

A Certified Model Reduction Approach for Robust Parameter Optimization with PDE Constraints

Alessandro Alla

*Department of Mathematics, PUC-Rio, R. Mg. S. Vicente 225, Rio de Janeiro 22453-900,
Brazil.*

Michael Hinze

*Department of Mathematics, Universität Hamburg, Bundesstr. 55, 20146 Hamburg,
Germany.*

Philip Kolvenbach

*Department of Mathematics, Technische Universität Darmstadt, Dolivostr. 15, 64293
Darmstadt, Germany.*

Oliver Lass

*Department of Mathematics, Technische Universität Darmstadt, Dolivostr. 15, 64293
Darmstadt, Germany.*

Stefan Ulbrich

*Department of Mathematics, Technische Universität Darmstadt, Dolivostr. 15, 64293
Darmstadt, Germany.*

Abstract

We investigate an optimization problem governed by an elliptic partial differential equation with uncertain parameters. We introduce a robust optimization framework that accounts for uncertain model parameters. The resulting non-linear optimization problem has a bi-level structure due to the min-max formulation. To approximate the worst-case in the optimization problem we propose linear and quadratic approximations. However, this approach still turns out to be very expensive, therefore we propose an adaptive model order reduction technique which avoids long offline stages and provides a certified reduced order surrogate model for the parametrized PDE which is then utilized in the numerical optimization. Numerical results are presented to validate the presented approach.

Email addresses: alla@mat.puc-rio.br (Alessandro Alla),
michael.hinze@uni-hamburg.de (Michael Hinze), kolvenbach@mathematik.tu-darmstadt.de
(Philip Kolvenbach), lass@mathematik.tu-darmstadt.de (Oliver Lass),
ulbrich@mathematik.tu-darmstadt.de (Stefan Ulbrich)

Keywords: model order reduction, parameter optimization, robust optimization, proper orthogonal decomposition
2000 MSC: 35Q93, 49J20, 49K20

1. Introduction

Parameter optimization governed by partial differential equations (PDEs) is a well-studied topic due to its relevance in industrial applications. If the model is considered to be *perfect* one can perform standard techniques to solve these type of problems (see e.g. [20, 39] and the references therein). However, its numerical approximation can be very expensive due to the dimension of the discretized PDE. For this reason, in the last decade, model order reduction techniques were introduced and successfully applied in the context of PDE constrained optimization. Model order reduction works in a Galerkin projection framework, where the basis functions are non-local and built upon information of the underlying system. Although a detailed description of these contributions goes beyond the scope of this paper, we want to mention Proper Orthogonal Decomposition (POD) and the reduced basis (RB) method. The former works in general situations such as time-dependent problems, and parametric steady and unsteady equations (see e.g. [42] for a presentation of the method within different applications). The latter is mainly used in the context of e.g. many-query scenarios (see e.g. [37]) for parametric steady problems where the basis functions are selected by means of a greedy algorithm. It is also possible to combine RB and POD in the so-called POD-Greedy algorithm [18] for parametric unsteady models. The strength of these methods is the presence of an a-posteriori error estimator which certifies the quality of the surrogate models. Model order reduction has been applied successfully to parameter optimization with PDE constraints in e.g. [13, 31]. However, we note that all these works strongly rely on the offline-online decomposition. In the current work, we propose to build the surrogate model towards the optimization. Similar approaches were studied in e.g. [35, 43], but our approach does not require offline-online decomposition and the reduced order model gets updated to desired accuracy during the optimization process.

For the purpose of this work we assume that the PDE which governs our system is given but material imperfections are present. These are due to e.g. manufacturing and introduce uncertainty to the model. This problem arises in many real life applications. One way to include this uncertainty into the optimization process is through robust optimization. In this case no probabilistic model of the uncertainty is required. Instead, a deterministic approach is applied by assuming that the uncertainty is restricted to a bounded uncertainty set. Using the notion of a robust counterpart the associated original uncertain optimization problem is reformulated. The solution obtained in this way stays feasible for all realizations from the uncertainty set and, at the same time, yields the best worst-case value of the objective function. Techniques which resolve the uncertainty by means of stochastic optimization e.g. [9, 25, 30, 40] are

an alternative to the presented approach. These methods depend on sampling the uncertain parameters and hence can become prohibitive expensive in the context of PDE constrained optimization. For a general discussion of robust optimization we refer to e.g. [6, 7, 8].

The idea is to utilize a suitable approximation of the robust counterpart, e.g. [11, 22, 26, 38, 44] or exploit specific properties e.g. [22]. We investigate the approximation of the robust counterpart using a quadratic model. This allows us to reformulate the robust optimization problem as a *mathematical program with equilibrium constraints* (MPEC). This approach has been investigated in [26, 38, 3] in the context of PDE constrained optimization problems, where [26] forms the starting point for the present investigations. Our model will be a linear elliptic parametric equation where an affine decomposition will be applied to work with a reference parameter. The worst-case problem leads a non-linear optimization problem with a min – max formulation. After approximating the inner maximization derivative based optimization is applied. In the setting of our work we will utilize a sensitivity based approach since we assume to deal with only a few parameters. This approach is computationally expensive because it requires the solution of multiple PDEs.

In this work the focus is on solving the robust counterpart in an efficient way utilizing a POD-based reduced order method. More specifically, we propose a certified robust optimization procedure which, following an idea proposed in [1], successively adapts the POD model during the iterative adjustment of the optimal robust design through appropriate enrichment of the snapshot set and avoids expensive offline stages. The method is certified by an a-posteriori error estimator for the state variable and the sensitivities. Therefore, error estimators for the state and the sensitivities are required. A generalized error estimator is derived to cover these needs. Here we build upon the robust optimization framework developed in [26], which for the convenience of the reader is summarized in Section 3, and which we have now complemented with a moving expansion point approach to further improve the accuracy of the utilized approximations, see Section 3.2.3 for details. Our approach to model reduction complements the method proposed in [26]. In fact, in [26] the model reduction in the application is performed by a POD-greedy procedure to reduce the degrees of freedom related to the rotation angle of a quasistatic model of an electrical machine. This POD surrogate model is then used to speed up the robust optimization. In this paper, we generate the reduced order model by applying POD to a snapshot set resulting from state and sensitivities corresponding to previous optimization iterates. The quality of the obtained optimal solution is evaluated by a posteriori error estimators. As long as the error is above a specified level, the snapshot set is updated by the state and sensitivities corresponding to the current iterate of the optimization variables. We demonstrate the performance and efficiency of our approach for robust design of electrical machines, where we use the setting of [26] in the static case, i.e., we do not consider rotation.

The paper is organized as follows: in Section 2 we present the mathematical model and in Section 3 we present the nominal and the robust optimization problem. Section 4 focuses on model order reduction for the optimization prob-

lem and in Section 5 we illustrate the effectiveness of the discussed methods by numerical examples.

2. Elliptic Parametric PDEs

We deal with an abstract linear stationary equation governed by a uniformly continuous and uniformly coercive bilinear form \tilde{a} stemming e.g. from a parametric elliptic PDE. The parameter sets are denoted by $\mathcal{D} \subset \mathbb{R}^{N_p}$ and $\mathcal{U}_k \subset \mathbb{R}^{N_\Phi}$, where parameters $p \in \mathcal{D}$ enter the problem via a p -dependent regular bounded domain $\Omega(p) \subset \mathbb{R}^2$ which give rise to a p -dependent real and separable Hilbert space $X(\Omega(p))$ and dual space $X'(\Omega(p))$. The parameters $\Phi \in \mathcal{U}_k$ will model the uncertainty in the problem. We note that for our abstract setting we impose the by now standard assumptions on parameter separability which are formulated for example in [17]. Those assumptions are in particular met for the application considered in Section 5. Our abstract problem then reads

$$\begin{cases} \text{For } p \in \mathcal{D} \text{ find } \tilde{u} \in X(\Omega(p)) \text{ s.t} \\ \tilde{a}(\tilde{u}, v; p) = \tilde{f}(v; p, \phi), \quad \forall v \in X(\Omega(p)), \end{cases} \quad (1)$$

where \tilde{u} is the unknown variable. In our application uncertainty only enters through the right hand side, where \tilde{a} and \tilde{f} are given by

$$\tilde{a}(w, v; p) = \int_{\Omega(p)} \nu \nabla w \cdot \nabla v \, dx \quad \text{and} \quad \tilde{f}(v; p, \phi) = \int_{\Omega(p)} \tilde{f}(\phi) v \, dx,$$

where $\nu \geq \bar{\nu} > 0$ is an isotropic material coefficient (see Section 3 for more details.). Note that under these assumptions problem (1) admits a unique solution.

The function space $X(\Omega(p))$ is such that $H_0^1(\Omega(p)) \subset X(\Omega(p)) \subset H^1(\Omega(p))$, with

$$H^1(\Omega(p)) := \{f \in L^2(\Omega(p)) : D^\alpha f \in (L^2(\Omega(p)))^2, |\alpha| \leq 1\},$$

where f is a measurable function, $\alpha \in \mathbb{N}_0^2$, $D^\alpha f$ denotes the weak α^{th} -partial derivative of f and the functional spaces $L^2(\Omega(p))$, $H_0^1(\Omega(p))$ are defined as follows:

$$L^2(\Omega(p)) := \left\{ f : \Omega(p) \rightarrow \mathbb{R}, \int_{\Omega(p)} f(x)^2 \, dx < \infty \right\},$$

$$H_0^1(\Omega(p)) := \{f \in H^1(\Omega(p)) : f \equiv 0 \text{ a.e. on } \partial\Omega(p) \text{ in the sense of traces}\}.$$

In order to obtain a computationally fast model and to avoid re-meshing when the parameter changes we require parameter separability, see, e.g., [17], and assume that the domains $\Omega(p)$ in (1) can be transformed to a fixed domain $\Omega(\bar{p}) := \cup_{q=1}^Q \Omega_q(\bar{p})$, where \bar{p} denotes the reference parameter. Without loss of generality, we assume that the domain of interest $\Omega(p)$ can be decomposed in

Q non-overlapping triangles and the transformation on each triangle is linear, whereas piecewise-linear and continuous over the whole domain according to:

$$\begin{aligned} \mathcal{T}_q(x; p) &: \Omega_q(\bar{p}) \rightarrow \Omega(p) \\ x &\mapsto C_q(p)x + d_q(p), \end{aligned} \quad (2)$$

for $q = 1, \dots, Q$, where $C_q(p) \in \mathbb{R}^{2 \times 2}$ and $d_q(p) \in \mathbb{R}^2$. From here onwards we write $\bar{\Omega}$ instead $\Omega(\bar{p})$ whenever this is appropriate.

As it is shown, e.g., in [37], the linear parametric map $\mathcal{T}_q(p)$ and its Jacobian $J_{\mathcal{T}_q}$ allow the definition of the bilinear and linear forms on the reference domain $\bar{\Omega}$. Then, problem (1) can be rewritten as

$$\begin{cases} \text{For } p \in \mathcal{D} \text{ find } u \in X(\bar{\Omega}) \text{ s.t.} \\ a(u, v; p) = f(v; p, \phi), \quad \forall v \in X(\bar{\Omega}), \end{cases} \quad (3)$$

where the assumptions in problem (1) hold true. The bilinear form $a(\cdot, \cdot; p)$ can be expressed with an affine linear decomposition:

$$a(u, v; p) = \sum_{q=1}^{Q_a} \sum_{i,j=1}^2 \hat{\Phi}_q^{i,j}(p) a_q^{i,j}(u, v), \quad (4)$$

such that $\hat{\Phi}_q^{i,j} : \mathcal{D} \rightarrow \mathbb{R}$ for $q = 1, \dots, Q_a, i, j = 1, 2$, is a function depending on p and $a_q^{i,j}$ are the parameter independent continuous bilinear forms. In our example we have

$$a_q^{i,j}(w, v) := \int_{\Omega_q(\bar{p})} \frac{\partial w}{\partial x_i} \frac{\partial v}{\partial x_j} dx, \quad \text{for } q = 1, \dots, Q_a \text{ and } i, j = 1, 2.$$

This allows us to improve the computational efficiency in the evaluation of $a(u, v; p)$; the discrete approximation of the components $a_q^{i,j}(u, v)$ can be then computed once. The same affine decomposition is applied to f with

$$f(v; p, \phi) = \sum_{q=1}^{Q_f} h_q(\phi) \hat{\Phi}_q^f(p) f_q(v) \quad (5)$$

with $h_q(\phi)$ being nonlinear differentiable functions depending on the uncertain parameter and

$$f_q(v) := \int_{\Omega_q(\bar{p})} (f \circ \mathcal{T}_q)v | \det J_{\mathcal{T}_q} | dx, \quad \text{for } q = 1, \dots, Q_f.$$

The assumption that the problem is affine dependent on the parameter p is the key for the purpose of model reduction as we will see in Section 4. We note that without loss of generality, in the present work, we consider the same number of subdomains $Q = Q_a = Q_f$ for the (bi)linear forms a and f .

