short-circuited bar network is not described.

### 3.4.1 Modified Nodal Analysis

To describe the connected electric circuit in the rotor bar, we use the Modified Nodal Analysis (MNA) [79]. Its idea is to describe the circuit with a directed graph. In this graph with $b$ edges (or branches) and $n$ nodes, we assign a potential $\varphi_j$, $j = 1, \dots, n$ to each node and a current $i_j$, $j = 1, \dots, b$ to every branch. Each branch represents an electric element, for example a resistance $R$ or an inductance $L$, which describe the physical relation between the currents and voltages along the edges. One can define an incidence matrix $A \in \{-1, 0, 1\}^{n \times b}$ for the graph, which has the following structure

$$a_{kj} = \begin{cases} 1 & \text{if the branch } j \text{ leaves the node } k, \\ -1 & \text{if the branch } j \text{ enters the node } k, \\ 0 & \text{else.} \end{cases}$$

Removing the last row of the incidence matrix gives the reduced incidence matrix. Since every row corresponds to one node, the removed node is defined as the mass (or reference) node with a potential of 0 and then the values of the other nodes are defined as voltages with respect to the mass node. We can use the (reduced) incidence matrix $A$ to describe Kirchhoff's circuit laws, i.e. Kirchhoff's current law and Kirchhoff's voltage law.

Kirchhoff's current law states, that all currents flowing in to and out of a given node have to sum up to 0, which implies

$$Ai = 0,$$

when using the incidence matrix.

Kirchhoff's voltage law states, that the sum of the potentials in a loop is zero. Let $\varphi = (\varphi_1, \dots, \varphi_n)^T$ be the vector of nodal potentials. If node $i$ and node $j$ are connected through branch $k$ (pointing from node $k$ to node $j$) then the branch voltage is given by

$$v_k = \varphi_k - \varphi_j.$$

For the vector $v = (v_1, \ldots, v_b)^T$ of branch voltages one has

$$v = A^T \varphi,$$

such that Kirchhoff's voltage law is fulfilled. As seen in Figure (3.1), in the rotor bar network, we have two different elements on the branches: resistances and FE-coupling. The behavior of the branches with underlying resistance can simply be described by Ohm's law. Selecting the rows of $A^T$ which correspond to resistance branches denoted by $A_R^T$, we can write

$$A_R^T \varphi = R i_R \;\Leftrightarrow\; R^{-1} A_R^T \varphi = G A_R^T \varphi = i_R,$$

where $R$ is a diagonal resistance matrix containing the resistances of the branches and $G = R^{-1}$ a conductance matrix. The branches with the underlying FE model also have a resistive term and an additional term which depends on the magnetic vector potential and which takes into account that the varying magnetic field induces currents in the FE branches. These additional terms are stated in the next section.

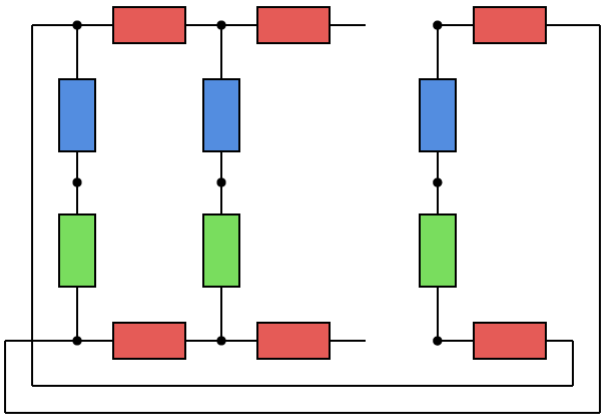

**Figure 3.1.** Graph of the rotor bar network. For every bar there are three nodes and four branches: two branches for the resistance of the outer rings (red), one branch for the resistance of the rotor bar outside of the FE model (blue) and one FE branch (green).

## 3.5 Coupling of electric circuits and electromagnetic fields

To couple the electric circuit equations for the rotor bars and the circuit for the windings in the stator with the electromagnetic field equations we use winding functions, specifically solid and stranded conductor models [75, 32, 5]. The next two sections follow [75].

### 3.5.1 Solid conductors

Solid conductor models model, as the name suggests, solid conductors, e.g. conductors made of one piece of metal. As we are in the 2D plane, they are simply functions with support in the cross-section of the rotor bars, which are divided by the length of the machine $\ell_z$ in $z$-direction to account for the fact, that we are only considering the cross-section of the machine. We denote the solid conductor models with $\vec{\xi}$.

$$\vec{\xi}(x,y,z) := \frac{\mathbf{1}_{\Omega_{\text{sol}}}(x,y)}{l_z}\vec{e}_z, \quad \mathbf{1}_{\Omega_{\text{sol}}}(x,y) = \begin{cases} 1 & \text{if } (x,y) \in \Omega_{\text{sol}}, \\ 0 & \text{else.} \end{cases}$$

We will denote the $z$-component of $\vec{\xi}$ simply as $\xi$. A voltage drop $u$ in the solid conductor is then applied as source current density $-\sigma\nabla q = \sigma\vec{\xi}u \;\Leftrightarrow\; -\sigma\partial_z q = \sigma\xi u$ in the right hand side of equation (3.6). The complete current density in the bars includes the eddy current density and is given by

$$\vec{J} = \sigma(\vec{\xi}u - \partial_t\vec{A}) = \vec{J}_s + \vec{J}_e.$$

To compute the currents $i_{rb}$ in the rotor bars one can integrate $\vec{J}$ over the electrodes or equivalently, integrate $\vec{\xi}\cdot\vec{J}$ over the whole $3D$ domain $V_{3D}$

$$\begin{aligned} i_{rb} &= \int_{V_{3D}} \vec{\xi} \cdot \vec{J} \,\mathrm{d}x \\ &= \int_{\Omega_{\mathrm{sol}}} \int_0^{\ell_z} \frac{1}{\ell_z} J_z \,\mathrm{d}z \,\mathrm{d}x \\ &= \int_{\Omega_{\mathrm{sol}}} J_z \,\mathrm{d}x \\ &= \int_{\Omega_{\mathrm{sol}}} \sigma \left( \frac{1}{\ell_z} u - \partial_t A_z \right) \mathrm{d}x \\ &= \int_{\Omega_{\mathrm{sol}}} \frac{\sigma}{\ell_z} \,\mathrm{d}x u - \int_{\Omega_{\mathrm{sol}}} \sigma \partial_t A_z \,\mathrm{d}x. \end{aligned}$$

The DC conductance $G$, which is inverse to the resistance, is defined by

$$G_\xi := \int_{V_{3D}} \sigma \xi \xi \,\mathrm{d}x = \int_{\Omega_{\mathrm{sol}}} \frac{\sigma}{\ell_z} \,\mathrm{d}x.$$

This leads to the two equations

$$\begin{aligned} &\sigma \partial_t A_z + \nabla \cdot \nu \nabla A_z = -\sigma \nabla q = \sigma \xi u, \qquad \text{on } \Omega_1 \\ &i_{rb} = G_\xi u - \int_\Omega \sigma \ell_z \xi \partial_t A_z \,\mathrm{d}x. \end{aligned}$$

To incorporate the rotor bar network in the form of the MNA described in the previous section, we use nodal potentials $\varphi$ and the voltage drop is given by $u = A^T \varphi$. For each of the 40 rotor bars, we have a solid conductor model, called $\xi_q$, $q = 1, \ldots, 40$, which has its support on the domain of the rotor bar. The reduced incidence matrix has dimension $119 \times 160$, since for each rotor bar, we have 4 branches and 3 nodes, where one node is chosen as mass node and removed. When ordering the incidence matrix, such that the first 40 columns correspond to the FE branches, we set up the matrix $G$ as

$$G := \mathrm{diag}(G_{\xi_1}, \ldots, G_{\xi_{40}}, G_{barext_1}, \ldots, G_{barext_{40}}, G_{ringext_1}, \ldots, G_{ringext_{80}})\},$$

where $G_{barext_i} \in \mathbb{R}$ is the DC conductance for the rotor bar branch $i$ outside of the FE model and $G_{ringext_i} \in \mathbb{R}$ is the DC conductance for the end ring branch $i$, see Figure (3.1). We can then write

$$\sigma \sum_{q=1}^{40} \xi_q A^T_{q,:} \varphi = \sigma \xi_a^T A^T \varphi$$

where $\xi_a = (\xi_1, \ldots, \xi_{40}, \mathbf{0})^T$. Further inserting leads to

$$i_{rb} = GA^T\varphi - \int_\Omega \sigma \ell_z \xi_a \partial_t A_z \, \mathrm{d}x.$$

Now multiplying this equation with the incidence matrix $A$, we incorporate Kirchhoff's current law

$$0 = Ai_{rb} = AGA^T\varphi - A\int_\Omega \sigma \xi_a \partial_t A_z \, \mathrm{d}x$$

So we end up with the equations in the rotor

$$\begin{aligned} \sigma\partial_t A_z + \nabla \cdot \nu \nabla A_z &= \sigma \xi_a^T A^T \varphi, \qquad \text{on } \Omega_1 \\ 0 &= AGA^T\varphi - A\int_\Omega \sigma \ell_z \xi_a \partial_t A_z \, \mathrm{d}x. \end{aligned}$$

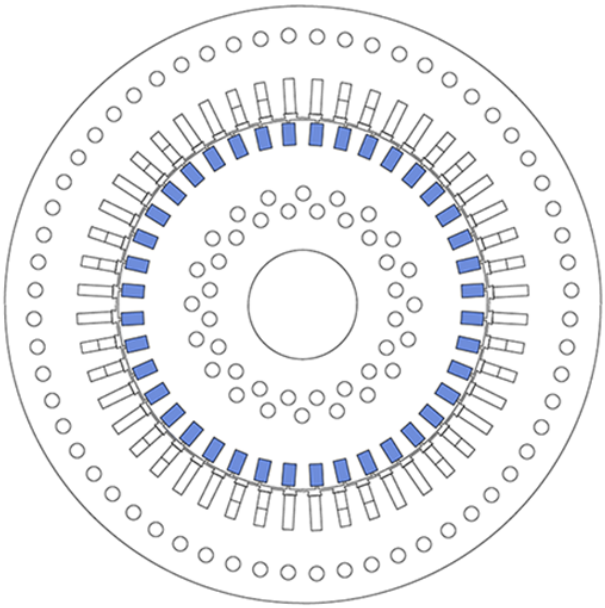

**Figure 3.2.** Support of the 40 solid conductor models on the domain.

### 3.5.2 Stranded conductors

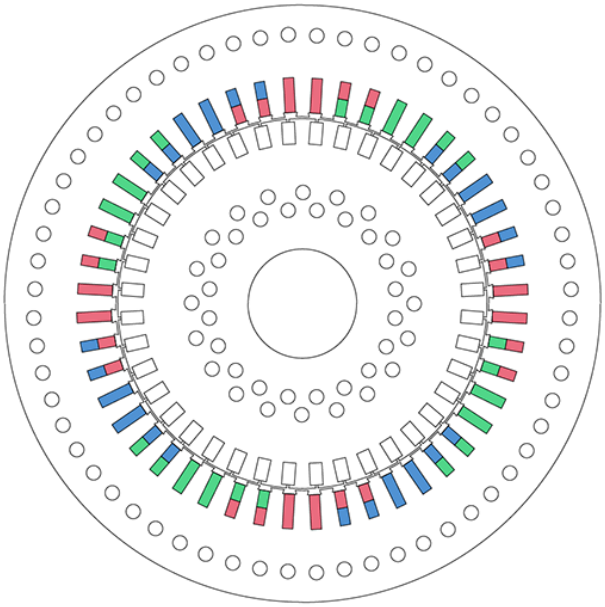

**Figure 3.3.** Support of the three stranded conductor models on the domain.

To model the windings in the stator, we use a stranded conductor models $\chi$. In $2D$ these stranded conductor models homogeneously distribute a current quantity $i(t)$ to an area $\Omega_{\text{str}}$, such that $\chi(x,y)i(t)$ is a current density. By denoting the measure of the support of $\chi$ with $|\Omega_{\text{str}}|$ we have

$$\vec{\chi}(x,y,z) := \frac{N_t \mathbf{1}_{\Omega_{\text{str}}}(x,y)}{|\Omega_{\text{str}}|}\vec{e}_z.$$

As with the solid conductor models, we call the $z$-component of $\vec{\chi}$ simply $\chi$. As they model windings, they need to fulfil, that the integral of the absolute value of the winding function over any intersecting plane is equal to the number of turns $N_t$ of the winding. Since the electric field $\vec{E}$ induces another voltage $\int_\Omega \chi E_z \,\mathrm{d}x$ and using $\vec{E} = -\frac{\mathrm{d}\vec{A}}{\mathrm{d}t}$ [23], we can describe the voltage drop $q$ for the stranded conductor as

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega \ell_z \chi A_z \,\mathrm{d}x = \frac{\mathrm{d}}{\mathrm{d}t}\int_{V_{3D}} \vec{\chi}\cdot\vec{A}\,\mathrm{d}V + Ri = q.$$

Here we have an induced voltage drop and a resistive voltage drop, where the resistance $R$ is given by

$$\begin{aligned} R &= \int_{V_{3D}} \frac{\vec{\chi}\cdot\vec{\chi}}{\tau\gamma}\,\mathrm{d}V \\ &= \int_{\Omega_{\mathrm{str}}}\int_0^{\ell_z} \frac{N_t^2}{\tau\gamma|\Omega_{\mathrm{str}}|^2}\,\mathrm{d}z\,\mathrm{d}x \\ &= \int_{\Omega_{\mathrm{str}}} \ell_z \frac{N_t^2}{\tau\gamma|\Omega_{\mathrm{str}}|^2}\,\mathrm{d}x \\ &= \ell_z \frac{N_t^2}{\tau\gamma|\Omega_{\mathrm{str}}|}, \end{aligned}$$

where $\tau$ is the conductivity of the wire material and $\gamma \in ]0,1]$ accounts for the fact, that only a portion of the stranded conductor support is made of conductive material.

As the induction machine is supplied by a three-phase current, we have 3 winding functions $\chi_n$, $n = 1,2,3$, i.e. one for each phase of the three-phase current. We denote in the following a three-phase current with $i_{st}(t) : \mathbb{R} \mapsto \mathbb{R}^3$. Since the cross-section of a coil is represented with only one winding function, there are regions of the support with a negative as well as a with a positive sign. As we want to account for inductances of the end-windings, which are not captured in our $2D$ model, we insert an additional end-winding inductance term $L$, see [34, 2].

We incorporate the windings functions into the eddy current model (3.6) via the source current density $J_{s,z}$. Since the conductance $\sigma$ is 0 in the stator domain, we have with $\chi_a := (\chi_1, \chi_2, \chi_3)^T$

$$\begin{aligned} &\nabla\cdot\nu\nabla A_z = \chi_a^T i_{st}, && \text{on } \Omega_2, \\ &q_n = R_n i_n + L_n \frac{\mathrm{d}i_n}{\mathrm{d}t} + \int_{\Omega_{\mathrm{str,n}}} \ell_z \chi \partial_t A_z\,\mathrm{d}x,\ n = 1,2,3. \end{aligned} \tag{3.8}$$

## 3.6 Equation of motion

To model the rotation of the machine, we use the equation of motion

$$I\frac{\mathrm{d}^2\theta(t)}{\mathrm{d}t^2} + \beta\frac{\mathrm{d}\theta(t)}{\mathrm{d}t} = -M_{\mathrm{load}} + M_{em}(A_z(x,t),\theta(t)).$$

Here, $\theta(t)$ is the angular position of the rotor, $\frac{\mathrm{d}\theta(t)}{\mathrm{d}t}$ the angular velocity and $\frac{\mathrm{d}^2\theta(t)}{\mathrm{d}t^2}$ the angular acceleration. $I$ is the moment of inertia of the rotor, $M_{\mathrm{load}}$ is the load

characteristic of the mechanical drive train and $\beta$ is the torsional friction coefficient. The term $M_{em}$ is the electromagnetic torque, which depends on the magnetic vector potential $A_z$ and the angle $\theta(t)$. In the next section we describe how the torque is calculated.

### 3.6.1 Torque computation

To compute the torque of the machine, we are using a formula which is derived and presented for example in [39]. The torque is a function depending on the magnetic vector potential in rotor $A_{z,1}$ domain and stator domain $A_{z,2}$ and the rotation angle $\theta$

$$M_{em}(A_z, \theta) = l_z \int_\Gamma n \cdot (\nu_2 \nabla A_{z,2}) n^\perp \cdot \nabla A_{z,1}(r_{-\theta} x) \, \mathrm{d}S.$$

Here $l_z$ is the length of the machine in $z$-direction, $n$ is the unit normal vector on the interface pointing from rotor to stator and $n^\perp$ is the vector tangential to the interface pointing counterclockwise. Note that since we have chosen $\vec{B} = \nabla \times \vec{A}$ and $\vec{A} = \begin{pmatrix} 0 \\ 0 \\ A_z \end{pmatrix}$. We have

$$\begin{aligned} H_\theta &= n^\perp \cdot \vec{H} = n^\perp \cdot \nu \vec{B} \\ &= n^\perp \cdot \nu \nabla \times \vec{A} = n^\perp \cdot \nu \begin{pmatrix} \partial_y A_z \\ -\partial_x A_z \\ 0 \end{pmatrix} \\ &= -n \cdot \nu \nabla A_z. \end{aligned}$$

If $n = \begin{pmatrix} n_1 \\ n_2 \\ 0 \end{pmatrix}$, then $n^\perp = \begin{pmatrix} -n_2 \\ n_1 \\ 0 \end{pmatrix}$ and therefore

$$B_n = n \cdot \vec{B} = -n^\perp \cdot \nabla A_z.$$

## 3.7 Right hand side

The right hand side which excites the whole system is given by the voltages applied to the 3 stator windings (3.8), which are connected in series.

$$
\begin{aligned}
q_1(t) &= \hat{V}\cos\left(2\pi f t\right), \\
q_2(t) &= \hat{V}\cos\left(2\pi f t - 2\frac{\pi}{3}\right), \\
q_3(t) &= \hat{V}\cos\left(2\pi f t + 2\frac{\pi}{3}\right).
\end{aligned}
$$

Here $\hat{V}$ is the amplitude (or peak value) of the voltages and the machine is operated at $f = 50\text{Hz}$.

## 3.8 Existence result for the state equation

From now on, we will denote the $z$-component of the magnetic vector potential as $u := A_z$. The system, which we call *strong state equation*, to describe the induction machine is given by

$$
\begin{aligned}
\sigma(x)\frac{\partial u_1}{\partial t}(x,t) - \nabla\cdot(\nu_1(x)\nabla u_1(x,t)) - \sigma(x)\xi_a^T(x)A^T\varphi(t) &= 0 && \Omega_1\times]0,T[\\
-\nabla\cdot(\nu_2(x)\nabla u_2(x,t)) - \chi_a^T(x)i_{st}(t) &= 0 && \Omega_2\times]0,T[\\
u_1(r_{-\theta(t)}x,t) - u_2(x,t) &= 0 && \Gamma\times]0,T[\\
\nu_1(r_{-\theta(t)}x)\frac{\partial u_1}{\partial n}(r_{-\theta(t)}x,t) - \nu_2(x)\frac{\partial u_2}{\partial n}(x,t) &= 0 && \Gamma\times]0,T[\\
-A\left(\int_{\Omega_1}\sigma(x)\frac{\partial u_1}{\partial t}(x,t)\xi_a(x)\ell_z\,\mathrm{d}x\right)^T + AGA^T\varphi(t) &= 0 &&\\
\int_{\Omega_{str,n}}\frac{\partial u_2}{\partial t}(x,t)\ell_z\chi_n\,\mathrm{d}x + R_{st,n}i_{st,n}(t) + L_n\frac{\mathrm{d}i_{st,n}}{\mathrm{d}t}(t) &= q_n(t) && n=1,2,3\\
\frac{\mathrm{d}\theta}{\mathrm{d}t}(t) - \dot\theta(t) &= 0 &&\\
I\frac{\mathrm{d}\dot\theta}{\mathrm{d}t}(t) - M_{em}(u,\theta) + M_{load} + \beta\dot\theta(t) &= 0 &&\\
\sigma u_1(x,0) &= 0 && \Omega_1\\
\int_{\Omega_{str,n}} u_2(x,0)\ell_z\chi_n\,\mathrm{d}x &= 0 && n=1,2,3\\
u_1(x,t) &= 0 && \partial\Omega_1\backslash\Gamma\times]0,T[\\
u_2(x,t) &= 0 && \partial\Omega_2\backslash\Gamma\times]0,T[\\
i(0) &= 0 &&\\
\theta(0) &= 0 &&\\
\dot\theta(0) &= 0 &&
\end{aligned}
\tag{3.9}
$$

The quantities which we call state variables and obtain from solving the state equation are

| | |
|---|---|
| $u_1$ | = magnetic vector potential in the rotor, |
| $u_2$ | = magnetic vector potential in the stator, |
| $\varphi$ | = nodal potentials of the rotor bar circuit, |
| $i_{st}$ | = currents of the three-phase current, |
| $M_{em}$ | = electromagnetic torque, |
| $\theta$ | = angle of rotation, |
| $\dot\theta$ | = angular speed of rotation. |

We denote $\mathfrak{y} = (u_1, u_2, \varphi, i_{st}, M_{em}, \theta, \dot{\theta})^T$. Other quantities involved in the state equation are

$q_n$ = exciting voltages of the stator coils,
$\xi_a$ = vector of winding functions for rotor bars with added zero entries,
$\chi_a$ = vector of winding function for stator coils,
$A$ = reduced incidence matrix of rotor circuit,
$G$ = conductance marix of rotor circuit,
$R_{st,n}$ = resistance of stator coil,
$L_n$ = end-winding inductance of stator coil,
$I$ = moment of inertia of the rotor,
$M_{load}$ = external torque,
$\beta$ = friction coefficient.

### 3.8.1 Existence

To the author's knowledge, there is no proof of existence in the literature for the entire state system above (3.9). But there are existence results for subsets of the equations in 2D and 3D. For example for the existence for the 3D equation magnetoquasistatic approximation of Maxwell's equations, which is coupled with an equation accounting for the magnetic induction is shown in [63], based on a theorem by Showalter on degenerate parabolic equations [77], here linear material is assumed. The same problem with nonlinear material properties is examined in [23], where gradient systems on Hilbert spaces and $\epsilon$-gradients are used in the analysis. For the 2D domain, the magnetoquasistatic approximation of Maxwell's equations for domains which have a rotating subdomain is investigated in [20].

We will show here existence of the magnetoquastatic approximation of Maxwell's equation in 2D with linear material, which is coupled with circuit equations for the rotor bar network, coupled with an ordinary differential equation to account for the (self) inductance of the stator coils and coupled with the equation of motion. In the following we will proof the theorem below:

**Theorem 3.2.** *Let $\nu \in C^1(\bar{\Omega})$ and Assumption 3.5 hold, then the system 3.9 admits a weak solution $(u, \varphi, i_{st}, M_{em}(u, \theta), \theta) \in W^{1,\infty}(0, T; L^2(\mathcal{C})) \cap H^1(0, T; H^1(\Omega_1 \cup \Omega_2)) \cap L^2(0, T; H^2(\Omega_1 \cup \Omega_2)) \times L^\infty(0, T) \times H^1(0, T; \mathbb{R}^3) \times L^2(0, T) \times H^2(0, T)$.*

We borrow the notation from [20]. We will show the existence of a weak solution

for problem (3.9) for a fixed rotation by choosing a $\theta(t) \in H^2(\mathbb{R})$ and then investigate the state eqation given the rotation $\theta(t)$ neglecting the equation of motion. Afterwards we show, that the mapping from a rotation $\theta(t)$ to $\tilde{\theta}(t)$, which is evaluated by solving the field/circuit coupled system with the given rotation $\theta(t)$ for the magnetic vector potential $u$ and the solving the equation of motion with the torque in right hand side computed with $u$ for $\tilde{\theta}(t)$ has a fixed point. Note that all the data $(\sigma, \nu, \chi^T q)$ is piecewise constant in space and therefore not continuous in space, but smooth in time and lies in $C^\infty(0,t;L^\infty(\Omega))$.

The whole domain $\Omega$ is composed of the subdomains $\Omega_1$ and $\Omega_2$

$$\Omega_1 = \{x \in \mathbb{R}^2 : r_{inner} < \|x\|_2 < r_{ag}\}, \quad \Omega_2 = \{x \in \mathbb{R}^2 : r_{ag} < \|x\|_2 < r_{outer}\}.$$

The support of $\sigma$ is contained in $\Omega_1$ and is denoted by $\mathcal{C}$, the insulating part is denoted by $\mathcal{I} = \Omega \backslash \mathcal{C}$. $\Gamma = \bar{\Omega}_1 \cap \bar{\Omega}_2$ is the interface which is called airgap. Note that $\Gamma = \{x \in \mathbb{R}^2 : \|x\|_2 = r_{ag}\}$ is a circle and all the involved boundaries are Lipschitz boundaries. For any $t \in \mathbb{R}$ we have two rotation operators

$$R(t) : H^1(\Omega_1) \to H^1(\Omega_1) \text{ and } \mathcal{R}(t) : H^1(\Omega_1) \times H^1(\Omega_2) \to H^1(\Omega_1) \times H^1(\Omega_2),$$

with

$$R(t)(u_1)(x) := u_1(r_t x), \ \mathcal{R}(t)(u_1,u_2)(x) := (R(t)(u_1)(x), u_2(x)).$$

Where $r_t x$ rotates $x$ counterclockwise around the angle $\theta(t)$ as in (3.7), where we simply write $r_t$ instead of $r_{\theta(t)}$. We define

$$\mathcal{H}^1(\Omega) := \{u = (u_1,u_2) \in H^1(\Omega_1) \times H^1(\Omega_2) : u_1|_{\partial\Omega_1 \backslash \Gamma} = 0, u_2|_{\partial\Omega_2 \backslash \Gamma} = 0\}$$

and

$$\mathcal{U}^t := \{u = (u_1,u_2) \in \mathcal{H}^1(\Omega) : u_1(r_{-t}x) = u_2(x) \ x \in \Gamma\}.$$

For fixed $t$ the space $\mathcal{U}^t$ is isomorphic to $H_0^1(\Omega)$ through the rotation operator $\mathcal{R}(-t)$ and is therefore a Hilbert space with the norm

$$\|u\|_\star^2 := \|u_1\|_{H^1(\Omega_1)}^2 + \|u_2\|_{H^1(\Omega_2)}^2.$$

We define the space of trial functions to be

$$L^2(0,T;\mathcal{U}^t) := \{u = (u_1,u_2) \in L^2(0,T;\mathcal{H}^1(\Omega)) : u_1(r_{-t}x) = u_2(x) \text{ f.a.a. } x \in \Gamma, t \in ]0,T[\}.$$

Note $L^2(0,T;\mathcal{U}^t)$ is a closed subspace of $L^2(0,T;\mathcal{H}^1(\Omega))$ and is isomorphic to $L^2(0,T;H_0^1(\Omega))$ through $\mathcal{R}(-t)$, see [20]. For the spatial test functions we choose:

$$\mathcal{V} := \{v \in L^2(0,T;\mathcal{U}^t) \,|\, \exists u \in H_0^1(\Omega) \ : \ v \equiv \mathcal{R}(t)(u) \text{ for a.a. } x \in \Omega, t \in ]0,T[\}.$$

The functions $v \in \mathcal{V}$ correspond to a unique function $u \in H_0^1(\Omega)$ which evolves in time through $\mathcal{R}(t)$.

In the following we will write $\chi$ for $\chi_a$ and $\xi$ for $\xi_a$.

When we inspect the equation for the currents in the stator windings

$$\int_{\Omega_2} \partial_t u_2(x,t)\chi(x)\,\mathrm{d}x + R_{st} i_{st}(t) + L\frac{\mathrm{d}i_{st}(t)}{\mathrm{d}t} = q(t),$$

we can write it as

$$\frac{\mathrm{d}i_{st}(t)}{\mathrm{d}t} = -L^{-1}R_{st}i_{st}(t) + L^{-1}q(t) - L^{-1}\int_{\Omega_2} \partial_t u_2(x,t)\chi(x)\,\mathrm{d}x. \tag{3.10}$$

We assume that the inductances and resistances are equal for all three windings, such that the matrices $L$ and $R_{st}$ are given by $L = lI$ and $R_{st} = rI$ for positive scalar inductance $l$ and scalar resistance $r$ and $I$ being the identity matrix. The solution $i_{st}(t)$ is a vector of size 3 and $q(t) \in C^\infty(\mathbb{R}, \mathbb{R}^3)$ are the exciting voltages for the three-phase currents depending on time.

By using ODE theory (variation of constant) we can give a closed form of the solution of (3.10)

$$i_{st} = e^{-\frac{r}{l}(t-t_0)}i_0 + \int_{t_0}^{t} e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}\left(-\int_{\Omega_2}\partial_t u_2(x,\tau)\chi\,\mathrm{d}x + q\right)\mathrm{d}\tau$$

where $i_0$ is the inital value of $i_{st}$ and which is equivalent to

$$i_{st} = e^{-\frac{r}{l}(t-t_0)}i_0 - \int_{t_0}^{t} e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}\int_{\Omega_2}\partial_t u_2(x,\tau)\chi\,\mathrm{d}x\,\mathrm{d}\tau + \int_{t_0}^{t} e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}q\,\mathrm{d}\tau.$$

Note that it suffices that $\int_{\Omega_2}\partial_t u_2(x,t)\chi\,\mathrm{d}x \in L^1([0,T],\mathbb{R}^3)$ for $i_{st}(t) \in C^0([0,T],\mathbb{R}^3)$, see for example [70]. We will now use integration by parts in the second summand to get rid of the time derivative. To do this, we need the involved

functions to be continuous in time. We will show later that this is the case.

$$\begin{aligned}
&\int_{t_0}^{t} e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}\int_{\Omega_2}\partial_t u_2(x,\tau)\chi\,\mathrm{d}x\,\mathrm{d}\tau \\
=&\int_{t_0}^{t} e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}\partial_t\int_{\Omega_2} u_2(x,\tau)\chi\,\mathrm{d}x\,\mathrm{d}\tau \\
=&\left[e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}\int_{\Omega_2} u_2(x,\tau)\chi\,\mathrm{d}x\right]_{t_0}^{t} \\
&-\int_{t_0}^{t}\frac{r}{l}e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}\int_{\Omega_2} u_2(x,\tau)\chi\,\mathrm{d}x\,\mathrm{d}\tau \\
=&\frac{1}{l}\int_{\Omega_2} u_2(x,t)\chi\,\mathrm{d}x - e^{-\frac{r}{l}(t-t_0)}\frac{1}{l}\int_{\Omega_2} u_2(x,t_0)\chi\,\mathrm{d}x \\
&-\int_{t_0}^{t}\frac{r}{l}e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}\int_{\Omega_2} u_2(x,\tau)\chi\,\mathrm{d}x\,\mathrm{d}\tau
\end{aligned} \tag{3.11}$$

We will now plug these findings into the equation for the field in the stator and get the following right hand side

$$\begin{aligned}
J_s(x,t) &= \chi^T(x) i_{st}(t) \\
&= -\chi^T(x)\frac{1}{l}\int_{\Omega_2} u_2(\tilde{x},t)\chi(\tilde{x})\,\mathrm{d}\tilde{x} \\
&\quad + \chi^T(x)e^{-\frac{r}{l}(t-t_0)}\left(i_0 + \frac{1}{l}\int_{\Omega_2} u_2(\tilde{x},t_0)\chi(\tilde{x})\,\mathrm{d}\tilde{x}\right) \\
&\quad + \chi^T(x)\int_{t_0}^{t}\frac{r}{l}e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}\int_{\Omega_2} u_2(\tilde{x},\tau)\chi(\tilde{x})\,\mathrm{d}\tilde{x}\,\mathrm{d}\tau \\
&\quad + \chi^T(x)\int_{t_0}^{t} e^{-\frac{r}{l}(t-\tau)}\frac{1}{l} q\,\mathrm{d}\tau.
\end{aligned}$$

We then get the following equation for the stator $\Omega_2$

$$\begin{aligned}
&-\nabla\cdot(\nu_2(x)\nabla u_2)(x,t) + \chi^T\frac{1}{l}\int_{\Omega_2} u_2(\tilde{x},t)\chi\,\mathrm{d}\tilde{x} \\
=&\,\chi^T e^{-\frac{r}{l}(t-t_0)}\left(i_0 + \frac{1}{l}\int_{\Omega_2} u_2(\tilde{x},t_0)\chi\,\mathrm{d}\tilde{x}\right) \\
&+\chi^T\int_{t_0}^{t}\frac{r}{l}e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}\int_{\Omega_2} u_2(\tilde{x},\tau)\chi\,\mathrm{d}\tilde{x}\,\mathrm{d}\tau \\
&+\chi^T\int_{t_0}^{t} e^{-\frac{r}{l}(t-\tau)}\frac{1}{l} q\,\mathrm{d}\tau.
\end{aligned}$$

We will later use a fixed point argument to establish existence for the equation with the right hand side depending on the solution. We replace the function $u_2$ in the second summand on the right hand side by a given function $z_2 \in L^2(0,T;L^2(\Omega_2))$.

For the rotor we solve the equation for the circuit potentials

$$-A\left(\int_{\Omega_1}\sigma(x)\frac{\partial u_1}{\partial t}(x,t)\xi_a(x)\ell_z\,\mathrm{d}x\right)^T + AGA^T\varphi(t) = 0$$

for the potentials of the rotor bar network

$$\varphi(t) = \left(AGA^T\right)^{-1}A\left(\int_{\Omega_1}\sigma(x)\frac{\partial u_1}{\partial t}(x,t)\xi(x)\ell_z\,\mathrm{d}x\right).$$

We can plug this into the equation for the rotor $\Omega_1$ and end up with

$$\begin{aligned}&\sigma(x)\frac{\partial u_1}{\partial t}(x,t) - \nabla\cdot(\nu_1(x)\nabla u_1)(x,t)\\ &= \sigma(x)\xi^T(x)A^T\left(AGA^T\right)^{-1}A\left(\int_{\Omega_1}\sigma(\tilde{x})\frac{\partial u_1}{\partial t}(\tilde{x},t)\xi(\tilde{x})\ell_z\,\mathrm{d}\tilde{x}\right).\end{aligned}$$

For shorter notation we define $B := A^T\left(AGA^T\right)^{-1}A$. Note that since $B$ is symmetric and positive definite one can compute a Cholesky factorization $B = EE^T$. The equation for the stator $\Omega_1$ now reads

$$\begin{aligned}&\sigma(x)\frac{\partial u_1}{\partial t}(x,t) - \sigma(x)\xi^T(x)B\int_{\Omega_1}\sigma(\tilde{x})\frac{\partial u_1}{\partial t}(\tilde{x},t)\xi(\tilde{x})\ell_z\,\mathrm{d}\tilde{x}\\ &\quad - \nabla\cdot(\nu_1(x)\nabla u_1)(x,t) = 0.\end{aligned} \tag{3.12}$$

**Definition 3.3** (Strong formulation)**.** *The strong formulation for which we seek a*

*solution is given by*

$$
\begin{aligned}
&\sigma(x)\frac{\partial u_1}{\partial t}(x,t) - \sigma(x)\xi^T(x)B\int_{\Omega_1}\sigma(\tilde{x})\frac{\partial u_1}{\partial t}(\tilde{x},t)\xi(\tilde{x})\ell_z\,\mathrm{d}\tilde{x}\\
&-\nabla\cdot(\nu_1(x)\nabla u_1)(x,t) = 0, && \Omega_1\times]0,T[,\\
&-\nabla\cdot(\nu_2(x)\nabla u_2)(x,t) + \chi^T(x)\frac{1}{l}\int_{\Omega_2}u_2(\tilde{x},t)\chi(\tilde{x})\,\mathrm{d}\tilde{x}\\
&= \chi^T(x)e^{-\frac{r}{l}(t-t_0)}\left(i_0 + \frac{1}{l}\int_{\Omega_2}u_2(\tilde{x},t_0)\chi(\tilde{x})\,\mathrm{d}\tilde{x}\right)\\
&+\chi^T(x)\int_{t_0}^{t}\frac{r}{l}e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}\int_{\Omega_2}u_2(\tilde{x},\tau)\chi(\tilde{x})\,\mathrm{d}\tilde{x}\,\mathrm{d}\tau\\
&+\chi^T(x)\int_{t_0}^{t}e^{-\frac{r}{l}(t-\tau)}\frac{1}{l}q\,\mathrm{d}\tau, && \Omega_2\times]0,T[,\\
&u_1(r_{-t}x,t) = u_2(x,t), && x\in\Gamma\times]0,T[,\\
&\nu_1(r_{-t}x)\partial_n u_1(r_{-t}x,t) = \nu_2(x)\partial_n u_2(x,t), && x\in\Gamma\times]0,T[,\\
&\sigma u_i(\cdot,0) = \sigma u_0, && \Omega_i,\\
&\int_{\Omega_2}\chi_j u_i(\cdot,0)\,\mathrm{d}x = \int_{\Omega_2}\chi_j u_0\,\mathrm{d}x, && j=1,\dots,3,\\
&u_i(\cdot,t) = 0 && (\partial\Omega_1\cup\partial\Omega_2)\backslash\Gamma\times]0,T[.
\end{aligned}
\tag{3.13}
$$

Multiplying the equations for rotor and stator with a test function $v(x,t)\in\mathcal{V}$, summing them up and using integration by parts in the following terms

$$
-\int_{\Omega_1}\nabla\cdot(\nu_1\nabla u_1(t))\,v_1(t)\,\mathrm{d}x - \int_{\Omega_2}\nabla\cdot(\nu_2\nabla u_2(t))\,v_2(t)\,\mathrm{d}x
$$

leads to

$$
\sum_{i=1}^{2}\int_{\Omega_i}\nu_i\nabla u_i(t)\cdot\nabla v_i(t)\,\mathrm{d}x - \int_{\Gamma}\nu_1\frac{\partial u_1}{\partial n}(t)v_1(t)\,\mathrm{d}\Gamma + \int_{\Gamma}\nu_2\frac{\partial u_2}{\partial n}(t)v_2(t)\,\mathrm{d}\Gamma.
$$

Where $n$ is the unit normal vector on $\Gamma$ pointing from $\Omega_1$ to $\Omega_2$. Testfunctions $v(x,t)\in\mathcal{V}$ correspond to functions $\tilde{v}\in H_0^1(\Omega)$ which are rotated on the rotor part,

i.e. $v_2(x,t) = \tilde{v}_2(x)$ and $v_1(x,t) = \tilde{v}_1(r_t x)$. Therefore we have

$$\begin{aligned}
& -\int_\Gamma \nu_1 \frac{\partial u_1}{\partial n}(t) v_1(t)\,\mathrm{d}\Gamma + \int_\Gamma \nu_2 \frac{\partial u_2}{\partial n}(t) v_2(t)\,\mathrm{d}\Gamma \\
= & -\int_\Gamma \nu_1 \frac{\partial u_1}{\partial n}(t) \tilde{v}_1(r_t x)\,\mathrm{d}\Gamma + \int_\Gamma \nu_2 \frac{\partial u_2}{\partial n}(t) v_2(t)\,\mathrm{d}\Gamma \\
= & -\int_\Gamma \nu_1(r_{-t}x) \frac{\partial u_1(r_{-t}x,t)}{\partial n} \tilde{v}_1\,\mathrm{d}\Gamma + \int_\Gamma \nu_2 \frac{\partial u_2}{\partial n}(t) v_2(t)\,\mathrm{d}\Gamma \\
= & -\int_\Gamma \nu_1(r_{-t}x) \frac{\partial u_1(r_{-t}x,t)}{\partial n} \tilde{v}_2\,\mathrm{d}\Gamma + \int_\Gamma \nu_2 \frac{\partial u_2}{\partial n}(t) v_2(t)\,\mathrm{d}\Gamma \\
= & -\int_\Gamma \nu_1(r_{-t}x) \frac{\partial u_1(r_{-t}x,t)}{\partial n} v_2(t)\,\mathrm{d}\Gamma + \int_\Gamma \nu_2 \frac{\partial u_2}{\partial n}(t) v_2(t)\,\mathrm{d}\Gamma \\
= & \int_\Gamma \left( \nu_2 \frac{\partial u_2}{\partial n}(t) - \nu_1(r_{-t}x) \frac{\partial u_1(r_{-t}x,t)}{\partial n} \right) v_2(t)\,\mathrm{d}\Gamma
\end{aligned}$$

which vanishes by the second coupling condition. Note that $\nu_1(x) = \nu_2(x) = \nu_{air}$ for $x \in \Gamma$.

**Definition 3.4** (Weak form)**.** *We arrive at the weak form which has to hold $\forall v \in \mathcal{V}$ and a.a. $t \in ]t_0, s[$. We change the notation here from $]0,T[$ to $]t_0,s[$, since we will obtain a solution only for a short time horizon and then iteratively extend the solution to the whole time interval, using initial values for $t_0 > 0$.*

$$\begin{aligned}
& \left( \sigma \frac{\partial u_1(t)}{\partial t}, v_1(t) \right)_{\Omega_1} - \left( E^T \int_\Omega \sigma v_1(t) \xi\,\mathrm{d}x, E^T \int_\Omega \sigma \frac{\partial u_1(t)}{\partial t} \xi\,\mathrm{d}x \right)_2 \\
& + \int_{\Omega_1} \nu_1 \nabla u_1(t) \cdot \nabla v_1(t)\,\mathrm{d}x + \int_{\Omega_2} \nu_2 \nabla u_2(t) \cdot \nabla v_2(t)\,\mathrm{d}x \\
& + \left( \frac{1}{\sqrt{l}} \int_{\Omega_2} u_2(t) \chi\,\mathrm{d}x, \frac{1}{\sqrt{l}} \int_{\Omega_2} v_2(t) \chi\,\mathrm{d}x \right)_2 \\
& = \left( \chi^T, v_2(t) \right)_{\Omega_2} \left( \int_{t_0}^t \frac{1}{l} e^{-\frac{r}{l}(t-\tau)} \left( \frac{r}{l} \int_{\Omega_2} u_2(x,\tau) \chi\,\mathrm{d}x + q(\tau) \right) \mathrm{d}\tau \right. \\
& \left. + e^{-\frac{r}{l}(t-t_0)} \left( i_0 + \frac{1}{l} \int_{\Omega_2} u_2^0 \chi\,\mathrm{d}x \right) \right)
\end{aligned} \tag{3.14}$$

*The initial data is given by $\sigma u_1(x,t_0) = \sigma u_1^0$ and $\int_{\Omega_2} \chi u_2(x,t_0)\,\mathrm{d}x = \int_{\Omega_2} \chi u_2^0\,\mathrm{d}x$ and the boundary data is encoded in the spaces.*

We define the two bilinear forms:

$$m : \mathcal{U}^t \times \mathcal{U}^t \to \mathbb{R},\ m(u,v) := (\sigma u_1, v_1)_{\Omega_1} - \int_{\Omega_1} \sigma v_1 \xi^T \,\mathrm{d}x B \int_{\Omega_1} \sigma u_1 \xi \,\mathrm{d}x,$$

$$a : \mathcal{U}^t \times \mathcal{U}^t \to \mathbb{R},\ a(u,v) := \sum_{i=1}^{2} \int_{\Omega_i} \nu_i \nabla u_i \cdot \nabla v_i \,\mathrm{d}x + \frac{1}{l} \int_{\Omega_2} v_2 \chi^T \,\mathrm{d}x \int_{\Omega_2} u_2 \chi \,\mathrm{d}x.$$

**Theorem.** *The bilinear form $a$ is continuous and coercive on $\mathcal{U}^t$.*

*Proof.* For all $u \in \mathcal{U}^t$, we have

$$\begin{aligned}
a(u,u) &= \sum_{i=1}^{2} \int_{\Omega_i} \nu_i \nabla u_i \cdot \nabla u_i \,\mathrm{d}x + \frac{1}{l} \int_{\Omega_2} u_2 \chi^T \,\mathrm{d}x \int_{\Omega_2} u_2 \chi \,\mathrm{d}x \\
&\geq \min_{x \in \Omega} \nu \sum_{i=1}^{2} \|\nabla u_i\|^2_{L^2(\Omega_i)} + \left\| \frac{1}{\sqrt{l}} \int_{\Omega_2} u_2 \chi \,\mathrm{d}x \right\|_2^2 \\
&\geq C \sum_{i=1}^{2} \|\nabla u_i\|^2_{L^2(\Omega_i)} \\
&\geq C \sum_{i=1}^{2} \|u_i\|^2_{H^1(\Omega_i)} \\
&= C \|u\|^2_* 
\end{aligned}$$

The last inequality is a generalized Poincaré inequality, which holds, since for $u = (u_1, u_2) \in \mathcal{U}^t$ both $u_i$ have zero outer boundary value (and the boundary is smooth, nonempty and has a positive Hausdorff measure).

For $u, v \in \mathcal{U}^t$, we have

$$\begin{aligned}
|a(u,v)| &= \left| \sum_{i=1}^{2} \int_{\Omega_i} \nu_i \nabla u_i \cdot \nabla v_i \,\mathrm{d}x + \frac{1}{l} \int_{\Omega_2} v_2 \chi^T \,\mathrm{d}x \int_{\Omega_2} u_2 \chi \,\mathrm{d}x \right| \\
&\leq \|\nu\|_{L^\infty(\Omega)} \sum_{i=1}^{2} \|\nabla u_i\|_{L^2(\Omega_i)} \|\nabla v_i\|_{L^2(\Omega_i)} \\
&\quad + \frac{1}{l} \|\chi\|^2_{L^2(\Omega_2)} \|u_2\|_{L^2(\Omega_2)} \|v_2\|_{L^2(\Omega_2)} \\
&\leq \|\nu\|_{L^\infty(\Omega)} \sum_{i=1}^{2} \|u_i\|_{H^1(\Omega_i)} \|v_i\|_{H^1(\Omega_i)} \\
&\quad + \frac{1}{l} \|\chi\|^2_{L^2(\Omega_2)} \left( \|u_2\|_{H^1(\Omega_2)} \|v_2\|_{H^1(\Omega_2)} + \|u_1\|_{H^1(\Omega_1)} \|v_1\|_{H^1(\Omega_1)} \right) \\
&= \left( \|\nu\|_{L^\infty(\Omega)} + \frac{1}{l} \|\chi\|^2_{L^2(\Omega_2)} \right) \sum_{i=1}^{2} \|u_i\|_{H^1(\Omega_i)} \|v_i\|_{H^1(\Omega_i)} \\
&\leq C \left( \|u_1\|^2_{H^1(\Omega_1)} + \|u_2\|^2_{H^1(\Omega_2)} \right)^{1/2} \left( \|v_1\|^2_{H^1(\Omega_1)} + \|v_2\|^2_{H^1(\Omega_2)} \right)^{1/2} \\
&= C \|u\|_* \|v\|_*
\end{aligned}$$

□

The bilinear form $m$ cannot be coercive on $\mathcal{U}^t$ since it does not capture the behavior of a function outside of $\mathcal{C}$, while the norm of $\mathcal{U}^t$ does, but we make the following assumption:

**Assumption 3.5.** *We assume that the parameters of the machine are such that the following form is coercive on $L^2(\mathcal{C})$:*

$$m(u,v) = (\sigma u, v)_\Omega - \int_{\Omega_1} \sigma v \xi^T \,\mathrm{d}x A^T (AGA^T)^{-1} A \int_{\Omega_1} \sigma u \xi \,\mathrm{d}x,$$

*meaning, there is a $C > 0$, such that*

$$m(u,u) \geq C \|u\|^2_{L^2(\mathcal{C})} \text{ for all } u \in L^2(\mathcal{C}).$$

*$G$ is the diagonal DC conductance matrix, which has $\int_{\Omega_1} \sigma \xi_i \xi_i \ell_z \,\mathrm{d}x$, $i = 1, \ldots, 40$ on its first 40 diagonal elements followed by 40 entries for the resistance of the bars outside of the FE model and $2 \cdot 40$ entries for the resistance of the parts of the outer ring of the squirrel cage, see (3.5.1).*

*Note: In our concrete examples we will consider later, this assumption is fulfilled.*

It is difficult to give conditions when the assumption holds, since even when the matrix $G$ has easy structure, the multiplication by $A$ and $A^T$ and the following inversion makes the matrix dense with elements being polynomials of the dimension of $AA^T$.

