# Combinatorial Optimal Control of Semilinear Elliptic PDEs

**Christoph Buchheim · Renke Kuhlmann · Christian Meyer**

November 2, 2017

**Abstract** Optimal control problems (OCP) containing both integrality and partial differential equation (PDE) constraints are very challenging in practice. The most wide-spread solution approach is to first discretize the problem, it results in huge and typically nonconvex mixed-integer optimization problems that can be solved to proven optimality only in very small dimensions. In this paper, we propose a novel outer approximation approach to efficiently solve such OCPs in the case of certain semilinear elliptic PDEs with static integer controls over arbitrary combinatorial structures, where we assume the nonlinear part of the PDE to be non-decreasing and convex. The basic idea is to decompose the OCP into an integer linear programming (ILP) master problem and a subproblem for calculating linear cutting planes. These cutting planes rely on the pointwise concavity or submodularity of the PDE solution operator in terms of the control variables. The decomposition allows us to use standard solution techniques for ILPs as well as for PDEs. We further benefit from reoptimization strategies due to the iterative structure of the algorithm. Experimental results show that the new approach is capable of solving the combinatorial OCP of a semilinear Poisson equation with up to 180 binary controls to global optimality within a five hour time limit. In the case of the screened Poisson equation, which yields semi-infinite integer linear programs, problems with even 1400 binary controls are solved.

Christoph Buchheim
Fakultät für Mathematik, TU Dortmund, Vogelpothsweg 87, 44227 Dortmund, Germany
E-mail: christoph.buchheim@math.tu-dortmund.de

Renke Kuhlmann
E-mail: renke.kuhlmann@math.tu-dortmund.de

Christian Meyer
E-mail: christian.meyer@math.tu-dortmund.de

## 1 Introduction

Optimal control is the optimization of a system described by partial or ordinary differential equations (PDEs/ODEs) over a control input. In a broad range of applications, all or some of the control variables have to be considered discrete, e.g. motor- or gear-switches in automotive engineering [7,27], state transitions or feed locations in chemical engineering [2,4] or – in case of PDEs – placements of wind turbines in a wind park [39] or switches for valves or compressors in gas or water networks [16,18]. Consequently, the demand for efficient algorithms to address optimal control problems with (partly) discrete controls, often referred to as mixed-integer optimal control problems (MIOCP), mixed-integer dynamic optimization (MIDO), or hybrid optimal control problems (HOCP), is very high. Most approaches discussed in the literature consider applications where the discrete variables are dynamic, i.e. depend on time or space, but the number of such variables usually remains very limited. Moreover, complicating combinatorial constraints can often not be handled satisfactorily, since the solution approaches are heuristic only (e.g. rounding strategies). In this paper, we address a different class of applications: we assume that the discrete controls are static but many, and subject to combinatorial constraints that may render the problem hard even in the absence of differential equations.

The most straightforward and widely used approach to address MIOCPs is to *first-discretize-then-optimize*. The basic idea is to discretize the control and, if desired, the state of the dynamic process in time or space, in order to approximate the MIOCP by a finite-dimensional mixed-integer nonlinear and typically nonconvex programming problem (MINLP) and then use standard techniques for solving the latter; see [6] for a recent survey on algorithms for MINLP. Although specific MIOCPs have been successfully solved to optimality by direct methods [19,38,2,3,13,31], the discretization approach often fails if applied to more general problem classes [36].

As a consequence, various numerical methods have been developed to quickly compute feasible, but suboptimal solutions. The most prominent heuristic is the Sum-Up Rounding strategy [26,33], which can also be applied to time-dependent controls in MIOCPs with PDE constraints [22]. It is capable of finding a feasible mixed-integer solution constructed out of an integrality-relaxed NLP solution, the latter being obtained by a direct method applied to a convexified MIOCP, so that the relaxed and rounded states are arbitrarily close (depending on the OCP discretization). However, combinatorial constraints may still be violated. To minimize the integrality error, the Combinatorial Integral Approximation (with a focus on restrictions on the number of switches) has been proposed [34], where the relaxed control is tracked by an integer control. It leads to a quickly solvable mixed-integer linear programming problem (MILP) and may serve as a first upper bound for other MINLP solution methods. However, both Sum-Up Rounding and Combinatorial Integral Approximation are designed to address time-dependent discrete controls.

The proposed convexification approach cannot handle problems with a large or even exponential number of integer feasible solutions, which constitute

the main challenge in combinatorial optimization. In particular, if the MIOCP contains a combinatorial substructure on a significant number of variables, convexification is not viable, as it is based on an explicit list of all feasible solutions in terms of the integer variables. In the literature, MIOCPs with static controls and PDE constraints are usually handled differently, either by concentrating on linear PDEs only [11] or by linearization [16,18].

As pointed out above, mixed-integer optimal control with static discrete controls and combinatorial as well as PDE constraints is an open field of research, especially if it comes to global solvers. In this paper, we consider a problem with integer decisions $u$ that define a linear cost function $c^\top u$ to be minimized, subject to combinatorial constraints $u \in \mathcal{U}$. We further require the state $y$ of a semilinear elliptic PDE (depending on $u$) to reach a given reference state $y_{\min}$. The problem can be written as

$$\left.\begin{array}{lrll} \min & c^\top u & & \\ \text{s.t.} & y(x) \geq y_{\min}(x) & \text{a.e. in } \Omega \\ & Ay + g(y) = \sum_{i=1}^{\ell} u_i\, \psi_i & \text{in } \Omega \\ & \frac{\partial y}{\partial n_A} + b(y) = \sum_{j=\ell+1}^{n} u_j\, \phi_j & \text{on } \Gamma_N \\ & y = 0 & \text{on } \Gamma_D \\ \text{and} & u \in \mathcal{U}. & \end{array}\right\} \qquad \text{(COCP)}$$

Herein $\Omega$ denotes a bounded domain in $\mathbb{R}^d$, $d \in \mathbb{N}$, and $\Gamma_D$ and $\Gamma_N$ are disjoint parts of its boundary such that $\Gamma_D \cup \Gamma_N = \partial\Omega$. Moreover, $A$ is a linear, elliptic operator, and $\partial/\partial n_A$ denotes the co-normal derivative associated with $A$. In addition, $g$ and $b$ denote Nemyzki operators associated with nonlinear functions. The functions $\psi_i$, $i = 1, \ldots, \ell$, and $\phi_j$, $j = \ell+1, \ldots, n$, are given and will be called form functions in all what follows. Finally, $\mathcal{U} \subseteq \mathbb{Z}^n$ is a bounded set of (discrete) admissible controls. The precise assumptions on the data and quantities in (COCP) are formulated in Section 2, where we also give an application example. The main assumption on which our approach is based will be the pointwise convexity of $g$ and $b$.

Generally speaking, Problem (COCP) can model applications in areas where the optimization of a static diffusion process is desired, subject to a given minimum state. Our algorithmic approach employs the special structure of (COCP), in particular the state constraints $y \geq y_{\min}$. In the context of classical optimal control problems without integrality constraints, pointwise state constraints of this form are known to cause severe difficulties from a theoretical as well as a numerical point of view; see [9,1,23,12,29] and the references therein. These difficulties are mainly caused by the poor regularity of the Lagrange multipliers associated with the state constraints, which are only Borel measures in general, see [10] and [30, Section 6.2]. By contrast, in our setting we benefit from the pointwise state constraints, as our algorithmic approach exploits the particular problem structure induced by these constraints.

In the second part of the paper, we will also consider other types of constraints and objective functions within the framework of Problem (COCP).

More specifically, we will deal with lower bound constraints $y \geq y_{\min}$ as well as general linear constraints involving both the control and the state variables. Finally, we will address tracking-type objective functions of the form

$$\min \ \|S(u) - y_d\|_{L^p(\Omega)}$$

for any $p \in [1, \infty]$, where $y_d \in L^\infty(\Omega)$ is a desired state.

From the discrete point of view, (COCP) is a nonlinear combinatorial optimization problem: the objective is to minimize a linear function over the combinatorial variables $u \in \mathcal{U}$, where the lower bounds on the state variables $y$ implicitly define an infinite number of additional nonlinear constraints on the feasible set. Under our standing assumptions listed in Section 2, in particular the convexity of the functions $g(x, .)$ and $b(x, .)$ for almost all $x$, we are able to show that the latter constraints define a convex set, and derive valid linear cutting planes in the discrete controls $u$. Moreover, in the case that all control variables are binary and all form functions are non-negative, we can show that the solution operator is submodular almost everywhere, allowing us to deal with much more general constraints and objective functions within the framework of Problem (COCP).