Furthermore, we define the coercivity constant that will be a key ingredient in the certification of the model reduction algorithm, by

$$\alpha(p) := \inf_{w \in X(\bar{\Omega}), w \neq 0} \frac{a(w, w; p)}{\|w\|_{X(\bar{\Omega})}^2}, \quad (6)$$

and the continuity constant as

$$\gamma(p) := \sup_{v \in X(\bar{\Omega}), v \neq 0} \sup_{w \in X(\bar{\Omega}), w \neq 0} \frac{a(w, v; p)}{\|w\|_{X(\bar{\Omega})} \|v\|_{X(\bar{\Omega})}}. \quad (7)$$

To ease the notations, from here onwards, we drop the dependence on ϕ (e.g. $h_q(\phi) := 1$ in (5)).

For the purpose of the optimization problem we will compute the sensitivities $u_i^1 := \frac{\partial u(p)}{\partial p_i}$ which are obtained by the derivative with respect to the parameters from equation (3) leading the following linear problem:

$$\begin{cases} \text{For } p \in \mathcal{D} \text{ find } u_i^1 \in X(\bar{\Omega}) \text{ s.t.} \\ a(u_i^1, v; p) = \frac{\partial f}{\partial p_i}(v; p) - \frac{\partial a}{\partial p_i}(u, v; p), \quad \forall v \in X(\bar{\Omega}), i = 1, \dots, N_p. \end{cases} \quad (8)$$

We note that due to the affine decomposition the i -th partial derivatives of $a(\cdot, \cdot; p)$ and $f(\cdot; p)$ are given by the derivatives of coefficient functions $\hat{\Phi}_q^{i,j}$ and $\hat{\Phi}_q^f$, $q = 1, \dots, Q$, $i, j = 1, 2$ and can be computed analytically. More generally, for a multiindex $\mu \in \mathbb{N}_0^{N_p}$, $|\mu| = n$, the μ -th sensitivity can be computed by the following proposition.

Proposition 2.1. *Let the coefficient functions $\hat{\Phi}_q^{i,j}(p)$ and $\hat{\Phi}_q^f(p, \phi)$, $q = 1 \dots, Q$, $i, j = 1, 2$ be n -times differentiable with respect to p . Then the solution u is differentiable with respect to p and the sensitivities $u_\mu^n = \frac{\partial^\mu u}{\partial p^\mu} \in X(\bar{\Omega})$, $|\mu| = n$, satisfy the sensitivity equation*

$$a(u_\mu^n, v; p) = \frac{\partial^\mu f}{\partial p^\mu}(v; p) - \sum_{1 \leq |\kappa| \leq n} \binom{\mu}{\kappa} \frac{\partial^\kappa a}{\partial p^\kappa}(u_{\mu-\kappa}^{n-|\kappa|}, v; p) \quad \forall v \in X(\bar{\Omega}), \quad (9)$$

where $\binom{\nu}{\kappa} = \prod_{i=1}^{N_p} \binom{\mu_i}{\kappa_i} = \frac{\mu!}{\kappa!(\mu-\kappa)!}$ denotes the binomial coefficient for multi-indices.

The proof of the proposition follows from the general Leibniz rule for the μ -th derivative applied to (4) and is omitted here. We can easily see that for $n = 0$ we find the state equation (3) and for $n = 1$ the first sensitivity equation (8).

Next, we introduce an high dimensional finite element (FE) discretization of our model problem in the space $X_h(\bar{\Omega}) \subset X(\bar{\Omega})$. The discrete problem then reads

$$\begin{cases} \text{For } p \in \mathcal{D} \text{ find } u_h \in X_h(\bar{\Omega}) \text{ s.t.} \\ a(u_h, v_h; p) = f(v_h; p), \quad \forall v_h \in X_h(\bar{\Omega}). \end{cases} \quad (10)$$

For the discrete problem we use the ansatz $u_h = \sum_{i=1}^N (\mathbf{u}_h)_i \varphi_i$, where φ_i are suitable FE ansatz functions. Problem (10) then is equivalent to the linear system

$$\begin{cases} \text{For } p \in \mathcal{D} \text{ find } \mathbf{u}_h(p) \in \mathbb{R}^N \text{ s.t.} \\ \mathbf{K}(p) \mathbf{u}_h(p) = \mathbf{f}(p), \end{cases} \quad (11)$$

where $\mathbf{K}(p) \in \mathbb{R}^{N \times N}$ is the stiffness matrix $(\mathbf{K}(p))_{ij} = a(\varphi_j, \varphi_i; p)$, $1 \leq i, j \leq N$, and the right hand side $\mathbf{f}(p) \in \mathbb{R}^N$ is obtained by $(\mathbf{f}(p))_i = f(\varphi_i; p)$, $1 \leq i \leq N$. The FE system matrix keeps the dependency on the parameter p and we apply the affine decomposition to it in the following way

$$\mathbf{K}(p) = \sum_{q=1}^Q \sum_{i,j=1}^2 \hat{\Phi}_q^{i,j}(p) \mathbf{K}_q^{i,j}, \quad (12)$$

where $\mathbf{K}_q^{i,j}$, $q = 1, \dots, Q$ and $i, j = 1, 2$, are the system matrices on the Q subdomains. We note again that the p dependency of \mathbf{K} appears only in the weight functions $\hat{\Phi}_q^{i,j}$ which are easy and inexpensive to evaluate. The same applies for the right hand side. Similarly, the discretized version of equation (8) reads

$$\mathbf{K}(p) \mathbf{u}_{h,i}^1(p) = \tilde{\mathbf{f}}_i^1, \quad \text{for } i = 1, \dots, N_p, \quad (13)$$

with

$$\tilde{\mathbf{f}}_i^1 = \frac{\partial \mathbf{f}}{\partial p_i}(p) - \frac{\partial \mathbf{K}}{\partial p_i}(p) \mathbf{u}_h \quad \text{for } i = 1, \dots, N_p,$$

where the subindex i indicates as above the derivative with respect to the i -th parameter p_i . Note that these derivatives are easy to compute due to the previously introduced affine decomposition (12). The derivative of the matrix $\mathbf{K}(p)$ and the vector $\mathbf{f}(p)$ are given by the derivatives of the functions $\hat{\Phi}_q^{i,j}(p)$ and $\hat{\Phi}_q^f(p, \phi)$, $q = 1, \dots, Q$, $i, j = 1, 2$, respectively. For the general sensitivity equations (9) we get with a multiindex $\mu \in \mathbb{N}_0^{N_p}$, $|\mu| = n$,

$$\mathbf{K}(p) \mathbf{u}_{h,\mu}^n = \frac{\partial^\mu \mathbf{f}}{\partial p^\mu}(p) - \sum_{1 \leq |\kappa| \leq n} \binom{\mu}{\kappa} \frac{\partial^\kappa \mathbf{K}}{\partial p^\kappa}(p) \mathbf{u}_{h,\mu-\kappa}^{n-|\kappa|}. \quad (14)$$

In the rest of the work we will continue to use the following compact notation when appropriate:

$$\mathbf{u}_{h,\mu}^n(p) := \frac{\partial^\mu \mathbf{u}_h}{\partial p^\mu}(p)$$

in order to denote the μ -th derivative of \mathbf{u}_h with respect to p for a multiindex $\mu \in \mathbb{N}_0^{N_p}$, $|\mu| = n$. Moreover, we sometimes use the fact that the state is the zeroth order sensitivity, i.e., $\mathbf{u}_{h,\mu}^0(p) = \mathbf{u}_h(p)$ with $|\mu| = 0$. Similarly, we will adopt \mathbf{f}_μ^n and \mathbf{K}_μ^n .

From now on we will focus on the discrete version (11) of the problem (1). All further steps are analogous in the continuous setting.

3. Optimization problem

This section is devoted to the optimization and the robust optimization problem. We will formulate an optimization problem governed by a parametrized PDE, then investigate the nominal optimization and its robust counterpart. By the robust formulation, uncertainties in model parameters are taken into account. This is done by utilizing a worst-case formulation. For an efficient realization, approximation techniques of different orders are investigated. Here we closely follow [26] and repeat the ingredients for the convenience of the reader.

3.1. The nominal optimization

We first introduce the nominal optimization problem. In this setting all model parameters ϕ are fixed to a nominal value $\hat{\phi}$. The optimization problem then reads as:

$$\begin{aligned} & \min_{p \in \mathbb{R}^{N_p}, \mathbf{u}_h \in \mathbb{R}^N} \tilde{g}_0(\mathbf{u}_h, p, \hat{\phi}), \\ & \text{subject to (s.t.) } \tilde{g}_i(\mathbf{u}_h, p, \hat{\phi}) \leq 0, \quad i = 1, \dots, N_g, \\ & e(\mathbf{u}_h, p, \hat{\phi}) = 0, \end{aligned} \quad (15)$$

where $p \in \mathbb{R}^{N_p}$ is the optimization variable and $\hat{\phi} \in \mathcal{U}_k \subset \mathbb{R}^{N_\phi}$ is the fixed model parameter. The function \tilde{g}_0 is the objective function and \tilde{g}_i are the $N_g \in \mathbb{R}$ inequality constraints with $\tilde{g}_i : \mathbb{R}^N \times \mathbb{R}^{N_p} \times \mathbb{R}^{N_\phi} \mapsto \mathbb{R}$ for $i = 0, \dots, N_g$. Furthermore, e is an equality constraint governed by the discretized parametric PDE in (11), i.e., $e(\mathbf{u}_h, p, \hat{\phi}) = 0 \iff K(p, \hat{\phi})\mathbf{u}_h = f(p, \hat{\phi})$. We assume \tilde{g}_i , $i = 0, \dots, N_g$, and e to be continuously differentiable. It directly follows from the coercivity of the bilinear form a that the parametric PDE for every admissible parameter p admits a unique solution \mathbf{u}_h , and that the Jacobian $\frac{\partial e}{\partial \mathbf{u}}$ is boundedly invertible, compare for example [20]. Further, let $e(\mathbf{u}_h, p, \hat{\phi}) = 0$ have a unique solution $\mathbf{u}_h = \mathbf{u}_h(p)$ for every admissible p . Then we can introduce the reduced objective and constraint functions

$$g_i(p, \hat{\phi}) := \tilde{g}_i(\mathbf{u}_h, p, \hat{\phi}) \quad \text{for } i = 0, \dots, N_g.$$

The reduced optimization problem associated with (15) then reads

$$\begin{aligned} & \min_{p \in \mathbb{R}^{N_p}} g_0(p, \hat{\phi}), \\ & \text{s.t. } g_i(p, \hat{\phi}) \leq 0, \quad i = 1, \dots, N_g. \end{aligned} \quad (16)$$

Note that the reduced objective and the reduced constraints $g_i(p)$, $i = 0, \dots, N_g$, are continuously differentiable due to the previous assumptions. Since e is continuously differentiable with invertible Jacobian $\frac{\partial e}{\partial \mathbf{u}}e(\mathbf{u}, p, \hat{\phi})$, the implicit function theorem guarantees that also \mathbf{u}_h is continuously differentiable with respect to the parameter. In the further investigation, we focus on the reduced formulation.

