DFG Deutsche Forschungsgemeinschaft
Priority Programme 1962

Multiobjective Model Predictive Control of a Parabolic Advection-Diffusion-Reaction Equation

Stefan Banholzer, Giulia Fabrini, Lars Grüne, Stefan Volkwein

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

Preprint Number SPP1962-139

received on April 18, 2020

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/

MDPI

Article

Multiobjective Model Predictive Control of a Parabolic Advection-Diffusion-Reaction Equation

Stefan Banholzer [1], Giulia Fabrini [1], Lars Grüne [2] and Stefan Volkwein [1]

[1] University of Konstanz; Department of Mathematics and Statistics; D-78457 Konstanz, Germany
[2] University of Bayreuth; Mathematical Institute; D-95440 Bayreuth, Germany
* Correspondence: stefan.banholzer@uni-konstanz.de (S.B.), lars.gruene@uni-bayreuth.de (L.G.), stefan.volkwein@uni-konstanz.de (S.V.)

Version April 8, 2020 submitted to Journal Not Specified

Abstract: In the present paper a multiobjective optimal control problem governed by a linear parabolic advection-diffusion-reaction equation is considered. The optimal controls are computed by applying model predictive control (MPC), which is a method for controlling dynamical systems over long or infinite time horizons by successively computing optimal controls over a moving finite time horizon. Numerical experiments illustrate that the proposed solution approach can be successfully applied although some of the assumptions made in [1,2] can not be guaranteed for the studied tests.

Keywords: Multiobjectice optimization; multiobjective optimal control; model predictive control; evolution problems; advection-diffusion equations.

1. Introduction

Model predictive control (MPC) is a method for controlling dynamical systems over long or infinite time horizons by successively computing optimal controls over a moving finite time horizon; cf., e.g., [3]. The successive re-optimization on the one hand introduces a feedback mechanism, which makes the method attractive as a real-time control scheme and as such it is widely used in industry [4]. On the other hand, MPC provides a method to reduce the complexity of optimal control problems by splitting up problems on long or infinite horizons into smaller subproblems over shorter horizons. As such, MPC can be seen as a model reduction technique in time. Clearly, this approach only makes sense if the solution generated by MPC is in some sense close to the true optimal solution. Fortunately, this approximate optimality property can be rigorously proved for many classes of optimal control problems, see, e.g., [5,6] and the references therein.

In practice, it is often the case that a single optimization criterion is not sufficient for modelling the demands in a given application. This leads to the concept of multiobjective optimal control, in which typically there does not exist a single optimal value but rather a whole set of optimal values, the so called Pareto front. Due to its usefulness in practice it is no surprise that this concept has been used and investigated also in the MPC context, see, e.g., [7–11]. Particularly, in [1,2] MPC algorithms were presented that allow for rigorous suboptimality estimates also in the multiobjective case. The main feature of this class of algorithms is a particular constraint that depends on the chosen point on the Pareto front when solving the first optimal control problem in the MPC scheme and that is applied in all subsequent optimal control problems. While the examples in [1,2] are limited to finite dimensional systems, in this paper we show that this idea can be successfully applied also to optimal control problems governed by partial differential equations (PDEs) and that the necessary constraint can be efficiently implemented using a gradient descent scheme. While we will see that some of the assumptions made in [1,2] are difficult to satisfy for PDE governed problems, our numerical tests will show that the algorithm nevertheless performs very well.

In the literature there is a vast amount of methods for multiobjectice optimization; see e.g. [12,13]. However, in the context of PDE constrained problems there are only a few papers available. We refer, e.g., to [14–18]. In this paper, for computing the Pareto set and/or the Pareto front of (MOCP) we use the Euclidean reference point method, which was introduced in [19,20] and already used for solving multiobjective optimal control problems with two cost functions in [21–23].

The paper is organized as follows: In Section 2 we pose the multiobjective optimal control problem (MOCP). The algorithmic approach for multiobjectice model predictive control (MOMPC) is explained and analyzed in Section 3. Numerical experiments are presented in Section 4. In the last section we draw a few conclusions.

2. The multiobjective optimal control problem

Let $\Omega \subset \mathbb{R}^2$ be a bounded domain with Lipschitz-continuous boundary $\Gamma = \partial\Omega$, $T > 0$, $Q = (0,T) \times \Omega$, $\Sigma = (0,T) \times \Gamma$. Given an initial condition $y_\circ \in L^2(\Omega)$ we consider the following parametrized parabolic advection-diffusion-reaction equation with the diffusion coefficient $d > 0$, the advection field $c \in L^\infty((0,T) \times \Omega; \mathbb{R}^2)$ and the reaction coefficient $r \geq 0$:

$$\begin{aligned} y_t(t,\boldsymbol{x}) - d\Delta y(t,\boldsymbol{x}) + c(t,\boldsymbol{x}) \cdot \nabla y(t,\boldsymbol{x}) + r y(t,\boldsymbol{x}) &= 0 && \text{a.e. in } Q, \\ d\,\frac{\partial y}{\partial \boldsymbol{n}}(t,\boldsymbol{x}) &= \sum_{i=1}^{m} u_i(t) b_i(\boldsymbol{x}) && \text{a.e. on } \Sigma, \\ y(0,\boldsymbol{x}) &= y_\circ(\boldsymbol{x}) && \text{a.e. in } \Omega. \end{aligned} \tag{1}$$

The functions $b_i \in L^\infty(\Gamma)$, $1 \leq i \leq m$, denote given control shape functions, e.g., characteristic functions:

$$\Gamma = \bigcup_{i=1}^{m} \Gamma^i, \quad b_i(\boldsymbol{s}) = \chi_{\Gamma^i}(\boldsymbol{s}),\ 1 \leq i \leq m,\ \|b_i\|^2_{L^2(\Gamma)} = |\Gamma^i|.$$

By setting $H = L^2(\Omega)$ and $V = H^1(\Omega)$, one can show that (1) has a unique weak solution $y = y(u, y_\circ) \in W(0,T) = H^1(0,T;V') \cap L^2(0,T;V)$ for all controls $u \in L^2(0,T;\mathbb{R}^m) =: U$ and all initial conditions $y_\circ \in H$. Recall that $W(0,T)$ is a Hilbert space endowed with its common inner product. Moreover, $W(0,T)$ is continuously embedded into $C([0,T];H)$; cf. [24]. The solution operator $S : U \times H \to W(0,T) \hookrightarrow L^2(0,T;H)$, which maps any control $u \in U$ and initial value $y_\circ \in H$ to the solution $y = S(u, y_\circ)$ of (1), is affine linear and continuous in both components.

Given a fixed initial value $y_\circ \in H$ the multiobjective optimal control problem (MOCP) reads

$$\min_{u \in U} \hat{J}(u, y_\circ) = \min_{u \in U} \begin{pmatrix} \hat{J}_1(u, y_\circ) \\ \hat{J}_2(u, y_\circ) \end{pmatrix}, \tag{MOCP}$$

where the two objectives $\hat{J}_1, \hat{J}_2 : U \times H \to \mathbb{R}$ are given by

$$\hat{J}_i(u, y_\circ) := \frac{\sigma_i^1}{2} \|\mathcal{C}_i^1 S(u, y_\circ) - y_i^1\|^2_{L^2(0,T;H)} + \frac{\sigma_i^2}{2} \|\mathcal{C}_i^2 S(u, y_\circ)(T) - y_i^2\|^2_H + \frac{\sigma_i^3}{2} \|u\|^2_{L^2(0,T;\mathbb{R}^m)} \tag{2}$$

for nonnegative weighting parameters σ_i^1, σ_i^2, σ_i^3 with $\sigma_i^1 + \sigma_i^2 + \sigma_i^3 > 0$, linear mappings $\mathcal{C}_i^1 \in \mathcal{L}(L^2(0,T;H))$ and $\mathcal{C}_i^2 \in \mathcal{L}(H)$ and desired states $y_i^1 \in L^2(0,T;H)$ and $y_i^2 \in H$ for $i = 1, 2$. Here, e.g., $\mathcal{L}(L^2(0,T;H))$ denotes the Banach space of all linear and bounded operators mapping from $L^2(0,T;H)$ into itself.

Problem (MOCP) involves the minimization of a vector-valued objective $\hat{J}(\cdot, y_\circ) : U \to \mathbb{R}^2$. This is done by using the concept of Pareto optimality; see, e.g., [12].

Definition 1. *The point $\bar{u} \in U$ is called* Pareto optimal *for* (MOCP)*, if there is no other control $u \in U$ with $\hat{J}_i(u, y_\circ) \leq \hat{J}_i(\bar{u}, y_\circ)$ for $i = 1, 2$ and $\hat{J}_l(u, y_\circ) < \hat{J}_l(\bar{u}, y_\circ)$ for at least one $l \in \{1, 2\}$. The set*

$$\mathcal{U}_{opt}(y_\circ) = \{u \in U \mid u \text{ is Pareto optimal}\}$$

is called the Pareto set*, and*

$$\mathcal{J}_{opt}(y_\circ) = \hat{J}(\mathcal{U}_{opt}, y_\circ)$$

is called the Pareto front *of* (MOCP).

3. Multiobjective model predictive control (MOMPC)

In this section we explain how MPC is combined with multiobjective optimal control. Given a final (large) time horizon $T > 0$ we use an equidistant time grid

$$0 = t_0 < t_1 < \ldots < t_L = T \quad \text{with} \quad t_n = n\Delta t \text{ for } n = 0, \ldots, L \text{ and } \Delta t = \frac{T}{L}.$$

The open-loop problems in the MPC algorithm are then solved on smaller time horizons $\ell \Delta t$ with $1 < \ell < L$. We set

$$t_{\mathsf{end}} = t_{L-\ell+1} = (L - \ell + 1)\Delta t \leq T.$$

By the MPC method we will finally compute an optimal control on the time horizon $[0, t_{\mathsf{end}}]$. For $n = 0, \ldots, L - \ell$ the optimal control to this problem on the interval $(t_n, t_n + \Delta t) = (t_n, t_{n+1})$ is stored and used to compute the next part of the state trajectory on the interval (t_n, t_{n+1}). Then the time horizon is shifted by Δt and the procedure is repeated until the final time horizon is reached. Notice that the end time for the computed MPC control is t_{end}.

