DFG Deutsche Forschungsgemeinschaft
Priority Programme 1962

Set-Oriented Multiobjective Optimal Control of PDEs using Proper Orthogonal Decomposition

Dennis Beermann, Michael Dellnitz, Sebastian Peitz, Stefan Volkwein

SPP 1962 | Non-smooth and Complementarity-based Distributed Parameter Systems: Simulation and Hierarchical Optimization

Preprint Number SPP1962-017

received on April 25, 2017

Edited by
SPP1962 at Weierstrass Institute for Applied Analysis and Stochastics (WIAS)
Leibniz Institute in the Forschungsverbund Berlin e.V.
Mohrenstraße 39, 10117 Berlin, Germany
E-Mail: spp1962@wias-berlin.de

World Wide Web: http://spp1962.wias-berlin.de/

Set-Oriented Multiobjective Optimal Control of PDEs using Proper Orthogonal Decomposition

Dennis Beermann and Michael Dellnitz and Sebastian Peitz and Stefan Volkwein

Abstract In this chapter, we combine a global, derivative-free subdivision algorithm for multiobjective optimization problems with a-posteriori error estimates for reduced-order models based on Proper Orthogonal Decomposition in order to efficiently solve multiobjective optimization problems governed by partial differential equations. An error bound for a semilinear heat equation is developed by which the errors in the conflicting objectives can be estimated individually. The resulting algorithm constructs a library of locally valid reduced-order models online using a Greedy (worst-first) search. Using this approach, the number of evaluations of the full-order model can be reduced by a factor of more than 1000.

1 Introduction

Many problems in engineering and physics can be modeled by partial differential equations (PDEs), starting from fairly simple problems such as the linear heat equation up to highly non-linear fluid flow problems governed by the Navier-Stokes equations. When designing an application where the underlying dynamical system is given by a PDE, we are faced with a PDE-constrained optimization or optimal control problem [28]. Due to the ever increasing complexity of technical systems

Dennis Beermann
Department of Mathematics and Statistics, University of Konstanz, e-mail: dennis.beermann@uni-konstanz.de

Michael Dellnitz
Department of Mathematics, Paderborn University, e-mail: dellnitz@math.upb.de

Sebastian Peitz
Department of Mathematics, Paderborn University, e-mail: speitz@math.upb.de

Stefan Volkwein
Department of Mathematics and Statistics, University of Konstanz, e-mail: stefan.volkwein@uni-konstanz.de

and design requirements, there are nowadays only few problems, where only one objective is of importance. For example, in buildings we want to provide a comfortable room temperature while at the same time minimizing the energy consumption. This example illustrates that many objectives are often equally important and also contradictory such that we are forced to accept a trade-off between them. This results in a *multiobjective optimization problem* (MOP), where multiple objectives have to be minimized at the same time. Similar to scalar optimization problems, we want to find an optimal solution to this problem. However, in a multiobjective optimization problem, we have to identify the set of *optimal compromises*, the so-called *Pareto set*.

Multiobjective optimization is an active area of research. Different approaches exist to address MOPs, e.g., *deterministic approaches* [8, 16], where ideas from scalar optimization theory are extended to the multiobjective situation. In many cases, the resulting numerical method involves the consecutive solution of multiple scalar optimization problems. *Continuation methods* make use of the fact that under certain smoothness assumptions the Pareto set is a manifold [11]. Another prominent approach is based on *evolutionary algorithms* [5], where the underlying idea is to evolve an entire set of solutions (population) during the optimization process. *Set-oriented methods* provide an alternative deterministic approach to the solution of MOPs. Utilizing subdivision techniques (cf. [7, 26]), the desired Pareto set is approximated by a nested sequence of increasingly refined box coverings.

When addressing PDE-constrained MOPs, many evaluations of this PDE are required and hence, the computational effort quickly becomes prohibitively large. The typical procedure is to discretize the spatial domain by a numerical mesh, which transforms the infinite-dimensional into a (potentially very large) finite-dimensional system (i.e. a system of coupled ordinary differential equations). With increasing computational capacities, the size of problems that can be solved has increased tremendously during the last decades [24]. However, many technical applications result in problems that are nowadays still very difficult or even impossible to solve. Consequently, solving optimal control problems involving PDE constraints is a major challenge and considering multiple criteria further increases the complexity.

To overcome the problem of expensive function evaluations, model-order reduction is a widely used concept. Here, the underlying PDE is replaced by a surrogate model which can be solved much faster [21, 24]. In this context, reduced-order models (ROMs) based on Galerkin projection and *Proper Orthogonal Decomposition (POD)* [12] have proven to be a powerful tool, in particular in a multi-query context such as parameter estimation, uncertainty quantification or optimization (see e.g. [10, 22]). During the last years, the first publications addressing PDE-constrained problems with multiple criteria have appeared. In [2], the model has been treated as a black box and evolutionary algorithms were applied. A comparison of different algorithms for multiobjective optimal control of the Navier-Stokes equations is presented in [20] and approaches using rigorous error analysis can be found in [3, 4, 13, 14].

In this work we extend the results from [19] and [22] in order to develop a global, derivative-free algorithm for PDE-constrained multiobjective optimization

problems based on POD-ROMs. To this end, the subdivision algorithm for inexact function values presented in [19] is combined with a localized reduced basis approach [1] and error estimates for the objectives. The chapter is structured in the following way: In Section 2, the PDE-constrained multiobjective optimization problem is introduced along with the gradient-based and gradient-free version of the subdivision algorithm. In Section 3, an a-posteriori error estimator for the individual objectives is derived. In Section 4, numerical results concerning both the error estimator and the overall algorithm are presented. Finally, we end with a conclusion and an outlook in Section 5.

Notation. If $x^1, x^2 \in \mathbb{R}^n$ are two vectors, we write $x^1 \leq x^2$ if $x_i^1 \leq x_i^2$ for $i = 1, ..., n$, and similarily for $x^1 < x^2$.

2 The Multiobjective Optimal Control Problem

Throughout this chapter, let $\Omega \subset \mathbb{R}^d$ be a bounded Lipschitz domain with boundary Γ. Further, let $(0,T) \subset \mathbb{R}$ be a time interval, $Q := (0,T) \times \Omega$ and $\Sigma := (0,T) \times \Gamma$. The domain contains subdomains $\Omega_i \subset \Omega$ and indicator functions are defined by $\chi_i(x) = 1$ if $x \in \Omega_i$ and $\chi_i(x) = 0$ otherwise $(i = 1, ..., m)$. The finite-dimensional control space is given by $\mathcal{U} = \mathbb{R}^m$ and we consider the following *Multiobjective Optimal Control Problem* (MOCP):

$$\min_{u \in \mathcal{U}} J(y,u) = \frac{1}{2} \begin{pmatrix} \int_\Omega |y(T,x) - y_{d,1}(x)|^2 \, \mathrm{d}x \\ \int_\Omega |y(T,x) - y_{d,2}(x)|^2 \, \mathrm{d}x \\ |u|_2^2 \end{pmatrix} \tag{1a}$$

subject to (s.t.) the semilinear PDE constraints

$$\begin{aligned} y_t(t,x) - \Delta y(t,x) + y^3(t,x) &= \sum_{i=1}^m u_i \chi_i(x) && \text{for } (t,x) \in Q, \\ \frac{\partial y}{\partial n}(t,s) &= 0 && \text{for } (t,s) \in \Sigma, \\ y(0,x) &= y_0(x) && \text{for } x \in \Omega \end{aligned} \tag{1b}$$

and the bilateral control constraints

$$u_{\mathsf{a}} \leq u \leq u_{\mathsf{b}} \quad \text{in } \mathcal{U}. \tag{1c}$$

In (1a), the functions $y_{d,1}$ and $y_{d,2} \in L^2(\Omega)$ are two conflicting desired states. Moreover, $|\cdot|_2$ denotes the Euclidean norm. The state variable y is given as the solution to the semilinear heat equation (1b) which we will call the *state equation* from now on. It will be shown in Section 2.1 that such a solution always exists, is unique and in particular belongs to $C([0,T]; L^2(\Omega))$, meaning that the integrals in (1a) are well-

defined. In (1c), the control variable u is bounded by bilateral constraints $u_\mathsf{a}, u_\mathsf{b} \in \mathcal{U}$ with $u_\mathsf{a} \leq u_\mathsf{b}$. Therefore, we define the *admissible set*

$$\mathcal{U}_\mathsf{ad} = \big\{ u \in \mathcal{U} \,\big|\, u_\mathsf{a} \leq u \leq u_\mathsf{b} \text{ in } \mathcal{U} \big\}.$$

2.1 The State Equation and its Galerkin Discretization

This section is dedicated to the analysis of the state equation (1b). First of all, let us introduce the Gelfand triple $V \hookrightarrow H = H' \hookrightarrow V'$, where $V = H^1(\Omega)$, $H = L^2(\Omega)$ and each embedding is continuous and dense. We define the solution space $\mathcal{Y} = W(0,T) \cap L^\infty(Q)$ with

$$W(0,T) := L^2(0,T;V) \cap H^1(0,T;V').$$

It is well-known [28] that $W(0,T)$ together with the common inner product is a Hilbert space and continuously embeds into $C([0,T];H)$. Next, we specify what is meant by a *solution* to (1b):