We define

$$F : \mathcal{U}^t \times L^2(0,s;L^2(\Omega_2)) \times [t_0,s] \to \mathbb{R},$$
$$F(v,z,t) := \left(\chi^T, v_2\right)_{\Omega_2} \left( \int_{t_0}^{t} \frac{1}{l} e^{-\frac{r}{l}(t-\tau)} \left( \frac{r}{l} \int_{\Omega_2} z(x,\tau)\chi \,\mathrm{d}x + q \right) \mathrm{d}\tau \right.$$
$$\left. + e^{-\frac{r}{l}(t-t_0)} \left( i_0 + \frac{1}{l} \int_{\Omega_2} u_2^0 \chi \,\mathrm{d}x \right) \right)$$

and now are able to write equation (3.14) as

$$m(\partial_t u(t), v(t)) + a(u(t), v(t)) = F(v(t), z_2, t), \forall v \in \mathcal{V}. \tag{3.15}$$

We further define

$$F_i(z,t) := \left( \int_{t_0}^{t} \frac{1}{l} e^{-\frac{r}{l}(t-\tau)} \left( \frac{r}{l} \int_{\Omega_2} z(x,\tau)\chi \,\mathrm{d}x + q \right) \mathrm{d}\tau \right.$$
$$\left. + e^{-\frac{r}{l}(t-t_0)} \left( i_0 + \frac{1}{l} \int_{\Omega_2} u_2^0 \chi \,\mathrm{d}x \right) \right).$$

**Definition 3.6.** *A function $u \in L^2(0,T;\mathcal{U}^t) \cap L^\infty(0,T;L^2(\mathcal{C}))$ is called weak solution to (3.9) if it satisfies the weak form (3.14) for all $v \in \mathcal{V}$ and for almost all $t \in [0,T]$ and the initial conditions are fulfilled.*

### 3.8.2 A-priori estimates

To get estimates for the solution, we plug in $u$ as a test function. Note, that by

$$\left( \sigma \frac{\partial u}{\partial t}(t), u(t) \right)_\Omega = \frac{1}{2} \frac{\mathrm{d}}{\mathrm{d}t} \|\sqrt{\sigma} u(t)\|^2_{L^2(C)}$$

and

$$\left( E^T \int_\Omega \sigma u(t) \xi \,\mathrm{d}x, E^T \int_\Omega \sigma \frac{\partial u}{\partial t}(t) \xi \,\mathrm{d}x \right)_2 = \frac{1}{2} \frac{\mathrm{d}}{\mathrm{d}t} \|E^T \int_\Omega \sigma u(t) \xi \,\mathrm{d}x\|_2^2,$$

we have $m(\partial_t u(t), u(t)) = \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t} m(u(t), u(t))$, such that we have

$$\begin{aligned} &\int_{t_0}^{s} m(\partial_t u(t), u(t))\,\mathrm{d}t = \int_{t_0}^{s} \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t} m(u(t), u(t))\,\mathrm{d}t \\ &= \frac{1}{2} m(u(s), u(s)) - \frac{1}{2} m(u(t_0), u(t_0)) \\ &\geq C_1 \|u(s)\|^2_{L^2(\mathcal{C})} - C_2 \|u(t_0)\|_{L^2(\mathcal{C})}, \end{aligned} \tag{3.16}$$

where $2C_1$ is the coercivity and $2C_2$ the continuity constant of $m$. We can use the coercivity of $a$ to obtain

$$\int_{t_0}^{s} a(u(t), u(t))\,\mathrm{d}t \geq C \int_{t_0}^{s} \|u(t)\|^2_*\,\mathrm{d}t. \tag{3.17}$$

In the following we will write for a function $f : X \to \mathbb{R}^3$

$$\|f\|_{L^2(\Omega)} = \sqrt{\int_\Omega f_1^2 + f_2^2 + f_3^2\,\mathrm{d}x}.$$

For the estimate of the right hand side, we have

$$\begin{aligned} |F(u(t), z_2, t)| = \Bigg| &\left(\chi^T, u_2(t)\right)_{\Omega_2} \left( \int_{t_0}^{t} \frac{1}{l} e^{-\frac{r}{l}(t-\tau)} \left( \frac{r}{l} \int_{\Omega_2} z_2(x,\tau)\chi\,\mathrm{d}x + q \right) \mathrm{d}\tau \right. \\ &\left. + e^{-\frac{r}{l}(t-t_0)} \left( i_0 + \frac{1}{l} \int_{\Omega_2} u_2^0 \chi\,\mathrm{d}x \right) \right) \Bigg| \\ \leq \|\chi\|_{L^2(\Omega_2)} &\|u_2(t)\|_{L^2(\Omega_2)} \Bigg( \left\| \frac{1}{l} e^{-\frac{r}{l}(t-\tau)} \right\|_{L^2(t_0,t)} \left( \left\| \frac{r}{l}\chi \right\|_{L^2(\Omega_2)} \|z_2\|_{L^2(t_0,t;L^2(\Omega_2))} \right. \\ &\left. + \|q\|_{L^2(t_0,t)} \right) + \left| e^{-\frac{r}{l}(t-t_0)} \right| \left( \|i_0\| + \frac{1}{l} \left\| \int_{\Omega_2} \chi u_2^0 \right\| \right) \Bigg). \end{aligned} \tag{3.18}$$

With Hölder's inequality we obtain

$$\left\| \frac{1}{l} e^{-\frac{r}{l}(t-\tau)} \right\|_{L^2(t_0,t)} \leq \sqrt{\int_{t_0}^{t} 1\,\mathrm{d}t}\, \operatorname{ess\,sup}_{\tau\in[t_0,t]} \frac{1}{l} e^{-\frac{r}{l}(t-\tau)} = \frac{\sqrt{t-t_0}}{l}.$$

We integrate (3.18) over $[t_0, s]$ and estimate the four summands in the right hand side of (3.18) with $C > 0$ we obtain for the first summand

$$\begin{aligned} & C \int_{t_0}^{s} \|u_2(t)\|_{L^2(\Omega_2)} \sqrt{t - t_0} \|z_2\|_{L^2(t_0,t;L^2(\Omega_2))} \,\mathrm{d}t \\ & \leq C\sqrt{s - t_0} \|z_2\|_{L^2(t_0,s;L^2(\Omega_2))} \int_{t_0}^{s} \|u_2(t)\|_{L^2(\Omega_2)} \,\mathrm{d}t \\ & \leq C(s - t_0) \|z_2\|_{L^2(t_0,s;L^2(\Omega_2))} \|u_2\|_{L^2(t_0,s;L^2(\Omega_2))} \\ & \leq \varepsilon \|u_2\|^2_{L^2(t_0,s;L^2(\Omega_2))} + \frac{C^2(s - t_0)^2}{4\varepsilon} \|z_2\|^2_{L^2(t_0,s;L^2(\Omega_2))}. \end{aligned} \tag{3.19}$$

Where we have used two times Hölder's inequality and Young's inequality with $\varepsilon > 0$. We can estimate the second summand exactly like the first (the $C$'s are not the same)

$$\begin{aligned} & C \int_{t_0}^{s} \|u_2(t)\|_{L^2(\Omega_2)} \sqrt{t - t_0} \|q\|_{L^2(t_0,t)} \,\mathrm{d}t \\ & \leq \varepsilon \|u_2\|^2_{L^2(t_0,s;L^2(\Omega_2))} + \frac{C^2(s - t_0)^2}{4\varepsilon} \|q\|^2_{L^2(t_0,s)}. \end{aligned}$$

For the third summand we get

$$\begin{aligned} & C\|i_0\| \int_{t_0}^{s} \|u_2(t)\|_{L^2(\Omega_2)} \left| e^{-\frac{r}{l}(t-t_0)} \right| \,\mathrm{d}t \\ & \leq C \left| e^{-\frac{r}{l}(t_0-t_0)} \right| \|i_0\| \int_{t_0}^{s} \|u_2(t)\|_{L^2(\Omega_2)} \,\mathrm{d}t \\ & \leq C\|i_0\| \sqrt{s - t_0} \|u_2\|_{L^2(t_0,s;L^2(\Omega_2))} \\ & \leq \varepsilon \|u_2\|^2_{L^2(t_0,s;L^2(\Omega_2))} + \frac{C^2(s - t_0)}{4\varepsilon} \|i_0\|^2. \end{aligned}$$

Estimating the fourth summand leads to

$$\begin{aligned} & C \left\| \int_{\Omega_2} \chi u_2^0 \,\mathrm{d}x \right\| \int_{t_0}^{s} \|u_2(t)\|_{L^2(\Omega_2)} \left| e^{-\frac{r}{l}(t-t_0)} \right| \,\mathrm{d}t \\ & \leq \varepsilon \|u_2\|^2_{L^2(t_0,s;L^2(\Omega_2))} + \frac{C^2(s - t_0)}{\varepsilon} \left\| \int_{\Omega_2} \chi u_2^0 \,\mathrm{d}x \right\|^2. \end{aligned}$$

We can further estimate

$$\varepsilon \|u_2\|^2_{L^2(0,s;L^2(\Omega_2))} \leq \varepsilon C \|u\|^2_{L^2(0,s;\mathcal{U}^t)} \tag{3.20}$$

and absorp it into the left hand side by choosing $\varepsilon$ small enough.

In total we get the estimate

$$\begin{aligned} &\|u\|^2_{L^\infty(t_0,s;L^2(\mathcal{C}))} + \|u\|^2_{L^2(t_0,s;\mathcal{U}^t)} \\ &\leq C(s)\Big(\|u_0\|^2_{L^2(\mathcal{C})} + \|q\|^2_{L^2(0,s)} + \|z_2\|^2_{L^2(0,s;L^2(\Omega_2))} \\ &\quad + \|i_0\|^2 + \left\|\int_{\Omega_2} \chi u_2^0 \,\mathrm{d}x\right\|^2\Big). \end{aligned} \tag{3.21}$$

Here $C(s) > 0$ is a positive constant depending on $s$.

To estimate the time derivative we observe

$$\begin{aligned} |m(\partial_t u(t), v(t))| &= |-a(u(t), v(t)) + F(v(t), z_2, t)| \\ &\leq C\|u(t)\|_*\|v(t)\|_* + C_1\|\chi\|_{L^2(\Omega_2)}\|v(t)\|_*\|F_i(z_2,t)\|_2. \end{aligned} \tag{3.22}$$

Hence

$$\|m(\partial_t u, \cdot)\|_{L^2(t_0,s;\mathcal{U}_t^*)} \leq C\left(\|u\|_{L^2(t_0,s;\mathcal{U}_t)} + \|F_i(z_2,\cdot)\|_{L^2(t_0,s)}\right),$$

where $\mathcal{U}_t^*$ is the dual space of $\mathcal{U}^t$. Therefore

$$\begin{aligned} \|m(\partial_t u, \cdot)\|_{L^2(t_0,s;\mathcal{U}_t^*)} \leq C(s)\Big(&\|u_0\|^2_{L^2(\mathcal{C})} + \|q\|^2_{L^2(0,s)} + \|z_2\|^2_{L^2(0,s;L^2(\Omega_2))} \\ &+ \|i_0\|^2 + \left\|\int_{\Omega_2} \chi u_2^0 \,\mathrm{d}x\right\|^2\Big) \end{aligned} \tag{3.23}$$

using the previous estimates.

Let $K \subset\subset \mathcal{C}$ be arbitrary with Lipschitz boundary. We show that

$$\partial_t u \in L^2(t_0, s; H^1(K)^*).$$

Since $\operatorname{dist}(\partial K, \partial\mathcal{C}) \geq \delta > 0$, we find a bounded extension operator

$$E : H^1(K) \to H_0^1(\mathcal{C}).$$

We know that

$$\|u\|_{L^2(t_0,s;H^1(K))} \leq C\|u\|_{L^2(t_0,s;\mathcal{U}^t)}.$$

We use for arbitrary $w \in L^2(t_0, s; H^1(K))$ the test function

$$v(t) = Ew(t)$$

in (3.22). Since $v \in L^2(t_0, s; \mathcal{U}^t)$ with

$$\|v\|_{L^2(t_0,s;\mathcal{U}^t)} = \|v\|_{L^2(t_0,s;H_0^1(\mathcal{C}))},$$

we see that

$$\begin{aligned}\int_{t_0}^{s} |m(\partial_t u, w)| \,\mathrm{d}t &\leq \int_{t_0}^{s} |m(\partial_t u, v)| \,\mathrm{d}t \\ &\leq C\|Ew\|_{L^2(t_0,s;H^1(K))}(\|u\|_{L^2(t_0,s;\mathcal{U}^t)} + \|F_i\|_{L^2(t_0,s)}) \\ &\leq C\|E\|_{L(H^1(K),H_0^1(\mathcal{C}))}\|w\|_{L^2(t_0,s;H^1(K))}(\|u\|_{L^2(t_0,s;\mathcal{U}^t)} + \|F_i\|_{L^2(t_0,s)}).\end{aligned}$$

Now consider $L^2(K)$ with scalar product $m(\cdot,\cdot)$ and the Gelfand triple

$$H^1(K) \hookrightarrow L^2(K) \hookrightarrow H^1(K)^*.$$

Then we have $\partial_t u \in L^2(t_0, s; H^1(K)^*)$, $u \in L^2(t_0, s; H^1(K))$ and thus $u \in C([t_0, s]; L^2(K))$.

### 3.8.3 Galerkin approximation

Since we have a parabolic equation in the conducting region $\mathcal{C}$ and an elliptic equation in $\mathcal{I}$, we need to consider this in the finite dimensional approximation. We start with the two spaces $H_0^1(\Omega)$ and $H_0^1(\mathcal{I})$. As both of these spaces are separable Hilbert spaces, we can find nested finite dimensional subsets

$$\begin{aligned} V_N \subset V_{N+1} \subset H_0^1(\mathcal{I}), \quad \bigcup_{N\geq 1} V_n \text{ dense in } H_0^1(\mathcal{I}), \\ W_N \subset W_{N+1} \subset H_0^1(\Omega), \quad \bigcup_{N\geq 1} W_N \text{ dense in } H_0^1(\Omega). \end{aligned}$$

The functions in $V_N$ are extended to $\Omega$ by 0 and we choose a basis for $W_N$, such that if for an element $w \in W_N$ it holds $\operatorname{supp} w \subseteq \mathcal{I}$, then it also has to hold $w \in \operatorname{span}(V_N)$. We can choose both of these basis to be $L^2$ orthonormal and $H^1$ orthogonal by the choice of the normalized eigenfunctions of the Laplace operator ($-\Delta u = \lambda u$, see [41]). Fixing $N \in \mathbb{N}$ we can use the Gram-Schmidt process to generate a set $V_N \cup \tilde{W}_N \subset H_0^1(\Omega)$ which suffices $\operatorname{span}(W_N) = \operatorname{span}(V_N \cup \tilde{W}_N)$ and $(v_N, \tilde{w}_N)_{L^2(\Omega)} = 0$ for all $v_N \in V_N$ and $\tilde{w}_N \in \tilde{W}_N$, i.e. vectors in $V_N$ and $\tilde{W}_N$ are $L^2(\Omega)$ orthogonal. Vectors $w_N \in W_N$ which lie in the span of $V_N$ are being removed such that the vectors in $\tilde{W}_N$ are linearly independent.

We can then use the rotation operator on the finite dimensional space $R(t)(V_N \cup$

$\tilde{W}_N$) to get a basis for $\mathcal{U}_N^t$. A function $u_N : [t_0, s] \to \mathcal{U}_N^t$ can then be written as

$$u_N(t) = \sum_{i=1}^{N} \phi_j(t)\mathfrak{z}_j(t) \text{ with } \mathfrak{z}_j(t) \in \mathcal{U}_N^t.$$

Inserting the finite-dimensional approximation $u_N(t)$ into (3.15) and testing with the basis functions of $\mathcal{U}_N^t$ we consider for almost all $t \in [t_0, s]$

$$m(\partial_t u_N(t), \mathfrak{z}_j(t)) + a(u_N(t), \mathfrak{z}_j(t)) = F(\mathfrak{z}_j(t), z_2, t), \forall \mathfrak{z}_j \in \mathcal{U}_N^t. \tag{3.24}$$

For the time derivative we have

$$\partial_t u_N(x,t) = \sum_{i=1}^{N} \phi_j'(t)\mathfrak{z}_j(x,t) + \sum_{i=1}^{N} \phi_j(t)\partial_t\mathfrak{z}_j(x,t).$$

Since a function $\mathfrak{z}_j(t) \in \mathcal{U}_N^t$ corresponds to a function $\tilde{\mathfrak{z}}_j \in V_N \cup \tilde{W}_N$ which is rotated by $R(t)$, we have $\mathfrak{z}_j(x,t) = \tilde{\mathfrak{z}}_j(r_t x)$ for $x \in \Omega_1$ and $\partial_t\mathfrak{z}_j(x,t) = (r_t' x)^T \nabla \mathfrak{z}_j(x,t)$. Equation (3.24) can equivalently be written in matrix form

$$\begin{aligned} &\begin{pmatrix} 0 & 0 \\ 0 & M_W(t) \end{pmatrix} \frac{\mathrm{d}}{\mathrm{d}t} \begin{pmatrix} \phi_v(t) \\ \phi_w(t) \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & \tilde{M}_W(t) \end{pmatrix} \begin{pmatrix} \phi_v(t) \\ \phi_w(t) \end{pmatrix} \\ &+ \begin{pmatrix} A_{VV}(t) & A_{VW}(t) \\ A_{WV}(t) & A_{WW}(t) \end{pmatrix} \begin{pmatrix} \phi_v(t) \\ \phi_w(t) \end{pmatrix} = \begin{pmatrix} J_v(t) \\ J_w(t) \end{pmatrix}. \end{aligned} \tag{3.25}$$

The entries of $\tilde{M}_W(t)$ are given by $m(\partial_t\mathfrak{z}_j(t), \mathfrak{z}_i(t)) = m((r_t' x)^T \nabla \mathfrak{z}_j(t), \mathfrak{z}_i(t))$ with $\mathfrak{z}_i(t), \mathfrak{z}_j(t) \in R(t)(\tilde{W}_N)$. We choose a $p(t) = \sum_{j=}^{N} \phi_j(t) v_j(t)$ with $v_j(t) \in R(t)(V_N)$ and $\|p(t)\|_* \neq 0$. It holds that $p(t) \in \mathcal{U}_N^t \subset \mathcal{U}^t$ and we have by the coercivity of $a$ on $\mathcal{U}^t$ for a.e. $t \in [t_0, s]$

$$0 < C\|p(t)\|_*^2 \leq a(p(t), p(t)) = \phi(t) A_{VV}(t) \phi(t)$$

since $p(t)$ was chosen arbitrary, $A_{VV}(t)$ is symmetric positive definite and invertible for a.a. $t \in [t_0, s]$. In the same way we can use Assumption 3.5 to show the invertibility of $M(t)$. By the invertibility of $A_{VV}(t)$ we can get from the first line in (3.25) $\phi_v(t)$ dependent on $\phi_w(t)$

$$\phi_v(t) = A_{VV}^{-1}(t)(-A_{VW}(t)\phi_w(t) + J_v(t))$$

and plug it into the second line, which is then (since $M_W(t)$ is invertible) an ordinary

differential equation for $\phi_w(t)$. We supply the ODE with initial values

$$\phi_j(t_0) = (u_1^0, w_j(t_0))_{\mathcal{C}},\ w_j(t_0) \in R(t_0)(\tilde{W}_N),\ j = 1, \dots, N.$$

As the time dependence of the basis functions in $\mathcal{U}_N^t$ is due to the rotation $\theta(t)$ which was assumed to be $H^2(\mathbb{R})$, we have that the involved matrices in (3.25) depend continuously on $t$. Also the right hand side depends continuously on $t$, since $q \in C^\infty(\mathbb{R}, \mathbb{R}^3)$ and $\int_{t_0}^t \frac{r}{l} e^{-\frac{r}{l}(t-\tau)} \frac{1}{l} \int_{\Omega_2} z_2(x,\tau)\chi \,\mathrm{d}x \,\mathrm{d}t \in C^0([t_0, s]; \mathbb{R}^3)$ due to $z_2 \in L^2(t_0, s; L^2(\Omega_2))$. The Theorem 2.11 of Picard-Lindelöf guarantees a solution for the linear ODE.

Note that equation (3.24) has the same structure as (3.15) and we get the same estimates for $u_N(t)$ (3.21) and the time derivative (3.23). Since the involved spaces are Hilbert spaces and therefore reflexive and the approximation solution is bounded, we can choose weakly convergent subsequences $u_{N_l}$ and $\int_{t_0}^s m(\partial_t u_{N_l}, \cdot) \,\mathrm{d}t$, see [41], such that

$$u_{N_l} \rightharpoonup u \text{ in } L^2(t_0, s; \mathcal{U}^t), \quad \int_{t_0}^s m(\partial_t u_{N_l}, \cdot) \,\mathrm{d}t \rightharpoonup \int_{t_0}^s m(\partial_t u, \cdot) \,\mathrm{d}t \text{ in } L^2(t_0, s; \mathcal{U}_t^*).$$

We choose a function $v(t) = \sum_{j=1}^N d_j(t)\mathfrak{z}_j(x,t)$, where $d_k(t)$ are smooth functions and $\mathfrak{z}_j(t) \in \mathcal{U}_N^t$. We test equation (3.24) with $v(t)$, integrate over $[t_0, s]$ and obtain for $\tilde{N} \geq N$

$$\int_{t_0}^s m(\partial_t u_{\tilde{N}}(t), v(t)) \,\mathrm{d}t + \int_{t_0}^s a(u_{\tilde{N}}(t), v(t)) \,\mathrm{d}t = \int_{t_0}^s F(v(t), z_2, t) \,\mathrm{d}t. \tag{3.26}$$

We set $\tilde{N} = \tilde{N}_l$ and pass to the the weak limit, since $v \in L^2(t_0, s; \mathcal{U}^t)$, we have

$$\int_{t_0}^s m(\partial_t u(t), v(t)) \,\mathrm{d}t + \int_{t_0}^s a(u(t), v(t)) \,\mathrm{d}t = \int_{t_0}^s F(v(t), z_2, t) \,\mathrm{d}t. \tag{3.27}$$

As the $d_k(t)$ are chosen arbitrary

$$m(\partial_t u(t), \mathfrak{z}(t)) + a(u(t), \mathfrak{z}(t)) = F(\mathfrak{z}(t), z_2, t) \tag{3.28}$$

holds for almost every $t \in (t_0, s)$ for every $\mathfrak{z}(t) \in \mathcal{U}_N^t$ for all $N \geq 1$. Since $\bigcup_{N\in\mathbb{N}} \mathcal{U}_N^t$ is dense in $\mathcal{U}^t$, equation (3.28) holds for each $\mathfrak{z}(t) \in \mathcal{U}^t$.

To show that the initial condition $\sigma(x)u_1(x, t_0) = \sigma(x)u_1^0(x)$, $x \in \Omega$ is fulfilled in the limit we test (3.27) with $v \in C^1(t_0, s; \mathcal{U}^t)$ with $v(s) = 0$ and use integration by

parts in the first summand to obtain

$$\int_{t_0}^{s} -m(u(t), \partial_t v(t))\,\mathrm{d}t + \int_{t_0}^{s} a(u(t), v(t))\,\mathrm{d}t = \int_{t_0}^{s} F(v(t), z_2, t)\,\mathrm{d}t + m(u(t_0), v(t_0)). \tag{3.29}$$

Similarly for (3.26) we obtain

$$\begin{aligned}\int_{t_0}^{s} -m(u_{\tilde{N}}(t), \partial_t v(t))\,\mathrm{d}t + \int_{t_0}^{s} a(u_{\tilde{N}}(t), v(t))\,\mathrm{d}t \\ = \int_{t_0}^{s} F(v(t), z_2, t)\,\mathrm{d}t + m(u_{\tilde{N}}(t_0), v(t_0)).\end{aligned} \tag{3.30}$$

We set $\tilde{N} = \tilde{N}_l$ and pass to the weak limit

$$\int_{t_0}^{s} -m(u(t), \partial_t v(t))\,\mathrm{d}t + \int_{t_0}^{s} a(u(t), v(t))\,\mathrm{d}t = \int_{t_0}^{s} F(v(t), z_2, t)\,\mathrm{d}t + m(u_1^0, v(t_0)), \tag{3.31}$$

since $u_{\tilde{N}_l}(t_0) \to u_1^0$ in $L^2(\mathcal{C})$. Due to $v(t_0)$ being arbitrary, comparing (3.31) and (3.29), we conclude $u_1(x, t_0) = u_1^0(x)$ for $x \in \mathcal{C}$.

We have shown that for arbitrary $z_2 \in L^2(t_0, s; L^2(\Omega_2))$ in the right hand side $F(v(t), z_2, t)$ equation (3.15) has a solution $u \in L^\infty(t_0, s; L^2(\mathcal{C})) \cap L^2(t_0, s; \mathcal{U}^t))$ for almost all $t \in [t_0, s]$ and all $v(t) \in \mathcal{V}$. We will now use a fixed point argument to prove that a solution exists for the right hand side $F(v(t), u_2, t)$. Let $z_2, \tilde{z}_2 \in L^2(t_0, s; L^2(\Omega_2))$ and $u$ and $\tilde{u}$ the corresponding solutions to equation (3.15) with right hand side $F(v(t), z_2, t)$ and $F(v(t), \tilde{z}_2, t)$. We substract the equations for $u$ and $\tilde{u}$ and since $F(\cdot, \cdot, \cdot)$ is linear in the second argument, we obtain in the right hand side

$$\begin{aligned}&F(v(t), z_2, t) - F(v(t), \tilde{z}_2, t) \\ &\qquad = (\chi^T, v_2(t))_{\Omega_2} \left( \int_{t_0}^{t} \frac{r}{l^2} e^{-\frac{r}{l}(t-\tau)} \int_{\Omega_2} \chi \left( z_2(x, \tau) - \tilde{z}_2(x, \tau) \right) \mathrm{d}x\,\mathrm{d}\tau \right).\end{aligned}$$

Since $a$ and $m$ are bilinear forms, we obtain on the left hand side

$$\begin{aligned}&m(\partial_t(u(t) - \tilde{u}(t)), v(t)) + a(u(t) - \tilde{u}(t), v(t)) \\ &\qquad = (\chi^T, v_2(t))_{\Omega_2} \left( \int_{t_0}^{t} \frac{r}{l^2} e^{-\frac{r}{l}(t-\tau)} \int_{\Omega_2} \chi \left( z_2(x, \tau) - \tilde{z}_2(x, \tau) \right) \mathrm{d}x\,\mathrm{d}\tau \right).\end{aligned} \tag{3.32}$$

Using $v(t) = u(t) - \tilde{u}(t)$ as test function in (3.32), integrating from $t_0$ to $s$ and using

the estimates (3.16), (3.17), (3.19) and (3.20) leads to

$$\|u - \tilde{u}\|_{L^\infty(t_0,s;L^2(\mathcal{C}))\cap L^2(t_0,s;\mathcal{U}^t))} \leq C(s-t_0)\|z_2 - \tilde{z}_2\|_{L^2(t_0,s;L^2(\Omega))}. \tag{3.33}$$

By the continuous embedding

$$L^\infty(t_0,s;L^2(\mathcal{C}))\cap L^2(t_0,s;\mathcal{U}^t)) \hookrightarrow L^2(t_0,s;L^2(\Omega_2)),$$

we can estimate the right hand side of (3.33) with a positive constant $\tilde{C}$

$$C(s-t_0)\|z_2 - \tilde{z}_2\|_{L^2(t_0,s;L^2(\Omega))} \leq \tilde{C}C(s-t_0)\|z_2 - \tilde{z}_2\|_{L^\infty(t_0,s;L^2(\mathcal{C}))\cap L^2(t_0,s;\mathcal{U}^t))},$$

to obtain in total

$$\|u - \tilde{u}\|_{L^\infty(t_0,s;L^2(\mathcal{C}))\cap L^2(t_0,s;\mathcal{U}^t))} \leq \tilde{C}C(s-t_0)\|z_2 - \tilde{z}_2\|_{L^\infty(t_0,s;L^2(\mathcal{C}))\cap L^2(t_0,s;\mathcal{U}^t))}. \tag{3.34}$$

This shows that the solution map $S : X \to X$, where $X = L^\infty(0,t;L^2(C)) \cap L^2(0,t;\mathcal{U}^t)$, which maps $z_2 \in X$ to the solution $u \in X$ is a contraction for $s_1 := t < \frac{1}{C\tilde{C}} + t_0$. Banach's fixed-point theorem guarantees a solution of

$$m(\partial_t u(t), v(t)) + a(u(t), v(t)) = F(v(t), u_2, t) \tag{3.35}$$

for almost all $t \in [t_0, s_1]$ and all $v(t) \in \mathcal{V}$. We can then solve equation (3.35) on $[t_0, s_1]$ and iteratively extend the solution by using $u_1(x, s_1)$, $x \in \mathcal{C}$ and $\int_{\Omega_2} \chi u_2(x, s_1)\,\mathrm{d}x$ as initial value to solve (3.35) on the interval $[s_1, 2s_1 - t_0]$. To do this, we need to show that the solution is continuous in time in the rotor bars and stator windings.

### 3.8.4 Improved regularity

Since we have used integration by parts in the beginning of the proof (3.11), we need to prove $\|\partial_t u_2\|_{L^2(t_0,s;L^2(\Omega_2))} \leq C$, since then $u_2$ has a continuous representative in $C(t_0,s;L^2(\Omega_2))$. To show, that the solution of our state equation is in fact more regular, we will differentiate the state equation with respect to time and get estimates in spaces with higher regularity. In order to show the improved regularity, we need better regularity of the data. We need, that the initial data is bounded in $H^2(\mathcal{C})$, such that we obtain initial values for the differentiated equation by the original equation. We further need that the time derivative of the right hand side is bounded in $L^2(t_0,s;L^2(\Omega_2))$. Due to the fixed point, we get a solution of our equation only on a short time interval and iteratively have to extend the solution to

larger time intervals. This means, we need to use our solution at given time points as initial values to extend the solution, we thus also need that $u(t) \in H^2(\mathcal{C})$. To obtain $H^2$ results by applying elliptic regularity theory (see (2.13)), we assume $\nu \in C^1(\Omega)$. We further need the time derivative $\partial_t u$ to be bounded in $L^2([t_0, s]; L^2(\mathcal{C}))$, since for the application of Theorem 2.13 we need for almost all $t \in [t_0, s]$ bounds for the right hand side of $a(u(t), v(t)) = (f(t), v(t))_{\Omega_2} - m(\partial_t u(t), v(t))$ of the form

$$\begin{aligned} |a(u(t), v(t))| &= |(f(t), v(t))_{\Omega_2} - m(\partial_t u(t), v(t))| \\ &\leq C(\|f(t)\|_{L^2(\Omega_2)} + \|\partial_t u(t)\|_{L^2(\mathcal{C})})\|v(t)\|_{L^2(\Omega_2)}. \end{aligned}$$

We start by showing, that the derivative in time of the right hand side is bounded in $L^2(t_0, s; L^2(\Omega_2))$. The right hand side is given by

$$\begin{aligned} f(x, t; q, z) = \chi^T(x) \Bigg( \int_{t_0}^{t} \frac{1}{l} e^{-\frac{r}{l}(t-\tau)} \left( \frac{r}{l} \int_{\Omega_2} z_2(\tilde{x}, \tau)\chi(\tilde{x}) \, \mathrm{d}\tilde{x} + q(\tau) \right) \mathrm{d}\tau \\ + e^{-\frac{r}{l}(t-t_0)} \left( i_0 + \frac{1}{l} \int_{\Omega_2} u_2^0 \chi(\tilde{x}) \, \mathrm{d}\tilde{x} \right) \Bigg) \end{aligned}$$

By inspecting for $g \in L^2([t_0, t+\varepsilon])$

$$\begin{aligned} &\frac{1}{h} \left( \int_{t_0}^{t+h} e^{-t-h+\tau} g(\tau) \, \mathrm{d}\tau - \int_{t_0}^{t} e^{-t+\tau} g(\tau) \, \mathrm{d}\tau \right) \\ &= \frac{1}{h} \Bigg( \int_{t_0}^{t+h} e^{-t-h+\tau} g(\tau) \, \mathrm{d}\tau - \int_{t_0}^{t} e^{-t-h+\tau} g(\tau) \, \mathrm{d}\tau \\ &\quad + \int_{t_0}^{t} e^{-t-h+\tau} g(\tau) \, \mathrm{d}\tau - \int_{t_0}^{t} e^{-t+\tau} g(\tau) \, \mathrm{d}\tau \Bigg) \\ &= \frac{1}{h} \left( \int_{t}^{t+h} e^{-t-h+\tau} g(\tau) \, \mathrm{d}\tau + \int_{t_0}^{t} \left( e^{-t-h+\tau} - e^{-t+\tau} \right) g(\tau) \, \mathrm{d}\tau \right) \\ &\quad \xrightarrow[h \to 0]{a.e.} g(t) + \int_{t_0}^{t} \partial_t \left( e^{-t+\tau} \right) g(\tau) \, \mathrm{d}\tau \end{aligned}$$

we see, that the weak derivative in time of the right hand side is given by

$$\begin{aligned}\partial_t f(x,t;q,z) = \chi^T(x)\Bigg( &\int_{t_0}^{t} -\frac{r}{l^2} e^{-\frac{r}{l}(t-\tau)} \left( \frac{r}{l} \int_{\Omega_2} z_2(\tilde{x},\tau)\chi(\tilde{x})\,\mathrm{d}\tilde{x} + q(\tau) \right) \mathrm{d}\tau \\ &+ \frac{r}{l} \int_{\Omega_2} z_2(\tilde{x},t)\chi(\tilde{x})\,\mathrm{d}\tilde{x} + q(t) \\ &- \frac{r}{l} e^{-\frac{r}{l}(t-t_0)} \left( i_0 + \frac{1}{l} \int_{\Omega_2} u_2^0 \chi(\tilde{x})\,\mathrm{d}\tilde{x} \right) \Bigg) \\ = &-\frac{r}{l} f(x,t;q,z) + \chi^T(x) \left( \frac{r}{l} \int_{\Omega_2} z_2(\tilde{x},t)\chi(\tilde{x})\,\mathrm{d}\tilde{x} + q(t) \right).\end{aligned}$$

We know $\|f(x,t;q,z)\|^2_{L^2(t_0,s;L^2(\Omega_2))} \leq C$ already. We can further estimate

$$\begin{aligned}&\int_{t_0}^{s} \int_{\Omega_2} (\chi^T q)^2 \,\mathrm{d}t\,\mathrm{d}x \\ \leq &\int_{t_0}^{s} \int_{\Omega_2} \|\chi\|_2^2 \|q\|_2^2 \,\mathrm{d}t\,\mathrm{d}x \\ \leq &\int_{\Omega_2} \|\chi\|_2^2 \,\mathrm{d}x \int_{t_0}^{s} \|q\|_2^2 \,\mathrm{d}t \\ = &\, C\|q\|^2_{L^2(t_0,s)}\end{aligned}$$

and

$$\begin{aligned}&\int_{t_0}^{s} \int_{\Omega_2} \left( \chi^T(x) \frac{r}{l} \int_{\Omega_2} z_2(\tilde{x},t)\chi(\tilde{x})\,\mathrm{d}\tilde{x} \right)^2 \mathrm{d}t\,\mathrm{d}x \\ \leq &\int_{t_0}^{s} \int_{\Omega_2} \left\| \frac{r}{l}\chi \right\|_2^2 \left\| \int_{\Omega_2} z_2(\tilde{x},t)\chi(\tilde{x})\,\mathrm{d}\tilde{x} \right\|_2^2 \mathrm{d}t\,\mathrm{d}x \\ \leq &\int_{\Omega_2} \|\chi\|_2^2 \,\mathrm{d}x \int_{t_0}^{s} \|\chi\|^2_{L^2(\Omega_2)} \|z_2\|^2_{L^2(\Omega_2)} \,\mathrm{d}t \\ \leq &\, C\|z_2\|^2_{L^2(t_0,s;L^2(\Omega_2))}.\end{aligned}$$

In total we get

$$\begin{aligned}\|\partial_t f(x,t;q,z)\|^2_{L^2(t_0,s;L^2(\Omega_2))} \leq C(s)\Big( &\|q\|^2_{L^2(0,s)} \\ &+ \|z_2\|^2_{L^2(0,s;L^2(\Omega_2))} \\ &+ \|i_0\|_2^2 + \| \int_{\Omega_2} \chi u_2^0 \,\mathrm{d}x\|_2^2 \Big).\end{aligned}$$

Note that we can choose $z_2 = u_2$, since we already know that $\|u_2\|_{L^2(t_0,s;L^2\Omega))}$ is

bounded.

Introducing

$$\begin{aligned} U_1(r_t x, t) = u_1(x,t),\ U_2(x,t) = u_2(x,t),\\ V_1(r_t x, t) = v_1(x,t),\ V_2(x,t) = v_2(x,t),\\ U = U_1 1_{\Omega_1} + U_2 1_{\Omega_2},\\ V = V_1 1_{\Omega_1} + V_2 1_{\Omega_2}. \end{aligned}$$

We have

$$U_1(t,x) = U_2(t,x) \text{ and } V_1(t,x) = V_2(t,x) \text{ for } x \in \Gamma$$

and if $u, v \in \mathcal{U}^t$ then $U, V \in H_0^1(\Omega)$. Moreover,

$$\partial_t u_1(x,t) = \partial_t U_1(r_t x, t) + \nabla U_1(r_t x, t)^T r_t' x, \quad \nabla u_1(t,x) = r_t^T \nabla U_1(t, r_t x).$$

We define the bilinear form $m$ with rotating data to be

$$\begin{aligned} m_R(u,v;t) &:= (\sigma(r_{-t}x)u, v)_{\Omega_1} \\ &- \int_{\Omega_1} \sigma(r_{-t}x) v \xi(r_{-t}x)^T \,\mathrm{d}x A^T (AGA^T)^{-1} A \int_{\Omega_1} \sigma(r_{-t}x) u \xi(r_{-t}x) \,\mathrm{d}x. \end{aligned}$$

Hence, the weak formulation can be written as (we use that integration with respect to $r_t x$ over $\Omega_1$ yields the same as integration with respect to $x$, since the rotation $r_t$ has determinant 1) for almost all $t \in [t_0, s]$

$$\begin{aligned} &m_R(\partial_t U_1(t), V_1; t) + (\nu(r_{-t}x) r_t^T \nabla U_1(t), r_t^T \nabla V_1)_{\Omega_1} + m_R(\nabla U_1(t)^T r_t' r_{-t} x, V_1; t) = 0,\\ &(\nu \nabla U_2(t), \nabla V_2)_{\Omega_2} + \frac{1}{l} \int_{\Omega_2} \chi^T U_2(t) \,\mathrm{d}x \int_{\Omega_2} \chi^T V_2 \,\mathrm{d}x = (f(x,t;q,U_2), V_2)_{\Omega_2}. \end{aligned}$$

Using that $r_t$ is orthogonal we can simplify to

$$\begin{aligned} &m_R(\partial_t U_1(t), V_1; t) + (\nu(r_{-t}x) \nabla U_1(t), \nabla V_1)_{\Omega_1} + m_R(\nabla U_1(t)^T r_t' r_{-t} x, V_1; t) = 0,\\ &(\nu \nabla U_2(t), \nabla V_2)_{\Omega_2} + \frac{1}{l} \int_{\Omega_2} \chi^T U_2(t) \,\mathrm{d}x \int_{\Omega_2} \chi^T V_2 \,\mathrm{d}x = (f(x,t;q,U_2), V_2)_{\Omega_2}. \end{aligned} \tag{3.36}$$

In an open neighborhood $\Omega'$ of the annular domain $S$ around the interface we have $\sigma = 0$, $\chi = 0$, $f = 0$ and $\nu$ is constant and thus (3.36) yields for almost all $t \in [t_0, s]$

$$(\nu \nabla U(t), \nabla V) = 0 \quad \forall V \in H_0^1(\Omega').$$

Hence, elliptic regularity yields

$$\|U\|_{L^2(t_0,s;H^2(S))} \leq C\|f\|_{L^2(t_0,s;L^2(\Omega_2))}.$$

The same bounds hold for $\|u_1\|_{L^2(t_0,s;H^2(S\cap\Omega_1))}$ and $\|u_2\|_{L^2(t_0,s;H^2(S\cap\Omega_2))}$.

We derive now formally a weak formulation for the time derivative and study its regularity. Differentiating

$$u_1(r_{-t}x,t) = u_2(x,t)$$

yields

$$\partial_t u_1(r_{-t}x,t) - \nabla u_1(r_{-t}x,t)^T r'_{-t}x = \partial_t u_2(x,t). \tag{3.37}$$

By the $H^2$ regularity on the interface, we know that

$$\nabla u_1(r_{-t}x,t)^T r'_{-t}x \in L^2(t_0,s;H^{1/2}(\Gamma)).$$

Let $w_1 \in L^2(t_0,s;H^1(\Omega_1))$ be an extension of $\nabla u_1(r_{-t}x,t)^T r'_{-t}x \in L^2(t_0,s;H^{1/2}(\Gamma))$. Now let $\psi \in C_c(\Omega')$ with $\psi \in [0,1]$ and $\psi|_S \equiv 1$ and set

$$w = \psi w_1.$$

Then $w \in L^2(t_0,s;H^1(\Omega_1))$ and vanishes on the support of $\sigma$. Now consider

$$u' := (\partial_t u_1 - w)1_{\Omega_1} + \partial_t u_2 1_{\Omega_2},$$

which by construction fulfills

$$u'_1(r_{-t}x,t) = u'_2(x,t),\ x \in \Gamma.$$

Therefore, by differentiating (3.13) with respect to $t$ and using the fact that the jump in (3.37) is corrected in $u'$, we obtain the following weak formulation satisfied by $u'$. For all $v(t) \in \mathcal{V}$ and almost all $t \in [t_0,s]$

$$m(\partial_t u'(t), v_1(t)) + a(\nabla u'(t), \nabla v_1(t)) = -(\nu\nabla w(t), \nabla v_1(t))_{\Omega_1} + (\partial_t f(x,t;q,u_2), v_2(t))_{\Omega_2}. \tag{3.38}$$

If the initial data is sufficiently smooth (the inital data for $t_0$ is 0 and therefore sufficiently smooth), we obtain initial data for $u'$ from the PDAE, i.e.

$$\sigma\frac{\partial u'}{\partial t}(x,t_0) - \sigma\xi^T B\int_{\Omega_1}\sigma\frac{\partial u'}{\partial t}(x,t_0)\xi\ell_z\,\mathrm{d}x = \nabla\cdot(\nu_1\nabla u^0),\ x \in \mathcal{C}.$$

Since $w \in L^2(H^1(\Omega_1))$, the linear functional $v_1 \mapsto (\nu\nabla w, \nabla v_1)_{\Omega_1}$ can be represented in $L^2(t_0,s;\mathcal{U}_t^*)$ and hence we obtain as above for $u$ the usual regularity of weak

solutions for $u'$. Since $\partial_t u_1 = u' + w$, we recover the same regularity for $\partial_t u_1$. Especially we have $\|\partial_t u_1\|^2_{L^2(t_0;s;L^2(\Omega_1))} \leq C$ and $\|\partial_t u_2\|^2_{L^2(t_0;s;L^2(\Omega_2))} \leq C$. With an Galerkin approximation the whole procedure can be justified rigorosly.

The regularity $\|\partial_t u_2\|^2_{L^2(t_0;s;L^2(\Omega_2))} \leq C$ combined with $\|u_2\|^2_{L^2(t_0;s;L^2(\Omega_2))} \leq C$ implies that $u_2$ has a continuous representative in $C([t_0,s];L^2(\Omega_2))$, which we needed in (3.11) to use integration by parts. The regularity $\|\partial_t u_1\|^2_{L^2(t_0;s;L^2(\Omega_1))} \leq C$ allows us to use elliptic regularity theory and obtain $u(t) \in H^2(\mathcal{C})$ for almost all $t \in [t_0,s]$, which we need to supply inital values for the differentiated equation.

#### Dependence on rotation

To make it clear, that the space $\mathcal{U}^t$ depends on a rotation, we denote it here for different rotations $\theta$ and $\tilde{\theta}$ as $\mathcal{U}^\theta$ and $\mathcal{U}^{\tilde{\theta}}$. The corresponding weak solutions to the field and circuit coupled equations are given by $u \in L^2(0,T;\mathcal{U}^\theta)$ and $\tilde{u} \in L^2(0,T;\mathcal{U}^{\tilde{\theta}})$. We define the difference $\delta\theta(t) := \theta(t) - \tilde{\theta}(t)$. We again choose continuous differentiable $\psi \geq 0$ with $\psi \equiv 1$ on $S$ and $\psi \equiv 0$ on $\Omega \backslash \Omega'$.

