DFG Deutsche Forschungsgemeinschaft
Priority Programme 1962

POD-Based Error Control for Reduced-Order Bicriterial PDE-Constrained Optimization

Stefan Banholzer, Dennis Beermann, Stefan Volkwein

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

Preprint Number SPP1962-015

received on April 11, 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/

POD-Based Error Control for Reduced-Order Bicriterial PDE-Constrained Optimization

Stefan Banholzer, Dennis Beermann, Stefan Volkwein

Department of Mathematics and Statistics, University of Konstanz
Universitätsstraße 10, D-78464 Konstanz

Abstract

In the present paper, a bicriterial optimal control problem governed by an abstract evolution problem and bilateral control constraints is considered. To compute Pareto optimal points and the Pareto front numerically, the (Euclidean) reference point method is applied, where many scalar constrained optimization problems have to be solved. For this reason a reduced-order approach based on proper orthogonal decomposition (POD) is utilized. An a-posteriori error analysis ensures a desired accuracy for the Pareto optimal points and for the Pareto front computed by the POD method. Numerical experiments for evolution problems with convection-diffusion illustrate the efficiency of the presented approach.

Keywords: Bicriterial PDE-constrained optimization, reference point method, optimal control, proper orthogonal decomposition, error analysis.

1. Introduction

In real applications, optimization problems are often described by introducing several objective functions conflicting with each other. This leads to *multiobjective* or *multicriterial* optimization problems; cf. [4, 12, 16]. One prominent example is given by an energy efficient heating, ventilation and air-conditioning (HVAC) operation of a building with conflicting objectives such as minimal energy consumption and maximal comfort; cf. [5, 11]. Finding the optimal control that represents a good compromise is the main issue in these problems. For that reason the concept of Pareto optimal or efficient points is developed. In contrast to scalar-valued optimization problems, the computation of a set of Pareto optimal points is required. Consequently, many scalar-valued constrained optimization problems have to be solved.

In this paper we apply the reference point method [14] in order to transform a bicriterial optimal control problem into a sequence of scalar-valued optimal control problems and solve them using well-known optimal control techniques; see

Email address: Stefan.Volkwein@uni-konstanz.de (Stefan Volkwein)

[17]. We extend our previous results obtained in [2] with respect to the following issues: Instead of the linear heat equation, general linear evolution problems are considered. Therefore, we can deal with parabolic problems involving convection which arise in HVAC operation of building applications. Further, we improve the a-posteriori error analysis for the control variable (cf. [2, Theorem 7] and present an error estimate for the Pareto front in Theorem 13. An a-priori error for the objective is stated in Theorem 14. Moreover, in our numerical experiments we illustrate that our a-posteriori error bounds can be ensured numerically. In particular, an adaptive basis update method known from scalar optimization [18] is extended to a bicriterial optimization problem. It is also shown how the Pareto front can be approximated numerically by equidistantly computed points.

Many results are derived by combining the POD error analysis presented in [6] with the master thesis [1]. Let us mention that preliminary results combining reduced-order modeling and multiobjective PDE-constrained optimization have recently been derived; cf. [8, 9, 13].

The paper is organized in the following manner: In Section 2 we introduce our linear evolution equation as well as our bicriterial optimization problem. The reference point method is recalled in Section 3. It turns out that the Pareto front is approximated by solving many scalar optimization problems. To speed-up the solution process a reduced-order approach is introduced in Section 4, where we utilize proper orthogonal decomposition (POD). This allows us to present convergence results. In Section 5 the numerical optimization and the reduced-order approach is explained in detail for our present problem. Numerical experiments are presented in Section 6.

Notation: Throughout this paper, if $x^1, x^2 \in \mathbb{R}^k$ are two vectors, we write $x^1 \leq x^2$ if $x_i^1 \leq x_i^2$ for $i = 1, ..., k$, and similarily for $x^1 \geq x^2$. For $x \in \mathbb{R}^k$ we define $\mathbb{R}^k_{\geq x} = \{y \in \mathbb{R}^k \,|\, y \geq x\}$ and $\mathbb{R}^k_{\leq x} = \{y \in \mathbb{R}^k \,|\, y \leq x\}$. For convenience, we set $\mathbb{R}^k_{\leq 0} = \mathbb{R}^k_{\leq}$.

2. Problem formulation

2.1 The state equation. Let V and H be real, separable Hilbert spaces and suppose that V is dense in H with compact embedding. By $\langle \cdot, \cdot \rangle_H$ and $\langle \cdot, \cdot \rangle_V$ we denote the inner products in H and V, respectively. By identifying H with its dual H' it follows that $V \hookrightarrow H = H' \hookrightarrow V'$, each embedding being continuous and dense. Let $T > 0$ denote the terminal time. Suppose that $a : V \times V \to \mathbb{R}$ is a bilinear form satisfying

$$\left|a(\varphi, \psi)\right| \leq \eta \left\|\varphi\right\|_V \left\|\psi\right\|_V \quad \text{and} \quad a(\varphi, \varphi) \geq \eta_1 \left\|\varphi\right\|_V^2 - \eta_2 \left\|\varphi\right\|_H^2 \quad \forall \varphi, \psi \in V \tag{1}$$

with constants $\eta \geq 0$, $\eta_1 > 0$ and $\eta_2 \geq 0$. Recall that the space

$$W(0,T) = \left\{\varphi \in L^2(0,T;V) \,\middle|\, \varphi_t \in L^2(0,T;V')\right\}$$

is a Hilbert space endowed with the common inner product; see, e.g., [3, p. 473]. It is well-known that $W(0,T)$ is continuously embedded into $C([0,T];H)$, the

space of all continuous functions from $[0,T]$ to H [3, p. 480]. Let $\mathcal{D}$ be an open and bounded subset of $\mathbb{R}^d$ with $d \in \mathbb{N}$ and $\mathcal{U} = L^2(\mathcal{D};\mathbb{R}^m)$. By $\mathcal{U}_{\mathsf{ad}} \subset \mathcal{U}$ we define the closed, convex and bounded subset

$$\mathcal{U}_{\mathsf{ad}} = \big\{ u \in \mathcal{U} \,\big|\, u_a(s) \le u(s) \le u_b(s) \text{ in } \mathbb{R}^m \text{ for almost all (f.a.a.) } s \in \mathcal{D} \big\}$$

with $u_a, u_b \in \mathcal{U}$ satisfying $u_a \le u_b$ almost everywhere (a.e.) in $\mathcal{D}$. We suppose that $y_\circ \in H$, $f \in L^2(0,T;V')$ and $u \in \mathcal{U}_{\mathsf{ad}}$ hold. Then the *state* $y \in \mathcal{Y} = W(0,T)$ is given by the linear evolution problem

$$\begin{aligned} \tfrac{\mathrm{d}}{\mathrm{d}t}\langle y(t),\varphi\rangle_H + a(y(t),\varphi) &= \langle (f+\mathcal{B}u)(t),\varphi\rangle_{V',V} \quad \forall \varphi \in V, \text{ f.a.a. } t \in (0,T], \\ \langle y(0),\varphi\rangle_H &= \langle y_\circ,\varphi\rangle_H \qquad\qquad\quad \forall \varphi \in V, \end{aligned} \tag{2}$$

where $\mathcal{B} : \mathcal{U} \to L^2(0,T;V')$ is a continuous, linear operator and $\langle \cdot\,,\cdot \rangle_{V',V}$ stands for the dual pairing between V' and V.

Example 1. Let us present an example for (2) which will be considered in our numerical experiments. Suppose that $\Omega \subset \mathbb{R}^{\mathsf{d}}$, $\mathsf{d} \in \{1,2,3\}$, is a bounded domain with Lipschitz-continuous boundary Γ. We assume that Ω and Γ are split into disjoint subsets $\Omega_1,\dots,\Omega_m$ and $\Gamma_1,\dots,\Gamma_{\mathsf{m}}$, respectively, satisfying $\overline{\Omega} = \bigcup_{i=1}^m \overline{\Omega}_i$ and $\overline{\Gamma} = \bigcup_{i=1}^{\mathsf{m}} \overline{\Gamma}_i$. We set $Q = (0,T)\times\Omega$ and $\Sigma_i = (0,T)\times\Gamma_i$ for $i = 1,\dots,m$. Moreover, let $H = L^2(\Omega)$ and $V = H^1(\Omega)$. As the state equation we consider the following diffusion-convection problem:

$$y_t(t,\boldsymbol{x}) - \Delta y(t,\boldsymbol{x}) + \beta(\boldsymbol{x})\cdot\nabla y(t,\boldsymbol{x}) = \sum_{i=1}^m u_i(t)\chi_i(\boldsymbol{x}) \quad \text{for } (t,\boldsymbol{x}) \in Q, \tag{3a}$$

$$1 \le i \le \mathsf{m}: \qquad \frac{\partial y}{\partial \boldsymbol{n}}(t,\boldsymbol{x}) + \alpha_i y(t,\boldsymbol{x}) = \alpha_i y_i^a(t) \qquad \text{for } (t,\boldsymbol{x}) \in \Sigma_i, \tag{3b}$$

$$y(0,\boldsymbol{x}) = y_\circ(\boldsymbol{x}) \qquad \text{for } \boldsymbol{x} \in \Omega. \tag{3c}$$

In (3a), we assume $\beta \in L^\infty(\Omega;\mathbb{R}^{\mathsf{d}})$. Moreover, χ_i stands for the characteristic function of Ω_i for $i = 1,\dots,m$. The control variable $u = (u_1,\dots,u_m)$ is assumed to be in the Hilbert space $\mathcal{U} = L^2(\mathcal{D};\mathbb{R}^m)$ with $\mathcal{D} = (0,T)$. In (3b), the α_i's are nonnegative scalars and $y_i^a \in L^\infty(0,T)$ denotes an essentially bounded outer temperature associated with the boundary set Γ_i for any $i = 1,\dots,\mathsf{m}$. In (3b) the vector $\boldsymbol{n} = \boldsymbol{n}(\boldsymbol{x})$ stands for the normal vector defined on Γ and $\frac{\partial y}{\partial \boldsymbol{n}}$ is the normal derivative. Finally, $y_\circ \in H$ is a given initial condition. We define the bilinear form $a : V \times V \to \mathbb{R}$

$$a(\varphi,\psi) = \int_\Omega \nabla\varphi\cdot\psi + \big(\beta\cdot\nabla\varphi\big)\psi \,\mathrm{d}\boldsymbol{x} + \sum_{i=1}^{\mathsf{m}} \alpha_i \int_{\Gamma_i} \varphi\psi \,\mathrm{d}\boldsymbol{s} \quad \forall \varphi,\psi \in V.$$

It follows by standard arguments that (1) holds; see [1, Lemma 5.1], for instance. Furthermore, $f \in L^2(0,T;V')$ and $\mathcal{B} : \mathcal{U} \to L^2(0,T;V')$ are chosen as

$$\langle f(t),\varphi\rangle_{V',V} = \sum_{i=1}^{\mathsf{m}} \alpha_i y_i^a(t) \int_{\Gamma_i} \varphi \,\mathrm{d}\boldsymbol{s}, \quad \langle (\mathcal{B}u)(t),\varphi\rangle_{V',V} = \sum_{i=1}^m u_i(t) \int_{\Omega_i} \varphi \,\mathrm{d}\boldsymbol{x}$$

for $u \in \mathcal{U}$ and for all $\varphi \in V$ and f.a.a. $t \in [0,T]$. Note that $\mathcal{B}$ is bounded and continuous; cf. [1, Lemma 5.2]. Now the weak formulation for (3) can be expressed in the form (2). ◊

Theorem 2. *Problem* (2) *admits a unique solution* $y \in \mathcal{Y}$. *Moreover, if even* $y_\circ \in V$ *and* $f, \mathcal{B}u \in L^2(0,T;H)$ *hold, we have* $y \in C([0,T];V)$.