Definition 1. A function $y \in \mathcal{Y}$ is called a *weak solution* to (1b) if it holds for every $\varphi \in V$:

$$\begin{aligned} \langle y_t(t), \varphi \rangle_{V' \times V} + \int_\Omega \big(\nabla y(t) \cdot \nabla \varphi + y(t)^3 \varphi \big) \,\mathrm{d}x = \sum_{i=1}^m u_i \int_{\Omega_i} \varphi \,\mathrm{d}x \quad \text{a.e. in } (0,T), \\ \int_\Omega y(0) \varphi \,\mathrm{d}x = \int_\Omega y_0 \varphi \,\mathrm{d}x. \end{aligned} \tag{2}$$

Here, 'a.e.' stands for 'almost everywhere'. We briefly state a solvability result on (1b) next:

Theorem 1 (Unique solvability of the state equation). *For every $u \in \mathcal{U}$, there exists a unique weak solution $y \in \mathcal{Y}$ of* (2)*. The control-to-state operator*

$$\mathcal{S} : \mathcal{U} \to \mathcal{Y}, \quad u \mapsto y = \mathcal{S}u, \text{ where } (y,u) \text{ solves } (2),$$

is Lipschitz-continuous, meaning there is a constant $L > 0$ such that for all controls $u_1, u_2 \in \mathcal{U}$ with solutions $y_i = \mathcal{S}u_i$ $(i = 1,2)$, it holds

$$\|y_1 - y_2\|_{W(0,T)} + \|y_1 - y_2\|_{L^\infty(Q)} \;\leq\; L |u_1 - u_2|_2.$$

Proof. We observe that the operator

$$\mathcal{B} : \mathcal{U} \to L^r(Q), \quad (\mathcal{B}u)(t,x) := \sum_{i=1}^m u_i \chi_i(x)$$

is linear and Lipschitz-continuous for every $r \in [1,\infty)$. The claim then follows from [28, Theorems 5.5 and 5.8]. □

The definition of the control-to-state operator $\mathcal{S}$ and the admissible set $\mathcal{U}_{\mathsf{ad}}$ allows us to rewrite the MOCP (1) as a minimization problem in the control variable only:

Definition 2. We define the *reduced cost function*

$$\hat{J}: \mathcal{U} \to \mathbb{R}^3, \quad \hat{J}(u) = J(\mathcal{S}u, u)$$

and the (1)-equivalent *reduced* Multiobjective Optimal Control Problem:

$$\min_{u \in \mathcal{U}} \quad \hat{J}(u) = \frac{1}{2} \begin{pmatrix} \|(\mathcal{S}u)(T) - y_{d,1}\|_H^2 \\ \|(\mathcal{S}u)(T) - y_{d,2}\|_H^2 \\ |u|_2^2 \end{pmatrix} \quad \text{s.t. } u \in \mathcal{U}_{\mathsf{ad}}. \tag{$\hat{\mathbf{P}}$}$$

System (2) represents a nonlinear problem posed in infinite-dimensional function spaces. It is solved in practice by a discretization method. Given linearily independent spatial basis functions $\phi_1, \ldots, \phi_n \in V$, the space V is replaced by an n-dimensional subspace $V^h = \operatorname{span}\{\phi_1, ..., \phi_n\}$. We endow V^h with the V-topology. Note that since V^h is of finite dimension, V^h can be identified with $(V^h)'$ so we have an isomorphism

$$L^2(0,T;V^h) \cap H^1(0,T,(V^h)') \cong H^1(0,T,V^h).$$

A Galerkin method is employed to replace the infinite-dimensional problem (2) with a finite-dimensional version. Typically, n is a very large number which is why we refer to the solution of the resulting system as a *high-fidelity solution*:

Definition 3. A function $y^h \in H^1(0,T;V^h)$ is called a *high-fidelity solution* to (1b) if it holds for every $\varphi^h \in V^h$:

$$\begin{aligned} \int_\Omega \left(y_t^h(t)\varphi^h + \nabla y^h(t) \cdot \nabla \varphi^h + y^h(t)^3 \varphi^h \right) \mathrm{d}x = \sum_{i=1}^m u_i \int_{\Omega_i} \varphi^h \,\mathrm{d}x \quad \text{a.e. in } (0,T), \\ \int_\Omega y^h(0)\varphi^h \,\mathrm{d}x = \int_\Omega y_0 \varphi^h \,\mathrm{d}x. \end{aligned} \tag{3}$$

Exactly as in Theorem 1, we obtain the unique solvability of the system (3) and the existence of a Lipschitz-continuous solution operator

$$\mathcal{S}^h: \mathcal{U} \to H^1(0,T;V^h), \quad u \mapsto y^h = \mathcal{S}^h u, \text{ where } (y^h, u) \text{ solves (3).}$$

Likewise, a high-fidelity reduced cost function can be introduced as

$$\hat{J}^h: \mathcal{U} \to \mathbb{R}^3, \quad \hat{J}^h(u) = J(\mathcal{S}^h u, u).$$

2.2 Multiobjective Optimization

Consider the general multiobjective optimization problem

$$\min_{u\in\mathcal{U}} \hat{J}(u) = \min_{u\in\mathcal{U}} \begin{pmatrix} \hat{J}_1(u) \\ \vdots \\ \hat{J}_k(u) \end{pmatrix}, \tag{4}$$

where $\hat{J} : \mathcal{U} \to \mathbb{R}^k$ is a vector-valued objective function with continuously differentiable scalar objective functions $\hat{J}_i : \mathcal{U} \to \mathbb{R}$ for $i = 1,\dots,k$. The space of the 'parameters' u is called the *decision space* and the function $\hat{J}$ is a mapping to the k-dimensional *objective space*. In contrast to single objective optimization problems, there exists no total order of the objective function values in $\mathbb{R}^k$ for $k \geq 2$. Consider for example the points $u_1 = (3,2)$ and $u_2 = (1,3)$ and $u_3 = (2,1)$. Then neither $u_2 < u_1$ nor $u_1 < u_2$. However, $u_3 < u_1$ since $u_{3,1} < u_{1,1}$ and $u_{3,2} < u_{1,2}$.

A consequence of the lack of a total order is that we cannot expect to find isolated optimal points. Instead, the solution of (4) is the set of optimal compromises, the so-called *Pareto set* named after Vilfredo Pareto:

Definition 4.

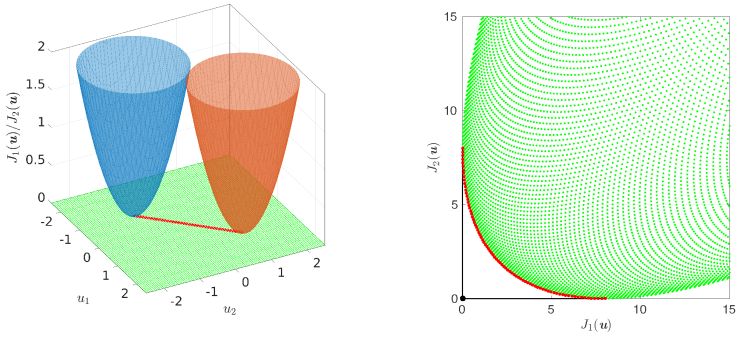
a) A point $u^* \in \mathcal{U}$ *dominates* a point $u \in \mathcal{U}$, if $\hat{J}(u^*) \leq \hat{J}(u)$ and $\hat{J}(u^*) \neq \hat{J}(u)$.
b) A point $u^* \in \mathcal{U}$ is called *(globally) Pareto optimal* if there exists no point $u \in \mathcal{U}$ dominating u^*. The image $\hat{J}(u^*)$ of a (globally) Pareto optimal point u^* is called a *(globally) Pareto optimal value*.
c) The set of non-dominated points is called the *Pareto set* $\mathscr{P}_S \subset \mathbb{R}^m$, its image the *Pareto front* $\mathscr{P}_F \subset \mathbb{R}^k$.
d) When comparing sets, a set $\mathscr{B}^* \subset \mathcal{U}$ *dominates* a set $\mathscr{B} \subset \mathcal{U}$ if for every point $u \in \mathscr{B}$ there exists at least one point $u^* \in \mathscr{B}^*$ dominating u.

Consequently, for each point that is contained in the Pareto set, one can only improve one objective by accepting a trade-off in at least one other objective (cf. Figure 1 (b)). A more detailed introduction to multiobjective optimization can be found in [8, 16], for instance.

As mentioned in the introduction, many different algorithms for solving MOPs exist. In this work, we focus on the set-oriented methods introduced in [7]. In the following, we will shortly describe both the gradient-based and the gradient-free approach. Since here, our results are obtained by the gradient-free version, this will be explained in more detail. The numerical realization of this technique motivates the use of localized reduced basis approaches.