### 1.1 Contribution

The main novelty of our approach lies in the combination of techniques from optimal control and integer programming. The approach relies on our basic observation that, for a large class of problems of type (COCP), the state is a concave and submodular function in the control variables almost everywhere.

The concavity implies that the feasible region of our problem containing lower bounds on the states becomes convex when projected onto the control space (except for the integrality requirement), which in turn allows us to apply the well-known outer approximation approach [14] to solve the problem to global optimality. The resulting algorithm iteratively solves PDEs and ILPs; standard solvers can be applied for both steps. The idea of projecting the problem onto the control space is essential here, since we deal with semilinear differential equations. In particular, the first-discretize-then-optimize approach will result in an MINLP with non-convex constraints, which is very likely to be unsolvable in practice for larger problem instances.

On the other hand, the submodularity allows us to deal with upper bounds on the states instead of lower bounds. In this case, each nonlinear constraint on the control variables $u$ derived at a point $x$ is essentially equivalent to a finite (but exponential) number of linear constraints, where the most violated one can be calculated efficiently, which can again be embedded into an outer approximation algorithm. Finally, exploiting both concavity and submodularity, we can address any linear constraint in both controls and states as well as $L^p$-tracking type objective functions for any $p \in [1, \infty]$. In all cases, we obtain a finite algorithm solving the problem to optimality.

### 1.2 Outline

In Section 2, we list the precise assumptions we need to make in order to address Problem (COCP) by our approach. Moreover, we recall some elementary results and discuss an application example. The main theoretical background of our approach is presented in Section 3, where we show the pointwise concavity of the solution operator and describe how to define cutting planes for the projection of the feasible region of (COCP) onto the control space. These cutting planes form the basis of an outer approximation algorithm for Problem (COCP) devised in Section 4. Next, we derive our submodularity results in Section 5, and explain how to combine these with the concavity results in order to address arbitrary linear constraints (Section 6) or tracking-type objective functions (Section 7). Due to the iterative structure of the algorithm, we apply reoptimization strategies to efficiently resolve the PDE for updated candidate solutions $u$; see Section 8. Finally, in Section 9, we present the results of a numerical study to demonstrate the benefits of our algorithm and its dependence on the problem parameters.

## 2 Standing Assumptions and Known Results

We start with the precise assumptions on the data and quantities in (COCP). Throughout the paper, $\Omega \subset \mathbb{R}^d$, $d \in \mathbb{N}$, denotes a bounded domain, i.e. bounded, open, and connected set, with Lipschitz boundary $\Gamma = \partial\Omega$ in the sense of [20, Def. 1.2.2.1] so that the trace on $\Gamma$ is well defined; see [20, Thm. 1.5.1.3]. Furthermore, $\Gamma_N$ and $\Gamma_D$ are disjoint parts of $\Gamma$ such that $\Gamma = \Gamma_D \cup \Gamma_N$. We define the space $V$ as the linear subset of $H^1(\Omega)$ given by

$$V = \{v \in H^1(\Omega) : v = 0 \text{ a.e. on } \Gamma_D\}$$

equipped with the standard $H^1$-norm. Furthermore, $V^*$ denotes its dual space. The operator $A : V \to V^*$ is given by the following linear elliptic differential operator of second order

$$Ay = -\sum_{i,j=1}^{d} \frac{\partial}{\partial x_i}\Big(a_{ij}(x)\frac{\partial}{\partial x_j}y(x)\Big) + \sum_{i=1}^{d} \beta_i(x)\frac{\partial}{\partial x_i}y(x) + a_0(x)y(x),$$

where $a_{ij}, \beta_k, a_0 \in L^\infty(\Omega)$, for $i, j, k = 1, \dots, d$, are such that $A$ is coercive on $V$, i.e., we have

$$\langle Av,\, v\rangle_{V^*,V} \geq \alpha \,\|v\|_V^2 \quad \forall\, v \in V \tag{1}$$

for some constant $\alpha > 0$. To keep the discussion concise, we restrict ourselves to coercive bilinear forms, satisfying (1). Depending on the particular structure of the nonlinearities, this assumption can be weakened, see Example 1 below. By $\partial/\partial n_A$ we denote the co-normal derivative associated with $A$, i.e.,

$$\frac{\partial y}{\partial n_A} = \sum_{i,j=1}^{n} n_i\, a_{ij}\, \frac{\partial y}{\partial x_j},$$

where $n : \Gamma \to \mathbb{R}^d$ is the outward unit normal on $\Gamma$. The form functions $\psi$ and $\phi$ are supposed to satisfy $\psi_i \in L^r(\Omega)$ with $r > d/2$ for all $i = 1, \dots, \ell$ and $\phi_j \in L^s(\Gamma_N)$ with $s > d - 1$ for all $j = \ell + 1, \dots, n$. We can therefore allow for comparatively irregular right hand sides in the PDE in (COCP).

Concerning the nonlinear functions $g : \Omega \times \mathbb{R} \to \mathbb{R}$ and $b : \Gamma_N \times \mathbb{R} \to \mathbb{R}$, we require the following conditions to hold:

(a) Both $g$ and $b$ satisfy the Carathéodory condition, i.e., $g(., y)$ is measurable for every fixed $y \in \mathbb{R}$ and $g(x, .)$ is continuous for almost every $x \in \Omega$, and analogously for $b$.
(b) Both $g(x, .)$ and $b(x, .)$ are non-decreasing for almost every $x \in \Omega$ and almost every $x \in \Gamma_N$, respectively.
(c) The mappings $g(x, .)$ and $b(x, .)$ are differentiable for almost every $x \in \Omega$ and almost every $x \in \Gamma_N$, respectively. Their derivatives are denoted by $g'(x, .)$ and $b'(x, .)$ and are assumed to satisfy the Carathéodory assumption as well.

We point out that these assumptions on the nonlinearities $g$ and $b$ are standard in the context of semilinear optimal control; see e.g. [37, Chapter 4]. In particular, the monotony assumption (b) is classical and ensures the existence and uniqueness of solutions to the PDE in (COCP) for a given right hand side, see Proposition 1 below.

We are now in the position to introduce the notion of weak solutions to the PDE appearing in (COCP). For this purpose let us define the space

$$Y := V \cap L^\infty(\Omega).$$

A function $y \in Y$ is said to be a weak solution to the PDE in (COCP) if it satisfies

$$\begin{aligned} \langle Ay,\, v\rangle_{V^*,V} + \int_\Omega g(x,y)\, v \,\mathrm{d}x + \int_{\Gamma_N} b(x,y)\, v \,\mathrm{d}s \\ = \sum_{i=1}^{\ell} \int_\Omega \psi_i\, v \,\mathrm{d}x\, u_i + \sum_{j=\ell+1}^{n} \int_{\Gamma_N} \phi_j\, v \,\mathrm{d}x\, u_j \quad \forall\, v \in V. \end{aligned} \tag{2}$$

Based on our assumptions on the nonlinear functions $g$ and $b$, one shows by means of classical arguments that the PDE in (COCP) admits a unique weak solution. Moreover, the associated solution operator is Fréchet-differentiable. We collect these observations in the following proposition. The underlying analysis is standard. For a detailed proof, we refer to [37, Chapter 4] and the references cited therein.

**Proposition 1**

*(i) For every $u \in \mathbb{R}^n$ there exists a unique weak solution $y \in Y$ of (2). We denote the associated solution operator by $S : \mathbb{R}^n \to Y$.*

*(ii) The operator $S$ is continuously Fréchet differentiable from $\mathbb{R}^n$ to $Y$ and its derivative $\eta = S'(u)h$ in direction $h \in \mathbb{R}^n$ is given by the solution of the linearized PDE*

$$\begin{aligned} \langle A\eta,\, v\rangle_{V^*,V} + \int_\Omega g'(x,y)\eta\, v\,\mathrm{d}x + \int_{\Gamma_N} b'(x,y)\eta\, v\,\mathrm{d}s \\ = \sum_{i=1}^{\ell} \int_\Omega \psi_i\, v\,\mathrm{d}x\, h_i + \sum_{j=\ell+1}^{n} \int_{\Gamma_N} \phi_j\, v\,\mathrm{d}x\, h_j \quad \forall\, v \in V\ . \end{aligned} \tag{3}$$

*Remark 1* Under mild additional assumptions on the problem data, in particular the coefficient function $a_{ij}$ and the domain, it can be shown that the state is even continuous and the same holds true for the linearized state, i.e., the solution of (3), see [21]. However, as the continuity of the state is not mandatory for our algorithmic approach, we do not impose these additional assumptions. We point out that no additional regularity assumptions on the form functions are needed for this continuity result.