Proof. The proof follows from [3, pp. 512-513] and [3, pp. 532-533]. □

Remark 3. Let $\hat{y} \in \mathcal{Y}$ be the unique solution to

$$\begin{aligned} \tfrac{\mathrm{d}}{\mathrm{d}t} \langle \hat{y}(t), \varphi \rangle_H + a(\hat{y}(t), \varphi) &= \langle f(t), \varphi \rangle_{V',V} \qquad && \forall \varphi \in V, \text{ f.a.a. } t \in (0,T], \\ \langle \hat{y}(0), \varphi \rangle_H &= \langle y_\circ, \varphi \rangle_H && \forall \varphi \in V. \end{aligned}$$

Moreover, we introduce the linear and bounded solution operator $\mathcal{S} : \mathcal{U} \to \mathcal{Y}$ as follows $\tilde{y} = \mathcal{S}u$ is the unique solution to

$$\begin{aligned} \tfrac{\mathrm{d}}{\mathrm{d}t} \langle \tilde{y}(t), \varphi \rangle_H + a(\tilde{y}(t), \varphi) &= \langle (\mathcal{B}u)(t), \varphi \rangle_{V',V} \qquad && \forall \varphi \in V, \text{ f.a.a. } t \in (0,T], \\ \langle \tilde{y}(0), \varphi \rangle_H &= 0 && \forall \varphi \in V, \end{aligned}$$

Then, $y = \hat{y} + \mathcal{S}u$ solves (2). ◊

2.2 The bicriterial optimal control problem. Let $\mathcal{H}$ be a real Hilbert space and $\mathcal{G} : \mathcal{Y} \to \mathcal{H}$ a bounded linear operator. We define the operator $\mathcal{S}_\mathcal{H} = \mathcal{G}\mathcal{S} : \mathcal{U} \to \mathcal{H}$. Since $\mathcal{G}$ and $\mathcal{S}$ are linear and bounded, $\mathcal{S}_\mathcal{H}$ is linear and bounded as well. For a given desired state $y_\mathcal{H} \in \mathcal{H}$ we introduce the bicriterial cost functional $\hat{J} : \mathcal{U} \to \mathbb{R}^2$ by

$$\hat{J}(u) = \begin{pmatrix} \hat{J}_1(u) \\ \hat{J}_2(u) \end{pmatrix} = \frac{1}{2} \begin{pmatrix} \|\mathcal{S}_\mathcal{H} u - y_\mathcal{H}\|_\mathcal{H}^2 \\ \|u\|_\mathcal{U}^2 \end{pmatrix} \quad \text{for } u \in \mathcal{U}. \tag{4}$$

Clearly, $\hat{J}_1$ and $\hat{J}_2$ are bounded from below. Furthermore, $\hat{J}_1$ is convex and $\hat{J}_2$ is strictly convex.

Example 4. In the context of Example 1 let $\mathcal{Y} \hookrightarrow L^2(0,T;H) =: \mathcal{H}$ and $\mathcal{G} : \mathcal{Y} \to \mathcal{H}$ the canonical (linear, continuous) embedding operator. It follows that $\mathcal{S}_\mathcal{H}$ is continuous and injective; cf. [1, Lemmas 5.6 and 5.7]. For given $y_Q \in \mathcal{H}$ we introduce the cost functional

$$\hat{J}(u) = \frac{1}{2} \begin{pmatrix} \int_0^T \int_\Omega |y(t,\boldsymbol{x}) - y_Q(t,\boldsymbol{x})|^2 \,\mathrm{d}\boldsymbol{x}\mathrm{d}t \\ \sum_{i=1}^m \int_0^T |u_i(t)|^2 \,\mathrm{d}t \end{pmatrix} \text{ for } u \in \mathcal{U} \text{ and } y = \hat{y} + \mathcal{S}u.$$

Then setting $y_\mathcal{H} = y_Q - \mathcal{G}\hat{y} \in \mathcal{H}$ we can express $\hat{J}$ in the form (4). ◊

Lemma 5. *The objectives $\hat{J}_1$ and $\hat{J}_2$ are twice Fréchet differentiable with the derivatives*

$$\begin{aligned} \hat{J}_1'(u) &= \langle \mathcal{S}_{\mathcal{H}}^\star(\mathcal{S}_{\mathcal{H}}u - y_{\mathcal{H}}), \cdot\rangle_{\mathcal{U}}, & \hat{J}_2'(u) &= \langle u, \cdot\rangle_{\mathcal{U}}, \\ \hat{J}_1''(u)(u^\delta, \cdot) &= \langle \mathcal{S}_{\mathcal{H}}^\star\mathcal{S}_{\mathcal{H}}u^\delta, \cdot\rangle_{\mathcal{U}}, & \hat{J}_2''(u)(u^\delta, \cdot) &= \langle u^\delta, \cdot\rangle_{\mathcal{U}} \quad (u^\delta \in \mathcal{U}) \end{aligned} \tag{5}$$

where $\mathcal{S}_{\mathcal{H}}^\star : \mathcal{H} \to \mathcal{U}$ is the (Hilbert) adjoint operator satisfying

$$\langle \mathcal{S}_{\mathcal{H}}^\star\varphi, u\rangle_{\mathcal{U}} = \langle \varphi, \mathcal{S}_{\mathcal{H}}u\rangle_{\mathcal{H}} \quad \textit{for all } (\varphi, u) \in \mathcal{H} \times \mathcal{U}.$$

If $\mathcal{S}_{\mathcal{H}}$ is injective, $\hat{J}_1$ is strictly convex and the ideal vector $y^{\mathsf{id}} = (y_1^{\mathsf{id}}, y_2^{\mathsf{id}}) \in \mathbb{R}^2$ defined by

$$y_i^{\mathsf{id}} = \min_{u \in \mathcal{U}_{\mathsf{ad}}} \hat{J}_i(u) \quad \textit{for } i = 1, 2.$$

is well-defined.

Proof. Since $\mathcal{H}$, $\mathcal{U}$ are Hilbert spaces and $\mathcal{S}_{\mathcal{H}}$ is linear, continuous, (5) follows directly. Since $\hat{J}_2$ is strictly convex, continuous and bounded from below, y_2^{id} is well-defined; cf. [17, Theorem 2.14]. If $\mathcal{S}_{\mathcal{H}}$ is injective, we infer from (5) and

$$\hat{J}_1''(u)(v, v) = \langle \mathcal{S}_{\mathcal{H}}^\star\mathcal{S}_{\mathcal{H}}v, v\rangle_{\mathcal{U}} = \|\mathcal{S}_{\mathcal{H}}v\|_{\mathcal{H}}^2 > 0 \quad \text{for all } v \in \mathcal{U} \setminus \{0\}.$$

Thus, $\hat{J}_1$ is strictly convex as well, which implies that y_1^{id} is also well-defined. □

Example 6. In the context of Examples 1 and 4 it can be shown that for given $u \in \mathcal{U}_{\mathsf{ad}}$ the derivative of $\hat{J}_1$ takes the form

$$\hat{J}_1'(u)u^\delta = \sum_{i=1}^{m} \int_0^T \Big(\int_{\Omega_i} p(t, \boldsymbol{x}) \,\mathrm{d}\boldsymbol{x} \Big) u_i^\delta(t) \,\mathrm{d}t \quad \text{for every } u^\delta = (u_1^\delta, \dots, u_m^\delta) \in \mathcal{U},$$

where $p = p(u) \in \mathcal{Y}$ is the weak solution to the *adjoint* or *dual equation*

$$\begin{aligned} -p_t - \Delta p - \nabla \cdot (\beta p) + \mathcal{S}_{\mathcal{H}}u &= y_Q - \mathcal{G}\hat{y} && \text{in } Q, \\ 1 \le i \le \mathsf{m}: \ \tfrac{\partial p}{\partial \boldsymbol{n}} + \big(\alpha_i + \beta \cdot \boldsymbol{n}\big)p &= 0 && \text{on } \Sigma_i, \\ p(T, \cdot) &= 0 && \text{in } \Omega; \end{aligned} \tag{6}$$

cf. [17, Section 3.6]. Let $\hat{p} \in W(0,T)$ and $\tilde{p} = \mathcal{A}u \in W(0,T)$ be the weak solutions to $\hat{p}(T) = 0$ in H,

$$\begin{aligned} -\tfrac{\mathrm{d}}{\mathrm{d}t}\langle \hat{p}(t), \varphi\rangle_H + a(\varphi, \hat{p}(t)) &= \langle y_Q(t) - \hat{y}(t), \varphi\rangle_H && \forall \varphi \in V, \text{ f.a.a. } t \in (0,T], \\ -\tfrac{\mathrm{d}}{\mathrm{d}t}\langle \tilde{p}(t), \varphi\rangle_H + a(\varphi, \tilde{p}(t)) &= -\langle (\mathcal{S}u)(t)(t), \varphi\rangle_H && \forall \varphi \in V, \text{ f.a.a. } t \in (0,T], \\ \hat{p}(T) = \tilde{p}(T) &= 0 && \text{in } H, \end{aligned}$$

respectively; cf. Remark 3. Then, $p = \hat{p} + \mathcal{A}u$ is the weak solution to (6). ◊

In this paper we investigate the following bicriterial optimal control problem

$$\min \hat{J}(u) \quad \text{subject to (s.t.)} \quad u \in \mathcal{U}_{\mathsf{ad}}. \tag{$\hat{\mathbf{P}}$}$$

Problem ($\hat{\mathbf{P}}$) involves the minimization of a vector-valued objective. This is done by using the concepts of *order relation* and *Pareto optimality*; see, e.g., [4].