2.2.1 Gradient-Based Multiobjective Optimization

Similar to single objective optimization, a necessary condition for optimality in multiobjective optimization is based on the gradients of the objective functions. In this



(a) (b)

Fig. 1 Red lines: Sketch of a Pareto set (a) and Pareto front (b) for a multiobjective optimization problem of the form (4) with $m = 2$ and $k = 2$.

section, we will restrict ourselves to unconstrained MOPs, i.e. $\mathcal{U}_{\mathsf{ad}} = \mathbb{R}^m$. In the multiobjective situation, the corresponding Karush-Kuhn-Tucker (KKT) condition is stated in the following theorem.

Theorem 2 ([15]). *Let u^* be a Pareto optimal point of* (4)*. Then, there exist non-negative scalars $\alpha_1, \ldots, \alpha_k \geq 0$ such that*

$$\sum_{i=1}^{k} \alpha_i = 1 \quad \textit{and} \quad \sum_{i=1}^{k} \alpha_i \nabla \hat{J}_i(u^*) = 0. \tag{5}$$

Observe that (5) is only a necessary condition for a point u^* to be a Pareto optimal point. The set of points satisfying (5) is called the set of *substationary points* $\mathscr{P}_{S,sub}$ which is obviously a superset of the Pareto set $\mathscr{P}_S$.

If $u \notin \mathscr{P}_S$ then a descent direction $q : \mathcal{U} \to \mathbb{R}^m$ can be defined for which all objectives are decreasing, i.e.

$$\nabla \hat{J}_i(u) \cdot q(u) < 0, \quad i = 1, \ldots, k.$$

One way to compute such a direction is to solve the following auxiliary optimization problem [23]

$$\min_{\alpha \in \mathbb{R}^k} \left\{ \left| \sum_{i=1}^{k} \alpha_i \nabla \hat{J}_i(u) \right|_2^2 \;\middle|\; \alpha_i \geq 0 \text{ for } i = 1, \ldots, k \text{ and } \sum_{i=1}^{k} \alpha_i = 1 \right\}. \tag{QOP}$$

Using (QOP) we obtain the following result.

Theorem 3 ([23]). *Define $q : \mathcal{U} \to \mathcal{U}$ by*

$$q(u) = -\sum_{i=1}^{k} \hat{\alpha}_i \nabla \hat{J}_i(u), \tag{6}$$

where $\hat{\alpha}$ is the solution of (QOP)*. Then either $q(u) = 0$ and u satisfies* (5)*, or $q(u)$ is a descent direction for all objectives $\hat{J}_1(u), \ldots, \hat{J}_k(u)$ in u. Moreover, $q(u)$ is locally Lipschitz-continuous.*

Using the result of Theorem 3, we can construct a global subdivision algorithm [7] by which a nested sequence of increasingly refined box coverings of the entire set of *substationary points* $\mathscr{P}_{S,sub}$ is computed. Besides globality, a benefit of this technique is that it can easily be applied to higher dimensions whereas in particular geometric approaches struggle with a larger number of objectives. The computational cost, however, increases exponentially with the dimension of the Pareto set such that in practice, we are restricted to a moderate number of objectives, i.e. $k \leq 5$.

In order to apply the subdivision algorithm to a multiobjective optimization problem, we first formulate the iteration scheme precisely. One step of the optimization procedure is given by

$$u^{(j+1)} = u^{(j)} + h^{(j)} q\big(u^{(j)}\big), \quad j = 0, 1, 2, \ldots, \tag{7}$$

where $q(u^{(j)})$ is the descent direction according to (6) and $h^{(j)}$ is an appropriately chosen step length (e.g. according to the Armijo rule [17] for all objectives). The subdivision algorithm has initially been developed to compute global attractors for dynamical systems [6]. Using the result of Theorem 3 and interpreting (7) as a dynamical system, one can show that (7) possesses an attractor for which the set of substationary points $\mathscr{P}_{S,sub}$ is a subset. If this set is connected, then it even coincides with the attractor (see [7] for details). Using a multilevel subdivision scheme, a sequence of sets $\mathscr{B}_0, \mathscr{B}_1, \ldots$ can be constructed, where each $\mathscr{B}_s$ is a subset of $\mathscr{B}_{s-1}$.

Remark 1. In the numerical realization, the elements of $\mathscr{B}_s$ are generalized rectangles B (from now on referred to as *boxes*). These are represented by a finite number of sample points; see Figure 2 for an illustration. Those sample points are distributed randomly or on an equidistant grid, for example. $\Diamond$

In the subdivision algorithm, the steps *subdivision* and *selection* are performed alternatingly. In the subdivision step, we construct a new collection of subsets $\hat{\mathscr{B}}_s$ from $\mathscr{B}_{s-1}$ such that

$$\bigcup_{B \in \hat{\mathscr{B}}_s} B = \bigcup_{B \in \mathscr{B}_{s-1}} B, \quad \operatorname{diam}(\hat{\mathscr{B}}_s) = \theta_s \operatorname{diam}(\mathscr{B}_{s-1}), \quad 0 < \theta_{min} \leq \theta_s \leq \theta_{max} < 1,$$

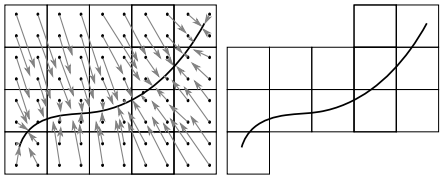
where $\operatorname{diam}(\mathscr{B}_s)$ is the *box diameter*:

$$\operatorname{diam}(\mathscr{B}_s) = \max_{B \in \mathscr{B}_s} \operatorname{diam}(B).$$

In practice, the subdivision is performed by consecutive bisection, cyclically with respect to the coordinate directions.

In the selection step, all boxes are discarded which do not possess a pre-image within $\hat{\mathscr{B}}_s$. In other words, only those boxes are kept which are hit by a descent step

of the dynamical system (7); cf. Figure 2 (b). In the limit $s \to \infty$, the algorithm yields an arbitrarily close covering of the set of substationary points $\mathscr{P}_{S,sub}$.



(a) (b)

Fig. 2 Global subdivision algorithm – selection step. (a) Evaluation of the dynamical system generated by (7). (b) All boxes that do not possess a pre-image within the collection are discarded.

2.2.2 Gradient-Free Multiobjective Optimization

In many applications, gradients are unknown or difficult to compute. Furthermore, in the majority of problems, additional constraints have to be taken into account. In this case, we can use a gradient-free version of the subdivision algorithm which is called the *sampling algorithm* in [7]. This algorithm also consists of a subdivision and a selection step with the difference that the selection step is a non-dominance test. Consequently, we directly compute the Pareto set $\mathscr{P}_S$ instead of the set of substationary points $\mathscr{P}_{S,sub}$. In this chapter, we consider again the constrained problem, i.e., the admissible set $\mathcal{U}_{\mathsf{ad}} \subset \mathcal{U}$ is compact.

Algorithm 1 (Sampling algorithm)

Let $\mathscr{B}_0$ be an initial collection of finitely many subsets of the compact set $\mathcal{U}_{\mathsf{ad}}$ such that $\bigcup_{B\in\mathscr{B}_0} B = \mathcal{U}_{\mathsf{ad}}$. Then, $\mathscr{B}_s$ is inductively obtained from $\mathscr{B}_{s-1}$ in two steps:

i) Subdivision. Construct from $\mathscr{B}_{s-1}$ a new collection of subsets $\hat{\mathscr{B}}_s$ such that

$$\bigcup_{B\in\hat{\mathscr{B}}_s} B = \bigcup_{B\in\mathscr{B}_{s-1}} B, \quad \operatorname{diam}(\hat{\mathscr{B}}_s) = \theta_s \operatorname{diam}(\mathscr{B}_{s-1}), \quad 0 < \theta_{min} \le \theta_s \le \theta_{max} < 1.$$

ii) Selection. Define the new collection $\mathscr{B}_s$ by

$$\mathscr{B}_s = \left\{ B \in \hat{\mathscr{B}}_s \;\middle|\; \nexists \hat{B} \in \hat{\mathscr{B}}_s \text{ such that } \hat{B} \text{ dominates } B \text{ according to Definition 4 d)} \right\}.$$

The numerical realization is similar to the gradient-based version. Every box B is discretized by a finite number of sample points. At each of these sample points, the objectives $\hat{J}$ are evaluated and all dominated points are identified using a non-dominance test [25]. Finally, all dominated boxes are discarded, i.e. boxes which contain only dominated points.

We can conclude that in order to solve (4) by Algorithm 1, we have to evaluate the objectives $\hat{J}$ for many different controls u which can quickly result in a prohibitively large computational effort if model evaluations are expensive. The idea is therefore to introduce a reduced-order model which is less expensive to solve but on the other hand only yields approximations of the exact function values.

2.2.3 Inexact Function Values

Suppose now that we only have approximations $\hat{J}_i^\ell(u)$ of the objectives $\hat{J}_i(u)$. To be more precise, we assume that

$$\hat{J}_i^\ell(u) = \hat{J}_i(u) + \varepsilon_i \quad \text{with } |\hat{J}_i^\ell(u) - \hat{J}_i(u)| \leq \Delta_i^J(u), \tag{8}$$

where the upper bounds $\Delta_i^J(u)$ can be evaluated more efficiently than an actual evaluation of $\hat{J}$. For a more detailed introduction to multiobjective optimization with uncertainties, also with respect to the treatment of inexact gradients, the reader is referred to [19]. In Section 3.2, we will introduce the particular bound that is used for the reduced cost function $\hat{J}$ in ($\hat{\mathbf{P}}$).