3.2. The robust optimization

After having introduced the nominal optimization problem we will now introduce the robust optimization framework, where the model parameter ϕ is considered to be uncertain. In the presented setting, the uncertain parameter is a model parameter but the framework is not limited to this setting. We will next formulate a robust version of (16). For this, we first have to make some assumptions on the uncertain parameter and introduce the *uncertainty set*

$$\mathcal{U}_k = \left\{ \phi \in \mathbb{R}^{N_\phi} \mid \|D^{-1}(\phi - \hat{\phi})\|_k \leq 1 \right\}, \quad (17)$$

where $\hat{\phi}$ is a nominal value and $D \in \mathbb{R}^{N_\phi \times N_\phi}$ an invertible scaling matrix. We assume that the uncertain parameter remains within the given bounded set \mathcal{U}_k . For different choices of k we get different uncertainty sets. The two most commonly used ones are $k = \{2, \infty\}$. The case $k = \infty$ is a special case and allows a representation using upper and lower bounds $\bar{\phi}, \underline{\phi}$, respectively. For this we set $D = \text{diag}((\bar{\phi} - \underline{\phi})/2)$ and $\hat{\phi} = (\underline{\phi} + \bar{\phi})/2$. The uncertainty set can then be written as

$$\mathcal{U}_\infty = \{ \phi \in \mathbb{R}^{N_\phi} \mid \underline{\phi} \leq \phi \leq \bar{\phi} \}.$$

Next we want to focus on the robust optimization problem utilizing the worst-case formulation [7, 11, 44]. Using the uncertainty set we can define the *worst-case function* as

$$\varphi_i(p) := \max_{\phi \in \mathcal{U}_k} g_i(p, \phi), \quad i = 0, \dots, N_g.$$

For an interpretation, for every fixed parameter p the function $\varphi_i : \mathcal{U}_k \rightarrow \mathbb{R}$ is the worst-case value of the function g_i with $\phi \in \mathcal{U}_k$. Using the definition of the uncertainty set, the worst-case function can be rewritten as

$$\begin{aligned} \varphi_i(p) &:= \max_{\phi \in \mathbb{R}^{N_\phi}} g_i(p, \phi) \\ \text{s.t.} \quad &\|D^{-1}(\phi - \hat{\phi})\|_k \leq 1, \quad i = 0, \dots, N_g. \end{aligned} \quad (18)$$

Using the worst case function we will formulate the robust optimization problem. The *robust counterpart* of (16) is given by

$$\begin{aligned} \min_{p \in \mathbb{R}^{N_p}} \quad &\varphi_0(p) \\ \text{s.t.} \quad &\varphi_i(p) \leq 0, \quad i = 1, \dots, N_g. \end{aligned} \quad (19)$$

A solution p of (19) is robust against uncertainties in the parameter ϕ and is hence referred to as *robust optimal solution*. Note that the solution is feasible for (16) for all $\phi \in \mathcal{U}_k$ and optimal with respect to the objective function.

The introduced robust optimization problem (19) is of bi-level structure and hence difficult to solve. Thus it is required to develop tailored methods to solve the problem. For general nonlinear problems in [7], it is proposed to employ approximations. The *approximated robust counterpart* of (19) is then introduced as

$$\begin{aligned} \min_{p \in \mathbb{R}^{N_p}} \quad &\hat{\varphi}_0(p) \\ \text{s.t.} \quad &\hat{\varphi}_i(p) \leq 0, \quad i = 1, \dots, N_g, \end{aligned} \quad (20)$$

where $\hat{\varphi}_i$ can be computed more efficiently compared to (18) and is referred to as the *approximated worst-case function*. First and second order approximations $\hat{\varphi}_i$ of the worst-case function φ_i will be considered. The first order approximation for the general nonlinear case was investigated in [11, 44] while the second order approximation is a recent approach [26] and can be seen as a modification of [38].

3.2.1. Linear approximation of the robust counterpart

In the first order approach suggested in [11, 44], a linearization of the worst-case function is carried out. For this a nominal value $\hat{\phi}$ for the uncertain parameter is chosen and the approximated worst-case function is then given by the first order Taylor expansion

$$\begin{aligned} \hat{\varphi}_i(p) := & \max_{\delta_i \in \mathbb{R}^{N_\phi}} g_i(p, \hat{\phi}) + \nabla_{\phi} g_i(p, \hat{\phi})^\top \delta_i \\ \text{s.t.} & \|D^{-1} \delta_i\|_k \leq 1, \quad i = 0, \dots, N_g, \end{aligned}$$

where $\delta_i = \phi - \hat{\phi}$. The solution of the resulting convex optimization problem can be given analytically in the form

$$\hat{\varphi}_i(p) = g_i(p, \hat{\phi}) + \|D \nabla_{\phi} g_i(p, \hat{\phi})\|_{k^*}$$

with $\|D \cdot\|_{k^*}$ the dual norm of $\|D^{-1} \cdot\|_k$ for $k^* = k/(k-1)$ with $k^* = 1$ for $k = \infty$. Note that this is a standard result for scaled Hölder norms. Utilizing these results, the linear approximated robust counterpart reads as

$$\begin{aligned} \min_{p \in \mathbb{R}^{N_p}} & g_0(p, \hat{\phi}) + \|D \nabla_{\phi} g_0(p, \hat{\phi})\|_{k^*} \\ \text{s.t.} & g_i(p, \hat{\phi}) + \|D \nabla_{\phi} g_i(p, \hat{\phi})\|_{k^*} \leq 0, \quad i = 1, \dots, N_g. \end{aligned} \quad (21)$$

We refer to (21) as the *linear approximated robust counterpart* of (16). Due to the norm the objective function and the inequality constraints are non-differentiable if the term inside the norm becomes zero. For the case $k = \infty$, the remedy is to introduce slack variables, i.e.,

$$\begin{aligned} \min_{p \in \mathbb{R}^{N_p}, \zeta_i \in \mathbb{R}} & g_0(p, \hat{\phi}) + \zeta_0 \\ \text{s.t.} & g_i(p, \hat{\phi}) + \zeta_i \leq 0, \quad i = 1, \dots, N_g, \\ & -\zeta_i \leq D \nabla_{\phi} g_i(p, \hat{\phi}) \leq \zeta_i, \quad i = 0, \dots, N_g. \end{aligned}$$

For the case $k = 2$, the problem (21) can be reformulated by using second order cone constraints that can be handled efficiently for example by interior point techniques. But usually the solution of (21) is sufficiently far away from points of nonsmoothness and in practice standard algorithms for smooth nonlinear problems work well without any modification. Nevertheless, one has to verify that the robust optimal solution does not lie in a non-differentiable point [11, 38].

The required derivatives for the approximations in this approach and the optimization can be computed using either the adjoint or the sensitivity approach

[20]. Note that the adjoint approach is desirable when the number of uncertain parameters is large. For a detailed discussion about the different approaches we refer the reader to [11].

While computationally attractive, the first order approximation can suffer from an inaccurate approximation. Hence the influence of the uncertain parameter ϕ might be described insufficiently, which was already observed in [12]. This leads to the development of second order methods [26, 38].

3.2.2. Quadratic approximation of the robust counterpart

In the second order approach, a quadratic approximation of the worst case function is utilized. This idea has been investigated in [26, 38]. We will apply the method directly to the reduced problem (16). Since this approach is recent, we will give a short overview of the strategy.

The second order approximation is generated using the second order Taylor expansion of $g_i(p, \cdot)$ around $\hat{\phi}$, i.e.,

$$g_i(p, \hat{\phi} + \delta_i) \approx q_i(p, \hat{\phi}, \delta_i) := \alpha_i(p, \hat{\phi}) + c_i(p, \hat{\phi})^\top \delta_i + \frac{1}{2} \delta_i^\top H_i(p, \hat{\phi}) \delta_i, \quad (22)$$

where $\alpha_i(p, \hat{\phi}) = g_i(p, \hat{\phi})$, $c_i(p, \hat{\phi}) = \nabla_{\phi} g_i(p, \hat{\phi})$, and $H_i(p, \hat{\phi}) = \nabla_{\phi\phi} g_i(p, \hat{\phi})$. As before, we have set $\delta_i = \phi - \hat{\phi}$. The corresponding approximated worst-case function is

$$\hat{\phi}_i(p) := \max_{\delta_i \in \mathbb{R}^{N_\phi}} q_i(p, \hat{\phi}, \delta_i) \quad \text{s.t.} \quad \|D^{-1} \delta_i\|_k \leq 1, \quad i = 0, \dots, N_g. \quad (23)$$

For the quadratic approximation we do not have a closed solution as in the linear case. Fortunately, the problem exhibits a well known structure. In the case $k = 2$, the problem corresponds to a trust region problem and is well-studied. The exact solutions of (23) are characterized as follows.

Theorem 3.1. *For given p , the vector $\delta_i \in \mathbb{R}^{N_\phi}$ is a global solution of the trust region problem (23) if and only if there exists a Lagrange multiplier $\lambda_i \in \mathbb{R}$ satisfying*

$$\lambda_i \geq 0, \quad \|D^{-1} \delta_i\|_2 \leq 1, \quad \lambda_i (\|D^{-1} \delta_i\|_2 - 1) = 0,$$

$$(-H_i(p, \hat{\phi}) + \lambda_i \mathbb{D}) \delta_i = c_i(p, \hat{\phi})$$

with $-H(p, \hat{\phi}) + \lambda_i \mathbb{D}$ positive semidefinite and $\mathbb{D} = D^{-\top} D^{-1}$.

The proof of this theorem can be found in [10, 20]. By adding a square to the norms in the constraints, we obtain the equivalent but differentiable formulation

of the *quadratic approximated robust counterpart* of (16) by

$$\begin{aligned}
\min \quad & \alpha_0(p, \hat{\phi}) + c_0(p, \hat{\phi})^\top \delta_0 + \frac{1}{2} \delta_0^\top H_0(p, \hat{\phi}) \delta_0 \\
\text{s.t.} \quad & \alpha_i(p, \hat{\phi}) + c_i(p, \hat{\phi})^\top \delta_i + \frac{1}{2} \delta_i^\top H_i(p, \hat{\phi}) \delta_i \leq 0, \quad i = 1, \dots, N_g, \\
& \begin{pmatrix} -c_i(p, \hat{\phi}) - H_i(p, \hat{\phi}) \delta_i + \lambda_i \mathbb{D} \delta_i \\ \lambda_i (\|D^{-1} \delta_i\|_2^2 - 1) \end{pmatrix} = 0, \quad i = 0, \dots, N_g, \\
& \|D^{-1} \delta_i\|_2^2 - 1 \leq 0, \quad i = 0, \dots, N_g, \\
& -\lambda_i \leq 0, \quad i = 0, \dots, N_g, \\
& H_i(p, \hat{\phi}) - \lambda_i \mathbb{D} \preceq 0, \quad i = 0, \dots, N_g,
\end{aligned} \tag{24}$$

where $A \preceq 0$ denotes that A is a negative semidefinite matrix. The optimization variables are $p \in \mathbb{R}^{N_p}$, $\delta_0, \dots, \delta_{N_g} \in \mathbb{R}^{N_\phi}$, and $\lambda_0, \dots, \lambda_{N_g} \in \mathbb{R}$. The semidefinite constraint can be reformulated using the smallest eigenvalues as was outlined in [38] or be treated, e.g., by interior point techniques. Problem (24) is a *mathematical program with equilibrium constraints* (MPEC). To solve these problems numerically one has to pay attention to the complementarity constraint. It turns out that SQP method under relatively mild assumptions and few modifications [15, 27] can solve these type of problems efficiently. In [26] the required changes are outlined in details for the given problem.

The computation of the derivatives required for the quadratic approximation can be done by different approaches. Depending on the number of parameters, the sensitivity or the adjoint approach should be chosen. In this work, it is assumed that the number of uncertain parameters is small and the sensitivity based approach is utilized.