To deal with this framework we have to consider (1) on subintervals of $[0, T]$. To this end we introduce further notations: For an arbitrary initial time $t_{\mathsf{a}} \in [0, T)$ and end time $t_{\mathsf{b}} \in (t_{\mathsf{a}}, T]$, we study the state equation on the time horizon $\mathcal{I} = (t_{\mathsf{a}}, t_{\mathsf{b}})$:

$$\begin{aligned} y_t(t, \boldsymbol{x}) - d\Delta y(t, \boldsymbol{x}) + c(t, \boldsymbol{x}) \cdot \nabla y(t, \boldsymbol{x}) + r y(t, \boldsymbol{x}) &= 0 && \text{a.e. in } \mathcal{I} \times \Omega, \\ d \frac{\partial y}{\partial \boldsymbol{n}}(t, \boldsymbol{x}) &= \sum_{i=1}^{m} u_i(t) b_i(\boldsymbol{x}) && \text{a.e. on } \mathcal{I} \times \Gamma, \\ y(t_{\mathsf{a}}, \boldsymbol{x}) &= y_{\mathsf{a}}(\boldsymbol{x}) && \text{a.e. in } \Omega \end{aligned} \tag{3}$$

for some initial value $y_{\mathsf{a}} \in H$. Again it is possible to show that for each $u \in U^{\mathcal{I}} = L^2(\mathcal{I}; \mathbb{R}^m)$ and $y_{\mathsf{a}} \in H$ there exists a unique weak solution $y = y(u, y_{\mathsf{a}}) \in W(\mathcal{I})$ to (3) and that the associated solution operator $S^{\mathcal{I}} : U^{\mathcal{I}} \times H \to W(\mathcal{I}) \hookrightarrow L^2(\mathcal{I}; H)$ is affine linear and continuous in both components.

The two objectives $\hat{J}_1^{\mathcal{I}}, \hat{J}_2^{\mathcal{I}} : U^{\mathcal{I}} \times H \to \mathbb{R}$ are given by

$$\hat{J}_i^{\mathcal{I}}(u, y_{\mathsf{a}}) := \frac{\sigma_i^1}{2} \|\mathcal{C}_i^1 S^{\mathcal{I}}(u, y_{\mathsf{a}}) - y_i^1\|^2_{L^2(\mathcal{I}; H)} + \frac{\sigma_i^2}{2} \|\mathcal{C}_i^2 S^{\mathcal{I}}(u, y_{\mathsf{a}})(t_{\mathsf{b}}) - y_i^2\|_H^2 + \frac{\sigma_i^3}{2} \|u\|^2_{L^2(\mathcal{I}; \mathbb{R}^m)} \tag{4}$$

for $i = 1, 2$. Note that we would actually have to write $\mathcal{C}_i^{1^{\mathcal{I}}} \in L^2(\mathcal{I}; H)$ for $i = 1, 2$ to display the dependence of the operators on the time interval $\mathcal{I}$. Since we define these operators in a canonical way by

$$\mathcal{C}_i^{1^{\mathcal{I}}} f := \Big(\mathcal{C}_i^1(\mathcal{E} f)\Big)\Big|_{\mathcal{I}} \quad \text{for } f \in L^2(\mathcal{I}; H),$$

where $\mathcal{E} : L^2(\mathcal{I}; H) \hookrightarrow L^2(0, T; H)$ extends any function $f \in L^2(\mathcal{I}; H)$ by 0 to a function in $L^2(0, T; H)$, we omit this dependence on $\mathcal{I}$ for the purpose of readability.

Throughout we denote by $\mathcal{U}_{opt}^{\mathcal{J}}(y_a)$ and $\mathcal{J}_{opt}^{\mathcal{J}}(y_a)$ the Pareto set and Pareto front, respectively, of the MOCP

$$\min_{u\in U^{\mathcal{J}}} \hat{J}^{\mathcal{J}}(u, y_a) = \min_{u\in U^{\mathcal{J}}} \begin{pmatrix} \hat{J}_1^{\mathcal{J}}(u, y_a) \\ \hat{J}_2^{\mathcal{J}}(u, y_a) \end{pmatrix}. \tag{MOCP$^{\mathcal{J}}$}$$

Especially for the open-loop problems in the MPC algorithm we make the following notations: Given $n \in \{0, \dots, L-\ell\}$ we set $t_n^\ell = t_n + \ell\Delta t \leq T$ for the final time instance and $U^n = L^2(t_n, t_n^\ell; \mathbb{R}^m)$. Furthermore, for brevity we define $S^n = S^{(t_n, t_n^\ell)}$, $\hat{J}_i^n = \hat{J}_i^{(t_n, t_n^\ell)}$ for $i = 1, 2$, and $\mathcal{U}_{opt}^n(y_a) = \mathcal{U}_{opt}^{(t_n, t_n^\ell)}(y_a)$ as well as $\mathcal{J}_{opt}^n(y_a) = \mathcal{J}_{opt}^{(t_n, t_n^\ell)}(y_a)$. In the context of MOMPC the notion of external stability turns out to be important, see [1, Definition 6] and Theorem 2 below.

Definition 2. *Let $\mathcal{J} = (t_a, t_b) \subset [0, T]$ be a time interval and the initial condition $y_a \in H$ be arbitrary. The set $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a)$ is called* externally stable, *if for every $y \in \hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a)$ there is $\bar{y} \in \mathcal{J}_{opt}^{\mathcal{J}}(y_a)$ with $\bar{y} \leq y$. This is equivalent to $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) \subset \mathcal{J}_{opt}^{\mathcal{J}}(y_a) + \mathbb{R}^2_{\geq 0}$.*

For the cost functions defined in (4) it is possible to show that the set $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a)$ is externally stable, as the following theorem shows.

Theorem 1. *Let $\mathcal{J} = (t_a, t_b) \subset [0, T]$ be a time interval and the initial condition $y_a \in H$ be arbitrarily given. We assume that $\sigma_i^3 > 0$ for $i = 1, 2$. Then the set $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a)$ is externally stable.*

Proof. It is possible to show that the Pareto optimal points of $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a)$ and $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0}$ are the same, i.e., it holds $\mathcal{J}_{opt}^{\mathcal{J}}(y_a) = (\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0})_{opt}$. So if we show that the set $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0}$ is externally stable, i.e., that it holds $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0} \subset (\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0})_{opt} + \mathbb{R}^2_{\geq 0}$, we are done, since then we have

$$\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) \subset \hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0} \subset (\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0})_{opt} + \mathbb{R}^2_{\geq 0} = \mathcal{J}_{opt}^{\mathcal{J}}(y_a) + \mathbb{R}^2_{\geq 0}.$$

According to [12, Theorem 2.21] we only need to show that $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0} \neq \emptyset$ and that for all $y \in \hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0}$ the set $(\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0}) \cap (y - \mathbb{R}^2_{\geq 0})$ is compact (this is the notion of $\mathbb{R}^2_{\geq 0}$-compactness). It is clear that $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) \neq \emptyset$ holds.
Since the set $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0}$ is bounded from below and $y - \mathbb{R}^2_{\geq 0}$ is bounded from above, the set $(\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0}) \cap (y - \mathbb{R}^2_{\geq 0})$ is bounded.
The closedness of $\hat{J}^{\mathcal{J}}(U^{\mathcal{J}}, y_a) + \mathbb{R}^2_{\geq 0}$ follows from the convexity, the continuity and the property

$$\lim_{\|u\|_{U^{\mathcal{J}}} \to 0} \hat{J}_i^{\mathcal{J}}(u, y_a) = \infty,$$

which follows from $\sigma_i^3 > 0$ for $i = 1, 2$, of the cost functions $\hat{J}_1^{\mathcal{J}}(\cdot, y_a)$, $\hat{J}_2^{\mathcal{J}}(\cdot, y_a)$, and was for example shown in the proof of Theorem 3.35 in [22]. □

The pseudocode for a general MOMPC algorithm is shown in Algorithm 1. Here we use a fixed time horizon Δt on which the computed control is applied before a re-optimization is performed. We note that it may be possible to use varying time horizons here, similar as discussed for the single-objective case, e.g., in [6, Section 10.4]. The most important question is how to perform step 5 of Algorithm 1. Clearly, one possibility is to compute the entire Pareto set $\mathcal{U}_{opt}^n(y_a)$, which is nonempty by Theorem 1, and then to choose a control $\bar{u}_n \in \mathcal{U}_{opt}^n(y_a)$ according to some given preferences. However, this procedure is computationally costly, and thus, infeasible for many applications. On the other hand, one could compute only one Pareto optimal control $\bar{u}_n \in \mathcal{U}_{opt}^n(y_a)$. Then the question is under

Algorithm 1 (Multiobjective model predictive control)

Given: Initial state $y_\circ \in H$.
1: Output: MOMPC feedback control μ.
2: Set $y_0(t_0) = y_\circ$.
3: **for** $n = 0$ **to** $L - \ell$ **do**
4: Set $t_\mathsf{a} = t_n$, $t_\mathsf{b} = t_\mathsf{a} + \ell\Delta t$ and $y_\mathsf{a} = y_n(t_n)$.
5: Compute a Pareto optimal control $\bar{u}_n \in \mathcal{U}^n_{opt}(y_\mathsf{a})$.
6: Set $y_{n+1}(t) = y_n(t)$ for all $t \in [0, t_n]$ and compute $y_{n+1}(t) = S^n(\bar{u}_n, y_\mathsf{a})(t)$ for all $t \in (t_n, t_{n+1}]$.
7: Set $\mu(t) = \bar{u}_n(t)$ for all $t \in (t_n, t_{n+1}]$.
8: **end for**

which criteria this control should be computed. In [1] the authors allowed an arbitrary choice for $\bar{u}_0 \in \mathcal{U}^0_{opt}(y_\circ)$ and used the additional constraint

$$\hat{J}^n_i(\bar{u}_n, y_\mathsf{a}) \leq \hat{J}^n_i(\tilde{u}, y_\mathsf{a}), \quad \text{for } i = 1, 2 \text{ and } y_\mathsf{a} = y_n(t_n), \tag{5}$$

for all $n \in \{1, \dots, L - \ell\}$, where $\tilde{u}$ is given by

$$\tilde{u}(t) = \begin{cases} \bar{u}_{n-1}(t), & \text{if } t \in [t_n, t^\ell_{n-1}] = [t_n, t_{n-1+\ell}], \\ \kappa(t), & \text{if } t \in (t^\ell_{n-1}, t^\ell_n] = (t_{n-1+\ell}, t_{n+\ell}] \end{cases} \tag{6}$$

and the feedback $\kappa \in U^{(t^\ell_{n-1}, t^\ell_n)}$ is chosen such that

$$\begin{aligned} \frac{\sigma^1_i}{2} \left\| \mathcal{C}^1_i S^{(t^\ell_{n-1}, t^\ell_n)}(\kappa, y_\kappa) - y^1_i \right\|^2_{L^2(t^\ell_{n-1}, t^\ell_n; H)} + \frac{\sigma^2_i}{2} \left\| \mathcal{C}^2_i S^{(t^\ell_{n-1}, t^\ell_n)}(\kappa, y_\kappa)(t^\ell_n) - y^2_i \right\|^2_H + \frac{\sigma^3_i}{2} \|\kappa\|^2_{L^2(t^\ell_{n-1}, t^\ell_n; \mathbb{R}^m)} \\ \leq \frac{\sigma^2_i}{2} \left\| \mathcal{C}^2_i S^{n-1}(\bar{u}_{n-1}, y_{n-1}(t_{n-1}))(t^\ell_{n-1}) - y^2_i \right\|^2_H, \end{aligned} \tag{7}$$

holds for $i = 1, 2$, with $y_\kappa = S^{n-1}(\bar{u}_{n-1}, y_{n-1}(t_{n-1}))(t^\ell_{n-1})$. The external stability of the set $\hat{J}^n(U^n, y_n(t_n))$ ensures that such a control $\bar{u}_n$ exists.