Based on the inexactness, we now extend the concept of *non-dominance* (cf. Definition 4) to inexact function values:

Definition 5. Consider the multiobjective optimization problem (4), where the objective functions $\hat{J}_i(u)$, $i = 1,\dots,k$, are only known approximately according to (8).

a) A point $u^* \in \mathcal{U}$ *confidently dominates* a point $u \in \mathcal{U}$, if $\hat{J}_i^\ell(u^*) + \Delta_i^J(u) \leq \hat{J}_i^\ell(u) - \Delta_i^J(u)$ for $i = 1,\dots,k$ and $\hat{J}_i^\ell(u^*) + \Delta_i^J(u) < \hat{J}_i^\ell(u) - \Delta_i^J(u)$ for at least one $i \in 1,\dots,k$.
b) A set $\mathscr{B}^* \subset \mathcal{U}$ *confidently dominates* a set $\mathscr{B} \subset \mathcal{U}$ if for every point $u \in \mathscr{B}$ there exists at least one point $u^* \in \mathscr{B}^*$ confidently dominating u.
c) The *set of almost non-dominated points*, which is a superset of the Pareto set $\mathscr{P}_S$, is defined as:

$$\mathscr{P}_{S,\Delta} = \left\{ u^* \in \mathcal{U} \middle| \nexists u \in \mathcal{U} \text{ with } \hat{J}_i^\ell(u) + \Delta_i^J(u) \leq \hat{J}_i^\ell(u^*) - \Delta_i^J(u),\ i = 1,\dots,k \right\}. \tag{9}$$

Using Definition 5, we can extend the gradient-free Algorithm 1 to the situation where the function values are only known approximately. This can simply be achieved by changing the non-dominance test in the selection step to the stronger condition in (9). This allows us to compute the set of almost non-dominated points $\mathscr{P}_{S,\Delta}$ based on reduced-order models using Algorithm 2. Consequently, if we are interested in approximating the exact Pareto set with a prescribed accuracy, we are faced with the challenge to control the error of the underlying reduced-order model such that it satisfies condition (8). Algorithm 2 requires a large number of function evaluations for different controls u with an error as small as possible. This task can be addressed by using localized reduced basis approaches [1, 18]. Instead of building one model which is globally valid, the concept there is to store several, locally valid ROMs in a library and evaluate the objective functions with the model with

Algorithm 2 (Inexact sampling algorithm)

Let $\mathscr{B}_0$ be an initial collection of finitely many subsets of the compact set $\mathcal{U}_{\mathsf{ad}}$ such that $\bigcup_{B\in\mathscr{B}_0} B = \mathcal{U}_{\mathsf{ad}}$. Then, $\mathscr{B}_s$ is inductively obtained from $\mathscr{B}_{s-1}$ in two steps:

i) Subdivision. Construct from $\mathscr{B}_{s-1}$ a new collection of subsets $\hat{\mathscr{B}}_s$ such that

$$\bigcup_{B\in\hat{\mathscr{B}}_s} B = \bigcup_{B\in\mathscr{B}_{s-1}} B, \quad \operatorname{diam}(\hat{\mathscr{B}}_s) = \theta_s \operatorname{diam}(\mathscr{B}_{s-1}), \quad 0 < \theta_{min} \le \theta_s \le \theta_{max} < 1.$$

ii) Selection. Define the new collection $\mathscr{B}_s$ by

$$\mathscr{B}_s = \left\{ B \in \hat{\mathscr{B}}_s \;\middle|\; \nexists \hat{B} \in \hat{\mathscr{B}}_s \text{ such that } \hat{B} \text{ confidently dominates } B \text{ according to (9)} \right\}.$$

the best accuracy. This has the advantage that the respective models can be smaller in size whereas a reduced model which is accurate within a large range of controls may become too high-dimensional to possess the necessary efficiency. This advantage comes with the price that many high-fidelity function evaluations need to be performed in order to build up the library of different ROMs.

3 Model-Order Reduction

Algorithm 1 implicitly presents us with the task of evaluating the cost function $\hat{J}(u)$ for all sample points $u \in B$ and all sets $B \in \mathscr{B}_s$. Each of these evaluations requires the system (2) (respectively (3) in the numerical implementation) to be solved for the current control. As it was stated before, it is reasonable in this multi-query context to apply model-order reduction techniques in order to reduce the computational effort for the optimization.

3.1 The POD Galerkin Scheme for the State Equation

In this work we utilize the POD method to compute the ROMs; cf. [12]. Suppose that we have chosen an admissible control $u \in \mathcal{U}_{\mathsf{ad}}$ and we would like to build a localized surrogate model which is highly accurate for the data associated with this control. Let $y^h = \mathcal{S}^h u$ denote the associated solution to (3). Then we consider the linear space of snapshots

$$\mathcal{V}^h = \operatorname{span}\left\{ y^h(t) \,\middle|\, t \in [0,T] \right\} \subset V^h \subset V \quad \text{with } \mathfrak{d} = \dim \mathcal{V}^h \le n.$$

For any finite $\ell \le \mathfrak{d}$ we are interested in determining a POD basis of rank ℓ which minimizes the mean square error between $y^h(t)$ and their corresponding ℓ-th partial Fourier sums in the resulting subspace on average in $[0,T]$:

$$\begin{cases} \min \int_0^T \left\| y^h(t) - \sum_{i=1}^{\ell} \langle y^h(t), \psi_i^h \rangle_V \, \psi_i^h \right\|_V^2 \mathrm{d}t \\ \text{s.t. } \{\psi_i^h\}_{i=1}^{\ell} \subset V^h \text{ and } \langle \psi_i^h, \psi_j^h \rangle_V = \delta_{ij} \text{ for } 1 \le i,j \le \ell. \end{cases} \tag{$\mathbf{P}^\ell$}$$

A solution $\{\psi_i^h\}_{i=1}^{\ell}$ to ($\mathbf{P}^\ell$) is called *POD basis of rank* ℓ. Let us introduce the linear, compact, selfadjoint and nonnegative operator $\mathscr{R}^h : V \to V^h$ by

$$\mathscr{R}^h \psi = \int_0^T \langle y^h(t), \psi \rangle_V \, y^h(t) \, \mathrm{d}t \quad \text{for } \psi \in V.$$

Then, it is well-known [10, Theorem 1.15] that a solution $\{\psi_i^h\}_{i=1}^{\ell}$ to ($\mathbf{P}^\ell$) is given by the eigenvectors associated with the ℓ largest eigenvalues of $\mathscr{R}^h$:

$$\mathscr{R}^h \psi_i^h = \lambda_i^h \psi_i^h \text{ for } 1 \le i \le \ell \quad \lambda_1^h \ge \ldots \ge \lambda_\ell^h \ge \lambda_{\ell+1}^h \ge \ldots \ge 0.$$

Moreover, the POD basis $\{\psi_i^h\}_{i=1}^{\ell}$ of rank ℓ satisfies $\psi_i^h \in V^h$ for $1 \le i \le \ell$ and

$$\int_0^T \left\| y^h(t) - \sum_{i=1}^{\ell} \langle y^h(t), \psi_i^h \rangle_V \, \psi_i^n \right\|_V^2 \mathrm{d}t = \sum_{i=\ell+1}^{\mathfrak{d}} \lambda_i.$$

Remark 2 (Discrete POD method). It was already mentioned at the end of Section 2.1 that in the numerical implementation, the space V is replaced by the high-fidelity space V^h. Apart from that, the integral in ($\mathbf{P}^\ell$) has to be approximated numerically. Let a time grid be given by $0 = t_0 < \ldots < t_{\mathsf{k}} = T$ along with integration weights $\gamma_0, \ldots, \gamma_{\mathsf{k}} > 0$. Further, suppose that $y_j^{h\mathsf{k}}$ is an approximation of $y^h(t)$ at the time instance $t = t_j$, $j = 0, \ldots, \mathsf{k}$. Then, the discrete version of ($\mathbf{P}^\ell$) takes the form

$$\begin{cases} \min \sum_{j=0}^{\mathsf{k}} \gamma_j \left\| y_j^{h\mathsf{k}} - \sum_{i=1}^{\ell} \langle y_j^{h\mathsf{k}}, \psi_i^{h\mathsf{k}} \rangle_V \, \psi_i^{h\mathsf{k}} \right\|_V^2 \\ \text{s.t. } \{\psi_i^{h\mathsf{k}}\}_{i=1}^{\ell} \subset V^h \text{ and } \langle \psi_i^{h\mathsf{k}}, \psi_j^{h\mathsf{k}} \rangle_V = \delta_{ij} \text{ for } 1 \le i,j \le \ell. \end{cases} \tag{$\mathbf{P}_{\mathsf{k}}^\ell$}$$