From the remaining quantities in (COCP) we require the following: the set $\mathcal{U}$ is assumed to be bounded and given by an integer linear description

$$\mathcal{U} = \{u \in \mathbb{Z}^n : Gu \leq h\}$$

with $G \in \mathbb{R}^{m\times n}$ and $h \in \mathbb{R}^m$, for some $m \in \mathbb{N}$, while the vector $c$ in the objective function is an arbitrary vector in $\mathbb{R}^n$. The reference state is supposed to satisfy $y_{\min} \in L^1(\Omega)$.

For the computation of global lower bounds we additionally require

**Assumption 1** *There exist numbers $y_a, y_b \in [-\infty, \infty]$, $y_a \leq y_b$, with*

$$S(u)(x) \in [y_a, y_b] \quad \text{a.e. in } \Omega \quad \forall\, u \in \operatorname{conv}(\mathcal{U}),$$

*such that the functions $g(x,.)\colon [y_a, y_b] \to \mathbb{R}$ and $b(x,.)\colon [y_a, y_b] \to \mathbb{R}$ are convex for almost all $x \in \Omega$ and almost all $x \in \Gamma_N$, respectively.*

While the above assumptions on $g$ and $b$ concerning their monotonicity and their differentiability are standard for the discussion of semilinear elliptic PDEs in the context of optimal control, as already indicated above, Assumption 1 is fairly restrictive. Nevertheless, the following example shows that there are application driven problems where this assumption is satisfied.

*Example 1* We consider the stationary heating of a metallic workpiece. If the workpiece is assumed to be homogeneous and isotropic, the operator $A$ is given by

$$A = -\kappa\Delta = -\kappa \sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2}\ ,$$

where $\kappa > 0$ denotes the (constant) heat conductivity of the material. If the material is heated up to higher temperatures, radiation has to be taken into account. To be more precise, the heat flux through the boundary equals the

difference of emitted and absorbed radiation. On the other hand, by Fourier law, this heat flux equals the normal derivative of $y$. This leads to Boltzmann type radiation boundary conditions of the form

$$\kappa \nabla y \cdot n + \sigma\, |y|^d y = \sigma\, y_0^{d+1}, \tag{4}$$

where $\sigma > 0$ denotes the Boltzmann radiation constant for the particular dimension $d$ and $y_0 > 0$ is a fixed external temperature. The boundary condition (4) models the radiation of an ideal black radiator, see [24, Section 12.3.3] for details. Note that the coercivity assumption in (1) is not satisfied in this example. However, this assumption is only needed to ensure existence and uniqueness of solutions, which, in case of this example, can be shown by the direct method of calculus of variations in combination with the generalized Friedrich's inequality.

If we assume that the workpiece is heated up by $n \in \mathbb{N}$ fixed volume sources $\psi_1, \dots, \psi_n$, generated for instance by induction heating, then the PDE modeling the stationary heat conduction with radiation boundary conditions reads, in its strong form, as follows:

$$\begin{aligned} -\kappa \Delta y &= \sum_{i=1}^{n} u_i \psi_i \quad \text{in } \Omega \\ \kappa \nabla y \cdot n + \sigma\, |y|^d y &= \sigma\, y_0^{d+1} \qquad \text{on } \Gamma. \end{aligned} \tag{5}$$

Herein the discrete control variables $u \in \mathcal{U} := \{0,1\}^n$ model the switching of the heat sources. By setting $g \equiv 0$, $\Gamma_D = \emptyset$, $\Gamma_N = \Gamma$, and

$$b : \mathbb{R} \to \mathbb{R}, \quad b(y) = \sigma(|y|^d y - y_0^{d+1}), \tag{6}$$

this problem fits into our general setting and the results of Proposition 1 apply so that there is a solution operator $S : \mathbb{R}^n \to Y$ associated with (5).

Assumption 1 however is not satisfied in general, since $b$, as defined in (6), is not convex. Nevertheless, if we focus on pure heating processes, then we may assume that $\psi_i(x) \geq 0$ a.e. in $\Omega$ for all $i = 1, \dots, n$. Consequently, the (weak) maximum principle gives $S(u)(x) \geq 0$ a.e. in $\Omega$ for all $u \in \operatorname{conv}(\mathcal{U})$. By the positivity of the trace operator, we obtain $S(u)(x) \geq 0$ a.e. on $\Gamma$. Thus, by setting $y_a = 0$ and $y_b = \infty$, we find that Assumption 1 is satisfied in this case, since $b : [0, \infty) \to \mathbb{R}$ is clearly convex.

If the vector $c \in \mathbb{R}^n$ in the objective function measures the cost of each source function $\psi_i$, e.g. in terms of energy consumption, then every solution of (COCP) yields a most efficient way of switching on the sources in order to pointwisely keep the temperature at a desired minimal temperature $y_{\min}$. This is of interest for the optimization of hardening processes of steel workpieces, where it is essential to pointwisely reach the austenitic temperature.

## 3 Pointwise concavity

In all of what follows, we denote by $\max(.,0)$ the function $\mathbb{R} \ni r \mapsto \max\{r,0\} \in \mathbb{R}$, and the associated Nemyzkii operators in $H^1(\Omega)$ and $L^2(\Gamma)$, respectively, are denoted in the same way for the sake of convenience.

**Lemma 1** *For every $v \in H^1(\Omega)$ we have $\tau \max(v,0) = \max(\tau v, 0)$ a.e. on $\Gamma$, where $\tau : H^1(\Omega) \to L^2(\Gamma)$ denotes the trace operator.*

*Proof* Let $v \in H^1(\Omega)$ be arbitrary. Since $\Omega$ has a Lipschitz boundary, [20, Thm. 1.4.2.1] implies the existence of a sequence $\{v_n\}$ in $C(\bar{\Omega})$ that strongly converges to $v$ in $H^1(\Omega)$. Then the continuity of $\max(.,0)$ in $H^1(\Omega)$ and $L^2(\Gamma)$, respectively, and the one of the trace $\tau : H^1(\Omega) \to L^2(\Gamma)$ imply

$$\begin{aligned}\tau \max(v,0) = \tau \max\big(\lim_{n\to\infty} v_n, 0\big) &= \lim_{n\to\infty} \tau \max(v_n,0) \\ &= \lim_{n\to\infty} \max(\tau v_n, 0) = \max(\tau v, 0),\end{aligned}$$

where we used the continuity of $v_n$ up to the boundary $\Gamma$. □

The following result forms the basis of our approach. It implies that the projection of the feasible set onto the control space is convex.

**Theorem 2** *Under our standing assumptions, in particular Assumption 1, the mappings*

$$\operatorname{conv}(\mathcal{U}) \ni u \mapsto S(u)(x) \in \mathbb{R} \quad \textit{and} \quad \operatorname{conv}(\mathcal{U}) \ni u \mapsto \big(\tau S(u)\big)(x) \in \mathbb{R}$$

*are concave for almost every $x \in \Omega$ and almost every $x \in \Gamma_N$.*

*Proof* The proof is similar to the one of the weak maximum principle. Consider $u_1, u_2 \in \operatorname{conv}(\mathcal{U})$ and $\lambda \in [0,1]$. Define $y_i \in Y$, $i = 1,2,3$, by

$$y_1 := S(u_1),\ y_2 := S(u_2),\ y_3 := S(\lambda u_1 + (1-\lambda)u_2).$$

If one subtracts the weak formulation for $y_3$ from the sum of the ones for $y_1$ and $y_2$ scaled by $\lambda$ and $(1-\lambda)$, respectively, it follows that

$$\begin{aligned}&\langle Ay_4,\, v\rangle_{V^*,V} + \int_\Omega \big(\lambda g(x,y_1) + (1-\lambda) g(x,y_2) - g(x,y_3)\big) v \,\mathrm{d}x \\ &\quad + \int_{\Gamma_N} \big(\lambda b(x,y_1) + (1-\lambda) b(x,y_2) - b(x,y_3)\big) v \,\mathrm{d}s = 0 \quad \forall\, v \in V,\end{aligned} \tag{7}$$

where $y_4 := \lambda y_1 + (1-\lambda) y_2 - y_3$. Next we choose $v = y_4^+ := \max(y_4, 0)$ as test function, which is in $V$ due to [25, Thm. A.1]. Let us define (up to sets of zero measure)

$$\Omega_+ := \{x \in \Omega : y_4(x) > 0\} \quad \text{and} \quad \Gamma_+ := \{x \in \Gamma_N : (\tau y_4)(x) > 0\}.$$