3.2.3. Improving the approximation by moving the expansion point

The quadratic approximation is often a notable improvement over the linear approximation. Nonetheless, Taylor approximations generally are accurate only locally, that is, in a neighborhood of the expansion point $\hat{\phi}$, and so the second-order model can sometimes be a poor predictor for the effects of the uncertain parameters, especially when the uncertainty set is relatively large and the expansion point is far away from the worst case. In these cases, the accuracy can be improved by moving the expansion point of the Taylor approximation towards a presumed maximum of the modeled function. This can be implemented with the aid of an additional set of variables $\hat{\phi}_i \in \mathbb{R}^{N_\phi}$, $i = 0, \dots, N_g$, for the individual expansion points of the models. The second-order Taylor expansion of $g_i(p, \cdot)$ around $\hat{\phi}_i$ evaluated near the center $\hat{\phi}$ of the uncertainty set is given by

$$\begin{aligned}
g_i(p, \hat{\phi} + \delta_i) &\approx q_i^s(p, \hat{\phi}_i, \delta_i) := g_i(p, \hat{\phi}_i) + \nabla_{\phi} g_i(p, \hat{\phi}_i)^\top (\delta_i - \hat{\delta}_i) \\
&\quad + \frac{1}{2} (\delta_i - \hat{\delta}_i)^\top \nabla_{\phi\phi} g_i(p, \hat{\phi}_i) (\delta_i - \hat{\delta}_i) \\
&= \alpha_i^s(p, \hat{\phi}_i) + c_i^s(p, \hat{\phi}_i)^\top \delta_i + \frac{1}{2} \delta_i^\top H_i(p, \hat{\phi}_i) \delta_i,
\end{aligned}$$

where $\hat{\delta}_i = \hat{\phi}_i - \hat{\phi}$ as well as $H_i(p, \hat{\phi}_i) = \nabla_{\phi\phi} g_i(p, \hat{\phi}_i)$, and additionally

$$\begin{aligned}\alpha_i^s(p, \hat{\phi}_i) &= g_i(p, \hat{\phi}_i) - \nabla_{\phi} g_i(p, \hat{\phi}_i)^\top \hat{\delta}_i, \\ c_i^s(p, \hat{\phi}_i) &= \nabla_{\phi} g_i(p, \hat{\phi}_i) - \nabla_{\phi\phi} g_i(p, \hat{\phi}_i) \hat{\delta}_i.\end{aligned}$$

Since a good choice for $\hat{\phi}_i$ is often not known a priori, it can be adaptively chosen in the course of the optimization. A schematic algorithm that does so is stated in Algorithm 1, which is based on a basic version of the SQP method by Powell, see [34]. The algorithm generates a sequence of optimization variables

Algorithm 1 (Schematic adaptive SQP algorithm)

Require: Initial guess $(p^0, (\delta_i^0)_i, (\lambda_i^0)_i)$ and $(\hat{\phi}_i^0)_i$, penalty parameter $\rho > 0$

- 1: **while** termination criterion not met **do**
 - 2: Solve a quadratic program with expansion points $(\hat{\phi}_i^k)_i$ to obtain update $(\Delta p^k, (\Delta \delta_i^k)_i, (\Delta \lambda_i^k)_i)$
 - 3: Determine step length α_k to achieve proper descent in P_ρ^1
 - 4: Set $(p^{k+1}, (\delta_i^{k+1})_i, (\lambda_i^{k+1})_i) = (p^k, (\delta_i^k)_i, (\lambda_i^k)_i) + \alpha_k \cdot (\Delta p^k, (\Delta \delta_i^k)_i, (\Delta \lambda_i^k)_i)$
 - 5: Determine update $\Delta \hat{\phi}_i^k$ to increase accuracy of $q_i^s(p^{k+1}, \hat{\phi}_i^k, \cdot)$, $i = 0, \dots, N_g$
 - 6: Set $\hat{\phi}_i^{k+1} = \hat{\phi}_i^k + \Delta \hat{\phi}_i^k$, $i = 0, \dots, N_g$
 - 7: $k \leftarrow k + 1$
 - 8: **end while**
-

$x^k := (p^k, (\delta_i^k)_i, (\lambda_i^k)_i)$ and a sequence of expansion points $\hat{\Phi}^k := ((\hat{\phi}_i^k)_i)$, where $k \in \mathbb{N}$ is the iteration counter. The feasibility is eventually enforced with an exact ℓ_1 -penalty function, denoted here by $P_\rho^1(x; \hat{\Phi})$, with a penalty parameter $\rho > 0$. Recall that, if we write (24) in the form $\min_x \{ f(x; \hat{\Phi}) : h_{\leq}(x; \hat{\Phi}) \leq 0, h_{=} (x; \hat{\Phi}) = 0 \}$ for brevity, the penalty function is defined by

$$P_\rho^1(x; \hat{\Phi}) = f(x; \hat{\Phi}) + \rho \cdot \|\max\{0, h_{\leq}(x; \hat{\Phi})\}\|_1 + \rho \cdot \|h_{=} (x; \hat{\Phi})\|_1,$$

where the maximum is to be understood componentwise, and $\|\cdot\|_1$ denotes the 1-norm. We assume that a sufficiently large ρ is known in advance, but in practise it is usually chosen dynamically.

The quadratic program in line 2 is obtained from (24) at the current iterate in the standard way, i.e., by linearization of the constraints and by a quadratic model of the objective function with a positive definite approximation of the Hessian of the associated Lagrangian. The novelty is that the problem functions of (24) are changed between iterations by replacing $\alpha_i(p, \hat{\phi})$, $c_i(p, \hat{\phi})$ and $H_i(p, \hat{\phi})$ by $\alpha_i^s(p, \hat{\phi}_i^k)$, $c_i^s(p, \hat{\phi}_i^k)$ and $H_i(p, \hat{\phi}_i^k)$ before line 2.

There is some freedom in how the updates $\Delta \hat{\phi}_i^k$ are determined. One option is to perform a single step of a projected gradient method to maximize $g_i(p^k, \hat{\phi}_i^k + \cdot)$. In this case, we choose $\hat{\phi}_i^{k+1}$ to be the projection of $\hat{\phi}_i^k + \sigma_k \nabla_{\phi} g_i(p^k, \hat{\phi}_i^k)$ onto the uncertainty set, where $\sigma_k > 0$ is an adequate step length. For details,

we refer to Section 2.2.2 of [20]. This strategy tries to increase the accuracy of the quadratic models by gradually moving the expansion points towards a local maximum of the modeled functions.

Since we change the MPEC formulation (24) in between two SQP iterations, standard convergence results do not apply. The major issue is that we cannot use the common monotonicity argument about the sequence of penalty function values, which is used to show that the penalty function values converge. A simple way to fix this is given in the next lemma.

Lemma 3.2. *Assume that the sequence of expansion points $(\hat{\Phi}^k)_k$ converges as $k \rightarrow \infty$. Furthermore, assume that*

$$\left| \sum_{k=0}^{\infty} P_{\rho}^1(x^{k+1}; \hat{\Phi}^{k+1}) - P_{\rho}^1(x^{k+1}; \hat{\Phi}^k) \right| < \infty.$$

If the sequence $(x^k)_k$ has an accumulation point, then

$$\lim_{k \rightarrow \infty} P_{\rho}^1(x^{k+1}; \hat{\Phi}^{k+1}) - P_{\rho}^1(x^k; \hat{\Phi}^k) = 0.$$

Proof. Let x^* be an accumulation point of $(x^k)_k$ and let $K \subseteq \mathbb{N}$ be an infinite subset such that $x^k \xrightarrow{K} x^*$. The continuity of P_{ρ}^1 implies that $(P_{\rho}^1(x^k; \hat{\Phi}^k))_k$ is a Cauchy sequence. Hence, for any $\varepsilon > 0$ there is some $N(\varepsilon) \in \mathbb{N}$ such that

$$|P_{\rho}^1(x^{k_2}; \hat{\Phi}^{k_2}) - P_{\rho}^1(x^{k_1}; \hat{\Phi}^{k_1})| < \varepsilon, \quad \text{for all } k_1, k_2 \in K, k_2 \geq k_1 \geq N(\varepsilon).$$

Noting $P_{\rho}^1(x^{k_2}; \hat{\Phi}^{k_2}) - P_{\rho}^1(x^{k_1}; \hat{\Phi}^{k_1}) = \sum_{k=k_1}^{k_2-1} P_{\rho}^1(x^{k+1}; \hat{\Phi}^{k+1}) - P_{\rho}^1(x^k; \hat{\Phi}^k)$, we obtain

$$\left| \sum_{k=k_1}^{k_2-1} P_{\rho}^1(x^{k+1}; \hat{\Phi}^{k+1}) - P_{\rho}^1(x^{k+1}; \hat{\Phi}^k) + P_{\rho}^1(x^{k+1}; \hat{\Phi}^k) - P_{\rho}^1(x^k; \hat{\Phi}^k) \right| < \varepsilon.$$

Since $P_{\rho}^1(x^{k+1}; \hat{\Phi}^k) - P_{\rho}^1(x^k; \hat{\Phi}^k) \leq 0$ by line 3 of Algorithm 1, the summability assumption shows that $0 > \sum_{k=k_1}^{k_2-1} P_{\rho}^1(x^{k+1}; \hat{\Phi}^k) - P_{\rho}^1(x^k; \hat{\Phi}^k) > -\infty$ as $k_2 \xrightarrow{K} \infty$. This implies $P_{\rho}^1(x^{k+1}; \hat{\Phi}^k) - P_{\rho}^1(x^k; \hat{\Phi}^k) \rightarrow 0$ for $k \rightarrow \infty$, from which the claim follows. \square

The lemma can be used to apply standard convergence results to Algorithm 1. For example, we can argue analogously to the proof of Theorem 1 in [34] in order to show that every accumulation point of $(x^k)_k$ is a KKT point of (24), under the additional, usual assumptions of SQP methods like uniform boundedness of the Hessian approximations of the Lagrangian. In this proof, the existence of a limit of the sequence $(P_{\rho}^1(x^k; \hat{\Phi}^k))_k$, which is established by Lemma 3.2, implies that every accumulation point of $(x^k)_k$ is feasible. The rest of the proof is simple to modify, so that we skip the details.

4. Proper Orthogonal Decomposition for Parametrized problems

Numerical approximation for robust optimization problems can be expensive since it involves the solution of several PDEs. Furthermore, the sensitivity approach enlarges the number of PDEs and it increases the computational costs of its approximation. For this reason, in this section, we introduce a model order reduction technique to reduce the complexity of the problem.

Here, we focus on the POD method for the approximate solution of the parametrized equation (11). As before, $\mathbf{u}_h(p) \in \mathbb{R}^N$ is the model vector associated to the FE solution of (11) for a given parameter $p \in \mathcal{D} \subset \mathbb{R}^{N_p}$ and $\mathbf{u}_{h,\mu}^n(p) \in \mathbb{R}^N$ are the sensitivities of order n according to (14). For this purpose let $\{p^j\}_{j=1}^m$ be some points in \mathcal{D} and let $\mathbf{u}_h(p^j)$, $\mathbf{u}_{h,\mu}^n(p^j)$ denote the corresponding solutions to (11), (14) for p^j . We define the snapshot set $Y \in \mathbb{R}^{n \times (mn_{max}+1)}$

$$Y := \text{span} \left\{ \left\{ \mathbf{u}_h(p^j), (\mathbf{u}_{h,\mu}^n(p^j))_{1 \leq |\mu|=n \leq n_{max}} \right\}_{1 \leq j \leq m} \right\}$$

including the states and the sensitivities up to order n_{max} and determine a POD reduced space $\mathcal{V}^\ell := \text{span}\{\psi_1, \dots, \psi_\ell\}$ of rank ℓ by solving the following minimization problem

$$\begin{aligned} \min_{\psi_1, \dots, \psi_\ell} \sum_{n=0}^{n_{max}} \sum_{|\mu|=n} \sum_{j=1}^m \beta_j \left\| \mathbf{u}_{h,\mu}^n(p^j) - \sum_{i=1}^{\ell} \langle \mathbf{u}_{h,\mu}^n(p^j), \psi_i \rangle_{\mathbf{W}} \psi_i \right\|_{\mathbf{W}}^2 \\ \text{s.t. } \langle \psi_j, \psi_i \rangle_{\mathbf{W}} = \delta_{ij} \quad \text{for } 1 \leq i, j \leq \ell, \end{aligned} \quad (25)$$

where we set $\mathbf{u}_{h,\mu}^0(p^j) = \mathbf{u}_h(p)$ with $|\mu| = 0$ for brevity, n_{max} is the maximum order of considered sensitivities, β_j are nonnegative weights, δ_{ij} denotes the Kronecker symbol, \mathbf{W} is a symmetric positive definite $N \times N$ matrix and $\psi_i \in \mathbb{R}^N$. The weighted inner product used is defined as follows: $\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{W}} := \mathbf{u}^\top \mathbf{W} \mathbf{v}$.