Theorem 2. *Let $n \in \{0, \dots, L - \ell\}$ and $y_\mathsf{a} = y_n(t_n)$. Given $\tilde{u} \in U^n$ there is $\bar{u}_n \in \mathcal{U}^n_{opt}(y_\mathsf{a})$ such that (5) is satisfied.*

Proof. This follows directly from Theorem 1, where it was shown that the set $\hat{J}^n(U^n, y_\mathsf{a})$ is externally stable. □

However, the existence of a feedback $\kappa \in U^{(t^\ell_{n-1}, t^\ell_n)}$ fulfilling (7) cannot be shown in general for our problem, but has to be assumed. Such a feedback exists if and only if the minimal value of the minimization problem

$$\begin{aligned} \min_{\kappa \in U^{(t^\ell_{n-1}, t^\ell_n)}} \max_{i \in \{1,2\}} \frac{\sigma^1_i}{2} \left\| \mathcal{C}^1_i S^{(t^\ell_{n-1}, t^\ell_n)}(\kappa, y_\kappa) - y^1_i \right\|^2_{L^2(t^\ell_{n-1}, t^\ell_n; H)} + \frac{\sigma^2_i}{2} \left\| \mathcal{C}^2_i S^{(t^\ell_{n-1}, t^\ell_n)}(\kappa, y_\kappa)(t^\ell_n) - y^2_i \right\|^2_H \\ + \frac{\sigma^3_i}{2} \|\kappa\|^2_{L^2(t^\ell_{n-1}, t^\ell_n; \mathbb{R}^m)} - \frac{\sigma^2_i}{2} \left\| \mathcal{C}^2_i S^{n-1}(\bar{u}_{n-1}, y_{n-1}(t_{n-1}))(t^\ell_{n-1}) - y^2_i \right\|^2_H. \end{aligned} \tag{8}$$

is smaller or equal than zero. Thus, by solving the minimization problem (8) we can on the one hand check, if such a feedback exists, and on the other hand compute it explicitly.

If the conditions (5) and (7) are fulfilled throughout Algorithm 1 we can show the following performance theorem, compare [1, Theorem 11], where also the proof was taken from and adapted to our situation.

Theorem 3. *Let $y_\circ \in H$ be arbitrary and assume that $\sigma_i^3 > 0$ holds for $i = 1, 2$. Denote by $\bar{u}_0 \in \mathcal{U}_{opt}^0(y_\circ)$ the initially chosen Pareto optimal control in Algorithm 1. Furthermore, assume that it is possible to choose a feedback $\kappa \in U^{(t_{n-1}^\ell, t_n^\ell)}$ fulfilling (7) for every $n \in \{1, \dots, L-\ell\}$. Let $\mu \in U^{(0,t_{\mathsf{end}})}$ be the resulting MOMPC feedback control resulting from Algorithm 1. Then it holds*

$$\hat{J}_{i,perf}^{(0,t_{\mathsf{end}})}(\mu, y_\circ) = \frac{\sigma_i^1}{2} \left\| \mathcal{C}_i^1 S^{(0,t_{\mathsf{end}})}(\mu, y_\circ) - y_i^1 \right\|^2_{L^2(0,t_{\mathsf{end}};H)} + \frac{\sigma_i^3}{2} \|\mu\|^2_{L^2(0,t_{\mathsf{end}};\mathbb{R}^m)} \leq \hat{J}_i^0(\bar{u}_0, y_\circ)$$

for $i = 1, 2$.

Proof. Note that it holds $\mu(t) = \bar{u}_n(t)$ for almost all $t \in (t_n, t_{n+1})$ and all $n \in \{0, \dots, L-\ell\}$ and that

$$\bigcup_{n=0}^{L-\ell} (t_n, t_{n+1}) = (0, t_{\mathsf{end}})$$

holds true. By defining $y_n = S^{(0,t_{\mathsf{end}})}(\mu, y_\circ)(t_n)$ for all $n \in \{0, \dots, L-\ell\}$ we can write

$$\begin{aligned}
&\frac{\sigma_i^1}{2} \left\| \mathcal{C}_i^1 S^{(0,t_{\mathsf{end}})}(\mu, y_\circ) - y_i^1 \right\|^2_{L^2(0,t_{\mathsf{end}};H)} + \frac{\sigma_i^3}{2} \|\mu\|^2_{L^2(0,t_{\mathsf{end}};\mathbb{R}^m)} \\
&= \frac{1}{2} \sum_{n=0}^{L-\ell} \left[\sigma_i^1 \left\| \mathcal{C}_i^1 S^n(\bar{u}_n, y_n) - y_i^1 \right\|^2_{L^2(t_n,t_{n+1};H)} + \sigma_i^3 \|\bar{u}_n\|^2_{L^2(t_n,t_{n+1};\mathbb{R}^m)} \right] \\
&= \frac{1}{2} \sum_{n=0}^{L-\ell} \Big[\sigma_i^1 \left\| \mathcal{C}_i^1 S^n(\bar{u}_n, y_n) - y_i^1 \right\|^2_{L^2(t_n,t_n^\ell;H)} + \sigma_i^3 \|\bar{u}_n\|^2_{L^2(t_n,t_n^\ell;\mathbb{R}^m)} \\
&\qquad + \sigma_i^2 \left\| \mathcal{C}_i^2 S^n(\bar{u}_n, y_n)(t_n^\ell) - y_i^2 \right\|^2_H - \sigma_i^1 \left\| \mathcal{C}_i^1 S^n(\bar{u}_n, y_n) - y_i^1 \right\|^2_{L^2(t_{n+1},t_n^\ell;H)} \\
&\qquad - \sigma_i^3 \|\bar{u}_n\|^2_{L^2(t_{n+1},t_n^\ell;\mathbb{R}^m)} - \sigma_i^2 \left\| \mathcal{C}_i^2 S^n(\bar{u}_n, y_n)(t_n^\ell) - y_i^2 \right\|^2_H \Big].
\end{aligned} \tag{9}$$

Since we assume the existence of a feedback κ fulfilling (7), we can conclude

$$\begin{aligned}
&- \sigma_i^2 \left\| \mathcal{C}_i^2 S^n(\bar{u}_n, y_n)(t_n^\ell) - y_i^2 \right\|^2_H \\
&\leq -\sigma_i^1 \left\| \mathcal{C}_i^1 S^{n+1}(\tilde{u}_{n+1}, y_{n+1}) - y_i^1 \right\|^2_{L^2(t_n^\ell,t_{n+1}^\ell;H)} \\
&\quad - \sigma_i^2 \left\| \mathcal{C}_i^2 S^{n+1}(\tilde{u}_{n+1}, y_{n+1})(t_{n+1}^\ell) - y_i^2 \right\|^2_H - \frac{\sigma_i^3}{2} \|\kappa\|^2_{L^2(t_n^\ell,t_{n+1}^\ell;\mathbb{R}^m)}
\end{aligned} \tag{10}$$

for all $n \in \{0, \dots, L-\ell\}$ with $\tilde{u}_{n+1} \in U^{(t_{n+1},t_{n+1}^\ell)}$ defined as in (6). Due to (6) we find that

$$\|\bar{u}_n\|^2_{L^2(t_{n+1},t_n^\ell;\mathbb{R}^m)} + \|\kappa\|^2_{L^2(t_n^\ell,t_{n+1}^\ell;\mathbb{R}^m)} = \|\tilde{u}_{n+1}\|^2_{L^2(t_{n+1},t_{n+1}^\ell;\mathbb{R}^m)}. \tag{11}$$

Moreover, it follows that

$$\begin{aligned}
&\left\| \mathcal{C}_i^1 S^n(\bar{u}_n, y_n) - y_i^1 \right\|^2_{L^2(t_{n+1},t_n^\ell;H)} + \left\| \mathcal{C}_i^1 S^{n+1}(\tilde{u}_{n+1}, y_{n+1}) - y_i^1 \right\|^2_{L^2(t_n^\ell,t_{n+1}^\ell;H)} \\
&\quad = \left\| \mathcal{C}_i^1 S^{n+1}(\tilde{u}_{n+1}, y_{n+1}) - y_i^1 \right\|^2_{L^2(t_{n+1},t_{n+1}^\ell;H)}.
\end{aligned} \tag{12}$$

Plugging (10) into (9) and utilizing (11)-(12) we deduce that

$$\begin{aligned}
&\frac{\sigma_i^1}{2}\left\|\mathcal{C}_i^1 S^{(0,t_{\mathsf{end}})}(\mu,y_\circ)-y_i^1\right\|^2_{L^2(0,t_{\mathsf{end}};H)}+\frac{\sigma_i^3}{2}\left\|\mu\right\|^2_{L^2(0,t_{\mathsf{end}};\mathbb{R}^m)}\\
&\leq \frac{1}{2}\sum_{n=0}^{L-\ell}\Big[\sigma_i^1\left\|\mathcal{C}_i^1 S^n(\bar{u}_n,y_n)-y_i^1\right\|^2_{L^2(t_n,t_n^\ell;H)}+\sigma_i^3\left\|\bar{u}_n\right\|^2_{L^2(t_n,t_n^\ell;\mathbb{R}^m)}\\
&\qquad+\sigma_i^2\left\|\mathcal{C}_i^2 S^n(\bar{u}_n,y_n)(t_n^\ell)-y_i^2\right\|^2_{H}-\sigma_i^1\left\|\mathcal{C}_i^1 S^{n+1}(\tilde{u}_{n+1},y_{n+1})-y_i^1\right\|^2_{L^2(t_{n+1},t_{n+1}^\ell;H)}\\
&\qquad-\sigma_i^3\left\|\tilde{u}_{n+1}\right\|^2_{L^2(t_{n+1},t_{n+1}^\ell;\mathbb{R}^m)}-\sigma_i^2\left\|\mathcal{C}_i^2 S^{n+1}(\tilde{u}_{n+1},y_{n+1})(t_{n+1}^\ell)-y_i^2\right\|^2_{H}\Big].
\end{aligned}$$