Then the convexity of $g(x,.)$ by Assumption 1 implies for the second addend on the left hand side of (7) that

$$\begin{aligned}\int_\Omega \big(\lambda g(x,y_1)+(1-\lambda)g(x,y_2)-g(x,y_3)\big)y_4^+\,\mathrm{d}x \\ = \int_{\Omega_+} \big(\lambda g(x,y_1)+(1-\lambda)g(x,y_2)-g(x,y_3)\big)y_4\,\mathrm{d}x \\ \geq \int_{\Omega_+} \big(g(x,\lambda y_1+(1-\lambda)y_2)-g(x,y_3)\big)y_4\,\mathrm{d}x \geq 0\end{aligned} \tag{8}$$

where the second inequality follows from monotonicity of $g(x,.)$ and since

$$\lambda y_1+(1-\lambda)y_2 > y_3 \quad \text{a.e. in } \Omega_+$$

by definition of $\Omega_+$. In view of Lemma 1, we can argue completely analogously in case of the third addend in (7) to obtain

$$\int_{\Gamma_N} \big(\lambda b(x,y_1)+(1-\lambda)b(x,y_2)-b(x,y_3)\big)y_4^+\,\mathrm{d}s \geq 0. \tag{9}$$

All in all, thanks to $\nabla y_4^+ = \chi_{\Omega_+}\nabla y_4$, see [25, Thm. A.1], (7)–(9) yield

$$\alpha\,\|y_4^+\|^2_{H^1(\Omega)} \leq \langle Ay_4^+,\, y_4^+\rangle_{V^*,V} = \langle Ay_4,\, y_4^+\rangle_{V^*,V} \leq 0,$$

and hence $y_4^+ = \max(y_4,0) = 0$. The definition of $y_4$ thus implies

$$\lambda y_1(x)+(1-\lambda)y_2(x) \leq y_3(x) \quad \text{a.e. in } \Omega,$$

which is the desired concavity of $u \mapsto S(u)(x)$. The result for the trace again follows from the positivity of the trace operator. □

Completely analogously one shows that $u \mapsto S(u)(x)$ and $u \mapsto (\tau S(u))(x)$ are *convex* provided that $g(x,.)$ and $b(x,.)$ are *concave*.

**Lemma 2** *For every $u \in \operatorname{conv}(\mathcal{U})$ and every $h \in \mathbb{R}^n$ with $u+h \in \operatorname{conv}(\mathcal{U})$ we have*

$$S(u+h)(x) \leq S(u)(x) + (S'(u)h)(x) \quad \textit{a.e. in } \Omega.$$

*Proof* The operator $S$ is Fréchet-differentiable from $\mathbb{R}^n$ to $L^\infty(\Omega) \hookrightarrow Y$ by Proposition 1(ii). For arbitrary $u,h \in \mathbb{R}^n$ we thus have

$$\lim_{t\to 0} \frac{S(u+t\,h)(x)-S(u)(x)}{t} = (S'(u)h)(x) \quad \text{f.a.a. } x \in \Omega.$$

Together with the pointwise concavity from Theorem 2, we obtain

$$S(u+h)(x)-S(u)(x) \leq \lim_{t\searrow 0} \frac{S(u+t\,h)(x)-S(u)(x)}{t} = (S'(u)h)(x)$$

for almost all $x \in \Omega$. □

**Corollary 1** *For all* $\overline{u} \in \mathrm{conv}\,(\mathcal{U})$ *and almost all* $x \in \Omega$*, the inequality*

$$S(\overline{u})(x) + (S'(\overline{u})(u - \overline{u}))(x) \geq y_{\min}(x)$$

*is valid for all feasible solutions of* (COCP)*.*

Note that the inequalities introduced in Corollary 1 are linear in the control variables $u$, for given $x$ and $\overline{u}$.

## 4 Outer Approximation Algorithm

In the following, we explain how to use the results of the previous section in order to address (COCP) by an outer approximation approach [14]. Let

$$\tilde{\mathcal{U}} := \{u \in \mathcal{U} : S(u)(x) \geq y_{\min}(x) \text{ a.e. in } \Omega\}$$

denote the feasible set of Problem (COCP), in terms of the control variables. We thus aim at solving the problem

$$\left.\begin{array}{ll} \min & c^\top u \\ \text{s.t.} & u \in \tilde{\mathcal{U}}\ . \end{array}\right\} \qquad \text{(COCP')}$$

The complexity of (COCP') is now hidden in the definition of the set $\tilde{\mathcal{U}}$. The results of the previous section allow us to define an outer approximation algorithm for (COCP'), as follows.

Outer Approximation Algorithm for (COCP)

1. Set $\mathcal{U}_0 := \mathcal{U}$.
2. Minimize $c^\top u$ over $u \in \mathcal{U}_0$, let $u^*$ be the resulting optimizer. If $\mathcal{U}_0 = \emptyset$, return "problem is infeasible".
3. Compute $y^*$ by solving

$$\begin{aligned} Ay + g(y) &= \sum_{i=1}^{\ell} u_i^*\, \psi_i && \text{in } \Omega \\ \frac{\partial y}{\partial n_A} + b(y) &= \sum_{j=\ell+1}^{n} u_j^*\, \phi_j && \text{on } \Gamma_N \\ y &= 0 && \text{on } \Gamma_D\ . \end{aligned}$$

4. If $y^* \geq y_{\min}$ a.e., return $u^*$ as optimal solution.
5. Choose some $x^* \in \Omega$ with $y^*(x^*) < y_{\min}(x^*)$ at random, add

$$y^*(x^*) + (S'(u^*)(u - u^*))(x^*) \geq y_{\min}(x^*)$$

as linear inequality in $u$ to $\mathcal{U}_0$, and go to Step 2.

This algorithm is based on Corollary 1, which yields an efficient method to cut off any vector $u \in \mathcal{U}$ violating some of the constraints

$$y^*(x) \geq y_{\min}(x)$$

by a cutting plane, i.e., by a linear constraint on $\mathcal{U}$ that is valid for $\tilde{\mathcal{U}}$.

**Theorem 3** *The above algorithm terminates in finite time. With probability one, it returns an optimal solution to Problem* (COCP).

*Proof* First note that in every iteration, the algorithm either cuts off a point from $\mathcal{U}$ or terminates. Indeed, the left hand side of the constraint added in Step 5 agrees with $S(u)(x^*)$ when $u = u^*$. The number of iterations is thus limited by $|\mathcal{U}|$, which is finite by the boundedness of $\mathcal{U} \subseteq \mathbb{Z}^n$. The correctness follows from the fact that all added linear inequalities are valid for $\mathcal{U}_0$ with probability one, which was shown in the previous section. □

*Remark 2* As indicated in Remark 1, the range of $S$ and $S'(u)$ is contained in $C(\overline{\Omega})$ under mild assumptions on the data. The inequality in Corollary 1 then holds for every $x \in \overline{\Omega}$ rather than almost everywhere. In particular, our outer approximation algorithm certainly returns an optimal solution in this case, not only with probability one.

For Step 2 of the above algorithm, one can use any standard ILP solver. Step 3 requires to solve a nonlinear PDE. We emphasize that the PDE associated with the inequality constraint in Step 5 is *linear*. Thus, to obtain $S'(u^*)u$ as a linear expression in the control vector $u$, one solves $n$ linear PDEs of the form (3) corresponding to the $n$ form functions and employs the superposition principle to obtain

$$S'(u^*)u = \sum_{i=1}^{\ell} u_i \, S'(u^*)\phi_i + \sum_{j=\ell+1}^{n} u_j \, S'(u^*)\psi_j \; .$$

In particular, the coefficient of $u_i$ is $S'(u^*)\phi_i$, $i = 1, \ldots, \ell$, and $S'(u^*)\psi_i$, $i = \ell + 1, \ldots, n$, respectively. Therefore, we need to solve $n$ linear PDEs to produce the cutting plane in Step 5.