We use the test function for the equation for $u$

$$1_{\Omega_2}(u_2(t) - \tilde{u}_2(t)) + 1_{\Omega_1}\left((u_1(t) - \tilde{u}_1(t))(1-\psi) + \psi(u_1(t) - \tilde{u}_1(r_{\delta\theta(t)}x,t))\right)$$

which lies in $\mathcal{U}^\theta$. On the domain $\Omega_1$ we obtain for the elliptic part

$$\begin{aligned}
&\int_{\Omega_1} \nu_1 \nabla u_1(t) \cdot \nabla \left((u_1(t) - \tilde{u}_1(t))(1-\psi) + \psi(u_1(t) - \tilde{u}_1(r_{\delta\theta(t)}x,t))\right) \mathrm{d}x \\
= &\int_{\Omega_1} \nu_1 \nabla u_1(t) \cdot \nabla \left(u_1(t) - \tilde{u}_1(t) + \psi(\tilde{u}_1(t) - \tilde{u}_1(r_{\delta\theta(t)}x,t))\right) \mathrm{d}x.
\end{aligned}$$

We use the test function for the equation for $\tilde{u}$

$$1_{\Omega_2}(\tilde{u}_2(t) - u_2(t)) + 1_{\Omega_1}\left((\tilde{u}_1(t) - u_1(t))(1-\psi) + \psi(\tilde{u}_1(t) - u_1(r_{-\delta\theta(t)}x,t))\right)$$

on the domain $\Omega_1$ we obtain for the elliptic part

$$\begin{aligned}
&\int_{\Omega_1} \nu_1 \nabla \tilde{u}_1(t) \cdot \nabla \left((\tilde{u}_1(t) - u_1(t))(1-\psi) + \psi(\tilde{u}_1(t) - u_1(r_{-\delta\theta(t)}x,t))\right) \mathrm{d}x \\
= &\int_{\Omega_1} \nu_1 \nabla \tilde{u}_1(t) \cdot \nabla \left(\tilde{u}_1(t) - u_1(t) + \psi(u_1(t) - u_1(r_{-\delta\theta(t)}x,t))\right) \mathrm{d}x.
\end{aligned}$$

Adding the equations for $u$ and $\tilde{u}$ leads to

$$\begin{aligned}
&m(\partial_t(u_1(t)-\tilde{u}_1(t),u_1(t)-\tilde{u}_1(t)))+a(u(t)-\tilde{u}(t),u(t)-\tilde{u}(t))\\
&+\int_{\Omega_1}\nu_1\nabla u_1(t)\cdot\nabla\left(\psi(\tilde{u}_1(t)-\tilde{u}_1(r_{\delta\theta(t)}x,t))\right)\mathrm{d}x\\
&+\int_{\Omega_1}\nu_1\nabla\tilde{u}_1(t)\cdot\nabla\left(\psi(u_1(t)-u_1(r_{-\delta\theta(t)}x,t))\right)\mathrm{d}x\\
=\ &\left(\chi^T,u(t)-\tilde{u}(t)\right)_{\Omega_2}\int_0^T\frac{r}{l^2}e^{-\frac{r}{l}(T-\tau)}\int_{\Omega_2}\chi(u_2(\cdot,\tau)-\tilde{u}_2(\cdot,\tau))\,\mathrm{d}x\,\mathrm{d}t.
\end{aligned}\tag{3.39}$$

We estimate the second and third line in (3.39) in the same way. We have

$$\begin{aligned}
&\int_{\Omega_1}\nu_1\nabla u_1(t)\cdot\nabla\left(\psi(\tilde{u}_1(t)-\tilde{u}_1(r_{\delta\theta(t)}x,t))\right)\mathrm{d}x\\
=&\int_{\Omega_1}\nu_1\nabla u_1(t)\cdot\nabla\psi\left(\tilde{u}_1(t)-\tilde{u}_1(r_{\delta\theta(t)}x,t)\right)\mathrm{d}x\\
&+\int_{\Omega_1}\nu_1\nabla u_1(t)\cdot\nabla\left(\tilde{u}_1(t)-\tilde{u}_1(r_{\delta\theta(t)}x,t))\right)\psi\,\mathrm{d}x.
\end{aligned}$$

The function $J:H^2(0,T)\to L^2(0,T;L^2(\Omega_1))$, $\theta\mapsto\left(\tilde{u}_1(\cdot,\cdot)-\tilde{u}_1(r_{\theta(\cdot)}\cdot,\cdot)\right)$ is continuous since $\tilde{u}_1\in L^2(0,T;H^1(\Omega_1))$ and has the bounded derivative

$$\theta\mapsto-\nabla\tilde{u}_1(r_{\theta(\cdot)}\cdot,\cdot)^T r'_{\theta(\cdot)}\cdot,$$

therefore we have for almost all $x$

$$\tilde{u}_1(x,t)-\tilde{u}_1(r_{\delta\theta(t)}x,t)=\int_0^{\delta\theta(t)}\nabla\tilde{u}_1(r_s x,t)^T r'_s x\,\mathrm{d}s.$$

We can estimate

$$\begin{aligned}
\|\tilde{u}_1(\cdot,t)-\tilde{u}_1(r_{\delta\theta(t)}\cdot,t)\|_{L^2(\Omega_1)}&\le\int_0^{\delta\theta(t)}\|\nabla\tilde{u}_1(r_s\cdot,t)^T r'_s\cdot\|_{L^2(\Omega_1)}\,\mathrm{d}s\\
&\le\int_0^{\delta\theta(t)}\|\nabla\tilde{u}_1(r_s\cdot,t)\|_{L^2(\Omega_1)}\|r'_s\cdot\|_{L^\infty(\Omega_1)}\,\mathrm{d}s\\
&\le C\int_0^{\delta\theta(t)}1\,\mathrm{d}s\|\nabla\tilde{u}_1(\cdot,t)\|_{L^2(\Omega_1)}\\
&=C\delta\theta(t)\|\nabla\tilde{u}_1(\cdot,t)\|_{L^2(\Omega_1)}.
\end{aligned}$$

And further

$$\begin{aligned}\|\tilde{u}_1(\cdot,\cdot) - \tilde{u}_1(r_{\delta\theta(\cdot)}\cdot,\cdot)\|^2_{L^2(0,T;L^2(\Omega_1))} &\leq \|C\delta\theta(\cdot)\|\nabla\tilde{u}_1(\cdot,\cdot)\|_{L^2(\Omega_1)}\|^2_{L^2(0,T)} \\ &\leq C(T)\|\delta\theta\|^2_{L^\infty(0,T)},\end{aligned}$$

which leads to

$$\int_0^T \int_{\Omega_1} \nu_1 \nabla u_1 \cdot \nabla\psi \left(\tilde{u}_1 - \tilde{u}_1(r_{\delta\theta(\cdot)}x,\cdot)\right) \mathrm{d}x\,\mathrm{d}t \leq C(T)\|\delta\theta\|^2_{L^\infty(0,T)}.$$

The same argument holds for $J_2 : H^2(0,T) \to L^2(0,T;L^2(\Omega_1)), \theta \mapsto \nabla\left(\tilde{u}_1(\cdot,\cdot) - \tilde{u}_1(r_{\theta(\cdot)}\cdot,\cdot)\right)$ since we have $\tilde{u}_1 \in L^2(0,T;H^2(\Omega_1))$. The derivative of the mapping is given by

$$\theta \mapsto -\left(\nabla^2\tilde{u}_1(r_{\theta(\cdot)}\cdot,\cdot)r'_{\theta(\cdot)}\cdot\right)^T r_{\theta(\cdot)}\cdot - \nabla\tilde{u}_1(r_{\theta(\cdot)}\cdot,\cdot)^T r'_{\theta(\cdot)}\cdot\,.$$

By using the estimate (3.34) we obtain in total for $T$ small enough

$$\|u-\tilde{u}\|^2_{L^\infty(0,T;L^2(\mathcal{C}))} + \int_0^T \|u-\tilde{u}\|^2_*\,\mathrm{d}t \leq C(T)\|\delta\theta\|^2_{L^\infty(0,T)}.$$

For the torque we have with $c = l_z \cdot r \cdot \nu_{air}$ and the second coupling condition

$$\begin{aligned}M_{em}(u(t),\theta(t)) &= c\int_\Gamma n\cdot\nabla u_2(t) n^\perp \cdot \nabla u_1(r_{-\theta(t)}x,t)\,\mathrm{d}S \\ &= c\int_\Gamma n\cdot\nabla u_1(r_{-\theta(t)}x,t) n^\perp \cdot \nabla u_1(r_{-\theta(t)}x,t)\,\mathrm{d}S \\ &= c\int_\Gamma n_\theta\cdot\nabla u_1(t) n_\theta^\perp \cdot \nabla u_1(t)\,\mathrm{d}S,\end{aligned} \tag{3.40}$$

where $n_{\theta(t)} := n(r_{\theta(t)}x)$ and $n^\perp_{\theta(r)} := n^\perp(r_{\theta(t)}x)$. We can compute the torque by evaluating the integral in (3.40) over spheres with radii from $r-\epsilon$ to $r$ with $\epsilon > 0$ and then average with $\frac{1}{\epsilon}$. $\varepsilon$ is chosen such that the sphere with radius $r-\epsilon$ is contained in the physical airgap. This method is related to Arkkio's method [2]. We have with $\Gamma^\epsilon := \{x \in \mathbb{R}^2 : r-\epsilon \leq \|x\|_2 \leq r\}$

$$M_{em}(u(t),\theta(t)) = l_z\nu_{air}\frac{1}{\epsilon}\int_{\Gamma^\epsilon} \|x\|_2 n_\theta\cdot\nabla u_1(t) n_\theta^\perp\cdot\nabla u_1\,\mathrm{d}x.$$

The difference of $M_{em}(u(t),\theta(t))$ and $M_{em}(\tilde{u}(t),\tilde{\theta}(t))$, where $u$ and $\tilde{u}$ have been attained by solving the field circuit coupled problem with rotation $\theta$ and $\tilde{\theta}$ can be

estimated

$$
\begin{aligned}
& l_z \nu_{air} \frac{1}{\epsilon} \int_{\Gamma^\epsilon} \|x\|_2 n_{\theta(t)} \cdot \nabla u_1(t) n_{\theta(t)}^\perp \cdot \nabla u_1(t) \, \mathrm{d}x \\
& - l_z \nu_{air} \frac{1}{\epsilon} \int_{\Gamma^\epsilon} \|x\|_2 n_{\tilde{\theta}(t)} \cdot \nabla \tilde{u}_1(t) n_{\tilde{\theta}(t)}^\perp \cdot \nabla \tilde{u}_1(t) \, \mathrm{d}x \\
= & \, l_z \nu_{air} \frac{1}{\epsilon} \int_{\Gamma^\epsilon} \|x\|_2 \Big( n_{\theta(t)} \cdot \nabla u_1(t) n_{\theta(t)}^\perp \cdot \nabla u_1(t) - n_{\tilde{\theta}(t)} \cdot \nabla \tilde{u}_1(t) n_{\tilde{\theta}(t)}^\perp \cdot \nabla \tilde{u}_1(t) \Big) \, \mathrm{d}x \\
= & \, l_z \nu_{air} \frac{1}{\epsilon} \int_{\Gamma^\epsilon} \|x\|_2 \Big( \Big( n_{\theta(t)} - n_{\tilde{\theta}(t)} \Big) \cdot \nabla u_1(t) n_{\theta(t)}^\perp \cdot \nabla u_1(t) \\
& + n_{\tilde{\theta}(t)} \cdot \left( \nabla u_1(t) - \nabla \tilde{u}_1(t) \right) n_{\theta(t)}^\perp \cdot \nabla u_1(t) \\
& + n_{\tilde{\theta}(t)} \cdot \nabla \tilde{u}_1(t) \Big( n_{\theta(t)}^\perp - n_{\tilde{\theta}(t)}^\perp \Big) \cdot \nabla u_1(t) \\
& + n_{\tilde{\theta}(t)} \cdot \nabla \tilde{u}_1(t) n_{\tilde{\theta}(t)}^\perp \cdot \left( \nabla u_1(t) - \nabla \tilde{u}_1(t) \right) \Big) \, \mathrm{d}x \\
\leq & \, C(t) |\delta\theta(t)|
\end{aligned}
$$

by the previous obtained bounds.

Since the right hand side of the equation of motion involves the spatial integral of the gradient of a $L^2(0,T;H^2(\Omega_1))$ function, the right hand side is in $L^2(0,T)$, therefore the solution of the second order ODE is in $H^2(0,T)$ and therefore in $C^1(0,T)$ (with possible modifiction on a set with measure zero). We estimate the solution of the equation of motion

$$I x''(t) + \beta x'(t) = M_{em}(u(t), \theta(t))$$

with

$$x(0) = 0 \text{ and } x'(0) = 0,$$

for $t \in [0,T]$. Integrating one time yields

$$\int_0^t I x''(s) \, \mathrm{d}x + \int_0^t \beta x'(s) \, \mathrm{d}x = I x'(t) + \beta x(t) = \int_0^t M_{em}(u(s), \theta(s)) \, \mathrm{d}s.$$

Integrating a second time yields

$$\int_0^t I x'(s) \, \mathrm{d}x + \int_0^t \beta x(s) \, \mathrm{d}x = I x(t) + \int_0^t \beta x(s) \, \mathrm{d}x = \int_0^t \int_0^z M_{em}(u(s), \theta(s)) \, \mathrm{d}s \, \mathrm{d}z,$$

such that we have the integral equation

$$x(t) = -\frac{\beta}{I} \int_0^t x(s) \, \mathrm{d}x + \frac{1}{I} \int_0^t \int_0^z M_{em}(u(s), \theta(s)) \, \mathrm{d}s \, \mathrm{d}z.$$

Taking the supremum of the absolute difference of the equation for $x(\cdot)$ and $\tilde{x}(\cdot)$ we can estimate

$$\begin{aligned}\frac{\beta}{I}\sup_{t\in[0,T]}\int_0^t |x(z)-\tilde{x}(z)|\,\mathrm{d}z &= \frac{\beta}{I}\int_0^T |x(z)-\tilde{x}(z)|\,\mathrm{d}z\\ &\leq T\frac{\beta}{I}\sup_{t\in[0,T]}|x(t)-\tilde{x}(t)|\\ &= T\frac{\beta}{I}\|x-\tilde{x}\|_\infty\end{aligned}$$

and

$$\begin{aligned}&\frac{1}{I}\sup_{t\in[0,T]}\int_0^t\int_0^z |M_{em}(u(s),\theta(s))-M_{em}(\tilde{u}(s),\tilde{\theta}(s))|\,\mathrm{d}s\,\mathrm{d}z\\ =\ &\frac{1}{I}\int_0^T\int_0^z |M_{em}(u(s),\theta(s))-M_{em}(\tilde{u}(s),\tilde{\theta}(s))|\,\mathrm{d}s\,\mathrm{d}z\\ \leq\ &T\frac{1}{I}\sup_{t\in[0,T]}\int_0^t |M_{em}(u(s),\theta(s))-M_{em}(\tilde{u}(s),\tilde{\theta}(s))|\,\mathrm{d}s\\ \leq\ &T^2\frac{1}{I}\sup_{t\in[0,T]}|M_{em}(u(t),\theta(t))-M_{em}(\tilde{u}(t),\tilde{\theta}(t))|.\end{aligned}$$

If $T<\frac{\beta}{I}$, one obtains with a positive constant $C>0$ the bound

$$\|x-\tilde{x}\|_\infty \leq CT^2\|\theta-\tilde{\theta}\|_\infty,$$

i.e. for $T$ being small enough the solution operator of the equation of motion is a contraction. This concludes the proof.

CHAPTER 4

# Discretization

In this chapter, the discretized version of the model of the asynchronous machine is described, which can be solved numerically. In the first section we consider the discretization in space by the finite element method (FEM). It is described how we treat the coupling conditions for different rotor positions and how we can compute the torque. Even with linearized material, the equation of state is nonlinear due to the torque calculation and the dependence on the rotor angle. In the second section, we describe the time-stepping schemes for linear and nonlinear material properties.

The discretization is based on the weak form for all $v \in \{v \in H^1(\Omega_1 \cup \Omega_2) : v(x) = 0 \text{ for } x \in \bar{\Omega} \backslash \{\Omega \cup \Gamma\}\}$

$$\int_{\Omega_1} \sigma \partial_t u_i v_i \,\mathrm{d}x + \sum_{i=1}^{2} \int_{\Omega_i} \nu_i \nabla u_i \cdot \nabla v_i \,\mathrm{d}x + \sum_{i=1}^{2} (-1)^i \int_{\Gamma} \nu_i \partial_n u_i v_i \,\mathrm{d}S$$
$$= (\sigma \xi A^T \varphi, v_1)_{\Omega_1} + (\chi^T i_{st}, v_2)_{\Omega_2} \qquad i = 1, 2 \quad (4.1)$$
$$u_1(r_{-\theta(t)} x, t) = u_2(x, t), \qquad x \in \Gamma \quad (4.2)$$
$$\nu_1 \partial_n u_1(r_{-\theta(t)} x, t) = \nu_2 \partial_n u_2(x, t), \qquad x \in \Gamma \quad (4.3)$$
$$-A \int_{\Omega_1} \ell_z \sigma \partial_t u_1 \xi^T \,\mathrm{d}x + AGA^T \varphi = 0 \quad (4.4)$$
$$\int_{\Omega_2} \ell_z \partial_t u_2 \chi^T \,\mathrm{d}x + R_{st} i_{st} + L \frac{\mathrm{d}}{\mathrm{d}t} i_{st} = q \quad (4.5)$$
$$\frac{\mathrm{d}}{\mathrm{d}t} \theta = \dot{\theta} \quad (4.6)$$
$$I \frac{\mathrm{d}}{\mathrm{d}t} \dot{\theta} + \beta \dot{\theta} = M_{em} \quad (4.7)$$
$$u(x, 0) = 0, \qquad x \in \Omega \quad (4.8)$$

$$u(x,t) = 0, \qquad x \in \bar{\Omega} \backslash \{\Omega \cup \Gamma\} \tag{4.9}$$

$$\theta(0) = 0 \tag{4.10}$$

$$\dot{\theta}(0) = 0 \tag{4.11}$$

$$i_{st}(0) = 0 \tag{4.12}$$

## 4.1 Finite element discretization

The code for the simluation of the induction machine is adapted from a code of Herbert De Gersem. The implemented asynchronous machine was first described by De Weerdt in his PhD-thesis [35]. Due to the symmetry of the motor, it is sufficient to consider only a quarter of the total cross-section of the motor and to introduce antiperiodic boundary conditions at the emerging artificial boundaries. Since we are now looking at a quarter, the number of rotor bars considered is reduced from 40 to 10. To discretize our problem in space, we are using the finite element method [24, 16]. The state variable which depends on space is the magnetic vector potential, which we call $u := A_z$ in this chapter. In order to apply the method, the stator and rotor domain are separated in a way, that they are no longer touching, i.e. the stator is pushed outwards, such that the airgap on rotor and stator do not overlap and it holds $\bar{\Omega}_s \cap \bar{\Omega}_r = \emptyset$. We then divide the region of the stator $\Omega_s$ and the region of the rotor $\Omega_r$ into triangles $T$ to get a polygonal approximation

$$\Omega_s^h := \bigcup_{T \in \mathcal{T}_s} T \subset \mathbb{R}^2$$

and

$$\Omega_r^h := \bigcup_{T \in \mathcal{T}_r} T \subset \mathbb{R}^2$$

where $\mathcal{T}$ is the collection the closed triangles $T$ with the properties

- triangles in $\mathcal{T}$ with $T \cap T' \neq \emptyset$ and $T \neq T'$ either intersect in a shared vertex or in an entire shared edge,
- it holds

  $$\max_{T \in \mathcal{T}} \frac{h_T}{\rho_T} \leq c$$

  for some $c > 0$ and $\rho_T$ is the radius of the incircle and $h_T$ the radius of the circumscribed circle of an element $T \in \mathcal{T}$,

- it holds

$$\max_{T\in\mathcal{T}} h_T \leq c \min_{T\in\mathcal{T}} h_T.$$

Since we are using interpolation as a coupling technique, we choose triangulations for $\Omega_s^h$ and $\Omega_r^h$, which have coinciding polygonal boundary on the interface in the airgap on stator $\Gamma^s$ and in the airgap on the rotor $\Gamma^r$ and the nodes on the airgap boundaries are equidistantly placed. Additionaly, we need, that there is a one to one correspondence of the nodes on the sides of the artificial boundaries to apply antiperiodic boundary conditions, see Figure 4.1. We choose a triangulation, which matches with the material, meaning in every triangle, we have only one material coefficient. We define the whole approximated domain to be

$$\Omega^h := \Omega_s^h \cup \Omega_r^h.$$

In the FEM method, to approximate the PDE solution, one uses element-wise polynomials which are globally continuous. We define the spaces of piecewise linear finite elements on the stator and rotor as

$$\mathcal{V}_s^h = \left\{ v_s^h \in C(\Omega_s^h) : v_s^h\big|_T \in \mathcal{P}_2^1(T), T \in \mathcal{T}_s \right\}$$

and

$$\mathcal{V}_r^h = \left\{ v_r^h \in C(\Omega_r^h) : v_r^h\big|_T \in \mathcal{P}_2^1(T), T \in \mathcal{T}_r \right\}.$$

Where the space of polynomials of degree less or equal to $d$ with dimension $n$ is denoted by $\mathcal{P}_n^d$. We define the combined space to be

$$\mathcal{V}^h := \mathcal{V}_s^h \times \mathcal{V}_r^h. \tag{4.13}$$

Note that this space is not coupled yet and the two functions can have different function values on $\Gamma^r$ and $\Gamma^s$. To define a basis of $\mathcal{V}^h$ we use the vertices of $\mathcal{T}_s$ and $\mathcal{T}_r$. We make distinct index sets for the nodes on the airgap, since they are involved in the coupling. We denote

$$\begin{aligned} &\mathcal{N}_r \cup \mathcal{N}_{\Gamma^r} \text{ are all the } n_r \text{ nodes in } \Omega_r^h, \\ &\mathcal{N}_s \cup \mathcal{N}_{\Gamma^s} \text{ are all the } n_s \text{ nodes in } \Omega_s^h, \\ &\mathcal{N} := \mathcal{N}_r \cup \mathcal{N}_{\Gamma^r} \cup \mathcal{N}_s \cup \mathcal{N}_{\Gamma^s}. \end{aligned} \tag{4.14}$$

The index set $\mathcal{N}_r$ contains all the nodes of the rotor domain $\Omega_r^h$, which are not used in the interface coupling, as these are contained in $\mathcal{N}_{\Gamma^r}$. We define the elementwise

linear nodal basis functions $\tilde{N}_i$ as follows:

$$\tilde{N}_i(x_j) = \delta_{ij},\ x_j \in \mathcal{N},\ i,j = 1,\dots,n_s+n_r.$$

In each triangle 3 basis functions are non vanishing, they are given by

$$\tilde{N}_i(x_1,x_2) = \frac{a_i + b_i x_1 + c_i x_2}{2\Delta},\ i = 1,2,3,$$

where $\Delta$ is the area of the triangle $T$ and the coefficients can be computed by

$$\begin{aligned} a_i &= x_1^j x_2^k - x_1^k x_2^j, \\ b_i &= x_2^j - x_2^k, \\ c_i &= x_1^k - x_1^j. \end{aligned}$$

Here $(x_1^i, x_2^i)$, $(x_1^j, x_2^j)$ and $(x_1^k, x_2^k)$ are the coordinates of the nodes of the triangle. The formula for the coefficients is due to linear polynomial interpolation in the triangle. Since the quantity $u$ we want to solve for is the $z$ component of the magnetic vector potential $\vec{A}$ and we only consider the cross-section of the machine, we scale the basis functions by the length of the machine $\ell_z$ and define $N_i = \frac{\tilde{N}_i}{\ell_z}$. Our discretized space is then given by

$$\mathcal{V}_h = \operatorname{span}\{N_1,\dots,N_{n_s+n_r}\} \subset H^1(\Omega_s^h \cup \Omega_r^h).$$

We are approximating the magnetic vector potential via the ansatz

$$u_{r,f}^{h,s}(x,t) = \sum_{i=1}^{n_r} u_i^{h,s}(t) N_i(x) \quad \text{and} \quad u_{s,f}^{h,s}(x,t) = \sum_{i=n_r+1}^{n_s+n_r} u_i^{h,s}(t) N_i(x) \tag{4.15}$$

and use $\mathcal{V}_h$ as the space of test functions in the weak form. Variables with $h,s$ index denote semi-discrete quantities which depend on time. Since the basis is finite, we

test with the basis elements. In case of linear material we have

$$\begin{aligned}
((M_1^h))_{ij} &= \int_{\Omega_r^h} \sigma \ell_z N_i N_j \,\mathrm{d}x, && i,j = 1,\dots,n_r,\\
((K_1^h))_{ij} &= \int_{\Omega_r^h} \nu_1 \ell_z \nabla N_i \cdot \nabla N_j \,\mathrm{d}x, && i,j = 1,\dots,n_r,\\
((K_2^h))_{ij} &= \int_{\Omega_s^h} \nu_2 \ell_z \nabla N_i \cdot \nabla N_j \,\mathrm{d}x, && i,j = n_r+1,\dots,n_s+n_r,\\
((\tilde{Q}^h))_{ij} &= \int_{\Omega_r^h} \sigma \ell_z \xi_j N_i \,\mathrm{d}x, && i = 1,\dots,n_r, j = 1,\dots,10,\\
((X^h))_{im} &= \int_{\Omega_s^h} \ell_z \chi_m N_i \,\mathrm{d}x, && i = n_r+1,\dots,n_s+n_r, m = 1,\dots,3,\\
((\tilde{G}))_{jj} &= \int_{\Omega_s^h} \ell_z \sigma \xi_j \xi_j \,\mathrm{d}x, && j = 1,\dots,10,\\
((B_1^h))_{ij} &= \int_{\Gamma^h} \nu \ell_z \partial_n N_i N_j \,\mathrm{d}S, && i,j = 1,\dots,n_r,\\
((B_2^h))_{ij} &= \int_{\Gamma^h} \nu \ell_z \partial_n N_i N_j \,\mathrm{d}S, && i,j = n_r+1,\dots,n_s+n_r.
\end{aligned} \tag{4.16}$$

To assemble the full DC conductance matrix of the rotor bar circuit, we set up the diagonal matrix $Z \in \mathbb{R}^{40\times 40}$ which contains the first 10 diagonal elements of $\tilde{G}$, followed by the 10 DC conductances of the rotor bar parts outside of the FE model and the last 20 diagonal elements are given by the DC conductances of the end ring parts of the squirrel cage. We define

$$G := AZA^T,$$

where $A \in \mathbb{R}^{29\times 40}$ is the reduced incidence matrix. The matrix $Q^h \in \mathbb{R}^{n_r\times 29}$ is defined to be the matrix

$$Q^h := \begin{pmatrix} \tilde{Q}^h & \mathrm{zeros}(n_r, 30) \end{pmatrix} A^T.$$

Note that since the basis functions in rotor and stator have no common support, we can simply combine the matrices for rotor and stator part and we can write for example

$$K^h = \begin{pmatrix} K_1^h & 0 \\ 0 & K_2^h \end{pmatrix} \text{ and } M^h = \begin{pmatrix} M_1^h & 0 \\ 0 & 0 \end{pmatrix}.$$

In the following we will use a specific ordering of the nodes and their basis functions, such that the coefficients in (4.15) have the order

$$u_{\mathcal{N}}^{h,s} = (u_r^{h,s}, u_{\Gamma^r}^{h,s}, u_s^{h,s}, u_{\Gamma^s}^{h,s})^T.$$

The spatial discretized system without applied boundary conditions for the equations (4.1),(4.4) and (4.5) looks like

$$\begin{aligned}
M_{rr}^h \partial_t u_r^{h,s} + K_{rr}^h u_r^{h,s} + K_{r\Gamma^r}^h u_{\Gamma^r}^{h,s} + Q_r^h \varphi &= 0 \\
K_{\Gamma^r r}^h u_r^{h,s} + (K_{\Gamma^r\Gamma^r}^h - B_{\Gamma^r\Gamma^r}^h) u_{\Gamma^r}^{h,s} &= 0 \\
K_{ss}^h u_s^{h,s} + K_{s\Gamma^s}^h u_{\Gamma^s}^{h,s} + X_s^h i &= 0 \\
K_{\Gamma^s s}^h u_s^{h,s} + (K_{\Gamma^s\Gamma^s}^h + B_{\Gamma^s\Gamma^s}^h) u_{\Gamma^s}^{h,s} &= 0 \\
-(Q_r^h)^T \partial_t u_r^{h,s} + G\varphi &= 0 \\
(X_s^h)^T \partial_t u_s^{h,s} + L\partial_t i_{st} + R_{st} i_{st} &= q,
\end{aligned} \tag{4.17}$$

Two subscripts at matrices denote, that the correpsonding rows and columns are chosen. Defining

$$K_{full} := \begin{pmatrix}
K_{rr}^h & K_{r\Gamma^r}^h & 0 & 0 & Q_r^h & 0 \\
K_{\Gamma^r r}^h & K_{\Gamma^r\Gamma^r}^h - B_{\Gamma^r\Gamma^r}^h & 0 & 0 & 0 & 0 \\
0 & 0 & K_{ss}^h & K_{s\Gamma^s}^h & 0 & X_s^h \\
0 & 0 & K_{\Gamma^s s}^h & K_{\Gamma^s\Gamma^s}^h + B_{\Gamma^s\Gamma^s}^h & 0 & 0 \\
0 & 0 & 0 & 0 & G & 0 \\
0 & 0 & 0 & 0 & 0 & R_{st}
\end{pmatrix},$$

$$M_{full} := \begin{pmatrix}
M_{rr}^h & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
-(Q_r^h)^T & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & (X_s^h)^T & 0 & 0 & L
\end{pmatrix}$$

and $b^{h,s}(t)$ being a vector of the size of the columns of $K_{full}$ with the last three entries being the right hand side $q$ we can write (4.17) as

$$M_{full}\partial_t y^{h,s}(t) + K_{full} y^{h,s}(t) = b^{h,s}(t), \tag{4.18}$$

where $y^{h,s}(t) = (u_r^{h,s}(t), u_{\Gamma^r}^{h,s}(t), u_s^{h,s}(t), u_{\Gamma^s}^{h,s}(t), \varphi(t), i_{st}(t))^T$. In the next section, we describe the coupling we are using and how the boundary conditions are applied.

## 4.2 Boundary and interface conditions

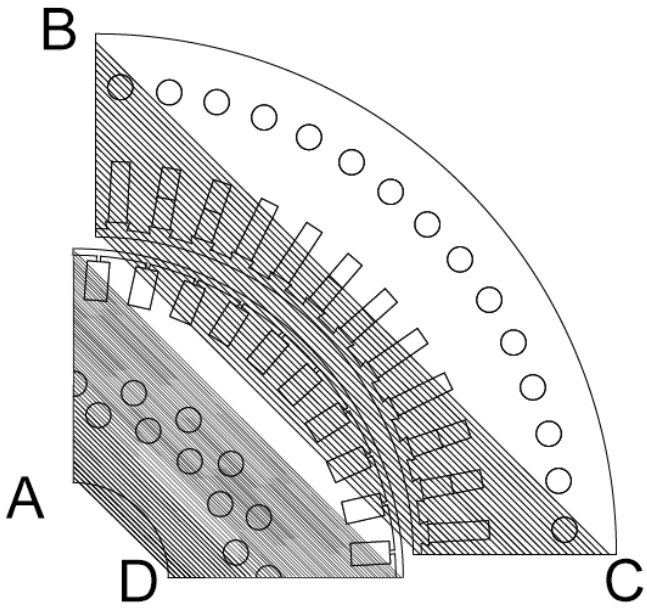


**Figure 4.1.** Antiperiodic boundary conditions.

Since we simulate only a quarter of the machine, we have two additional artificial edges that do not exist in the physical machine, in addition to natural ones. Using the symmetry of the machine, the correct choice for boundary conditions on the artificial boundaries are antiperiodic boundary conditions, which can be written as

$$u|_{AB} = -u|_{CD},$$

see Figure (4.1). On the natural outer boundaries of the rotor and stator, we have homogeneous dirichlet boundary conditions

$$u|_{BC} = u|_{DA} = 0.$$

To incorporate the boundary conditions into the discrete framework, we project the matrices (4.16) onto the degrees of freedom (dof) which is a subset of all the nodes. To do so, we choose one of the artificial boundaries as master and one as slave and set up two projection matrices $P_r \in \mathbb{R}^{|\mathcal{N}_r| \times |\mathcal{N}_{r,dof}|}$ and $P_s \in \mathbb{R}^{|\mathcal{N}_s| \times |\mathcal{N}_{s,dof}|}$. These projection matrices do not act on the airgap nodes and $\mathcal{N}_{r,dof}$ is the subset

of $\mathcal{N}_r$ containing the degrees of freedom contained in $\mathcal{N}_r$, anlogous for the stator. The projection matrix for the rotor (analogous for the stator) can be set up in the following way: Starting with the unit matrix $I \in \mathbb{R}^{|\mathcal{N}_r|\times|\mathcal{N}_r|}$, putting a $-1$ in the column/row of every master/slave pair of the antiperiodic boundary condition, then collecting the columns $\mathcal{N}_{r,dof}$. The only matrix in (4.16) in which the integrals involving the basis functions corresponding to nodes on the boundary do not vanish is the stiffness matrix, we can project it onto the degrees of freedom by

$$\begin{pmatrix} P_r^T & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & P_s^T & 0 \\ 0 & 0 & 0 & I \end{pmatrix} K^h \begin{pmatrix} P_r & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & P_s & 0 \\ 0 & 0 & 0 & I \end{pmatrix} = \begin{pmatrix} P_r^T K_{rr}^h P_r & 0 & 0 & 0 \\ 0 & K_{\Gamma^r\Gamma^r}^h & 0 & 0 \\ 0 & 0 & P_s^T K_{ss}^h P_s & 0 \\ 0 & 0 & 0 & K_{\Gamma^s\Gamma^s}^h \end{pmatrix},$$

where $P_s$ and $P_t$ are the projection matrices for rotor and stator nodes. For all the other matrices, we can simply choose the columns and rows of the dofs (in Matlab syntax), for example

$$P_r^T M_{rr}^h P_r = M_{rr}^h(\mathcal{N}_{r,dof}, \mathcal{N}_{r,dof}) \text{ and } P_r^T Q_r^h = Q_r^h(\mathcal{N}_{r,dof}, :).$$

Different methods for the coupling of the rotor and stator have been developed [30], methods where the grid between rotor and stator has to match are, for example, the locked step method [56], where only rotations with angles where the grid matches can be computed. Another method is the moving band approach [36], in which a tube is placed around the airgap and the grid is distorted within this tube during a rotation. Since the grid cannot be distorted arbitrarily, when a certain rotation angle is exceeded, the grid must jump, resulting in non-smooth behavior in the simulation. Methods for coupling where the grid of rotor and stator do not have to coincide are, for example, interpolation and so-called mortaring, which has been intensively researched in recent years and is described in more detail in section 4.2.1.

To incorporate the interface conditions for different angles we use a smooth interpolation. We will show that for some rotor positions this is equivalent to solving a saddle point problem that results from a weak coupling between rotor and stator where functions which are bi-orthogonal to the hat on the interface are used as testspace.

For the coupling of rotor and stator with interpolation, we construct a coupling matrix depending on the angle. In our simulation, the full airgap is divided into 360 parts. As we are simulating only one quarter of the machine, we are left with 91 nodes on the stator and 91 nodes on the rotor. One node on the boundary of the airgap is used in the anti-periodic boundary condition, leaving 90 nodes for the coupling. For every occuring rotation angle $\theta(t)$ we compute a full jump $n = \left\lfloor \frac{\theta(t)\cdot 360}{2\cdot\pi} \right\rfloor$ on the mesh intervals and a fractional jump $\epsilon = \frac{\theta(t)\cdot 360}{2\cdot\pi} - \left\lfloor \frac{\theta(t)\cdot 360}{2\cdot\pi} \right\rfloor$. When

using linear interpolation, one can then use the coupling matrix $R_{full}(\theta) \in \mathbb{R}^{360\times 360}$ (we drop here the dependence on $t$):

$$\begin{aligned} R_{full}(\theta) &\triangleq R_{full}\left(n = \left\lfloor \frac{\theta \cdot 360}{2\cdot\pi} \right\rfloor, \epsilon = \frac{\theta\cdot 360}{2\cdot\pi} - \left\lfloor \frac{\theta\cdot 360}{2\cdot\pi}\right\rfloor\right) \\ &:= \begin{pmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ 1 & & & 0 \end{pmatrix}^n \begin{pmatrix} 1-\epsilon & \epsilon & & \\ & \ddots & \ddots & \\ & & \ddots & \epsilon \\ \epsilon & & & 1-\epsilon \end{pmatrix} \end{aligned}$$

If $\epsilon = 0$, then we have matching nodes and the interpolation is exact. $R_{full}(\theta)$ then reduces to

$$\begin{pmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ 1 & & & 0 \end{pmatrix}^n,$$

which is an orthonormal matrix which shifts elements of vectors (or rows of matrices) $n$-times upwards. As we are simulating only a quarter of the machine and only 90 nodes are used in the coupling we define with $I \in \mathbb{R}^{90\times 90}$ being the unit marix

$$R(\theta) := \begin{pmatrix} I & 0 \end{pmatrix} R_{full}(\theta) \begin{pmatrix} I & -I & I & -I \end{pmatrix}^T \tag{4.19}$$

to account for the antiperiodic behavior of the quarters. The coupling condition (4.2) can then be written in discrete form

$$R(\theta(t)) u^{h,s}_{\Gamma^s}(t) = u^{h,s}_{\Gamma^r}(t).$$

Note that, in this case, the nodes on the interface part on rotor and stator need to have the same ordering. In our case the nodes are indexed counterclockwise. Then the coupling condition corresponds to a counterclockwise rotation of the rotor for a positive $\theta$. We take the nodal values of the stator airgap as degrees of freedom. The degrees of freedom of the magnetic vector potential are

$$u^{h,s}_{dof}(t) = (u^{h,s}_{r,dof}(t), u^{h,s}_{\Gamma^s}(t), u^{h,s}_{s,dof}(t))^T,$$

and we define

$$y^{h,s}_{dof}(t) := (u^{h,s}_{dof(t)}, \varphi(t), i_{st}(t))^T,$$

where $\varphi(t)$ is the vector of nodal potentials in the rotor bar network. We use the

projection

$$\mathcal{P}(\theta) := \begin{pmatrix} P_r & 0 & 0 & 0 & 0 \\ 0 & R(\theta) & 0 & 0 & 0 \\ 0 & 0 & P_s & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{pmatrix} \quad \text{and} \quad \mathcal{P}_u(\theta) := \begin{pmatrix} P_r & 0 & 0 \\ 0 & R(\theta) & 0 \\ 0 & 0 & P_s \\ 0 & I & 0 \end{pmatrix}.$$

We have $\mathcal{P}(\theta) \in \mathbb{R}^{|\mathcal{N}_r|+|\mathcal{N}_{\Gamma^r}|+|\mathcal{N}_s|+|\mathcal{N}_{\Gamma^s}|+n_\varphi+n_i \times |\mathcal{N}_{r,dof}|+|\mathcal{N}_{\Gamma^s}|+|\mathcal{N}_{s,dof}|+n_\varphi+n_i}$. $n_\varphi$ is the number of nodes in the rotor bar circuit, in our case 29 and $n_i$ is the number of current phases, in our case 3. $\mathcal{P}_u(\theta)$ is the projector for the magnetic vector potential and has $n_{dof} := |\mathcal{N}_{r,dof}| + |\mathcal{N}_{\Gamma^s}| + |\mathcal{N}_{s,dof}|$ columns.

The coupled space which also satisfies the outer boundary conditions then reads

$$\begin{aligned} \mathcal{V}_\theta^h := \Bigg\{ & \left( \sum_{i=1}^{n_r} u_i^h N^i(x), \sum_{i=n_r+1}^{n_s+n_r} u_i^h N^i(x) \right) \in H^1(\Omega_r^h) \times H^1(\Omega_s^h) \\ & : u^h = \mathcal{P}_u(\theta) u_{dof}^h,\ u_{dof}^h \in \mathbb{R}^{n_{dof}} \Bigg\}. \end{aligned} \tag{4.20}$$

Projecting the semi-discrete system (4.18) onto the degrees of freedom leads to

$$\mathcal{P}(\theta(t))^T M_{full} \mathcal{P}(\theta(t)) \partial_t y_{dof}^{h,s}(t) + \mathcal{P}(\theta(t))^T K_{full} \mathcal{P}(\theta(t)) y_{dof}^{h,s}(t) = \mathcal{P}(\theta(t))^T b^{h,s}(t).$$

Here, the projected stiffness part is given by

$$\mathcal{P}(\theta(t))^T K_{full} \mathcal{P}(\theta(t)) = \begin{pmatrix} P_r^T K_{rr}^h P_r & P_r^T K_{r\Gamma^r}^h R(\theta(t)) & 0 & P_r^T Q_r^h & 0 \\ R(\theta(t))^T K_{\Gamma^r r}^h P_r & R(\theta(t))^T K_{\Gamma^r\Gamma^r}^h R(\theta(t)) + K_{\Gamma^s\Gamma^s}^h & K_{\Gamma^s s}^h P_s & 0 & 0 \\ 0 & P_s^T K_{s\Gamma^s}^h & P_s^T K_{ss}^h P_s & 0 & X_s^h P_s \\ 0 & 0 & 0 & G & 0 \\ 0 & 0 & 0 & 0 & R_{st} \end{pmatrix} \tag{4.21}$$

The application of $\mathcal{P}(\theta)^T$ from the left projects the test functions from $\mathcal{V}^h$ into $\mathcal{V}_\theta^h$ and the application of $\mathcal{P}(\theta)$ from the right projects the solution into the coupled space. Note that by the discrete second coupling condition we have

$$\begin{aligned} & B_{\Gamma^s\Gamma^s}^h u_{\Gamma^s}^{h,s}(t) = R(\theta(t))^T B_{\Gamma^r\Gamma^r}^h u_{\Gamma^r}^{h,s}(t) \\ \Leftrightarrow & - R(\theta(t))^T B_{\Gamma^r\Gamma^r}^h R(\theta(t)) u_{\Gamma^s}^{h,s}(t) + B_{\Gamma^s\Gamma^s}^h u_{\Gamma^s}^{h,s}(t) = 0 \end{aligned}$$

and the Neumann terms vanish.

As we apply the robust optimization framework described in chapter 6 to the induction machine, we will see, that we need derivatives of the state equation, the objective function and the constraint function up to third order with respect to the state variables. Since $\theta$ is a state variable, we need derivatives of $R(\theta)$ up to third order. As the matrix $R(\theta)$ is not differentiable with respect to $\theta$ using linear interpolation, we use a smoothed version of $R(\theta)$ given by

$$\tilde{R}(\theta) := \begin{pmatrix} I & 0 \end{pmatrix} \tilde{R}_{full}(\theta) \begin{pmatrix} I & -I & I & -I \end{pmatrix}^T$$

with

$$\tilde{R}_{full}(n, \epsilon) := \begin{pmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ 1 & & & 0 \end{pmatrix}^n \begin{pmatrix} f(\epsilon) & & f(1-\epsilon) & \\ & \ddots & & \ddots \\ & & \ddots & f(1-\epsilon) \\ f(1-\epsilon) & & & f(\epsilon) \end{pmatrix} \in \mathbb{R}^{360 \times 360}. \tag{4.22}$$

Here $f$ (see Figure 4.2) is a smooth approximation of the hat function on $[-1, 1]$ with the properties

- $f(x) + f(1-x) = 1, \ x \in [0, 1]$,
- $f(1/2) = 1/2$,
- $f(x) > 0, \ x \in [0, 1)$,
- $\lim_{x \to 0} f(x) = \lim_{x \to 1} f(x) = 0$,

i.e. we have a partition of unity and a convex combination in the interpolation.

$\tilde{R}_{full}(\theta)$ is multiplied by an analytic skew factor to account for the skewing of the rotor bars [31].

As we have seen in (3.6.1), the torque can be computed via

$$M_{em}(u, \theta) = l_z \int_\Gamma n \cdot \nu_2 \nabla u_2 n^\perp \cdot \nabla u_1(r_{-\theta} x) \, \mathrm{d}S \tag{4.23}$$

An approach widely used recently to couple rotor and stator is mortaring. In the mortaring approach, the quantity $n \cdot \nu \nabla u$ in (4.23) is introduced as an additional variable in the coupling, which simplifies the torque computation. In the next subsection we will compare mortaring and interpolation and describe how we compute the torque with interpolation coupling.

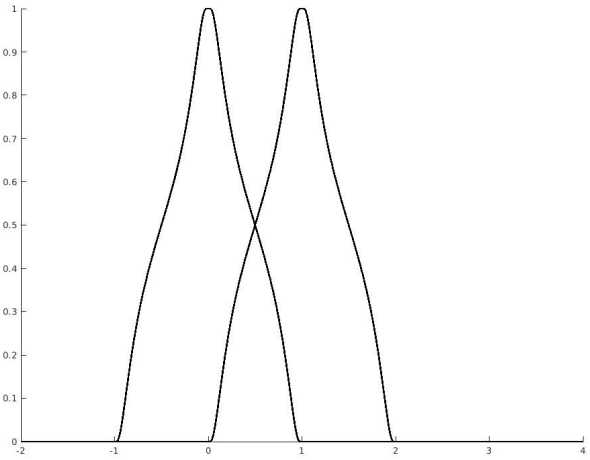


**Figure 4.2.** Smooth coupling functions.

### 4.2.1 Mortaring

Mortaring is a technique to couple decomposed domains with differing meshes possibly leading to non-matching grids or different discretization schemes, see [6, 69, 83, 3, 11, 17, 19]. The idea of mortaring is to relax the strong pointwise continuity in matching conditions by imposing coupling in a weak sense, meaning, enforcing orthogonality between the jump on the interface and a chosen trace space, which can be seen as a Lagrange multiplier. There are two different approaches for mortaring. We will shortly state two equivalent problems, to keep it simple, we drop the time dependence of the problem and introduce the elliptic problem with the same region $\Omega$ as in the previous section

$$\begin{aligned} -\nabla \cdot (\nu_i \nabla u_i) &= r_i, && \text{on } \Omega_i, i = 1, 2, \\ u_1(r_{-\theta} x) &= u_2, && \text{for } x \in \Gamma, \\ \nu_1 \partial_n u_1(r_{-\theta} x) &= \nu_2 \partial_n u_2, && \text{for } x \in \Gamma, \\ u_i &= 0, && \text{for } x \in \partial\Omega_i \backslash \Gamma, \\ u|_{AB} &= u|_{CD}. \end{aligned} \tag{4.24}$$

We define

$$\lambda := \nu \partial_n u_2,$$

and require the first coupling condition to be fulfilled in a weak sense

$$\begin{aligned} &b : H^1(\Omega_1 \cup \Omega_2) \times H^{1/2}(\Gamma) \to \mathbb{R}, \\ &b(u,\mu) := \int_\Gamma \mu(u_1(r_{-\theta}x) - u_2)\,\mathrm{d}S \quad = 0, \quad \text{for } \mu \in H^{1/2}(\Gamma). \end{aligned}$$

In the first approach in mortaring, problem (4.24) is transformed into its weak form

Find $(u_1, u_2, \lambda) \in H^1_{\partial\Omega_1,0}(\Omega_1) \times H^1_{\partial\Omega_2,0}(\Omega_2) \times H^{1/2}(\Gamma)$, such that

$$\begin{aligned} \sum_{i=1}^{2} \int_{\Omega_i} \nu_i \nabla u_i \cdot \nabla v_i \,\mathrm{d}x + b(v,\lambda) &= \sum_{i=1}^{2} \int_{\Omega_i} r_i v_i \,\mathrm{d}x, \qquad \forall v_i \in H^1_{\partial\Omega_i,0}(\Omega_i), \\ b(u,\mu) &= 0, \qquad\qquad \forall \mu \in H^{1/2}(\Gamma). \end{aligned} \tag{4.25}$$

Where $H^1_{\partial\Omega_i,0}$ is the space of $H^1(\Omega_i)$ functions having zero trace on $\partial\Omega_i \backslash \Gamma$. This approach is called the primal-dual approach, since the Lagrange multiplier $\lambda$ is involved. A necessary condition for the existence of a solution is the inf-sup-stability of the bilinearform $b$, i.e. the existence of a positive $\beta$, such that

$$\inf_{\lambda \in H^{1/2}(\Gamma)} \sup_{v \in V} \frac{b(v,\lambda)}{\sqrt{\|v_1\|^2_{H^1(\Omega_1)} + \|v_2\|^2_{H^1(\Omega_2)}}\|\lambda\|_{H^{1/2}(\Gamma)}} \geq \beta > 0,$$

where $V := H^1_{\partial\Omega_1,0}(\Omega_1) \times H^1_{\partial\Omega_2,0}(\Omega_2)$.

In the second approach, the coupling conditions are inserted into the space. Introducing the space

$$\tilde{V} := \{(v_1, v_2) \in H^1_{\partial\Omega_1,0}(\Omega_1) \times H^1(\Omega_2)_{\partial\Omega_2,0} : b(v,\mu) = 0, \forall \mu \in H^{1/2}(\Gamma)\}.$$

We then want to find $u \in \tilde{V}$ fulfilling

$$\sum_{i=1}^{2} \int_{\Omega_i} \nu_i \nabla u_i \cdot \nabla v_i \,\mathrm{d}x = \sum_{i=1}^{2} \int_{\Omega_i} r_i v_i \,\mathrm{d}x, \quad \forall v \in \tilde{V}.$$

This approach is called the primal approach. These two formulations are equivalent [83].

We will now look at the discretization of (4.25). Let $\Lambda^h$ be an $n_m$-dimensional subspace of $H^{1/2}(\Gamma)$ and a basis for $\Lambda^h$ be given by $\{w_i\}_{i=1}^{n_m}$, we will associate the dual basis to the rotor movement, such that we have $\lambda^h(\alpha) = \sum_{i=1}^{n_m} \tilde{w}_i w_i(r\cos(\alpha - \theta, r\sin(\alpha - \theta))$, since $\lambda$ is a quantity on stator side, here $\tilde{w}_i \in \mathbb{R}$ and $\theta$ is the rotor position. For the magnetic vector potential we use the discretization from the

previous section (without time dependence). To evaluate the bilinear form $b$, we compute two integrals. The first one over the interface and basis functions of the stator, tested with the basis of $\Lambda^h$

$$((R^{m,s}(\alpha)))_{ij} = r\int_0^{2\pi} N_j(r\cos(\theta), r\sin(\theta)) w_i(r\cos(\theta-\alpha), r\sin(\theta-\alpha))\mathrm{d}\theta,$$

where $i = 1, \dots, n_m$ and $j \in \mathcal{N}_{\Gamma^s}$ and the second integral over the interface of the rotor

$$((R^{m,r}))_{ij} = r\int_0^{2\pi} N_j(r\cos(\theta), r\sin(\theta)) w_i(r\cos(\theta), r\sin(\theta))\mathrm{d}\theta, \tag{4.26}$$

where again $i = 1, \dots, n_m$ and $j \in \mathcal{N}_{\Gamma^r}$. We neglect here, that the discretized interface is no circle but a polygonal chain. In the discrete setting, the coupling condition (4.2.1) has to hold for all the basis function of the dual space

$$b(w_i, (u^h_{r,f}, u^h_{s,f})) = 0,\ \forall w_i \in \Lambda^h \ \Leftrightarrow\ R^{m,r}u^h_{\Gamma^r} = R^{m,s}(\theta)u^h_{\Gamma^s}.$$

Testing the right hand sides of (4.25)

$$((j_r))_i = \int_{\Omega^h_r} r_1 N^i \,\mathrm{d}x, i \in \mathcal{N}_r,$$
$$((j_s))_i = \int_{\Omega^h_s} r_2 N^i \,\mathrm{d}x, i \in \mathcal{N}_s.$$

leads to the discrete saddle point problem

$$\begin{pmatrix} P_r^T K^h_{rr} P_r & P_r^T K^h_{r\Gamma^r} & 0 & 0 & 0 \\ K^h_{\Gamma^r r} P_r & K^h_{\Gamma^r\Gamma^r} & 0 & 0 & R^{m,r\,T} \\ 0 & 0 & P_s^T K^h_{ss} P_s & P_s^T K^h_{s\Gamma^s} & 0 \\ 0 & 0 & K^h_{\Gamma^s s} & K^h_{\Gamma^s\Gamma^s} & -R^{m,s}(\theta)^T \\ 0 & R^{m,r} & 0 & -R^{m,s}(\theta) & 0 \end{pmatrix} \begin{pmatrix} u^h_{r,dof} \\ u^h_{\Gamma^r} \\ u^h_{s,dof} \\ u^h_{\Gamma^s} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} j_r \\ 0 \\ j_s \\ 0 \\ 0 \end{pmatrix}.$$

As we have chosen $\lambda = \nu\partial_n u_2$ we can write (4.23) as

$$M_{em}(\lambda, u, \theta) = l_z \int_\Gamma \lambda n^\perp \cdot \nabla u_1(r_{-\theta}x)\,\mathrm{d}S \tag{4.27}$$

and with our chosen discretization

$$M^h_{em}(\lambda^h, u^h, \theta) = l_z \int_\Gamma \sum_{i=1}^{n_c} \tilde{w}_i w_i \sum_{i=1}^{n} u^h_i n^\perp \cdot \nabla N^i \,\mathrm{d}S = \ell_z \tilde{w}^T S u^h_{\Gamma^r} \tag{4.28}$$

with

$$((S))_{ij} := r \int_0^{2\pi} w_i n^{\perp} \cdot \nabla N^i \mathrm{d}\theta.$$

In the next section we examine the connection between a special choice of the Lagrange multiplier space and interpolation as a coupling technique which we will use to compute the torque of the machine.