It is well-known (see [16]) that problem (25) admits a unique solution $\{\psi_1, \dots, \psi_\ell\}$, where ψ_i denotes the i -th eigenvector of the self-adjoint linear operator $\mathcal{R} : \mathbb{R}^n \rightarrow \mathbb{R}^n$, i.e., $\mathcal{R}\psi_i = \lambda_i \psi_i$ with $\lambda_i \in \mathbb{R}$ non-negative, where \mathcal{R} is defined as

$$\mathcal{R}\psi = \sum_{k=1}^{n_{max}} \sum_{|\mu|=n} \sum_{j=1}^m \beta_j \langle \mathbf{u}_{h,\mu}^k(p^j), \psi \rangle_{\mathbf{W}} \mathbf{u}_{h,\mu}^k(p^j) \quad \text{for } \psi \in \mathbb{R}^n.$$

Furthermore, the error in (25) can be expressed as

$$\sum_{n=0}^{n_{max}} \sum_{|\mu|=n} \sum_{j=1}^m \beta_j \left\| \mathbf{u}_{h,\mu}^n(p^j) - \sum_{i=1}^{\ell} \langle \mathbf{u}_{h,\mu}^n(p^j), \psi_i \rangle_{\mathbf{W}} \psi_i \right\|_{\mathbf{W}}^2 = \sum_{i=\ell+1}^d \lambda_i, \quad (26)$$

where d is the rank of the snapshots matrix Y .

4.1. POD approximation for state and sensitivities

We briefly recall how to generate the reduced order modeling by means of POD. Suppose we have computed the POD basis $\{\psi_i, \dots, \psi_\ell\}$ of rank ℓ according to the minimization problem (25). For the weight matrix we choose

$$\mathbf{W} := \mathbf{K}(\bar{p}) + \mathbf{M}(\bar{p}),$$

where \bar{p} is a fixed reference parameter and \mathbf{M} denotes the mass matrix. Then, \mathbf{W} is the matrix associated to the discrete H^1 -norm. We define the POD ansatz for the state as $\mathbf{u}_h^\ell(p) := \sum_{i=1}^{\ell} (\bar{\mathbf{u}}^\ell)_i \psi_i$. This ansatz in (11) leads to an ℓ -dimensional linear system for the unknowns $\{\bar{\mathbf{u}}_i\}_{i=1}^{\ell}$, namely

$$\mathbf{K}^\ell(p) \bar{\mathbf{u}}(p) = \mathbf{f}^\ell(p). \quad (27)$$

Here the entries of the stiffness matrix are given by $(\mathbf{K}^\ell)_{ij} = \psi_j^\top \mathbf{K}(p) \psi_i$ for $1 \leq i, j \leq \ell$. The right hand side is given by $(\mathbf{f}^\ell)_i = \psi_i^\top \mathbf{f}(p)$, $1 \leq i \leq \ell$. We recall that due to the previously introduced affine decomposition this projection has to be computed only once and the system matrix can be written as

$$\mathbf{K}^\ell(p) = \sum_{q=1}^Q \hat{\Phi}_q^a(p) \mathbf{K}^{q,\ell},$$

where $(\mathbf{K}^{q,\ell})_{ij} = \psi_i^\top \mathbf{K}^q \psi_j$ for $1 \leq i, j \leq \ell$ and $q = 1, \dots, Q$. The same structure can be used for the right hand side. Note that this is very important in order to obtain an efficient reduced order model, since the system can be set up for different values of p without the need of the original high dimensional matrices and right hand sides. The reduced order state equation reads:

$$\begin{cases} \text{For } p \in \mathcal{D} \text{ find } \mathbf{u}_h^\ell \in \mathbb{R}^\ell \text{ s.t.} \\ \mathbf{K}^\ell(p) \mathbf{u}_h^\ell(p) = \mathbf{f}^\ell(p). \end{cases} \quad (28)$$

In an analogous way we obtain the general reduced sensitivity equation from (14). We need to make an ansatz for the sensitivities $\mathbf{u}_{h,\mu}^{n,\ell}$ and project the system onto the subspace spanned by the POD basis. In the present work we use the same basis functions for the state and the sensitivity variables and to achieve a better approximation property of the reduced order sensitivities we add the solution of the sensitivity equation to the snapshots set. Note that then the stiffness matrix in the reduced sensitivity equation is the same as in the reduced state equation, so that only the right-hand side needs to be projected.

4.2. A-posteriori error estimations

A-posteriori error estimators have a crucial role in model order reduction. They provide a certification of the surrogate model without the need of the computation of the truth solution. In the present work, we consider as truth

solution the finite element approximation. If the transformation on a subdomain is given by (2), the bilinear form reads

$$a(u, v; p) = \sum_{q=1}^Q \sum_{i,j=1}^2 [(C_q(p))^{-1} \nu_q (C_q(p))^{-T}]_{ij} |\det C_q(p)| \int_{\Omega_q(\bar{p})} \frac{\partial u}{\partial x_i} \frac{\partial v}{\partial x_j} dx,$$

and a lower bound for the coercivity constant $\alpha(p)$ in (6) can be computed from the following estimate

$$a(v, v; p) \geq \min_q \{ |\det(C_q(p))| \lambda_{\min}(C_q(p)^{-1} \nu_q C_q(p)^{-T}) \} a(v, v; \bar{p}), \quad (29)$$

where λ_{\min} denotes the smallest eigenvalue of the matrix $(C_q(p)^{-1} \nu_q C_q(p)^{-T})$ and \bar{p} is the fixed reference parameter. An upper bound for the continuity constant $\gamma(p)$ in (7) can be computed with the same approach.

Then, we can derive an error bound for the reduced state and sensitivity equations (see [37] for more details) in terms of the reduced residual of the aforementioned equations.

For this purpose we define the residual for equation (14) as follows:

Definition 4.1. Let $u_{h,\mu}^{n,\ell} \in X_h(\bar{\Omega})$, $n = 0, \dots, n_{max}$, be the reduced order solution of (14). We define the residual

$$r_{u_\mu^n}(v; p) := \frac{\partial^\mu f}{\partial p^\mu}(v; p) - \sum_{1 \leq |\kappa| \leq n} \binom{\mu}{\kappa} \frac{\partial^\kappa a}{\partial p^\kappa}(u_{h,\mu-\kappa}^{n-|\kappa|,\ell}, v; p) - a(u_{h,\mu}^{n,\ell}, v; p). \quad (30)$$

Then, we have

Theorem 4.2. Let $u_{h,\mu}^n \in X_h(\bar{\Omega})$ be the solution to (14) for $|\mu| = n$ and $u_{h,\mu}^{n,\ell} \in X_h(\bar{\Omega})$ be the corresponding reduced solution of (14). Then, the following inequality holds

$$\|u_{h,\mu}^n - u_{h,\mu}^{n,\ell}\|_{X_h(\bar{\Omega})} \leq \Delta_{u_\mu^n}^\ell(p) := \frac{1}{\alpha(p)} \left(\|r_{u_\mu^n}(p)\|_{(X_h(\bar{\Omega}))'} + \sum_{1 \leq |\kappa| \leq n} \binom{\mu}{\kappa} \gamma_\kappa \Delta_{u_{\mu-\kappa}^{n-|\kappa|}}^\ell(p) \right), \quad (31)$$

where γ_κ is the continuity constant of the κ -th derivative of the coercive bilinear form.

Proof. We denote the error by $e_{u_\mu^n} := u_{h,\mu}^n - u_{h,\mu}^{n,\ell}$. With (9) we find that

$$\begin{aligned} a(e_{u_{h,\mu}^n}, v; p) &= a(u_{h,\mu}^n, v; p) - a(u_{h,\mu}^{n,\ell}, v; p) \\ &= \frac{\partial^\mu f}{\partial p^\mu}(v; p) - \sum_{1 \leq |\kappa| \leq n} \binom{\mu}{\kappa} \frac{\partial^\kappa a}{\partial p^\kappa}(u_{h,\mu-\kappa}^{n-|\kappa|}, v; p) - a(u_{h,\mu}^{n,\ell}, v; p) \\ &= r_{u_\mu^n}(v; p) + \sum_{1 \leq |\kappa| \leq n} \binom{\mu}{\kappa} \frac{\partial^\kappa a}{\partial p^\kappa}(u_{h,\mu-\kappa}^{n-|\kappa|,\ell} - u_{h,\mu-\kappa}^{n-|\kappa|}, v; p), \end{aligned}$$

Now we set $v = e_{u^n}$ and obtain

$$a(e_{u_{h,\mu}^n}, e_{u_{h,\mu}^n}; p) = r_{u_\mu^n}(e_{u_{h,\mu}^n}; p) + \sum_{1 \leq |\kappa| \leq n} \binom{\mu}{\kappa} \frac{\partial^\kappa a}{\partial p^\kappa}(u_{h,\mu-\kappa}^{n-|\kappa|,\ell} - u_{h,\mu-\kappa}^{n-|\kappa|}, e_{u_{h,\mu}^n}; p)$$

By applying Cauchy-Schwarz and using the coercivity of a as well as the continuity of $\frac{\partial^\kappa a}{\partial p^\kappa}$ we find

$$\begin{aligned} \alpha(p) \|e_{u_{h,\mu}^n}\|_{X_h(\bar{\Omega})}^2 &\leq \|r_{u_\mu^n}(p)\|_{(X_h(\bar{\Omega}))'} \|e_{u_{h,\mu}^n}\|_{X_h(\bar{\Omega})} \\ &+ \sum_{1 \leq |\kappa| \leq n} \binom{\mu}{\kappa} \gamma_\kappa \|u_{h,\mu-\kappa}^{n-|\kappa|,\ell} - u_{h,\mu-\kappa}^{n-|\kappa|}\|_{X_h(\bar{\Omega})} \|e_{u_{h,\mu}^n}\|_{X_h(\bar{\Omega})}. \end{aligned}$$

Dividing by $\alpha(p) \|e_{u_{h,\mu}^n}\|_{X_h(\bar{\Omega})}$ leads to (31) first for $n = 0$ and by inductively using the bound $\|u_{h,\mu-\kappa}^{n-|\kappa|,\ell} - u_{h,\mu-\kappa}^{n-|\kappa|}\|_{X_h(\bar{\Omega})} \leq \Delta_{u_{\mu-\kappa}^{n-|\kappa|}}^\ell$ for $n \leq n_{max}$. \square

Remark 4.1. We note that (31) is a generalization of well-known error bounds for state and first order sensitivity equations, compare e.g. [31], and [13] for time-dependent problems.

Remark 4.2. If only the linear form f depends on the parameter p , i.e., $a(u, v) = f(v; p)$, the error estimator (31) reads

$$\|u_{h,\mu}^n - u_{h,\mu}^{n,\ell}\|_{X_h(\bar{\Omega})} \leq \frac{\|r_{u_\mu^n}(p)\|_{(X_h(\bar{\Omega}))'}}{\alpha}.$$

We note that the solution $u(\mu)$ in this case lives in a Q_f -dimensional linear subspace, where Q_f is defined in (5), see e.g. [17]. Then, for $\ell \geq Q_f$ the error with the reduced model is zero and no error bound is needed. We note that sensitivity equations and error estimators for the uncertainties ϕ are analogous to (30)-(31) and are left to the reader.