In practice, the main challenge is to keep the number of outer iterations of the above algorithm as small as possible. For this, it is preferable to compute more than one cutting plane per iteration, e.g., by considering several points $x \in \Omega$. The details of our implementation are discussed in Section 9. Moreover, the iterative structure of the algorithm suggests to use reoptimization techniques, in particular for initializing the solution algorithm for the PDE in Step 3. This is exploited in Section 8.

## 5 Pointwise submodularity

In order to extend our approach to more general problems than (COCP), we now show that the pointwise solution operator $S(u)(x)$ is submodular for almost all $x$ under additional assumptions (which are satisfied in Example 1). A submodular function $f$ is a function $f\colon \{0,1\}^n \to \mathbb{R}$ satisfying

$$f(u) + f(v) \geq f(u \wedge v) + f(u \vee v)$$

for all $u, v \in \{0,1\}^n$, where we define

$$(u \wedge v)_i := \min\{u_i, v_i\}$$
$$(u \vee v)_i := \max\{u_i, v_i\}$$

for $i = 1, \ldots, n$. This property can be equivalently described as a diminishing returns property. Interpreted in terms of Example 1, submodularity means that the increase in the temperature in a given point $x$ that is induced by switching on a particular heat source is smaller when other heat sources are already switched on. In combinatorial optimization, submodularity is a key property for designing efficient algorithms; see, e.g., [17]. We will discuss below how submodularity allows to generate valid cutting planes modeling lower bounds on the states in (COCP).

In the following, we assume that all control variables are binary, i.e., that $\mathcal{U} \subseteq \{0,1\}^n$. Moreover, we now need

**Assumption 4** *The form functions $\psi_i$ for $i = 1, \ldots, \ell$ and $\phi_j$ for $j = \ell + 1, \ldots, n$ are non-negative almost everywhere in $\Omega$ and almost everywhere on $\Gamma_N$, respectively.*

Under these assumptions, and using the convexity of the functions $g$ and $b$, we can show the submodularity of the states $S(u)(x)$ in the binary control variables $u \in \{0,1\}^n$. As a first step, we show the following auxiliary result.

**Lemma 3** *Let $h\colon \mathbb{R}_+ \to \mathbb{R}_+$ be a convex and monotonously increasing function. Then for all non-negative $b, c \in \mathbb{R}$ and $a \geq b + c$, we have $h(0) + h(a) \geq h(b) + h(c)$.*

*Proof* Since $b, c \geq 0$, we have $0 \leq b \leq b + c$, so there exists $\lambda \in [0,1]$ with $b = \lambda(b + c)$ and hence $c = (1 - \lambda)(b + c)$. Using convexity, this implies

$$(1 - \lambda)h(0) + \lambda h(b + c) \geq h((1 - \lambda)0 + \lambda(b + c)) = h(b)$$

and

$$\lambda h(0) + (1 - \lambda)h(b + c) \geq h(\lambda 0 + (1 - \lambda)(b + c)) = h(c)\ .$$

Summing up, we obtain $h(0) + h(b + c) \geq h(b) + h(c)$. By monotonicity, the assumption $a \geq b + c$ implies $h(a) \geq h(b + c)$. This completes the proof. □

**Theorem 5** *Under our standing assumptions, the function*

$$\{0,1\}^n \ni u \mapsto S(u)(x) \in \mathbb{R}$$

*is submodular for almost all $x \in \Omega$.*

*Proof* Choosing $u, w \in \{0,1\}^n$, we need to show

$$S(u) + S(w) \geq S(u \wedge w) + S(u \vee w) \quad \text{a.e. in } \Omega. \tag{10}$$

To this end, we set

$$\begin{aligned} y_{10} &:= S(u) - S(u \wedge w), \quad & y_{11} &:= S(u \vee w) - S(u \wedge w), \\ y_{01} &:= S(w) - S(u \wedge w), \quad & y_{00} &:= S(u \wedge w). \end{aligned}$$

Now define $\eta := y_{10} + y_{01} - y_{11} \in V$, so that (10) is equivalent to

$$\eta(x) \geq 0 \quad \text{a.e. in } \Omega\,. \tag{11}$$

Combining the weak formulations for $S(u)$, $S(v)$, $S(u \wedge w)$, and $S(u \vee w)$, and taking into account that $u + w = (u \wedge w) + (u \vee w)$, we obtain

$$\begin{aligned} \langle A\eta,\, v\rangle_{V^*,V} = &\int_\Omega \big(g(y_{00}) + g(y_{11} + y_{00}) - g(y_{10} + y_{00}) - g(y_{01} + y_{00})\big) v \,\mathrm{d}x \\ &+ \int_{\Gamma_N} \big(b(y_{00}) + b(y_{11} + y_{00}) - b(y_{10} + y_{00}) - b(y_{01} + y_{00})\big) v \,\mathrm{d}s \end{aligned}$$

for all $v \in V$. By [25, Thm. A.1], the function $\eta_-(x) := \min\{0, \eta(x)\}$ belongs to $V$ and can thus be inserted as test function. We obtain

$$\begin{aligned} &\int_\Omega \big(g(y_{00}) + g(y_{11} + y_{00}) - g(y_{10} + y_{00}) - g(y_{01} + y_{00})\big)\eta_- \,\mathrm{d}x \\ &\quad = \int_{\Omega_-} \big(g(y_{00}) + g(y_{11} + y_{00}) - g(y_{10} + y_{00}) - g(y_{01} + y_{00})\big) \\ &\qquad\qquad \big(y_{10} + y_{01} - y_{11}\big) \,\mathrm{d}x, \end{aligned}$$

where

$$\Omega_- := \{x \in \Omega : \eta(x) < 0\} = \{x \in \Omega : y_{10}(x) + y_{01}(x) - y_{11}(x) < 0\}\,.$$

Due to Assumption 4 and the monotonicity of $g$ and $b$, the weak maximum principle applies to the PDE in (2). Thus, as $u \geq u \wedge w$, $w \geq u \wedge w$, and $u \vee w \geq u \wedge w$, we obtain

$$y_{10},\ y_{01},\ y_{00} \geq 0 \quad \text{a.e. in } \Omega.$$

Hence we can apply Lemma 3 to the function $\mathbb{R}_+ \ni s \mapsto h(s) := g(x, s + y_{00}(x)) \in \mathbb{R}_+$, which is convex for almost all $x \in \Omega$ by Assumption 1, and derive

$$g(y_{00}) + g(y_{11} + y_{00}) - g(y_{10} + y_{00}) - g(y_{01} + y_{00}) \geq 0 \quad \text{a.e. in } \Omega_-$$

which in turn implies

$$\int_\Omega \big(g(y_{00}) + g(y_{11} + y_{00}) - g(y_{10} + y_{00}) - g(y_{01} + y_{00})\big)\eta_- \,\mathrm{d}x \leq 0\,.$$

Using the positivity of the trace operator, see [25, Prop. 5.2], one can show analogously that

$$\int_{\Gamma_N} \big(b(y_{00}) + b(y_{11} + y_{00}) - b(y_{10} + y_{00}) - b(y_{01} + y_{00})\big)\eta_- \,\mathrm{d}s \leq 0 \,,$$

which in summary shows $\langle A\eta,\, \eta_- \rangle_{V^*,V} \leq 0$. Thanks to (1), this implies

$$\|\eta_-\|^2_{H^1_0(\Omega)} = 0 \,,$$

so that $\eta_- = 0$ almost everywhere in $\Omega$. This yields (11) and thus the desired result. □

In Section 3, we have shown that the function $u \mapsto S(u)(x)$ is concave under the given assumptions, in particular the convexity of $g$, so that the point-wise lower bound $S(u)(x) \geq y_{\min}(x)$ is a convex constraint for almost all $x \in \Omega$. In a similar way, Theorem 5 can be used to deal with *upper* bound constraints on the states in the form $S(u)(x) \leq y_{\max}(x)$. Indeed, if $u \mapsto S(u)(x)$ is a submodular function, then the latter constraint is equivalent to an exponential size class of linear constraints on $u$ which can be separated efficiently. More precisely, the separation problem for the polyhedron

$$\mathrm{epi}(S(.)(x)) = \mathrm{conv}\,\{(u, z) \in \{0,1\}^n \times \mathbb{R} \mid z \geq S(u)(x)\}$$

can be solved in $O(nF + n \log n)$ time, where $F$ is the time needed to calculate $S(u)(x)$ for given $u$ [15,28]; for more details, see, e.g., Theorem 1 in [5]. Embedding this separation algorithm into our ILP-based outer approximation scheme, we again obtain an exact optimization algorithm. As every infeasible binary point $u^*$ can be cut off by an appropriate cutting plane of this type, the resulting algorithm is finite again.