As for the continuous version ($\mathbf{P}^\ell$), a solution to ($\mathbf{P}_{\mathsf{k}}^\ell$) is given by the eigenvectors to the ℓ largest eigenvalues of the operator

$$\mathscr{R}^{h\mathsf{k}} : V \to V^h, \quad \mathscr{R}^{h\mathsf{k}} \psi = \sum_{j=0}^{\mathsf{k}} \gamma_j \, \langle y_j^{h\mathsf{k}}, \psi \rangle_V \, y_j^{h\mathsf{k}} \quad \text{for } \psi \in V,$$

which is an approximation of $\mathscr{R}^{h\mathsf{k}}$. ◊

Now suppose that we have computed a POD basis $\{\psi_i^h\}_{i=1}^{\ell} \subset V^h$ of rank $\ell \ll n$. We define the finite dimensional subspace $V^{h\ell} = \operatorname{span}\{\psi_1^h, \ldots, \psi_\ell^h\} \subset V^h$. Then the POD solution operator $\mathcal{S}^{h\ell} : \mathcal{U} \to H^1(0,T;V^{h\ell}) \hookrightarrow \mathcal{Y}$ is defined as follows: $y^{h\ell} = \mathcal{S}^{h\ell} u$ with $y^\ell(t) \in V^\ell$ for all $t \in [0,T]$ solves the following POD Galerkin scheme for every $1 \le i \le \ell$:

$$\int_\Omega \left(y_t^{h\ell}(t)\psi_i^h + \nabla y^{h\ell}(t)\cdot\nabla\psi_i^h + (y^{h\ell}(t)^3\psi_i^h \right) \mathrm{d}x = \sum_{i=1}^m u_i \int_{\Omega_i} \psi_i^h \,\mathrm{d}x \text{ a.e. in } (0,T),$$
$$y^{h\ell}(0) = \mathscr{P}^{h\ell} y_0,$$

where the linear projection operator $\mathscr{P}^{h\ell} : H \to V^{h\ell}$ is given by

$$y_0^{h\ell} = \mathscr{P}^{h\ell} y_0 = \operatorname*{arg\,min}_{\varphi^{h\ell}\in V^{h\ell}} \|\varphi^\ell - y_0\|_H.$$

We assume that the operator $\mathscr{P}^{h\ell}$ is a bounded operator from V to V. Then, we can apply [27, Theorem 5.3]. It follows from [22] that the operator $\mathcal{S}^{h\ell}$ is well-defined and

$$\|\mathcal{S}^h u - \mathcal{S}^{h\ell} u\|^2_{L^2(0,T;V)} \le C \sum_{i=\ell+1}^{\mathfrak{d}} \lambda_i^h \,\|\psi_i^h - \mathscr{P}^{h\ell}\psi_i^h\|_V^2 < \infty \text{ for any } u \in \mathcal{U}_{\mathsf{ad}}. \quad (10)$$

Note that this error estimate for the reduced state solution $\mathcal{S}^{h\ell}u$ is only valid for the particular control $u \in \mathcal{U}_{\mathsf{ad}}$ that was used to generate the reduced-order model. However, we are interested in estimating the error for an arbitrarily given admissible control, and this will be addressed in the following section. Similarly to the reduced cost function $\hat{J}^h$, we introduce the *reduced-order cost function*

$$\hat{J}^{h\ell} : \mathcal{U} \to \mathbb{R}^3, \quad \hat{J}^{h\ell}(u) = J(\mathcal{S}^{h\ell}u, u) = \frac{1}{2}\begin{pmatrix} \|(\mathcal{S}^{h\ell}u)(T) - y_{d,1}\|_H^2 \\ \|(\mathcal{S}^{h\ell}u)(T) - y_{d,2}\|_H^2 \\ |u|_2^2 \end{pmatrix}.$$

3.2 Error Estimation

In this section, we will present an error estimator similar to (10) but for an arbitrary control $u \in \mathcal{U}_{\mathsf{ad}}$ which was not necessarily used to build the reduced-order model.

Theorem 4 ([22]). *Let a finite-dimensional subspace V^ℓ be given as described in the previous section and $u \in \mathcal{U}_{\mathsf{ad}}$ an arbitrary admissible control. Define the state and reduced state solutions as $y^h = \mathcal{S}^h u$ and $y^{h\ell} = \mathcal{S}^{h\ell} u$. Then the following a-posteriori estimate for the state holds:*

$$\|y^h(t) - y^{h\ell}(t)\|_H^2 + \int_0^t \|y^h(s) - y^{h\ell}(s)\|_V^2 \,\mathrm{d}s \le \Delta^{\mathsf{pr}}(t, y^{h\ell}) \quad \textit{for all } t \in [0,T] \quad (11)$$

with the a-posteriori estimator

$$\Delta^{\mathsf{pr}}(t, y^{h\ell}) = e^{2t}\left(\|y_0 - y_0^{h\ell}\|_H^2 + \int_0^t \|R^{h\ell}(s)\|_{V'}^2 \,\mathrm{d}s \right),$$

where the residual term is defined for $t \in [0,T]$ and $\varphi^h \in V^h$ as:

$$\langle R^{h\ell}(t), \varphi^h \rangle_{V' \times V} = \int_\Omega y_t^{h\ell}(t)\varphi^h + \nabla y^{h\ell}(t) \cdot \nabla \varphi^h + y^{h\ell}(t)^3 \varphi^h \, \mathrm{d}x - \sum_{i=1}^m u_i \int_\Omega \chi_i \varphi^h \, \mathrm{d}x.$$

From this, we can immediately derive an estimator for the cost function:

Lemma 1. *Let $V^{h\ell}$ be introduced as in the previous section and $u \in \mathcal{U}_{\mathsf{ad}}$ an arbitrary admissible control. Then the following a-posteriori estimate for the cost function holds for $i = 1,2$:*

$$\left| \hat{J}_i^h(u) - \hat{J}_i^{h\ell}(u) \right| \leq \sqrt{2\Delta^{\mathsf{pr}}(T, y^{h\ell}) \hat{J}^{h\ell}(u)} + \frac{1}{2}\Delta^{\mathsf{pr}}(T, y^{h\ell}). \tag{12}$$

Proof. We fix $i \in \{1,2\}$ and observe using very basic estimations:

$$\begin{aligned}
\left| \hat{J}_i^h(u) - \hat{J}_i^{h\ell}(u) \right| &= \frac{1}{2} \left| \|y^h(T) - y_{d,i}\|_H^2 - \|y^{h\ell}(T) - y_{d,i}\|_H^2 \right| \\
&= \frac{1}{2} \left| \|y^h(T) - y_{d,i}\|_H + \|y^{h\ell}(T) - y_{d,i}\|_H \right| \\
&\quad \cdot \left| \|y^h(T) - y_{d,i}\|_H - \|(y^{h\ell}(T) - y_{d,i}\| \right| \\
&\leq \frac{1}{2} \left(\|y^h(T) - y_{d,i}\|_H + \|y^{h\ell}(T) - y_{d,i}\|_H \right) \cdot \|y^h(T) - y^{h\ell}(T)\|_H \\
&\leq \frac{1}{2} \left(2\|y^{h\ell}(T) - y_{d,i}\|_H + \|y^h(T) - y^{h\ell}(T)\|_H \right) \cdot \|y^h(T) - y^{h\ell}(T)\|_H \\
&= \sqrt{2\hat{J}^{h\ell}(u)} \cdot \|y^h(T) - y^{h\ell}(T)\|_H + \frac{1}{2}\|y^h(T) - y^{h\ell}(T)\|_H^2.
\end{aligned}$$

The term $\|y^h(T) - y^{h\ell}(T)\|_H^2$ can be further bounded by the state estimator (11), and this yields (12). □

We can observe here that $y^{h\ell} \in H^1(0,T;V)$, so both $y^{h\ell}(t)$ and $y_t^{h\ell}(t)$ in fact belong to the space V. However, it is analytically necessary to consider these terms as elements in V'. Therefore, the dual norm of $R(t)$ is to be understood in the canonical way:

$$\|R^{h\ell}(t)\|_{V'} = \max_{\varphi^h \in V^h \setminus \{0\}} \frac{\left| \langle R^{h\ell}(t), \varphi^h \rangle_{V' \times V} \right|}{\|\varphi^h\|_V}. \tag{13}$$

At this point, we would like to consider the numerical realization in more detail. We define the *mass matrix* $M \in \mathbb{R}^{n \times n}$ and *stiffness matrix* $A \in \mathbb{R}^{n \times n}$ by

$$M_{ij} = \int_\Omega \phi_i \phi_j \, \mathrm{d}x, \quad A_{ij} = \int_\Omega \nabla \phi_i \cdot \nabla \phi_j \, \mathrm{d}x \qquad (i,j = 1,...,n).$$