4.3. The POD method for optimization problem

In this section we explain how to solve the parametrized optimization problem. In our construction the POD spaces depend on the parameter p . Since the solution to the parameter optimization problem is not known in advance the POD space has to be adapted/enriched during the parameter optimization procedure. To achieve this goal we here propose a certified extension of the approach suggested in [1] to our robust setting, where the error bound, $\Delta_{u_\mu^n}^\ell(p)$ introduced in Section 4.2, helps in the selection of the snapshot sets. The algorithm works as follows: we start with a very coarse parameter sample choosing only one parameter p^0 and solve the full problem together with the sensitivity equations associated to this parameter. Then, we compute the POD basis functions and perform the reduced optimization procedure. At the end of the process we find a new parameter p^1 which is an approximation of the optimal desired design, we update the parameter set $\mathcal{D} = \{p^0, p^1\}$, solve the full problem

and the sensitivity equations related to the new parameter p^1 . Then, we enlarge the snapshots set and compute new POD basis functions. We iterate this process until the stopping criterion is reached. The procedure is summarized in Algorithm 2.

Algorithm 2 (Adaptive POD optimization)

Require: $p^0, tol > 0$,

- 1: Set $k = 0, \mathcal{V} = []$,
- 2: Set Snapshot set

$$\mathcal{V} = [\mathcal{V}, \{\mathbf{u}_h(p^k), (\mathbf{u}_{h,\mu}^n(p^k))_{1 \leq |\mu|=n \leq n_{max}}\}],$$

- 3: Compute POD basis functions $\{\psi_i\}_{i=1}^\ell$ with $\ell = \text{rank}(\mathcal{V})$
 - 4: Find p^{k+1} solving the OCP with the reduced order model (28)
 - 5: **if** $\max_{|\mu| \leq n_{max}} (\Delta_{u_\mu}^\ell(p^k)) > tol$ **then**
 - 6: Set $k=k+1$
 - 7: GOTO 2
 - 8: **end if**
-

In our simulations this approach turned out to be very efficient since it avoids long pre-computations. In this way we are able to update the snapshot set and the POD basis functions. Our update contains information on the optimization problem and it improves the quality of our surrogate model. Note that every reduced optimization problem contains the a-posteriori error for the state and sensitivity equations introduced in Section 4.2.

5. Numerical tests

In our numerical example we consider an optimal design problem for a permanent magnet synchronous machine. We start by introducing the model and geometry under consideration. We consider a three-phase six-pole permanent magnet synchronous machine (PMSM) with one buried permanent magnet per pole. The geometry is shown in Figure 1. The goal of the design optimization is to change the size and location of the permanent magnet such that the material of the magnet is minimized while maintaining a lower bound on the electromotive force. We consider a description using three parameters: p_1 the width, p_2 the height and p_3 the central perpendicular distance between the rotor and the surface of the magnet. The region around the permanent magnet (Figure 1 red box) is decomposed into twelve triangles (Figure 1 blue lines). The introduced triangulation of the parametrized domain allows to perform the affine linear decomposition as introduced in (12) with $Q_a = 12 = Q_f$.

PMSMs can be described sufficiently accurate by the magneto-static approximation of Maxwell's equation. The governing parametrized equation is given by

$$\nabla \times (\nu \nabla \times u(p)) = J_{src}(p) - \nabla \times H_{pm}(p), \quad (32)$$

with boundary conditions

$$u|_{BC} = u|_{DA} = 0 \quad \text{and} \quad u|_{AB} = -u|_{CD}$$

where ν is the reluctivity, J_{src} is the source current density and H_{pm} the field of the permanent magnets (PM). We note that equation (32) fits into the abstract formulation (1). In the 2D planar setting together with a finite element method for the magnetic vector potential, we lead to the discrete form given by the linear model presented in (11). To extract performance values the loading method is used to exploit the frequency domain [36]. To obtain quantities like the electromotive force (EMF) a Fourier analysis of the magnetic vector potential around the inner surface of the stator is carried out. This can be written as a linear function $E_0 = \mathbb{E}^\top \mathbf{u}_h$. More details on the configuration we adopt in the present work can be found in [2, 21, 32].

Let us next formulate the optimization problem. We start by introducing the nominal optimization problem. The goal of the optimization is to minimize the required material for the permanent magnet while maintaining a lower bound on the EMF E_0 . In the mathematical model this leads to a cost function of the form

$$\min_{p \in \mathbb{R}^3} g_0(p) := p_1 p_2 + \rho \max(0, E_0^d - E_0(p, \mathbf{u}(p))),$$

where E_0^d is the desired lower bound on the EMF and $\rho \in \mathbb{R}^+$ a weight parameter. Additionally, we have the constraints

$$(1, 1, 5) \leq (p_1, p_2, p_3) \leq (\infty, \infty, 14), \quad p_2 + p_3 \leq 15 \quad \text{and} \quad 3p_1 - 2p_3 \leq 50.$$

The upper and lower bounds for the parameters and the first inequality are due to the parametrization of the geometry and the restriction that the permanent

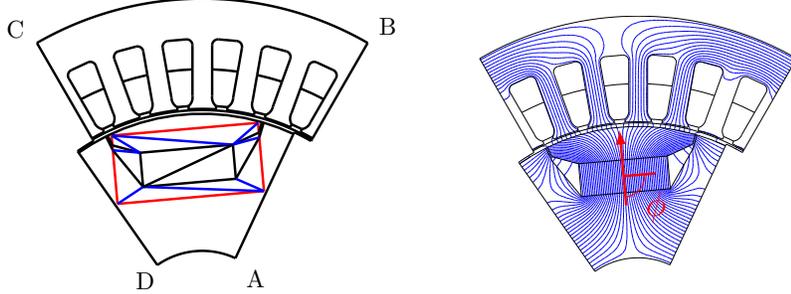


Figure 1: Geometry configuration with region for the affine decomposition (red box) and triangulation for the decomposition (blue lines) (left plot). Magnetic vector potential for the geometry configuration and magnetic field angle ϕ (right plot).

magnet has to stay within the red box in Figure 1 (left). The last inequality is a design restriction that avoids that the corner of the permanent magnet comes too close to the rotor surface. Note that we right away use the reduced formulation as introduced in (16). To obtain a smooth formulation we introduce a slack variable. We reformulate the nominal optimization problem by using the variable $x = (p, \xi)$ as

$$\min_{x \in \mathbb{R}^4} g_0(x) := p_1 p_2 + \rho \xi \quad \text{s.t.} \quad g_{1, \dots, 8}(x) = \begin{pmatrix} p_2 + p_3 - 15 \\ 3p_1 - 2p_3 - 50 \\ E_0^d - E_0(\mathbf{u}_h, p) - \xi \\ 1 - p_1 \\ 1 - p_2 \\ 5 - p_3 \\ -\xi \\ p_3 - 14 \end{pmatrix} \leq 0. \quad (33)$$

In this form the optimization problem fits exactly into the framework of (16).

Next let us introduce the uncertainty. For our numerical example we assume that the magnetic field angle in the permanent magnet is uncertain [29]. This can be due to manufacturing imprecision. In the nominal optimization the magnetic field is aligned perfectly, i.e. the field angle is 90, see Figure 1 (right plot). In practice this can not be met and a deviation is to be expected. The field angle enters the model problem (32) nonlinearly through the right hand side, in particular in the term H_{pm} . In our numerical example we allow a field angle ϕ in the range $[85, 95]$. Following the definition of the uncertainty set in Section 3.2 we get

$$\mathcal{U}_k = \{\phi \in \mathbb{R} \mid \phi = 90 + \delta, \|0.2\delta\|_k \leq 1\}$$

with $k \in \{2, \infty\}$, i.e. $\hat{\phi} = 90$ and $D = 5$ in (17). Using these settings we can now solve the linear and quadratic approximation of the robust counterpart

(21) and (24). In the case of the linear approximation we choose $k = \infty$ and for the quadratic approximation we set $k = 2$. Note that in the presented setting these two uncertainties sets are identical. The different choices are for technical reason as outlined in the derivation.

Before presenting the numerical results let us give a short overview of the numerical strategy utilized to solve the optimization problems. The computations are carried out in MATLAB. To solve the nominal and robust optimization problems an SQP method with Armijo-backtracking strategy using a ℓ_1 -penalty function is used [28]. The Hessian is computed via BFGS updates. Alternatively also routines like `fmincon` in MATLAB can be used obtaining similar results. The derivative of g_i , $i = 0, \dots, 8$ are computed using the sensitivity approach [20]. Also the derivatives with respect to the uncertain parameter ϕ are computed using this approach. The structure of the sensitivity equations are as outlined in (13).

5.1. Results obtained by the finite element approximation

We start by presenting the numerical results utilizing the finite element approximation. Piecewise linear and continuous finite elements are used to discretize equation (32) leading to a system with 61013 degrees of freedom.

The initial geometry configuration corresponding to $p = (19, 7, 7)$ is shown in Figure 1 (left) together with the corresponding magnetic vector potential (right). From the magnetic vector potential we extract the EMF which we will use as the desired value $E_0^d = 30.34$ in our optimization problem.

In Table 1 we show the results obtained in the optimization. In column V_{pm} the volume of the permanent magnets is given. Note that through the optimization a significant reduction in size is achieved. The ratio is given in percent in the second column. For the nominal optimization a reduction by 53% is obtained and in the robust case reductions of 50% (linear case) and 49% (quadratic case). The respective parameters are given in the third column. Lastly, in column four and five the EMF E_0 for the field angle $\phi = 90$ and the worst-case EMF E_0^{worst} are given. It can be seen that the uncertainty in the magnetic field of the permanent magnet has an impact on the performance. In the robust optimization this influence is incorporated. Hence it can be seen that in the case of the quadratic approximation (Rob. Quad and Rob. Shift) very good results can be achieved: the worst-case EMF stays above the target value of 30.34. Shifting the expansion point (Rob. Shift) helps to avoid being overly conservative, so that the EMF target is precisely met, which allows to reduce the volume slightly. In the case of the nominal optimization a significant decay can be observed. In the case of the linear approximation the target can not be met since the approximation is not accurate enough. It can also be observed that by performing only a nominal optimization the worst-case can decrease compared to the initial configuration. In the presented case the difference is small but can become more significant in different settings.

The approximation quality and the behavior of the EMF is shown in Figure 2 for the different optimal designs. In the left plot the behavior of the EMF for different permanent magnet field angles ϕ is shown for the initial and the

	V_{pm}	%	p	E_0 (90)	E_0^{worst}
Init	133.00	100	(19.00, 7.00, 7.00)	30.3483	29.8873
Nom. Opt	62.36	47	(21.08, 2.96, 6.62)	30.3483	29.8873
Rob. Lin	66.46	50	(21.10, 3.15, 6.64)	30.5817	30.2322
Rob. Quad	67.95	51	(21.10, 3.22, 6.65)	30.6995	30.3487
Rob. Shift	67.94	51	(21.10, 3.22, 6.65)	30.6992	30.3483

Table 1: Comparison of the results obtained by the optimization and robust optimization using different approximation orders.

nominal optimal configuration. It can be seen that the target E_0^d is only reached in $\phi = 90$. In the middle plot the linear approximation and the actual EMF are compared. It can be seen that the approximation is not accurate enough to determine the worst-case. The right plot corresponds to the quadratic approximation. It can be seen that the EMF is approximated very well by the quadratic model. Hence also good results in the robust optimization using this approximation can be expected. We note that we do not include a plot for the quadratic approximation with a shifted expansion point because the difference is not visible from the standard quadratic approximation.

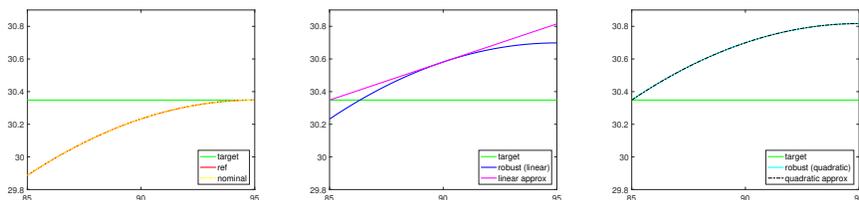


Figure 2: Influence of the permanent magnet field angle on the electromotive force for different geometry configuration.