### Mortaring with bi-orthogonal Lagrange Multipliers

For the choice of the test functions, i.e. the choice of the space of the Lagrange multiplier, there are many possibilities, for example one can simply take hat functions which might already be used for the spatial discretization. Recently, the so-called harmonic coupling has received a lot of attention, where the basis functions of the test space consist of sine and cosine with different frequencies [33, 39, 40, 14]. As the Lagrange multiplier is the tangential component of the magnetic field strength, it can be approximated well with the use of oscillating basis functions and rotations can efficiently be treated in the coupling matrix with the use of trigonometric summation formulas, see [39].

We will describe here the use of spaces which basis functions are dual to the piecewise linear hat functions on the interface, where the dual functions are assigned to the mesh on the rotor interface [84]. We introduce the set of test functions

$$\Lambda^h = \{w^h : \Gamma \mapsto \mathbb{R} \mid w^h(x) = \sum_{i=1}^{360} c_i w_i^h(\alpha), c_i \in \mathbb{R}\}.$$

Where the basis functions are given by (see Figure 4.3)

$$w_1^h(\alpha) = \begin{cases} -1/2, & \text{for } \alpha \in [-\frac{2\pi}{360}, -\frac{2\pi}{2\cdot 360}) \cup (\frac{2\pi}{2\cdot 360}, \frac{2\pi}{360}], \\ 3/2, & \text{for } \alpha \in [-\frac{2\pi}{2\cdot 360}, \frac{2\pi}{2\cdot 360}], \\ 0, & \text{else.} \end{cases}$$

$w_2^h, \ldots, w_{360}^h$ are given by rotating $w_1^h$ around $i$ degrees, $i = 1, \ldots, 359$. Dual means here, that the integral over the airgap of the product between the spatial basis function $N_i$ for the magnetic vector potential and the basis functions of the dual space vanishes for differing indices, i.e. (4.26) leads to

$$R_{ij}^{m,r} = \delta_{ij},$$

such that the coupling matrix is easy to compute. To fulfill this property, the basis functions $w_i$ are scaled by $h := \frac{2\pi}{360} \cdot r$, where $r$ is the radius of the airgap. We look

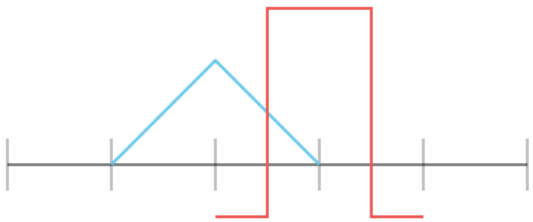

**Figure 4.3.** Hat function in blue and dual basis function in red.

again at the discretized saddle point problem

$$\begin{pmatrix} P_r^T K_{rr}^h P_r & P_r^T K_{r\Gamma^r}^h & 0 & 0 & 0 \\ K_{\Gamma^r r}^h P_r & K_{\Gamma^r\Gamma^r}^h & 0 & 0 & R^{m,r\,T} \\ 0 & 0 & P_s^T K_{ss}^h P_s & P_s^T K_{s\Gamma^s}^h & 0 \\ 0 & 0 & K_{\Gamma^s s}^h & K_{\Gamma^s\Gamma^s}^h & -R^{m,s}(\theta)^T \\ 0 & R^{m,r} & 0 & -R^{m,s}(\theta) & 0 \end{pmatrix} \begin{pmatrix} u_{r,dof}^h \\ u_{\Gamma^r}^h \\ u_{s,dof}^h \\ u_{\Gamma^s}^h \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} j_r \\ 0 \\ j_s \\ 0 \\ 0 \end{pmatrix}.$$

We can use the last row to eliminate $u_{\Gamma^r}^h = R^{m,r\,-1} R^{m,s}(\theta) u_{\Gamma^s}^h$ and obtain

$$\begin{pmatrix} P_r^T K_{rr}^h P_r & 0 & P_r^T K_{r\Gamma^r}^h R^{m,r-1} R^{m,s}(\theta) & 0 \\ K_{\Gamma^r r}^h P_r & 0 & K_{\Gamma^r\Gamma^r}^h R^{m,r-1} R^{m,s}(\theta) & R^{m,r\,T} \\ 0 & P_s^T K_{ss}^h P_s & P_s^T K_{s\Gamma^s}^h & 0 \\ 0 & K_{\Gamma^s s}^h P_s & K_{\Gamma^s\Gamma^s}^h & -\tilde{R(\alpha)}^T \end{pmatrix} \begin{pmatrix} u_{r,dof}^h \\ u_{s,dof}^h \\ u_{\Gamma^s}^h \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} j_r \\ 0 \\ j_s \\ 0 \\ 0 \end{pmatrix}.$$

Solving the second equation for

$$\tilde{w} = -R^{m,r-T}(K_{\Gamma^r r}^h P_r u_{r,dof}^h + K_{\Gamma^r\Gamma^r}^h R^{m,r-1} R^{m,s}(\theta) u_{\Gamma^s}^h) \tag{4.29}$$

and plugging it into the fourth equation leads to

$$\begin{pmatrix} P_r^T K_{rr}^h P_r & 0 & P_r^T K_{r\Gamma^r}^h R^{m,r-T} R^{m,s}(\theta) \\ 0 & P_s^T K_{ss}^h P_s & P_s^T K_{s\Gamma^s}^h \\ R^{m,s}(\theta)^T R^{m,r-T} K_{\Gamma^r r}^h P_r & K_{\Gamma^s s}^h P_s & K_{\Gamma^s\Gamma^s}^h + R^{m,s}(\theta)^T R^{-T} K_{\Gamma^r\Gamma^r}^h R^{m,r-1} R^{m,s}(\theta)) \end{pmatrix}. \tag{4.30}$$

By comparing (4.30) and (4.21), we see, whenever $R^{m,r-T}R^{m,s}(\theta)$ coincides with the interpolation matrix $\tilde{R}(\theta)$, mortaring and interpolation coincide, this is for example the case, when the grid matches and both methods are exact. With the bi-orthogonal multipliers, we always have $R^{m,r} = I$. Especially we can use $u^h_{\Gamma^s}$ and $u^h_{r,dof}$ which were computed with coupling by interpolation and (4.29) to reconstruct the coefficients for the dual basis functions $w_i$ to compute the torque via the matrix

$$((S))_{ij} = \int_0^{2\pi} \frac{w_i}{h} n^\perp \cdot \nabla N^i \mathrm{d}\theta = \begin{cases} -\frac{1}{2h}, & \text{if } i = j-1, \\ 0, & \text{if } i = j, \\ \frac{1}{2h}, & \text{if } i = j+1. \end{cases}$$

Leading to

$$\begin{aligned} M^h_{em}(u^h, \theta) &:= \ell_z r(w^h)^T S \tilde{R}(\theta) u^h_{\Gamma^s} \\ &= -\ell_z r(u^h_{r,dof})^T P_r^T K^T_{\Gamma^r r} S \tilde{R}(\theta) u^h_{\Gamma^s} - \ell_z r(u^h_{\Gamma^s})^T \tilde{R}(\theta)^T K^T_{\Gamma^r \Gamma^r} S \tilde{R}(\theta) u^h_{\Gamma^s}. \end{aligned} \tag{4.31}$$

Note that when the moment of inertia and the friction coefficient in the equation of motion (see 3.6) is given for the full rotor, we need the torque for the full rotor, i.e. we have to compute $S$ for the full airgap and extend $u^h_{\Gamma^s}$ to the full interface respecting the anti-periodicity of the quarters of the machine.

We do not use the dual mortar coupling in our optimization since the hat functions on the interface are $C^0$ and the dual test functions on the interface are not even continuous. The integrals in the entries of $R^{m,s}(\theta)$ depend on the angle, which has a smoothing effect with respect to the angle since this is a convolution. But as the involved functions are to rough, $R^{m,s}(\theta)$ is only a $C^1$ function of the angle. In the robust optimization with a quadratic Taylor model we need $C^2$ to compute the quadratic model which then needs to be differentiated to obtain a descent direction. We therefore use the smooth interpolation presented in the previous section and compute the torque via (4.31).

The whole system in semi-discrete form is given by

$$\begin{aligned}
\mathcal{P}(\theta)^T M_{full} \mathcal{P}(\theta) \partial_t y^{h,s}(t) + \mathcal{P}(\theta)^T K_{full} \mathcal{P}(\theta) y^{h,s}(t) &= \mathcal{P}(\theta)^T b^{h,s}(t) \\
\frac{\mathrm{d}\theta(t)}{\mathrm{d}t} &= \dot{\theta}(t) \\
I \frac{\mathrm{d}\dot{\theta}(t)}{\mathrm{d}t} + \beta \theta(t)^2 &= M^h_{em}(u^{h,s}(t), \theta(t)) \\
i(0) &= 0 \\
\dot{\theta}(0) &= 0 \\
\theta(0) &= 0 \\
u^{h,s}(0) &= 0
\end{aligned} \tag{4.32}$$

In the next section we will describe the setting for the parametrized shape optimization of the induction machine and how we deal with the parameter dependent domain in the discrete setting.

## 4.3 Parametrized models

Since we want to do robust shape optimization, we introduce parameters $p \in P \subset \mathbb{R}^{n_p}$ into the state equation. In our optimization, we will assume that we can control some of the parameters and some of them are uncertain. Our design parameters are the width and height of the rotor bars and in the optimization with the linear material, we assume that the width of the rotor bar slot openings is uncertain as these can get distorted by the huge forces acting on them when the machine is started up. To perform shape optimization, we make our rotor domain parameter dependent $\Omega^h(p) = \Omega^h_r(p) \cup \Omega^h_s$. We will use an affine parametrization of the domain, which has the property, that the Jacobian of the transformation is piecewise constant. This approach follows [72] and has been succesfully used in [51, 56, 15]. Choosing a reference parameter $\hat{p}$ we can define a reference domain $\hat{\Omega}^h := \Omega^h(\hat{p})$. We will see, that this allows us to pre-calculate integrals on the reference domain which are needed in the system matrices. We define $L$ non overlapping rectangles $\hat{\Omega}^i$ in our reference domain $\hat{\Omega}^h$, such that the transformation only takes place in these rectangles and the boundary of the rectangles is not affected by the transformation. The domain not yet covered by rectangles is defined as $\hat{\Omega}^{L+1}$, we then have

$$\operatorname{cl} \hat{\Omega}^h = \bigcup_{i=1}^{L+1} \operatorname{cl} \hat{\Omega}^i, \text{ and } \Omega^i \cap \Omega^j = \emptyset, \text{ for } i \neq j$$

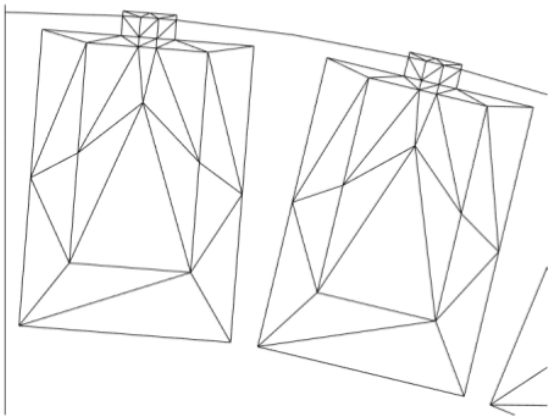

**Figure 4.4.** Triangular division around the rotor bars and the opening of the rotor bar slots.

where $\operatorname{cl}\hat{\Omega}^i$ is the closure of the open set $\hat{\Omega}^i$. We now further subdivide the rectangles into non overlapping triangles, see Figure 4.4,

$$\hat{\Omega}^h = \bigcup_{i=1}^{\tilde{L}} \hat{\Omega}^i \cup \Omega^{\tilde{L}+1}$$

where every triangular subdomain has only one material coefficient. Therefore $\sigma$ and $\nu$ are constant in the subdomains. On every subtriangle $\hat{\Omega}^k$, $k = 1, \ldots, \tilde{L}$, we can define an affine linear mapping

$$\begin{aligned} T_k(\hat{x}, p): \ \hat{\Omega}^k &\rightarrow \Omega^k(p) \\ \hat{x} &\mapsto C_k(p)\hat{x} + d_k(p) =: x \end{aligned}$$

which depends on the parameter $p$ and the matrices $C_k(p) \in \mathbb{R}^{2\times 2}$ are invertible. When triangles share an edge, then their transformations coincide in that shared edge

$$\hat{x} \in \operatorname{cl}\hat{\Omega}^i \cap \operatorname{cl}\hat{\Omega}^j \ \Rightarrow \ T_i(\hat{x}, p) = T_j(\hat{x}, p).$$

As the mappings $T_k$ have four degrees of freedom in the matrix $C_k(p)$ and two degrees of freedom in the vector $d_k(p)$ they are described by the evaluation of their three nodes in the reference domain onto their transformed positions.

Using these transformations, we can rewrite

$$
\begin{aligned}
&\int_{\Omega^h(p)} \nu \nabla u \cdot \nabla v \,\mathrm{d}x - \int_{\hat{\Omega}^{\tilde{L}+1}} \nu \nabla u \cdot \nabla v \,\mathrm{d}x \\
&= \int_{\Omega^h(p)} \nu \left( \frac{\partial u}{\partial x_1}\frac{\partial v}{\partial x_1} + \frac{\partial u}{\partial x_2}\frac{\partial v}{\partial x_2} \right) \mathrm{d}x - \int_{\hat{\Omega}^{\tilde{L}+1}} \nu \nabla u \cdot \nabla v \,\mathrm{d}x \\
&= \sum_{k=1}^{\tilde{L}} \int_{\hat{\Omega}^k} \nu \left( \frac{\partial \hat{u}}{\partial x_1}\frac{\partial \hat{v}}{\partial x_1} + \frac{\partial \hat{u}}{\partial x_2}\frac{\partial \hat{v}}{\partial x_2} \right) |\det C_k(p))| \,\mathrm{d}\hat{x} \\
&= \sum_{k=1}^{\tilde{L}} \int_{\hat{\Omega}^k} \nu \Bigg( \left( \frac{\partial \hat{u}}{\partial \hat{x}_1}\frac{\partial \hat{x}_1}{\partial x_1} + \frac{\partial \hat{u}}{\partial \hat{x}_2}\frac{\partial \hat{x}_2}{\partial x_1} \right) \left( \frac{\partial \hat{v}}{\partial \hat{x}_1}\frac{\partial \hat{x}_1}{\partial x_1} + \frac{\partial \hat{v}}{\partial \hat{x}_2}\frac{\partial \hat{x}_2}{\partial x_1} \right) \\
&\quad + \frac{\partial \hat{u}}{\partial x_2}\frac{\partial \hat{v}}{\partial x_2} \Bigg) |\det C_k(p))| \,\mathrm{d}\hat{x} \\
&= \sum_{k=1}^{\tilde{L}} \int_{\hat{\Omega}^k} \nu \Bigg( \left( \frac{\partial \hat{u}}{\partial \hat{x}_1} C_k(p)^{-1}_{1,1} + \frac{\partial \hat{u}}{\partial \hat{x}_2} C_k(p)^{-1}_{2,1} \right) \\
&\quad \left( \frac{\partial \hat{v}}{\partial \hat{x}_1} C_k(p)^{-1}_{1,1} + \frac{\partial \hat{v}}{\partial \hat{x}_2} C_k(p)^{-1}_{2,1} \right) + \frac{\partial \hat{u}}{\partial x_2}\frac{\partial \hat{v}}{\partial x_2} \Bigg) |\det C_k(p))| \,\mathrm{d}\hat{x} \\
&= \sum_{k=1}^{\tilde{L}} \sum_{i,j}^{2} \left[ C_k(p)^{-1} \nu_k C_k(p)^{-\top}) \right]_{ij} |\det C_k(p)| \int_{\hat{\Omega}^k} \frac{\partial \hat{u}}{\partial \hat{x}_i}\frac{\partial \hat{v}}{\partial \hat{x}_j} \,\mathrm{d}\hat{x} \\
&= \sum_{k=1}^{\tilde{L}} \sum_{i,j}^{2} \Theta_k^{ij}(p) \int_{\hat{\Omega}^k} \frac{\partial \hat{u}}{\partial \hat{x}_i}\frac{\partial \hat{v}}{\partial \hat{x}_j} \,\mathrm{d}\hat{x}.
\end{aligned}
$$

The FE grid is chosen appropriate to the subdomains, which means that the intersection of the boundary of any subdomain $\hat{\Omega}^k$ with the closure of any FE triangle is either empty, a vertex of the triangle or an edge of the triangle. This allows us to assemble the stiffness matrix on the reference domain and to compute it for a specific design by evaluating the sum

$$
K(p) = K^{\tilde{L}+1} + \sum_{k=1}^{\tilde{L}} \sum_{i,j}^{2} \Theta_k^{ij}(p) K^{i,j,k}, \tag{4.33}
$$

where $K^{\tilde{L}+1}$ is the stiffness matrix for the domain $\Omega_r^{\tilde{L}+1}$. In regions that get transformed, we do not only have to compute the integrals $\int_{\hat{\Omega}^h} \partial_{\hat{x}} \hat{u} \partial_{\hat{x}} \hat{v} \,\mathrm{d}\hat{x}$ and $\int_{\hat{\Omega}^h} \partial_{\hat{y}} \hat{u} \partial_{\hat{y}} \hat{v} \,\mathrm{d}\hat{x}$ but also the mixed integrals $\int_{\hat{\Omega}^h} \partial_{\hat{x}} \hat{u} \partial_{\hat{y}} \hat{v} \,\mathrm{d}\hat{x}$ and $\int_{\hat{\Omega}^h} \partial_{\hat{y}} \hat{u} \partial_{\hat{x}} \hat{v} \,\mathrm{d}\hat{x}$. We

can do the same preassembling for the mass matrix

$$\begin{aligned}
&\int_{\Omega^h(p)} \sigma uv \,\mathrm{d}x - \int_{\hat{\Omega}^{\tilde{L}+1}} \sigma uv \,\mathrm{d}x \\
&= \sum_{k=1}^{\tilde{L}} |\det C_k(p)| \sigma_k \int_{\hat{\Omega}^k} \hat{u}\hat{v} \,\mathrm{d}\hat{x} \\
&= \sum_{k=1}^{\tilde{L}} \Theta_k^m(p) \int_{\hat{\Omega}^k} \hat{u}\hat{v} \,\mathrm{d}\hat{x}
\end{aligned}$$

Leading to

$$M_r(p) = M_r^{\tilde{L}+1} + \sum_{k=1}^{\tilde{L}} \Theta_k^m(p) M_r^k.$$

Also the matrix $Q^h$ and $G$ (see (4.16)) can be precomputed on the reference domain. Using these equations, it is easy to compute derivatives of the involved matrices with respect to the design, since we only have to differentiate the scalar coefficient functions $\Theta(p)$.

## 4.4 Nonlinear material

In this section we will describe how the stiffness matrix is assembled when we consider nonlinear material as described in (3.1.1). In the following, let $B = |\vec{B}|$ and $H = |\vec{H}|$. To approximate the B-H-curve, one can use an extended Brauer model [18, 45], which depends on three parameters $k_1, k_2, k_2$ that can be adjusted to model a particular type of steel. The relation between $H$ and $B$ is modelled by

$$H_{br}(B) = (k_1 e^{k_2 B^2} + k_3) B$$

until the slope reaches the reluctivity of vacuum $\nu_0$ at $B_s$, which is the case when the material is saturated. Then ones uses a linear model

$$H_{sat}(B) = \nu_0 (B - B_s) + H_s.$$

The reluctivity is then defined as

$$\nu(x) = \begin{cases} k_1 e^{k_2 x^2} + k_3, & \text{if } x < B_s, \\ \frac{\nu_0 (x - B_s) + H_s}{x}, & \text{if } x \geq B_s. \end{cases}$$

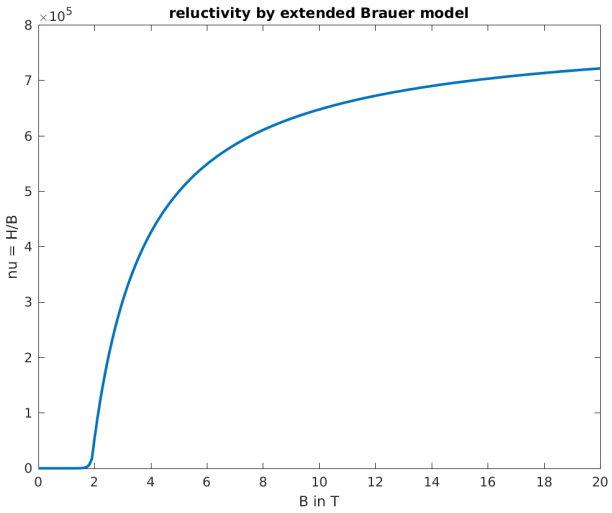


**Figure 4.5.** Extended Brauer curve for $k_1 = 0.3774, k_2 = 2.970, k_3 = 388.33$.

The extended Brauer model has a good approximation of the reluctivity for low and high saturation, the drawback of the extended Brauer curve is, that it is not three times continuously differentiable, which we need in the robust optimization with quadratic Taylor polynomials. We therefore use a smoothened version in the robust optimization.

To calculate $\vec{B}$, we have $\vec{B} = \nabla \times \vec{A}$, $\vec{A} = (0, 0, u)^T$ and therefore $\vec{B} = (\partial_y u, -\partial_x u, 0)$ which leads to

$$|\vec{B}| = |\nabla u|.$$

In every triangle, 3 basis functions $N^i$ are different from zero, so to compute $|\vec{B}|$ in a triangle, let the indices of its three nodes be in the index set $Z$. To account for the parametrized domain, let the triangle be inside the subdomain $\hat{\Omega}_k$ and let the

parameters $p$ be given, then we have in that triangle

$$\begin{aligned}
|\vec{B}(t)| &= |\nabla \sum_{i\in Z} u_i^{h,s}(t) N^i(x)| \\
&= \left| \sum_{i\in Z} u_i^{h,s}(t) \nabla N^i(x) \right| \\
&= \left| \sum_{i\in Z} u_i^{h,s}(t) (\nabla T_k^{-1}(x;p))^T \hat{\nabla} \hat{N}^i(\hat{x}) \right| \\
&= \left| \sum_{i\in Z} u_i^{h,s}(t)) (\nabla T_k^{-1}(x;p))^T \hat{\nabla} \left( \frac{a_i + b_i \hat{x} + c_i \hat{y}}{2\Delta_i} \right) \right| \\
&= \left| \sum_{i\in Z} u_i^{h,s}(t) (C_k^{-T}(p) \left( \frac{1}{2\Delta_i} \begin{pmatrix} b_i \\ c_i \end{pmatrix} \right) \right|,
\end{aligned}$$

where $a_i, b_i, c_i$ are the parameters of the linear basis function $N^i$ on the reference domain in the given triangle and $\Delta_i$ its area. In the following we define the transformation mapping in the parameter independent domain $\Omega^{\tilde{L}+1}$ to be given by $T_{\tilde{L}+1}(\hat{x}) = \hat{x}$. Note that not the whole domain has nonlinear material properties, see Figure 4.6, so $\nu$ is precisely given by $\nu(x, |\nabla|) = \mathbf{1}_{\Omega_{nonlin}}(x) \nu_{nonlin}(x, |\nabla u|) + \mathbf{1}_{\Omega \setminus \Omega_{nonlin}}(x) \nu_{lin}(x)$.

The stiffness matrix is obtained by plugging in

$$\sum_{k=1}^{\tilde{L}+1} \int_{\hat{\Omega}^k} \nu(|C_k^{-T}(p) \hat{\nabla} \hat{u}|^2) \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \cdot \left( C_k^{-T}(p) \hat{\nabla} \hat{v} \right) |\det C_k(p)| \, \mathrm{d}\hat{x}$$

the finite element approximation $\sum_{i=1}^{n_s+n_r} u_i^{h,s}(t) N^i(x)$ for $\hat{u}$ and test with $N^i(x)$, $i = 1, \ldots, n_s + n_r$.

The function $g : \mathbb{R}^2 \to \mathbb{R}^2$ given by

$$g(x) := \left( f(|x|^2) x \right) = \begin{pmatrix} f(x_1^2 + x_2^2) x_1 \\ f(x_1^2 + x_2^2) x_2 \end{pmatrix}$$

has for continuously differentiable $f : \mathbb{R} \to \mathbb{R}$ the Jacobian

$$\begin{aligned}
J_g(x) &= \begin{pmatrix} 2f'(x_1^2 + x_2^2) x_1^2 + f(x_1^2 + x_2^2) & 2f'(x_1^2 + x_2^2) x_1 x_2 \\ 2f'(x_1^2 + x_2^2) x_1 x_2 & 2f'(x_1^2 + x_2^2) x_2^2 + f(x_1^2 + x_2^2) \end{pmatrix} \\
&= f(|x|^2) I + 2f'(|x|^2) x x^T.
\end{aligned} \tag{4.34}$$

We can give a formula for the derivative of the stiffness matrix with respect to the

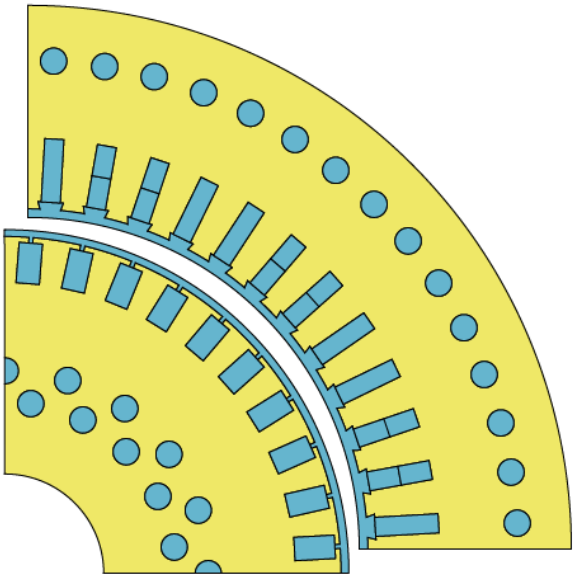

**Figure 4.6.** Region with nonlinear material in yellow.

state $u$ which we need in Newton's method and adjoint and sensitivity computations.

Plugging $C_k^{-1}(p)\hat{\nabla}\hat{u}$ into (4.34) leads us to the derivative of the stiffness matrix in direction $s \in \mathcal{V}^h$

$$\sum_{k=1}^{\tilde{L}+1} \int_{\hat{\Omega}_k} \Bigg( \Big( \nu(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2) I + 2\nu'(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2) \left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right) \left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)^T \Big) \left(C_k^{-T}\hat{\nabla}\hat{s}\right) \Bigg)^T (C_k^{-T}\hat{\nabla}\hat{v}) |\det C_k(p)| \, \mathrm{d}\hat{x}. \tag{4.35}$$

To compute the derivative, we first compute the material coefficients for every triangle and then compute the integrals. In transformable regions $\hat{\Omega}^i$ with nonlinear material properties we have to compute $4 \cdot 4 = 16$ integrals, in regions which have nonlinear material but are not transformable, we have to calculate 4 integrals. For region with linear material which are transformable we have 4 precomputed integrals, which need to be scaled by the coefficient $\Theta(p)$ depending on the design, see (4.33). Regions with linear material that are nontransformable require 2 precom-

puted integrals. For details, see (A.1). In the second derivative with respect to the state, the integrals involving linear material vanish.

## 4.5 Time stepping scheme

To solve the semi-discrete problem (4.32) numerically, we are using a time stepping scheme. To apply the scheme, we divide the interval $[0, T]$ into $k \in \mathbb{N}$ parts $[t_{i-1}, t_i]$, $i = 1, \ldots, k$. We choose $t_0 = 0$ and an equidistant timegrid $t_i = i\Delta t$ with $\Delta t = T/k$ such that we have $t_k = T$. We will use implicit time stepping schemes, since the spatial discretized eddy current equation is a differential algebraic system, see [64]. A time stepping scheme can be written as the system of equations

$$C(\mathfrak{y}, p) = \begin{pmatrix} C^1(\mathfrak{y}_0, \mathfrak{y}_1, p) \\ C^2(\mathfrak{y}_1, \mathfrak{y}_2, p) \\ \vdots \\ C^k(\mathfrak{y}_{k-1}, \mathfrak{y}_k, p) \end{pmatrix} = 0$$

where the blocks $C^i$ represent one timestep and it holds that in timestep $i$ we need to have computed $\mathfrak{y}_{i-1}$ ($\mathfrak{y}_0$ is given as an initial value) and want to solve for $\mathfrak{y}_i$. The blocks depend on the state variables $\mathfrak{y} = \{\mathfrak{y}_1, \ldots, \mathfrak{y}_k\}$ and the control or design $p$. The index $i$ in the state $\mathfrak{y}_i$ means the evaluation at timepoint $t_i$, i.e. $\mathfrak{y}_i = \mathfrak{y}(t_i) \in \mathbb{R}^{n_{dof}+n_\varphi+n_i+3}$ with $\mathfrak{y}_i = (y^{h,i}_{dof}, \mathfrak{T}^i, \theta^i, \dot{\theta}^i)^T = (u^{h,i}_{dof}, \varphi^i, i^i_{st}, \mathfrak{T}^i, \theta^i, \dot{\theta}^i)^T$.

### 4.5.1 Linear material

When using linear material, the magnetoquasistatic equation is linear, but the state system is still nonlinear, since the torque and the interface coupling behave nonlinear. We therefore use the following scheme which does not involve the solution of a nonlinear system. We split the solving of our state system into three parts:

1. Equation of motion
2. Field/circuit equations
3. Torque computation

To solve a block $C^i(\mathfrak{y}_{i-1}, \mathfrak{y}_i, p)$ we start by computing $\theta^i$ and $\dot{\theta}^i$ by using the torque from the previous point in time

$$I\frac{\dot{\theta}^i - \dot{\theta}^{i-1}}{\Delta t} = \mathfrak{T}^{i-1} - \beta(\dot{\theta}^{i-1})^2.$$

Here $I$ is the moment of inertia of the rotor. We update $\theta^i$ by

$$\frac{\theta^i - \theta^{i-1}}{\Delta t} = \dot{\theta}^i.$$

With this computed $\theta^i$ we solve

$$\mathcal{P}(\theta^i)^T(K_{full}(p)+\frac{1}{\Delta t}M_{full}(p))\mathcal{P}(\theta^i)y_{dof}^{h,i} = \frac{1}{\Delta t}\mathcal{P}(\theta^i)^T M_{full}\mathcal{P}(\theta^i)y_{dof}^{h,i-1}+\mathcal{P}(\theta^i)^T b^{h,i}.$$

We assume here, that $\mathcal{P}(\theta^i)^T(K_{full}(p) + \frac{1}{\Delta t}M_{full}(p))\mathcal{P}(\theta^i)$ is regular. Then we use the computed magnetic vector potential $u_{dof}^{h,i}$ to compute the torque via (4.31)

$$\mathfrak{T}^i = M_{em}^h(u_{dof}^{h,i}, \theta^i).$$

Therefore, in every time step we solve the system

$$C^i(\mathfrak{y}_i, \mathfrak{y}_{i-1}, p) = \begin{pmatrix} I\frac{\dot{\theta}^i - \dot{\theta}^{i-1}}{\Delta t} - M_{em}^{i-1} - \beta\dot{(}\theta^{i-1})^2 \\ \frac{\theta^i - \theta^{i-1}}{\Delta t} - \dot{\theta}^i \\ \frac{1}{\Delta t}M_{full}(\cdot,p)(y_{dof}^{h,i} - y_{dof}^{h,i-1}) + K_{full}(\theta^i, p)y_{dof}^{h,i} - \mathcal{P}(\theta^i)^T b^{h,i} \\ \mathfrak{T}^i - M_{em}^h(u_{dof}^{h,i}, \theta^i) \end{pmatrix} = 0,$$

with $K_{full}(\theta, p) := \mathcal{P}(\theta)^T K_{full}(p)\mathcal{P}(\theta)$ and $M_{full}(\cdot, p) := \mathcal{P}(\cdot)^T M_{full}(p)\mathcal{P}(\cdot)$. The $\cdot$ in $\mathcal{P}(\cdot)$ indicates, that the angle has no influence.

### 4.5.2 Nonlinear material

In the simulation with nonlinear material we use the same time stepping scheme as for the linear material, but in step 2 we have to solve the nonlinear system

$$\begin{aligned}\mathcal{P}(\theta^i)^T(K_{full}(u_{dof}^{h,i}, \theta^i, p) + \frac{1}{\Delta}M_{full})\mathcal{P}(\theta)^T y_{dof}^{h,i} \\ = \frac{1}{\Delta}M_{full}(\cdot,p)y_{dof}^{h,i-1} + \mathcal{P}(\theta)^T b^{h,i}.\end{aligned}$$

To compute a root of the following function

$$\tilde{C}^i_{nonlin}(y^{h,i}_{dof}) := \mathcal{P}(\theta^i)^T(K_{full}(u^{h,i}_{dof},\theta^i,p) + \frac{1}{\Delta}M_{full})\mathcal{P}(\theta)^T y^{h,i}_{dof} - \frac{1}{\Delta}M_{full}(\cdot,p)y^{h,i-1}_{dof} - \mathcal{P}(\theta)^T b^{h,i}$$

we use Newton's method, i.e. we solve in loop (for $k = 1,\dots,k^*$)

$$\nabla\tilde{C}^i_{nonlin}(y^{h,i,k}_{dof})d = -\tilde{C}^i_{nonlin}(y^{h,i,k}_{dof})$$

for $d$ and set $y^{h,i,k+1}_{dof} = y^{h,i,k}_{dof} + d$ until $|\tilde{C}^i_{nonlin}(y^{h,i,k}_{dof})|$ is sufficiently small. The formula for the derivative of the stiffness matrix is given in (4.35). As the first iterate for Newton's method, we take the solution of the previous point in time.

## 4.6 Adjoint and Sensitivity

In our optimization, we use the adjoint and sensitivity approach to compute derivatives, see (6.2). In these approaches we need the derivative of the discrete state equation $C(\mathfrak{y},p)$ with respect to $\mathfrak{y}$ and $p$. The derivative with respect to the state variable is given by

$$C_{\mathfrak{y}}(\mathfrak{y},p) = \begin{pmatrix} C^1_{\mathfrak{y}_1}(\mathfrak{y}_0,\mathfrak{y}_1,p) & 0 & 0 & 0 \\ C^2_{\mathfrak{y}_1}(\mathfrak{y}_1,\mathfrak{y}_2,p) & C^2_{\mathfrak{y}_2}(\mathfrak{y}_1,\mathfrak{y}_2,p) & 0 & 0 \\ 0 & \dots & \dots & 0 \\ 0 & 0 & C^k_{\mathfrak{y}_{k-1}}(\mathfrak{y}_{k-1},\mathfrak{y}_k,p) & C^k_{\mathfrak{y}_k}(\mathfrak{y}_{k-1},\mathfrak{y}_k,p) \end{pmatrix}$$

The two types of blocks are given for linear material

$$C^i_{\mathfrak{y}_i}(\mathfrak{y}_i,\mathfrak{y}_{i-1},p) = \begin{pmatrix} 0 & 0 & 0 & I(\Delta t)^{-1} \\ 0 & 0 & (\Delta t)^{-1} & -1 \\ (\Delta t)^{-1}M_{full}(\cdot,p) + K_{full}(\theta^i,p) & 0 & \partial_\theta K_{full}(\theta^i,p)y^{h,i}_{dof} & 0 \\ -\partial_y M_{em}(u^{h,i}_{dof},\theta^i) & 1 & 0 & -\partial_\theta M_{em}(u^{h,i}_{dof},\theta^i) \end{pmatrix} \quad (4.36)$$

and

$$C^i_{\mathfrak{y}_{i-1}}(\mathfrak{y}_i, \mathfrak{y}_{i-1}, u) = \begin{pmatrix} 0 & -1 & 0 & -I(\Delta t)^{-1} - 2\beta\dot{\theta}^{i-1} \\ 0 & 0 & -(\Delta t)^{-1} & 0 \\ -(\Delta t)^{-1} M_{full}(\cdot, p) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

The derivative of the block $K_{full}(\theta, p)$ with respect to $\theta$ has the form

$$\partial_\theta K_{full}(\theta, p) = \begin{pmatrix} 0 & P_r^T K^h_{r\Gamma^r} R'(\theta) & 0 & 0 & 0 \\ R'(\theta)^T K^h_{\Gamma^r r} P_r^T & R'(\theta)^T K^h_{\Gamma^r \Gamma^r} R(\theta) + R(\theta)^T K^h_{\Gamma^r \Gamma^r} R'(\theta) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{4.37}$$

The discrete adjoint equation involves the derivative of the objective or constraint function with respect to the state variable, i.e.

$$C_{\mathfrak{y}}(\mathfrak{y}, p)^T \lambda^h = -f_{\mathfrak{y}}.$$

Exploiting the structure of $C_{\mathfrak{y}}(\mathfrak{y}, p)^T$, we can solve for $\lambda^h$ with the scheme

$$\begin{aligned} (C^k_{\mathfrak{y}_T})^T \lambda^{h,k} &= -f_{\mathfrak{y}_k} \\ (C^{k-1}_{\mathfrak{y}_{k-1}})^T \lambda^{h,k-1} &= -f_{\mathfrak{y}_{k-1}} - (C^T_{\mathfrak{y}_{k-1}})^T \lambda^{h,k} \\ &\vdots \\ (C^1_{\mathfrak{y}_1})^T \lambda^{h,1} &= -f_{\mathfrak{y}_1} - (C^2_{\mathfrak{y}_1})^T \lambda^{h,2}. \end{aligned}$$

In the optimization, we use the width and height of the rotor bars as design parameters. Since the bars are in the interior of the rotor, the boundary of the rotor and the full stator is not affected by varying the design parameters, which leads to the derivative

$$\partial_{p_1}(\mathcal{P}(\theta(t))^T K_{full}(p) \mathcal{P}(\theta(t)) = \begin{pmatrix} \partial_{p_1} P_r^T K^h_{rr}(p) P_r & 0 & 0 & \partial_{p_1} P_r^T Q^h_r(p) & 0 \\ 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \partial_{p_1} G(p) & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{4.38}$$

such that the derivatives with respect to the design do not depend on the rotation

angle. The derivative of a block is given by

$$\partial_{p_1} C^i(\mathfrak{y}_i, \mathfrak{y}_{i-1}, p) = \begin{pmatrix} 0 \\ 0 \\ \frac{1}{\Delta t}\mathcal{P}(\cdot)^T \partial_{p_1} M_{full}(p)\mathcal{P}(\cdot)(y_{dof}^{h,i} - y_{dof}^{h,i-1}) + \mathcal{P}(\cdot)^T \partial_{p_1} K_{full}(p)\mathcal{P}(\cdot) y_{dof}^{h,i} \\ 0 \end{pmatrix}.$$

CHAPTER 5

# Reduced Order Models

In our optimzation procedure, we have to solve our PDE various times. To obtain accurate results, we need a sufficiently fine discretization which leads to a high amount of degrees of freedom and therefore the system of equations we are dealing with has a high dimension. As we have to solve our equation for different designs, the hope is, that by changing the design, the solution trajectory does not change too much and we can use a low dimensional space to compute solutions in. To do this, we apply methods from the field of model order reduction.

We will use the *Proper Orthogonal Decomposition Method* (POD), which has been successfully used in the context of the optimization and simulation of electrical machines for example in [52, 56, 15] for synchronous machines. The method has also been applied in other contexts [13, 44, 55, 62]. Another approach for the reduction of dimension is for example the reduced basis method, see [46, 49].

## 5.1 Proper Orthogonal Decomposition

The idea of the proper orthogonal decomposition method (POD) is to approximate a space spanned by so called snapshots $\{y_i\}_{i=1}^m$ which lie inside a (real) separable Hilbert space $\mathcal{W}$, by an $l$-dimensional space $V^l$, which is spanned by the orthonormal basis vectors $\{\psi_i\}_{i=1}^l$. In the following we will restrict ourselves here to the finite dimensional case where $\mathcal{W} \subset \mathbb{R}^n$, but the method can be used in more generic settings, see for example [74]. The basis is computed by solving the following optimization

problem

$$\begin{aligned} \min_{\{\psi_i\}_{i=1}^l} \quad & \sum_{j=1}^m \alpha_j \|y_j - \sum_{i=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}} \psi_i\|_{\mathcal{W}}^2, \\ \text{s.t.} \quad & \langle \psi_i, \psi_j \rangle_{\mathcal{W}} = \delta_{ij} \quad \text{for } 1 \le i, j \le l, \end{aligned} \tag{5.1}$$

where $\alpha_j$ are nonnegative weights, which can be used to emphasize certain snapshots $y_j$ more or less, which is useful, for example if a time stepping scheme with a non-equidistant timegrid was used to generate the snapshots and

$$\delta_{ij} := \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases}$$

is the Kronecker delta. The norm $\|\cdot\|_{\mathcal{W}} = \langle \cdot, \cdot \rangle_{\mathcal{W}}^{1/2}$ of $\mathcal{W}$ is induced by its inner product $\langle \psi_i, \psi_j \rangle_{\mathcal{W}} = \psi_i^T W \psi_j$ with a given matrix $W \in \mathbb{R}^{n \times n}$, which needs to be symmetric and positive definite. Note that we can find an orthonormal basis $V$ of $\mathbb{R}^n$ with $W = VDV^T$ and $D$ is a $n$-dimensional diagonal matrix. We can then use the matrix $D^{1/2}$ which has the elements $\sqrt{D_{ii}}$ as diagonal elements and define the square root of $W$ as $W^{1/2} = VD^{1/2}V^T$. Note that

$$(W^{1/2})^2 = VD^{1/2}V^T VD^{1/2}V^T = VD^{1/2}D^{1/2}V^T = VDV^T = W$$

and that $W^{1/2}$ is symmetric positive definite aswell. As our time stepping scheme has an equidistant grid, we will set the $\alpha_i$'s to 1 in the following.

Using the bilinearity and the homogenity of the inner product, one can reformulate the objective funtion of (5.1) in the following way:

$$\begin{aligned} \|y_j - \sum_{i=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}} \psi_i\|_{\mathcal{W}}^2 &= \langle y_j - \sum_{i=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}} \psi_i, y_j - \sum_{i=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}} \psi_i \rangle_{\mathcal{W}} \\ &= \|y_j\|_{\mathcal{W}}^2 - 2\langle y_j, \sum_{i=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}} \psi_i \rangle_{\mathcal{W}} + \langle \sum_{i=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}} \psi_i, \sum_{k=1}^l \langle y_j, \psi_k \rangle_{\mathcal{W}} \psi_k \rangle_{\mathcal{W}} \\ &= \|y_j\|_{\mathcal{W}}^2 - 2\sum_{i=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}}^2 + \sum_{i=1}^l \sum_{k=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}} \langle y_j, \psi_k \rangle_{\mathcal{W}} \langle \psi_i, \psi_k \rangle_{\mathcal{W}} \\ &= \|y_j\|_{\mathcal{W}}^2 - \sum_{i=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}}^2 \end{aligned}$$

where we have used, that the $\psi_i$ are $\mathcal{W}$-orthogonal. This shows that instead of minimizing $\|y_j - \sum_{i=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}} \psi_i\|_{\mathcal{W}}^2$ one can maximize $\sum_{i=1}^l \langle y_j, \psi_i \rangle_{\mathcal{W}}^2$. The La-

grangian for the maximization problem

$$\max_{\{\psi_i\}_{i=1}^{l}} \sum_{j=1}^{m}\sum_{i=1}^{l} \langle y_j, \psi_i\rangle_{\mathcal{W}}^2, \quad \text{s.t. } \langle \psi_i, \psi_j\rangle_{\mathcal{W}} = \delta_{ij} \text{ for } 1 \le i,j \le l$$

can be written as

$$\mathcal{L}(\psi_1,\dots,\psi_l,\Lambda) = \sum_{j=1}^{m}\sum_{i=1}^{l} \langle y_j,\psi_i\rangle_{\mathcal{W}}^2 + \sum_{i,j=1}^{l} \lambda_{ij}(\delta_{ij} - \langle \psi_i,\psi_j\rangle_{\mathcal{W}}).$$

The first-order necessary optimality conditions for (5.1) are now given by requiring that the partial derivatives of the Lagrangian $\mathcal{L}$ vanish, which is used to proof the following Theorem.

**Theorem 5.1** ([82], Theorem 1.3.2)**.**
*Let $Y = [y_1,\dots,y_m] \in \mathbb{R}^{n\times m}$ be a given matrix with rank $d \le \min\{m,n\}$, $W$ a symmetric, positive definite matrix, $\bar{Y} = W^{1/2}Y$. Further, let $\bar{Y} = \bar{\Psi}\Sigma\bar{\Phi}^T$ be the singular value decomposition of $\bar{Y}$, where $\bar{\Psi} = [\bar{\psi}_1,\dots,\bar{\psi}_n] \in \mathbb{R}^{n\times n}$, $\bar{\Phi} = [\bar{\phi}_1,\dots,\bar{\phi}_m] \in \mathbb{R}^{m\times m}$ are orthogonal matrices and the matrix $\Sigma \in \mathbb{R}^{n\times m}$ has the form*

$$\bar{\Psi}^T\bar{Y}\bar{\Phi} = \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} = \Sigma \in \mathbb{R}^{n\times n}.$$

*Then for any $l \in \{1,\dots,d\}$ the solution to (5.1) is given by the vectors*

$$\psi_i = W^{-1/2}\bar{\psi}_i, \quad i = 1,\dots,l.$$

*Proof.* See for example [82]. □

When given a set of snaptshots $Y$, we assume that at least one of the snapshots is different from zero, such that the dimension of $\operatorname{span}\{y_i : i = 1,\dots,m\}$ is greater or equal to 1. There are different ways to compute a POD basis. These different approaches are favorable depending on the dimensions $m$ and $n$. We can solve the eigenvalue problem

$$\bar{Y}\bar{Y}^T\bar{\psi}_i = \lambda_i\bar{\psi}_i, \quad 1 \le i \le l$$

and set $\psi_i = W^{-1/2}\bar{\psi}_i$, $i = 1,\dots,l$. In this case, the eigenvalue problem which has to be solved is of dimension $n$. Another way to compute a basis is to compute the solution of

$$\bar{Y}^T\bar{Y}\bar{\phi}_i = \lambda_i\bar{\phi}_i, \quad 1 \le i \le l.$$

We can then use the knowledge of the svd $\bar{Y} = \bar{U}\Sigma\bar{V}^T$ and $\bar{Y}^T\bar{Y} = \bar{V}\Sigma^2\bar{V}^T$ to

compute $\bar{\psi}_i = \bar{Y}\frac{1}{\sqrt{\lambda_i}}\bar{\phi}_i$, $i = 1, \dots, l$, such that the basis vectors are given by

$$\psi_i = W^{-1/2}\bar{\psi}_i = W^{-1/2}\bar{Y}\frac{1}{\sqrt{\lambda_i}}\bar{\phi}_i = Y\frac{1}{\sqrt{\lambda_i}}\bar{\phi}_i,\ i = 1, \dots, l,$$

i.e. $W^{1/2}$ does not need to be computed. The eigenvalue problem which has to be solved is of dimension $m$. The third approach to compute the POD basis uses the singular value decomposition of $\bar{Y}$, i.e.

$$\bar{Y} = \bar{\Psi}\Sigma\bar{\Phi}^T.$$

Again, we set $\psi_i = W^{-1/2}\bar{\psi}_i$, $i = 1, \dots, l$.

In the following we will set $V = \begin{pmatrix}\psi_1 & \dots & \psi_l\end{pmatrix}$. The projection of a vector $y \in \mathbb{R}^n$ onto the POD space is given by

$$y^l = \sum_{i=1}^{l}\langle y, \psi_i\rangle_{\mathcal{W}}\psi_i = \sum_{i=1}^{l} y^T W\psi_i\psi_i = VV^T Wy.$$

An important quantity for the computed reduced basis is the energy

$$\varepsilon(\ell) := \frac{\sum_{i=1}^{\ell}\lambda_i}{\sum_{i=1}^{n}\lambda_i} \in [0, 1]$$

which sets the $l$ largest eigenvalues into relation to all existing ones. A value $\varepsilon(\ell)$ of 1 implies, that we captured all the information that is stored in the span of the snapshots within our reduced basis. Therefore in our numerics we will use lower bounds for $\varepsilon(\ell)$ close to one.

In our numerics, instead of using $W^{1/2}$, we use $L^T$, where $W = LL^T$ and the matrix $\hat{Y} = L^T Y$.

### 5.1.1 POD for induction machines

In this section we describe how we use the proper orthogonal decomposition method introduced in the previous one in the simulation of an asynchronous machine. In the numerical simulation and optimization of an induction machine, the state variable with the largest dimension is the magnetic vector potential, which has the dimension of the finite element state space (4.13). We therefore want to reduce this space. Using POD for the induction machine, we have to make sure, that we are still able to perform the interface coupling for different rotor positions.