Let us express the functions $y^{h\ell}(t)$, φ^h and χ_i in the definition of $R^{h\ell}(t) \in (V^h)'$ in terms of the finite element basis $\{\phi_1, ..., \phi_n\}$ via

$$y^{h\ell}(t) = \sum_{i=1}^{n} a_i^\ell(t)\phi_i, \quad \varphi^h = \sum_{i=1}^{n} b_i\phi_i, \quad \chi_i = \sum_{i=1}^{n} c_i\phi_i$$

with coordinate vectors $a^\ell(t)$, b, $c \in \mathbb{R}^n$. We can then express the quotient in (13) as follows:

$$\frac{\left|\langle R^{h\ell}(t), \varphi^h\rangle_{V'\times V}\right|}{\|\varphi^h\|_V} = \frac{\left|b^\top d(t)\right|}{\sqrt{b^\top (M+A)b}} \quad \text{for } t \in [0,T]$$

with $d(t) := Ma_t^\ell(t) + Aa^\ell(t) + M(a^\ell(t))^3 - \sum_{i=1}^m u_i Mc_i$. We can further conclude

$$\begin{aligned}\frac{\left|\langle R(t), \varphi^h\rangle_{V'\times V}\right|}{\|\varphi^h\|_V} &= \frac{\left|b^\top (M+A)^{1/2}(M+A)^{-1/2}d(t)\right|}{\sqrt{b^\top (M+A)b}} \\ &\leq \frac{|(M+A)^{1/2}b|_2\,|(M+A)^{-1/2}d(t)|_2}{\sqrt{((M+A)^{1/2}b)^\top((M+A)^{1/2}b)}} \leq |(M+A)^{-1/2}d(t)|_2,\end{aligned}$$

where $(M+A)^{1/2}$ is the square root of the symmetric positive definite matrix $M+A$. Summarizing, we have derived for any $t \in [0,T]$

$$\|R(t)\|_{(V^h)'} \leq \left|(M+A)^{-1/2}\left(Ma_t^\ell(t) + Aa^\ell(t) + M(a^\ell(t))^3 - \sum_{i=1}^m u_i Mc_i\right)\right|_2,$$

which is used in practice to compute $\Delta^{\mathsf{pr}}(\cdot, y^{h\ell}(\cdot))$.

Remark 3. The estimator $\Delta^{\mathsf{pr}}(\cdot, y^{h\ell}(\cdot))$ from (11) was derived in [22] as a so-called rigorous estimator. As a result, it was assumed during the proof that the unknown true error always behaves according to the worst-case scenario. In a concrete application, it is therefore often observed that this estimator overshoots the true error by a roughly constant factor. Depending on the application, it may be desirable to increase the tightness of the estimator by heuristically incorporating data from a precomputed offline phase. Since this is the case in our application of inexact multiobjective optimization, we will introduce such a heuristic in Section 4.1. However, it has to be noted that the mathematical rigor that is prevalent at this point will be lost by doing this. ◊

4 Numerical Algorithm and Results

In this section, we will combine the results from Sections 2.2 and 3.2 in order to efficiently and globally solve PDE-constrained MOCPs with set-oriented techniques. To this end, the error estimates will be tightened by a heuristic factor and then concepts from the localized reduced basis method will be adapted to the multiobjective setting.

As the domain we consider the unit square $\Omega = (0,1)^2$ and the time interval $[0,T] = [0,1]$. The desired states are given by

$$y_{d,1}(x) = \begin{cases} 0.5, & x_2 \leq 0.5, \\ 0.3, & x_2 > 0.5, \end{cases} \quad y_{d,2}(x) = \begin{cases} -0.5, & x_1 \leq 0.5, \\ 0.5, & x_1 > 0.5, \end{cases}$$

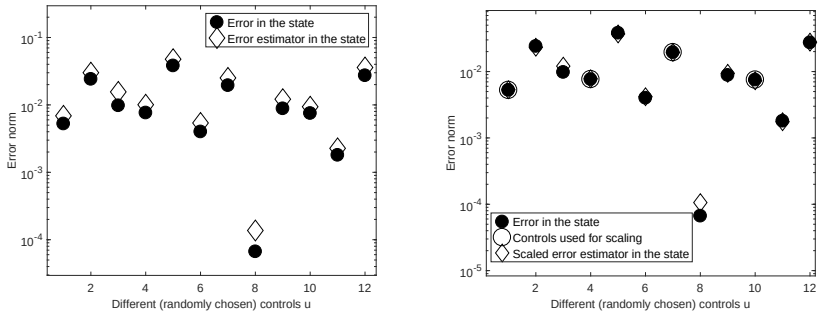
such that there are both conflicting and non-conflicting areas in the domain. The subdomains are given by

$$\Omega_1 = [0,0.5] \times [0,0.5], \quad \Omega_2 = [0,0.5] \times (0.5,1],$$
$$\Omega_3 = (0.5,1] \times [0,0.5], \quad \Omega_4 = (0.5,1] \times (0.5,1].$$

The initial condition is $y_0(x) = 0$ for all $x \in \Omega$ and we allow controls for the constraints $u_{\mathsf{b}} = (1,1,1,1)^\top$ and $u_{\mathsf{a}} = -u_{\mathsf{b}}$.

4.1 Error Estimation

It has been mentioned in Remark 3 that the error estimator (11) often tends to constantly overshoot the true error, a fact that can be observed for the given setup in Figure 3. We would like to improve the tightness of the estimator by using a heuris-



(a) (b)

Fig. 3 (a) Error $\|y^h(T) - y^{h\ell}(T)\|_H^2$ in the state and error estimator $\Delta^{\mathsf{pr}}(T, y^{h\ell}(T))$ for randomly chosen controls. (b) Error $\|y^h(T) - y^{h\ell}(T)\|_H^2$ in the state and scaled error estimator $\Delta_{\mathsf{sc}}^{\mathsf{pr}}(T, y^{h\ell}(T))$ for the same controls.

tic. We assume an over-estimation of the true error of the form

$$\Delta^{\mathsf{pr}}(T, y^{h\ell}) \approx C_{\mathsf{sc}} \cdot \|y^h(T) - y^{h\ell}(T)\|_H^2,$$

where $C_{\mathsf{sc}} > 1$ is an unknown scaling factor. Let a finite sample set $\mathcal{U}_{\mathsf{sc}} \subset \mathcal{U}_{\mathsf{ad}}$ be given prior to the optimization. For a given reduced-order model, we compute the full and reduced state solutions $y_u = \mathcal{S}^h u$, $y_u^\ell = \mathcal{S}^{h\ell} u$ for all sample controls $u \in \mathcal{U}_{\mathsf{sc}}$ and set C_{sc} to be the geometric average

$$C_{\mathsf{sc}} = \left(\prod_{u \in \mathcal{U}_{\mathsf{sc}}} \frac{\Delta^{\mathsf{pr}}(T, y_u^{h\ell}(T))}{\|y_u^h(T) - y_u^{h\ell}(T)\|_H^2} \right)^{\frac{1}{|\mathcal{U}_{\mathsf{sc}}|}}.$$

Having thus obtained a heuristic correctional scaling factor, we replace the error estimator by

$$\Delta_{\mathsf{sc}}^{\mathsf{pr}}(t, y^{h\ell}(t)) = \frac{\Delta^{\mathsf{pr}}(t, y^{h\ell}(t))}{C_{\mathsf{sc}}}.$$

We illustrate the result of this procedure for randomly chosen controls in Figure 3. It can be observed that on the one hand, we lose a rigorous estimate, i.e. it may happen that $\Delta_{\mathsf{sc}}^{\mathsf{pr}}$ is smaller than the true error. On the other hand, $\Delta_{\mathsf{sc}}^{\mathsf{pr}}$ is significantly tighter than Δ^{pr} itself.

In Figure 4, we present some function values for the reduced-order and high-fidelity function evaluations. The error margin indicated by the estimator Δ^J is so small that it is not visible. When zooming in, we can see that the high-fidelity cost function lies roughly within the error bar. It is therefore apparent that the performance of the estimator is satisfactory for the purposes at hand.

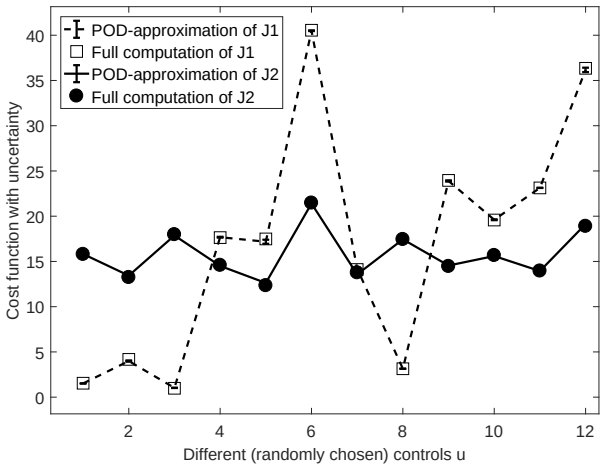


Fig. 4 For the random controls in Figure 3, these are the cost function values $\hat{J}_{1/2}^h(u)$ and $\hat{J}_{1/2}^{h\ell}(u)$. Unvisible due to size, the reduced-order cost function plots are accomponied by an error bar of size $\Delta^J(u)$.