Definition 7. *The point $\bar{u} \in \mathcal{U}_{\mathsf{ad}}$ is called Pareto optimal for $(\hat{\mathbf{P}})$ if there is no other control $u \in \mathcal{U}_{\mathsf{ad}} \setminus \{\bar{u}\}$ with $\hat{J}_i(u) \leq \hat{J}_i(\bar{u})$, $i = 1, 2$, and $\hat{J}_j(u) < \hat{J}_j(\bar{u})$ for at least one $j \in \{1, 2\}$.*

3. The reference point method

3.1 The Euclidean reference point problem. The theoretical and numerical challenge is to present the decision maker with an approximation of the *Pareto set* $\mathcal{P}_{\mathsf{s}}$ and *Pareto front* $\mathcal{P}_{\mathsf{f}}$ given by

$$\mathcal{P}_{\mathsf{s}} = \big\{u \in \mathcal{U}_{\mathsf{ad}} \,|\, u \text{ is Pareto optimal}\big\} \subset \mathcal{U} \quad \text{and} \quad \mathcal{P}_{\mathsf{f}} = \hat{J}(\mathcal{P}_{\mathsf{s}}) \subset \mathbb{R}^2,$$

respectively. In order to do so, we follow the ideas laid out in [13, 14] and make use of the (Euclidean) *reference point method*: Given a reference point $z = (z_1, z_2) \in \mathcal{P}_{\mathsf{f}} + \mathbb{R}^2_{\leqq}$ we introduce the *distance function* $F_z : \mathcal{U} \to \mathbb{R}$ by

$$F_z(u) = \tfrac{1}{2}\,\|\hat{J}(u) - z\|_2^2 = \tfrac{1}{2}\big(\hat{J}_1(u) - z_1\big)^2 + \tfrac{1}{2}\big(\hat{J}_2(u) - z_2\big)^2.$$

The mapping F_z measures the Euclidean distance between $\hat{J}(u)$ and z. It follows from [2, Lemma 2] that the mapping F_z is strictly convex, if $z \leq y^{\mathsf{id}}$ holds. The goal is that – by finding a point that approximates z as best as possible – we get a Pareto optimal point for $(\hat{\mathbf{P}})$. Therefore, we have to solve the (Euclidean) *reference point problem*

$$\min F_z(u) \quad \text{s.t.} \quad u \in \mathcal{U}_{\mathsf{ad}} \tag{$\hat{\mathbf{P}}_z$}$$

which is a scalar-valued minimization problem. The following result, which extends Theorem 4 in [2], follows from Lemma 5 and [1, Theorem 3.35].

Theorem 8. *Let $z \in \mathcal{P}_{\mathsf{f}} + \mathbb{R}^2_{\leqq}$ and $\mathcal{S}_{\mathcal{H}}$ be injective. Then, $(\hat{\mathbf{P}}_z)$ has a unique solution $\bar{u}_z = (\bar{u}_{z,1}, \ldots, \bar{u}_{z,m}) \in \mathcal{U}_{\mathsf{ad}}$, which is Pareto-optimal for $(\hat{\mathbf{P}})$.*

By solving $(\hat{\mathbf{P}}_z)$ consecutively with an adaptive variation of z, we are able to move along the Pareto front in a uniform manner. This way, we get a sequence $\{z^k\}_{k\in\mathbb{N}} \subset \mathbb{R}^2$ of reference points along with optimal controls $\{u^k\}_{k\in\mathbb{N}} \subset \mathcal{U}_{\mathsf{ad}}$ that solve $(\hat{\mathbf{P}}_z)$ with $z = z^k$ as well as $\{\hat{J}^k\}_{k\in\mathbb{N}} \subset \mathbb{R}^2$ with $\hat{J}^k = \hat{J}(u^k)$.

3.2 Optimality conditions. To compute a solution to $(\hat{\mathbf{P}}_z)$ we make use of optimality conditions. Applying the chain rule and (5), we get for any $u \in \mathcal{U}$

$$F_z'(u) = \Big\langle \big(\hat{J}_1(u) - z_1\big)\mathcal{S}^{\star}_{\mathcal{H}}(\mathcal{S}_{\mathcal{H}}u - y_{\mathcal{H}}) + \big(\hat{J}_2(u) - z_2\big)u, \cdot \Big\rangle_{\mathcal{U}}.$$

The gradient $\nabla F_z(u) \in \mathcal{U}$ of F_z at $u \in \mathcal{U}_{\mathsf{ad}}$ is given by

$$\nabla F_z(u) = \big(\hat{J}_1(u) - z_1\big)\mathcal{S}^{\star}_{\mathcal{H}}(\mathcal{S}_{\mathcal{H}}u - y_{\mathcal{H}}) + \big(\hat{J}_2(u) - z_2\big)u \tag{7}$$

so that we have $F_z'(u) = \langle \nabla F_z(u), \cdot \rangle_{\mathcal{U}}$. The next results follows from (7) and [1, Corollary 3.37].

Theorem 9. *Let $\mathcal{S}_{\mathcal{H}}$ be injective and $z = (z_1, z_2) \in \mathcal{P}_{\mathsf{f}} + \mathbb{R}^2_{\leq}$. If $\bar{u}_z \in \mathcal{U}_{\mathsf{ad}}$ is an optimal solution to $(\hat{\mathbf{P}}_z)$, then a first-order necessary optimality condition reads*

$$\langle \nabla F_z(\bar{u}_z), u - \bar{u}_z \rangle_{\mathcal{U}} \geq 0 \quad \text{for all } u \in \mathcal{U}_{\mathsf{ad}}. \tag{8}$$

Example 10. We continue Examples 1, 4 and 6. Then, (8) has the form

$$\int_0^T \sum_{i=1}^m \Big(\big(\hat{J}_1(\bar{u}_z) - z_1\big) \int_{\Omega_i} \bar{p}_z(t, \boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} + \big(\hat{J}_2(\bar{u}_z) - z_2\big) \bar{u}_{z,i}(t) \Big) \big(u_i(t) - \bar{u}_{z,i}(t)\big) \, \mathrm{d}t \geq 0$$

for all $u \in \mathcal{U}_{\mathsf{ad}}$, where $\bar{p}_z \in \mathcal{Y}$ solves (6) for $u = \bar{u}_z$. ◊

Utilizing (5) we investigate the second derivative of F_z. We are interested in whether F_z'' is coercive. Let $z \in \mathcal{P}_{\mathsf{f}} + \mathbf{R}^2_{\leq}$ and $u \in \mathcal{U}_{\mathsf{ad}}$. Then, we have for every $v \in \mathcal{U}$:

$$\begin{aligned} F_z''(u)(v,v) &= \sum_{i=1}^2 \big(\hat{J}_i(u) - z_i\big) \hat{J}_i''(u)(v,v) + \|\hat{J}_i'(u)v\|^2_{\mathcal{U}'} \\ &\geq \sum_{i=1}^2 \big(\hat{J}_i(u) - z_i\big) \hat{J}_i''(u)(v,v) \\ &= \Big\langle \big((\hat{J}_1(u) - z_1)\mathcal{S}^\star_{\mathcal{H}}\mathcal{S}_{\mathcal{H}} + (\hat{J}_2(u) - z_2)\mathcal{I}_{\mathcal{U}}\big)v, v \Big\rangle_{\mathcal{U}}. \end{aligned} \tag{9}$$

Theorem 11. *Let $z \in \mathcal{P}_{\mathsf{f}} + \mathbb{R}^2_{\leq}$ and $\bar{u}_z \in \mathcal{U}_{\mathsf{ad}}$ the associated unique solution to $(\hat{\mathbf{P}}_z)$. If $z_1 \leq \hat{J}_1(\bar{u}_z)$ and $z_2 < \hat{J}_2(\bar{u}_z)$ hold, then $F_z''(\bar{u}_z)$ is coercive, i.e., for $\kappa_z = \hat{J}_2(\bar{u}_z) - z_2 > 0$ we have*

$$F_z''(\bar{u}_z)(u,u) \geq \kappa_z \|u\|^2_U \quad \text{for all } u \in \mathcal{U}.$$

In particular, (8) is also a first-order sufficient optimality condition for a strict local minimum. If additionally $z \leq y^{\mathsf{id}}$ holds, (8) is even a first-order sufficient optimality condition for a strict global minimum.

Proof. From (9), $\hat{J}_1(\bar{u}_z) - z_1 \geq 0$, $\kappa_z > 0$ and

$$F_z''(\bar{u}_z)(u,u) \geq (\hat{J}_1(\bar{u}_z) - z_1) \|\mathcal{S}_{\mathcal{H}}u\|^2_{\mathcal{H}} + (\hat{J}_2(\bar{u}_z) - z_2) \|u\|^2_{\mathcal{U}} \geq \kappa_z \|u\|^2_{\mathcal{U}}$$

for all $u \in \mathcal{U}$ we derive that $F_z''(\bar{u}_z)$ is coercive. This implies that $u \mapsto F_z(u)$ is strictly convex in a neighborhood of $\bar{u}_z$. Therefore, (8) is also sufficient for a strict local minimum.