Since the control $\bar{u}_{n+1}$ fulfills (5) for $n\in\{0,\dots,L-\ell-1\}$, we can further estimate

$$\begin{aligned}
&\frac{\sigma_i^1}{2}\left\|\mathcal{C}_i^1 S^{(0,t_{\mathsf{end}})}(\mu,y_\circ)-y_i^1\right\|^2_{L^2(0,t_{\mathsf{end}};H)}+\frac{\sigma_i^3}{2}\left\|\mu\right\|^2_{L^2(0,t_{\mathsf{end}};\mathbb{R}^m)}\\
&\leq \frac{1}{2}\sum_{n=0}^{L-\ell}\Big[\sigma_i^1\left\|\mathcal{C}_i^1 S^n(\bar{u}_n,y_n)-y_i^1\right\|^2_{L^2(t_n,t_n^\ell;H)}+\sigma_i^3\left\|\bar{u}_n\right\|^2_{L^2(t_n,t_n^\ell;\mathbb{R}^m)}\\
&\qquad+\sigma_i^2\left\|\mathcal{C}_i^2 S^n(\bar{u}_n,y_n)(t_n^\ell)-y_i^2\right\|^2_{H}\Big]\\
&\quad-\frac{1}{2}\sum_{n=0}^{L-\ell-1}\Big[\sigma_i^1\left\|\mathcal{C}_i^1 S^{n+1}(\bar{u}_{n+1},y_{n+1})-y_i^1\right\|^2_{L^2(t_{n+1},t_{n+1}^\ell;H)}+\sigma_i^3\left\|\bar{u}_{n+1}\right\|^2_{L^2(t_{n+1},t_{n+1}^\ell;\mathbb{R}^m)}\\
&\qquad+\sigma_i^2\left\|\mathcal{C}_i^2 S^{n+1}(\bar{u}_{n+1},y_{n+1})(t_{n+1}^\ell)-y_i^2\right\|^2_{H}\Big]\\
&\quad-\frac{\sigma_i^1}{2}\left\|\mathcal{C}_i^1 S^{L-\ell+1}(\tilde{u}_{L-\ell+1},y_{L-\ell})-y_i^1\right\|^2_{L^2(t_{L-\ell+1},t_{L-\ell+1}^\ell;H)}-\frac{\sigma_i^3}{2}\left\|\tilde{u}_{L-\ell+1}\right\|^2_{L^2(t_{L-\ell+1},t_{L-\ell+1}^\ell;\mathbb{R}^m)}\\
&\quad-\frac{\sigma_i^2}{2}\left\|\mathcal{C}_i^2 S^{L-\ell+1}(\tilde{u}_{L-\ell+1},y_{L-\ell+1})(t_{L-\ell+1}^\ell)-y_i^2\right\|^2_{L^2(t_{L-\ell+1},t_{L-\ell+1}^\ell;H)}.
\end{aligned}$$

The first two sums on the right-hand side are a telescopic sum. Together with the nonnegativity of the norms, this finally implies

$$\begin{aligned}
&\frac{\sigma_i^1}{2}\left\|\mathcal{C}_i^1 S^{(0,t_{\mathsf{end}})}(\mu,y_\circ)-y_i^1\right\|^2_{L^2(0,t_{\mathsf{end}};H)}+\frac{\sigma_i^3}{2}\left\|\mu\right\|^2_{L^2(0,t_{\mathsf{end}};\mathbb{R}^m)}\\
&\leq \frac{\sigma_i^1}{2}\left\|\mathcal{C}_i^1 S^0(\bar{u}_0,y_0)-y_i^1\right\|^2_{L^2(t_0,t_0^\ell;H)}+\frac{\sigma_i^3}{2}\left\|\bar{u}_0\right\|^2_{L^2(t_0,t_0^\ell;\mathbb{R}^m)}+\frac{\sigma_i^2}{2}\left\|\mathcal{C}_i^2 S^0(\bar{u}_0,y_0)(t_0^\ell)-y_i^2\right\|^2_{H}\\
&=\hat{J}_i^0(\bar{u}_0,y_\circ),
\end{aligned}$$

which is what we wanted to show. □

Remark 1. 1) *The statement from Theorem 3 is important, since it gives us performance bounds on the cost functions values of the MOMPC feedback control μ already after choosing the initial control $\bar{u}_0\in\mathcal{U}_{opt}^0(y_\circ)$, but before performing the MOMPC Algorithm 1. Thus, one strategy is to compute the entire Pareto set $\mathcal{U}_{opt}^0(y_\circ)$, which is computationally cheap due to the small time horizon (t_0,t_0^ℓ), and then to choose the initial control $\bar{u}_0\in\mathcal{U}_{opt}^0(y_\circ)$ according to the desired upper bounds on the cost functions.*

2) *Theorem 3 holds for arbitrary $T>0$. In particular, by taking the limit $T\to\infty$, the result can also be shown for the infinite-horizon case.*

Now we want to present a gradient descent scheme for computing a control $\bar{u}_n\in\mathcal{U}_{opt}^n(y_{\mathsf{a}})$ fulfilling (5). To this end we use the following result which is taken from e.g. [25, Theorem 2.1], where it is proved for the case of finite-dimensional controls. However, the extension to our infinite-dimensional

setting $U^n = L^2(t_n, t_n^\ell; \mathbb{R}^m)$ is straight-forward, since the proof of Theorem 2.1 in [25] does not explicitly use the finite-dimensionality of the space U^n, but only its property of being a Hilbert space.

Theorem 4. *Let $n \in \{0, \dots, L-\ell\}$ and $\sigma_i^3 > 0$ for $i = 1, 2$. Suppose that $y_\mathsf{a} \in H$, $u \in U^n$ are arbitrarily given and*

$$\hat{\alpha}^n = \operatorname*{arg\,min}_{\alpha=(\alpha_1,\alpha_2)\in\mathbb{R}^2} \left\{ \left\| \alpha_1 \nabla_u \hat{J}_1^n(u, y_\mathsf{a}) \right\|_U^2 + \left\| \alpha_2 \nabla_u \hat{J}_2^n(u, y_\mathsf{a}) \right\|_U^2 \,\middle|\, \alpha_1, \alpha_2 \geq 0 \text{ and } \alpha_1 + \alpha_2 = 1 \right\}. \tag{13}$$

Then either

$$q^n(u) = -\hat{\alpha}_1^n \nabla_u \hat{J}_1^n(u, y_\mathsf{a}) - \hat{\alpha}_2^n \nabla_u \hat{J}_2^n(u, y_\mathsf{a})$$

is a descent direction for the cost functions $\hat{J}_1^n(\cdot, y_\mathsf{a})$ and $\hat{J}_2^n(\cdot, y_\mathsf{a})$, or $q^n(u) = 0$ holds. Since the cost functions $\hat{J}_1^n(\cdot, y_\mathsf{a})$, $\hat{J}_2^n(\cdot, y_\mathsf{a})$ are strictly convex, the latter implies that u is a Pareto optimal point, c.f. [12].

Remark 2. *Problem* (13) *is a quadratic optimization problem in the variable $\alpha \in \mathbb{R}^2$ with the constraints $\alpha \in [0,1]^2$ and $\alpha_1 + \alpha_2 = 1$. By substituting $\alpha_2 = 1 - \alpha_1$, this problem can be reformulated as a box-constrained quadratic optimization problem in only one variable, which is easy to solve. In our numerical experiments we utilize the* MATLAB *routine* `quadprog`.

With Theorem 4 we can set up Algorithm 2, which takes an arbitrary point $u \in U^n$ as an input and results in a Pareto optimal point $\bar{u}$, compare [26, Algorithm 1]. From [26, Lemma 4] it follows that

Algorithm 2 (Multiobjective Gradient Descent Method)

Given: Current iterate n, initial control $u \in U^n$, tolerance $\varepsilon > 0$, Armijo parameter $0 < \beta \ll 1$.

1: Set $k = 0$ and $y_\mathsf{a} = y_n(t_n)$.
2: Get $\hat{\alpha}^n$ by solving (13) and set $q^n(u) = -\sum_{i=1}^2 \hat{\alpha}_i^n \nabla_u \hat{J}_i^n(u, y_\mathsf{a})$.
3: **while** $\|q^n(u)\|_{U^n} > \varepsilon$ **do**
4: Set $k = k + 1$.
5: Armijo backtracking: determine the step size $t_k > 0$ as the maximal $t \in \{(1/2)^l \mid l \in \mathbb{N}_0\}$ with

$$\hat{J}_i^n\left(u + t q^n(u), y_\mathsf{a}\right) \leq \hat{J}_i^n(u, y_\mathsf{a}) + \beta t \nabla_u \hat{J}_i^n(u, y_\mathsf{a})^\top q^n(u), \quad i = 1, 2.$$

6: Set $u^+ = u + t_k q^n(u)$ and then $u = u^+$.
7: Get $\hat{\alpha}^n$ by solving (13) and set $q^n(u) = -\sum_{i=1}^2 \hat{\alpha}_i^n \nabla_u \hat{J}_i^n(u, y_\mathsf{a})$.
8: **end while**
9: Set $\bar{u}_n = u$.

a step length $t_k > 0$ can always be found in step 5 of Algorithm 2. Moreover, in [26, Theorem 1] the authors proved the following convergence result. Again, they assumed a finite-dimensional space U^n, but the arguments of the proof transfer directly to our setting $U^n = L^2(t_n, t_n^\ell; \mathbb{R}^m)$.

Theorem 5. *Every accumulation point of the sequence $(u_k)_{k\in\mathbb{N}}$ produced by Algorithm 2 is a Pareto optimal point.*

Remark 3. 1) *Note that we cannot prove that the sequence $(u_k)_{k\in\mathbb{N}}$ has an accumulation point in the infinite-dimensional case. However, we will not encounter this problem in our numerical implementation, since the space U^n will be discretized. Therefore, Algorithm 2 will in practice terminate in a finite number of steps.*

2) *By construction of the algorithm it holds $\hat{J}_i^n(\bar{u}_n, y_\mathsf{a}) \leq \hat{J}_i^n(u, y_\mathsf{a})$, $i = 1, 2$, for any initial control $u \in U^n$, so that* (5) *is fulfilled, if we choose $u = \tilde{u}$.*

Finally, the MOMPC algorithm looks as follows.

Algorithm 3 (Multiobjective Model Predictive Control)

Given: Initial state $y_\circ \in H$.
for $n = 0$ **to** L_ℓ **do**
  Set $y_{\mathsf{a}} = y_n(t_n)$.
  if $n == 0$ **then**
    Choose an arbitrary initial control $u \in \mathcal{U}^0_{opt}(y_\circ)$. Go to step 7.
  end if
  Use Algorithm 2 to compute a Pareto optimal control $\bar{u}_n \in \mathcal{U}^n_{opt}(y_{\mathsf{a}})$ fulfilling (5).
  Set $y_{n+1}(t) = y_n(t)$ for all $t \in [0, t_n]$ and compute $y_{n+1}(t) = S^n(\bar{u}_n, y_{\mathsf{a}})(t)$ for all $t \in (t_n, t_{n+1}]$.
  Set $\mu(t) = \bar{u}_n(t)$ for all $t \in (t_n, t_{n+1}]$.
  Compute a feedback κ by solving (8) and set $\tilde{u}$ according to (6).
end for

4. Numerical Tests

Throughout the numerical experiments let the domain Ω be given by $\Omega = (0,1)^2$ with points $\boldsymbol{x} = (x_1, x_2)$. This domain is discretized by using linear finite elements with 494 degrees of freedom. We choose $t_{\mathsf{end}} = 1$ and $\Delta t = 1/198$. The number of total time steps L is set to $L = 197 + \ell$ in dependence of the smaller time horizons ℓ of the MPC open-loop problems. Accordingly, the final time T is given by $T = L\Delta t$. For the time integration of the PDE we use the Crank-Nicolson method. In Algorithm 2 we set the tolerance $\varepsilon = 10^{-4}$. In Algorithm 3 the initial control $\bar{u}_0$ is chosen as an element of the Pareto set $\mathcal{U}^0_{opt}(y_\circ)$. Therefore, in the following we will always compute a finite approximation $\mathcal{U}^0_{opt,appr}(y_\circ)$ of this set, which depends on the MPC horizon length ℓ, by the Euclidean reference point method.