## 6 Linear constraints involving the states

Using the results of the previous sections and assuming $\mathcal{U} \subseteq \{0,1\}^n$, our outer approximation approach presented in Section 4 can also deal with arbitrary linear constraints of the form

$$a^\top u + \int_\Omega f(x) S(u)(x) \,\mathrm{d}x \leq b \,, \tag{12}$$

where $a \in \mathbb{R}^n$, $b \in \mathbb{R}$, and $f \in L^1(\Omega)$. After the computation of $u^*$ and $y^*$ in Steps 2 and 3, we first check whether

$$a^\top u^* + \int_\Omega f(x) y^*(x) \,\mathrm{d}x \leq b$$

holds. If not, we can again derive a valid but violated cutting plane as follows: we first rewrite (12) as

$$a^\top u + f_1(u) - f_2(u) \leq b$$

where

$$f_1 : \{0,1\}^n \to \mathbb{R}, \quad f_1(u) := \int_\Omega \max\{f(x), 0\} S(u)(x)\,\mathrm{d}x$$

and

$$f_2 : \mathbb{R}^n \to \mathbb{R}, \quad f_2(u) := \int_\Omega \max\{-f(x), 0\} S(u)(x)\,\mathrm{d}x\ .$$

By Theorem 5 and thanks to the monotonicity of the Lebesgue integral, the function $f_1$ is submodular. Using the same technique as described in Section 5, we can now derive an inequality of the form $a_1^\top u + b_1 \leq f_1(u)$ that is valid for all $u \in \{0,1\}^n$ and satisfied with equality for $u^*$. Similarly, using Theorem 2 and Corollary 1, we can find $a_2 \in \mathbb{R}^n$ and $b_2 \in \mathbb{R}$ with $a_2^\top u + b_2 \geq f_2(u)$ for all $u \in \mathbb{R}^n$ and $a_2^\top u^* + b_2 = f_2(u^*)$. Combining these results, we obtain a linear inequality

$$a^\top u + a_1^\top u - a_2^\top u \leq b - b_1 + b_2$$

that is valid for all $u \in \{0,1\}^n$ and violated by the given $u^*$. Adding this inequality will thus exclude $u^*$ in further iterations without cutting off any feasible binary solution.

It is easy to see that Theorem 3 still holds after these extensions, provided that the number of linear constraints of type (12) being considered is finite. More precisely, the outer approximation algorithm extended as described above will still terminate in finite time, and the resulting solution will satisfy all additional linear contraints with probability one.

## 7 Tracking-type objective functions

Still assuming $\mathcal{U} \subseteq \{0,1\}^n$, the results presented in the previous sections can be used to address tracking type problems, provided that the distance to a given desired state is measured in an $L^p$-norm, $p \in [1, \infty]$. While the reflexive cases, where $1 < p < \infty$, can be tackled with a unified approach, the cases $p = 1$ and $p = \infty$ require a particular treatment.

### 7.1 $L^\infty$-norm

Let us first focus on the $L^\infty$-norm, as this turns out to be closest to the problem with pointwise state constraints discussed above. Given a desired state $y_d \in L^\infty(\Omega)$, we consider the problem

$$\begin{aligned} \min \quad & \|S(u) - y_d\|_{L^\infty(\Omega)} \\ \text{s.t.} \quad & u \in \{0,1\}^n\ . \end{aligned}$$

Note that $S : \mathbb{R}^n \to Y \hookrightarrow L^\infty(\Omega)$ so that the objective is well defined. We equivalently rewrite the problem as

$$\begin{aligned} \min \quad & z \\ \text{s.t.} \quad & -z \leq S(u)(x) - y_d(x) \leq z \quad \text{a.e. in } \Omega \\ & u \in \{0,1\}^n \end{aligned}$$

and again obtain a problem with pointwise state constraints. Consequently, we can design an ILP-based outer approximation algorithm similar to the one proposed above: for given $u^* \in \{0,1\}^n$ and $z^* \in \mathbb{R}$, we check whether the convex constraint $-z^* \leq S(u^*)(x) - y_d(x)$ is violated for some $x^* \in \Omega$. If so, we can separate $(u^*, z^*)$ by a tangent inequality based on Theorem 2. If the submodular constraint $S(u^*)(x) - y_d(x) \leq z^*$ is violated at some $x^* \in \Omega$, we can separate $(u^*, z^*)$ as described in Section 5.

The finite convergence of the resulting algorithm is still guaranteed, provided that we always derive a constraint yielding the tightest possible bound on $z$ for a given $u^*$. Indeed, this implies that any solution of the form $(u^*, z^*)$ appearing in a later iteration of our algorithm will be feasible, so that again the number of iterations can be bounded by $|\mathcal{U}|$.

### 7.2 $L^1$-norm

Next, let us turn to $L^1$-tracking type problems of the form

$$\begin{aligned} \min \quad & \int_\Omega |S(u)(x) - y_d(x)| \,\mathrm{d}x \\ \text{s.t.} \quad & u \in \{0,1\}^n \,. \end{aligned} \tag{13}$$

Again, we can combine Theorem 2 and Theorem 5: the latter shows that

$$f_1 : \{0,1\}^n \to \mathbb{R}, \quad f_1(u) := \int_\Omega S(u)(x) \,\mathrm{d}x$$

is a submodular function, while the former implies that

$$f_2 : \mathbb{R}^n \to \mathbb{R}, \quad f_2(u) := \int_\Omega \min\{y_d(x), S(u)(x)\} \,\mathrm{d}x$$

is a concave function. Now using that

$$\int_\Omega |S(u)(x) - y_d(x)| \,\mathrm{d}x = f_1(u) - 2f_2(u) + \int_\Omega y_d(x) \,\mathrm{d}x \,,$$

where the last term is constant, we derive that Problem (13) is equivalent to

$$\left.\begin{aligned} \min \quad & z \\ \text{s.t.} \quad & f_1(u) - 2f_2(u) \leq z \\ & u \in \{0,1\}^n \,. \end{aligned}\right\} \tag{14}$$

Now, similar to Section 6, we can underestimate the terms $f_1(u)$ and $-2f_2(u)$ independently by affine linear functions in $u$ and thus obtain cutting planes that can be used in our outer approximation algorithm. For $f_1$, we can use submodularity exactly as before. However, the situation for $f_2$ is slightly more complicated now, since we have to deal with non-differentiability. The following result shows how to compute an affine linear overestimator for $f_2$.

**Proposition 2** *Let $u^* \in \mathbb{R}^n$ and $y^* := S(u^*)$. Moreover, let $g \in L^\infty(\Omega)$ be an arbitrary measurable selection of the convex subdifferential of the* max*-function at $y_d - y^*$, i.e.,*

$$g(x) \in \partial \max(y_d(x) - y^*(x)) \quad \text{a.e. in } \Omega. \tag{15}$$

*Then the inequality*

$$f_2(u) \leq f_2(u^*) - \int_\Omega g(x)\,(S'(u^*)(u-u^*))(x)\,\mathrm{d}x \tag{16}$$

*holds true for all $u \in \mathbb{R}^n$.*

*Proof* The convex functional $-f_2$ is the composition of the solution mapping $S$ and

$$\psi : L^1(\Omega) \ni v \mapsto \int_\Omega \big(\max\{0, y_d(x) - v(x)\} - y_d(x)\big)\,\mathrm{d}x \in \mathbb{R}.$$

The monotonicity of the integral implies that every function $\chi \in L^\infty(\Omega) = L^1(\Omega)^*$ satisfying $\chi(x) \in \partial \max(y_d(x) - v(x))$ a.e. in $\Omega$, i.e.,

$$\chi \in [0,1] \text{ a.e. in } \Omega, \quad \chi = -1 \text{ a.e., where } y_d > v, \quad \chi = 0 \text{ a.e., where } y_d < v,$$

forms an element of $\partial\psi(v)$. Since $\mathrm{dom}(\psi) = L^1(\Omega)$ and $\psi$ is continuous, we are allowed to apply the chain rule for convex subdifferentails, cf. [35], to obtain that

$$\mathbb{R}^n \ni h \mapsto \int_\Omega g(x)(S'(u^*)h)(x)\,\mathrm{d}x \in \mathbb{R}$$

where $g$ as in (15) is an element of $\partial(-f_2)(u^*)$. This implies

$$-f_2(u) \geq -f_2(u^*) + \int_\Omega g(x)\,(S'(u^*)(u-u^*))(x)\,\mathrm{d}x$$

and hence the result. □

Note that the right hand side in (16) is an affine linear expression in $u$ and can hence be combined with the underestimator of $f_1(u)$ to obtain a cutting plane for the outer approximation approach. Finite convergence is again guaranteed in this approach. Compared to the $L^\infty$-case, the situation is even easier, since the generated constraint automatically gives the tightest possible bound on $z$ for given $u^*$.