If $z \leq y^{\mathsf{id}}$ holds, we know that F_z is strictly convex. Hence, (8) is sufficient for a strict global minimum in this case. □

Remark 12. It is possible to show that (8) is sufficient for a strict global minimum, if $z \in \mathcal{P}_{\mathsf{f}} + \mathbb{R}^2_{\leq}$ with $z \leq \hat{J}(\bar{u}_z)$ and $z \neq \hat{J}(\bar{u}_z)$ holds. A proof can be found in [1, Theorem 3.45]. ◊

3.3 A-posteriori error estimation. We want to estimate the error $\|\bar{u}_z - u^p\|_{\mathcal{U}}$, where $\bar{u}_z \in \mathcal{U}_{\mathsf{ad}}$ is the (unknown) optimal control of $(\hat{\mathbf{P}}_z)$ and $u^p \in \mathcal{U}_{\mathsf{ad}} \setminus \{\bar{u}_z\}$ is a given suboptimal control. Let $z \in \mathcal{P}_{\mathsf{f}} + \mathbb{R}^2_{\leq}$ and $\mathcal{S}_{\mathcal{H}}$ be injective. We follow along the lines of [10] and define the perturbation $\zeta \in \mathcal{U}$ by

$$\zeta_i(s) = \begin{cases} [(\nabla F_z(u^p))_i(s)]_- & \text{for } s \in \mathcal{D} \text{ with } u_i^p(s) = u_{a,i}(s), \\ -[(\nabla F_z(u^p))_i(s)]_+ & \text{for } s \in \mathcal{D} \text{ with } u_i^p(s) = u_{b,i}(s), \\ -(\nabla F_z(u^p))_i(s) & \text{otherwise} \end{cases} \tag{10}$$

for $i = 1, \dots, m$, where we have used the decomposition for a real number $\xi \in \mathbb{R}$ as $\xi = [\xi]_+ - [\xi]_-$ with $[\xi]_+ = \max(0, \xi)$ and $[\xi]_- = -\min(0, \xi)$. The following theorem is proved in [1, Theorem 3.51].

Theorem 13. *Let $z = (z_1, z_2) \in \mathcal{P}_{\mathsf{f}} + \mathbb{R}^2_{\leq}$, $\bar{u}_z \in \mathcal{U}_{\mathsf{ad}}$ the associated solution to $(\hat{\mathbf{P}}_z)$ and $u^p \in \mathcal{U}_{\mathsf{ad}}$. If $\hat{J}_2(\bar{u}_z) + \hat{J}(u^p) > 2z_2$ is satisfied, we have*

$$\|\bar{u}_z - u^p\|_{\mathcal{U}} \leq \tfrac{1}{C(\bar{u}_z, u^p, z_2)} \|\zeta\|_{\mathcal{U}}, \quad \|\hat{J}(\bar{u}_z) - \hat{J}(u^p)\|_2 \leq \tfrac{1}{2\sqrt{C(\bar{u}_z, u^p, z_2)}} \|\zeta\|_{\mathcal{U}}, \tag{11}$$

where ζ is given by (10) *and $C(\bar{u}_z, u^p, z_2) = (\hat{J}_2(\bar{u}_z) + \hat{J}_2(u^p))/2 - z_2 > 0$.*

4. Reduced-order modelling (ROM) by POD

The previously described reference point method requires for $(\hat{\mathbf{P}}_z)$ to be solved repeatedly. In this multi-query context, it is reasonable to apply model-order reduction techniques in order to reduce the computational effort for the optimization. In this work we focus in particular on POD; see, e.g., [7]. Suppose that we have chosen an admissible control $u \in \mathcal{U}_{\mathsf{ad}}$. Let $y = \mathcal{S}u$ denote the associated state variable and p be the solution to (6). Then we consider the linear space of snapshots $\mathcal{V} = \operatorname{span}\{y(t), p(t) \,|\, t \in [0,T]\} \subset V$ with $\mathfrak{d} = \dim \mathcal{V} \leq \infty$. For any finite $\ell \leq \mathfrak{d}$ we are interested in determining a POD basis of rank ℓ which minimizes the mean square error between $y(t)$, $p(t)$ and their corresponding ℓ-th partial Fourier sums on average in $[0,T]$:

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

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

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

Then, it is well-known [6, Theorem 1.15] that a solution $\{\psi_i\}_{i=1}^{\ell}$ to $(\mathbf{P}^\ell)$ is given by the eigenvectors associated with the ℓ largest eigenvalues of $\mathcal{R}$, i.e.,

$$\mathcal{R}\psi_i = \lambda_i \psi_i \text{ for } 1 \leq i \leq \ell, \quad \lambda_1 \geq \ldots \geq \lambda_\ell \geq \lambda_{\ell+1} \geq \ldots \geq 0.$$

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

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

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

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} \langle y^\ell(t), \psi_j \rangle_H + a(y^\ell(t), \psi) &= \langle \mathcal{B}u(t), \psi \rangle_{V',V} \ \forall \psi \in V^\ell, \text{ f.a.a. } t \in (0,T], \\ y^\ell(0) &= 0. \end{aligned} \tag{12}$$

The adjoint equation (6) is also reduced in a similar way; cf. [6, Section 1.4.4]. Let us introduce the following linear, H-orthogonal projection:

$$\mathcal{P}^\ell_{H,V^\ell} : H \to V^\ell, \text{ and for } \varphi \in H, \ \mathcal{P}^\ell \varphi \text{ minimizes } \inf_{\varphi^\ell \in V^\ell} \|\varphi - \varphi^\ell\|_H.$$

We suppose that $\mathcal{P}^\ell_{H,V^\ell}$ is a bounded operator from V to V. Then, we can apply [15, Theorem 5.3]. It follows from [6, Propositions 1.27 and 1.32] that the operator $\mathcal{S}^\ell$ is well-defined and

$$\|(\mathcal{S} - \mathcal{S}^\ell)u\|^2_{L^2(0,T;V)} \leq C \sum_{i=\ell+1}^{\mathfrak{d}} \lambda_i \, \|\psi_i - \mathcal{P}^\ell_{H,V^\ell}\psi_i\|_V^2 < \infty$$

for a $u \in \mathcal{U}_{\mathsf{ad}}$. If $y \neq \mathcal{S}u$ and $y \in H^1(0,T;V)$ we still have $\lim_{\ell\to\infty} \|(\mathcal{S} - \mathcal{S}^\ell)u\|_{L^2(0,T;V)} = 0$.

The POD approximation to (4) is given by the minimization problem

$$\min \hat{J}^\ell(u) \quad \text{s.t.} \quad u \in \mathcal{U}_{\mathsf{ad}}, \tag{$\hat{\mathbf{P}}^\ell$}$$

where we set $\hat{J}^\ell(u) = J(\hat{y} + \mathcal{S}^\ell u, u)$. In particular, $\hat{J}_2^\ell = \hat{J}_2$ holds true. To solve ($\hat{\mathbf{P}}^\ell$) we apply the reference point method utilizing the corresponding distance function

$$F_z^\ell(u) = \tfrac{1}{2} \|\hat{J}^\ell(u) - z\|_2^2 = \tfrac{1}{2}\big(\hat{J}_1^\ell(u) - z_1\big)^2 + \tfrac{1}{2}\big(\hat{J}_2(u) - z_2\big)^2$$

with a reference point $z \in \mathcal{P}_{\mathsf{f}} + \mathbb{R}^2_{\leq}$. The low-order correspondant of ($\hat{\mathbf{P}}_z$) reads

$$\min F_z^\ell(u) \quad \text{s.t.} \quad u \in \mathcal{U}_{\mathsf{ad}} \tag{$\hat{\mathbf{P}}_z^\ell$}$$

Thereby, we obtain a POD suboptimal control $\bar{u}_z^\ell \in \mathcal{U}_{\mathsf{ad}}$ for any z. The resulting error is then estimated using Theorem 13 with $u^p = \bar{u}_z^\ell$. This indicates the quality of the current POD basis, which can then be recomputed if necessary. Moreover, the next estimate follows from [6, Propositions 1.27 & 1.32] and [1, Theorem 5.31]

Theorem 14. *There is a constant $C > 0$ which does not depend on ℓ such that*

$$\left|\hat{J}_1(u) - \hat{J}_1^\ell(u)\right| \leq C \sum_{i=\ell+1}^{\mathfrak{d}} \lambda_i \left\|\psi_i\right\|_V^2 < \infty \quad \textit{for any } \ell \leq \mathfrak{d}.$$

If $\tilde{u} \in \mathcal{U}_{\mathsf{ad}} \setminus \{u\}$ holds and $y = \hat{y} + \mathcal{S}\tilde{u}$ even belongs to $H^1(0,T;V)$, we have $\lim_{\ell\to\infty} |\hat{J}_1(\tilde{u}) - \hat{J}_1^\ell(\tilde{u})| = 0$.

Remark 15. We refer the reader to the Theorems 5.39, 5.41 and 5.45 in [1] where also POD convergence results for $\|\nabla \hat{J}_1(u) - \nabla \hat{J}_1^\ell(u)\|_{\mathcal{U}}$, $\|\bar{u}_z - \bar{u}_z^\ell\|_{\mathcal{U}}$, $\|\nabla F_z(u) - \nabla F_z^\ell(u)\|_{\mathcal{U}}$ and $\|\zeta\|_{\mathcal{U}}$ are given. ◊

5. The optimization algorithm

5.1 The reference point algorithm. We understand the task of numerically solving the multiobjective optimization problem $(\hat{\mathbf{P}})$ (respectively $(\hat{\mathbf{P}}^\ell)$) as computing an approximation of the Pareto front $\mathcal{P}_{\mathsf{f}}$ and the Pareto set $\mathcal{P}_{\mathsf{s}}$ by discrete, finite substitutes

$$\widetilde{\mathcal{P}}_{\mathsf{s}} = \{u^{(1)}, ..., u^{(N)}\} \subset \mathcal{P}_{\mathsf{s}} \quad \text{and} \quad \widetilde{\mathcal{P}}_{\mathsf{f}} = \hat{J}(\widetilde{\mathcal{P}}_{\mathsf{s}}) \subset \mathcal{P}_{\mathsf{f}}$$

The process of computing these N numerical solutions is of recursive nature. The first optimal control $u^{(1)} \in \mathcal{U}_{\mathsf{ad}}$ is obtained by applying a *weighted-sum method* to $(\hat{\mathbf{P}})$. For a weight $\alpha > 0$, this problem takes the form

$$\min \hat{J}_1(u) + \alpha \hat{J}_2(u) \quad \text{subject to (s.t.)} \quad u \in \mathcal{U}_{\mathsf{ad}} \qquad (\hat{\mathbf{P}}_\alpha)$$