### State variable

Since the different parts of the machine have different behavior, for example the stator being static, the rotor rotating, and the rotor bars having parabolic behavior, we will compute individual reduced order models for different subregions. Specifically we subdivide the domain into 13 parts (separated by commata): the stator without the airgap, the rotor without the bars and without the airgap, the bars, the airgap. We apply POD to the magnetic vector potential (MVP) solution on the subdomains individually, i.e. we compute a basis for each of the rotor bars, for rest of the rotor and the stator part. We leave the airgap at full resolution. We do this, since then we are still able to compute the coupling of rotor and stator the same way as we are in the unreduced model. Since the objective function in our optimization (see chapter 7) – the Joule losses in the rotor bars – depend on the magnetic vector potential in the rotor bars and the torque function depends on the magnetic vector potential in the airgap, with this approach we obtain good approximations of our quantities of interest.

We define the following nodal index sets

$$\mathcal{N}_{s,dof}, \mathcal{N}_{r,dof-b}, \mathcal{N}_{b_1}, \dots, \mathcal{N}_{b_{10}}, \mathcal{N}_{\Gamma^s},$$

where the union of the sets is the set of the indices of the FE nodes which are degrees of freedom. $\mathcal{N}_{r,dof-b}$ is the subset of indices of $\mathcal{N}_{r,dof}$ where the indices of the nodes located in the rotor bars are taken out. $\mathcal{N}_{b_i}$ contains the indices of the nodes contained in rotor bar $i$ for $i = 1, \dots, 10$.

Since a magnetic vector potential snapshot restricted to a subdomain $\Omega_{sub} \subseteq \Omega^h$ with $\Omega_{sub} \cap \partial\Omega^h = \emptyset$ lies in a finite dimensional subspace of $H^1(\Omega_{sub})$, we use the weighting matrix $W = \mathcal{P}(0)^T(M_1 + K_1)\mathcal{P}(0) \in \mathbb{R}^{n_{dof} \times n_{dof}}$ by choosing the columns and rows of the index sets $\mathcal{N}_{s,dof}, \mathcal{N}_{r,dof-b}, \mathcal{N}_{b_1}, \dots, \mathcal{N}_{b_{10}}, \mathcal{N}_{\Gamma^s}$. The matrices $M_1$ and $K_1$ are the mass and stiffness matrix computed for $\sigma = \nu = 1$. We do not need to pay attention to the angle $\theta$ in the projection, since the columns and rows which are affected by changing $\theta$ are not used in any weighting, since the airgap is not reduced. Using POD, we obtain 12 $\mathcal{W}$-orthonormal basis

$$\begin{aligned}
\Psi_s &:= \{\psi_1^s, \dots, \psi_{l_s}^s\}, \\
\Psi_{r-b} &:= \{\psi_1^{r-b}, \dots, \psi_{l_{r-b}}^{r-b}\}, \\
\Psi_{b_1} &:= \{\psi_1^{b_1}, \dots, \psi_{l_{b_1}}^{b_1}\}, \\
&\vdots \\
\Psi_{b_{10}} &:= \{\psi_1^{b_{10}}, \dots, \psi_{l_{b_{10}}}^{b_{10}}\},
\end{aligned}$$

with their individual dimensions $l_s$, $l_r$, $l_{b_1}$, …, $l_{b_{10}}$. The dimension of the combined ROM is

$$l = l_s + l_r + l_{b_1} + \cdots + l_{b_{10}}.$$

Setting

$$V_s := \begin{pmatrix} \psi_1^s & \cdots & \psi_{l_s}^s \end{pmatrix} \in \mathbb{R}^{|\mathcal{N}_{s,dof}| \times l_s},$$

and equivalently for the other basis, we obtain

$$V_s,\, V_{r-b},\, V_{b_1},\, \ldots,\, V_{b_{10}}.$$

We will combine $V_{r-b}$ and $V_{b_1}$, …, $V_{b_{10}}$ to $V_{r+b} \in \mathbb{R}^{n_{r,dof} \times l_{r-b} + l_{b_1} + \ldots + l_{b_{10}}}$ by

$$V_{r+b}(\mathcal{N}_{r-b}(i), j) = V_{r-b}(i,j),\ i = 1, \ldots, |\mathcal{N}_{r-b}|,\ j = 1, \ldots, l_{r-b}$$

and

$$\begin{aligned} V_{r+b}(\mathcal{N}_{b_1}(i), j + l_{r-b}) &= V_{b_1}(i,j),\ i = 1, \ldots, |\mathcal{N}_{b_1}|,\ j = 1, \ldots, l_{b_1} \\ V_{r+b}(\mathcal{N}_{b_2}(i), j + l_{r-b} + l_{b_1}) &= V_{b_2}(i,j),\ i = 1, \ldots, |\mathcal{N}_{b_2}|,\ j = 1, \ldots, l_{b_2} \\ &\vdots \\ V_{r+b}(\mathcal{N}_{b_{10}}(i), j + l_{r-b} + l_{b_1} + \ldots + l_{b_9}) &= V_{b_{10}}(i,j),\ i = 1, \ldots, |\mathcal{N}_{b_{10}}|,\ j = 1, \ldots, l_{b_{10}}. \end{aligned}$$

and then collect the $\mathcal{N}_{r,dof}$ (nonzero) rows $V_{r-b}$, since the sets $\mathcal{N}_{r-b}, \mathcal{N}_{b_1}, \ldots, \mathcal{N}_{b_{10}}$ contain nodal indices and not dof indices we first obtain a matrix on rotor nodal dimension of $n_r$.

When we have computed the reduced basis for the individual parts, we can use an Galerkin ansatz and project our state system onto this low dimensional space via the matrix

$$V = \begin{pmatrix} V_{r+b} & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ 0 & 0 & V_s & 0 & 0 \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{pmatrix} \in \mathbb{R}^{n_{dof} + n_\varphi + n_i \times l + n_\Gamma^s + n_\varphi + n_i}$$

Again we define a projector for the MVP

$$V_u = \begin{pmatrix} V_{r+b} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & V_s \end{pmatrix} \in \mathbb{R}^{n_{dof} \times l + n_\Gamma^s}.$$

Setting

$$\mathcal{P}^V(\theta) := \mathcal{P}(\theta)V \quad \text{and} \quad \mathcal{P}_u^V(\theta) := \mathcal{P}_u(\theta)V_u$$

we can define the POD space as

$$\begin{aligned}\mathcal{V}_\theta^l := \Bigg\{ &\left( \sum_{i=1}^{n_r} u_i^h N^i(x), \sum_{i=n_r+1}^{n_s+n_r} u_i^h N^i(x) \right) \in H^1(\Omega_r^h) \times H^1(\Omega_s^h) \\ &: u^h = \mathcal{P}_u^V(\theta) u^l,\, u^l \in \mathbb{R}^{l+n_\Gamma^s} \Bigg\}.\end{aligned} \tag{5.2}$$

We denote the POD approximation of the magnetic vector potential evaluated at timepoint $t_i$ $u^{l,i} \in \mathbb{R}^{l+n_\Gamma^s}$. The field/circuit equations for timestep $i$ on ROM level are given by

$$\frac{1}{\Delta t}\mathcal{P}^V(\theta^i)^T M_{full}\mathcal{P}^V(\theta^i)(y^{l,i} - y^{l,i-1}) + \mathcal{P}^V(\theta^i)^T K_{full}\mathcal{P}^V(\theta^i)y^{l,i} - \mathcal{P}^V(\theta^i)^T b^{h,i},$$

where $y^{l,i} = (u^{l,i}, \varphi^i, i_{st}^i)^T$. The projected $K_{full}$ matrix is given by

$$\mathcal{P}^V(\theta)^T K_{full}\mathcal{P}^V(\theta) =$$

$$\begin{pmatrix} V_{r+b}^T P_r^T K_{rr}^h P_r V_{r+b} & V_{r+b}^T P_r^T K_{r\Gamma r}^h R(\theta) & 0 & V_{r+b}^T P_r^T Q_r^h & 0 \\ R(\theta)^T K_{\Gamma r r}^h P_r V_{r+b} & R(\theta)^T K_{\Gamma^r\Gamma^r}^h R(\theta) + K_{\Gamma^s\Gamma^s}^h & K_{\Gamma^s s}^h P_s & 0 & 0 \\ 0 & V_s^T P_s^T K_{s\Gamma^s}^h & V_s^T P_s^T K_{ss}^h P_s V_s & 0 & X_s^h P_s V_s \\ 0 & 0 & 0 & G & 0 \\ 0 & 0 & & 0 & R_{st} \end{pmatrix} \tag{5.3}$$

We see that only 3 entries depend on the angle $\theta$, where two are the same when transposed such that we can precompute most of the matrices.

The torque in ROM is computed by

$$M_{em}^l(u^l, \theta) = -\ell_z r(u_{b+r}^l)^T V_{r+b}^T P_r^T K_{\Gamma^r r}^T S\tilde{R}(\theta)u_{\Gamma^s}^h - \ell_z r(u_{\Gamma^s}^h)^T \tilde{R}(\theta)^T K_{\Gamma^r\Gamma^r}^T S\tilde{R}(\theta)u_{\Gamma^s}^h$$

and the equation of motion is not affected by the reduction of the magnetic vector potential.

To solve the state equation on the ROM level, the initial values need to be projected onto the ROM space

$$u_0^l = V^T W\mathcal{P}(0)u_0^h.$$

If we want to run an optimization and not just solve the state equation, we also need to consider, that the derivatives need to be approixmated well in the reduced space. Therefore in the optimization, we also compute POD approximation spaces for the adjoint and sensitivity.

### Adjoint and Sensitivities

Since we are using the adjoint state and sensitivties in our robust optimization (see 6.2) we reduce the adjoint and sensitivity space aswell. We make the same distribution of the region as for the state variable and we use the same procedure as for the state variable to compute bases for the adjoint state and the sensitivity, such that we obtain the matrices

$$V_s^\lambda,\, V_{r-b}^\lambda,\, V_{b_1}^\lambda,\, \ldots,\, V_{b_{10}}^\lambda,$$

which contain the basis vectors for the reduced adjoint space and have column dimension $l_s^\lambda,\, l_{r-b}^\lambda,\, l_{b_1}^\lambda,\, \ldots,\, l_{b_{10}}^\lambda$ and

$$V_s^{y_p},\, V_{r-b}^{y_p},\, V_{b_1}^{y_p},\, \ldots,\, V_{b_{10}}^{y_p},$$

which contain the basis vectors for the sensitivities and have column dimension $l_s^{y_p},\, l_{r-b}^{y_p},\, l_{b_1}^{y_p},\, \ldots,\, l_{b_{10}}^{y_p}$. We then use an orthogonalization procedure on the columns of

$$\begin{aligned}
&\left(V_s, V_s^\lambda, V_s^{y_p}\right),\\
&\left(V_{r-b}, V_{r-b}^\lambda, V_{r-b}^{y_p}\right),\\
&\left(V_{b_1}, V_{b_1}^\lambda, V_{b_1}^{y_p}\right),\\
&\qquad\vdots\\
&\left(V_{b_{10}}, V_{b_{10}}^\lambda, V_{b_{10}}^{y_p}\right),
\end{aligned}$$

to obtain the combined basis (this means linear dependent vectors get removed during orthogonalization)

$$V_s^c, V_{r-b}^c, V_{b_1}^c, \ldots, V_{b_{10}}^c,$$

with dimension $l_s^c \leq l_s + l_s^\lambda + l_s^{y_p}$ (analogous for the other basis), where we have equality if the basis vectors for state, adjoint and sensitivity are linear independent. We then use $V_s^c, V_{r-b}^c, V_{b_1}^c, \ldots, V_{b_{10}}^c$ as a basis for the state variable, adjoint state and the sensitivity. In our numerics we use the orth function of Matlab, which orthogonalizes a set of vectors with respect to the Euclidean inner product, therefore we lose $\mathcal{W}$ orthogonality in the procedure, but the span of the orthogonalized basis stays the same.

## 5.2 Nonlinearities

When we are using nonlinear material, the stiffness matrix depends on the magnetic vector potential. If we use the POD ansatz from the previous section we have the term

$$\mathcal{P}_u^V(\theta)^T K^h(\mathcal{P}_u^V(\theta)u^{l,i})\mathcal{P}_u^V(\theta)u^{l,i}$$

in the state equation. To evaluate the term, we need to project the POD dimensional vector $u^{l,i}$ to finite element dimension, then evaluate the nonlinear material $\nu(|\nabla u_f^h|)$ for every triangle, then solve the integrals to obtain the stiffness matrix in FEM dimension and project the matrix back to the POD dimension. Hence, this is not efficient. To overcome this inefficiency we use the *Discrete Interpolation Method* (DEIM), see [22, 38]. Since the domain of the induction machine has subregions with linear and nonlinear material, we can split the stiffness matrix into a linear (denoted $K_{\text{lin}}$) and a nonlinear (denoted $K_{\text{nonlin}}$) part. As the airgap in the induction machine is air, which does not depend on the state variable and the rotor and stator coupling acts on the nodes in the airgap, only the linear part depends on the angle $\theta$ and we have on the dof level

$$\mathcal{P}_u(\theta)^T K^h(\mathcal{P}_u(\theta)u_{dof}^h)\mathcal{P}_u(\theta) = K_{\text{lin}}(\theta) + K_{\text{nonlin}}(u_{dof}^h), \tag{5.4}$$

where

$$K_{\text{lin}}(\theta) := \mathcal{P}_u(\theta)^T K_{\text{lin}}\mathcal{P}_u(\theta) \text{ and } K_{\text{nonlin}}(u_{dof}^h) := \mathcal{P}_u(\cdot)^T K(\mathcal{P}_u(\cdot)u_{dof}^h)\mathcal{P}_u(\cdot)$$

here $(\cdot)$ denotes that the input has no influence.

We define (note $u^l = V_u^T u_{dof}^h$)

$$F(V_u u^l) := K_{\text{nonlin}}(V_u u^l)V_u u^l,$$

such that

$$F : \mathbb{R}^{n_{dof}} \to \mathbb{R}^{n_{dof}}.$$

The idea behind the method is to approximate the nonlinearity

$$F(x) \approx Wc(x), \tag{5.5}$$

where the matrix $W \in \mathbb{R}^{n_{dof} \times n_{deim}}$ contains linear independent columns, denoted $w_i$ and $c : \mathbb{R}^{n_{dof}} \to \mathbb{R}^{n_{deim}}$. To obtain the matrix $W$ one can use nonlinear snapshots $F_i = F(x_i), i = 1, \ldots, k$ and use the POD method to compute an orthonormal basis which approximates the space spanned by the nonlinear snapshots. The $F_i$'s can

be collected during the solve of the state equation on the FE level. The system (5.5) is overdetermined, since we want to achieve a reduction of dimension with the POD approach, hence $n_{deim} < n_{dof}$. The idea to solve the overdetermined system is to choose $n_{deim}$ rows of $W$, such that $W_{n_{deim},:}$ is invertible. We define a selection operator $P^T \in \mathbb{R}^{n_{deim} \times n_{dof}}$ which selects chosen $n_{deim}$ rows. Then the problem

$$P^T F(x) = P^T W c(x),$$

can be solved for $c(x)$ by

$$c(x) = (P^T W)^{-1} P^T F(x).$$

Inserting the computed $c(x)$ into the ansatz

$$F(x) \approx W(P^T W)^{-1} P^T F(x) =: \tilde{F}(x),$$

we see, that since the selection operator $P^T$ chooses rows, by $P^T F(x)$ we just need to compute $n_{deim}$ rows of $F(x)$. Note that at rows, which are selected by $P^T$, the approximation is exact

$$P^T \tilde{F}(x) = P^T W(P^T W)^{-1} P^T F(x) = P^T F(x),$$

such that the values of non chosen rows are interpolated by the evaluation of the chosen rows. As suggested in [22] the interpolation indices can be chosen by the greedy Algorithm 5.1. As a first step, the algorithm picks the index of the first basis vector with the largest absolute value. Then iteratively the next basis vectors are interpolated with the current basis and the current interpolation set and the index of the largest absolute value of the residual is picked as interpolation index (and the basis vector is added to the current basis).

**Algorithm 5.1** Greedy DEIM selection

**Input:** $\{w_i\}_{i=1}^{n_{deim}} \subset \mathbb{R}^{n_{dof}}$ linearly independent
**Output:** $P^T \in \mathbb{R}^{n_{deim} \times n_{dof}}$

1: $[\sim, p_1] \leftarrow \max\{|w_1|\}$
2: $W \leftarrow [w_1], P \leftarrow [e_{p_1}], p \leftarrow [p_1]$
3: **for** $i = 2$ to $n_{deim}$ **do**
4: Solve $P^T W c = P^T w_i$ for $c \in \mathbb{R}^{i-1}$
5: $r = w_i - Wc$
6: $[\sim, p_i] = \max\{|r|\}$
7: $W \leftarrow [W\, w_i], P \leftarrow [P\, e_{p_i}], p \leftarrow [p\, p_i]$

There are two errors introduced by approximating the nonlinear function with

DEIM. The first one lies in the choice of the basis $W$ and the second one lies in the selection of the interpolation indices. In [22] the error is analyzed when using Algorithm 5.1. The solution of

$$\min_{c(x)\in\mathbb{R}^{n_{deim}}} \|F(x) - Wc(x)\|_2^2$$

is given by the solution of the normal equation

$$W^T W c(x) = W^T F(x),$$

which is equal to

$$c(x) = W^T F(x),$$

if the basis $W$ is orthonormal, i.e. $W^T W = I$, such that $F(x) \approx WW^T F(x)$. Thus

$$\|(I - WW^T)F(x)\|$$

is the error of the best 2-norm approximation. The total error is given by

$$\|F(x) - P^T(P^T W)^{-1}F(x)\| \leq \|(P^T W)^{-1}\| \cdot \|(I - WW^T)F(x)\|.$$

Choosing the interpolation indices with Algorithm 5.1 leads to

$$\|(P^T W)^{-1}\| \leq \frac{(1+\sqrt{2(n_{dof})})^{n_{deim}-1}}{|e_{p_1}^T w_1|} = (1+\sqrt{2(n_{dof})})^{n_{deim}-1}\|w_1\|_\infty^{-1}.$$

To improve the performance of DEIM, we use a reformulation of our nonlinearity

$$K_{\text{nonlin}}(u_{dof}^h)u_{dof}^h = K_{\text{nonlin}}(\bar{u})\bar{u} + dK_{\text{nonlin}}(\bar{u})(u_{dof}^h - \bar{u}) + R(u_{dof}^h),$$

where $dK_{\text{nonlin}}(\bar{u})$ is the derivative of $K_{\text{nonlin}}(\bar{u})\bar{u}$ and $\bar{u} \in \mathbb{R}^{n_{dof}}$ is a prechosen expansion point. In our simulation, we choose $\bar{u}$ as the first unit eigenvector of $FF^T$, scaled by the average length $\sum_{i=1}^k \|F_i\|/k$, where $F$ contains the collected nonlinear snapshots as columns $F_i$. We compute

$$R(u_{dof}^{h,j}) = K_{\text{nonlin}}(u_{dof}^{h,j})u_{dof}^{h,j} - K_{\text{nonlin}}(\bar{u})\bar{u} - dK_{\text{nonlin}}(\bar{u})(u_{dof}^{h,j} - \bar{u}),\ j = 1,\ldots,k$$

and use POD on the set $\{R(u_{dof}^{h,j}) : j = 1,\ldots,k\}$ to construct $W$ with orthonormal columns. We define

$$K_{\bar{u}} := (K_{\text{nonlin}}(\bar{u}) - dK_{\text{nonlin}}(\bar{u}))\bar{u}$$

and

$$\hat{R}(u_{dof}^{h,j}) := W(P^T W)^{-1} P^T R(u_{dof}^{h,j}),$$

with $P^T$ being a row selector. The stiffness term (5.4) can then be approximated on ROM level by

$$V^T K_{\text{lin}}(\theta) V u^l + V^T \hat{R}(\mathcal{P}_u^V(\theta) u^l) + V^T K_{\bar{u}} + V^T dK_{\text{nonlin}}(\bar{u}) \mathcal{P}_u^V(\theta) u^l.$$

and the jacobian

$$V^T K_{\text{lin}}(\theta) V + V^T \hat{R}'(\mathcal{P}_u^V(\theta) u^l) \mathcal{P}_u^V(\theta) + V^T dK_{\text{nonlin}}(\bar{u}) \mathcal{P}_u^V(\theta),$$

where

$$\hat{R}'(u_{dof}^h) := W(P^T W)^{-1} P^T R'(u_{dof}^h) \text{ and}$$
$$R'(u_{dof}^h) = dK_{\text{nonlin}}(u_{dof}^h) - dK_{\text{nonlin}}(\bar{u}).$$

CHAPTER 6

# Robust Optimization

This chapter deals with the topic of robust optimization. In many real applications there are parameters whose exact value is unknown or uncertain. For example, in the production of an electric motor, only certain manufacturing tolerances can be maintained, parts can be deformed under load during use, or materials do not behave the same at different temperatures. There are different techniques to approach uncertainty. One can choose a stochastic approach where one needs to have knowledge about the underlying probability distribution and then optimize for example the expecation value, the standard deviation or some risk measures [65, 54]. Recently an approximation scheme to solve distributionally robust optimization problems for NLPs [59] and PDE constrained optimization [60] was introduced, where in the underlying problem one deals with uncertain probability distributions and wants to minimize the worst-case of the expectation value. The focus in this thesis is on robust optimization where the uncertainties are treated with a worst-case approach, [9, 12, 78]. In this approach, we assume that the uncertainties lie in a bounded set, called the uncertainty set, which we know. We then solve the uncertain optimization problem in such a way that in the optimum we have the best possible (over the optimization variable) worst-case (over the uncertainty). We require that the constraints are satisfied for all parameters in the uncertainty set. The described optimization problem is called the robust counterpart and will be introduced in this chapter. The methods described in the following have been applied to synchronous machines [56, 14, 52] and to the robust optimization of trusses [51, 52].

To describe the robust optimization method we are using, we introduce the fol-

lowing optimization problem

$$\begin{aligned} \min_{y,\mu} \quad & F^0(y,\mu,p) \\ \text{s.t.} \quad & F^i(y,\mu,p) \leq 0, \text{ for } i \in I \\ & e(y,\mu,p) = 0. \end{aligned} \tag{6.1}$$

This optimization problem involves an optimization or design variable $\mu \in U$, a state variable $y \in Y$ and an uncertain parameter $p \in P$, where the parameter space $P \subseteq \mathbb{R}^{n_p}$ is finite-dimensional and $Y$ and $U$ are reflexive Banach spaces. The objective function $F^0 : Y \times U \times P \to \mathbb{R}$ is to be minimized. The mapping $e : Y \times U \times P \to Z$, where $Z$ is a Banach space describes a physical behavior and sets the design $\mu$, the uncertainty $p$ and the state $y$ into relation. $F^i : Y \times U \times P \to \mathbb{R}$ with $i \in I$, where $I \subseteq \{1, 2, \ldots, |I|\}$, denote the inequality constraint functions. We assume that all involved functions $F^i$ for $i \in \{0\} \cup I$ are continuously differentiable.

We assume, that the state equation $e(y,\mu,p) = 0$ has a unique solution $y \in Y$ for every $\mu \in U$ and $p \in P$. This can be guaranteed for example by the implicit function theorem, which would require that $e(y,\mu,p)$ is continuously differentiable, that for every $(\mu,p) \in U \times P$ there is a unique $y \in Y$ with $e(y,\mu,p) = 0$ and, that the Jacobian $e_y(y,\mu,p)$ is invertible in $e(y,\mu,p) = 0$. Then there exists a mapping $y : U \times P \to Y$ with $e(y(\mu,p),\mu,p) = 0$ and we can then define the reduced functions $f^i : U \times P \to \mathbb{R}$ by

$$f^i(\mu,p) = F^i(y(\mu,p),\mu,p).$$

This lets us formally eliminate the state variable $y$ from the optimization problem and we obtain the optimization problem in reduced form.

$$\min_u f^0(\mu,p) \quad \text{s.t.} \quad f^i(\mu,p) \leq 0, \quad i \in I.$$

We further assume, that the uncertain parameter $p$ lies in a set $\mathcal{U} \subseteq P$ which is a comptact set that can be described by

$$\mathcal{U} = \{p \in \mathbb{R}^{n_p} : \|p - \bar{p}\|_B \leq 1\}, \tag{6.2}$$

where $B$ is a symmetric positive definite matrix with the Cholesky decomposition $B = LL^T$ and

$$\|u\|_B^2 = u^T B u = u^T L L^T u = \|L^T u\|_2^2, \tag{6.3}$$

and $\bar{p} \in \mathbb{R}^{n_p}$ is the so called nominal parameter. Note that the uncertainty set in (6.2) is an ellipsoidal with the center $\bar{p}$. We address the uncertainty by a robust worst-case approach, which means, we need our constraints to be fulfilled for every $p \in \mathcal{U}$ and we want to find the design $\mu$ which minimizes the worst-case (over $\mathcal{U}$)

of the objective function. We define the robust counterpart [9, 10] of the problem (6.1) as

$$\begin{aligned} \min_{\mu} \quad & \max_{p \in \mathcal{U}} \; f^0(\mu, p) \\ \text{s.t.} \quad & f^i(\mu, p) \leq 0, \; \forall p \in \mathcal{U} \text{ for } i \in I \end{aligned} \tag{6.4}$$

which is equivalent to

$$\begin{aligned} \min_{\mu} \quad & \max_{p \in \mathcal{U}} \; f^0(\mu, p) \\ \text{s.t.} \quad & \max_{p \in \mathcal{U}} \; f^i(\mu, p) \leq 0, \; \text{for } i \in I \end{aligned} \tag{6.5}$$

This problem is of bi-level structure and therefore difficult to solve, since the $f^i$ can be nonlinear and nonconvex and are therefore nontractable and it cannot be guaranteed to find the global maximum. A way to takle the bi-level problems is to use approximations of the involved functions $f^i$ for example by Taylor expansion or interpolation models which can be solved to global optimality over the uncertainty set.

## 6.1 Derivative based approximations

A way to approximate the objective and constraint functions is by Taylor polynomials with respect to the uncertain parameter. As the error of the approximation depends on the distance to the expansion point, smaller uncertainty sets lead to better approximations. To compute the model, we need the underlying functions $f^i$ to be sufficiently smooth. For the first order model, the maximum over the uncertain set can be given explicitly and in the case of a quadratic model, the resulting inner maximization problems are the well known trust region subproblem.

### 6.1.1 First Order

The linear Taylor expansion of a functions $f^i$ around the nominal parameter $\bar{p}$ is defined by

$$f_l^i(\mu, p_i) := f^i(\mu, \bar{p}) + \nabla_p f^i(\mu, \bar{p}) \cdot (p_i - \bar{p}). \tag{6.6}$$

Here $\cdot : P \times P$ denotes the usual Euclidean inner product in $\mathbb{R}^{n_p}$. In the linear approximation (6.6), only the second summand depends on the uncertain parameter and has to be maximized over the uncertain set $\mathcal{U}$. Introducing the deviation $\delta$ from

the nominal parameter $\bar{p}$, we can write

$$\mathcal{U} = \{p : \|p - \bar{p}\|_B \leq 1\} = \{\bar{p} + \delta : \|\delta\|_B \leq 1\}$$

which leads to the inner maximization problem in (6.5)

$$\begin{aligned}\max_{\|\delta_i\|_B \leq 1} f_l^i(\mu, \delta_i) &= \max_{\|\delta_i\|_B \leq 1} (f^i(\mu, \bar{p}) + \nabla_p f^i(\mu, \bar{p}) \cdot \delta_i) \\ &= f^i(\mu, \bar{p}) + \max_{\|\delta_i\|_B \leq 1} \nabla_p f^i(\mu, \bar{p}) \cdot \delta_i.\end{aligned}$$

The maximum can be characterized using the dual norm

$$\max_{\|\delta_i\|_B \leq 1} f_l^i(\mu, \delta_i) = f^i(\mu, \bar{p}) + \|\nabla_p f^i(\mu, \bar{p})\|_{B^*}.$$

This follows directly from the definition of the dual norm

$$\|z\|_{B^*} = \max_{\|y\|_B \leq 1} \frac{|z \cdot y|}{\|y\|_B}. \tag{6.7}$$

In our case we have (6.3) and the dual norm is given by

$$\|z\|_{B^*} = \|L^{-1} z\|_2,$$

which can be seen by using choosing $x = L^T y$ and applying the Cauchy Schwarz inequality in (6.7). The linear approximation of the robust counterpart is given by

$$\begin{aligned}\min_{\mu} \quad & f^0(\mu, \bar{p}) + \|\nabla_p f^0(\mu, \bar{p})\|_{B^*} \\ \text{s.t.} \quad & f^i(\mu, \bar{p}) + \|\nabla_p f^i(\mu, \bar{p})\|_{B^*} \leq 0, \text{ for } i \in I.\end{aligned} \tag{6.8}$$

Since the norm is non differentiable in 0, the objective and constraint functions are nonsmooth. There are smooth reformulations involving slack variables, when the norm to describe the uncertainty set is chosen as $\|L^T \cdot\|_1$ or $\|L^T \cdot\|_\infty$, see [37, 78]. Since we are using the Euclidean norm, we will use an optimization solver which can handle non smoothness and uses a technique similiar to gradient sampling, see [29].

### 6.1.2 Second Order

If the $f^i$'s are highly nonlinear, the first order expansion might lead to a bad approximation of the worst-case, which can result in a non robust solution, if the

worst-case is underestimated, or in an infeasible problem, when the worst-case is overestimated, such that the use of higher order models is suggested. We introduce the quadratic approximation

$$f_q^i(\mu, \delta_i) := f^i(\mu, \bar{p}) + \nabla_p f^i(\mu, \bar{p}) \cdot \delta_i + \frac{1}{2}(\nabla_{pp} f^i(\mu, \bar{p})\delta_i) \cdot \delta_i. \tag{6.9}$$

The quadratic approximation of the robust counterpart is given by

$$\begin{aligned} \min_{\mu} \quad & \max_{\delta_0 \in \mathcal{U}} \; f^0(\mu, \bar{p}) + \nabla_p f^0(\mu, \bar{p}) \cdot \delta_0 + \tfrac{1}{2}(\nabla_{pp} f^i(\mu, \bar{p})\delta_0) \cdot \delta_0 \\ \text{s.t.} \quad & \max_{\delta_i \in \mathcal{U}} \; f^i(\mu, \bar{p}) + \nabla_p f^i(\mu, \bar{p}) \cdot \delta_i + \tfrac{1}{2}(\nabla_{pp} f^i(\mu, \bar{p})\delta_i) \cdot \delta_i \leq 0, \text{ for } i \in I \end{aligned} \tag{6.10}$$

The solution to the inner maximization problems in (6.10) cannot be written in a closed form, but denoting

$$\tilde{g}_i := L^{-1} \nabla_p f^i(\mu, \bar{p}) \;\text{ and }\; \tilde{H}_i := L^{-1} \nabla_{pp} f^i(\mu, \bar{p}) L^{-T},$$

and choosing

$$\tilde{\delta}_i = L^T \delta_i, \tag{6.11}$$

we can write the inner maximization problems as

$$\max_{\|\tilde{\delta}_i\|_2 \leq 1} \quad \tilde{g}_i \cdot \tilde{\delta}_i + \frac{1}{2}(\tilde{H}_i \tilde{\delta}_i) \cdot \tilde{\delta}_i, \tag{6.12}$$

where problems of these type are known as trust region subproblems (TRS). Note that by the transformation (6.11), the maximum value does not change, but we have to transform the solution $\tilde{\delta}_i$ back to $\delta_i$. (TRS) problems are well studied and solutions can be characterized.

**Theorem 6.1** (TRS solution, [56] Theorem 6.1)**.**
*Let $\tilde{H}_i$ be a symmetric matrix. Then the trust region problem (6.12) possesses at least one global solution. Moreover, $\tilde{\delta}_i$ is a global solution if and only if there exists a Lagrange multiplier $\lambda_i \in \mathbb{R}$ such that*

$$\begin{aligned} (-\tilde{H}_i + \lambda_i I)\tilde{\delta}_i - \tilde{g}_i &= 0, \\ \lambda_i(\|\tilde{\delta}_i\|_2 - 1) &= 0, \\ \|\tilde{\delta}_i\|_2 &\leq 1, \\ \lambda_i &\geq 0, \\ \tilde{H}_i - \lambda_i I &\preccurlyeq 0, \end{aligned} \tag{6.13}$$

*where $\tilde{H}_i - \lambda_i I \preccurlyeq 0$ denotes that the matrix $\tilde{H}_i - \lambda_i I$ is negative semidefinite and $I$ is the identity matrix.*

*Proof.* See for example [26]. □

There are different ways of approaching the quadratic approximation (6.10) of the robust counterpart. In [56] and [78] the optimality conditions (6.13) are added as constraints to the problem (6.10) to obtain a mathematical programm with complementarity constraints MPCC. We will use a different approach involving non-smooth objective and constraint functions, which was introduced in [52]. Defining the worst-case functions

$$\phi^i(\mu) := \max_{\delta_i \in \mathcal{U}} \; f^i(\mu, \bar{p}) + \nabla_p f^i(\mu, \bar{p}) \cdot \delta_i + \frac{1}{2}(\nabla_{pp} f^i(\mu, \bar{p})\delta_i) \cdot \delta_i,$$

we can write the quadratic approximation (6.10) of the robust counterpart as

$$\begin{aligned} \min_{\mu} \quad & \phi^0(\mu) \\ \text{s.t.} \quad & \phi^i(\mu) \leq 0, \text{ for } i \in I. \end{aligned} \tag{6.14}$$

In the next section, we will present properties of the worst-case functions $\phi^i(\mu)$.

### 6.1.3 Properties of worst-case functions

Functions involving max are in general not smooth, therefore the worst-case function $\phi^i(\mu)$ are in general not smooth. But under some condition they are regular in the sense of Clarke. We introduce the notation of the directional derivative and the generalized (or Clarke's) directional derivative which was introduced by Clarke in [25], since they are used in the subsequent Theorem 6.4. We start with the definition of the directional derivative.

**Definition 6.2** (Directional derivative)**.** *The directional derivative of $f : X \to \mathbb{R}$ at $x \in X$ in direction $v \in X$ is*

$$f'(x; v) = \lim_{t \searrow 0} \frac{f(x + tv) - f(x)}{t},$$

*if the limit exists.*

Accordingly we define the generalized directional derivative and the generalized gradient.

**Definition 6.3.** *Let $f : D \subseteq U \to \mathbb{R}$ be a locally Lipschitz-continuous function.*

1. *The generalized directional derivative of $f$ at $\mu \in D$ in the direction $v \in U$ is defined as*
$$f^\circ(\mu; v) = \limsup_{\substack{y \to \mu \\ t \searrow 0}} \frac{f(y + tv) - f(y)}{t},$$
*where $y + tv \in U$ and $t > 0$.*
2. *The generalized gradient or subdifferential of $f$ at $\mu \in D$, denoted by $\partial f(\mu)$, is the multifunction $\partial f : U \rightrightarrows U$*
$$\partial f(\mu) = \{g \in U : f^\circ(\mu; v) \geq \langle v, g \rangle \text{ for all } v \in U\}.$$
*An element $g \in \partial f(\mu)$ is called subgradient of $f$ at $\mu$.*

Under the assumptions of the following theorem, the worst-case functions $\phi^i(\mu)$ are regular in the sense of Clarke.

**Theorem 6.4** ([51] Theorem 2.25)**.**
*Consider the function $\phi(\mu) = \max_{\|p\| \leq 1} f(\mu, p)$ with $f : U \times P \to \mathbb{R}$. Suppose*

1. *$f$ is locally Lipschitz continuous in $(\mu, p)$.*
2. *$f^\circ_\mu(\mu, p; v) = f'_\mu(\mu, p; v)$ for all $v \in U$.*

*Then, the following holds for the worst-case function $\phi$:*

1. *$\phi$ is locally Lipschitz continuous.*
2. *$\phi$ is regular, i.e., $\phi'(\mu; v)$ exists and $\phi'(\mu; v) = \phi^\circ(\mu; v)$ for all $v \in U$.*
3. *The directional derivatives are*
$$\phi'(\mu; v) = \phi^\circ(\mu; v) = \max\{\langle g, v \rangle : g \in \partial_\mu f(\mu, p), p \in W(\mu)\},$$
*where the set*
$$W(\mu) := \{p \in \mathcal{U} : \phi_i(\mu) = f(\mu, p)\},$$
*contains the maximizers $p \in \mathcal{U}$ of the functions $f(\mu, p)$.*
4. *The subdifferential is given by*
$$\partial\phi(\mu) = \operatorname{conv}\{\partial_\mu f(\mu, p) : p \in W(\mu)\}.$$

When we assume that the objective and constraint functions are smooth enough, then the first and second assumption of the theorem are fulfilled, since differentiable functions are locally Lipschitz continuous and for differentiable functions the generalized gradient coincides with the gradient.

As we have seen in the theorem, if the set of maximizers $W(\mu)$ is a singleton, then the worst-case functions are differentiable. Since in our case, the worst-case functions are the trust region problems, we see in the next lemma when the solution to the TRS is unique, i.e. the worst-case functions are differentiable.

**Lemma 6.5** ([51] Lemma 3.13)**.**
*Let $\lambda_i \geq 0$ be the unique Lagrange multiplier at global maximizers of problem (6.12). Then one of the following two cases holds:*

1. *If $\lambda_i$ is greater than the maximum eigenvalue of $\tilde{H}_i$, there is a unique solution given by*
$$\tilde{\delta}_i = (-\tilde{H}_i + \lambda_i I)^{-1}\tilde{g}_i.$$

2. *Otherwise, $\lambda_i$ is equal to the maximum eigenvalue of $\tilde{H}_i$, and the solution set is given by*
$$W = \{\tilde{\delta}_i = \tilde{\delta}_i^* + Y\tau : \tau \in \mathbb{R}^m,\, \tilde{\delta}_i^T\tilde{\delta}_i \leq 1,\, \lambda_i(\tilde{\delta}_i^T\tilde{\delta}_i - 1) = 0\},$$
*where $m$ is the dimension of the null space of $\tilde{H}_i - \lambda_i I$, and $Y$ is a $n \times m$ matrix whose columns form an orthogonal basis of that null space. Let $\tilde{\delta}_i^* = (-\tilde{H}^i + \lambda_i I)^\dagger \tilde{g}_i$ be the least-norm solution of the stationarity equation, where $^\dagger$ denotes the Moore-Penrose pseudoinverse. Then one of the following two cases holds:*
   a) *The least-norm solution lies on the boundary, i.e., $\|\tilde{\delta}_i^*\| = 1$. In this case, the solution set $W$ is a singleton with $W = \{\tilde{\delta}_i^*\}$.*
   b) *The least-norm solution lies in the interior, i.e., $\|\tilde{\delta}_i^*\| < 1$. If $\tilde{H}^i$ is not negative semidefinite because of $\lambda_i > 0$ and $\lambda_i$ is equal to the largest eigenvalue of $\tilde{H}^i$, the solution set is an $(m-1)$-dimensional sphere, and if $\tilde{H}^i$ is negative semidefinite due to $\lambda_i = 0$, the solution set is an $m$-dimensional ball.*

In the literature, the case where the solution to the TRS problem is non-unique is known as the trust region hard case.

### 6.1.4 Shifting expansion point

If the approximated worst-case is not close to the nominal parameter, the approximated worst-case value might deviate from the true function value, which can lead to infeasible problems or non-robust solutions. To improve the approximation of the worst-case and the corresponding worst-case value, one has the freedom to shift

the expansion point in the quadratic model. Since we want to shift the expansion point, but keep the uncertainty set the same, we introduce a new parameter $\delta_e$ with $\bar{p} + \delta_e \in \mathcal{U}$ and expand the quadratic model in the point $\bar{p} + \delta_e$

$$\begin{aligned} f_e^i(\mu, \bar{p}, \delta_e^i, p) :=& f^i(\mu, \bar{p} + \delta_e^i) + \nabla_p f^i(\mu, \bar{p} + \delta_e^i) \cdot (p - (\bar{p} + \delta_e^i)) \\ &+ \frac{1}{2}(\nabla_{pp} f^i(\mu, \bar{p} + \delta_e^i)(p - (\bar{p} + \delta_e^i))) \cdot (p - (\bar{p} + \delta_e^i)), \end{aligned}$$

where $p \in \mathcal{U}$. By again, defining the deviation of the nominal parameter as $\delta$, we have $p - (\bar{p} + \delta_e) = p - \bar{p} - \delta_e = \delta - \delta_e$ and

$$\begin{aligned} f_e^i(\mu, \bar{p}, \delta_e^i, \delta_i) &= f^i(\mu, \bar{p} + \delta_e^i) + \nabla_p f^i(\mu, \bar{p} + \delta_e^i) \cdot \delta_i - \nabla_p f^i(\mu, \bar{p} + \delta_e^i) \cdot \delta_e^i \\ &+ \frac{1}{2}(\nabla_{pp} f^i(\mu, \bar{p} + \delta_e^i)\delta_i) \cdot \delta_i + \frac{1}{2}(\nabla_{pp} f^i(\mu, \bar{p} + \delta_e^i)\delta_e^i) \cdot \delta_e^i - (\nabla_{pp} f^i(\mu, \bar{p} + \delta_e^i)\delta_e^i) \cdot \delta_i \end{aligned}$$

Defining

$$\begin{aligned} c^i(\mu, \bar{p}, \delta_e^i) &:= f^i(\mu, \bar{p} + \delta_e^i) - \nabla_p f^i(\mu, \bar{p} + \delta_e^i) \cdot \delta_e^i + \frac{1}{2}(\nabla_{pp} f^i(\mu, \bar{p} + \delta_e^i)\delta_e^i) \cdot \delta_e^i, \\ g^i(\mu, \bar{p}, \delta_e^i) &:= \nabla_p f^i(\mu, \bar{p} + \delta_e^i) - (\nabla_{pp} f^i(\mu, \bar{p} + \delta_e^i)\delta_e^i), \end{aligned}$$

we can write

$$f_e^i(\mu, \bar{p}, \delta_e^i, \delta_i) = c^i(\mu, \bar{p}, \delta_e^i) + g^i(\mu, \bar{p}, \delta_e^i) \cdot \delta_i + \frac{1}{2}(\nabla_{pp} f^i(\mu, \bar{p} + \delta_e^i)\delta_i) \cdot \delta_i,$$

which is again a quadratic function in $\delta_i$. We then obtain the quadratic approximation of the robust counterpart with shifted expansion point

$$\begin{aligned} \min_{\mu} \quad & \max_{\delta_0 \in \mathcal{U}} \; f_e^0(\mu, \bar{p}, \delta_e^0, \delta_0) \\ \text{s.t.} \quad & \max_{\delta_i \in \mathcal{U}} \; f_e^i(\mu, \bar{p}, \delta_e^i, \delta_i) \leq 0, \text{ for } i \in I. \end{aligned} \tag{6.15}$$

We incorporate the shifting of the expansion point into our optimization in the following way: We start with the nominal parameter as expansion point for all the $f^i$. We then set up the quadratic model and compute the worst-case disturbance $\bar{\delta}_i$. We then compute $f^i(\mu, \bar{p} + \bar{\delta}_i)$ and compare it to $f_q^i(\mu, \bar{\delta}_i)$ (which requires to solve the state equation), if the difference exceeds a given threshold, we perform some steps of a global convergent optimization algorithm for $\max_{p \in \mathcal{U}} f^i(\mu, p)$ to obtain $p_i^*$ and use it as the new expansion point for $f^i$. Note that this can be done for every of the $f^i$ and when different expansion points for each of the involved functions are used, we have to solve the state equation and the sensitivity equation for different uncertain parameters to obtain $y(\mu, p_i^*)$ and $y_p(\mu, p_i^*)$ which is quite expensive.

## 6.2 Computation of Derivatives

In order to set up the model, one does need to compute derivatives of the reduced functions $f^i$ which depend implicitly on the solution operator of the state equation. For the first order derivative there are two approaches: the adjoint method und the sensitivity method. In the adjoint method an additional linearized state equation has to be solved backwards in time, where the right hand side involves the derivatives of the $F^i(y, \mu, p)$ with respect to the state variables. In the sensitivity approach a linearized state equation has to be solved $n_p$ times. In the following, subscripts will denote partial derivatives, i.e. $F_y(y, \mu, p) = \partial_y F(y, \mu, p)$.

### 6.2.1 Adjoint approach

To construct the first and second order Taylor expansion, we need the first order derivative of the reduced objective functions $f^i$ with respect to the uncertainty $p$. To compute a descent direction in the optimization, we need a derivative of the term $f^i(\mu, \bar{p})$ with respect to the design $\mu$ in the robust and non-robust optimization. We follow here [47]. Recall that the reduced functions are defined as (dropping here the index $i$ for readability)

$$f(\mu, p) := F(y(\mu, p), \mu, p),$$

where $y(\mu, \bar{p})$ is the solution of the state equation for given design $\mu$ and nominal uncertain parameter $\bar{p}$. Derivatives of $f(\mu, \bar{p})$ with respect to $\mu$ in direction $\delta u \in U$ are given by

$$\begin{aligned}\left\langle \frac{\partial}{\partial \mu} f(\mu, \bar{p}), \delta u \right\rangle_{U^*,U} &= \langle F_y(y(\mu, \bar{p}), \mu, \bar{p}), y_\mu(\mu, \bar{p})\delta u\rangle_{Y^*,Y} + \langle F_\mu(y(\mu, \bar{p}), \mu, \bar{p}), \delta u\rangle_{U^*,U} \\ &= \left\langle y_\mu^*(\mu, \bar{p}) F_y(y(\mu, \bar{p}), \mu, \bar{p}), \delta u\right\rangle_{U^*,U} + \langle F_\mu(y(\mu, \bar{p}), \mu, \bar{p}), \delta u\rangle_{U^*,U},\end{aligned}$$

where $y_\mu^*(\mu, \bar{p})$ is the adjoint mapping of $y_\mu(u, \bar{p})$. This leads to the formula for the derivative

$$\frac{\partial}{\partial \mu} f(\mu, \bar{p}) = y_\mu^*(\mu, \bar{p}) F_y(y(\mu, \bar{p}), \mu, \bar{p}) + F_\mu(y(\mu, \bar{p}), \mu, \bar{p}), \tag{6.16}$$

this involves the the derivative of the design-to-state mapping $y$, which depends on the dimension $U$. To bypass the possibly expensive computation of $y_\mu$ we use the

following. We know

$$e(y(\mu,\bar{p}),\mu,\bar{p}) = 0$$

has the derivative with respect to the design

$$e_y(y(\mu,\bar{p}),\mu,\bar{p})y_\mu(\mu,\bar{p}) + e_\mu(y(\mu,\bar{p}),\mu,\bar{p}) = 0$$

and therefore, the sensitivity is given by

$$y_\mu(\mu,\bar{p}) = -e_y(y(\mu,\bar{p}),\mu,\bar{p})^{-1}e_\mu(y(\mu,\bar{p}),\mu,\bar{p}) \tag{6.17}$$

plugging this into (6.16)

$$\frac{\partial}{\partial\mu}f(\mu,\bar{p}) = -e_\mu(y(\mu,\bar{p}),\mu,\bar{p})^* e_y(y(\mu,\bar{p}),\mu,\bar{p})^{-*}F_y(y(\mu,\bar{p}),\mu,\bar{p}) + F_\mu(y(\mu,\bar{p}),\mu,\bar{p})$$

setting $\lambda(\mu,\bar{p})$

$$-e_y(y(\mu,\bar{p}),\mu,\bar{p})^{-*}F_y(y(\mu,\bar{p}),\mu,\bar{p}) = \lambda(\mu,\bar{p})$$

which can be computed by solving

$$e_y(y(\mu,\bar{p}),\mu,\bar{p})^*\lambda(\mu,\bar{p}) = -F_y(y(\mu,\bar{p}),\mu,\bar{p}) \tag{6.18}$$

we can use the so called adjoint state $\lambda(\mu,\bar{p})$ to compute the derivative by

$$\frac{\partial}{\partial\mu}f(\mu,\bar{p}) = e_\mu(y(\mu,\bar{p}),\mu,\bar{p})^*\lambda(\mu,\bar{p}) + F_\mu(y(\mu,\bar{p}),\mu,\bar{p}).$$

This approch circumvents computing the whole sensitivity operator $y_\mu(\mu,\bar{p})$, but depends on the functions $f^i$, such that we have to compute the adjoint state for the objective and constraint functions. The computation of a derivative with respect to the uncertain parameter $p$ is analogous.