We note that in all our examples we use fixed temporal and spatial discretizations. Clearly, this is not necessarily efficient and one may rather prefer to use adaptive discretizations that only use fine grids in space and/or time when this is really relevant for obtaining a sufficiently accurate numerical approximation for the control function. For single-objective PDE governed MPC problems, such methods were developed in, e.g., [27,28]. We conjecture that they could be adapted to the multiobjective setting.

4.1. Example 1

Let the advection field be given by

$$c(t,\boldsymbol{x}) = \begin{cases} (1,1)^\top, & \text{if } t \le 0.5, \\ \sqrt{2}\,(\cos(2\pi(t-0.5)+\pi/4), \sin(2\pi(t-0.5)+\pi/4))^\top, & \text{if } t \in (0.5,1), \\ (-1,-1)^\top, & \text{if } t \in [1,T], \end{cases} \tag{14}$$

i.e., it is space-independent and at first constantly pointing into the direction $(1,1)^\top$ before it then smoothly rotates towards the direction $(-1,-1)^\top$. The diffusion coefficient is set to $d = 1$, the reaction term is chosen as $r = 0$, i.e., there is no reaction in the system, and the control input $u \in L^2(0,1;\mathbb{R}^4)$ acts on all four edges of the domain individually.
Furthermore, we choose $y_\circ(\boldsymbol{x}) = 15$,

$$y_1^1(\boldsymbol{x},t) := \begin{cases} 15 + x_1x_2t, & \text{if } t \in [0,1], \\ 15 + x_1x_2, & \text{if } t \in [1,T], \end{cases}$$

$y_1^2(x) := 15 + x_1x_2$ and set $\sigma_1^1 = 1$, $\sigma_1^2 = 0.1$, $\sigma_1^3 = 0.001$ in the cost function $\hat{J}_1$ and $\sigma_2^1 = 0$, $\sigma_1^2 = 0$, $\sigma_1^3 = 1$ in the cost function $\hat{J}_2$. The linear mappings $\mathcal{C}_1^1 \in \mathcal{L}(L^2(0,T;H))$ and $\mathcal{C}_1^2 \in \mathcal{L}(H)$ are chosen to be the identity, respectively. This is a quite typical framework for MOCPs: the first cost function penalizes the deviation from a desired state, whereas the second cost function measures the control

costs.
Given these data we can apply Theorem 1 and show the external stability of the sets $\hat{J}^n(U^n, y_n(t_n))$, which implies the feasibility of the steps 4 and 6 in Algorithm 3.
However, we can in general not expect that there is a feedback κ fulfilling (7). The reason for this is that such a κ would have to fulfill the inequality

$$\frac{1}{2}\|\kappa\|^2_{L^2(t^\ell_{n-1},t^\ell_n;\mathbb{R}^m)} \leq 0,$$

due to the choices $\sigma_2^1 = \sigma_2^2 = 0$ in the second cost function, which implies $\kappa = 0$. Plugging this into (7) for $i = 1$ we see that the inequality is only fulfilled, if the uncontrolled system would move towards the desired temperatures. This is unlikely to happen in our setting, since the desired temperatures are larger than the initial condition and increasing in time. Nevertheless we can still compute a minimizer of (8) and use it to define $\tilde{u}$ in (6). Note that in this case, the assumptions of Theorem 3 are not fulfilled, so that this choice of κ and $\tilde{u}$ is of heuristic nature.
Another heuristic approach for determining $\tilde{u}$ that we want to test in the following is motivated by the criteria used in step 2 of Algorithm 4 in [2], which was designed for problems without terminal condition and translates to our problem as follows: During the n-th loop iteration of Algorithm 3 ($n \in \{0, \dots, L-\ell\}$) compute $\kappa \in \mathcal{U}_{opt}^{(t^\ell_{n-1},t^\ell_{n+1})}(y_n(t^\ell_{n-1}))$ in step 9 such that

$$\|\kappa\|^2_{L^2(t^\ell_{n-1},t^\ell_n;\mathbb{R}^4)} \leq \|\bar{u}_n\|^2_{L^2(t^\ell_{n-1},t^\ell_n;\mathbb{R}^4)} \tag{15}$$

holds. Then we set

$$\tilde{u}(t) = \begin{cases} \bar{u}_n(t), & \text{if } t \in [t_{n+1}, t^\ell_{n-1}] = [t_{n+1}, t_{n-1+\ell}], \\ \kappa(t), & \text{if } t \in (t^\ell_{n-1}, t^\ell_{n+1}] = (t_{n-1+\ell}, t_{n+1+\ell}]. \end{cases} \tag{16}$$

Here we do not impose the inequality (15) on κ explicitly. The reason for this is that demanding (15) would not guarantee us any performance results for our framework, but only increase the computational time. Therefore, we use again the gradient descent method presented in Algorithm 2 for computing a control $\kappa \in \mathcal{U}_{opt}^{(t^\ell_{n-1},t^\ell_{n+1})}$. As initial control u we choose

$$u(t) = \begin{cases} \bar{u}_n(t), & \text{if } t \in (t^\ell_{n-1}, t^\ell_n], \\ \bar{u}_n(t-\Delta t), & \text{if } t \in (t^\ell_n, t^\ell_{n+1}]. \end{cases}$$

Again, it can be shown that for any accumulation point $\bar{u}$ of the sequence $(u_k)_{k\in\mathbb{N}}$ produced by Algorithm 2, it holds $\bar{u} \in \mathcal{U}_{opt}^{(t^\ell_{n-1},t^\ell_{n+1})}(y_n(t^\ell_{n-1}))$. Moreover, although the inequality (15) is not guaranteed directly by this method, we still get $\hat{J}^{(t^\ell_{n-1},t^\ell_{n+1})}(\kappa, y_n(t^\ell_{n-1})) \leq \hat{J}^{(t^\ell_{n-1},t^\ell_{n+1})}(u, y_n(t^\ell_{n-1}))$. In particular, this implies

$$\|\kappa\|^2_{L^2(t^\ell_{n-1},t^\ell_{n+1};\mathbb{R}^4)} \leq \|u\|^2_{L^2(t^\ell_{n-1},t^\ell_n;\mathbb{R}^4)} = 2\,\|\bar{u}_n\|^2_{L^2(t^\ell_{n-1},t^\ell_n;\mathbb{R}^4)}, \tag{17}$$

since the time grid is equidistant. Therefore, it can be expected that (15) is satisfied in most cases.

Test 1. In our first test the main focus is on investigating how well the MOMPC Algorithm 3 performs compared to the open-loop problem on the time intervall $[0, 1]$ for different MPC horizons. Therefore, we perform Algorithm 3 with all controls from $\mathcal{U}^0_{opt,appr}(y_\circ)$ for the MPC horizon lengths of $\ell = 25$, $\ell = 50$ and $\ell = 99$ with two different strategies in step 9:

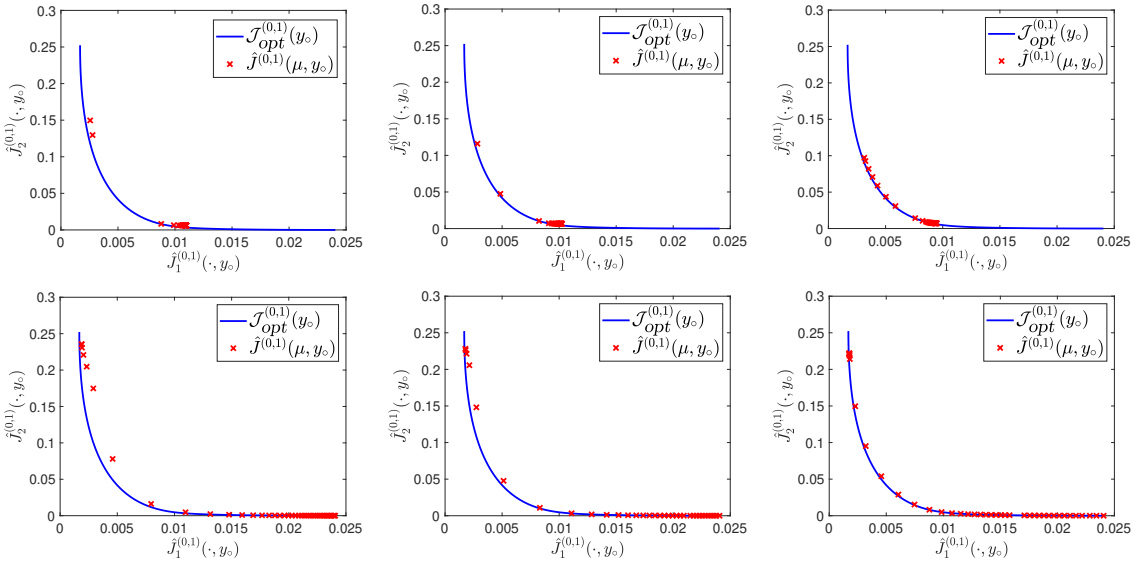
1. Compute κ by minimizing (8) and set $\tilde{u}$ as in (6).
2. Compute κ by using Algorithm 2 as described above and set $\tilde{u}$ according to (16).

Note that the set $\mathcal{U}^0_{opt,appr}(y_\circ)$ contains 42, 34 and 40 elements for the horizon lengths $\ell = 25$, $\ell = 50$ and $\ell = 99$, respectively.

The obtained results when using strategy 1 for computing κ are shown in the top row of Figure 1 (a)-(c). It is clearly visible that an increasing MPC horizon length has the positive effect that the MOMPC points produced by Algorithm 3 are located closer to the Pareto front $\mathcal{J}^{(0,1)}_{opt}(y_\circ)$. This can be explained by the fact that a larger MPC horizon allows for a better prediction of the future behavior of the system dynamics. However, we see a clustering of MOMPC points in the middle of the Pareto front, which only slightly improves, if the MPC horizon length is increased. So it is not possible to obtain the whole extent of the Pareto front by varying the initial control $\bar{u}_0 \in \mathcal{U}^0_{opt,appr}(y_\circ)$.