It follows from [1, Lemma 3.14] that $(\hat{\mathbf{P}}_\alpha)$ admits a unique solution $\bar{u} \in \mathcal{U}_{\mathsf{ad}}$ which is Pareto optimal. We can therefore choose the associated optimal control as a first point $u^{(1)} = \bar{u} \in \mathcal{P}_{\mathsf{s}}$ of the approximate Pareto set. The parameter α influences how dominant the individual components $\hat{J}_1$ and $\hat{J}_2$ are for $(\hat{\mathbf{P}}_\alpha)$. The choice of $\alpha \gg 1$ will result in a large impact of $\hat{J}_2$, thereby inducing a solution $\bar{u}$ with low control effort $\hat{J}_2(\bar{u})$ and high state penalization $\hat{J}_1(\bar{u})$. Conversely, a parameter $\alpha \ll 1$ will enforce a solution with low state penalization and a high control effort. In our simulations, we choose the latter variant and thus accept $u^{(1)}$ as the Pareto point with the lowest value for $\hat{J}_1$ that we will achieve.
Since $u^{(1)}$ is Pareto optimal, Definition 7 now implies that decreasing the component $\hat{J}_2$ further than $\hat{J}_2(u^{(1)})$ by varying the admissible control can only be achieved by increasing the first component $\hat{J}_1$ from $\hat{J}_1(u^{(1)})$. If we view the objective space as a two-dimensional plane, this means that $\mathcal{P}_{\mathsf{f}}$ continues from $\hat{J}(u^{(1)})$ to the lower right. The idea behind the recursive algorithm is to iteratively generate a series of reference points $z^{(2)}, ..., z^{(N)}$ and choose $u^{(n)}$ as the solution to $(\hat{\mathbf{P}}_z)$ with $z = z^{(n)}$ $(i = 2, ..., N)$. In this way, we would like to move along $\mathcal{P}_{\mathsf{f}}$ in an equidistant manner. Since we know that $\mathcal{P}_{\mathsf{f}}$ continues to the lower right, the reference points should be chosen similarily. This is realized by iteratively defining a reference point $z^{(n+1)} \in \mathbb{R}^2$ which is located at the lower

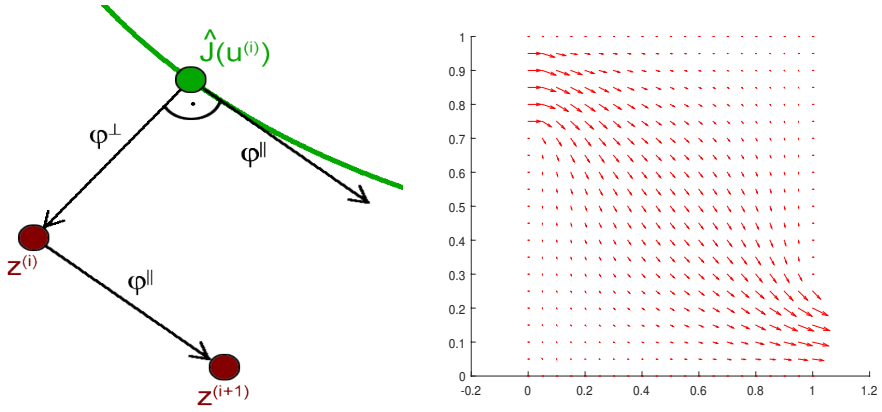


Figure 1: Choice of $z^{(n+1)}$ using $h^{\perp} = h^{||} = 1$ (left); and flow field $\tilde{\beta}$ (right).

right of the current discrete Pareto point $\hat{J}(u^{(n)})$ and captures the local geometry of $\mathcal{P}_{\mathsf{f}}$. For this, let us observe the following properties of Euclidian reference point problems: From [1, Lemma 3.42] it follows that $u^{(n)}$ is also the unique solution to each reference point $z = \hat{J}(u^{(n)}) + \lambda(z^{(n)} - \hat{J}(u^{(n)}))$ for every $\lambda \geq 0$. This implies that $\varphi^{\perp} := z^{(n)} - \hat{J}(u^{(n)}) \in \mathbb{R}^2$ lies perpendicular to $\mathcal{P}_{\mathsf{f}}$ at $\hat{J}(u^{(n)})$; cf. left plot of Figure 1. Consequently, the vector $\varphi^{||} := (-\varphi_2^{\perp}, \varphi_1^{\perp})^t \in \mathbb{R}^2$ lies tangential to $\mathcal{P}_{\mathsf{f}}$ at $\hat{J}(u^{(n)})$ and points to the lower right. This means that by scaling $\varphi^{||}$ accordingly, we may determine how far along $\mathcal{P}_{\mathsf{f}}$ the next reference point will be chosen. Likewise, scaling $\varphi^{\perp}$ allows to determine how close to $\mathcal{P}_{\mathsf{f}}$ the reference points are located. This translates to choosing scalars $h^{||}$, $h^{\perp} > 0$ and recursively defining for $n = 2, ..., N-1$:

$$z^{(n+1)} := \hat{J}(u^{(n)}) + h^{||} \cdot \frac{\varphi^{||}}{\|\varphi^{||}\|} + h^{\perp} \cdot \frac{\varphi^{\perp}}{\|\varphi^{\perp}\|}. \tag{13}$$

Note that we can not define the reference point $z^{(2)}$ in this manner since $u^{(1)}$ was computed by the weighted sum method rather than by the reference method. However, due to the high dominance of $\hat{J}_1$ at $u^{(1)}$, we may assume that $\mathcal{P}_{\mathsf{f}}$ is approximately vertical in this area. In (13), we can therefore set $\varphi^{||} := (0, -1)^t$ and $\varphi^{||} := (-1, 0)^t$. In other words,

$$z^{(2)} := \hat{J}(u^{(1)}) - \begin{pmatrix} h^{\perp} \\ h^{||} \end{pmatrix}. \tag{14}$$

The following lemma assures when the above strategy will actually generate a series of Pareto optimal points.

120 **Lemma 16.** *Let $\widetilde{\mathcal{P}}_{\mathsf{s}} = \{u^{(1)}, ..., u^{(N)}\} \subset \mathcal{U}_{\mathsf{ad}}$ be generated as described above. Further, assume that $\mathcal{S}_{\mathcal{H}}$ is injective and $z^{(n)} \in \mathcal{P}_{\mathsf{f}} + \mathbb{R}^2_{\leq}$ holds for $n = 2, ..., N$. Then, $\widetilde{\mathcal{P}}_{\mathsf{s}} \subset \mathcal{P}_{\mathsf{s}}$ and $\widetilde{\mathcal{P}}_{\mathsf{f}} \subset \mathcal{P}_{\mathsf{f}}$.*

Proof. First, the Pareto optimality of $u^{(1)}$ follows from [1, Lemma 3.14]. Each subsequent control $u^{(n)}$ $(n = 2, ..., N)$ is the solution to a reference point problem $(\hat{\mathbf{P}}_z)$. It then follows from Theorem 8 that these controls are Pareto optimal. □

For the numerical implementation, the condition $z^{(n)} \in \mathcal{P}_{\mathsf{f}} + \mathbb{R}^2_{\leq}$ is not easy to verify since $\mathcal{P}_{\mathsf{f}}$ is not known in advance. Instead, we employ a heuristic termination condition. As it was already explained, the algorithm tracks the Pareto front along decreasing $\hat{J}_2$ and increasing $\hat{J}_1$. For two subsequent solutions, we therefore expect the difference $\hat{J}_2(u^{(n+1)}) - \hat{J}_2(u^{(n)})$ to be negative. Once the algorithm has reached the approximate end of the Pareto front, this difference will become very close to zero or even positive. Once this is the case, we abort the algorithm. The procedure is summarized in Algorithm 1.

Algorithm 1 (Reference point method for the full-order model)

Require: Maximal number $N \in \mathbb{N}$ of Pareto points, recursive parameters $h^{||}, h^{\perp} > 0$, weighted-sum parameter $\alpha > 0$, termination parameter $\varepsilon \geq 0$.

Solve $(\hat{\mathbf{P}}_\alpha)$ and the solution as $u^{(1)}$. Set $\widetilde{\mathcal{P}}_{\mathsf{s}} \leftarrow \{u^{(1)}\}$ and $\widetilde{\mathcal{P}}_{\mathsf{f}} \leftarrow \{\hat{J}(u^{(1)})\}$.
Compute $z^{(2)}$ by (14).
for $n = 2, ..., N$ **do**
    Solve $(\hat{\mathbf{P}}_z)$ with reference point $z^{(n)}$ and save the solution as $u^{(n+1)}$.
    if $\hat{J}_2(u^{(n+1)}) - \hat{J}_2(u^{(n)}) > -\varepsilon$ **then**
        return $\widetilde{\mathcal{P}}_{\mathsf{s}}$, $\widetilde{\mathcal{P}}_{\mathsf{f}}$
    else
        Add $\widetilde{\mathcal{P}}_{\mathsf{s}} \leftarrow \widetilde{\mathcal{P}}_{\mathsf{s}} \cup \{u^{(n+1)}\}$, $\widetilde{\mathcal{P}}_{\mathsf{f}} \leftarrow \widetilde{\mathcal{P}}_{\mathsf{f}} \cup \{\hat{J}(u^{(n+1)})\}$.
        if $i < n$ **then**
            Compute $z^{(n+1)}$ by (13).
return $\widetilde{\mathcal{P}}_{\mathsf{s}}$, $\widetilde{\mathcal{P}}_{\mathsf{f}}$.