### 6.2.2 Sensitivity approach

When the sum of the number of objective and constraint functions exceeds the dimension of the design (or uncertain) parameters, the sensitivity approach is more efficient than the adjoint approach. One can compute the whole sensitivity $y_\mu$ via (6.17)

$$e_y(y(\mu,\bar{p}),\mu,\bar{p})y_\mu = -e_\mu(y(\mu,\bar{p}),\mu,\bar{p})$$

and use it to compute the derivatives of the involved functions

$$\frac{\mathrm{d}}{\mathrm{d}\mu} f(\mu) = y_\mu^* F_y(y(\mu, \bar{p}), \mu, \bar{p}) + F_\mu(y(\mu, \bar{p}), \mu, \bar{p}).$$

We will see, that for the efficient computation of second order derivatives, we will need the sensitivity aswell as the adjoint state.

### 6.2.3 Second order derivatives

For the quadratic model, we need the second order derivative of $f$ with respect to the uncertain parameter. To present a formula, we will follow [47] and use the Lagrangian.

**Definition 6.6.** *We define the Lagrange function for an equality constrained optimization problem of the form*

$$\begin{aligned} \min_{y,\mu} \quad & F(y, \mu, \bar{p}) \\ s.t. \quad & e(y, \mu, \bar{p}) = 0. \end{aligned}$$

*as* $L : Y \times U \times P \times Z^* \to \mathbb{R}$ *with*

$$L(y, \mu, p, \lambda) = F(y, \mu, p) + \langle \lambda, e(y, \mu, p) \rangle_{Z^*, Z},$$

*where* $\langle z, v \rangle_{Z^*, Z}$ *denotes the dual pairing* $z(v)$.

With this definition, we can write the function $f$ as

$$f(\mu, \bar{p}) = F(y(\mu, \bar{p}), \mu, \bar{p}) = L(y(\mu, \bar{p}), \mu, \bar{p}, \lambda)$$

since the second summand of the Lagrangian vanishes by the equality constraint being fulfilled, i.e. $e(y(\mu, \bar{p}), \mu, \bar{p})$ being 0. Differentiating $f$ in direction $s_1 \in P$ leads to

$$\begin{aligned} \langle f_p(\mu, \bar{p}), s_1 \rangle_{P^*, P} = & \langle L_y(y(\mu, \bar{p}), \mu, \bar{p}, \lambda), y_p(\mu, \bar{p}) s_1 \rangle_{Y^*, Y} \\ & + \langle L_p(y(\mu, \bar{p}), \mu, \bar{p}, \lambda), s_1 \rangle_{P^*, P}. \end{aligned}$$

Note that we did not use a specific $\lambda$ here. As a next step, we differentiate again in

direction $s_2 \in P$ and obtain

$$\begin{aligned}\langle f_{pp}(\mu,\bar{p})s_2, s_1\rangle_{P^*,P} = &\langle L_y(y(\mu,\bar{p}),\mu,\bar{p},\lambda), y_{pp}(\mu,\bar{p})(s_1,s_2)\rangle_{Y^*,Y}\\ &+ \langle L_{yy}(y(\mu,\bar{p}),\mu,\bar{p},\lambda)y_p(\mu,\bar{p})s_2, y_p(\mu,\bar{p})s_1\rangle_{Y^*,Y}\\ &+ \langle L_{yp}(y(\mu,\bar{p}),\mu,\bar{p},\lambda)s_2, y_p(\mu,\bar{p})s_1\rangle_{Y^*,Y}\\ &+ \langle L_{py}(y(\mu,\bar{p}),\mu,\bar{p},\lambda)y_p(\mu,\bar{p})s_2, s_2\rangle_{P^*,P}\\ &+ \langle L_{pp}(y(\mu,\bar{p}),\mu,\bar{p},\lambda)s_2, s_1\rangle_{P^*,P}.\end{aligned}$$

Since we do not want to compute the second order sensitivity $y_{pp}(\mu,\bar{p})$ we choose $\lambda = \lambda(\mu,\bar{p})$ such that $L_y(y(\mu,\bar{p}),\mu,\bar{p},\lambda(\mu,\bar{p})) = 0$ and arrive at

$$\begin{aligned}\langle f_{pp}(\mu,\bar{p})s_2, s_1\rangle_{P^*,P} = &\langle L_{yy}(y(\mu,\bar{p}),\mu,\bar{p},\lambda(\mu,\bar{p}))y_p(\mu,\bar{p})s_2, y_p(\mu,\bar{p})s_1\rangle_{Y^*,Y}\\ &+ \langle L_{yp}(y(\mu,\bar{p}),\mu,\bar{p},\lambda(\mu,\bar{p}))s_2, y_p(\mu,\bar{p})s_1\rangle_{Y^*,Y}\\ &+ \langle L_{py}(y(\mu,\bar{p}),\mu,\bar{p},\lambda(\mu,\bar{p}))y_p(\mu,\bar{p})s_2, s_2\rangle_{P^*,P}\\ &+ \langle L_{pp}(y(\mu,\bar{p}),\mu,\bar{p},\lambda(\mu,\bar{p}))s_2, s_1\rangle_{P^*,P}.\end{aligned}$$

Which means we get the formula for the second derivative

$$\begin{aligned}f_{pp}(\mu,\bar{p}) = &\, y_p^*(\mu,\bar{p})L_{yy}(y(\mu,\bar{p}),\mu,\bar{p},\lambda(\mu,\bar{p}))y_p(\mu,\bar{p}) + y_p^*(\mu,\bar{p})L_{yp}(y(\mu,\bar{p}),\mu,\bar{p},\lambda(\mu,\bar{p}))\\ &+ L_{py}(y(\mu,\bar{p}),\mu,\bar{p},\lambda(\mu,\bar{p}))y_p(\mu,\bar{p}) + L_{pp}(y(\mu,\bar{p}),\mu,\bar{p},\lambda(\mu,\bar{p})).\end{aligned}$$

This can be rewritten as

$$\frac{\partial^2}{\partial p^2} f(\mu,\bar{p}) = \begin{bmatrix} y_p \\ id_p \end{bmatrix}^* \begin{bmatrix} Lyy & Lyp \\ Lpy & Lpp \end{bmatrix} \begin{bmatrix} y_p \\ id_p \end{bmatrix}$$

where $id_p : P \to P$ is the identity map of the uncertainty set $P$. Note that we have computed $\lambda(\mu,\bar{p})$ such that $L_y(y(\mu,\bar{p}),\mu,\bar{p},\lambda)$ vanishes. Since

$$L_y(y(\mu,\bar{p}),\mu,\bar{p},\lambda) = F_y(y(\mu,\bar{p}),\mu,\bar{p}) + e_y^*(y(\mu,\bar{p}),\mu,\bar{p})\lambda$$

this is exactly solving the adjoint equation (6.18).

### 6.2.4 Computing descent directions

We have already seen, how we can compute derivatives for $f(\mu,\bar{p})$. In the Taylor expansion of first and second order, the first order derivative term $\nabla_p f(\mu,\bar{p})\cdot\delta$ has to be differentiated with respect to the design variable $\mu$. If we have computed the

sensitivity with respect to the uncertain parameter $y_p(\mu, \bar{p})$, we can write

$$\nabla_p f(\mu, \bar{p}) \cdot \delta = F_p(y(\mu, \bar{p}), \mu, \bar{p})\delta + F_y(y(\mu, \bar{p}), \mu, \bar{p}) y_p(\mu, \bar{p})\delta.$$

We define $\delta y(\mu, \bar{p}) := y_p(\mu, \bar{p})\delta$ . We follow Kolvenbach [52] and define

$$\begin{aligned} G(y, \mu, \delta y; z_1, z_2) := & F_y(y, \mu, \bar{p})\delta y + F_p(y, \mu, \bar{p})\delta \\ & + \langle z_1, e(y, \mu, \bar{p})\rangle + \langle z_2, e_y(y, \mu, \bar{p})\delta y + e_p(y, \mu, \bar{p})\delta\rangle \\ = & \langle z_1, e(y, \mu, \bar{p})\rangle + L_y(y, \mu, \bar{p}, z2)\delta y + L_p(y, \mu, \bar{p}, z_2)\delta. \end{aligned}$$

Inserting $y(\mu, \bar{p})$ and $\delta y = \delta y(\mu, \bar{p})$ and the second line vanishes, since then the state and sensitivity equation are fulfilled. The derivative with respect to the design $\mu$ is then given by

$$\frac{\mathrm{d}G(y(\mu, \bar{p}), \mu, \delta y(\mu, \bar{p}); z_1, z_2)}{\mathrm{d}\mu} = G_\mu + G_y y_\mu + G_{\delta y}\delta y_\mu,$$

where the $G's$ on the right hand side and in the following are evaluated at $(y(\mu, \bar{p}), \mu, \delta y(\mu, \bar{p}); z_1, z_2)$ and the $F's$ are evaluated at $(y(\mu, \bar{p}), \mu, \bar{p})$. As we do not want to compute the sensitivity $y_\mu$ and the second order sensitivity $\delta y_\mu = \partial_\mu y_p(\mu, \bar{p})\delta$, we choose $z_1$ and $z_2$ such that $G_y$ and $G_{\delta y}$ vanish. By inspecting

$$G_{\delta y}(y(\mu, \bar{p}), \mu, \delta y(\mu, \bar{p}); z_1, z_2) = F_y + \langle z_2, e_y(y(\mu, \bar{p}), \mu, \bar{p})\rangle \stackrel{!}{=} 0,$$

we can conclude, that $z_2(\mu, \bar{p}) = \lambda(\mu, \bar{p})$. Further,

$$G_y = F_{yy}\delta y + F_{py}\delta + \langle z_1, e_y\rangle + \langle z_2, e_{yy}\delta y + e_{py}\delta\rangle \stackrel{!}{=} 0,$$

leads to

$$\begin{aligned} e_y^* z_1 &= -F_{yy}\delta y - F_{py}\delta - \langle z_2, e_{yy}\delta y + e_{py}\delta\rangle \\ &= -L_{yy}\delta y - L_{py}\delta, \end{aligned}$$

since $z_2(\mu, \bar{p}) = \lambda(\mu, \bar{p})$. We have

$$\begin{aligned} & \frac{\mathrm{d}G(y(\mu, \bar{p}), \mu, \delta y(\mu, \bar{p}); z_1(\mu, \bar{p}), z_2(\mu, \bar{p}))}{\mathrm{d}\mu} \\ = & F_{y\mu}\delta y + F_{p\mu}\delta + \langle z_1(\mu, \bar{p}), e_\mu\rangle + \langle z_2(\mu, \bar{p}), e_{y\mu}\delta y + e_{p\mu}\delta\rangle \\ = & L_{y\mu}\delta y + L_{p\mu}\delta + \langle z_1(\mu, \bar{p}), e_\mu\rangle. \end{aligned}$$

This means that we have to solve an additional adjoint equation for the computation of the derivative with respect to the design of the first order terms in the linear and quadratic Taylor expansion.

The same procedure can be done to calculate a derivative of the second order term. We define

$$H(y,\mu,p,\delta y,\lambda;z_3,z_4,z_5) := L_{yy}[\delta y,\delta y] + L_{yp}[\delta y,\delta] + L_{py}[\delta,\delta y] + L_{pp}[\delta,\delta] \\ + \langle z_3, e\rangle + \langle z_4, e_y\delta y + e_p\delta\rangle + \langle z_5, L_y\rangle,$$

where the $L's$ depend on $(y,\mu,p,\lambda)$ and $[\cdot,\cdot]$ contains the two directions for the second order derivatives.

Inserting $y = y(\mu,\bar{p}), \delta y = \delta y(\mu,\bar{p}), \lambda = \lambda(\mu,\bar{p})$ the second line vanishes. The derivative of $H(y(\mu,\bar{p}),\mu,\bar{p},\delta y(\mu,\bar{p}),\lambda(\mu,\bar{p});z_1,z_2,z_3)$ with respect to $\mu$ is given by $H_y y_\mu + H_\mu + H_{\delta y}\delta y_\mu + H_\lambda\lambda_\mu$. We will again choose $z_3, z_4, z_5$ such that we do not have to compute $y_\mu$, $\delta y_\mu$ and $\lambda_\mu$. The evaluation of the $L's$ in the following is $(y(\mu,\bar{p}),\mu,\bar{p},\lambda(\mu,\bar{p}))$. The equation

$$H_{\delta y}(y(\mu,\bar{p}),\mu,\bar{p},\delta y(\mu,\bar{p}),\lambda(\mu,\bar{p});z_3,z_4,z_5) = 2L_{yy}[\delta_y,\cdot] + L_{yp}[\cdot,\delta] + L_{py}[\delta,\cdot] \\ + \langle z_4, e_y\rangle \stackrel{!}{=} 0,$$

leads to the adjoint equation

$$e_y^* z_4 = -2L_{yy}[\delta_y,\cdot] - L_{yp}[\cdot,\delta] - L_{py}[\delta,\cdot].$$

Note that $z_4(\mu,\bar{p}) = 2z_1(\mu,\bar{p})$, which we can reuse and do not need to compute twice. The equation

$$H_\lambda = e_{yy}[\delta_y,\delta_y] + e_{yp}[\delta y,\delta] + e_{py}[\delta,\delta y] + e_{pp}[\delta,\delta] \\ + \langle z_5, e_y^*\rangle \stackrel{!}{=} 0,$$

leads to the sensitivity equation

$$e_y z_5 = -e_{yy}[\delta_y,\delta_y] - e_{yp}[\delta y,\delta] - e_{py}[\delta,\delta y] - e_{pp}[\delta,\delta].$$

And finally

$$H_y = L_{yyy}[\delta_y,\delta_y,\cdot] + L_{ypy}[\delta y,\delta,\cdot] + L_{pyy}[\delta,\delta y,\cdot] + L_{ppy}[\delta,\delta,\cdot] \\ + \langle z_3, e_y\rangle + \langle z_4, e_{yy}\delta y + e_{py}\delta\rangle + \langle z_5, L_{yy}\rangle \stackrel{!}{=} 0,$$

leads to the adjoint equation

$$e_y^* z_3 = -L_{yyy}[\delta_y,\delta_y,\cdot] - L_{ypy}[\delta y,\delta,\cdot] - L_{pyy}[\delta,\delta y,\cdot] - L_{ppy}[\delta,\delta,\cdot] \\ - \langle z_4, e_{yy}\delta y + e_{py}\delta\rangle - \langle z_5, L_{yy}\rangle.$$

After solving the two additional adjoint equations and the sensitivity equation, we can compute the derivative of the Hessian with respect to the design via

$$\begin{aligned}\frac{\mathrm{d}H}{\mathrm{d}\mu} &= L_{yy\mu}[\delta y, \delta y, \cdot] + L_{yp\mu}[\delta y, \delta, \cdot] + L_{py\mu}[\delta, \delta y, \cdot] + L_{pp\mu}[\delta, \delta, \cdot] \\ &\quad + \langle z_1(\mu, \bar{p}), e_\mu \rangle + \langle z_2(\mu, \bar{p}), e_{y\mu}\delta y + e_{p\mu}\delta \rangle + \langle z_3(\mu, \bar{p}), L_{y\mu} \rangle.\end{aligned}$$

## 6.3 Interpolation based approximation

As we have seen in the previous section, to construct a quadratic Taylor model, we have to compute a lot of derivatives, which can be quite cumbersome and cannot be done if one only has access to a blackbox solver and no access to the code. A way to obtain a quadratic model which does not require derivative computation relies on polynomial interpolation. The model is then generated using only function evaluations, see [67, 27, 71]. The objective and constraint functions in (6.5) are then replaced by an interpolation model. Since we want the interpolation model to be a good approximator of the inner maximization problem, the model is iteratively improved by adjusting the set of interpolation points. In the following description, we fix a specific design $\mu \in U$ and inspect the function $g : \mathbb{R}^{n_p} \to \mathbb{R}$ where $g(\cdot) = f(\mu, \cdot)$.

We are interested in constructing a polynomial of degree $d$ in dimension $n_p$ ($n_p$ is the dimension of the set of uncertainties). We call the space of polynomials $\mathcal{P}^d_{n_p}$ of degree less or equal to $d$. The dimension of the space is denoted by $\mathfrak{p}_1 = \mathfrak{p} + 1$. For a linear polynominal in $\mathbb{R}^{n_p}$ we have $\mathfrak{p}_1 = n_p + 1$ and for a quadratic polynomial it is known, that $\mathfrak{p}_1 = \frac{1}{2}(n_p + 1)(n_p + 2)$.

Let $\phi = [\phi_0(x), \ldots, \phi_{\mathfrak{p}}(x)]$ be a basis of $\mathcal{P}^d_{n_p}$, i.e. any polynomial $m(x)$ in $\mathcal{P}^d_{n_p}$ can be written as $m(x) = \sum_{i=0}^{\mathfrak{p}} \alpha_i \phi_i(x)$ with $\alpha_i \in \mathbb{R}$. Since we want to construct an interpolation polynomial, we require the polynominal to fullfil $\mathfrak{p}_1$ interpolation conditions, i.e.

$$m(y^j) = \sum_{i=0}^{\mathfrak{p}} \alpha_i \phi_i(y^j) = g(y^j), \ j = 0, \ldots, \mathfrak{p}, \tag{6.19}$$

with an interpolation set $Y = \{y^0, \ldots, y^{\mathfrak{p}}\}$ with $y^i \in \mathbb{R}^{n_p}$. The condititions (6.19) can be rewritten in matrix form:

$$M(\phi, Y)\alpha_\phi = g(Y), \tag{6.20}$$

with

$$M(\phi, Y) := \begin{pmatrix} \phi_0(y^0) & \phi_1(y^0) & \cdots & \phi_{\mathfrak{p}}(y^0) \\ \phi_0(y^1) & \phi_1(y^1) & \cdots & \phi_{\mathfrak{p}}(y^1) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_0(y^{\mathfrak{p}}) & \phi_1(y^{\mathfrak{p}}) & \cdots & \phi_{\mathfrak{p}}(y^{\mathfrak{p}}) \end{pmatrix},$$

$$\alpha_\phi := \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{\mathfrak{p}} \end{pmatrix} \text{ and } g(Y) := \begin{pmatrix} g(y^0) \\ g(y^1) \\ \vdots \\ g(y^{\mathfrak{p}}) \end{pmatrix}$$

**Definition 6.7.** *The set $Y = \{y^0, \ldots, y^{\mathfrak{p}}\}$ is poised for polynomial interpolation in $\mathbb{R}^{n_p}$ if the corresponding matrix $M(\phi, Y)$ is nonsingular for some basis $\phi$ in $\mathcal{P}^d_{n_p}$ (the dimension of $\mathcal{P}^d_{n_p}$ is $\mathfrak{p}_1 = \mathfrak{p} + 1$).*

**Lemma 6.8** ([27] Lemma 3.2)**.**
*Given a function $g : \mathbb{R}^{n_p} \to \mathbb{R}$ and a poised set $Y \in \mathbb{R}^{n_p}$, the interpolating polynomial $m(x)$ exists and is unique.*

Note when the interpolation set contains more than $\mathfrak{p}_1$ points, then the system (6.20) is overdetermined, i.e. a regression problem which can be treated with a least squares approach.

**Definition 6.9.** *The set $Y = \{y^0, \ldots, y^q\}$ with $q > \mathfrak{p}$ is poised for polynomial regression in $\mathbb{R}^{n_p}$ if the corresponding matrix $M(\phi, Y)$ has full column rank for some basis $\phi$ in $\mathcal{P}^d_{n_p}$ (the dimension of $\mathcal{P}^d_{n_p}$ is $\mathfrak{p}_1 = \mathfrak{p} + 1$).*

Since in PDE optimization the objective and constraint functions may depend on the solution of the PDE, the function evaluations are expensive and we therefore want to use as few interpolation points as possible. As we want to use the interpolation model to solve the inner maximization problem in (6.5) and it is known, that by using linear models and therefore exploiting no curvature information slows down the convergence significantly [27], we use underdetermined quadratic interpolation, i.e. constructing a quadratic model by using more interpolation points than in linear interpolation and less than in quadratic interpolation. We then hope that enough curvature of the underlying function is captured, such that we experience an empirical superlinear rate of local convergence. Using fewer points in quadratic interpolation than $\mathfrak{p}_1$, lets say $n_p + 1 < \tilde{\mathfrak{p}}_1 := \tilde{\mathfrak{p}} + 1 < \mathfrak{p} + 1 = \mathfrak{p}_1 = \frac{1}{2}(n_p + 2)(n_p + 1)$ leads to the system matrix

$$\tilde{M}(\phi, \tilde{Y}) = \begin{pmatrix} \phi_0(y^0) & \phi_1(y^0) & \cdots & \phi_{\mathfrak{p}}(y^0) \\ \phi_0(y^1) & \phi_1(y^1) & \cdots & \phi_{\mathfrak{p}}(y^1) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_0(y^{\tilde{\mathfrak{p}}}) & \phi_1(y^{\tilde{\mathfrak{p}}}) & \cdots & \phi_{\mathfrak{p}}(y^{\tilde{\mathfrak{p}}}) \end{pmatrix} \in \mathbb{R}^{\tilde{\mathfrak{p}}_1 \times \mathfrak{p}_1},$$

i.e. given $\tilde{M}(\phi, \tilde{Y})$ has column rank $\tilde{\mathfrak{p}}_1$, solutions for arbitrary right hand side exist, but are not unique. We denote models obtained from such an underdetermined system *quadratic underdetermined interpolation models.*

**Assumption 6.10.** *We assume, that $\tilde{Y} = \{y^0, \ldots, y^{\mathfrak{p}}\} \subset \mathbb{R}^{n_p}$ is a set of sample points poised in the linear interpolation sense (or in the linear regression sense if $\mathfrak{p} > n_p$) contained in the ball $B(y^0; \Delta(\tilde{Y}))$ of radius $\Delta = \Delta(\tilde{Y})$. Further, we assume that the function $g$ is continuously differentiable on an open domain $\Omega$ containing $B(y^0; \Delta)$ and $\nabla g$ is Lipschitz continuous in $\Omega$ with constant $\nu > 0$. Let*

$$\hat{L} := \frac{1}{\Delta} L = \frac{1}{\Delta} \begin{pmatrix} y^1 - y^0 & \cdots & y^{\mathfrak{p}} - y^0 \end{pmatrix}^T$$

*and $\hat{L}^\dagger = (\hat{L}^T \hat{L})^{-1} \hat{L}^T$ be its pseudoinverse.*

**Theorem 6.11** ([27] Theorem 5.4)**.**
*Let Assumption (6.10) hold. Then, for all points $y$ in $B(y^0; \Delta(Y))$, we have that*

- *the error between the gradient of a quadratic underdetermined interpolation model and the gradient of the function satisfies*

$$\|\nabla g(y) - \nabla m(y)\| \leq \frac{5\sqrt{\mathfrak{p}}}{2} \|\hat{L}^\dagger\| (\nu + \|H\|) \Delta,$$

- *the error between a quadratic underdetermined interpolation model and the function satisfies*

$$\|g(y) - m(y)\| \leq \frac{5\sqrt{p}}{2} \|\hat{L}^\dagger\| (\nu + \|H\|) \Delta^2 + \frac{1}{2} (\nu + \|H\|) \Delta^2,$$

*where $H$ is the Hessian of the model.*

As the error estimators in Theorem (6.11) suggests, we will use quadratic underdetermined interpolation models with minimal Frobenius norm of the Hessian, as used in [67, 27, 71].

We can write the interpolation conditions as

$$m(y^i) = c + \mathfrak{g}^T y^i + \frac{1}{2}(y^i)^T H y^i = g(y^i),\ i = 0, \dots, \tilde{\mathfrak{p}}$$

where the degrees of freedom are given by $c \in \mathbb{R}$, $\mathfrak{g} \in \mathbb{R}^{n_p}$, $H \in \mathbb{R}^{n_p \times n_p}$ and the degrees of freedom correspond to the $\phi_\alpha$'s in (6.20).

In the following we will center the model $m(x)$ around the interpolation point $y^0$. The optimization problem associated to the construction of a minimal Frobenius norm model is given by

$$\begin{aligned} \min_{c,\mathfrak{g},H}\ & \frac{1}{4}\|H\|_F^2 \\ \text{s.t. } & m(\delta_j) = g(y^0 + \delta_j), \quad j = 0, \dots, \tilde{\mathfrak{p}} \\ & H = H^T \end{aligned} \tag{6.21}$$

where the model fulfills $m(0) = g(y^0)$ and $\delta_j := y^j - y^0, j = 0, \dots, \tilde{\mathfrak{p}}$. The Lagrangian $\mathcal{L} : \mathbb{R} \times \mathbb{R}^{n_p} \times \mathbb{R}^{n_p \times n_p} \times \mathbb{R}^{\tilde{\mathfrak{p}}_1} \to \mathbb{R}$ for this problem is given by

$$\begin{aligned} \mathcal{L}(c, \mathfrak{g}, H, \lambda) &= \frac{1}{4}\|H\|_F^2 - \sum_{j=0}^{\tilde{\mathfrak{p}}} \lambda_j (c + \mathfrak{g}^T \delta_j + \frac{1}{2}\delta_j^T H \delta_j - g(y^0 + \delta_j)) \\ &= \frac{1}{4}\sum_{i,j=1}^{n_p} H_{i,j}^2 - \sum_{j=0}^{\tilde{\mathfrak{p}}} \lambda_j (c + \mathfrak{g}^T \delta_j + \frac{1}{2}\delta_j^T H \delta_j - g(y^0 + \delta_j)), \end{aligned}$$

where the condition $H = H^T$ is omitted, since as we will see, it is fulfilled in the solution. The KKT conditions for (6.21) are given by

$$\partial_c \mathcal{L} = \sum_{j=0}^{\tilde{\mathfrak{p}}} \lambda_j = 0, \tag{6.22}$$

$$\partial_{\mathfrak{g}} \mathcal{L} = \sum_{j=0}^{\tilde{\mathfrak{p}}} \lambda_j \delta_j = 0, \tag{6.23}$$

$$\partial_H \mathcal{L} = \frac{1}{2} H - \sum_{j=0}^{\tilde{\mathfrak{p}}} \lambda_j (\frac{1}{2}\delta_j \delta_j^T) = 0, \tag{6.24}$$

$$\partial_{\lambda_j} \mathcal{L} = c + \mathfrak{g}^T \delta_j + \frac{1}{2}\delta_j^T H \delta_j - g(y^0 + \delta_j) = 0,\ j = 0, \dots, \tilde{\mathfrak{p}}. \tag{6.25}$$

By (6.24) we see, that $H = \sum_{j=0}^{\tilde{\mathfrak{p}}} \lambda_j (\delta_j \delta_j^T)$ and since $(\delta_j \delta_j^T)^T = \delta_j \delta_j^T$ it follows that $H$ is a weighted sum of symmetric matrices, i.e. $H$ is symmetric. Plugging (6.24)

into (6.25) leads to

$$A\lambda + ec + X^T\mathfrak{g} = G,$$

where $A \in \mathbb{R}^{\tilde{\mathfrak{p}}_1 \times \tilde{\mathfrak{p}}_1}$ with the components $A_{i,j} = \frac{1}{2}(\delta_i^T \delta_j)^2$, $e \in \mathbb{R}^{\tilde{\mathfrak{p}}_1}$ is a column vector of ones, $X \in \mathbb{R}^{n_p \times \tilde{\mathfrak{p}}_1}$ is a matrix whose $j$-th column is the $j$-th interpolation point $\delta_j$ and $G \in \mathbb{R}^{\tilde{\mathfrak{p}}_1}$ is a column vector with the function evaluation $g(y^0 + \delta_j)$ in the $j$-th row. Together with (6.22) and (6.23) this leads to the system

$$\left(\begin{array}{c|cc} A & e & X^T \\ \hline e^T & & \\ X & & 0 \end{array}\right) \begin{pmatrix} \lambda \\ \hline c \\ \mathfrak{g} \end{pmatrix} = \begin{pmatrix} G \\ \hline 0 \\ 0 \end{pmatrix}, \tag{6.26}$$

where the Hessian $H$ can be calculated after solving the system (6.26) by (6.24). In the robust optimization, the reduced functions are $f : U \times P \to \mathbb{R}$ to compute a model in the variable $p$. Further the model can then be used to solve the inner maximization problem in (6.5) by iteratively finding its maximum and then replacing one of the interpolation points with the new point, see Algorithm (6.1). In Algorithm

**Algorithm 6.1** Derivative free optimization

**Input:** $Y = \{y^0, \ldots, y^{\tilde{\mathfrak{p}}}\} \subset \mathbb{R}^{n_p}$ poised for quadratic underdetermined interpolation, function $f(\mu, \cdot) : \mathbb{R}^{n_p} \to \mathbb{R}$ to maximize over the uncertainty set $\mathcal{U}$, tolerance $\varepsilon > 0$, maximum number of improving steps $\tilde{k}$

**Output:** $p* = \underset{p \in \mathcal{U}}{\operatorname{argmax}} f(\mu, p)$.

1: Use $Y$ to compute initial minimal curvature model $m(p)$ for $f(\mu, \cdot)$ by solving (6.21).
2: Compute $p^* = \underset{p \in \mathcal{U}}{\operatorname{argmax}}\, m(p)$.
3: **for** $k = 1, 2, \ldots, \tilde{k}$ **do**
4: Set $Y = \big(Y \backslash \{y^i\}\big) \cup \{p^*\}$, $i \in \{1, \ldots, \tilde{\mathfrak{p}}\}$.
5: Update model $m(p)$ for new interpolation set $Y$.
6: Compute $p^* = \underset{p \in \mathcal{U}}{\operatorname{argmax}}\, m(p)$.
7: Set $f_k = f(\mu, p^*)$.
8: **if** k>1 **then**
9: **if** $|f_k - f_{k-1}| \leq \varepsilon$ **then**
10: Terminate.

(6.1), step 4, we need to pick a point in the interpolation set, which is replaced by the point $p^*$ which maximizes the current model $m(p)$. We follow [71] and use

$$i = \underset{j=0,\ldots,\tilde{\mathfrak{p}}}{\operatorname{argmax}} |\mathfrak{l}_j(y^0 + p^*)| \cdot \max\left(\|y_j - p^*\|^4\right).$$

Here $\mathfrak{l}_j(\cdot)$ denotes for $j = 0, \ldots, \tilde{\mathfrak{p}}$ the Lagrange polynomials which are the minimal curvature models which are obtained by solving (6.26) for $G$ being the 0 vector except for the $j$-th component being 1, i.e. we have $\mathfrak{l}_j(y^i - y^0) = \delta_{ij}$ for $i, j = 0, \ldots, \tilde{\mathfrak{p}}$ ($\delta_{ij}$ is the Kronecker delta). It is known that $\max_{p\in\mathcal{U}} \max_{j=0,\ldots,\tilde{\mathfrak{p}}} |\mathfrak{l}_j(p)|$ is a measure on how well an interpolation set is poised for interpolating a function on $\mathcal{U}$ and smaller values are favorable, see [27]. To compute a search direction for the design variable in the optimization we construct a model for the design derivative. This can be done by simply replacing the function evaluations contained in $G$ in (6.26) by the evaluation of the partial derivatives, i.e. $f_\mu(\mu, y_0 + \delta_j)$.

## 6.4 Numerical aspects

As we have seen in this chapter, in order to compute Taylor expansions and their design derivatives, we need to assemble many derivatives. We will present here some of the formulas for the derivatives of the discrete state equation. We have seen in (4.6) that the derivative with respect to the state variable of the discrete state equation has the structure (we now have the dependence on the design variable $\mu$ and the uncertain parameter $p$)

$$C_\mathfrak{y}(\mathfrak{y}, \mu, p) = \begin{pmatrix} C^1_{\mathfrak{y}_1}(\mathfrak{y}_0, \mathfrak{y}_1, \mu, p) & 0 & 0 & 0 \\ C^2_{\mathfrak{y}_1}(\mathfrak{y}_1, \mathfrak{y}_2, \mu, p) & C^2_{\mathfrak{y}_2}(\mathfrak{y}_1, \mathfrak{y}_2, \mu, p) & 0 & 0 \\ 0 & \ldots & \ldots & 0 \\ 0 & 0 & C^k_{\mathfrak{y}_{k-1}}(\mathfrak{y}_{k-1}, \mathfrak{y}_k, \mu, p) & C^k_{\mathfrak{y}_k}(\mathfrak{y}_{k-1}, \mathfrak{y}_k, \mu, p) \end{pmatrix}.$$

Remember $\mathfrak{y}_i = (y^{h,i}_{dof}, \mathfrak{T}^i, \theta^i, \dot\theta^i)^T = (u^{h,i}_{dof}, \varphi^i, i^i_{st}, \mathfrak{T}^i, \theta^i, \dot\theta^i)^T$. We denote $n_\mathfrak{y} := (n_{dof} + n_\varphi + n_i + 3)k$, where $k$ is the number of timesteps. If we multiply the matrix $C_\mathfrak{y}(\mathfrak{y}, \mu, p)$ with a direction $d\mathfrak{y} \in \mathbb{R}^{n_\mathfrak{y}}$ (we will use the same partition for the direction as for the state variable, i.e. $d\mathfrak{y}_i$ has the components $(dy^{h,i}, d\mathfrak{T}^i, d\theta^i, d\dot\theta^i)^T = (du^{h,i}, d\varphi^i, di^i_{st}, d\mathfrak{T}^i, d\theta^i, d\dot\theta^i)^T$ for $i = 1, \ldots, k$, we obtain the vector (we drop the inputs of the $C^i$ for readability)

$$C_\mathfrak{y} d\mathfrak{y} = \begin{pmatrix} C^1_{\mathfrak{y}_1} d\mathfrak{y}_1 \\ C^2_{\mathfrak{y}_1} d\mathfrak{y}_1 + C^2_{\mathfrak{y}_2} d\mathfrak{y}_2 \\ \vdots \\ C^k_{\mathfrak{y}_{k-1}} d\mathfrak{y}_{k-1} + C^k_{\mathfrak{y}_k} d\mathfrak{y}_k, \end{pmatrix}$$

which we can multiply with the discrete adjoint $\lambda^h$ from the left

$$(\lambda^h)^T C_{\mathfrak{y}} d\mathfrak{y} = (\lambda^{h,1})^T C^1_{\mathfrak{y}_1} d\mathfrak{y}_1 + (\lambda^{h,2})^T (C^2_{\mathfrak{y}_1} d\mathfrak{y}_1 + C^2_{\mathfrak{y}_2} d\mathfrak{y}_2) + ... + (\lambda^{h,k})^T (C^k_{\mathfrak{y}_{k-1}} d\mathfrak{y}_{k-1} + C^k_{\mathfrak{y}_k} d\mathfrak{y}_k).$$

Differentiating with respect to $\mathfrak{y}$ leads to

$$((\lambda^h)^T C_{\mathfrak{y}} d\mathfrak{y})_{\mathfrak{y}} = \begin{pmatrix} ((\lambda^{h,1})^T C^1_{\mathfrak{y}_1} d\mathfrak{y}_1 + (\lambda^{h,2})^T (C^2_{\mathfrak{y}_1} d\mathfrak{y}_1 + C^2_{\mathfrak{y}_2} d\mathfrak{y}_2))_{\mathfrak{y}_1} \\ ((\lambda^{h,2})^T (C^2_{\mathfrak{y}_1} d\mathfrak{y}_1 + C^2_{\mathfrak{y}_2} d\mathfrak{y}_2) + (\lambda^{h,3})^T (C^3_{\mathfrak{y}_2} d\mathfrak{y}_2 + C^3_{\mathfrak{y}_3} d\mathfrak{y}_3))_{\mathfrak{y}_2} \\ \vdots \\ ((\lambda^{h,k-1})^T (C^{k-1}_{\mathfrak{y}_{k-2}} d\mathfrak{y}_{k-2} + C^{k-1}_{\mathfrak{y}_{k-1}} d\mathfrak{y}_{k-1}) + (\lambda^{h,k})^T (C^k_{\mathfrak{y}_{k-1}} d\mathfrak{y}_{k-1} + C^k_{\mathfrak{y}_k} d\mathfrak{y}_k))_{\mathfrak{y}_{k-1}} \\ (\lambda^{h,k})^T (C^k_{\mathfrak{y}_{k-1}} d\mathfrak{y}_{k-1} + C^k_{\mathfrak{y}_k} d\mathfrak{y}_k))_{\mathfrak{y}_k} \end{pmatrix}^T,$$

in our state system the mixed derivatives

$$C^i_{\mathfrak{y}_i, \mathfrak{y}_{i-1}} \text{ and } C^i_{\mathfrak{y}_{i-1}, \mathfrak{y}_i}$$

vanish, such that we have

$$((\lambda^h)^T C_{\mathfrak{y}} d\mathfrak{y})_{\mathfrak{y}} = \begin{pmatrix} ((\lambda^{h,1})^T C^1_{\mathfrak{y}_1} d\mathfrak{y}_1)_{\mathfrak{y}_1} + ((\lambda^{h,2})^T C^2_{\mathfrak{y}_1} d\mathfrak{y}_1)_{\mathfrak{y}_1} \\ ((\lambda^{h,2})^T C^2_{\mathfrak{y}_2} d\mathfrak{y}_2))_{\mathfrak{y}_2} + ((\lambda^{h,3})^T C^3_{\mathfrak{y}_2} d\mathfrak{y}_2)_{\mathfrak{y}_2} \\ \vdots \\ ((\lambda^{h,k-1})^T C^{k-1}_{\mathfrak{y}_{k-1}} d\mathfrak{y}_{k-1}))_{\mathfrak{y}_{k-1}} + ((\lambda^{h,k})^T C^k_{\mathfrak{y}_{k-1}} d\mathfrak{y}_{k-1})_{\mathfrak{y}_{k-1}} \\ (\lambda^{h,k})^T (C^k_{\mathfrak{y}_k} d\mathfrak{y}_k)_{\mathfrak{y}_k} \end{pmatrix}^T.$$

The two blocks which are needed are given for linear material by

$$(C^i_{\mathfrak{y}_{i-1}} d\mathfrak{y}_{i-1})_{\mathfrak{y}_{i-1}} = \begin{pmatrix} 0 & 0 & 0 & -2\beta d\dot{\theta}^{i-1} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

and

$$(C^i_{\mathfrak{y}_i} d\mathfrak{y}_i)_{\mathfrak{y}_i} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ a & 0 & b & 0 \\ c & 0 & d & 0 \end{pmatrix},$$

with

$$
\begin{aligned}
a &:= \partial_\theta K_{full}(\theta^i, \mu, p) d\theta^i, \\
b &:= (\partial_\theta)^2 K_{full}(\theta^i, \mu, p) y^i_{dof} d\theta_i + \partial_\theta K_{full}(\theta^i, \mu, p) dy^{h,i}, \\
c &:= \left[ -(\partial_y)^2 M_{em}(u^{h,i}_{dof}, \theta^i) dy^{h,i} - \partial_\theta \partial_u M_{em}(u^{h,i}_{dof}, \theta^i) d\theta^i, 0 \in \mathbb{R}^{1 \times n_\varphi}, 0 \in \mathbb{R}^{1 \times n_i} \right], \\
d &:= -\partial_y \partial_\theta M_{em}(u^{h,i}_{dof}, \theta^i) du^{h,i} - (\partial_\theta)^2 M_{em}(u^{h,i}_{dof}, \theta^i) d\theta^i.
\end{aligned}
$$

CHAPTER 7

# Optimization

In this chapter, the numerical results for the (robust) optimization and the model order reduction of the induction machine are presented. In the first section, the results for the optimization problem with linearized material are stated and in the second one, the results for the model with the nonlinear material. In all of the performed optimizations, the optimization parameters are the width and the height of rotor bars. The transformations are performed as described in (4.3).

## 7.1 Linear material

In the optimization with the model of an induction machine with linearized material, the material (for the material distribution see Figure 7.1) is given by

| | air | cust | curt | I200 |
|---|---|---|---|---|
| $\nu$ | $7.9 \cdot 10^{05}$ | $7.9 \cdot 10^{05}$ | $7.9 \cdot 10^{05}$ | 142.439 |
| $\sigma$ | 0 | 0 | 43000000 | 0 |

**Table 7.1.** Material values in the linearized model.

As uncertainty, we use the width of the opening of the rotor bar slots as uncertain and parametrize them by 20 variables. As the uncertainty set we have chosen $\mathcal{U} = \{\bar{p} + \delta \in \mathbb{R}^{20} : \|\delta\|_2 \leq 0.25\}$ (see Figure 7.2). The division in triangles to perform the transformations is shown in Figure 4.4. We have a total of 27333 nodes in $\Omega^h$, 20023 in $\Omega_r^h$ and 7310 in $\Omega_s^h$. The uneven distribution of nodes comes from the division of the rotor into subdomains for shape optimization. The time step is $\Delta t = 10^{-03}$. We perform the optimization for the steady state. We therefore

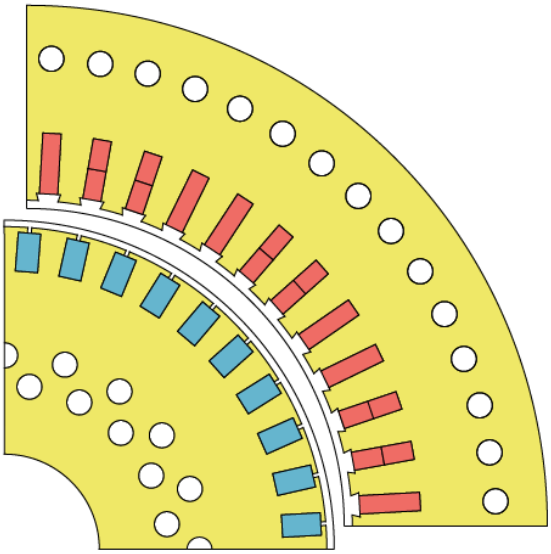

**Figure 7.1.** Material distribution in the machine: white - air, yellow - I200, red - cust, blue - curt.

computed 2000 timesteps with initial value 0 and then used the state $\mathfrak{y}^{h,2000}$ as initial value for the simulation in the optimization. We supplied box constraints for the design $\mu \in \{\mu \in \mathbb{R}^2 : \mu_1 \in [1, 28], \mu_2 \in [5, 19]\}$ ($\mu_1$ is the height and $\mu_2$ is the width). Whenever we have to solve the trust-region subproblem, we use a safeguarded Newton algorithm, see [26, 61].

The nominal optimization problem we are solving is given in continuous form

$$\begin{aligned} \min_{\mathfrak{y},\mu} \quad & \frac{4}{T-\mathrm{th}} \int_{\mathrm{th}}^{T} \sum_{i=1}^{10} \int_{\Omega_{\mathrm{bar},i}(\mu)} \sigma \left( \mathbf{1}_{\Omega_{\mathrm{bar},i}(\mu)}(x) \mathbf{A}_{i,\cdot}^{\top} \varphi(t) - \frac{\mathrm{d}u_1(x,t)}{\mathrm{d}t} \right)^2 \mathrm{d}x \, \mathrm{d}t \\ \text{s.t.} \quad & D(\mu) \leq 0, \\ & T^d \leq \frac{1}{T-\mathrm{th}} \int_{\mathrm{th}}^{T} M_{em}(u,\theta) \, \mathrm{d}t, \\ & e(\mathfrak{y}, \mu, \bar{p}) = 0. \end{aligned}$$

The objective function are the averaged (over a given time-horizon $[th, T]$) Joule losses in the rotor bars for the full machine (assuming symmetry), due to the squared term in the integral being the voltages in the rotor bars, which are then multiplied by the conductivity $\sigma$ and $Power = Current \times Voltage$. The constraint function states that the averaged torque should be at least a given desired torque $T^d$. And $D(\mu)$ contains lower and upper bounds for the width and height of the rotor bars. In the optimization we perform 500 timesteps and use the last 300 for averaging.

After discretization the optimization problem is given by

$$\begin{aligned}
\min_{\mathfrak{y}^h,\mu} \ & \frac{4}{th} \sum_{i=t_k}^{t_k-th} \sum_{br=1}^{10} e_{br}(u_{r,dof}^{h,i}, u_{r,dof}^{h,i-1}, \varphi_i)^\top M_{br,br}^h(\mu, p) e_{br}(u_{r,dof}^{h,i}, u_{r,dof}^{h,i-1}, \varphi_i) \\
\text{s.t.} \quad & D(p) \leq 0, \\
& T^d \leq \frac{1}{th} \sum_{i=t_k}^{t_k-th} M_{em}^h(u_{dof}^{h,i}, \theta_i), \\
& e^h(\mathfrak{y}^h, \mu, \bar{p}) = 0.
\end{aligned}$$

In the objective function $M_{br,br}^h(\mu, p)$ is the mass matrix, where the rows and columns of the nodes in rotor bar $br$ are chosen and we have

$$e_{br}(u_{r,dof}^{h,i}, u_{r,dof}^{h,i-1}, \varphi_i) = \mathbf{1}_{br} A_{br,:} \varphi_i - \frac{u_{\mathcal{N}_{br}}^{h,i-1} - u_{\mathcal{N}_{br}}^{h,i-1}}{\Delta t},$$

where $\mathbf{1}_{br}$ is a vector with row length of the number of nodes in rotor bar $br$ and $A_{br,:}$ is the $br$-th row of the reduced incidence matrix.

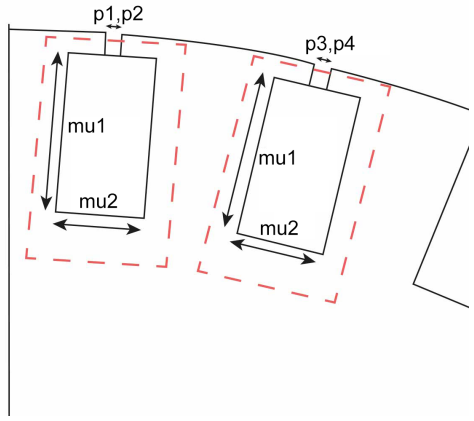


**Figure 7.2.** Design variable $\mu = (\text{mu1}, \text{mu2})$ and uncertainty $p = (\text{p1}, \text{p2}, \text{p3}, \text{p4}, \ldots, \text{p20})$.

### 7.1.1 Nominal optimization

In the nominal optimization, we solve the optimization problem without occuring uncertainties. As desired torque, we want to maintain the average torque in the initial configuration which is 246.12. The optimization is stopped when a first order optimality of $10^{-04}$ is attained. For the optimization we use fmincon from Matlab which uses a BFGS SQP method, see for example [81]. The results are shown in Table 7.2, where the first column is the iteration counter, the second is the value of the objective function, the third is the violation of the torque constraint, the fourth is the design and the last column contains the first order optimality measure. As we can see, we are able to reduce the initial Joule losses from 25.79 to 14.82 which is a reduction of $\approx 42\%$. The optimization took $1.6307 \cdot 10^{04}$s.

| iteration | f | max(c,0) | $\mu$ | opt |
|---|---|---|---|---|
| 0 | 25.790 | 0.00 | $(22.000, 12.000)$ | 1.79 |
| 1 | 25.052 | 0.00 | $(22.055, 12.415)$ | 1.63 |
| 2 | 18.823 | 0.00 | $(22.943, 16.648)$ | $8.97 \cdot 10^{-01}$ |
| 3 | 17.924 | $7.76 \cdot 10^{-01}$ | $(23.154, 17.392)$ | $7.99 \cdot 10^{-01}$ |
| 4 | 17.553 | 0.00 | $(23.298, 17.870)$ | $6.01 \cdot 10^{-01}$ |
| 5 | 17.306 | 0.00 | $(23.397, 18.151)$ | $4.93 \cdot 10^{-01}$ |
| 6 | 16.390 | $4.19 \cdot 10^{-01}$ | $(24.044, 18.996)$ | $4.21 \cdot 10^{-01}$ |
| 7 | 15.242 | 0.00 | $(27.119, 18.878)$ | $3.74 \cdot 10^{-01}$ |
| 8 | 15.074 | 0.00 | $(27.557, 18.874)$ | $2.34 \cdot 10^{-01}$ |
| 9 | 15.008 | 0.00 | $(27.752, 18.859)$ | $1.38 \cdot 10^{-01}$ |
| 10 | 15.027 | 0.00 | $(27.732, 18.840)$ | $1.02 \cdot 10^{-01}$ |
| 11 | 14.869 | 0.00 | $(27.941, 18.963)$ | $2.22 \cdot 10^{-02}$ |
| 12 | 14.864 | 0.00 | $(27.948, 18.968)$ | $2.01 \cdot 10^{-02}$ |
| 13 | 14.824 | 0.00 | $(28.000, 19.000)$ | $9.68 \cdot 10^{-04}$ |
| 14 | 14.824 | 0.00 | $(28.000, 19.000)$ | $2.01 \cdot 10^{-04}$ |
| 15 | 14.824 | 0.00 | $(28.000, 19.000)$ | $4.03 \cdot 10^{-06}$ |

**Table 7.2.** Nominal optimization results for 500 timesteps.

### 7.1.2 Nominal optimization with POD

We present here the results for the reduced order model. In Table 7.3 it is shown how many modes are needed to capture a given energy $\varepsilon$ of the snapshot space. Since we compute a basis for the stator without the airgap, the rotor without the airgap and without the rotor bars and a basis for each of the rotor bars, in the third column the maximal amount of modes of the bases for the 10 rotor bars is shown.

The amount of modes needed are the highest in the rotor part. This could be due to the rotation compared to the static stator and due to the fact that the rotor domain contains the most nodes.