On the other hand, we can see the results of strategy 2 in the bottom row of Figure 1 (a)-(c). Let us first note that the inequality (15) is fulfilled in most steps of Algorithm 3 for all initial controls $\bar{u}_0 \in \mathcal{U}^0_{opt,appr}(y_\circ)$. Only for some initial controls $\bar{u}_0$, for which $\hat{J}^0(\bar{u}_0, y_\circ)$ is located on the top left part of the Pareto front $\mathcal{J}^0_{opt}(y_\circ)$, i.e., for initial controls corresponding to the part of the Pareto front, where the main goal is to minimize $\hat{J}^0_1(\cdot, y_\circ)$ almost regardless of the function values of $\hat{J}^0_2(\cdot, y_\circ)$, the condition (15) is not fulfilled in some steps of the MOMPC algorithm. The reason for this is the following: When using Algorithm 2 for computing a feedback $\kappa \in \mathcal{U}^{(t^\ell_{n-1}, t^\ell_{n+1})}_{opt}$, the resulting minimizer $\hat{\alpha}$ of (13) displays the weight of the cost functions $\hat{J}^{(t^\ell_{n-1}, t^\ell_{n+1})}(\cdot, y_n(t^\ell_{n-1}))$ at the Pareto optimal point $\kappa \in \mathcal{U}^{(t^\ell_{n-1}, t^\ell_{n+1})}_{opt}(y_n(t^\ell_{n-1}))$. If we start with a point $\bar{u}_0$, for which $\hat{J}^0(\bar{u}_0, y_\circ)$ is located on the top left part of the Pareto front $\mathcal{J}^0_{opt}(y_\circ)$, the weighting of the cost functions will be $\hat{\alpha} \approx (1, 0)^\top$. Therefore, the descent direction $q^n(u)$ from Algorithm 2 will mostly point in the direction of the negative gradient of the first cost function, which might lead to (15) not being satisfied.

Looking at the results, we observe again that a larger MPC horizon length leads to a better result in the sense that the MOMPC points are closer to the Pareto front. In contrast to strategy 1, we do not see a clustering of MOMPC points in the middle of the Pareto front, but rather in the lower right part of the Pareto front. This clustering improves, if the MPC horizon length is increased, so that the entire scale of the Pareto front is obtained for a horizon length of $\ell = 99$.



(a) $\ell = 25$ **(b)** $\ell = 50$ **(c)** $\ell = 99$

Figure 1. Results of Algorithm 3 for all initial controls $\bar{u}_0 \in \mathcal{U}^0_{opt,appr}(y_\circ)$ for different horizon lengths. Top: Strategy 1 for computing κ. Bottom: Strategy 2 for computing κ.

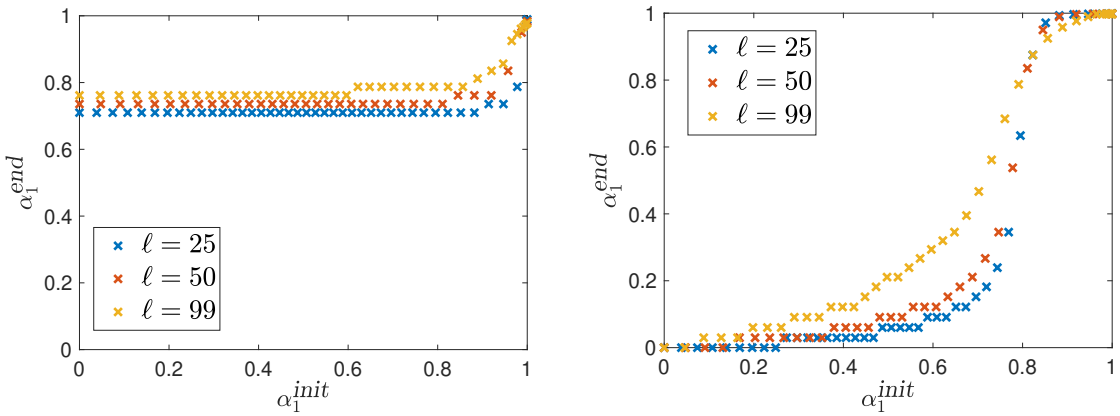
The difference in the clustering behavior of strategy 1 compared to strategy 2 can be seen by looking at the following relation: Every initial control vector $\bar{u}_0 \in \mathcal{U}^0_{opt,appr}(y_\circ)$ is the solution to a weighted-sum problem

$$\min_{u \in U^0} \alpha_1^{init} \hat{J}_1^0(u, y_\circ) + \alpha_2^{init} \hat{J}_2^0(u, y_\circ)$$

for some weight vector $\alpha^{init} \in \mathbb{R}^2_{\geq 0}$ with $\alpha_1^{init} + \alpha_2^{init} = 1$; see, e.g., [12]. This weight vector can be determined easily when using the Euclidean reference point method to compute $\mathcal{U}^0_{opt,appr}(y_\circ)$; cf. [19, Lemma 5]. After executing Algorithm 3 we can check to which Pareto optimal point $\bar{y} = \hat{J}^{(0,1)}(\bar{u}, y_\circ) \in \mathcal{J}^{(0,1)}_{opt}(y_\circ)$ the MOMPC point $\hat{J}^{(0,1)}(\mu, y_\circ)$ has the smallest distance. Again, $\bar{u}$ is the solution to a weighted-sum problem

$$\min_{u \in U^{(0,1)}} \alpha_1^{end} \hat{J}_1^{(0,1)}(u, y_\circ) + \alpha_2^{end} \hat{J}_2^{(0,1)}(u, y_\circ)$$

for a weight vector $\alpha^{end} \in \mathbb{R}^2_{\geq 0}$ with $\alpha_1^{end} + \alpha_2^{end} = 1$, which can again be determined. The mapping $\alpha_1^{init} \mapsto \alpha_1^{end}$ can be seen in Figure 2 for both strategies. Since the mapping displays the first weight



(a) Strategy 1 **(b)** Strategy 2

Figure 2. The mapping $\alpha_1^{init} \mapsto \alpha_1^{end}$ for different MPC horizon lengths.

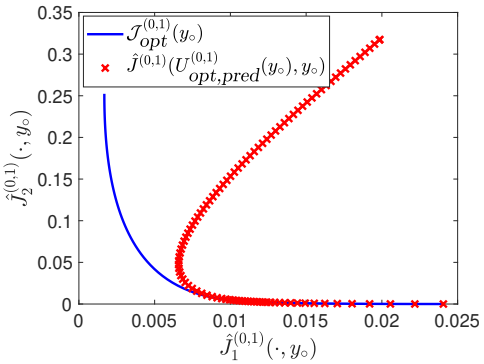
vector, a small value in the plot corresponds to a small weighting of the first cost function (the tracking term) and a large weighting of the second cost function (the control costs), and vice versa.
Note that the 'ideal' result of this mapping would be the identity, since this would imply that the weight α^{init} of the initial control $\bar{u}_0$ remains constant throughout the MOMPC algorithm.
From this plot the clustering of the MOMPC points for both strategies can be deduced. For strategy 1 it can be seen that the clustering of the points happen at a weight of $\alpha_1^{end} \in [0.7, 0.8]$ for all MPC horizon lengths. Up to an initial weight of $\alpha_1^{init} \approx 0.8$ all the initial controls lead more or less to the same result of the MOMPC algorithm. Only for initial controls with an initial weight of $\alpha_1^{init} > 0.8$, we observe that the upper part of the Pareto front can be reached.
For strategy 2 the clustering in the lower part of the Pareto front for MPC horizon lengths of $\ell = 25$ and $\ell = 50$ can be deduced, since up to an initial weight of $\alpha_1^{init} \approx 0.8$, the value of α_1^{end} stays below a value of 0.1, which means that the resulting MOMPC point is in a region of the Pareto front, where the cost function $\hat{J}_2^{(0,1)}(\cdot, y_\circ)$ is weighted much more in comparison to the first cost function $\hat{J}_1^{(0,1)}(\cdot, y_\circ)$. Moreover, for these two MPC horizons, we can see a clear cut-off behavior at a value of 0.8. If α_1^{init} is larger than 0.8, the value of α_1^{end} is larger than 0.9. For an MPC horizon length of $\ell = 99$, we do not observe such a clear cut-off behavior. Although the plot is still far from being the identity, it is visible that one can control the outcome of the MOMPC algorithm more precisely by varying the initial control.

In conclusion we can say that for strategy 2 a larger MPC horizon leads to a better distribution of the MOMPC points on the Pareto front, and that it is possible to approximate the full scale of the Pareto front by choosing different initial controls $u_0 \in U^0_{opt}(y_\circ)$. Strategy 1, however, seems not to be well-suited for this problem framework, since it is not possible to reach to whole extent of the Pareto front by choosing different initial controls. Quite on the contrary, the MOMPC algorithm steers almost all initial controls to the same region of the Pareto front, independently of the MPC horizon length.

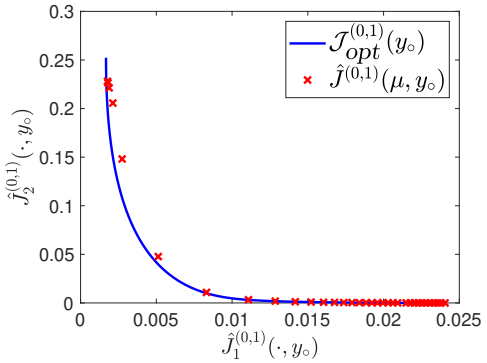
Test 2. Now we want to show why the use of MPC in the multiobjective context is needed. For this we consider the following setup: Imagine that we want to compute the Pareto front $\mathcal{J}^{(0,1)}_{opt}(y_\circ)$. However, there is only a prediction of the advection field available, which is given by

$$c^{pred}(t,x) = (1,1)^\top \quad \text{for all } (t,x) \in [0,T] \times \Omega.$$

In the end it turns out that the prediction of the advection field is not very accurate, and the true advection is given by the function c in (14), i.e., it deviates from the prediction c^{pred} after $t = 0.5$.



(a) Pareto front $\mathcal{J}^{(0,1)}_{opt}$ and the function values $\hat{J}^{(0,1)}(U^{(0,1)}_{opt,pred}(y_\circ), y_\circ)$



(b) Results obtained from Algorithm 3 for $\ell = 50$

Figure 3. Comparison of the predictive multiobjective optimal control and Algorithm 3.

Denote by $U^{(0,1)}_{opt,pred}(y_\circ)$ the Pareto set of the problem (MOCP) using the predicted advection c^{pred}. Again we used the Euclidean reference point method to compute an approximation both of $U^{(0,1)}_{opt}(y_\circ)$ and of $U^{(0,1)}_{opt,pred}(y_\circ)$. Figure 3 (a) displays the Pareto front $\mathcal{J}^{(0,1)}_{opt}(y_\circ)$ together with the function values $\hat{J}^{(0,1)}(U^{(0,1)}_{opt,pred}(y_\circ), y_\circ)$. One can clearly see that the controls in $U^{(0,1)}_{opt,pred}(y_\circ)$ are far from being Pareto optimal for the problem with advection c, especially in the upper part of the Pareto front. This can be explained by the fact that the changing advection totally changes the strategy that has to be used to come close to the desired temperature, which is not captured by the open-loop problem using the prediction c^{pred}.