4.2 Localized Reduced Bases: A Numerical Algorithm

Using the tightened error bounds, we are now in the position to address problem ($\hat{\mathbf{P}}$) both with a finite element discretization and a POD approach. In the FEM approach, we simply evaluate the discretized state equation at each sample point and then determine the value of the cost functionals $\hat{J}_1^h$ to $\hat{J}_3^h$. In the POD approach, we have to ensure that the error estimator Δ_{sc}^J is less than a prescribed bound $\Delta_{\mathsf{sc,max}}^J \in \mathbb{R}^k$ everywhere in the parameter domain. If we want to achieve this goal with a single ROM, the model may become high-dimensional in order to satisfy the error bounds everywhere and hence, inefficient. We therefore adopt ideas from localized reduced basis methods [1, 18] and construct a library of locally valid models during the subdivision procedure.

Before the first computation, we determine the factor C_{sc} using several ROMs at randomly distributed controls $u_{\mathsf{ref}} \in \mathcal{U}_{\mathsf{sc}}$ within the parameter domain. In each of these points, the factor between the true error and the error estimator is computed and we set C_{sc} as the mean value of these computations. Note that in our example, this factor was ≈ 2.5 everywhere in the parameter domain such that this approach is justified. The library $\mathscr{L}$ is initialized by constructing a ROM for each of these FEM solutions. This library can grow or shrink during the subdivision algorithm.

In each subdivision step, all sample points (the indices of which are contained in the set $\mathscr{N}$) are evaluated using the *closest* ROM, where the distance is defined via the Euclidean distance between the control u and the reference control u_{ref}^j at which the j^{th} ROM was created. In the beginning, all points are denoted as *insufficiently approximated*:

$$\mathscr{I} = \left\{ i \in \mathscr{N} \;\middle|\; \Delta_{\mathsf{sc}}^J(u^i) \not\leq \Delta_{\mathsf{sc,max}}^J \right\},$$

meaning that they have not yet been approximated well enough using a ROM. After the evaluation of $\hat{J}^\ell$, all points with a satisfactory error estimate according to (12) are eliminated from $\mathscr{I}$. Since the remaining points violate the desired error bound $\Delta_{\mathsf{sc,max}}^J \in \mathbb{R}^k$, we evaluate the full model and add a ROM to the library $\mathscr{L}$. This is done in a *greedy* way (see also [9]), i.e. we add the ROM at the point with the maximum error. The above steps are repeated until all points are approximated sufficiently accurately and consequently, the set $\mathscr{I}$ is empty. Finally, all ROMs are removed from $\mathscr{L}$ which have not been used. This is done in order to keep the number of locally valid ROMs at an acceptable number. Moreover, ROMs belonging to regions in the parameter domain which have been identified as dominated will not be required any further. The procedure is summarized in Algorithm 3. We want to emphasize that the approach presented here is only a first step towards using local reduced bases within multiobjective optimization. We expect that the efficiency can be further increased by implementing more sophisticated rules for clustering the points than using the Euclidean distance. Moreover, we expect that making use of online enrichment (see e.g. [1, 18]) or a combination of different bases will be beneficial for the overall performance.

Algorithm 3 (Greedy localized reduced basis approach)

Require: $\Delta^J_{\mathsf{sc,max}} \in \mathbb{R}^k$, C_{sc}, set of sample points $\mathscr{N} \subset \mathcal{U}_{\mathsf{ad}}$;
1: Consider all sample points as *insufficiently approximated*, i.e. $\mathscr{I} = \mathscr{N}$;
2: **while** $\mathscr{I} \neq \emptyset$ **do**
3: **for** $i = 1, \ldots, |\mathscr{I}|$ **do**
4: Identify the *closest* ROM with respect to the 2-norm:

$$\hat{\imath} = \operatorname*{arg\,min}_{j \in \{1,\ldots,|\mathscr{L}|\}} \left| u^i - u^j_{\mathsf{ref}} \right|_2.$$

5: Compute $\hat{J}^{h\ell}(u^i)$ using ROM $\hat{\imath}$;
6: Evaluate the error $\Delta^J_{\mathsf{sc}}(u^i)$ for ROM $\hat{\imath}$ using (12);
7: **if** $\Delta^J_{\mathsf{sc}}(u^i) \leq \Delta^J_{\mathsf{sc,max}}$ **then**
8: Accept $\hat{J}^{h\ell}(u^i)$ as sufficiently accurate;
9: Remove i from the set $\mathscr{I}$;
10: Identify the sample point with the largest error:

$$i_{\mathsf{max}} = \operatorname*{arg\,max}_{s \in \mathscr{I}} \Delta^J_{\mathsf{sc}}(u^s)$$

11: Add ROM to library $\mathscr{L}$ with $u_{\mathsf{ref}} = u^{i_{\mathsf{max}}}$;
12: Remove all ROMs from $\mathscr{L}$ that have not been used;

4.3 Numerical Results

In this section we compare numerical results for the MOP ($\hat{\mathbf{P}}$) obtained by a finite element discretization (Algorithm 1) as well as a reduced-order model (Algorithm 2), where the sample points in each subdivision step are evaluated using Algorithm 3.

Each box B is represented by an equidistant grid of two points in each direction, i.e. by 16 sample points in total. The exact Pareto set of ($\hat{\mathbf{P}}$) is shown in Figure 5 (a), where the boxes are colored according to the fourth component u_4. The corresponding Pareto front is given in Figure 5 (c) in green. The Pareto set for the same problem, obtained by Algorithms 2 and 3, is shown in Fig. 5 (b), the corresponding Pareto front in Figure 5 (c) in red. We observe a good agreement both between the Pareto sets and the Pareto fronts. The error bound $\Delta^J_{\mathsf{sc,max}}$ for the objectives $\hat{J}^h_1$ and $\hat{J}^h_2$ is satisfied as desired. In order to also bound the error in the decision space, further assumptions on the objectives have to be made.

For the inexact Pareto set, only 444 evaluations of the full model were required, i.e., the FEM evaluations were reduced by a factor of more than 1000. The FEM evaluations are mainly performed during the first subdivision steps (cf. Figure 6 (a)) whereas later, almost the entire decision space can be approximated by the existing models. This is also illustrated in Figure 6 (b), where we observe an exponential increase of the ratio of high-fidelity solutions between the FEM and the POD-based approach. Therefore, it might be interesting to investigate the benefit of an offline phase similar to classical reduced basis approaches.

Due to the inexactness, the number of boxes is larger in the inexact computation, which is shown in Fig. 6 (c). Similar to the observations in [19], this effect becomes

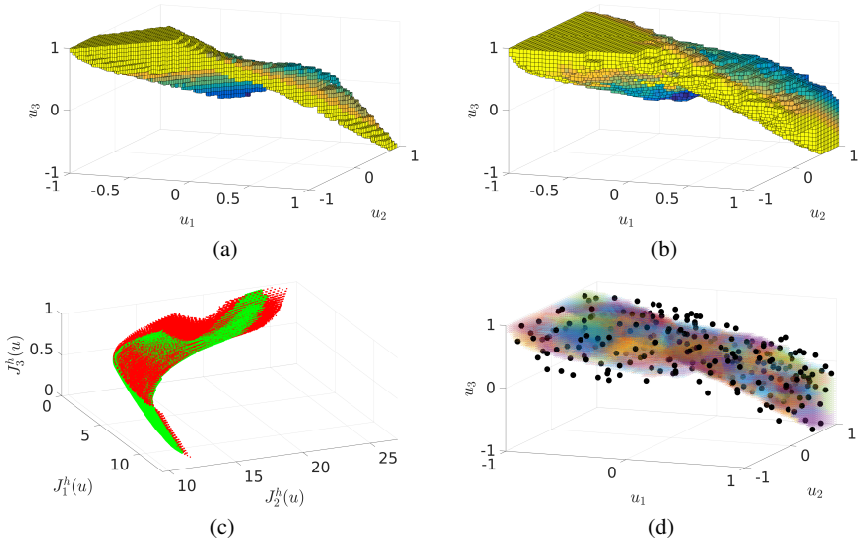


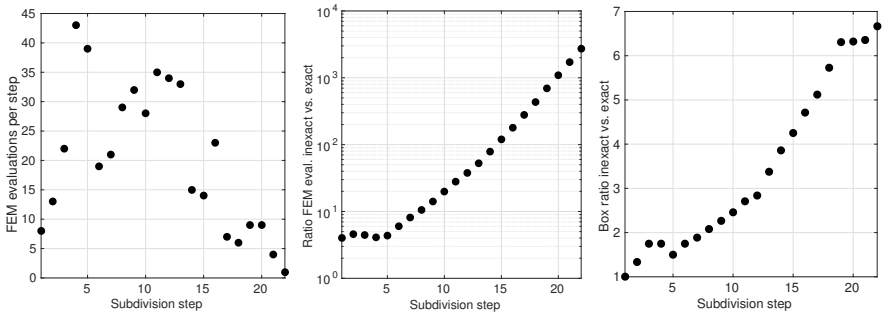
Fig. 5 (a) Pareto set of ($\hat{\mathbf{P}}$) after 22 subdivision steps using a finite element discretization. Projection onto the first three components of u, and u_4 is visualized by the box coloring. (b) The Pareto set based on POD reduced-order models and the inexact sampling algorithm with $\Delta^J_{\mathsf{sc,max}} = (0.025, 0.025, 0)^\top$. (c) The corresponding Pareto fronts, where the FEM-based solution is depicted in green and the POD-based solution in red. The points are the images of the box centers. (d) Clustering of the sample points. Each of the colored patches has been assigned to one ROM, which are represented by black dots.

more severe during higher order subdivision steps. All boxes in the vicinity of the exact Pareto set are not eliminated which results in an exponential increase in the number of boxes. Consequently, further research is required to address this issue.