In Figure 3 the magnetic vector potentials are shown for the three optimal designs. To conclude we have a look at the computational expenses for the optimization. These are summarized in Table 2, where the computational time in seconds, the number of iterations and the number of required PDE solves are compared. While the computational time in this example is still very low it can be seen that the robust optimization is more expensive. Especially the number of PDE solves increases significantly compared to the nominal optimization. Hence we will use model order reduction to reduce the computational costs. This will be outlined in the next section.

5.2. Results obtained by the reduced order model

Let us start our analysis of the reduced order model by dealing with the approximation of the state and the sensitivity variables. To this aim we select

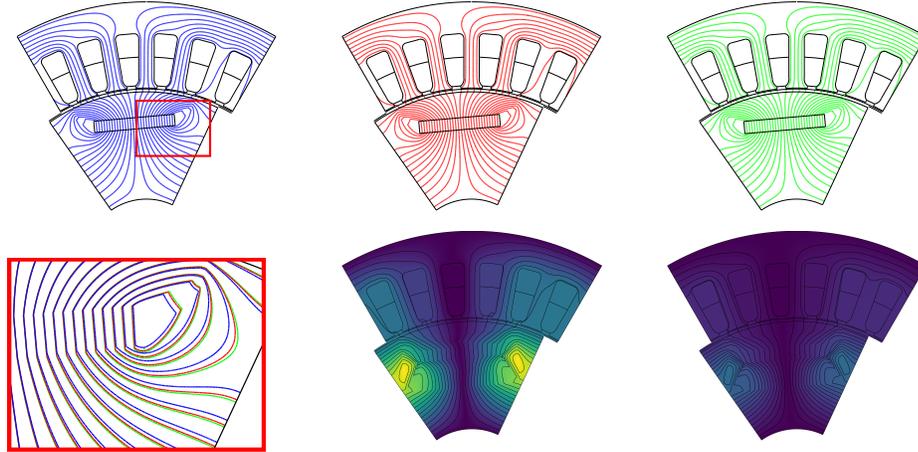


Figure 3: Magnetic vector potential for the geometry configuration obtained by the optimization. First row, left to right: nominal optimal design, robust optimal design with linear approximation, and with quadratic approximation. Second row, left to right: magnified view of the combined potentials, difference between linear and nominal, difference between quadratic and linear (brighter means larger difference).

	CPU (s)	Iter	PDE solves
Nom. Opt	38.76	18	72
Rob. Lin	42.71	8	64
Rob. Quad	46.47	6	72
Rob. Shift	88.51	7	84

Table 2: Performance of the SQP method and computational cost.

125 parameters chosen as follows:

$$\mathcal{D}_{train} = \{1, 5.75, 10.5, 15.25, 20\} \times \{1, 2, 3, 4, 5\} \times \{4, 5.5, 7, 8.5, 10\}$$

and we compute an enlarged snapshot set with state and sensitivities for each parameter in \mathcal{D}_{train} . Altogether, we collect 500 snapshots. We note that we do not need to compute a reduced order model with respect to the slack variable ξ since it is not a variable that is involved in the PDE.

In Figure 4, we present an error analysis to check the quality of our a-posteriori error discussed in Section 4. We compare the error between the POD solution and the high dimensional approximation for state and sensitivities with the error bound presented in Section 4.2. With the \mathbf{W} -norm introduced in Section 4, we set

$$\mathcal{E}_{|\mu|}(p) := \max_{p \in \mathcal{D}_{test}} \|\mathbf{u}_{h,\mu}^n(p) - \mathbf{u}_{h,\mu}^{n,\ell}(p)\|_{\mathbf{W}}, \quad |\mu| = 0, 1,$$

where \mathcal{D}_{test} is chosen in the centers of the boxes of \mathcal{D}_{train} as follows:

$$\mathcal{D}_{test} = \{3.375, 8.125, 12.875, 17.625\} \times \{1.5, 2.5, 3.5, 4.5\} \times \{4.75, 6.25, 7.750, 9.25\}.$$

As one can see model order reduction is able to reach only an error of size 10^{-4} (with a large amount of basis functions) and we would need to increase the number of basis functions to achieve higher accuracy. We also want to emphasize that a standard POD approach, with only state snapshots in \mathcal{D}_{train} , is able to reach an accuracy of order 10^{-2} with 50 basis functions.

In Figure 5, we show that it is possible to reach an accuracy of order 10^{-6} or higher with only 4 POD basis functions if we compute the snapshots with respect to one parameter in \mathcal{D}_{train} and then compute the error in a ball of radius 0.1, centered in the chosen parameter. This is our motivation to introduce Algorithm 1 for the successive enrichment of the POD model. In fact, we start our algorithm with only one parameter and we require the combined snapshots since in any other case we will not have enough data to generate a surrogate model from the simulation and the sensitivities. We note that if we add the sensitivities, the POD basis functions are improved and the parameter domain can be better explored. We refer the interested reader to [19] for more details.

Let us now draw our attention to the optimization problem and its performances when combined with model order reduction. Table 3 shows the convergence of Algorithm 2 for the nominal optimization. The number of updates of the snapshot set is given in the first column. Since our goal is to reach a desired electromotive force, we will also compare $E_0(p, \mathbf{u}_{h,\mu}^{0,\ell}(p), 90)$ obtained from the POD, and $E_0(p, \mathbf{u}_{h,\mu}^0(p), 90)$ obtained from the full simulation. As one can see with the surrogate-based optimization obtain the same results as with the full model. The fifth column presents the number of iterations needed in each sub-optimization problem and the sixth column presents the worst case of the error estimate $\Delta_{u_\mu}^\ell(p)$ for $0 \leq |\mu| = n \leq 1$. In the stopping criterium of Algorithm 2 we choose $tol = 1e-4$. In the last column we show the number of basis functions which also represents the dimension of the reduced problem. The dimension of

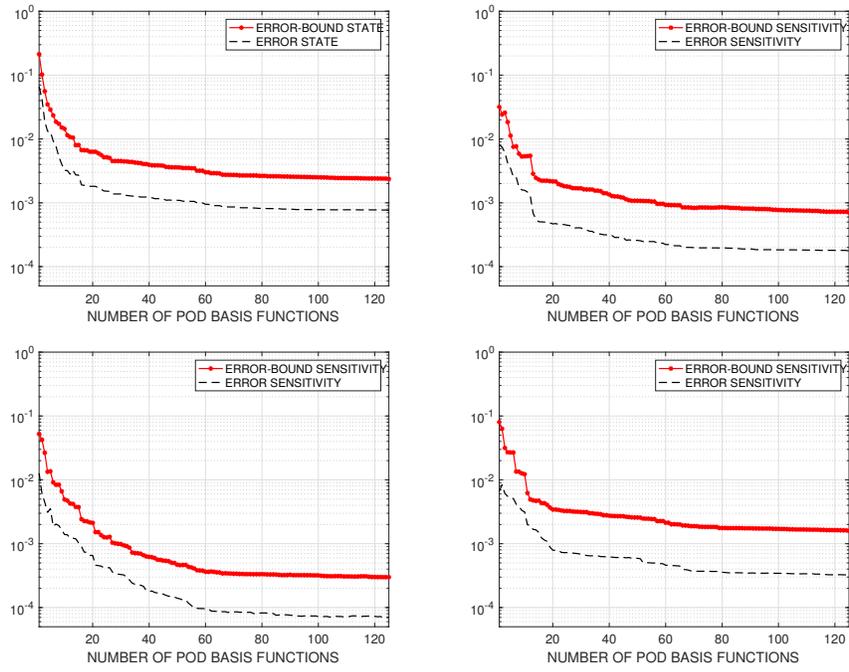


Figure 4: Maximum error over all parameter configurations related to \mathcal{D}_{test} . Error behavior \mathcal{E}_0 and error bound for state equation (top-left), Error behavior $\mathcal{E}_{|\mu|}$ for $|\mu| = 1$ and error bound for the sensitivity equations (top-right and bottom).

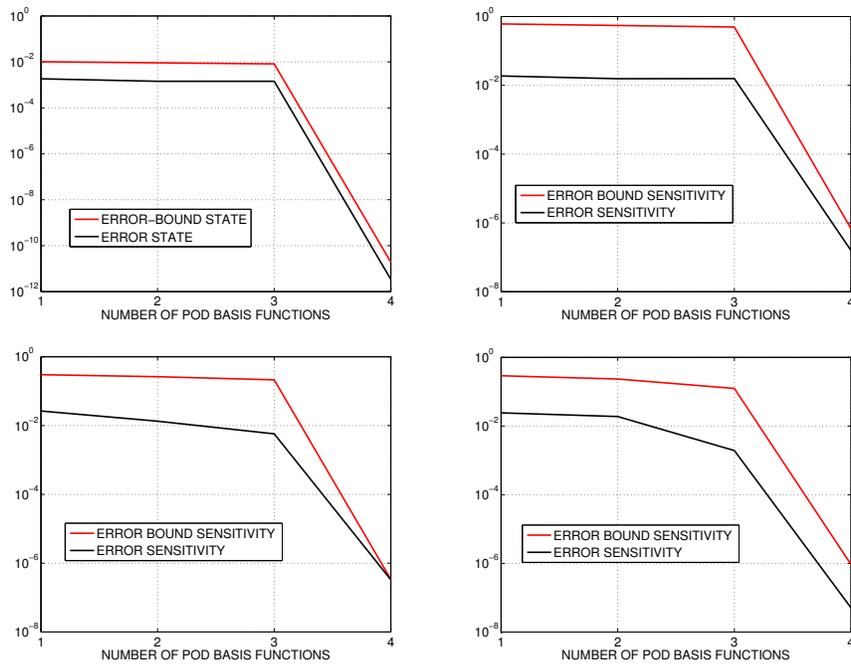


Figure 5: Maximum error of the surrogate model computed with respect to a parameter in the neighborhood of \mathcal{D}_{train} . Error behavior for state equation (top-left), and for the sensitivity equations for each parameter (top-right and bottom).

the reduced space corresponds to the rank of current snapshot matrix. We note that the results of the algorithm are close to what we have shown in Table 2. The CPU time will be discussed at the end of the section.

iter	$E_0(p, \mathbf{u}_h(p), 90)$	$E_0(p, \mathbf{u}_h^\ell(p), 90)$	V	$\#it$	$\max_{ \mu \leq 1} \Delta_{u_\mu^\ell}^\ell(p^k)$	ℓ
0	30.3483	30.3483	133.00			
1	30.3764	30.3483	62.82	8	3.26e-02	4
2	30.3483	30.3483	62.36	4	2.97e-06	8

Table 3: Performance of POD with nominal optimization

Next we analyze the performance of POD for the robust optimization problem. In Table 4 we present the results of POD with a linear approximation of the robust counterpart with $n_{max} = 2$. As one can see after two iterations we are able to recover the same results obtained without model order reduction, obtaining also a good approximation for the electromotive force (compare column 2 and column 3 in Table 4).

iter	$E_0(p, \mathbf{u}_h(p), 90)$	$E_0(p, \mathbf{u}_h^\ell(p), 90)$	V	$\#it$	$\max_{ \mu \leq 2} \Delta_{u_\mu^\ell}^\ell(p^k)$	ℓ
0	30.3483	30.3483	133.00			
1	30.7319	30.5729	69.41	8	7.12e-01	8
2	30.5817	30.5816	66.46	4	9.83e-05	16

Table 4: Performance of POD with linear robust optimization

Then, we present the results of the surrogate models for the solution of the robust optimization problem with quadratic approximation with $n_{max} = 3$ in Table 5 and with quadratic approximation with moving expansion point in Table 6. The same considerations discussed in the linear example hold true. We note that the number of POD basis functions used in each optimization in this case is larger due to a richer snapshot set.

iter	$E_0(p, \mathbf{u}_h(p), 90)$	$E_0(p, \mathbf{u}_h^\ell(p), 90)$	V	$\#it$	$\max_{ \mu \leq 3} \Delta_{u_\mu^\ell}^\ell(p^k)$	ℓ
0	30.3483	30.3483	133.00			
1	30.8112	30.7332	68.12	6	3.32e-02	8
2	30.6997	30.6995	67.96	5	8.07e-05	16

Table 5: Performance of POD with quadratic robust optimization

Also in the quadratic case we obtain the same approximation quality as in the full model at lower computational costs (see Table 5 and Table 6).