In Table 7.4 we see how many modes are needed for the adjoint state for the objective function (index $f$) and the constraint function (index $c$) to capture a given energy $\varepsilon$. We see that we need more modes than for the state variable.

In Table 7.5 the running time of the state equation and the two adjoint equations on ROM level is compared to the running time on the FE level. Solving the state equation on FE level takes 149.86s, solving the adjoint for the objective 192.94s and solving the adjoint for the constraint 185.63s. Note that the computation times are for the basis which consists of the orthogonalized bases of the state, adjoint for objective and adjoint for constraint, which are all computed for the energy given in the first column. We see that even for energies close to 1, i.e. $1 - 10^{-12}$ we can save almost 50% of the computation time. In Figure 7.3 the error of the ROM solution of the state variable is given, computed in the discrete Bochner norm

$$\Delta t \sum_{i=1}^{500} \|u^{l,i} - u^{h,i}\|_{H^1(\Omega_s^h \cup \Omega_r^h)}.$$

The error does not decrease for the first three energies, since as we see in Table 7.3, the same amount of basis variables is needed to obtain a basis with the given energy. In Figure 7.4 the error of the adjoint variable is shown, computed in the discrete Bochner norm.

In the optimization with the proper orthogonal decomposition method, whenever the objective or the constraint function has to be evaluated for a new design, we solve the state equation on POD level to obtain $y^l$ and then use an error indicator, introduced in [43], which is based on the residual for the field/circuit equation. In the first iteration of the optimization we set the tolerance for the error to Inf, and after the first step, the tolerance is set to the maximum of the first order optimality measure and 0.02. If the error is too high, we compute new snapshots for the state equation and the two adjoint equations for the different parts of the domain and use the snapshots to compute new bases. We then orthogonalize them to one individual basis per subdomain. If the error is still to high, we check in which subdomain the error is the hightest, and increase the energy of the underlying basis and compute a new basis for the new energy. If the energy is then still not sufficiently low, we increase all the energies. We use the same energy for the state and adjoint.

In Table 7.6 we see that we obtain the same optimal design for the optimization with the reduced order model as without it. Since we start with an initial energy of $10^{-12}$, the ROM has not to be updated unil the 11th iteration. Using the ROM,

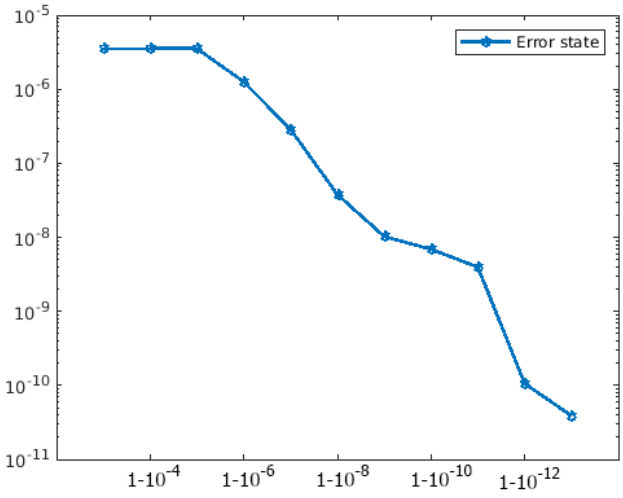


**Figure 7.3.** Error of the state variable in discrete Bochner norm for given energy $\varepsilon$.

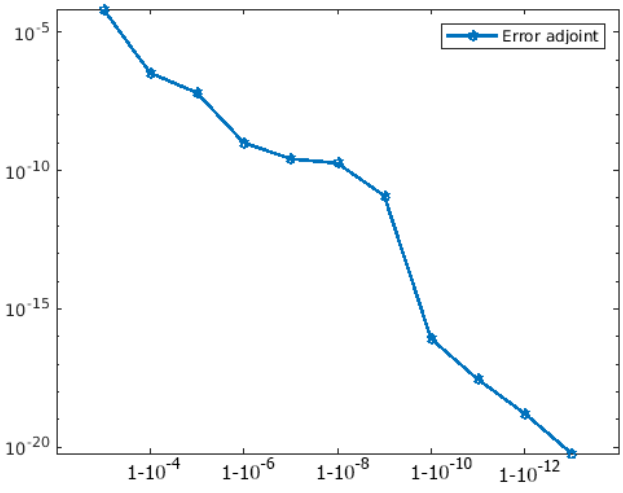


**Figure 7.4.** Error of the adjoint state in discrete Bochner norm for given energy $\varepsilon$.

the optimization took $8.3586 \cdot 10^{03}$s, which is a speedup of approximately factor 2 compared to the nominal optimization without the ROM.

| energy | number of modes | | | |
|---|---|---|---|---|
| $\varepsilon$ | $l_s$ | $l_{r-b}$ | $\max_{i=1,\dots,10} l_{b_i}$ | $\Sigma$ |
| $1-10^{-03}$ | 3 | 3 | 2 | 26 |
| $1-10^{-04}$ | 3 | 3 | 3 | 27 |
| $1-10^{-05}$ | 3 | 3 | 3 | 27 |
| $1-10^{-06}$ | 4 | 5 | 3 | 30 |
| $1-10^{-07}$ | 6 | 9 | 4 | 39 |
| $1-10^{-08}$ | 11 | 12 | 4 | 53 |
| $1-10^{-09}$ | 15 | 18 | 6 | 70 |
| $1-10^{-10}$ | 21 | 28 | 6 | 93 |
| $1-10^{-11}$ | 32 | 37 | 8 | 119 |
| $1-10^{-12}$ | 41 | 48 | 9 | 152 |
| $1-10^{-13}$ | 52 | 59 | 10 | 158 |
| $1-10^{-14}$ | 60 | 66 | 12 | 216 |

**Table 7.3.** Dimension of the reduced state for 500 timesteps and given energy $\varepsilon$.

| energy | number of modes | | | | | | |
|---|---|---|---|---|---|---|---|
| $\varepsilon$ | $l^\lambda_{s,f}$ | $l^\lambda_{r-b,f}$ | $l^\lambda_{b_i,f}$ | $l^\lambda_{s,c}$ | $l^\lambda_{r-b,c}$ | $l^\lambda_{b_i,c}$ | $\Sigma$ |
| $1-10^{-03}$ | 7 | 9 | 4 | 6 | 4 | 4 | 89 |
| $1-10^{-04}$ | 11 | 13 | 4 | 9 | 12 | 4 | 125 |
| $1-10^{-05}$ | 16 | 20 | 5 | 15 | 15 | 5 | 156 |
| $1-10^{-06}$ | 20 | 29 | 7 | 20 | 22 | 6 | 206 |
| $1-10^{-07}$ | 29 | 39 | 7 | 29 | 29 | 7 | 248 |
| $1-10^{-08}$ | 38 | 49 | 8 | 34 | 40 | 8 | 297 |
| $1-10^{-09}$ | 50 | 65 | 10 | 52 | 50 | 9 | 375 |
| $1-10^{-10}$ | 63 | 75 | 11 | 62 | 63 | 11 | 449 |
| $1-10^{-11}$ | 71 | 89 | 13 | 68 | 79 | 12 | 529 |
| $1-10^{-12}$ | 77 | 110 | 15 | 74 | 93 | 14 | 609 |
| $1-10^{-13}$ | 84 | 125 | 17 | 79 | 106 | 16 | 690 |
| $1-10^{-14}$ | 89 | 140 | 20 | 85 | 119 | 19 | 769 |

**Table 7.4.** Dimension of the reduced adjoint state for 500 timesteps for the objective $f$ and constraint $c$ and given energy $\varepsilon$.

### 7.1.3 Robust optimization: Linear model

We perform the robust optimization with the linear Taylor expansion by using Granso (see [29]) as a solver for non-convex and non-smooth optimization problems. The most time consuming part in the optimization is computing the full sensitivity. We therefore use a factorization of the system matrix $C^i_{\mathfrak{y}_i}$ (see (4.36)) in every timestep to efficiently solve the system for the 20 right hand sides, which

| energy | running time in [s] | | | | | |
|---|---|---|---|---|---|---|
| $\varepsilon$ | state | adjoint f | adjoint c | ratio state | ratio adj f | ratio adj c |
| $1-10^{-03}$ | 14.70 | 36.49 | 35.75 | 0.0981 | 0.1891 | 0.1926 |
| $1-10^{-04}$ | 16.19 | 38.72 | 38.14 | 0.1080 | 0.2007 | 0.2055 |
| $1-10^{-05}$ | 17.91 | 40.75 | 40.12 | 0.1195 | 0.2112 | 0.2161 |
| $1-10^{-06}$ | 21.35 | 45.40 | 44.68 | 0.1425 | 0.2353 | 0.2407 |
| $1-10^{-07}$ | 25.18 | 51.30 | 50.49 | 0.1680 | 0.2659 | 0.2720 |
| $1-10^{-08}$ | 30.18 | 57.51 | 56.06 | 0.2014 | 0.2981 | 0.3020 |
| $1-10^{-09}$ | 38.29 | 68.08 | 67.09 | 0.2555 | 0.3529 | 0.3614 |
| $1-10^{-10}$ | 51.08 | 88.38 | 87.23 | 0.3409 | 0.4581 | 0.4699 |
| $1-10^{-11}$ | 61.01 | 97.68 | 96.41 | 0.4071 | 0.5063 | 0.5194 |
| $1-10^{-12}$ | 79.54 | 124.67 | 120.67 | 0.5308 | 0.6462 | 0.6501 |
| $1-10^{-13}$ | 124.46 | 171.72 | 171.11 | 0.8301 | 0.8900 | 0.9218 |
| $1-10^{-14}$ | 152.58 | 207.04 | 211.18 | 1.0176 | 1.0730 | 1.1376 |

**Table 7.5.** Computation times for the reduced state and adjoint for 500 timesteps and given energy $\varepsilon$ compared to the solution on the FE level.

| iteration | f | c | $\mu$ | opt |
|---|---|---|---|---|
| 0 | 25.793 | $3.19 \cdot 10^{-05}$ | $(22.000, 12.000)$ | 1.79 |
| 1 | 25.053 | 0.00 | $(22.056, 12.416)$ | 1.63 |
| 2 | 18.769 | 0.00 | $(22.939, 16.678)$ | $9.02 \cdot 10^{-01}$ |
| 3 | 17.899 | $7.46 \cdot 10^{-01}$ | $(23.140, 17.412)$ | $8.02 \cdot 10^{-01}$ |
| 4 | 17.544 | 0.00 | $(23.267, 17.876)$ | $6.01 \cdot 10^{-01}$ |
| 5 | 17.300 | 0.00 | $(23.363, 18.156)$ | $4.83 \cdot 10^{-01}$ |
| 6 | 16.401 | $4.35 \cdot 10^{-01}$ | $(23.993, 18.996)$ | $4.16 \cdot 10^{-01}$ |
| 7 | 15.309 | 0.00 | $(27.008, 18.877)$ | $3.56 \cdot 10^{-01}$ |
| 8 | 15.133 | 0.00 | $(27.502, 18.872)$ | $3.36 \cdot 10^{-01}$ |
| 9 | 15.068 | 0.00 | $(27.712, 18.854)$ | $2.39 \cdot 10^{-01}$ |
| 10 | 15.079 | 0.00 | $(27.712, 18.837)$ | $1.04 \cdot 10^{-01}$ |
| 11 | 14.931 | 0.00 | $(27.927, 18.960)$ | $2.40 \cdot 10^{-02}$ |
| | | Calculate new Snapshots | | |
| 12 | 14.868 | 0.00 | $(27.940, 18.967)$ | $5.32 \cdot 10^{-02}$ |
| 13 | 14.864 | 0.00 | $(27.949, 18.968)$ | $2.00 \cdot 10^{-02}$ |
| 14 | 14.832 | 0.00 | $(27.990, 18.994)$ | $4.32 \cdot 10^{-03}$ |
| | | Calculate new Snapshots | | |
| 15 | 14.824 | 0.00 | $(28.000, 19.000)$ | $2.65 \cdot 10^{-04}$ |
| | | Calculate new Snapshots | | |
| 16 | 14.824 | 0.00 | $(28.000, 19.000)$ | $4.38 \cdot 10^{-05}$ |

**Table 7.6.** Nominal optimization with ROM for 500 timesteps for linear material.

contain the derivatives of the state equation with respect to the uncertain parameters. The result of the optimization is shown in Table 7.7. Note that in the second column, the approximated worst-case value of the objective function is stated, since we minimize the worst-case. Also the third column states the approximated constraint violation in the approximated worst-case. We use the torque of the initial design in the nominal uncertain parameter as desired torque. We can see in the 0th iteration in Table 7.7, that the initial design is not a feasible design, since the constraint function is not fulfilled in the worst-case. Therefore the objective value cannot be lowered as much as in the nominal optimization. In the optimal robust design (14.63, 19.00) the Joule losses in the nominal parameter are 24.02 and the torque constraint is at $-0.44$. The optimization took $3.6178 \cdot 10^{05}$s. The true objective and constraint values in the approximated worst-cases in the optimal design are at 24.2516 and 246.3038, which shows that the approximated worst-case values are very accurate.

### 7.1.4 Robust optimization: Linear model with POD

Using the linear Taylor expansion in the robust optimization involves the sensitivity of the state variable with respect to the uncertainty $y_p(\mu, \bar{p})$, therefore in the optimization with the ROM, we compute bases for the sensitivity and add them to the orthogonalization procedure (together with the bases for state and adjoint state). We keep the modes for the sensitivity fixed during the optimization: 20 modes for each of the rotor bars and stator and 40 modes for the rotor part. On the FE level the computation of the sensitivity takes 1582.03s. In Table 7.8 results for the reduced order model involving the sensitivity are shown: in the second column the number of basis variables is shown, where the basis consists of the orthogonalized bases for the state variable, the two adjoint variables and the sensitivity. In the third column, the computation time for the sensitivity using the ROM is stated and we can see that computing the sensitivity on ROM level is even for high energies as $1 - 10^{-12}$ cheaper than on the FE level by a factor 3. In the fourth column the ratio between the computation time for the sensitivity using the ROM and FE is shown and in the last column the error in discrete Bochner norm of the sensitivity is stated.

We have performed the same optimization with the linear Taylor expansion using the reduced order model. The results are shown in Table 7.9. We started with an initial energy of $1 - 10^{-12}$ for the ROM. In the optimization we had to calculate new bases 2 times. After 52 iterations, the optimization terminated with almost the same design as in the optimization without the reduced order model, since the step

| iteration | $f^{wc}$ | $\max(0, c^{wc})$ | $\mu$ | opt |
|---|---|---|---|---|
| 0 | 26.032 | $2.04 \cdot 10^{-01}$ | $(22.000, 12.000)$ | 1.00 |
| 2 | 25.165 | 0.00 | $(22.189, 12.428)$ | $1.94 \cdot 10^{-01}$ |
| 4 | 24.881 | 0.00 | $(22.238, 12.574)$ | $9.16 \cdot 10^{-02}$ |
| 6 | 24.814 | 0.00 | $(22.241, 12.613)$ | $4.78 \cdot 10^{-02}$ |
| 8 | 24.784 | 0.00 | $(22.236, 12.633)$ | $2.96 \cdot 10^{-02}$ |
| 10 | 24.767 | 0.00 | $(22.229, 12.647)$ | $2.17 \cdot 10^{-02}$ |
| 12 | 24.755 | 0.00 | $(22.221, 12.657)$ | $1.65 \cdot 10^{-02}$ |
| 14 | 24.748 | 0.00 | $(22.212, 12.666)$ | $1.33 \cdot 10^{-02}$ |
| 16 | 24.742 | 0.00 | $(22.202, 12.675)$ | $1.14 \cdot 10^{-02}$ |
| 18 | 24.738 | 0.00 | $(22.191, 12.682)$ | $1.04 \cdot 10^{-02}$ |
| 20 | 24.735 | 0.00 | $(22.180, 12.690)$ | $1.01 \cdot 10^{-02}$ |
| 22 | 24.732 | 0.00 | $(22.167, 12.698)$ | $1.03 \cdot 10^{-02}$ |
| 24 | 24.730 | 0.00 | $(22.153, 12.706)$ | $1.11 \cdot 10^{-02}$ |
| 26 | 24.727 | 0.00 | $(22.135, 12.717)$ | $1.31 \cdot 10^{-02}$ |
| 28 | 24.725 | 0.00 | $(22.108, 12.732)$ | $1.82 \cdot 10^{-02}$ |
| 30 | 24.720 | 0.00 | $(22.058, 12.760)$ | $4.00 \cdot 10^{-02}$ |
| 32 | 24.704 | 0.00 | $(21.458, 13.083)$ | $4.33 \cdot 10^{-03}$ |
| 34 | 24.686 | 0.00 | $(20.763, 13.481)$ | $4.24 \cdot 10^{-01}$ |
| 36 | 24.630 | 0.00 | $(19.976, 13.991)$ | $4.55 \cdot 10^{-02}$ |
| 38 | 24.596 | 0.00 | $(19.365, 14.410)$ | $1.75 \cdot 10^{-01}$ |
| 40 | 24.556 | 0.00 | $(18.627, 14.954)$ | $2.25 \cdot 10^{-01}$ |
| 42 | 24.511 | 0.00 | $(17.412, 15.939)$ | $4.72 \cdot 10^{-01}$ |
| 44 | 24.464 | $5.01 \cdot 10^{-03}$ | $(17.571, 15.845)$ | $2.14 \cdot 10^{-01}$ |
| 46 | 24.403 | 0.00 | $(16.562, 16.784)$ | $6.80 \cdot 10^{-02}$ |
| 48 | 24.364 | $7.98 \cdot 10^{-04}$ | $(16.013, 17.353)$ | $2.97 \cdot 10^{-01}$ |
| 50 | 24.328 | 0.00 | $(15.148, 18.313)$ | $2.57 \cdot 10^{-01}$ |
| 52 | 24.246 | 0.00 | $(14.635, 19.000)$ | $2.33 \cdot 10^{-03}$ |
| 53 | 24.247 | $1.01 \cdot 10^{-05}$ | $(14.632, 19.000)$ | $1.43 \cdot 10^{-05}$ |

**Table 7.7.** Robust optimization with linear Taylor approximation for 500 timesteps for linear material.

tolerance was met. The optimization took $1.0880 \cdot 10^{05}$s, which is an approximate speedup of factor 3 in comparison to the optimization without the reduced order model.

### 7.1.5 Robust optimization: Quadratic model

In this section the results using the quadratic Taylor model are presented. We again used Granso for the optimization. In Table 7.10 we see, that in comparison to the

| energy | Results for ROM involving the sensitivity | | | |
|---|---|---|---|---|
| $\varepsilon$ | basis variables | running time [s] | ratio sens | $\|y_p^l - y_p\|$ |
| $1-10^{-03}$ | 375 | 86.77 | 0.0548 | $1.23 \cdot 10^{-06}$ |
| $1-10^{-04}$ | 412 | 98.09 | 0.0620 | $2.92 \cdot 10^{-07}$ |
| $1-10^{-05}$ | 443 | 107.06 | 0.0677 | $2.82 \cdot 10^{-07}$ |
| $1-10^{-06}$ | 496 | 134.01 | 0.0847 | $2.56 \cdot 10^{-07}$ |
| $1-10^{-07}$ | 547 | 146.95 | 0.0929 | $2.06 \cdot 10^{-07}$ |
| $1-10^{-08}$ | 597 | 174.41 | 0.1102 | $1.42 \cdot 10^{-07}$ |
| $1-10^{-09}$ | 674 | 209.81 | 0.1326 | $2.79 \cdot 10^{-08}$ |
| $1-10^{-10}$ | 744 | 243.27 | 0.1538 | $3.70 \cdot 10^{-09}$ |
| $1-10^{-11}$ | 831 | 364.33 | 0.2303 | $7.64 \cdot 10^{-10}$ |
| $1-10^{-12}$ | 934 | 502.61 | 0.3177 | $6.50 \cdot 10^{-10}$ |

**Table 7.8.** Number of modes for basis consisting of orthognonalized basis of state, adjoint and sensitivity for 500 timesteps and given energy $\varepsilon$, solution time of sensitivity equation with given basis and the maximal error in the discrete Bochner norm over the sensitivities with respect to the uncertain parameters $\max_{i=1,\dots,20} \|y_{p_i}^l(\mu,\bar{p}) - y_{p_i}(\mu,\bar{p})\|$ is shown.

linear model, the approximated worst-case of the constraint function in the initial design is higher. The optimization took less iterations than with the linear model and terminated in a different design. In the optimal robust design $(21.425, 12.894)$ the Joule losses in the nominal parameter are 24.02 and the torque constraint is at $-0.44$, leading to a higher torque, i.e. 246.56, than the desired torque. The optimization took $2.8135 \cdot 10^{05}$s. Evaluating the objective and constraint function in the approximated worst-case in the optimal design returns Joule losses of 25.0361 and a torque of 246.3150. This shows that the approximated worst-case values are accurate. The robust optimal design might differ from the robust optimal design found with the linear Taylor model, since the model might have overestimated a worst-case of the constraint function during the optimization, which might have shrinked down the feasible region or, a different local maximizer was found and tracked.

We have performed the same optimization with the quadratic Taylor expansion using the reduced order model. The results are shown in Table 7.11. We started with an initial energy of $1 - 10^{-12}$ for the ROM. In the optimization we had to calculate a new basis only 1 time. After 14 iterations, the optimization terminated with almost the same design as in the optimization without the reduced order model, since the step tolerance was met. The optimization took $8.8338 \cdot 10^{04}$s, which is an approximate speedup of factor 3 in comparison to the optimization without the reduced order model.

| iteration | $f^{wc}$ | $\max(0, c^{wc})$ | $\mu$ | opt |
|---|---|---|---|---|
| 0 | 26.032 | $2.04 \cdot 10^{-01}$ | $(22.000, 12.000)$ | 1.00 |
| 2 | 25.166 | 0.00 | $(22.189, 12.428)$ | $1.94 \cdot 10^{-01}$ |
| 4 | 24.882 | 0.00 | $(22.238, 12.574)$ | $9.16 \cdot 10^{-02}$ |
| 6 | 24.814 | 0.00 | $(22.241, 12.613)$ | $4.78 \cdot 10^{02}$ |
| 8 | 24.784 | 0.00 | $(22.236, 12.633)$ | $3.06 \cdot 10^{-02}$ |
| 10 | 24.767 | 0.00 | $(22.229, 12.646)$ | $2.17 \cdot 10^{-02}$ |
| 12 | 24.756 | 0.00 | $(22.221, 12.657)$ | $1.65 \cdot 10^{-02}$ |
| 14 | 24.748 | 0.00 | $(22.212, 12.666)$ | $1.33 \cdot 10^{-02}$ |
| 16 | 24.742 | 0.00 | $(22.202, 12.674)$ | $1.14 \cdot 10^{-02}$ |
| 18 | 24.738 | 0.00 | $(22.191, 12.682)$ | $1.04 \cdot 10^{-02}$ |
| 20 | 24.735 | 0.00 | $(22.180, 12.690)$ | $9.99 \cdot 10^{-03}$ |
| 22 | 24.732 | 0.00 | $(22.168, 12.697)$ | $1.02 \cdot 10^{-02}$ |
| 24 | 24.730 | 0.00 | $(22.153, 12.706)$ | $1.10 \cdot 10^{-02}$ |
| 26 | 24.728 | 0.00 | $(22.136, 12.716)$ | $1.29 \cdot 10^{-02}$ |
| 28 | 24.725 | 0.00 | $(22.110, 12.730)$ | $1.77 \cdot 10^{-02}$ |
| 30 | 24.721 | 0.00 | $(22.065, 12.756)$ | $3.54 \cdot 10^{-02}$ |
| 32 | 24.705 | 0.00 | $(21.461, 13.080)$ | $1.32 \cdot 10^{-03}$ |
| | | Calculate new Snapshots | | |
| 34 | 24.699 | 0.00 | $(21.136, 13.262)$ | $1.82 \cdot 10^{-01}$ |
| 36 | 24.596 | $4.19 \cdot 10^{-03}$ | $(19.861, 14.087)$ | $2.75 \cdot 10^{-01}$ |
| 38 | 24.583 | 0.00 | $(19.104, 14.596)$ | $2.53 \cdot 10^{-01}$ |
| 40 | 24.522 | 0.00 | $(18.201, 15.297)$ | $8.74 \cdot 10^{-03}$ |
| 42 | 24.485 | 0.00 | $(17.071, 16.248)$ | $4.80 \cdot 10^{-01}$ |
| 44 | 24.443 | $4.98 \cdot 10^{-03}$ | $(17.285, 16.099)$ | $2.09 \cdot 10^{-01}$ |
| 46 | 24.423 | 0.00 | $(16.722, 16.613)$ | $5.96 \cdot 10^{-02}$ |
| 48 | 24.396 | 0.00 | $(15.928, 17.401)$ | $5.14 \cdot 10^{-01}$ |
| 50 | 24.356 | 0.00 | $(15.749, 17.621)$ | $9.72 \cdot 10^{-02}$ |
| | | Calculate new Snapshots | | |
| 52 | 24.240 | $4.98 \cdot 10^{-03}$ | $(14.639, 19.000)$ | $6.85 \cdot 10^{-03}$ |

**Table 7.9.** Optimization using ROM and linear Taylor model for 500 timesteps for linear material.

### 7.1.6 Robust optimization: Quadratic model shift expansion point

In this section the results for the robust optimization with the quadratic Taylor expansion with the shifted expansion point are presented. The shifting was performed if the approximated worst-case value had a relative difference of 0.03 to the true function value at the predicted worst-case. In Table 7.12 we can see, that we never had to change the expansion point for the objective function $f$, but had to shift the

| iteration | $f^{wc}$ | $\max(0, c^{wc})$ | $\mu$ | opt |
|---|---|---|---|---|
| 0 | 26.033 | $3.87 \cdot 10^{-01}$ | $(22.000, 12.000)$ | 1.01 |
| 2 | 24.769 | $1.83 \cdot 10^{-01}$ | $(22.428, 12.912)$ | $1.17 \cdot 10^{-01}$ |
| 4 | 25.030 | $8.30 \cdot 10^{-03}$ | $(22.217, 12.522)$ | $5.36 \cdot 10^{-03}$ |
| 6 | 25.048 | 0.00 | $(22.193, 12.498)$ | $3.13 \cdot 10^{-03}$ |
| 8 | 25.043 | 0.00 | $(22.004, 12.593)$ | $1.35 \cdot 10^{-03}$ |
| 10 | 25.033 | $9.27 \cdot 10^{-04}$ | $(21.651, 12.774)$ | $7.11 \cdot 10^{-02}$ |
| 12 | 25.034 | $5.71 \cdot 10^{-04}$ | $(21.442, 12.887)$ | $6.79 \cdot 10^{-03}$ |
| 14 | 25.036 | $2.94 \cdot 10^{-05}$ | $(21.380, 12.918)$ | $1.71 \cdot 10^{-02}$ |
| 16 | 25.036 | 0.00 | $(21.429, 12.892)$ | $1.76 \cdot 10^{-04}$ |
| 18 | 25.036 | $3.37 \cdot 10^{-09}$ | $(21.426, 12.894)$ | $1.23 \cdot 10^{-04}$ |
| 20 | 25.036 | 0.00 | $(21.425, 12.894)$ | $4.68 \cdot 10^{-03}$ |
| 21 | 25.036 | $8.40 \cdot 10^{-10}$ | $(21.425, 12.894)$ | $3.77 \cdot 10^{-07}$ |

**Table 7.10.** Robust optimization using the quadratic Taylor model for 500 timesteps for linear material.

| iteration | $f^{wc}$ | $\max(0, c^{wc})$ | $\mu$ | opt |
|---|---|---|---|---|
| 0 | 26.033 | $3.77 \cdot 10^{-01}$ | $(22.000, 12.000)$ | 1.01 |
| 1 | 24.122 | $8.50 \cdot 10^{-01}$ | $(22.428, 12.912)$ | $3.73 \cdot 10^{-01}$ |
| 2 | 24.769 | $1.84 \cdot 10^{-01}$ | $(22.275, 12.624)$ | $1.17 \cdot 10^{-01}$ |
| 3 | 24.989 | $3.05 \cdot 10^{-02}$ | $(22.217, 12.522)$ | $2.99 \cdot 10^{-02}$ |
| 4 | 25.031 | $8.35 \cdot 10^{-03}$ | $(22.202, 12.504)$ | $5.37 \cdot 10^{-03}$ |
| 5 | 25.048 | $2.56 \cdot 10^{-04}$ | $(22.193, 12.498)$ | $2.91 \cdot 10^{-03}$ |
| 6 | 25.048 | 0.00 | $(22.179, 12.505)$ | $3.01 \cdot 10^{-03}$ |
| | | Calculate new Snapshots | | |
| 7 | 25.045 | 0.00 | $(21.719, 12.737)$ | $3.13 \cdot 10^{-01}$ |
| 8 | 25.043 | 0.00 | $(21.851, 12.671)$ | $6.48 \cdot 10^{-03}$ |
| 9 | 25.042 | $2.42 \cdot 10^{-06}$ | $(21.914, 12.640)$ | $5.04 \cdot 10^{-02}$ |
| 10 | 25.040 | 0.00 | $(21.775, 12.711)$ | $3.38 \cdot 10^{-02}$ |
| 11 | 25.038 | 0.00 | $(21.547, 12.829)$ | $6.10 \cdot 10^{-02}$ |
| 12 | 25.036 | $3.50 \cdot 10^{-04}$ | $(21.591, 12.808)$ | $1.05 \cdot 10^{-01}$ |
| 14 | 25.036 | $7.79 \cdot 10^{-05}$ | $(21.424, 12.895)$ | $9.90 \cdot 10^{-03}$ |

**Table 7.11.** Robust optimization using the quadratic Taylor model and ROM for 500 timesteps for linear material.

expansion point of the constraint function almost every second iteration. Note that since the constraint function has values close to 0, small deviations lead to large relative differences. To find a better expansion point, we used fmincon of Matlab to find a local maximum of the constraint function and had the optimality tolerance set to $10^{-03}$, as the starting point we used the previous expansion point (and in the first iteration the nominal parameter $\bar{p}$). The optimality tolerance for the outer

optimization problem was set to $10^{-04}$. As we can see, we were able to lower the worst-case more than in the optimization with the quadratic and linear model. In the optimal design, the objective function evaluated in the approximated worst-case is at 23.886, which is a relative deviation less than the required 3% from the approximated worst-case value of 24.056. Since the expansion point was shifted before the last iteration, the approximated worst-case value of the constraint function matches its true worst-case value.

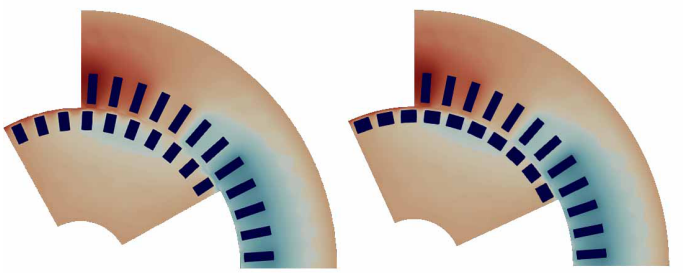

**Figure 7.5.** Comparison of the initial design (left) and the robust optimal design (right) obtained with the quadratic Taylor model with the shifted expansion point.

### 7.1.7 Robust optimization: Quadratic interpolation model

The results of the optimization using the quadratic interpolation model are shown in Table 7.13. We have limited the number of improving iterations to 3. We can see, that the designs during the optimization are similiar to the ones obtained with the quadratic Taylor model with the shifted expansion point. In the optimal design, the approximated worst-case values are equal to the true objective and constraint function values in the approximated worst-cases. Overall, the optimal design is similiar to the one obtained from the robust optimization with the linear Taylor model, and the one obtained with the quadratic Taylor model using the shifted expansion point.

## 7.2 Nonlinear material

In this section, we present numerical results for the simulation and optimization using nonlinear material. In the optimization, we use almost the same objective

function as for the linear material

$$\begin{aligned}
\min_{\mathfrak{y},\mu} \quad & \frac{4}{T-\mathrm{th}} \int_{\mathrm{th}}^{T} s(t) \sum_{i=1}^{10} \int_{\Omega_{\mathrm{bar},i}(p)} \sigma \left( \mathbf{1}_{\Omega_{\mathrm{bar},i}(p)}(x) \mathbf{A}_{i,\cdot}^{\top} \varphi(t) - \frac{\mathrm{d}a(x,t)}{\mathrm{d}t} \right)^2 \mathrm{d}x \, \mathrm{d}t \\
\text{s.t.} \quad & D(p) \leq 0, \\
& T^d \leq \frac{1}{T-\mathrm{th}} \int_{\mathrm{th}}^{T} s(t) \ell_z \int_{\Gamma} \nu_1(x) \partial_n u_1(x,t) \partial_\theta u_2(x_{r_{-\theta(t)}}, t) \, \mathrm{d}S \, \mathrm{d}t, \\
& e(y, \mu, \bar{p}) = 0.
\end{aligned}$$

The difference is, that we use a smooth weighting function $s(t)$, which puts more weight to the values being close to the center of the time horizon and less weight to the values being close to the boundary of the time horizon. This helps to lower the influence of differing periods. We treat the value of the conductivity $\sigma$ as uncertain, which might be due to imperfect material or due to different temperatures of the rotor bars. We choose $\mathcal{U} = \{\sigma : |57500000 - \sigma| \leq 14500000\}$. To model the reluctivity for the material I200, we use a smoothed BH-curve (see Figure 7.6). The smoothed BH-curve is three times continuously differentiable and it approaches the reluctivity of vacuum for $|B| \to \infty$, such that the behavior of saturated material is approximated well. Again we choose the steady state as an initial value. In the optimization we perform 200 timesteps and use 150 for averaging. As the desired torque in the constraint we choose the (weighted) averaged torque in the initial design, which is 121.98.

### 7.2.1 Nonlinear material: Nominal Optimization

The results for the nominal optimization with nonlinear material are shown in Table 7.14. We have used again fmincon from Matlab and stopped the optimization when the first order optimality condition was lower than $10^{-04}$. The initial point is already close to a local optimum, such that we are able to lower the weighted Joule losses by around 14.7%.

### 7.2.2 Nonlinear material: Linear model

For the robust optimization we again use Granso as a solver for non-convex and non-smooth optimization problems. The results of the robust optimization using

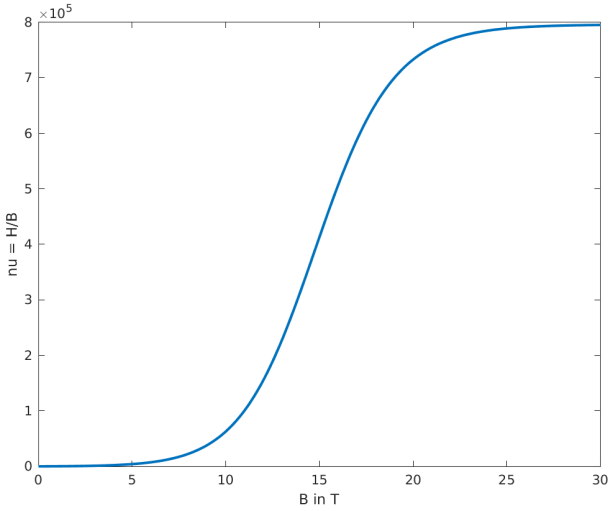


**Figure 7.6.** Smoothed BH-curve

the linear Taylor model and nonlinear material is shown in Table 7.7. In the second column the approximated worst-case value of the objective function is stated. The third column contains the approximated violation of the constraint function in the approximated worst-case. As we use the torque of the initial design in the nominal uncertain parameter as desired torque, we can see that the initial design is not feasible and therefore a feasible point needs to be found until the objective can be lowered, which results in only a small decrease of the Joule losses. Note that in the nominal parameter, the Joule losses in the optimal design $(21.475, 12.871)$ are at 11.2029 which is lower than the Joule losses in the nominal parameter in the initial design and a decrease in the nominal parameter (when no uncertainties occur) is given.

### 7.2.3 Nonlinear material: Quadratic Taylor model

The results for the optimization with the quadratic Taylor model are shown in Table 7.16. Here, the same problem as with the linear model occurs. The initial design is infeasible and therefore the Joule losses can only be lowered by a small amount. In the optimal design the Joule losses in the nominal parameter are at 10.8392, which is lower compared to the optimization with the linear model.

### 7.2.4 Nonlinear material: Quadratic Taylor model with shifted expansion point

The results for the robust optimization using the Quadratic Taylor model with shifted expansion point for the nonlinear material are shown in Table 7.17. The tolerance for shifting the expansion point was set to a relative difference of 0.03 between the approximated worst-case and the true underlying function/constraint value. We have set the initial worst-case disturbance of the objective function to the left boundary of the uncertainty set, since the worst-case is attained there. Therefore we never had to change the expansion point during the optimization. As we can see, we had to shift the expansion point for the constraint function almost every iteration, until the change in the design became small. The optimal design found differs from the optimal design using the linear and quadratic Taylor model. We were able to lower the objective function from 14.410 to 12.449, which is a reduction of $\approx 13\%$. Note that these are not the Joule losses, but the averaged weighted Joule losses. The optimal design is closer to the optimal design of the nominal optimization compared to the designs obtained with the robust optimization using the linear and quadratic Taylor model.

In the following we present here the numerical results for using DEIM for the simulation of the machine. In Table 7.18 it is shown, how many basis variables are needed to approximate the space spanned by the nonlinear snapshots for given energy $\varepsilon$. We have solved the state equation (on FEM level) with the DEIM approximation and computed the error of the magnetic vector potential in the discrete Bochner norm, which is shown in the last column. We can see, that for the energy $1 - 10^{-03}$, the error is already below $10^{-06}$. Only in the simulation with the energy of $1 - 10^{-03}$ we achieved a minor speedup in the solution of the state equation. This is due to Newton's method needing more iterations to converge and if a certain number of basis variables is exceeded, the evaluation of the nonlinearity with DEIM is equally expensive as evaluating the full nonlinearity. In Table 7.19 the results are shown for keeping the dimension of the DEIM fixed (2 basis vectors for the rotor bars, 3 for the stator and 4 for the rotor) and then use a POD basis for the magnetic vector potential, computed for the energy given in the first column, to solve the state equation. The error is again computed in the discrete Bochner norm. For low energies, the speedup in the simulation was up to 50% (from 13882.31s on FE level to 6850.23s for $\varepsilon = 1 - 10^{-03}$), but in some instances, Newton's method had trouble converging, such that the solution on the ROM took longer than on the FE level, this also occured in the optimization, such that we did not use the reduced order model in the optimization with the nonlinear material. Figure 7.7 shows the first four POD eigenmodes of the magnetic vector potential in the stator.

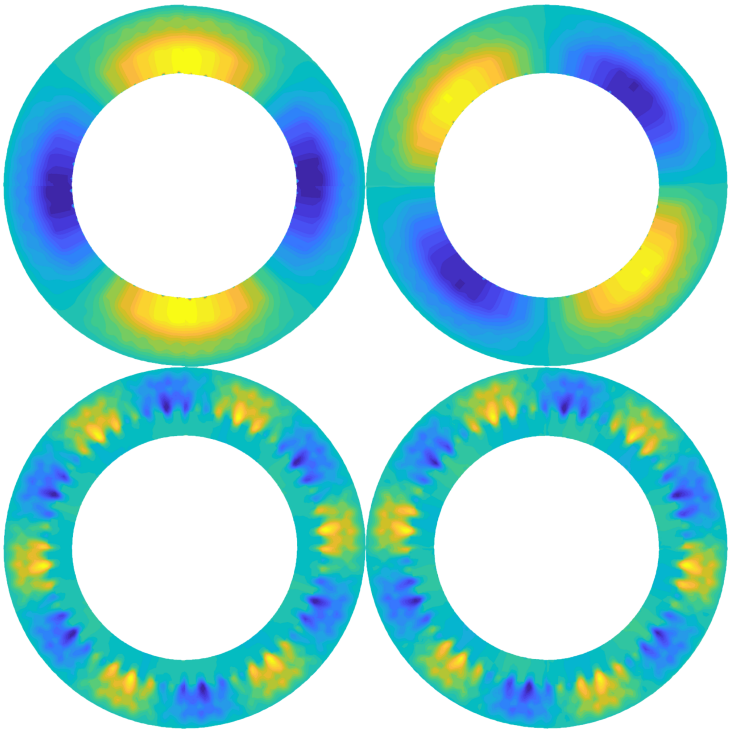

**Figure 7.7.** First four eigenmodes of the magnetic vector potential in the stator domain for nonlinear material.

## 7.3 Conclusion of the numerical results

Dealing with highly nonlinear functions in the robust opimization as we do in our case, we can see that the non-adaptive linear and quadratic Taylor models are not well suited, since they cannot capture the behavior of the underlying functions good enough. They can over- or under-estimate the worst-case which leads to non-optimal or non-robust solutions. If it is known, that the underlying functions are linear or quadratic, then these models are exact and using these models in combination with reduced order models one can compute the optimal robust solutions in reasonable time. We found that using reduced order models speeds up the optimzation by factor 2-3. One can benefit the most out of an implementation, if one wants to perform the optimization multiple times, e.g. for varying input data.

In our application, using the adaptive second order models leads to the best results, meaning the lowest worst-case of the objective function. This is the case, since these adaptive models iteratively improve the approximation of the worst-case

value, which eliminates over- and under-estimation.

To use the quadratic Taylor model with the moving expansion point, derivatives up to third order have to be implemented, which can be cumbersome. In contrast, to perform the optimization with the quadratic interpolation model, we only have to implement the derivative of objective and constraint functions with respect to the design to perform the outer minimization. This design derivative computation can also be bypassed by using an interpolation model for the outer optimization aswell, which makes the robust optimization applicable for blackbox simulations. The drawback of the interpolation model is that the number of state equation solvings to setup the initial interpolation model depends on the degrees of freedom of the uncertainty. To compute the initial model one can use parallel computation. The optimization with the derivative-free interpolation model would therefore be a good fit for industry companies with high computing power.

| iteration | $f^{wc}$ | $\max(0, c^{wc})$ | $\mu$ | opt |
|---|---|---|---|---|
| | Shifting expansion point constraint | | | |
| 0 | 26.033 | $2.87 \cdot 10^{-01}$ | $(22.000, 12.000)$ | 1.01 |
| | Shifting expansion point constraint | | | |
| 1 | 24.125 | $6.56 \cdot 10^{-01}$ | $(22.427, 12.910)$ | $2.67 \cdot 10^{-01}$ |
| | Shifting expansion point constraint | | | |
| 2 | 25.147 | 0.00 | $(22.183, 12.443)$ | $2.26 \cdot 10^{-01}$ |
| | Shifting expansion point constraint | | | |
| 3 | 24.370 | $3.66 \cdot 10^{-01}$ | $(22.350, 12.814)$ | $1.63 \cdot 10^{-01}$ |
| 4 | 24.998 | 0.00 | $(22.195, 12.527)$ | $1.77 \cdot 10^{-01}$ |
| 5 | 24.735 | $6.70 \cdot 10^{-03}$ | $(22.251, 12.655)$ | $1.10 \cdot 10^{-02}$ |
| | Shifting expansion point constraint | | | |
| 6 | 24.746 | 0.00 | $(22.235, 12.657)$ | $1.06 \cdot 10^{-02}$ |
| 7 | 24.734 | $3.20 \cdot 10^{-03}$ | $(22.192, 12.685)$ | $3.61 \cdot 10^{-02}$ |
| | Shifting expansion point constraint | | | |
| 8 | 24.704 | 0.00 | $(21.725, 12.943)$ | $7.64 \cdot 10^{-02}$ |
| 9 | 24.684 | 0.00 | $(21.021, 13.337)$ | $9.95 \cdot 10^{-02}$ |
| | Shifting expansion point constraint | | | |
| 10 | 24.644 | $9.82 \cdot 10^{-03}$ | $(21.334, 13.189)$ | $2.88 \cdot 10^{-01}$ |
| 11 | 24.621 | 0.00 | $(20.842, 13.479)$ | $2.70 \cdot 10^{-01}$ |
| 12 | 24.600 | 0.00 | $(19.743, 14.161)$ | $8.25 \cdot 10^{-01}$ |
| | Shifting expansion point constraint | | | |
| 13 | 24.576 | 0.00 | $(20.465, 13.729)$ | $3.99 \cdot 10^{-01}$ |
| 14 | 24.523 | 0.00 | $(19.776, 14.192)$ | $2.35 \cdot 10^{-02}$ |
| | Shifting expansion point constraint | | | |
| 15 | 24.466 | 0.00 | $(19.105, 14.682)$ | $3.92 \cdot 10^{-01}$ |
| 16 | 24.403 | 0.00 | $(18.432, 15.215)$ | $4.76 \cdot 10^{-01}$ |
| | Shifting expansion point constraint | | | |
| 17 | 24.341 | 0.00 | $(17.768, 15.782)$ | $5.40 \cdot 10^{-01}$ |
| 18 | 24.281 | 0.00 | $(17.153, 16.351)$ | $5.65 \cdot 10^{-01}$ |
| | Shifting expansion point constraint | | | |
| 19 | 24.228 | 0.00 | $(16.574, 16.925)$ | $5.87 \cdot 10^{-01}$ |
| 20 | 24.179 | 0.00 | $(16.044, 17.489)$ | $5.83 \cdot 10^{-01}$ |
| | Shifting expansion point constraint | | | |
| 21 | 24.136 | 0.00 | $(15.554, 18.044)$ | $5.82 \cdot 10^{-01}$ |
| 22 | 24.098 | 0.00 | $(15.105, 18.586)$ | $5.24 \cdot 10^{-01}$ |
| | Shifting expansion point constraint | | | |
| 23 | 24.065 | $1.23 \cdot 10^{-02}$ | $(14.784, 19.000)$ | $6.81 \cdot 10^{-03}$ |
| | Shifting expansion point constraint | | | |
| 24 | 24.056 | $7.09 \cdot 10^{-05}$ | $(14.790, 19.000)$ | $6.23 \cdot 10^{-05}$ |

**Table 7.12.** Robust optimization with quadratic Taylor model and shifting expansion point for 500 timesteps for linear material.

| iteration | $f^{wc}$ | $\max(0, c^{wc})$ | $\mu$ | opt |
|---|---|---|---|---|
| 0 | 26.276 | $2.83 \cdot 10^{-01}$ | $(22.000, 12.000)$ | 1.01 |
| 1 | 24.124 | $6.45 \cdot 10^{-01}$ | $(22.427, 12.910)$ | $2.63 \cdot 10^{-01}$ |
| 2 | 25.135 | 0.00 | $(22.185, 12.449)$ | $2.27 \cdot 10^{-01}$ |
| 3 | 24.362 | $3.65 \cdot 10^{-01}$ | $(22.351, 12.818)$ | $1.63 \cdot 10^{-01}$ |
| 4 | 24.991 | 0.00 | $(22.196, 12.531)$ | $1.77 \cdot 10^{-01}$ |
| 5 | 24.726 | $5.03 \cdot 10^{-03}$ | $(22.252, 12.660)$ | $1.10 \cdot 10^{-02}$ |
| 6 | 24.731 | 0.00 | $(22.242, 12.662)$ | $1.06 \cdot 10^{-02}$ |
| 7 | 24.684 | 0.00 | $(21.526, 13.060)$ | $1.17 \cdot 10^{-02}$ |
| 8 | 24.674 | 0.00 | $(21.143, 13.275)$ | $9.48 \cdot 10^{-03}$ |
| 9 | 24.660 | 0.00 | $(20.691, 13.541)$ | $3.92 \cdot 10^{-01}$ |
| 10 | 24.505 | $3.00 \cdot 10^{-03}$ | $(19.990, 14.067)$ | $1.10 \cdot 10^{-01}$ |
| 11 | 24.493 | 0.00 | $(18.664, 14.976)$ | $7.06 \cdot 10^{-01}$ |
| 12 | 24.392 | $3.92 \cdot 10^{-02}$ | $(19.297, 14.600)$ | 1.31 |
| 13 | 24.395 | 0.00 | $(18.520, 15.154)$ | $2.91 \cdot 10^{-01}$ |
| 14 | 24.339 | 0.00 | $(17.918, 15.661)$ | $1.07 \cdot 10^{-01}$ |
| 15 | 24.272 | 0.00 | $(17.348, 16.188)$ | $4.39 \cdot 10^{-01}$ |
| 16 | 24.222 | 0.00 | $(16.714, 16.797)$ | $5.44 \cdot 10^{-01}$ |
| 17 | 24.168 | 0.00 | $(16.222, 17.317)$ | $4.83 \cdot 10^{-01}$ |
| 18 | 24.128 | 0.00 | $(15.693, 17.901)$ | $5.78 \cdot 10^{-01}$ |
| 19 | 24.085 | 0.00 | $(15.245, 18.435)$ | $5.19 \cdot 10^{-01}$ |
| 20 | 24.050 | 0.00 | $(14.803, 18.990)$ | $1.13 \cdot 10^{-02}$ |
| 21 | 24.033 | $4.12 \cdot 10^{-04}$ | $(14.807, 19.000)$ | $3.50 \cdot 10^{-04}$ |
| 22 | 24.034 | 0.00 | $(14.807, 19.000)$ | $3.69 \cdot 10^{-06}$ |