In Figure 5 (d), the locations of the reference controls u_{ref} of the (remaining) ROMs are shown as black dots. The remaining points depict the sample points that were evaluated in the 18^{th} subdivision step, where the coloring depends on the ROM the point was assigned to. Due to the fact that the selection is being based on the Euclidean distance the size of all the patches is comparable. Since there is no formal reason to choose this specific way of assigning sample points to ROMs, this motivates the investigation of more advanced clustering approaches in order to further reduce the number of required ROMs.

5 Conclusion and Outlook

In this chapter we have applied an inexact version of the gradient-free subdivision algorithm from [19] to a multiobjective optimal control problem with a semilinear parabolic state equation. The inexactness was in particular given by employing



(a) (b) (c)

Fig. 6 (a) The number of FEM evaluations in each subdivision step. (b) Ratio of the total number of high-fidelity evaluations within the POD-based inexact subdivision algorithm 2 and the FEM-based exact subdivision algorithm 1. (c) Ratio of the respective number of boxes.

POD model-order reduction to the state equation in order to speed up cost function evaluations. Due to the existence of analytical error estimates for the resulting state error, we were able to present an a-posteriori estimator for the cost function error in Section 3. The maximal error for the reduced-order model is given a-priori by the problem definition. In order to ensure the necessary accuracy, we employ a localized-bases strategy. The ROMs are constructed iteratively using a Greedy worst-first search which is described in Section 4. Our numerical results for a simple test problem confirm that the reduced-order approach is able to qualitatively approximate both the Pareto front and the Pareto set. The Pareto front is captured accurately according the prescribed error bounds on the cost function values. Furthermore, the error estimator for the state variable and the cost function values are coupled with a simple heuristic to compensate for a constant over-estimation factor.

It has to be mentioned that there is room for improvement to the presented algorithm. Due to the difference in the Definitions 4 (exact dominance) and 5 (inexact dominance), it is a natural consequence that the inexact algorithm can eliminate fewer boxes in each iteration. This makes it necessary to perform more function eliminations in the next iteration than for the exact algorithm. It may therefore be beneficial to develop alternative or additional criteria in order reduce the number of non-dominated boxes.

As far as the construction of reduced-order models is concerned, it has to be stated that the current strategy only follows the heuristic that controls which lie close to each other in a geometrical sense may share similar state and cost function behavior. This straight-forward approach may result in an inefficiently large number of reduced-order models since there may be more distant controls which are still sufficiently well approximated by the given reduced model. As a result, more sophisticated clustering techniques in the control space need to be developed which result in fewer, more cleverly spread reduced models for the state equation.

Let us mention that there is also a gradient-based version of Algorithm 1 [19] based on a dynamical systems approach. In order to combine model-order reduc-

tion and in particular error estimation with this method, we would have to estimate the error between the gradients $\nabla \hat{f}_i^h(u)$ and the reduced-order gradients $\nabla \hat{f}_i^{h\ell}(u)$ by using an a-posteriori estimator for the *adjoint equation* to the system (1).

Acknowledgement

This work is supported by the Priority Programme SPP 1962 *Non-smooth and Complementarity-based Distributed Parameter Systems* of the German Research Foundation (DFG) and by the project *Hybrides Planungsverfahren zur energieeffizienten Wärme- und Stromversorgung von städtischen Verteilnetzen* funded by the German Ministry for Economic Affairs and Energy.

References

1. F. Albrecht, B. Haasdonk, S. Kaulmann, and M. Ohlberger. The localized reduced basis multiscale method. In *Proceedings of ALGORITHMY 2012*, pages 393–403, 2012.
2. M. N. Albunni, V. Rischmuller, T. Fritzsche, and B. Lohmann. Multiobjective Optimization of the Design of Nonlinear Electromagnetic Systems Using Parametric Reduced Order Models. *IEEE Transactions on Magnetics*, 45(3):1474–1477, 2009.
3. S. Banholzer, D. Beermann, and S. Volkwein. POD-Based Bicriterial Optimal Control by the Reference Point Method. In *2nd IFAC Workshop on Control of Systems Governed by Partial Differential Equations*, pages 210–215, 2016.
4. S. Banholzer, D. Beermann, and S. Volkwein. POD-based error control for reduced-order bicriterial PDE-constrained optimization. *http://nbn-resolving.de/urn:nbn:de:bsz:352-0-394180*, 2017. Submitted.
5. C. A. Coello Coello, G. B. Lamont, and D. A. Van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems*, volume 2. Springer Science & Business Media, 2007.
6. M. Dellnitz and A. Hohmann. A subdivision algorithm for the computation of unstable manifolds and global attractors. *Numerische Mathematik*, 75(3):293–317, 1997.
7. M. Dellnitz, O. Schütze, and T. Hestermeyer. Covering Pareto sets by multilevel subdivision techniques. *Journal of Optimization Theory and Applications*, 124(1):113–136, 2005.
8. M. Ehrgott. *Multicriteria Optimization*. Springer Berlin Heidelberg New York, second edition, 2005.
9. M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(3):575–605, 2007.
10. M. Gubisch and S. Volkwein. Proper orthogonal decomposition for linear-quadratic optimal control. *To appear in P. Benner, A. Cohen, M. Ohlberger, and K. Willcox (eds.), Model Reduction and Approximation: Theory and Algorithms. SIAM, Philadelphia, PA*, 2017.
11. C. Hillermeier. *Nonlinear Multiobjective Optimization: A Generalized Homotopy Approach*. Birkhäuser, 2001.
12. P. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge Monographs on Mechanics. Cambridge University Press, Cambridge, 2nd ed., 2012.
13. L. Iapichino, S. Trenz, and S. Volkwein. Reduced-order multiobjective optimal control of semilinear parabolic problems. In *Numerical Mathematics and Advanced Applications (ENUMATH 2015)*, volume 112 of *Lecture Notes in Computational Science and Engineering*, pages 389–397. Springer Switzerland, 2016.

14. L. Iapichino, S. Ulbrich, and S. Volkwein. Multiobjective PDE-constrained optimization using the reduced-basis method. *http://kops.uni-konstanz.de/handle/123456789/25019*, 2013.
15. H. W. Kuhn and A. W. Tucker. Nonlinear programming. In *Proceedings of the 2nd Berkeley Symposium on Mathematical and Statsitical Probability*, pages 481–492. University of California Press, 1951.
16. K. Miettinen. *Nonlinear Multiobjective Optimization*. Springer Science & Business Media, 2012.
17. J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer Science & Business Media, second edition, 2006.
18. M. Ohlberger and F. Schindler. Error control for the localized reduced basis multiscale method with adaptive on-line enrichment. *SIAM Journal on Scientific Computing*, 37(6):2865–2895, 2015.
19. S. Peitz and M. Dellnitz. Gradient-Based Multiobjective Optimization with Uncertainties. In *NEO 2016 Proceedings (to appear, preprint: arXiv:1612.03815)*. Springer, 2017.
20. S. Peitz, S. Ober-Blöbaum, and M. Dellnitz. Multiobjective Optimal Control Methods for Fluid Flow Using Model Order Reduction. *arXiv:1510.05819*, 2015.
21. N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and K. P. Tucker. Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, 41:1–28, 2005.
22. S. Rogg, S. Trenz, and S. Volkwein. Trust-Region POD using A-Posteriori Error Estimation for Semilinear Parabolic Optimal Control Problems. *https://kops.uni-konstanz.de/handle/123456789/38240*, 2017.
23. S. Schäffler, R. Schultz, and K. Weinzierl. Stochastic Method for the Solution of Unconstrained Vector Optimization Problems. *Journal of Optimization Theory and Applications*, 114(1):209–222, 2002.
24. W. H. A. Schilders, H. A. van der Vorst, and J. Rommes. *Model Order Reduction*. Springer Berlin Heidelberg, 2008.
25. O. Schütze. A New Data Structure for the Nondominance Problem in Multi-objective Optimization. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 509–518. Springer Berlin Heidelberg, 2003.
26. O. Schütze, K. Witting, S. Ober-Blöbaum, and M. Dellnitz. Set Oriented Methods for the Numerical Treatment of Multiobjective Optimization Problems. In E. Tantar, A.-A. Tantar, P. Bouvry, P. Del Moral, P. Legrand, C. A. Coello Coello, and O. Schütze, editors, *EVOLVE - A Bridge between Probability, Set Oriented Numerics and Evolutionary Computation*, volume 447 of *Studies in Computational Intelligence*, pages 187–219. Springer Berlin Heidelberg, 2013.
27. J. R. Singler. New POD error expressions, error bounds, and asymptotic results for reduced order models of parabolic PDEs. *SIAM J. Numer. Anal.*, 52:852–876, 2014.
28. F. Tröltzsch. Optimal Control of Partial Differential Equations. *Graduate studies in mathematics*, 112, 2010.