To summarize, all the performances of the surrogate models are presented in Table 7. Again, we emphasize the results agree with Table 1. Strictly speaking,

iter	$E_0(p, \mathbf{u}_h(p), 90)$	$E_0(p, \mathbf{u}_h^\ell(p), 90)$	V	#it	$\max_{ \mu \leq 3} \Delta_{u_\mu}^\ell(p^k)$	ℓ
0	30.3483	30.3483	133.00			
1	30.7882	30.7384	67.59	9	4.05e-02	8
2	30.7908	30.7913	67.74	9	7.11e-04	15
3	30.7934	30.7933	67.87	9	7.29e-05	16

Table 6: Performance of POD with quadratic robust optimization with shift

	V_{pm}	%	p	$E_0(90)$	E_0^{worst}
Init	133.00	100	(19.00 7.00 7.00)	30.34	29.88
Nom. Opt	61.24	46%	(21.25 2.88 6.87)	30.23	29.77
Rob. Lin	66.71	50%	(20.76 3.21 6.14)	30.56	30.21
Rob. Quad	67.48	51%	(21.21 3.17 6.92)	30.76	30.29
Rob. Shift	67.86	51%	(21.31 3.18 6.96)	30.79	30.32

Table 7: Comparison of the results obtained by POD coupled with the optimization and robust optimization using different approximation orders.

none of the solutions obtained with reduced models are robust for the high-fidelity problem. This is demonstrated by the last column of Table 7. All of the numbers should be, but none of them is greater than or equal to 30.34. The relative worst-case infeasibility violation ranges from 1.9% for the nominal solution to less than 0.07% for the improved quadratic model. Seeing that the reduced models are able to reduce the volume more than the high-fidelity models, this is to be expected. In many applications, an infeasibility as low as 0.07% in the worst case presumably is acceptable; if not, a post-optimization analysis is inevitable.

The computational costs of the POD approach are summarized in Table 8. As one can see we obtain a reduction of the amount of PDEs solved (compare Table 2), keeping the same accuracy attained in the high dimensional problem. For instance, in the quadratic robust optimization with model order reduction we only have to perform 36 PDE solves in contrast to 72 PDE solves in the optimization with the high-fidelity finite element model. The POD approach for quadratic approximation with shift presents a speed up of factor 5. Although we solve more (reduced) PDEs the CPU time is reduced due to the number of full dimensional PDEs to be solved in the full setting, compare Tables 2 and 8. Finally, we note that the CPU time involves both offline and online stage, so that the speed up is considerable.

6. Conclusion

In this paper we present a new approach which combines model order reduction to robust optimal control. We investigate a parameter optimization problem governed by a parametric elliptic partial differential equation with uncertain

	CPU (s)	PDE solves	Reduced PDE solves
Nom. Opt	7.7	8	48
Rob. Lin	11.3	24	110
Rob. Quad	14.2	36	132
Rob. Shift	20.9	48	324

Table 8: Performance of the POD method and computational cost.

parameters. We introduce a robust optimization framework that accounts for uncertain model and optimization parameters. By shifting the expansion point of the utilized Taylor approximation, we are able to model the nonlinear effects of the uncertain parameter more accurately and improve the robust solutions slightly. The resulting non-linear optimization problem has a bi-level structure due to the min-max formulation. We propose an adaptive model order reduction approach which avoids long offline stages and provides a certified reduced order surrogate model for the parametrized PDE which then is utilized in the numerical optimization. The presented numerical results clearly illustrate the validity and performance of the presented approach.

7. Acknowledgements

The authors wish to acknowledge the support of the German BMBF in the context of the SIMUROM project (grant number 05M2013), and the support of the German Research Foundation in the context of SFB 805.

References

- [1] K. Afanasiev and M. Hinze. *Adaptive control of a wake flow using proper orthogonal decomposition*. Lecture Notes in Pure and Applied Mathematics, **216**, 2001, 317–332.
- [2] A. Alla, M. Hinze, O. Lass and S. Ulbrich. *Model order reduction approaches for the optimal design of permanent magnets in electro-magnetic machines*. IFAC-PapersOnLine, **43**, 2015, 242-247.
- [3] A. Alla, U. Matthes. *Model order reduction for a linearized robust PDE constrained optimization*. IFAC-PapersOnLine, **49**, 2016, 321-326.
- [4] A.C. Antoulas. *Approximation of Large-Scale Dynamical Systems*. SIAM, 2005.
- [5] E. Arian, M. Fahl and E. Sachs. *Trust-region proper orthogonal decomposition models by optimization methods*. In Proceedings of the 41st IEEE Conference on Decision and Control, Las Vegas, Nevada, 2002, 3300–3305.

- [6] A. Ben-Tal, A. Goryashko and A. Nemirovski. *Robust optimization*. Princeton University Press, 2009.
- [7] A. Ben-Tal and A. Nemirovski. *Robust optimization – methodology and application*. Mathematical Programming, **92**, 2002, 453–480.
- [8] D. Bertsimas, D.B. Brown and C. Caramanis. *Theory and application of robust optimization*. SIAM Review, **53**, 2011, 464–501.
- [9] J. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer, 1997.
- [10] A.R. Conn, N.I.M. Gould and P.L. Toint. *Trust-Region Methods*. Society for Industrial and Applied Mathematics, 2000.
- [11] M. Diehl, H.G. Bock and E. Kostina. *An approximation technique for robust nonlinear optimization*. Mathematical Programming, **107**, 2006, 213–230.
- [12] M. Diehl, J. Gerhard, W. Marquardt and M. Mönnigmann. *Numerical solution approaches for robust nonlinear optimal control problems*. Computer & Chemical Engineering, **32**, 2008, 1287–1300.
- [13] M. Dihlmann and B. Haasdonk. *Certified PDE-constrained parameter optimization using reduced basis surrogate models for evolution problems*. Computational Optimization and Applications, **60**, 2015, 753–787.
- [14] L.C. Evans. *Partial Differential Equations*. American Math. Society, Providence, Rhode Island, 2008.
- [15] R. Fletcher, S. Leyffer, D. Ralph and S. Scholtes. *Local convergence of SQP methods for mathematical programs with equilibrium constraints*. SIAM Journal on Optimization, **17**, 2006, 259–286.
- [16] M. Gubisch and S. Volkwein. *Proper Orthogonal Decomposition for Linear-Quadratic Optimal Control*. In P. Benner, A. Cohen, M. Ohlberger, and K. Willcox (eds.), *Model Reduction and Approximation: Theory and Algorithms*. 5-64, SIAM, Philadelphia, PA, 2017.
- [17] B. Haasdonk. *Reduced basis methods for parametrized PDEs - a tutorial introduction for stationary and instationary problems*. Chapter in P. Benner, A. Cohen, M. Ohlberger and K. Willcox (eds.): *Model Reduction and Approximation: Theory and Algorithms*, 65-136, SIAM, Philadelphia, 2017.
- [18] B. Haasdonk, and M. Ohlberger. *Reduced Basis Method for Finite Volume Approximations of Parametrized Linear Evolution Equations*. M2AN, Math. Model. Numer. Anal., **42**, 2008, 277–302.
- [19] A. Hay, J. Borggaard, D. Pelletier. *Local improvements to reduced-order models using sensitivity analysis of the proper orthogonal decomposition*, J. Fluid Mech. **629**, 2009, 41–72.

- [20] M. Hinze, R. Pinnau, M. Ulbrich and S. Ulbrich. *Optimization with PDE Constraints. Mathematical Modelling: Theory and Applications*, 23. Springer Verlag, 2009.
- [21] S. Hennerberger, U. Pahner, K. Hameyer and R. Belmans. *Computation of a highly saturated permanent magnet synchronous motor for a hybrid electric vehicle*. IEEE Trans. Magn., **33**, 1997, 4086–4088.
- [22] B. Houska and M. Diehl. *Nonlinear robust optimization via sequential convex bilevel programming*. Mathematical Programming, **142**, 2013, 539–577.
- [23] E. Kammann, F. Tröltzsch and S. Volkwein. *A method of a-posteriori error estimation with application to proper orthogonal decomposition*. ESAIM: M2AN, **47**, 2013, 555–581.
- [24] M. Kärcher and M. Grepl. *A certified reduced basis method for parametrized elliptic optimal control problems*. ESAIM: COCV, **20**, 416–441, 2014.
- [25] D.P. Kouri and T.M. Surowiec. *Risk-averse PDE-constrained optimization using the conditional value-at-risk*. SIAM Journal on Optimization, **26**, 2016, 365–396.
- [26] O. Lass and S. Ulbrich. *Model order reduction techniques with a posteriori error control for nonlinear robust optimization governed by partial differential equations*. SIAM J. on Sc. Comp., **39**, 2017, S112-S139.
- [27] S. Leyffer. *Complementarity constraints as nonlinear equations: Theory and numerical experience*, in Optimization with Multivalued Mappings: Theory, Applications, and Algorithms, S. Dempe and V. Kalashnikov (eds.), **2** of Springer Series in Optimization and Its Applications, Springer, 2006, 169–208.
- [28] J. Nocedal and S.J. Wright. *Numerical Optimization, second edition*. Springer Series in Operation Research and Financial Engineering, 2006.
- [29] P. Offermann and K. Hameyer. *A polynomial chaos meta-model for nonlinear stochastic magnet variations*. COMPEL - The international journal for computation and mathematics in electrical and electronic engineering, **32**, 2013, 1211–1218.
- [30] B. Øksendal. *Optimal control of stochastic partial differential equations*. Stochastic Analysis and Applications, **23**, 2005, 165–179.
- [31] I.B. Oliveira and A.T. Patera. *Reduced-basis techniques for rapid reliable optimization of systems described by affinely parametrized coercive elliptic partial differential equations*. Optim Eng **8**, 2007, 43–65.
- [32] U. Pahner. *A general design tool for theoretical optimization of electromagnetic energy transducers*. PhD Thesis, KU Leuven, 1998.

- [33] A. T. Patera and G. Rozza. *Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations* MIT book, 2007.
- [34] M.J.D. Powell. *Variable Metric Methods for Constrained Optimization*. In: *Mathematical Programming – The State of the Art: Bonn 1982*, A. Bachem, B. Korte, and M. Grtschel (eds.), Springer, 1983, 288–311
- [35] E. Qian, M. Grepl, K. Veroy and K. Willcox. *A Certified Trust Region Reduced Basis Approach to PDE-Constrained Optimization*. *SIAM J. Sci. Comput.*, **39**, 2016, S434-S460.
- [36] M.A. Rahman and P. Zhou. *Determination of saturated parameters of PM motors using loading magnetic fields*. *IEEE Trans. Magn.*, **27**, 1991, 3947–3950.
- [37] 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*. *Arch. Comput. Methods Eng.*, **15**, 2008, 229–275.
- [38] A. Sichau. *Robust Nonlinear Programming with Discretized PDE Constraints using Second-order Approximations*. PhD Thesis, TU Darmstadt, 2013.
- [39] F. Tröltzsch. *Optimal Control of Partial Differential Equations: Theory, Methods and Application*, American Mathematical Society, 2010.
- [40] H. Tiesler, R.M. Kirby, D. Xiu, and T. Preusser. *Stochastic Collocation for Optimal Control Problems with Stochastic PDE Constraints*. *J. Control Optim.*, **50**, 2012, 2659–2682.
- [41] F. Tröltzsch and S. Volkwein. *POD a-posteriori error estimates for linear-quadratic optimal control problems*. *Comput. Optim. and Appl.*, **44**, 2009, 83–115.
- [42] S. Volkwein. *Model Reduction using Proper Orthogonal Decomposition*, Lecture Notes, University of Konstanz, 2013.
- [43] M. Zahr and C. Farhat. *Progressive Construction of a Parametric Reduced-Order Model for PDE-Constrained Optimization*. *Internat. J. Numer. Methods Engrg.* **102**, 2015, 1111–1135.
- [44] Y. Zhang. *General robust-optimization formulation for nonlinear programming*. *J. Optim. Theory and Appl.*, **132**, 2007, 111–124.