**Table 7.13.** Robust optimization with quadratic interpolation model for 500 timesteps for linear material.

| iteration | f | max(0,c) | $\mu$ | opt |
|---|---|---|---|---|
| 0 | 11.43 | 0.00 | $(22.00, 12.00)$ | $5.81 \cdot 10^{-01}$ |
| 1 | 11.43 | 0.00 | $(21.97, 12.02)$ | $2.68 \cdot 10^{-02}$ |
| 2 | 11.42 | 0.00 | $(21.81, 12.12)$ | $3.65 \cdot 10^{-02}$ |
| 3 | 11.41 | 0.00 | $(21.16, 12.59)$ | $1.76 \cdot 10^{-01}$ |
| 4 | 11.38 | 0.00 | $(21.38, 12.50)$ | $4.27 \cdot 10^{-01}$ |
| 5 | 11.37 | 0.00 | $(20.49, 13.59)$ | $3.13 \cdot 10^{-01}$ |
| 6 | 10.96 | 0.00 | $(22.18, 13.01)$ | $2.78 \cdot 10^{-01}$ |
| 7 | 09.77 | 0.00 | $(28.00, 12.52)$ | $2.15 \cdot 10^{-01}$ |
| 8 | 09.74 | 0.00 | $(28.00, 12.85)$ | $3.43 \cdot 10^{-02}$ |
| 9 | 09.74 | 0.00 | $(28.00, 12.91)$ | $3.53 \cdot 10^{-03}$ |
| 10 | 09.74 | 0.00 | $(28.00, 12.90)$ | $6.69 \cdot 10^{-05}$ |

**Table 7.14.** Nominal optimization for 200 timesteps for nonlinear material.

| iteration | $f^{wc}$ | $\max(0, c^{wc})$ | $\mu$ | opt |
|---|---|---|---|---|
| 0 | 13.539 | $2.53 \cdot 10^{-01}$ | $(22.000, 12.000)$ | $7.02 \cdot 10^{-01}$ |
| 1 | 12.969 | $5.59 \cdot 10^{-02}$ | $(22.293, 12.638)$ | $4.71 \cdot 10^{-02}$ |
| 2 | 13.032 | $2.36 \cdot 10^{-02}$ | $(22.379, 12.475)$ | $2.83 \cdot 10^{-01}$ |
| 3 | 12.985 | $3.90 \cdot 10^{-02}$ | $(22.442, 12.512)$ | $9.52 \cdot 10^{-02}$ |
| 4 | 12.984 | $3.63 \cdot 10^{-02}$ | $(22.805, 12.294)$ | $8.18 \cdot 10^{-02}$ |
| 5 | 12.969 | $3.69 \cdot 10^{-02}$ | $(22.652, 12.407)$ | $2.31 \cdot 10^{-01}$ |
| 6 | 12.937 | $4.52 \cdot 10^{-02}$ | $(23.229, 12.129)$ | $1.10 \cdot 10^{-01}$ |
| 7 | 12.937 | $3.80 \cdot 10^{-02}$ | $(23.012, 12.248)$ | $3.79 \cdot 10^{-01}$ |
| 8 | 12.924 | $3.94 \cdot 10^{-02}$ | $(23.255, 12.134)$ | $1.00 \cdot 10^{-01}$ |
| 9 | 12.923 | $3.86 \cdot 10^{-02}$ | $(23.255, 12.134)$ | $1.77 \cdot 10^{-01}$ |
| 10 | 12.923 | $3.77 \cdot 10^{-02}$ | $(23.239, 12.144)$ | $5.70 \cdot 10^{-02}$ |
| 11 | 12.923 | $3.70 \cdot 10^{-02}$ | $(23.217, 12.156)$ | $3.26 \cdot 10^{-02}$ |
| 12 | 12.926 | $3.64 \cdot 10^{-02}$ | $(23.202, 12.159)$ | $2.38 \cdot 10^{-02}$ |
| 13 | 12.925 | $3.64 \cdot 10^{-02}$ | $(23.199, 12.162)$ | $1.95 \cdot 10^{-02}$ |
| 14 | 12.933 | $3.50 \cdot 10^{-02}$ | $(23.134, 12.185)$ | $2.29 \cdot 10^{-02}$ |
| 15 | 12.938 | $3.41 \cdot 10^{-02}$ | $(23.089, 12.203)$ | $1.37 \cdot 10^{-01}$ |
| 16 | 12.987 | $2.76 \cdot 10^{-02}$ | $(22.692, 12.355)$ | $2.63 \cdot 10^{-01}$ |
| 17 | 13.004 | $2.58 \cdot 10^{-02}$ | $(22.573, 12.401)$ | $9.63 \cdot 10^{-03}$ |
| 18 | 13.022 | $2.41 \cdot 10^{-02}$ | $(22.441, 12.452)$ | $2.97 \cdot 10^{-03}$ |
| 19 | 13.041 | $2.24 \cdot 10^{-02}$ | $(22.441, 12.452)$ | $2.51 \cdot 10^{-03}$ |
| 20 | 13.061 | $2.04 \cdot 10^{-02}$ | $(22.176, 12.556)$ | $7.65 \cdot 10^{-02}$ |
| 21 | 13.130 | $1.32 \cdot 10^{-02}$ | $(21.741, 12.735)$ | $3.75 \cdot 10^{-01}$ |
| 22 | 13.141 | $5.34 \cdot 10^{-03}$ | $(21.584, 12.831)$ | $1.49 \cdot 10^{-01}$ |
| 23 | 13.170 | 0.00 | $(21.584, 12.831)$ | $5.97 \cdot 10^{-02}$ |
| 24 | 13.165 | $1.18 \cdot 10^{-04}$ | $(21.475, 12.871)$ | $4.77 \cdot 10^{-02}$ |
| 25 | 13.164 | $1.25 \cdot 10^{-04}$ | $(21.480, 12.868)$ | $4.92 \cdot 10^{-03}$ |
| 26 | 13.164 | $3.87 \cdot 10^{-05}$ | $(21.476, 12.870)$ | $6.36 \cdot 10^{-04}$ |
| 27 | 13.165 | 0.00 | $(21.474, 12.871)$ | $1.91 \cdot 10^{-03}$ |
| 28 | 13.164 | $1.88 \cdot 10^{-05}$ | $(21.476, 12.870)$ | $1.87 \cdot 10^{-04}$ |
| 29 | 13.165 | $1.20 \cdot 10^{-05}$ | $(21.476, 12.870)$ | $1.88 \cdot 10^{-04}$ |
| 30 | 13.165 | $5.05 \cdot 10^{-06}$ | $(21.476, 12.870)$ | $4.41 \cdot 10^{-05}$ |
| 31 | 13.165 | 0.00 | $(21.475, 12.871)$ | $6.74 \cdot 10^{-05}$ |

**Table 7.15.** Robust optimization with linear Taylor approximation for 200 timesteps for nonlinear material.

| iteration | $f^{wc}$ | $\max(0, c^{wc})$ | $\mu$ | opt |
|---|---|---|---|---|
| 0 | 14.156 | $2.69 \cdot 10^{-02}$ | $(22.000, 12.000)$ | $5.62 \cdot 10^{-01}$ |
| 1 | 12.990 | $9.10 \cdot 10^{-02}$ | $(22.941, 14.041)$ | $2.91 \cdot 10^{-01}$ |
| 2 | 12.943 | $1.67 \cdot 10^{-02}$ | $(22.932, 13.346)$ | $4.33 \cdot 10^{-01}$ |
| 3 | 12.712 | $4.70 \cdot 10^{-02}$ | $(24.106, 12.866)$ | $2.18 \cdot 10^{-02}$ |
| 4 | 12.872 | $2.52 \cdot 10^{-02}$ | $(23.326, 13.123)$ | $6.69 \cdot 10^{-02}$ |
| 5 | 13.017 | $7.58 \cdot 10^{-03}$ | $(22.711, 13.338)$ | $2.66 \cdot 10^{-02}$ |
| 6 | 13.040 | $4.77 \cdot 10^{-03}$ | $(22.590, 13.423)$ | $7.47 \cdot 10^{-03}$ |
| 7 | 13.083 | 0.00 | $(22.431, 13.483)$ | $7.86 \cdot 10^{-03}$ |
| 8 | 13.082 | $2.25 \cdot 10^{-06}$ | $(22.429, 13.490)$ | $2.27 \cdot 10^{-04}$ |
| 9 | 13.082 | $1.98 \cdot 10^{-06}$ | $(22.429, 13.491)$ | $7.52 \cdot 10^{-03}$ |
| 10 | 13.082 | 0.00 | $(22.429, 13.491)$ | $1.79 \cdot 10^{-04}$ |
| 11 | 13.082 | $1.65 \cdot 10^{-09}$ | $(22.429, 13.491)$ | $1.17 \cdot 10^{-05}$ |

**Table 7.16.** Robust optimization with quadratic Taylor model for 200 timesteps for nonlinear material.

| iteration | $f^{wc}$ | $\max(0, c^{wc})$ | $\mu$ | opt |
|---|---|---|---|---|
| | Shifting expansion point constraint | | | |
| 0 | 14.410 | $2.90 \cdot 10^{-02}$ | $(22.000, 12.000)$ | $5.49 \cdot 10^{-01}$ |
| | Shifting expansion point constraint | | | |
| 1 | 13.138 | 0.00 | $(22.938, 13.985)$ | $1.82 \cdot 10^{-01}$ |
| | Shifting expansion point constraint | | | |
| 2 | 12.710 | $2.21 \cdot 10^{-02}$ | $(24.764, 12.901)$ | 1.09 |
| | Shifting expansion point constraint | | | |
| 3 | 12.372 | $2.87 \cdot 10^{-02}$ | $(26.517, 13.081)$ | 1.72 |
| | Shifting expansion point constraint | | | |
| 4 | 12.454 | $1.39 \cdot 10^{-02}$ | $(25.442, 13.270)$ | $5.28 \cdot 10^{-01}$ |
| | Shifting expansion point constraint | | | |
| 5 | 12.398 | $6.71 \cdot 10^{-03}$ | $(25.929, 13.555)$ | $1.21 \cdot 10^{-01}$ |
| 6 | 12.403 | $5.81 \cdot 10^{-03}$ | $(25.985, 13.587)$ | $4.44 \cdot 10^{-02}$ |
| 7 | 12.416 | $4.23 \cdot 10^{-03}$ | $(25.943, 13.620)$ | $2.18 \cdot 10^{-02}$ |
| 8 | 12.426 | $2.94 \cdot 10^{-03}$ | $(25.916, 13.648)$ | $6.44 \cdot 10^{-05}$ |
| 9 | 12.446 | $7.63 \cdot 10^{-04}$ | $(25.878, 13.696)$ | $6.20 \cdot 10^{-03}$ |
| 10 | 12.449 | $5.27 \cdot 10^{-04}$ | $(25.874, 13.701)$ | $3.60 \cdot 10^{-05}$ |

**Table 7.17.** Robust optimization with quadratic Taylor model and shifted expansion point for 200 timesteps for nonlinear material.

| energy | number of modes | | | | error in norm |
|---|---|---|---|---|---|
| $\varepsilon$ | $u_{st}^{nonlin}$ | $u_{rt}^{nonlin}$ | $u_{b}^{nonlin}$ | $\Sigma$ | $\|u_{deim}^{h} - u^{h}\|$ |
| $1-10^{-03}$ | 7 | 15 | 7 | 84 | $4.0409 \cdot 10^{-07}$ |
| $1-10^{-04}$ | 13 | 22 | 9 | 115 | $1.9011 \cdot 10^{-08}$ |
| $1-10^{-05}$ | 17 | 28 | 13 | 153 | $7.0053 \cdot 10^{-09}$ |
| $1-10^{-06}$ | 25 | 36 | 17 | 200 | $4.2081 \cdot 10^{-09}$ |
| $1-10^{-07}$ | 36 | 48 | 22 | 271 | $3.7863 \cdot 10^{-09}$ |
| $1-10^{-08}$ | 44 | 46 | 27 | 341 | $3.5506 \cdot 10^{-09}$ |
| $1-10^{-09}$ | 58 | 77 | 33 | 422 | $2.4296 \cdot 10^{-09}$ |
| $1-10^{-10}$ | 74 | 96 | 37 | 523 | $5.8002 \cdot 10^{-10}$ |

**Table 7.18.** Dimension of the DEIM basis for 200 timesteps and resulting error and given energy $\varepsilon$ for the nonlinear basis.

| energy | number of modes | | | | error in norm | |
|---|---|---|---|---|---|---|
| $\varepsilon$ | $a_{st}$ | $a_{rt}$ | $a_{b}$ | $\Sigma$ | $\|u_{deim}^{l} - u^{h}\|$ | $\|\lambda_{deim}^{l} - \lambda^{h}\|$ |
| $1-10^{-03}$ | 2 | 3 | 2 | 25 | $2.0739 \cdot 10^{-05}$ | $1.7640 \cdot 10^{-04}$ |
| $1-10^{-04}$ | 3 | 3 | 2 | 26 | $2.0323 \cdot 10^{-05}$ | $1.4242 \cdot 10^{-04}$ |
| $1-10^{-05}$ | 3 | 5 | 2 | 28 | $1.6694 \cdot 10^{-05}$ | $6.3805 \cdot 10^{-05}$ |
| $1-10^{-06}$ | 5 | 7 | 3 | 33 | $4.7458 \cdot 10^{-05}$ | $6.2536 \cdot 10^{-04}$ |
| $1-10^{-07}$ | 9 | 10 | 4 | 43 | $9.3243 \cdot 10^{-07}$ | $9.5459 \cdot 10^{-06}$ |
| $1-10^{-08}$ | 15 | 13 | 6 | 60 | $1.6571 \cdot 10^{-06}$ | $7.1421 \cdot 10^{-06}$ |
| $1-10^{-09}$ | 17 | 14 | 7 | 75 | $2.5979 \cdot 10^{-06}$ | $6.8764 \cdot 10^{-06}$ |
| $1-10^{-10}$ | 21 | 23 | 8 | 99 | $2.9915e \cdot 10^{-06}$ | $5.9496 \cdot 10^{-06}$ |

**Table 7.19.** Fixed DEIM basis for 200 timesteps and resulting error and given energy $\varepsilon$ for the basis.

CHAPTER 8

# Conclusion

In this thesis we have investigated the robust shape optimization of an induction machine using reduced order models. We have introduced the physical principles that can be used to describe the behavior of an asynchronous machine and gave a proof of the existence of a solution to the equations describing the fields of the machine by the magnetoquasistatic approximation of Maxwell's equations, which are coupled with equations describing the rotor network and equations describing the inductance effects of the stator windings. The set of equations is also coupled with an equation of motion excited by the electromagnetic torque produced in the airgap of the machine.

Having set the physical foundation for the simulation of an induction machine, we have described how we use the finite element method to discretize the magnetic vector potential in space. We have described how the boundary conditions are applied in the numerical simulation and seen different methods for the coupling of rotor and stator when using a domain decomposition approach. Further we have introduced the parametric shape optimization approach we are using and how we can utilize the affine parametrization of our domain to pre-compute system matrices on a reference domain, which can then be summed up with weighted coefficients depending on the design to obtain the system matrices for different parameters, which we need to solve the state equation for different designs and uncertainties. Since in realistic models, in ferromagnetic regions the material coefficient depends nonlinearly on the underlying magnetic flux density, we have seen how we can consider this in the simulation and how we set up the needed nonlinear stiffness matrix. The time stepping scheme we are using splits the solution of the state system into the solution of the equation of motion, the field and circuit coupled equations and the torque computation which leads to an efficient scheme that does not involve the solution of a nonlinear system, when linearized material is considered. If nonlinear material is

considered, the nonlinear field/circuit coupled equations are solved using Newton's method.

As the discretization of parabolic partial differential equations in space and time leads to systems with a huge amount of degrees of freedom, we have described how the proper orthogonal decomposition method can be used in the simulation and optimization to reduce the degrees of freedom by approximating the state, adjoint and sensitivity space. If nonlinear material is considered, we can make use of the discrete empirical interpolation method to efficiently assemble the nonlinear terms.

Since we consider uncertainties in our optimization and treat them via a worst-case approach, we have described approximation techniques for the worst-case of objective and constraint functions involving derivatives aswell as derivative free methods.

Finally we have used the described techniques to perform robust shape optimization with reduced order models on an induction machine. Here the objective was to minimize the averaged Joule losses in the rotor bars while preserving a given averaged torque. We have considered geometric aswell as material uncertainties and have seen, that respecting uncertainties in the optimization can make a significant difference in the solution, compared to non-robust solutions.

CHAPTER A

# Derivation of Equations

## A.1 Derivatives of nonlinear stiffness matrix

We will present here some of the formulas needed to compute the derivative of the stiffness matrix, when the material depends on the underlying magnetic flux density.

### Derivatives with respect to the state

We will assume here, that the whole domain has nonlinear material properties. (In regions where this is not the case, the first order derivative of the stiffness matrix with respect to the state is constant and the second order derivative vanishes.)

As we have seen in (4.35), the derivative of the stiffness matrix is given by

$$\sum_{k=1}^{\tilde{L}+1} \int_{\hat{\Omega}_k} \left( \begin{pmatrix} f_1\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right) & f_2\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right) \\ f_3\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right) & f_4\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right) \end{pmatrix} \left(C_k^{-T}(p)\hat{\nabla}\hat{s}\right) \right)^T (C_k^{-T}\hat{\nabla}\hat{v})|\det C_k(p)|\,\mathrm{d}\hat{x},$$

where

$$\begin{aligned} f_1\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right) &= \nu(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2) + 2\nu'(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2)A_{1,1}, \\ f_2\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right) &= 2\nu'(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2)A_{1,2}, \\ f_3\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right) &= 2\nu'(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2)A_{2,1}, \\ f_4\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right) &= \nu(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2) + 2\nu'(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2)A_{2,2}, \end{aligned}$$

with

$$A := \left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)^T.$$

We have

$$C_k^{-T}(p)\hat{\nabla}\hat{u} = \begin{pmatrix} \left(C_k^{-1}(p)\right)_{1,1}\hat{\partial}_x\hat{u} + \left(C_k^{-1}(p)\right)_{2,1}\hat{\partial}_y\hat{u} \\ \left(C_k^{-1}(p)\right)_{1,2}\hat{\partial}_x\hat{u} + \left(C_k^{-1}(p)\right)_{2,2}\hat{\partial}_y\hat{u} \end{pmatrix}$$

Leading to

$$\begin{aligned}\sum_{k=1}^{\tilde{L}+1}\int_{\hat{\Omega}_k}\Bigg(&\begin{pmatrix} f_1\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)\left(C_k^{-1}(p)\right)_{1,1}\hat{\partial}_x\hat{s} + f_1\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)\left(C_k^{-1}(p)\right)_{2,1}\hat{\partial}_y\hat{s} \\ f_3\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)\left(C_k^{-1}(p)\right)_{1,1}\hat{\partial}_x\hat{s} + f_3\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)\left(C_k^{-1}(p)\right)_{2,1}\hat{\partial}_y\hat{s} \end{pmatrix} \\ &+ \begin{pmatrix} f_2\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)\left(C_k^{-1}(p)\right)_{1,2}\hat{\partial}_x\hat{s} + f_2\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)\left(C_k^{-1}(p)\right)_{2,2}\hat{\partial}_y\hat{s} \\ f_4\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)\left(C_k^{-1}(p)\right)_{1,2}\hat{\partial}_x\hat{s} + f_4\left(C_k^{-T}(p)\hat{\nabla}\hat{u}\right)\left(C_k^{-1}(p)\right)_{2,2}\hat{\partial}_y\hat{s} \end{pmatrix}\Bigg)^T \\ &\begin{pmatrix} \left(C_k^{-1}(p)\right)_{1,1}\hat{\partial}_x\hat{v} + \left(C_k^{-1}(p)\right)_{2,1}\hat{\partial}_y\hat{v} \\ \left(C_k^{-1}(p)\right)_{1,2}\hat{\partial}_x\hat{v} + \left(C_k^{-1}(p)\right)_{2,2}\hat{\partial}_y\hat{v} \end{pmatrix} |\det C_k(p)|\,\mathrm{d}\hat{x},\end{aligned}$$

where we have to compute the integrals

$$
\begin{aligned}
\sum_{k=1}^{\tilde{L}+1} \int_{\hat{\Omega}_k} ( & f_1 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{1,1} \left( C_k^{-1}(p) \right)_{1,1} \hat{\partial}_x \hat{s} \hat{\partial}_x \hat{v} \\
& + f_1 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{2,1} \left( C_k^{-1}(p) \right)_{1,1} \hat{\partial}_y \hat{s} \hat{\partial}_x \hat{v} \\
& + f_1 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{1,1} \left( C_k^{-1}(p) \right)_{2,1} \hat{\partial}_x \hat{s} \hat{\partial}_y \hat{v} \\
& + f_1 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{2,1} \left( C_k^{-1}(p) \right)_{2,1} \hat{\partial}_y \hat{s} \hat{\partial}_y \hat{v} \\
& + f_2 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{1,2} \left( C_k^{-1}(p) \right)_{1,1} \hat{\partial}_x \hat{s} \hat{\partial}_x \hat{v} \\
& + f_2 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{2,2} \left( C_k^{-1}(p) \right)_{1,1} \hat{\partial}_y \hat{s} \hat{\partial}_x \hat{v} \\
& + f_2 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{1,2} \left( C_k^{-1}(p) \right)_{2,1} \hat{\partial}_x \hat{s} \hat{\partial}_y \hat{v} \\
& + f_2 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{2,2} \left( C_k^{-1}(p) \right)_{2,1} \hat{\partial}_y \hat{s} \hat{\partial}_y \hat{v} \\
& + f_3 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{1,1} \left( C_k^{-1}(p) \right)_{1,2} \hat{\partial}_x \hat{s} \hat{\partial}_x \hat{v} \\
& + f_3 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{2,1} \left( C_k^{-1}(p) \right)_{1,2} \hat{\partial}_y \hat{s} \hat{\partial}_x \hat{v} \\
& + f_3 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{1,1} \left( C_k^{-1}(p) \right)_{2,2} \hat{\partial}_x \hat{s} \hat{\partial}_y \hat{v} \\
& + f_3 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{2,1} \left( C_k^{-1}(p) \right)_{2,2} \hat{\partial}_y \hat{s} \hat{\partial}_y \hat{v} \\
& + f_4 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{1,2} \left( C_k^{-1}(p) \right)_{1,2} \hat{\partial}_x \hat{s} \hat{\partial}_x \hat{v} \\
& + f_4 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{2,2} \left( C_k^{-1}(p) \right)_{1,2} \hat{\partial}_y \hat{s} \hat{\partial}_x \hat{v} \\
& + f_4 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{1,2} \left( C_k^{-1}(p) \right)_{2,2} \hat{\partial}_x \hat{s} \hat{\partial}_y \hat{v} \\
& + f_4 \left( C_k^{-T}(p) \hat{\nabla} \hat{u} \right) \left( C_k^{-1}(p) \right)_{2,2} \left( C_k^{-1}(p) \right)_{2,2} \hat{\partial}_y \hat{s} \hat{\partial}_y \hat{v} ) | \det C_k(p) | \, \mathrm{d}\hat{x}.
\end{aligned}
\tag{A.1}
$$

For the derivation of the second order derivative of the nonlinear stiffness matrix with respect to the state variable we inspect the derivative

$$
\begin{pmatrix} \left(2\nu'(x_1^2+x_2^2)x_1^2 + \nu(x_1^2+x_2^2)\right) s_1 + \left(2\nu'(x_1^2+x_2^2)x_1x_2\right) s_2 \\ \left(2\nu'(x_1^2+x_2^2)x_1x_2\right) s_1 + \left(2\nu'(x_1^2+x_2^2)x_2^2 + \nu(x_1^2+x_2^2)\right) s_2 \end{pmatrix}' =: \begin{pmatrix} a & b \\ c & d \end{pmatrix},
$$

which is given by

$$\begin{aligned}
a(x,s) &= 2\nu'\left(x^Tx\right)s_1x_1 + 4\nu''\left(x^Tx\right)x_1^2\left(x_1s_1+x_2s_2\right) + 2\nu'\left(x^Tx\right)\left(2x_1s_1+x_2s_2\right)\\
&= 4\nu''\left(x^Tx\right)x_1^2\left(x^Ts\right) + 2\nu'\left(x^Tx\right)\left(2x_1s_1+x^Ts\right),\\
b(x,s) &= 2\nu'\left(x^Tx\right)s_1x_2 + 4\nu''\left(x^Tx\right)x_1x_2\left(x_1s_1+x_2s_2\right) + 2\nu'\left(x^Tx\right)x_1s_2,\\
c(x,s) &= 2\nu'\left(x^Tx\right)s_2x_1 + 4\nu''\left(x^Tx\right)x_1x_2\left(x_1s_1+x_2s_2\right) + 2\nu'\left(x^Tx\right)x_2s_1,\\
d(x,s) &= 2\nu'\left(x^Tx\right)s_2x_2 + 4\nu''\left(x^Tx\right)x_2^2\left(x_1s_1+x_2s_2\right) + 2\nu'\left(x^Tx\right)\left(x_1s_1+2x_2s_2\right).
\end{aligned}$$

Defining $\mathrm{d}uu(x,s) := \begin{pmatrix} a(x,s) & b(x,s)\\ c(x,s) & d(x,s)\end{pmatrix}$, we get for the second order derivative of the stiffness matrix with respect to the state variable in direction $\hat{s}$ and $\hat{s}_2$

$$\sum_{k=1}^{\tilde{L}+1}\sum_{i,j}^{2}\int_{\hat{\Omega}_k}\left(\mathrm{d}uu\left(C_k^{-T}(p)\hat{\nabla}\hat{u}, C_k^{-T}\hat{\nabla}\hat{s}\right)\left(C_k^{-T}\hat{\nabla}\hat{s}_2\right)\right)^T\left(C_k^{-T}\hat{\nabla}\hat{v}\right)|\det C_k(p)|\,\mathrm{d}\hat{x}.$$

This equation has the same structe as (A.1) only the $f_i$ are replaced by $(\mathrm{d}uu)_{i,j}$.

For the third order derivative of the stiffness matrix with respect to the state we have to differentiate $\mathrm{d}uu(x,s)\tilde{s}$ with respect to $x$. We make the following definitions

$$\begin{aligned}
&h(x,s) = x^Ts, \quad h_2(x,s) = x_1s_2 + x_2s_1,\\
&\epsilon(x,s) = 4\nu''(x^Tx)h(x,s), \quad \delta(x) = 2\nu'(x^Tx),\\
&Z^u(x,s,\tilde{s}) = x_1^2\tilde{s}_1 + x_1x_2\tilde{s}_2, \quad Z^l(x,s,\tilde{s}) = x_2^2\tilde{s}_2 + x_1x_2\tilde{x}_1,\\
&X^u(x,s,\tilde{s}) = (h(x,s)+2x_1s_1)\tilde{s}_1 + h_2(x,s)\tilde{s}_2,\\
&X^l(x,s,\tilde{s}) = (h_2(x,s)+2x_2s_2)\tilde{s}_2 + h(x,s)\tilde{s}.
\end{aligned}$$

Then $\mathrm{d}uu(x,s)\tilde{s}$ can be written as

$$\mathrm{d}uu(x,s)\tilde{s} = \begin{pmatrix}\epsilon(x,s)Z^u(x,s,\tilde{s}) + \delta(x)X^u(x,s,\tilde{s})\\ \epsilon(x,s)Z^l(x,s,\tilde{s}) + \delta(x)X^l(x,s,\tilde{s})\end{pmatrix}$$

and the derivative $\mathrm{d}uuu(x,s,\tilde{s}) :- \left(\nabla_x \mathrm{d}uu(x,s)\tilde{s}\right)^T$ is given by

$$\begin{aligned}
\mathrm{d}uuu(x,s,\tilde{s}) = &\begin{pmatrix}(\nabla_x\epsilon(x,s))^TZ^u(x,s,\tilde{s}) + (\nabla\delta(x))^TX^u(x,s,\tilde{s})\\ (\nabla_x\epsilon(x,s))^TZ^l(x,s,\tilde{s}) + (\nabla\delta(x))^TX^l(x,s,\tilde{s})\end{pmatrix}\\
&+\begin{pmatrix}\epsilon(x,s)(\nabla_xZ^u(x,s,\tilde{s}))^T + \delta(x)(\nabla_xX^u(x,s,\tilde{s}))^T\\ \epsilon(x,s)(\nabla_xZ^l(x,s,\tilde{s}))^T + \delta(x)(\nabla_xX^l(x,s,\tilde{s}))^T\end{pmatrix}.
\end{aligned}$$

The third order derivative in direction $\hat{s}_3$ is given by

$$\sum_{k=1}^{\tilde{L}+1} \int_{\hat{\Omega}_k} \left( \mathrm{d}uuu \left( C_k^{-T}(p)\hat{\nabla}\hat{u}, C_k^{-T}\hat{\nabla}\hat{s}, C_k^{-T}\hat{\nabla}\hat{s}_2 \right) \left( C_k^{-T}\hat{\nabla}\hat{s}_3 \right) \right)^T \left( C_k^{-T}\hat{\nabla}\hat{v} \right) |\det C_k(p)| \, \mathrm{d}\hat{x}.$$

This equation again has the same structe as (A.1) only the $f_i$ are replaced by $(duuu)_{i,j}$.

## Derivatives with respect to parameters

We will state here derivatives of the stiffness matrix with respect to the parameter $p$. We define some needed terms and their derivatives

$$\begin{aligned}
a^{k,i,j} &:= \left[ C_k^{-1}(p) C_k^{-T}(p) \right]_{ij} |\det C_k(p)| \,, \\
a_{p_1}^{k,i,j} &= \left[ C_k^{-1}(p) C_k^{-T}(p) \right]_{ij} \frac{\partial}{\partial p_1} |\det C_k(p)| \\
&\quad + \left[ \frac{\partial}{\partial p_1} C_k^{-1}(p) C_k^{-T}(p) + C_k^{-1}(p) \frac{\partial}{\partial p_1} C_k^{-T}(p) \right]_{ij} |\det C_k(p)|, \\
b^k &:= \nu(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2), \\
b_{p_1}^k &= 2\nu'(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2) \left( \frac{\partial}{\partial p_1} C_k^{-T}(p)\hat{\nabla}\hat{u} \right)^T C_k^{-T}(p)\hat{\nabla}\hat{u}.
\end{aligned}$$

For the derivative of the stiffness matrix with respect to one parameter $p_1$ of the design, we have the following formula

$$\sum_{k=1}^{\tilde{L}+1} \sum_{i,j}^{2} \left( a_{p_1}^{k,i,j} \int_{\hat{\Omega}_k} b^k \frac{\partial u}{\partial \hat{x}_i} \frac{\partial v}{\partial \hat{x}_j} \, \mathrm{d}\hat{x} + a^{k,i,j} \int_{\hat{\Omega}_k} b_{p_1}^k \frac{\partial u}{\partial \hat{x}_i} \frac{\partial v}{\partial \hat{x}_j} \, \mathrm{d}\hat{x} \right).$$

Here $a$ and $a_{p_i}$ are precomputed with symbolic values for the parameter. Meaning, we have to compute 8 matrices. All the parts outside the transformable region vanish since there $\frac{d}{dp} C_k = 0$.

For the second order derivative with respect to parameters $p_1$ and $p_2$ we will again

introduce some notation

$$
\begin{aligned}
a^k_{p_1p_2} = &\left[\frac{\partial}{\partial p_2}C_k^{-1}(p)C_k^{-T}(p) + C_k^{-1}(p)\frac{\partial}{\partial p_2}C_k^{-T}(p)\right]_{ij}\frac{\partial}{\partial p_1}|\det C_k(p)| \\
&+ \left[C_k^{-1}(p)C_k^{-T}(p)\right]_{ij}\frac{\partial}{\partial p_1\partial p_2}|\det C_k(p)| \\
&+ \left[\frac{\partial}{\partial p_1}C_k^{-1}(p)C_k^{-T}(p) + C_k^{-1}(p)\frac{\partial}{\partial p_1}C_k^{-T}(p)\right]_{ij}\frac{\partial}{\partial p_2}|\det C_k(p)| \\
&+ \Big[2\frac{\partial}{\partial p_1\partial p_2}C_k^{-1}(p)C_k^{-T}(p) + \frac{\partial}{\partial p_1}C_k^{-1}(p)\frac{\partial}{\partial p_2}C_k^{-T}(p) \\
&+ \frac{\partial}{\partial p_2}C_k^{-1}(p)\frac{\partial}{\partial p_1}C_k^{-T}(p)\Big]_{ij}|\det C_k(p)|, \\
b^k_{p_1p_2} = &\, 4\nu''(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2)\left(\frac{\partial}{\partial p_2}C_k^{-T}(p)\hat{\nabla}\hat{u}\right)^T C_k^{-T}(p)\hat{\nabla}\hat{u} \\
&\left(\frac{\partial}{\partial p_1}C_k^{-T}(p)\hat{\nabla}\hat{u}\right)^T C_k^{-T}(p)\hat{\nabla}\hat{u} \\
&+ 2\nu'(|C_k^{-T}(p)\hat{\nabla}\hat{u}|^2)\Bigg(\left(\frac{\partial}{\partial p_1\partial p_2}C_k^{-T}(p)\hat{\nabla}\hat{u}\right)^T C_k^{-T}(p)\hat{\nabla}\hat{u} \\
&+ \left(\frac{\partial}{\partial p_1}C_k^{-T}(p)\hat{\nabla}\hat{u}\right)^T \frac{\partial}{\partial p_2}C_k^{-T}(p)\hat{\nabla}\hat{u}\Bigg).
\end{aligned}
$$

We get for the second order derivative in direction $p_1$ and $p_2$

$$
\begin{aligned}
\sum_{k=1}^{\tilde{L}+1}\sum_{i,j}^{2}\Bigg(&a^k_{p_1p_2}\int_{\hat{\Omega}_k} b^k\frac{\partial u}{\partial\hat{x}_i}\frac{\partial v}{\partial\hat{x}_j}\,\mathrm{d}\hat{x} + a^k_{p_1}\int_{\hat{\Omega}_k} b^k_{p_2}\frac{\partial u}{\partial\hat{x}_i}\frac{\partial v}{\partial\hat{x}_j}\,\mathrm{d}\hat{x} \\
&+a^k_{p_2}\int_{\hat{\Omega}_k} b^k_{p_1}\frac{\partial u}{\partial\hat{x}_i}\frac{\partial v}{\partial\hat{x}_j}\,\mathrm{d}\hat{x} + a^k\int_{\hat{\Omega}_k} b^k_{p_1p_2}\frac{\partial u}{\partial\hat{x}_i}\frac{\partial v}{\partial\hat{x}_j}\,\mathrm{d}\hat{x}\Bigg).
\end{aligned}
$$

This requires to compute 16 integrals, where we can reuse some from the computation of the first derivative.

In the nonlinear material optimization, we use the conductivity as uncertainty and the second derivative wrt. design is constant for different designs and the third order derivative vanishes.

# List of Figures

# List of Tables

# List of Algorithms

# Bibliography


[1] H. W. Alt. *Lineare Funktionalanalysis, 6. Auflage*. Springer, 2012.

[2] A. Arkkio. *Analysis of induction motors based on the numerical solution of the magnetic field and circuit equations*. PhD thesis, Helsinki University of Technology, 1987.

[3] I. Babuška. The finite element method with lagrangian multipliers. *Numerische Mathematik*, 20(3):179–192, 1973.

[4] F. Bachinger, U. Langer, and J. Schöberl. Numerical analysis of nonlinear multiharmonic eddy current problems. *Numerische Mathematik*, 100:593–616, 2005.

[5] G. Bedrosian. A new method for coupling finite element field solutions with external circuits and kinematics. *IEEE Transactions on Magnetics*, 29(2):1664–1668, 1993.

[6] F. B. Belgacem. The mortar finite element method with lagrange multipliers. *Numerische Mathematik*, 84:173–197, 1999.

[7] A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of operations research*, 23(4):769–805, 1998.

[8] A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations research letters*, 25(1):1–13, 1999.

[9] A. Ben-Tal and A. Nemirovski. Robust optimization–methodology and applications. *Mathematical programming*, 92:453–480, 2002.

[10] A. Ben-Tal and A. Nemirovski. Selected topics in robust convex optimization. *Mathematical Programming*, 112:125–158, 2008.

[11] C. Bernardi. A new nonconforming approach to domain decomposition: the mortar element method. *Nonlinear partial equations and their applications*, 1989.

[12] D. Bertsimas, D. B. Brown, and C. Caramanis. Theory and applications of robust optimization. *SIAM review*, 53(3):464–501, 2011.

[13] J. K. Biehl. *Adaptive Multilevel Optimization of Fluid-Structure Interaction Problems*. PhD thesis, TU Darmstadt, 2020.

[14] Z. Bontinck, J. Corno, S. Schöps, and H. De Gersem. Isogeometric analysis and harmonic stator–rotor coupling for simulating electric machines. *Computer Methods in Applied Mechanics and Engineering*, 334:40–55, 2018.

[15] Z. Bontinck, O. Lass, O. Rain, and S. Schöps. Model order reduction for rotating electrical machines. In *Reduced-order modeling (ROM) for simulation and optimization*, pages 121–140. Springer, 2018.

[16] D. Braess. *Finite Elemente: Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie*. Springer-Verlag, 2013.

[17] D. Braess, W. Dahmen, and C. Wieners. A multigrid algorithm for the mortar finite element method. *SIAM Journal on Numerical Analysis*, 37(1):48–69, 1999.

[18] J. Brauer. Simple equations for the magnetization and reluctivity curves of steel. *IEEE Transactions on Magnetics*, 11(1):81–81, 1975.

[19] F. Brezzi and M. Fortin. *Mixed and hybrid finite element methods*, volume 15. Springer Science & Business Media, 2012.

[20] A. Buffa, Y. Maday, and F. Rapetti. A slideing mesh-mortar method for a two dimensional currents model of electric engines. *ESAIM: Mathematical Modelling and Numerical Analysis*, 35(2):191–228, 2001.

[21] S. Chaturantabut and D. C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

[22] S. Chaturantabut and D. C. Sorensen. A state space error estimate for pod-deim nonlinear model reduction. *SIAM Journal on numerical analysis*, 50(1):46–63, 2012.

[23] R. Chill, T. Reis, and T. Stykel. Analysis of a quasilinear coupled magneto-quasistatic model. part i: Solvability and regularity of solutions. *Hamburger Beiträge zur Angewandten Mathematik*, 1, 2021.

[24] P. G. Ciarlet. *The finite element method for elliptic problems*. SIAM, 2002.

[25] F. H. Clarke. Generalized gradients and applications. *Transactions of the American Mathematical Society*, 205:247–262, 1975.

[26] A. R. Conn, N. I. Gould, and P. L. Toint. *Trust region methods*. SIAM, 2000.

[27] A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to derivative-free optimization*. SIAM, 2009.

[28] A. R. Conn and L. N. Vicente. Bilevel derivative-free optimization and its application to robust optimization. *Optimization Methods and Software*, 27(3):561–577, 2012.

[29] F. E. Curtis, T. Mitchell, and M. L. Overton. A bfgs-sqp method for nonsmooth, nonconvex, constrained optimization and its evaluation using relative minimization profiles. *Optimization Methods and Software*, 32(1):148–181, 2017.

[30] H. De Gersem, J. Gyselinck, P. Dular, K. Hameyer, and T. Weiland. Comparison of sliding-surface and moving-band techniques in frequency-domain finite-element models of rotating machines. *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering*, 2004.

[31] H. De Gersem, K. Hameyer, and T. Weiland. Skew interface conditions in 2-d finite-element machine models. *IEEE Transactions on magnetics*, 39(3):1452–1455, 2003.

[32] H. De Gersem, K. Hameyer, and T. Weiland. Field–circuit coupled models in electromagnetic simulation. *Journal of computational and applied mathematics*, 168(1-2):125–133, 2004.

[33] H. De Gersem and T. Weiland. Harmonic weighting functions at the sliding interface of a finite-element machine model incorporating angular displacement. *IEEE Transactions on Magnetics*, 40(2):545–548, 2004.

[34] A. M. de Oliveira, R. Antunes, P. Kuo-Peng, N. Sadowski, and P. Dular. Electrical machine analysis considering field–circuit–movement and skewing effects. *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering*, 23(4):1080–1091, 2004.

[35] R. De Weerdt. *Eindige elementen modellering van kooianker inductiemotoren.* PhD thesis, Katholieke Universiteit Leuven, 1997.

[36] A. Demenko. Movement simulation in finite element analysis of electric machine dynamics. *IEEE Transactions on Magnetics*, 32(3):1553–1556, 1996.

[37] M. Diehl, H. G. Bock, and E. Kostina. An approximation technique for robust nonlinear optimization. *Mathematical Programming*, 107(1):213–230, 2006.

[38] Z. Drmac and A. K. Saibaba. The discrete empirical interpolation method: Canonical structure and formulation in weighted inner product spaces. *SIAM Journal on Matrix Analysis and Applications*, 39(3):1152–1180, 2018.

[39] H. Egger, M. Harutyunyan, R. Löscher, M. Merkel, and S. Schöps. On torque computation in electric machine simulation by harmonic mortar methods. *Journal of Mathematics in Industry*, 12(1):1–12, 2022.

[40] H. Egger, M. Harutyunyan, M. Merkel, and S. Schöps. On the stability of harmonic coupling methods with application to electric machines. In *Scientific Computing in Electrical Engineering: SCEE 2020, Eindhoven, The Netherlands, February 2020*, pages 117–125. Springer, 2021.

[41] L. C. Evans. *Partial differential equations*, volume 19. American Mathematical Society, 2022.

[42] P. Gangl. *Sensitivity-based topology and shape optimization with application to electrical machines*. PhD thesis, Universität Linz, 2016.

[43] M. A. Grepl and A. T. Patera. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(1):157–181, 2005.

[44] M. Gubisch and S. Volkwein. Proper orthogonal decomposition for linear-quadratic optimal control. *Model reduction and approximation: theory and algorithms*, 15(1), 2017.

[45] J. Gyselinck. *Twee-dimensionale dynamische eindige-elementenmodellering van statische en roterende elektromagnetische energieomzetters*. PhD thesis, Universiteit Gent, 2000.

[46] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(2):277–302, 2008.

[47] M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich. *Optimization with PDE constraints*, volume 23. Springer Science & Business Media, 2008.

[48] T. J. Hughes, J. A. Cottrell, and Y. Bazilevs. Isogeometric analysis: Cad, finite elements, nurbs, exact geometry and mesh refinement. *Computer methods in applied mechanics and engineering*, 194(39-41):4135–4195, 2005.

[49] K. Ito and S. S. Ravindran. A reduced basis method for control problems governed by pdes. Technical report, North Carolina State University. Center for Research in Scientific Computation, 1997.

[50] J. Kerler-Back and T. Stykel. Model reduction for linear and nonlinear magneto-quasistatic equations. *International Journal for Numerical Methods in Engineering*, 111(13):1274–1299, 2017.

[51] P. Kolvenbach. *Robust optimization of PDE-constrained problems using second-order models and nonsmooth approaches*. PhD thesis, TU Darmstadt, 2019.

[52] P. Kolvenbach, O. Lass, and S. Ulbrich. An approach for robust PDE-constrained optimization with application to shape optimization of electrical engines and of dynamic elastic structures under uncertainty. *Optim. Eng.*, 19(3):697–731, 2018.

[53] A. Kost. *Numerische Methoden in der Berechnung elektromagnetischer Felder*. Springer-Verlag, 2013.

[54] D. P. Kouri and T. M. Surowiec. Risk-averse pde-constrained optimization using the conditional value-at-risk. *SIAM Journal on Optimization*, 26(1):365–396, 2016.

[55] O. Lass. *Reduced order modeling and parameter identification for coupled nonlinear PDE systems*. PhD thesis, University of Konstanz, 2014.

[56] O. Lass and S. Ulbrich. Model order reduction techniques with a posteriori error control for nonlinear robust optimization governed by partial differential equations. *SIAM Journal on Scientific Computing*, 39(5):S112–S139, 2017.

[57] J. C. Maxwell. Viii. a dynamical theory of the electromagnetic field. *Philosophical transactions of the Royal Society of London*, (155):459–512, 1865.

[58] M. Merkel, P. Gangl, and S. Schöps. Shape optimization of rotating electric machines using isogeometric analysis. *IEEE Transactions on Energy Conversion*, 36(4):2683–2690, 2021.

[59] J. Milz and M. Ulbrich. An approximation scheme for distributionally robust nonlinear optimization. *SIAM Journal on Optimization*, 30(3):1996–2025, 2020.

[60] J. Milz and M. Ulbrich. An approximation scheme for distributionally robust pde-constrained optimization. *SIAM Journal on Control and Optimization*, 60(3):1410–1435, 2022.

[61] J. J. Moré and D. C. Sorensen. Computing a trust region step. *SIAM Journal on scientific and statistical computing*, 4(3):553–572, 1983.

[62] N. Nguyen, G. Rozza, D. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for parametrized parabolic pdes: Application to real-time bayesian parameter estimation. *Large-Scale Inverse Problems and Quantification of Uncertainty*, pages 151–177, 2010.

[63] S. Nicaise and F. Tröltzsch. A coupled maxwell integrodifferential model for magnetization processes. *Mathematische Nachrichten*, 287(4):432–452, 2014.

[64] A. Nicolet and F. Delincé. Implicit runge-kutta methods for transient magnetic field computation. *IEEE transactions on Magnetics*, 32(3):1405–1408, 1996.

[65] B. Oksendal. Optimal control of stochastic partial differential equations. *Stochastic analysis and applications*, 23(1):165–179, 2005.

[66] C. Pechstein. *Multigrid-Newton-methods for nonlinear magnetostatic problems.* Master's thesis, 2004.

[67] M. J. Powell. The bobyqa algorithm for bound constrained optimization without derivatives. *Cambridge NA Report NA2009/06, University of Cambridge, Cambridge*, 26, 2009.

[68] R. Rannacher. *Numerik gewöhnlicher Differentialgleichungen.* Heidelberg University Publishing, 2017.

[69] P.-A. Raviart and J.-M. Thomas. Primal hybrid finite element methods for 2nd order elliptic equations. *Mathematics of computation*, 31(138):391–413, 1977.

[70] M. Renardy and R. C. Rogers. *An introduction to partial differential equations*, volume 13. Springer Science & Business Media, 2006.

[71] L. Roberts. *Derivative-free algorithms for nonlinear optimisation problems.* PhD thesis, University of Oxford, 2019.

[72] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229, 2008.

[73] M. Růžička. *Nichtlineare Funktionalanalysis*. Springer, 2020.

[74] E. W. Sachs and S. Volkwein. Pod-galerkin approximations in pde-constrained optimization. *GAMM-Mitteilungen*, 33(2):194–208, 2010.

[75] S. Schöps, H. De Gersem, and T. Weiland. Winding functions in transient magnetoquasistatic field-circuit coupled simulations. *COMPEL: The international journal for computation and mathematics in electrical and electronic engineering*, 2013.

[76] B. Schweizer. *Partielle differentialgleichungen*. Springer, 2013.

[77] R. E. Showalter. *Hilbert space methods in partial differential equations*. Courier Corporation, 2010.

[78] A. Sichau. *Robust nonlinear programming with discretized PDE constraints using second-order approximations*. PhD thesis, TU Darmstadt, 2014.

[79] C. Tischendorf. Coupled systems of differential algebraic and partial differential equations in circuit and device simulation. *Modeling and numerical analysis*, 2003.

[80] F. Tröltzsch. *Optimal control of partial differential equations: theory, methods, and applications*, volume 112. American Mathematical Soc., 2010.

[81] M. Ulbrich and S. Ulbrich. *Nichtlineare Optimierung*. Springer-Verlag, 2012.

[82] S. Volkwein. Proper orthogonal decomposition: Theory and reduced-order modelling. *Lecture Notes*, 2013. URL: `http://www.math.uni-konstanz.de/numerik/personen/volkwein/teaching/POD-Book.pdf`.

[83] B. I. Wohlmuth. A mortar finite element method using dual spaces for the lagrange multiplier. *SIAM journal on numerical analysis*, 38(3):989–1012, 2000.

[84] B. I. Wohlmuth. A comparison of dual lagrange multiplier spaces for mortar finite element discretizations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 36(6):995–1012, 2002.

# Wissenschaftlicher Werdegang

| | |
|---|---|
| 2018–2023 | Wissenschaftlicher Mitarbeiter am Fachbereich Mathematik der Technischen Universität Darmstadt |
| 2016–2018 | Master of Science in Mathematik/Wirtschaftsmathematik an der Technischen Universität Darmstadt |
| 2011–2016 | Bachelor of Science in Mathematik/Wirtschaftsmathematik an der Technischen Universität Darmstadt |