We compare this to the results obtained by using the MOMPC Algorithm 3 together with strategy 2 from the first test, where we assumed that the true advection is known to the open-loop problems in the MPC algorithm. This is reasonable since the MPC horizons are reasonably small to allow a precise prediction of the future behavior of the advection field. Just as an example we look at an MPC horizon length of $\ell = 50$ in Figure 3 (b) and see that in contrast to the predicted Pareto optimal controls $U^{(0,1)}_{opt,pred}(y_\circ)$ the MOMPC feedback control μ comes quite close to the true Pareto front. This underlines that the use of MOMPC is necessary in situations, in which there are only predictions of data available.

222 ### 4.2. Example 2

The parameter choices of the previous experiment imply that the choice of a feedback κ fulfilling (7) is not possible. In this section we want to present numerical results for a setup that allows such a feedback in principle.
To this end we choose the following parameter values: The diffusion coefficient is set to $d = 0.5$, the reaction coefficient to $r = 0.5$, and the advection field is chosen as

$$c(t, \boldsymbol{x}) = \begin{cases} (1,0)^\top, & \text{if } x_2 \leq 0.5, \\ (-1,0)^\top, & \text{if } x_2 \geq 0.5. \end{cases}$$

As initial condition we choose

$$y_\circ(\boldsymbol{x}) = -3 \cdot (1 + \tanh((x_1 - 0.5)/0.2)) + 3,$$

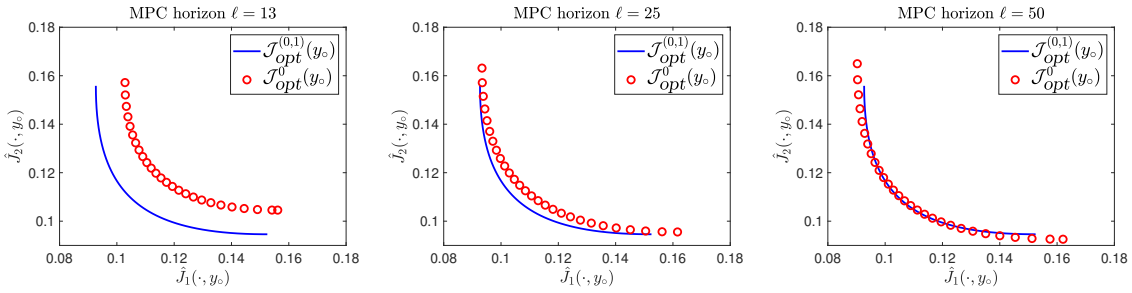
which is a smooth approximation of the discontinuous function

$$\boldsymbol{x} \mapsto \begin{cases} 3, & \text{if } x_1 \leq 0.5, \\ -3, & \text{if } x_1 \geq 0.5. \end{cases}$$

It can be shown that $y = 0$ is a stable steady state of the PDE (1) to the control $u = 0$. Therefore, we choose the desired temperatures $y_1^1 = y_2^1 = 0 \in L^2(0,1;H)$ and $y_1^2 = y_2^2 = 0 \in H$ together with the parameter values $\sigma_1^1 = \sigma_2^1 = 1$, $\sigma_1^2 = \sigma_2^2 = 0.1$ and $\sigma_1^3 = \sigma_2^3 = 0.1$. The linear functionals in the cost functions are given by

$$\begin{aligned}
&\mathcal{C}_1^1 : L^2(0,1;H) \to L^2(0,1;H), && f \mapsto f \cdot \left(\chi_{\{(t,\boldsymbol{x})|x_1 \leq 0.5\}} + 0.01\chi_{\{(t,\boldsymbol{x})|x_1 \geq 0.5\}}\right), \\
&\mathcal{C}_2^1 : L^2(0,1;H) \to L^2(0,1;H), && f \mapsto f \cdot \left(0.01\chi_{\{(t,\boldsymbol{x})|x_1 \leq 0.5\}} + \chi_{\{(t,\boldsymbol{x})|x_1 \geq 0.5\}}\right), \\
&\mathcal{C}_1^2 : H \to H, && f \mapsto f \cdot \left(\chi_{\{\boldsymbol{x}|x_1 \leq 0.5\}} + 0.01\chi_{\{\boldsymbol{x}|x_1 \geq 0.5\}}\right), \\
&\mathcal{C}_2^2 : H \to H, && f \mapsto f \cdot \left(0.01\chi_{\{\boldsymbol{x}|x_1 \leq 0.5\}} + \chi_{\{\boldsymbol{x}|x_1 \geq 0.5\}}\right).
\end{aligned}$$

223 Thus, both cost functions measure the deviation of the state from the steady state $y = 0$ together with
224 some control costs. To make the cost functions conflicting, the linear functionals are chosen such that
225 the first cost function penalizes deviations from the steady state mostly in the left half of the domain,
226 whereas the second cost function penalizes the deviation in the right half of the domain.
227 If we run Algorithm 3 with this parameter setting, we still observe that in many iterations there is no
228 feedback κ fulfilling (7), especially for the larger MPC horizons of $\ell = 25$ and $\ell = 50$. The reason for
229 this is that the initial Pareto front $\mathcal{J}_{opt}^0(y_\circ)$ is quite close or even dominates the Pareto front $\mathcal{J}_{opt}^{(0,1)}(y_\circ)$ in some points for a MPC horizon of $\ell = 25$ or $\ell = 50$, see Figure 4. Since the MOMPC points

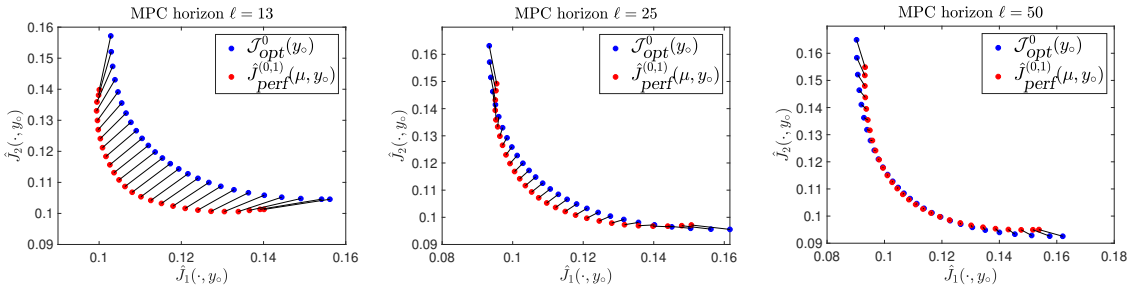


(a) $\ell = 13$ **(b)** $\ell = 25$ **(c)** $\ell = 50$

Figure 4. Comparison between $\mathcal{J}_{opt}^{(0,1)}(y_\circ)$ and $\mathcal{J}_{opt}^0(y_\circ)$ for different MPC horizon lengths.

230

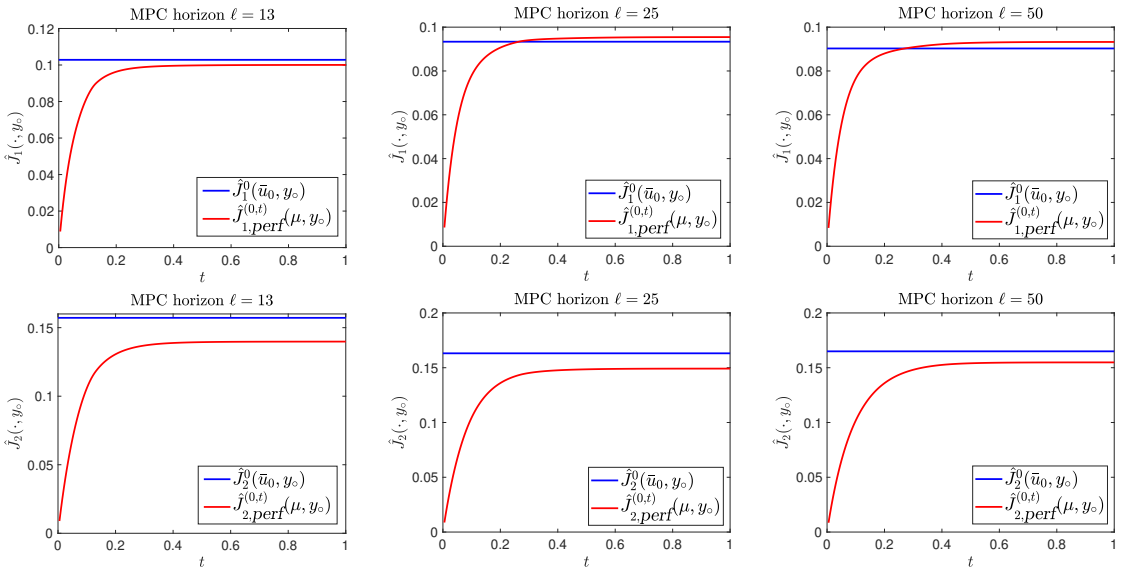
cannot perform better than any point on the Pareto front $\mathcal{J}_{opt}^{(0,1)}(y_\circ)$, it is not possible to achieve the performance result from Theorem 3 in those parts. As all the other assumptions of Theorem 3 are satisfied, the reason for not achieving the performance result has to be that it is not possible to find a feedback κ fulfilling (7) in all steps of the algorithm. This is what we observe numerically.
However, if we look at Figure 5 we see that we still obtain the performance result from Theorem 3 for many points, although a feedback κ cannot be found for all these points. The black lines indicate which performance value $\hat{J}$ corresponds to which initial cost. Therefore, if the black lines are pointing to the top right (starting from the red point), this means that the performance result is satisfied for this point. We observe that the performance result holds for all points for an MPC horizon length of $\ell = 13$ and for most points for $\ell = 25$. Even for $\ell = 50$ there are some points in the middle of the Pareto front, for which the performance result holds, although there are only very few steps, where a feedback κ fulfilling (7) can be found. Figure 6 indicates that this performance behavior does not change drastically, if we increase the time horizon t_{end}. The reason for this is that at time $t_{\mathsf{end}} = 1$ the



(a) $\ell = 13$ **(b)** $\ell = 25$ **(c)** $\ell = 50$

Figure 5. Performance Results.