### 7.3 Reflexive norms

Finally, we address $L^p$-tracking type problems with $p \in (1, \infty)$, i.e.,

$$\begin{array}{ll} \min & \|S(u)(x) - y_d(x)\|_{L^p(\Omega)} \\ \text{s.t.} & u \in \{0,1\}^n \,. \end{array} \tag{17}$$

Since $L^p(\Omega)$, $1 < p < \infty$, is reflexive, the unit ball in $L^p(\Omega)$ equals the set

$$\Big\{ v \in L^p(\Omega) : \int_\Omega v(x)\, w(x)\, \mathrm{d}x \le \|w\|_{L^q(\Omega)} \quad \forall\, w \in L^q(\Omega) \Big\},$$

where $q$ is the exponent conjugate to $p$, i.e., $1/p + 1/q = 1$. Restricting ourselves to test functions in the surface of the unit ball in $L^q(\Omega)$, denoted by $\partial B_q(0,1)$, Problem (17) can be reformulated as follows:

$$\begin{array}{ll} \min & z \\ \text{s.t.} & \displaystyle\int_\Omega \big(S(u) - y_d\big) w \,\mathrm{d}x \le z \quad \forall\, w \in \partial B_q(0,1) \\ & u \in \{0,1\}^n \,. \end{array}$$

For a given $w$, we can use the same techniques as before in order to replace the constraint

$$\int_\Omega \big(S(u) - y_d\big) w \,\mathrm{d}x \le z \tag{18}$$

by a linear cutting plane. It thus remains to find a test function $w \in \partial B_q(0,1)$ violating (18) for given $u$ and $z$ – or to decide that none exists. For this, we just need to check if $y^* - y_d \in Y$ and $z^* \in \mathbb{R}$ satisfy $\|y^* - y_d\|_{L^p(\Omega)} \le z^*$. If not, then (18) is violated with

$$w := \frac{1}{\|(y^* - y_d)^{p-1}\|_{L^q(\Omega)}} \, (y^* - y_d)^{p-1} \, \operatorname{sign}(y^* - y_d)^p.$$

It is easy to see that the resulting cutting plane yields the tightest possible bound on $z$ again, for a given $u^*$ (but over all $w \in \partial B_q(0,1)$). This again guarantees finite convergence of the outer approximation algorithm.

## 8 Reoptimization

Due to the iterative structure of our outer approximation algorithm presented in the previous sections, the semilinear elliptic PDE in (COCP) has to be solved many times for different values of $u$. Due to this, it is crucial to develop fast reoptimization techniques that can exploit the information collected in prior iterations. More precisely, when solving the PDE, we propose to speed up the Newton method by deriving an initial solution from either Taylor approximations or interpolations of $S(u)$ in the new control vector $u$. Both approaches are evaluated experimentally in Section 9.

### 8.1 Taylor Approximation

The first approach is to approximate $S(u)$ for a new control vector $u$ by using a first order Taylor approximation in one of the vectors $\overline{u}$ considered in an earlier iteration, assuming that

$$S(u)(x) \approx S(\overline{u})(x) + (S'(\overline{u})(u-\overline{u}))(x), \tag{19}$$

see Proposition 1(ii). Note that in our algorithm the derivatives $S'(\overline{u})h$ are calculated anyway for the construction of cutting planes as devised in Corollary 1, using the linearized PDE in Proposition 1; we thus obtain the Taylor approximation for free. It can easily be shown that for linear functions $g$ and $b$ equality holds in (19). More generally, this approach can be expected to work well whenever the PDE in (COCP) is nearly linear.

### 8.2 Inter-/Extrapolation

The second approach uses inter- or extrapolation. It aims at predicting the solution $S(u)(x)$ for a new $u$, if enough sample points $(u^{(j)}, y^{(j)})$, $j = 1, \dots, t$, are available. The approach depends on the specific semilinear elliptic PDE. We assume in the following that the inverse function $g^{-1}(x,.)\colon \mathbb{R} \to \mathbb{R}$ of $g(x,.)$ exists and restrict ourselves to the special case of $\Gamma_N = \emptyset$ for sake of simplicity. The PDE can then be written as

$$g(x,y) = \sum_{i=1}^{\ell} u_i\, \psi_i(x) - (Ay)(x)\ .$$

By neglecting the term $Ay$, we assume that $S(u)(x)$ depends on $u$ in the form

$$g(x, S(u)(x)) \approx g_1(x)^\top u + g_0(x)$$

for each fixed $x \in \Omega$ and some $g_1(x) \in \mathbb{R}^\ell$, $g_0(x) \in \mathbb{R}$. Within an interpolation scheme, one first calculates the coefficients $g_1(x)$ and $g_0(x)$ by solving a least squares problem based on the equations

$$g(x, y^{(j)}(x)) = g_1(x)^\top u^{(j)} + g_0(x),\ j = 1, \dots, t\ .$$

The initial guess for the state $y$ in $u$ can then be chosen as

$$y_{\text{init}}(x) = g^{-1}\left(g_1(x)^\top u + g_0(x)\right).$$

Note that this interpolation has to be performed for all $x \in \Omega$.

This approach can be accelerated further by simply setting $g_1(x)_i = \psi_i(x)$ and construct $g_0(x)$ out of the previously calculated $Ay$, e.g.,

$$g_0(x) = \text{mean}_{j \in J}\left(g(x, y^{(j)}(x)) - \sum_{i=1}^{\ell} u_i^{(j)}\, \psi_i(x)\right).$$

The index set $J$ can be chosen differently from $\{1, \dots, t\}$. For example, one may consider only the solutions $u^{(j)}$ nearest to the new iterate $u$.

## 9 Experimental Results

To evaluate the potential of our algorithm experimentally, we implemented it in MATLAB R2016A, using CPLEX 12.6 as ILP solver (`cplexbilp`). CPLEX is run in default settings except that the parallel mode is switched off. All computations have been performed on a 64bit Linux system with an Intel Xeon E5-2640 CPU @ 2.5 GHz. In all experiments, we set the time limit to five CPU hours.

### 9.1 Test Instances with Lower Bounds

Throughout our experiments we consider a square domain $\Omega = [0,1]^2$ and partition this domain into as many parts as we have binary optimization variables, i.e. $\Omega = \cup_i^n P_i$ with pairwise disjoint $P_i$, $i = 1, \dots, n$. The test problem is defined as follows:

$$\left.\begin{array}{lll} \min & c^\top u & \\ \text{s.t.} & y \geq 0.5\chi_{[0.1,0.9]^2} & \\ & -\Delta y + \dfrac{1}{2p} y^p = 100 \displaystyle\sum_{i=1}^{\ell} u_i \, \chi_{P_i} & \text{in } \Omega \\ & y = 0 & \text{on } \partial\Omega \\ \text{and} & u \in \mathcal{U}. & \end{array}\right\} \tag{20}$$

In particular, we do not consider any Neumann boundary conditions in our experiments, so that $\ell = n$.

Unless stated differently, we choose $n = 25$, $p = 2$, $c_i = 1$ for $i = 1, \dots, n$, and $\mathcal{U} = \{0,1\}^n$, i.e., $u_{\max} = 1$. Note that the above problem satisfies the conditions of Section 2. In particular, $g(x,y) = \frac{1}{2p} y^p$ is non-decreasing and convex since (20) is defined so that $y \geq 0$ holds for all $x \in \Omega$. The factors, e.g. $\frac{1}{2p}$, and the other constants are chosen in order to avoid trivial solutions where all switches are on (or all switches are off).

For the solution of the semilinear elliptic PDE we use a finite element method. To be more precise, we employ a standard Galerkin scheme with continuous and piecewise linear ansatz and test functions. For the computational mesh, we use a uniform Friedrich-Keller-triangulation with 10201 vertices. The discrete system arising in this way is solved by Newton's method, which terminates successfully if the residual is less than $10^{-6}$. The linear systems of equations are solved by direct solvers based on sparse LU decompositions. The computational mesh is aligned with the above mentioned partitioning defining the sets $P_i$, so that these sets are resolved exactly. In our experiments we use the `kmeans` algorithm with as many clusters as discrete controls $n$. Figure 1 shows examples of the sets $P_i$ for $n = 10, 50, 100$.