5.2 Solution of the reference point problem. The minimization problems $(\hat{\mathbf{P}}_z)$ and $(\hat{\mathbf{P}}_\alpha)$ are both of the type

$$\min\ F(u) \quad \text{s.t. } u \in U_{ad}, \tag{$\hat{\mathbf{P}}$}$$

where $F : \mathcal{U} \to \mathbb{R}$ is twice Fréchet differentiable and strictly convex. They are numerically solved by an *iterative descent algorithm*: A sequence $\{u^{(k)}\}_{k=0}^{\infty} \subset \mathcal{U}_{\mathsf{ad}}$ of controls is generated by the recursion

$$u^{(k+1)} \ = \ \mathcal{P}_{\mathsf{ad}}\big(u^{(k)} + \theta_k \delta^{(k)}\big) \quad \text{for } k = 0, 1, \ldots,$$

where $\mathcal{P}_{\mathsf{ad}} : \mathcal{U} \to \mathcal{U}$ denotes the projection onto the admissible set, meaning that

$$[\mathcal{P}_{\mathsf{ad}} u]_i(s) = \begin{cases} u_{a,i}(s), & \text{if } u_i(s) < u_{a,i}(s), \\ u_i(s), & \text{if } u_i(s) \in [u_{a,i}(s), u_{b,i}(s)], \\ u_{b,i}(s), & \text{if } u_i(s) > u_{b,i}(s). \end{cases}$$

Further, the direction $\delta^{(k)} \in U$ is computed by a *Projected Newton-CG method*: At $u^{(k)}$, the set of active indices is defined by

$$\mathcal{A}(u^{(k)}) := \big\{(s,i) \in \mathcal{D} \times \{1,...,m\} \,\big|\, u_i^{(k)}(s) \in \{u_{a,i}(s), u_{b,i}(s)\}\big\}.$$

The control space is now split into two subspaces: $\mathcal{U} = \mathcal{U}_{\mathcal{A}(u^{(k)})} \oplus \mathcal{U}_{\mathcal{I}(u^{(k)})}$ with

$$\mathcal{U}_{\mathcal{A}(u^{(k)})} = \big\{u \in \mathcal{U} \,\big|\, u_i(s) = 0 \text{ if } (s,i) \in \mathcal{A}(u^{(k)})\big\},$$
$$\mathcal{U}_{\mathcal{I}(u^{(k)})} = \big\{u \in \mathcal{U} \,\big|\, u_i(s) = 0 \text{ if } (s,i) \notin \mathcal{A}(u^{(k)})\big\}.$$

Let $\mathcal{P}_\mathcal{A}, \mathcal{P}_\mathcal{I} : \mathcal{U} \to \mathcal{U}$ be the projections onto the respective subspaces. Naturally, for every $\delta \in \mathcal{U}$, it is then $\delta = \mathcal{P}_\mathcal{A}\delta + \mathcal{P}_\mathcal{I}\delta$. The idea behind the Projected Newton's method is now to include second-order information of F only for the active indices: The *Projected Hessian* is therefore defined as

$$\nabla^2_\mathcal{P} F(u^{(k)}) : \mathcal{U} \to \mathcal{U}, \quad [\nabla^2_\mathcal{P} F(u^{(k)})]\delta = \mathcal{P}_\mathcal{A} \nabla^2 F(u) \mathcal{P}_\mathcal{A}\delta + \mathcal{P}_\mathcal{I}\delta \text{ for } \delta \in \mathcal{U}.$$

The search direction $\delta^{(k)} \in \mathcal{U}$ is now determined in the very same way as the canonical Newton's method:

$$[\nabla^2_\mathcal{P} F(u^{(k)})]\delta^{(k)} = -\nabla F(u^{(k)}) \quad \text{for } k = 0, 1, ... \tag{15}$$

For the solution of the linear system (15), a CG-algorithm is employed. It is easy to verify that since $\nabla^2 F(u^{(k)})$ is positive definite, $\nabla^2_\mathcal{P} F(u^{(k)})$ is as well. However, it has to be noted that the definiteness may sometimes be lost in the numerical experiments due to discretization inaccuracies. If this is the case, the Newton scheme (15) is temporarily abandoned in favor of a gradient step, i.e. we set $\delta^{(k)} = -\nabla F(u^{(k)})$. After the direction $\delta^{(k)} \in \mathcal{U}$ has been determined, a backtracking line search is employed to find a step size $\theta_k \in \mathbb{R}_>$ which satisfies the *Armijo rule*:

$$F(u^{(k)} + \theta_k \delta^{(k)}) \;\leq\; F(u^{(k)}) + c_A \theta_k \langle \nabla F(u^{(k)}), \delta^{(k)} \rangle_\mathcal{U} \tag{16}$$

with a constant $c_A \in (0,1)$. Starting with a full Newton step at $\theta_k = 1$, the step size is recursively decremented by setting $\theta_k \leftarrow \beta\theta_k$ with a constant $\beta \in (0,1)$ until (16) is satisfied. This procedure helps to ensure a satisfactory descent of the objective function.

5.3 Model-order reduction. Section 4 has introduced the concept of model-order reduction. This leads to a *reduced gradient* ∇F_z^ℓ which is needed in the optimization. The general strategy for the application of model-order reduction will always follow the same steps: In the beginning of the optimization, use a certain control $u \in \mathcal{U}_{\mathsf{ad}}$ to generate full-order solutions $y = \mathcal{S}u \in W(0,T)$ of (2) and $p \in W(0,T)$ of (6). From this point on, replace (2) and (6) by their low-order POD-Galerkin schemes. Then, use e.g. the a-posteriori error estimator (11) in order to assess the quality of solution to the arising low-order optimization problems. If the error estimator deteriorates too drastically, repeat 1. for a current control, i.e. recompute the reduced-order model. There are many different strategies for a good choice of the full-order data whereby the reduced model is generated. In Algorithm 2, we present a reduced-order variation of Algorithm 1 which contains one possible strategy for the choice of data.

Algorithm 2 (Reference point method for the reduced-order model)

Require: Maximal number $N \in \mathbb{N}$ of Pareto points, recursive parameters $h^{||}, h^{\perp} > 0$, weighted-sum parameter $\alpha \geq 0$, termination parameter $\varepsilon \geq 0$.

Solve $(\hat{\mathbf{P}}_\alpha)$ to get the control $u^{(1)}$. Set $\widetilde{\mathcal{P}}_{\mathsf{s}} \leftarrow \{u^{(1)}\}$ and $\widetilde{\mathcal{P}}_{\mathsf{f}} \leftarrow \{\hat{J}(u^{(1)})\}$.
Get $\{\psi_i\}_{i=1}^{\ell}$ from $(\mathbf{P}^\ell)$ using the state y and the dual p associated with $u^{(1)}$.
Compute $z^{(2)}$ by (14).
for $n = 2, ..., N$ **do**
    Solve $(\hat{\mathbf{P}}_z^\ell)$ with reference point $z^{(n)}$ and save the solution as $u^{(n+1)}$.
    if $\hat{J}_2(u^{(n+1)}) - \hat{J}_2(u^{(n)}) > -\varepsilon$ **then**
        return $\widetilde{\mathcal{P}}_{\mathsf{s}}$, $\widetilde{\mathcal{P}}_{\mathsf{f}}$
    else
        Add $\widetilde{\mathcal{P}}_{\mathsf{s}} \leftarrow \widetilde{\mathcal{P}}_{\mathsf{s}} \cup \{u^{(n+1)}\}$, $\widetilde{\mathcal{P}}_{\mathsf{f}} \leftarrow \widetilde{\mathcal{P}}_{\mathsf{f}} \cup \{\hat{J}(u^{(n+1)})\}$.
        if $i < n$ **then**
            Compute the a-posteriori error estimator $\Delta(u^{(n)})$ for $u^{(n)}$ by (11).
            if $\Delta(u^{(n)}) > \varepsilon_\Delta$ **then**
                Compute the full-order state y and adjoint p to control $u^{(n)}$.
                Use y and p to recompute the POD-ROM by solving $(\mathbf{P}^\ell)$.
            Compute $z^{(n+1)}$ by (13).
return $\widetilde{\mathcal{P}}_{\mathsf{s}}$, $\widetilde{\mathcal{P}}_{\mathsf{f}}$.

6. Numerical experiments

6.1 Setting. In our numerical tests we consider the bicriterial optimal control problem presented in Example 4. The spatial domain is $\Omega := (0,1)^2 \subset \mathbb{R}^2$ and we choose $T = 1$. The diffusion parameter is given by $\kappa = 0.5$. For the convection term of the PDE (3a) we use $\beta := c_b\tilde{\beta}$, where $\tilde{\beta}$ is a stationary solution of a Navier-Stokes equation (cf. right plot of Figure 1) and $c_b \geq 0$ is a parameter to control the strength of the convection. It holds $\|\tilde{\beta}\|_{L^\infty(\Omega;\mathbb{R}^2)} \approx 6.6$. We impose a floor heating of the whole room with four uniformly distributed heaters in the domains $A_1 = (0, 0.5)^2$, $A_2 = (0, 0.5) \times (0.5, 1)$, $A_3 = (0.5, 1) \times (0, 0.5)$ and $A_4 = (0.5, 1)^2$, i.e. $m = 4$. We refer to the different regions of the heaters by calling the set A_i region i. We set bilateral constraints $u_a(t) = 0$ and $u_b(t) = 3$ on the control u. This yields $\mathcal{U}_{\mathsf{ad}} = \{u \in L^2(0,T;\mathbb{R}^4) \mid 0 \leq u(t) \leq 3 \text{ for all } t \in [0,1]\}$. The room is supposed to be perfectly isolated, i.e. it holds $\alpha_1 = 0$ on the whole boundary $\Gamma_1 = \partial\Omega$. This yields a homogeneous Neumann boundary condition. As an initial condition we suppose that there is a constant temperature of 16° in the whole room, i.e. $y_\circ(\boldsymbol{x}) = 16$ for all $\boldsymbol{x} \in \Omega$. For the desired temperature we want a uniform increase of the temperature from 16° at $t = 0$ to 18° at $t = T$: $y_Q(t, \boldsymbol{x}) = 16 + 2t$ for all $(t, \boldsymbol{x}) \in Q$. All computations were carried out on a standard PC, Intel(R) Core(TM)2 Duo CPU P8700 @ 2.53GHz, 4 GB RAM.