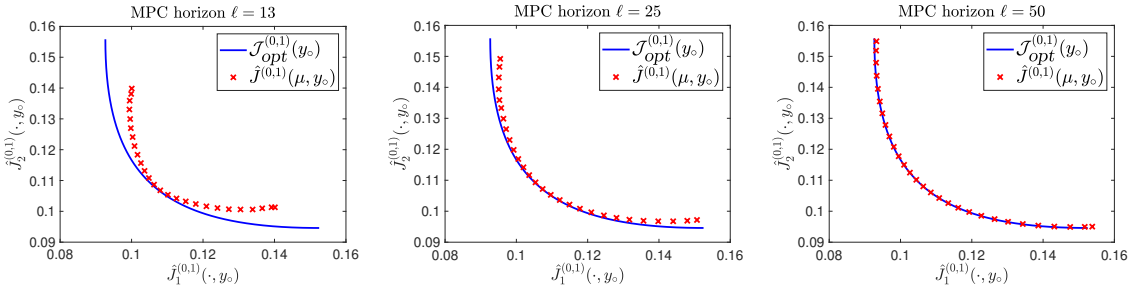
steady state $y = 0$ is almost reached, so that almost no further costs are needed after this point. If



(a) $\ell = 13$ **(b)** $\ell = 25$ **(c)** $\ell = 50$

Figure 6. Performance Results for the top left point of the Pareto front.

we compare the MOMPC results obtained from Algorithm 3 with the Pareto front $\mathcal{J}_{opt}^{(0,1)}(y_\circ)$ of the open-loop problem in Figure 7, we observe that increasing the MPC horizon has two positive effects on the results: Firstly, the MOMPC points are located closer to the Pareto front, and secondly, they are spread more evenly over the entire Pareto front. Already for an MPC horizon length of $\ell = 50$ we can see that the Pareto front is almost perfectly approximated.



(a) $\ell = 13$ **(b)** $\ell = 25$ **(c)** $\ell = 50$

Figure 7. Comparison the results from Algorithm 3 and the Pareto front.

5. Conclusion and outlook

In this paper we have adopted the framework of MOMPC presented in [1,2] for ODEs to parabolic advection-diffusion-reaction equations. A key ingredient for the MOMPC algorithm is a gradient-based descent scheme for multiobjective optimization taken from [25,26]. On the one hand, this scheme results in a Pareto optimal point, and on the other hand, it intrinsically implies that the crucial condition (5) is satisfied. Assuming the existence of a feedback κ satisfying (7) we could show that the performance result from [1, Theorem 11] also holds for our framework of the MOMPC algorithm. Numerically, it turns out that it is hard to guarantee that such a feedback exists. However, even in the case that such a feedback does not exist, the results of the MOMPC algorithm are comparable with the Pareto optimal points of the full-horizon open-loop problem. Especially when some parameters of the PDE can only be predicted, the MOMPC algorithm turns out to be a useful tool, if we still want to compute many compromises between the cost functions.

Both the analysis in Section 3 and the numerical experiments in Section 4 were only carried out for two cost functions, but it is possible to extend both to an arbitrary number of cost function as long as all cost functions are of the form (2). For the analysis this extension is straight-forward, since all the arguments can be transferred directly to this more general case. On the numerical side we can still use the Euclidean reference point method for computing the initial Pareto set $\mathcal{U}_{opt}^0(y_\circ)$, see e.g., [29]. In general, the effort for computing the Pareto set $\mathcal{U}_{opt}^0(y_\circ)$ scales exponentially with the number of cost functions. However, this computation has only to be performed once in step 4 of Algorithm 3, which is in many cases not underlying any time restrictions. While performing the rest of the MOMPC algorithm the number of cost functions only influences the complexity of steps 6 and 9. In step 6 we need to perform Algorithm 2, in which the dimension of the variable space of the minimization problem (13) is equal to the number of cost functions. Since we only have to compute one Pareto optimal control $\bar{u}_n \in \mathcal{U}_{opt}^n(y_a)$ this step is computationally not expensive. In step 9 the number of cost functions in the min-max problem 8 corresponds to the total number of cost functions. Again the effort for solving this problem does not cause complex calculations.

During the MOMPC Algorithm 3 numerous open-loop problems on the MPC horizon have to be solved. Since this involves the solution of the PDE and its associated adjoint equation, this is numerically expensive. Therefore, in a future work one can apply model-order reduction techniques (e.g., the proper orthogonal decomposition method, cf. [30,31]) for lowering the computational effort. Strategies presented in [16,17] could be used to show convergence to the full solution.

Acknowledgments: S. Banholzer gratefully acknowledges support by the German DFG-Priority Program 1962 and by the Landesgraduiertenförderung of Baden-Württemberg.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MO	Multiobjective optimization
MOP	Multiobjective optimization problem
MOCP	Multiobjective optimal control problem
MPC	Model predictive control
MOMPC	Multiobjective model predictive control
ODE	Ordinary differential equation
PDE	Partial differential equation

References

1. Grüne, L.; Stieler, M. Performance guarantees for multiobjective model predictive control. *IEEE 56th Annual Conference on Decision and Control (CDC)* **2017**, pp. 5545–5550.
2. Grüne, L.; Stieler, M. Multiobjective model predictive control for stabilizing cost criteria. *Discrete & Continuous Dynamical Systems-B* **2019**, *24*, 3905.
3. Rawlings, J.; Mayne, D. *Model Predictive Control: Theory and Design*; Nob Hill Publishing, 2009.
4. Forbes, M.G.; Patwardhan, R.S.; Hamadah, H.; Gopaluni, R.B. Model Predictive Control in Industry: Challenges and Opportunities. Proceedings of the 9th IFAC Symposium on Advanced Control of Chemical Processes — ADCHEM 2015; , 2015; Vol. 48, *IFAC-PapersOnLine*, pp. 531–538.
5. Grüne, L. Approximation properties of receding horizon optimal control. *Jahresber. DMV* **2016**, *118*, 3–37.
6. Grüne, L.; Pannek, J. *Nonlinear Model Predictive Control: Theory and Algorithms*, 2 ed.; Communications and Control Engineering, Springer, 2017.
7. Laabidi, K.; Bouani, F.; Ksouri, M. Multi-criteria optimization in nonlinear predictive control. *Mathematics and Computers in Simulation* **2008**, *76*, 363–374. doi:http://dx.doi.org/10.1016/j.matcom.2007.04.002.
8. Bemporad, A.; Muñoz de la Peña, D. Multiobjective model predictive control. *Automatica* **2009**, *45*, 2823–2830.
9. He, D.; Wang, L.; Sun, J. On stability of multiobjective NMPC with objective prioritization. *Automatica* **2015**, *57*, 189–198.
10. Zavala, V.M.; Flores-Tlacuahuac, A. Stability of multiobjective predictive control: A utopia-tracking approach. *Automatica* **2012**, *48*, 2627–2632.
11. Peitz, S.; Schäfer, K.; Ober-Blöbaum, S.; Eckstein, J.; Köhler, U.; Dellnitz, M. A Multiobjective MPC Approach for Autonomously Driven Electric Vehicles. *IFAC PapersOnLine* **2017**, *50*, 8674–8679.
12. Ehrgott, M. *Multicriteria Optimization*, 2 ed.; Springer Berlin Heidelberg New York, 2005.
13. Miettinen, K. *Nonlinear Multiobjective Optimization*; Springer US, 1998.
14. Iapichino, L.; Trenz, S.; Volkwein, S. Multiobjective optimal control of semilinear parabolic problems using POD. In *Numerical Mathematics and Advanced Applications (ENUMATH 2015)*; Karasözen, B.; Manguoglu, M.; Tezer-Sezgin, M.; Goktepe, S.; Ugur, Ö., Eds.; Springer, 2016; pp. 389–397.
15. Iapichino, L.; Ulbrich, S.; Volkwein, S. Multiobjective PDE-constrained optimization using the reduced-basis method. *Advances in Computational Mathematics* **2017**, *43*, 945–972.
16. Peitz, S. Exploiting Structure in Multiobjective Optimization and Optimal Control. PhD thesis, Paderborn University, 2017.
17. Peitz, S.; Dellnitz, M., Gradient-Based Multiobjective Optimization with Uncertainties. In *NEO 2016: Results of the Numerical and Evolutionary Optimization Workshop NEO 2016 and the NEO Cities 2016 Workshop held on September 20-24, 2016 in Tlalnepantla, Mexico*; Maldonado, Y.; Trujillo, L.; Schütze, O.; Riccardi, A.; Vasile, M., Eds.; Springer International Publishing, 2017; pp. 159–182. doi:10.1007/978-3-319-64063-1_7.
18. Banhozer, S.; Dellnitz, M.; Gebken, B.; Peitz, S.; Volkwein, S. ROM-based multiobjective optimization of elliptic PDEs via numerical continuation. Technical report, arXiv:1906.09075v1, 2019.
19. Wierzbicki, A.P. The Use of Reference Objectives in Multiobjective Optimization. Multiple Criteria Decision Making Theory and Application; Springer Berlin Heidelberg: Berlin, Heidelberg, 1980; pp. 468–486.
20. Wierzbicki, A.P. On the completeness and constructiveness of parametric characterizations to vector optimization problems. *Operations-Research-Spektrum* **1986**, *8*, 73–87.
21. Banholzer, S.; Beermann, D.; Volkwein, S. POD-Based Bicriterial Optimal Control by the Reference Point Method. *IFAC-PapersOnLine* **2016**, *49*, 210–215.

22. Banholzer, S. POD-Based Bicriterial Optimal Control of Convection-Diffusion Equations. Master's thesis, University of Konstanz, 2017.
23. Banholzer, S.; Beermann, D.; Volkwein, S. POD-Based error control for reduced-order bicriterial PDE-constrained optimization. *Annual Reviews in Control* **2017**, *44*, 226–237.
24. Dautray, R.; Lions, J.L. Mathematical Analysis and Numerical Methods for Science and Technology. Volume 5: Evolution Problems I **2000**.
25. Schäffler, S.; Schultz, R.; Weinzierl, K. Stochastic Method for the Solution of Unconstrained Vector Optimization Problems. *Journal of Optimization Theory and Applications* **2002**, *114*, 209–222.
26. Fliege, J.; Svaiter, B. Steepest descent methods for multicriteria optimization. *Mathematical Methods of Operations Research* **2000**, *51*, 479—-494.
27. Grüne, L.; Schaller, M.; Schiela, A. Sensitivity analysis of optimal control for a class of parabolic PDEs motivated by model predictive control. *SIAM J. Control Optim.* **2019**, *57*, 2753–2774.
28. Grüne, L.; Schaller, M.; Schiela, A. Exponential sensitivity analysis for Model Predictive Control of PDEs. Preprint, Department of Mathematics, University of Bayreuth, 2020. Available from http://nbn-resolving.org/urn:nbn:de:bvb:703-epub-4590-3.
29. Banholzer, S.; Volkwein, S. Hierarchical Convex Multiobjective Optimization by the Euclidean Reference Point Method. Technical report, 2019.
30. Sirovich, L. Turbulence and the dynamics of coherent structures part I: coherent structures. *Quarterly of Applied Mathematics* **1987**, *XLV*, 561–571.
31. Gubisch, M.; Volkwein, S., Proper Orthogonal Decomposition for Linear-Quadratic Optimal Control. In *Model Reduction and Approximation: Theory and Algorithms*; Benner, P.; Cohen, A.; Ohlberger, M.; Willcox, K., Eds.; SIAM, 2017; pp. 5–66. doi:10.1137/1.9781611974829.

Sample Availability: Samples of the compounds ..... are available from the authors.