6.2 Results. In our first experiment we run the algorithm for $c_b = 1$. The left plot of Figure 2 shows the Pareto front $\mathcal{P}_{\mathsf{f}}$ that we obtain in this case. We

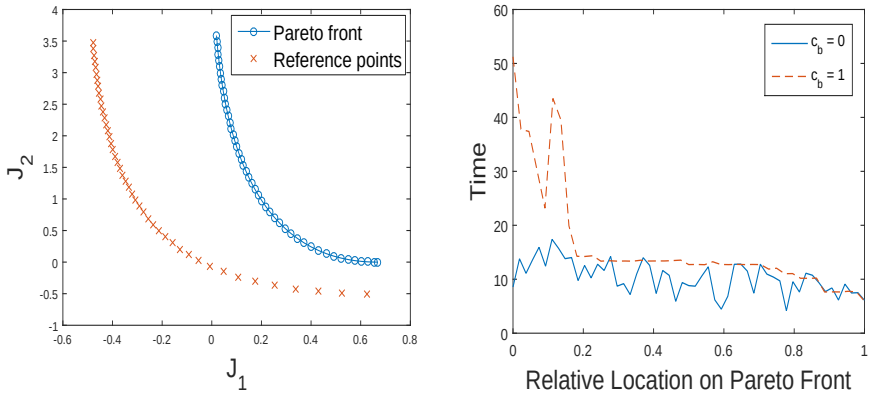
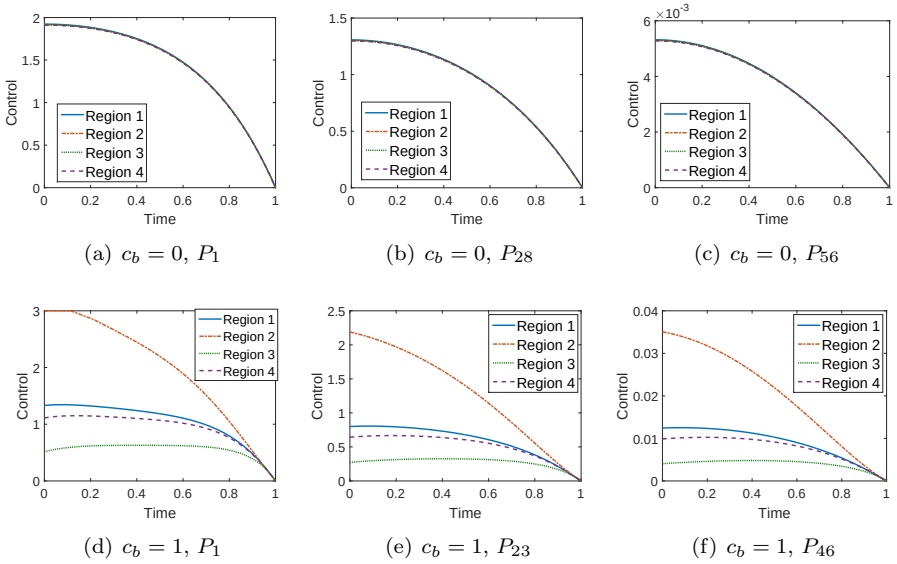


Figure 2: $\mathcal{P}_\mathsf{f}$ and reference points for $c_b = 1$ (left); Computation times for $c_b = 0$ and $c_b = 1$ (right).

observe that $\mathcal{P}_\mathsf{f}$ is smoothly approximated by 47 Pareto optimal points. Hereby, $\mathcal{P}_\mathsf{f}$ ranges from $P^1 = (0.0191, 3.5876)$ to $P^{47} = (0.6667, 0)$, i.e. the desired temperature can be achieved quite closely in the upper part of the Pareto front. In a next step we want to investigate the heating strategies for different optimal controls and compare them with the strategies of optimal controls obtained by running the algorithm for a system without convection, i.e. $c_b = 0$. Therefore, we pick three optimal controls in both cases, which are situated in the top, the middle and the bottom of the Pareto front.

The influence of the convection term on the optimal controls can be immediately seen in Figure 3. As the system without convection is totally symmetric, all four heaters heat the same up to numerical inaccuracies. In contrast to that the optimal controls of the system with convection adapt to the air flow, which goes from the top left corner of the room to the right bottom corner by using different heating strategies for all four heaters. Heater two needs to heat the most because the warm air is transported from the second region mainly into the third region. Consequently, heater three has to heat the least. By especially looking at the optimal control $\bar{u}^1$ of the system with $c_b = 1$, we see that the control of the second heater is active on the upper bound of the constraints in the beginning of the heating process. The consequence is that the temperature in this region is actually overshooting the desired temperature distribution in the beginning. In the further progress the excessive heat of this region is transported into the other regions by the air flow, so that heaters one, three and four actually have to heat way less than in the case without convection.

Another interesting aspect worth considering is the difference in the computation times for the system with and without convection. As expected one can see in the right plot of Figure 2 that the system without convection needs less computation time because on the one hand, the convection term adds dynamics to the optimal control problem, which are more difficult to handle, i.e. more



(a) $c_b = 0$, P_1 (b) $c_b = 0$, P_{28} (c) $c_b = 0$, P_{56}

(d) $c_b = 1$, P_1 (e) $c_b = 1$, P_{23} (f) $c_b = 1$, P_{46}

Figure 3: Optimal controls for different values of c_b.

Newton-CG iterations are needed for convergence than without the convection term. On the other hand, solving the state and adjoint equation gets more costly due to the fact that the arising linear equation systems become non-symmetric. In addition, the computation times of both systems are monotonically decreasing with some fluctuations when traversing the Pareto front from top to bottom. By looking at the computation time of the system with convection, several jumps can actually be seen. These jumps correspond to a decrease of needed Newton-CG iterations in the optimization routine. The reason for this decrease is that the coercivity constant of $F_z''(\bar{u})$ is given by the difference $\hat{J}_2(\bar{u}) - z_2$ (see Theorem 11), which is increasing in comparison to $\hat{J}_1(\bar{u}) - z_1$ while traversing the Pareto front from top to bottom and thus making the problem smoother.

A second experiment is conducted to investigate the influence of the parameter $h^\perp$ in the generation of the reference points (see equation (13)) on the approximation quality of the Pareto front. It was shown in [1] that this choice of reference points leads to $\|\hat{J}(\bar{u}^{n+1} - \bar{u}^n)\|_2 \leq h^\perp$. Indeed, Figure 4 shows that the distances between consecutive Pareto optimal points adapt nicely to the parameter $h^\perp$. Additionally, we observe that in regions in which $\mathcal{P}_\mathsf{f}$ has a high curvature the distance between consecutive Pareto optimal points gets automatically smaller without having to change the parameter $h^\perp$. This is an important property because regions with high curvature need a finer approximation.

Now we turn to analysing the results we get when applying the reduced-order model. First of all, we are interested in the influence of the convection term on the quality of the approximation. In [2] it was shown that already two basis functions yield good results for a system without convection. However,

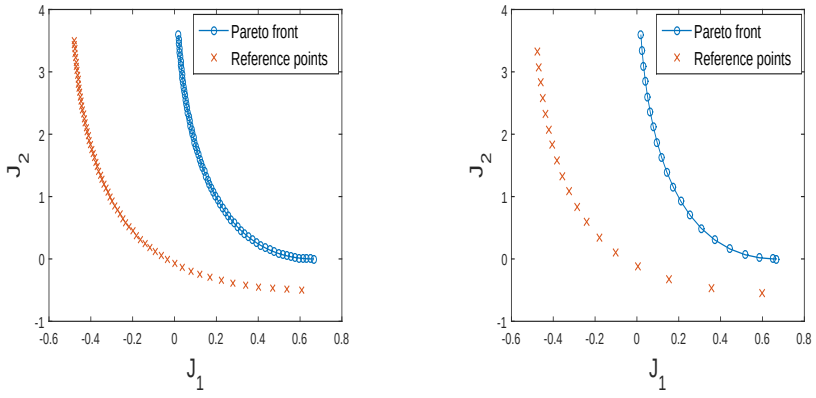
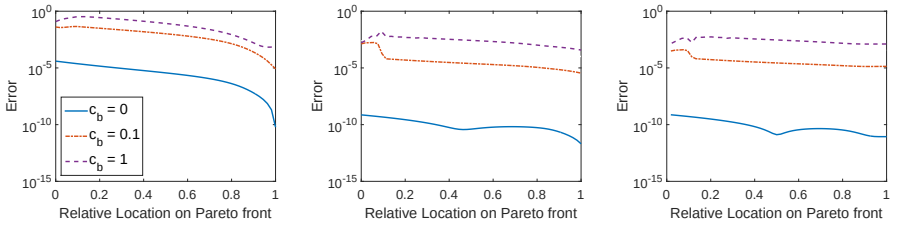


Figure 4: Pareto front for $h_x = 0.07$ (left) and $h_x = 0.25$ (right).

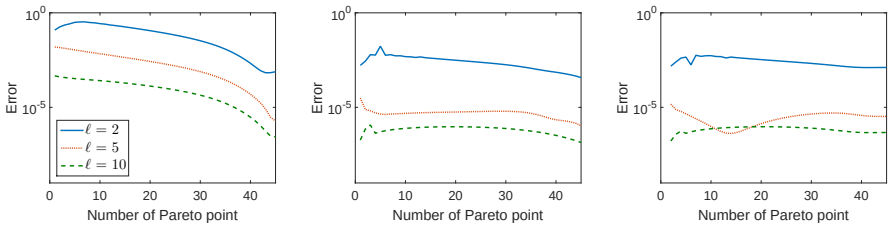
we observe in Figure 5 that switching on convection immediately increases the errors of the reduced-order model by a factor of about 10^3 in the control space and 10^6 in the objective space and for the reference points. On the other



(a) Error in control space (b) Error in objective space (c) Error of reference points

Figure 5: Errors between the full and the POD solution for $\ell = 2$ for $c_b = 0$ (blue), $c_b = 0.1$ (magenta) and $c_b = 1$ (purple).

hand, increasing the strength of the convection does not have such a strong influence on the errors. This can be explained by the fact that adding convection to the system completely changes the dynamics of the state and the adjoint equation, whereas increasing the strength of the convection only intensifies these dynamics. To get a mathematical explanation for this, we can look at the eigenvalues $\{\bar{\lambda}_i\}_{i=1}^{\mathfrak{d}}$ of the operator $\mathcal{R}$ for the different values of c_b. It was shown in Section 4 that the sum $\sum_{i=\ell+1}^{\mathfrak{d}} \bar{\lambda}_i^n$ measures in some sense the error of the POD approximation. It turns out that these eigenvalues decrease much faster for $c_b = 0$ than the ones for $c_b \in \{0.1, 1\}$. So in a next step we increase the number of POD basis functions to see how the approximation errors behave. Therefore, we take $c_b = 1$ fixed and run the algorithm with $\ell \in \{2, 5, 10\}$. Figures 6 (a), (b) and (c) clearly show that all three errors decrease for an increasing number of basis functions. The fact that the error of the reference points is lower for $\ell = 5$



(a) Error in control space (b) Error in objective space (c) Error of reference points

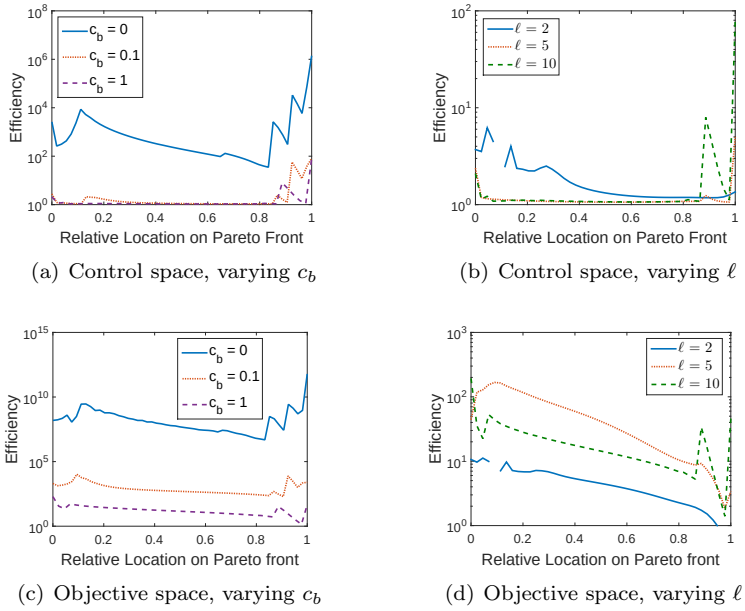
Figure 6: Errors between the full and the POD solution for $c_b = 1$ for $\ell = 2$ (blue), $\ell = 5$ (magenta) and $\ell = 10$ (green)

than for $\ell = 10$ for some reference points is not due to a better approximation, but due to coincidence in the computation of the reference points. This can be seen by looking at the errors in the objective space, which are always higher for $\ell = 5$ than for $\ell = 10$.

As there is a lack of a-priori analysis for the POD method, it is crucial for the numerical implementation to have good a-posteriori estimates. One such estimate was introduced in Theorem 13. Here, we want to test its efficiency. Therefore, we consider the quotient of the a-posteriori estimates (11) and the real approximation error in the control and the objective space, respectively. The closer this quotient is to 1, the more efficient the estimate is. The results for different parameter settings can be seen in Figure 7. We observe that the efficiency of the a-posteriori estimate in the control space is very good except for the cases $c_b = 0$ and $\ell = 2$. For all of the other parameter settings the efficiency is close to 1 in large parts of the Pareto front. On the other hand, the efficiencies of the a-posteriori estimates in the objective space are worse by a factor 10-100 in most cases. Note that the gaps in the plots in Figures 7 (b) and (d) for $\ell = 2$ are due to the fact that the optimization routine does not converge properly in this case, so that $\hat{J}_2(\bar{u}^5) < z_2^5$ holds and the a-posteriori estimates cannot be computed.

Using the good efficiency of the a-posteriori estimate in the control space, we propose a straightforward strategy to adaptively increase the number of utilised POD basis functions. Algorithm 6 shows the routine for computing the n-th Pareto optimal point. In a nutshell the number of POD basis functions is increased, if the a-posteriori estimate for the controls exceeds a predefined threshold μ.

We test this algorithm in two different versions: First, the Pareto front is as usual computed from top to bottom, and secondly from bottom to top. The later approach appears more natural in the sense that we start with the part of the Pareto front, in which the scalar optimization problem is well conditioned and only small dynamics are expected due to little control input. In the process the POD basis can then be extended with the dynamics getting stronger and the problem getting worse conditioned. The first approach has the advantage



(a) Control space, varying c_b

(b) Control space, varying ℓ

(c) Objective space, varying c_b

(d) Objective space, varying ℓ

Figure 7: Efficiencies of the a-posteriori error estimate.

that the POD basis is computed using a snapshot space, which already contains strong dynamics. In both versions we start with $\ell_0 = 2$ basis functions. The Figures 8 show exactly the difference of the two approaches. While in the case of the computation direction being from top to bottom the POD basis is mainly extended during the first optimization problem, the POD basis is sequentially extended in the second approach. However, both approaches end with the same number of 15 basis functions. For both versions the control of the approximation error works very efficiently. The instances in which the real error overshoots the a-posteriori estimate are due to inaccuracies in the computation of the solution of the full system. As the POD basis gets immediately updated to 13 basis function in the beginning for the version from top to bottom, it is clear that this approach yields better results than the version from bottom to top.

In a last step we compare the computational times for different settings. We see in Table 1 that the algorithms with a fixed number of basis functions perform the fastest. They are up to a factor 30 faster than the full system. However, there is the risk that on the one hand, not enough basis functions are chosen, which leads to bad convergence properties and thus high computational times (see the results for $\ell = 2$) and on the other hand, there is no error control during the algorithm, which might lead to unsatisfying results.

In comparison to that the adaptive basis extension algorithms need more time. One share of the additional time consumption is simply that the a-posteriori estimate has to be computed after each optimization problem. For this purpose

Algorithm 3 (Adaptive basis extension algorithm)

Require: Threshold $\mu > 0$.

while 1 **do**
  Solve $(\hat{\mathbf{P}}_z^\ell)$ with reference point $z^{(n)}$ with Algorithm 2.
  Compute the a-posteriori estimate μ_{apost} from (11) for the controls.
  if $\mu_{apost} > \mu$ **then**
    Set $\ell = \ell + 1$
  else
    return

	$c_b = 0$	$c_b = 0.1$	$c_b = 0.5$	$c_b = 1$
Full system	566.3 s	678.8 s	729.2 s	701.5 s
$\ell = 2$	23.6 s	69.6 s	103.8 s	164.2s
$\ell = 5$	27.3 s	20.5 s	27.9 s	39.4 s
$\ell = 10$	29.2 s	22.6 s	56.6 s	49.3 s
Algorithm 6 from top to bottom	49.0 s	101.4 s	150.8 s	184.1 s
Algorithm 6 from bottom to top	50.7 s	87.6 s	148.4 s	175.9 s

Table 1: Computation times for the different methods and different values of c_b.

both the state and adjoint equation have to be solved once with the full system. The second reason for the higher computation time is the basis extension itself. Each time the basis is extended, the current optimization problem has to be solved again using the larger POD system. In our results we observe an 13 basis extensions for both algorithms, i.e. 13 optimization problems are computed in vain. Yet, they are still faster by a factor of 5-10 than solving the full system and their big advantage is that they can control the maximal approximation error.

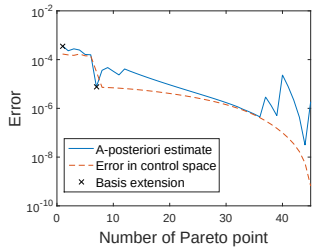
References

[1] S. Banholzer. *POD-Based Bicriterial Optimal Control of Convection-Diffusion Equations*. Master thesis, University of Konstanz, Department of Mathematics and Statistics, 2017.

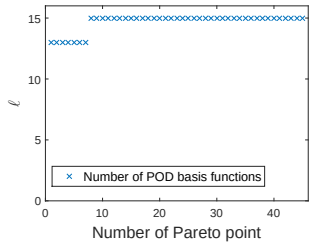
[2] S. Banholzer, D. Beermann, and S. Volkwein. POD-Based bicriterial optimal control by the reference point method. IFAC-PapersOnLine, 49:210-215, 2016.

[3] R. Dautray and J.-L. Lions. *Mathematical Analysis and Numerical Methods for Science and Technology. Volume 5: Evolution Problems I.* Springer-Verlag, Berlin, 2000.

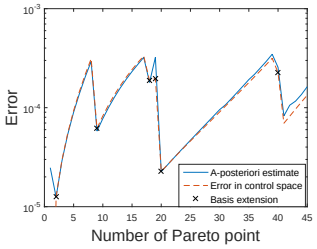
[4] M. Ehrgott. *Multicriteria Optimization.* Springer, Berlin, 2005.



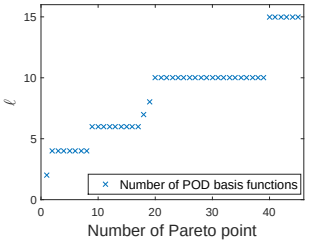
(a) Error in the control space, top to bottom



(b) Number of used POD basis functions, top to bottom



(c) Error in the control space, bottom to top



(d) Number of used POD basis functions, bottom to top

Figure 8: Results for the POD extension algorithm

[5] K. F. Fong, V. I. Hanby, and T.-T. Chow. HVAC system optimization for energy management by evolutionary programming. *Energy and Buildings*, 38:220-231, 2006.

[6] 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.

[7] 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. edition, 2012.

[8] L. Iapichino, S. Trenz and S. Volkwein: Reduced-order multiobjective optimal control of semilinear parabolic problems. *Lecture Notes in Computational Science and Engineering*, 112:389-397, 2016.

[9] L. Iapichino, S. Ulbrich and S. Volkwein: Multiobjective PDE-constrained optimization using the reduced-basis method. *Advances in Computational Mathematics*, to appear, 2017, see `DOI: 10.1007/s10444-016-9512-x`

[10] E. Kammann, F. Tröltzsch and S. Volkwein: A posteriori error estimation for semilinear parabolic optimal control problems with application to model reduction by POD. *ESAIM: Mathematical Modelling and Numerical Analysis*, 47:555-581, 2013.

[11] A. Kusiak, F. Tang, and G. Xu. Multi-objective optimization of HVAC system with an evolutionary computation algorithm. *Energy*, 36:2440-2449, 2011.

[12] K. Miettinen: *Nonlinear Multiobjective Optimization*. International Series in Operations Research & Management Science, Springer, 1998.

[13] S. Peitz, S. Oder-Blöbaum and M. Dellnitz. Multiobjective optimal control methods for fluid flow using reduced order modeling. *24th Congress of Theoretical and Applied Mechanics (ICTAM)*, 21-26 August 2016, Montreal, Canada, see `http://arxiv.org/pdf/1510.05819v2.pdf`.

[14] C. Romaus, J. Böcker, K. Witting, A. Seifried, and O. Znamenshchykov. Optimal energy management for a hybrid energy storage system combining batteries and double layer capacitors. *IEEE*, pages 1640-1647, San Jose, CA, USA, 2009.

[15] J. R. Singler. New POD error expressions, error bounds, and asymptotic results for reduced order models of parabolic PDEs. *SIAM Journal on Numerical Analysis*, 52:852-876, 2014.

[16] W. Stadler. *Multicriteria Optimization in Engineering and in the Sciences*. Plenum Press, New York, 1988.

[17] F. Tröltzsch. *Optimal Control of Partial Differential Equations: Theory, Methods and Applications*. AMS American Mathematical Society, 2nd ed., 2010.

[18] F. Tröltzsch and S. Volkwein: POD a-posteriori error estimates for linear-quadratic optimal control problems. *Computational Optimization and Applications*, 44:83-115, 2009.