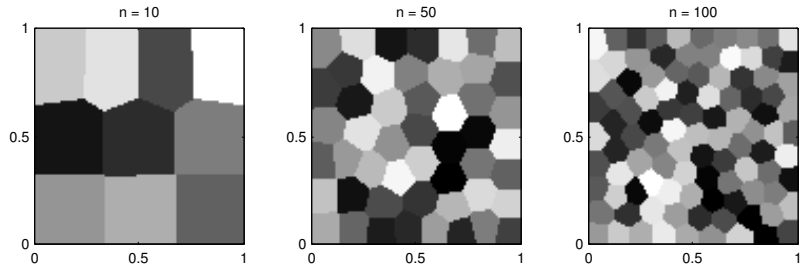


**Fig. 1** Heat clusters for 10, 50 and 100 heat sources.

### *9.1.1 Choice of Reoptimization Strategy*

As described in Section 8, the solution of the semilinear elliptic PDE can be sped up by reoptimization. We compare the Taylor approximation and the interpolation approach with two straightforward heuristics. The resulting four strategies differ in the choice of the initial solution for the Newton method:

`PDE_ZERO`: Zero vector, i.e., $y(x) = 0$ for all $x \in \Omega$.
`PDE_LINEAR`: Solution of a linear PDE obtained by neglecting the term $g(x, y)$.
`PDE_TAYLOR`: The first order Taylor approximation of $S(u)$, calculated in the closest point $\overline{u}$ to $u$ that has been considered before; see Section 8.1.
`PDE_INTERP`: Interpolation of $S(u)$, taking into account only the nearest solutions $u^{(j)}$ of $u$ (maximum five); see Section 8.2.

Whenever an initialization as described above is not applicable, e.g. in the first iteration, we choose the zero vector.

We compare the four choices above for a fixed number of $n = 25$ binary controls. As the nonlinearity of $g$, i.e. the exponent $p$ in (20), has got the highest influence on the performance of the reoptimization methods, we evaluate the cases $p \in \{1, 2, 3, 4\}$. For each $p$, Problem (20) is solved, so that each iteration of our algorithm serves as a test instance for the reoptimization heuristics. The results (number of iterations of the Newton method and time) are shown in Table 1.

The dependence of the Newton method on the exponent $p$ is highly visible. In particular, the number of iterations grows with $p$ in a similar order for all reoptimization strategies except `PDE_INTERP`. The latter method clearly dominates all others for $p = 2, 3, 4$, needing less iterations and CPU time. For $p = 1$ it is known from theory that the Taylor approximation is exact, which leads to zero further Newton iterations.

Based on these results, we choose the interpolation strategy of Section 8.2 throughout the following experiments.

### *9.1.2 Choice of Cutting Planes*

The inequalities of Corollary 1, which form the basis of our outer approximation algorithm, are valid for almost all $x \in \Omega$. This allows us to add,

| | PDE_ZERO | | PDE_TAYLOR | |
|---|---|---|---|---|
| $p$ | pde iter | cpu time [s] | pde iter | cpu time [s] |
| 1 | 2 / 2 / 2.0 | 0.46 / 0.47 / 0.46 | 1 / 1 / 1.0 | 0.10 / 0.13 / 0.11 |
| 2 | 21 / 23 / 22.6 | 7.42 / 8.04 / 7.75 | 9 / 22 / 16.6 | 2.94 / 7.62 / 5.63 |
| 3 | 31 / 35 / 33.3 | 10.87 / 12.56 / 11.82 | 16 / 32 / 25.4 | 5.54 / 11.29 / 8.91 |
| 4 | 39 / 46 / 43.5 | 14.05 / 25.30 / 16.83 | 23 / 41 / 34.7 | 8.63 / 15.51 / 13.14 |
| | PDE_LINEAR | | PDE_INTERP | |
| $p$ | pde iter | cpu time [s] | pde iter | cpu time [s] |
| 1 | 2 / 2 / 2.0 | 0.46 / 0.47 / 0.46 | 2 / 2 / 2.0 | 0.46 / 0.47 / 0.46 |
| 2 | 20 / 22 / 21.6 | 6.78 / 7.64 / 7.38 | 6 / 13 / 9.0 | 1.87 / 4.36 / 2.95 |
| 3 | 30 / 34 / 32.3 | 10.64 / 12.23 / 11.49 | 7 / 13 / 9.9 | 2.26 / 4.43 / 3.33 |
| 4 | 38 / 45 / 42.5 | 14.31 / 17.64 / 16.21 | 8 / 15 / 10.9 | 2.72 / 5.51 / 3.95 |

**Table 1** Comparison of different reoptimization heuristics for the solution of the semilinear elliptic PDE for four different exponents $p$. We state the minimum, maximum, and mean of the number of Newton iterations and of the running time needed for solving a single PDE. For the minimum and mean, the first trivial solution with zero iterations is neglected.

for any infeasible $u \in \mathcal{U}$, as many cutting planes as there are vertices $x_i^*$ of the finite element discretization that violate $y(x_i^*) \geq y_{\min}(x_i^*)$. We noticed, however, that nearby points often lead to inequalities cutting off the same vectors from $\mathcal{U}$, and thus have a negative influence on the efficiency of the overall algorithm since the solution of the ILPs is slowed down. Therefore we choose some minimal distance $r$ and enumerate all points $x_i^*$ in descending order according to the violation of the constraint $y(x_i^*) \geq y_{\min}(x_i^*)$, adding the corresponding cutting plane if and only if no point closer than $r$ to $x_i^*$ has been used before to produce a cutting plane. In particular, we obtain a set $J \subseteq \{i : y^*(x_i^*) < y_{\min}(x_i^*)\}$ such that

$$\left\|x_i^* - x_j^*\right\|_2 \geq r \quad i, j \in J,\ i \neq j\ .$$

The influence of the choice of $r$ on the performance of our algorithm is shown in Figure 2 and Figure 3, for $p = 2$ and $p = 3$, respectively. As expected, the difficulty of solving the ILPs decreases with growing $r$, as the number of constraints becomes smaller, whereas the number of iterations (and hence the total time needed to solve the PDEs) increases, as less vectors from $\mathcal{U}$ are cut off in one iteration. When combining these two effects in terms of the overall CPU time, it turns out in our experiments that the minimum is attained at $r \approx 0.04$. Although it may not be the optimal choice for arbitrary parameters, we choose $r = 0.04$ in the following for sake of comparability.

### *9.1.3 Example 1: Uniform Costs and Precedence Constraints*

We first investigate the case of uniform costs and binary variables, i.e. we keep $c_i = 1$ for all $i = 1, \ldots, n$. In other words, we minimize the number of sources we need to switch on in order to reach the pointwise minimum temperature. In our experiments, we vary the number of controls $n$ as well as

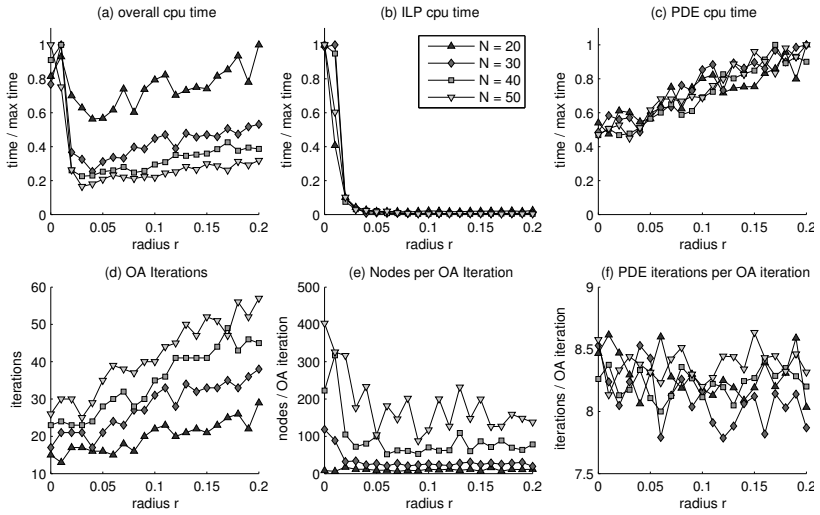


**Fig. 2** Comparison of different radii $r$ for the choice of cutting planes with $p = 2$.

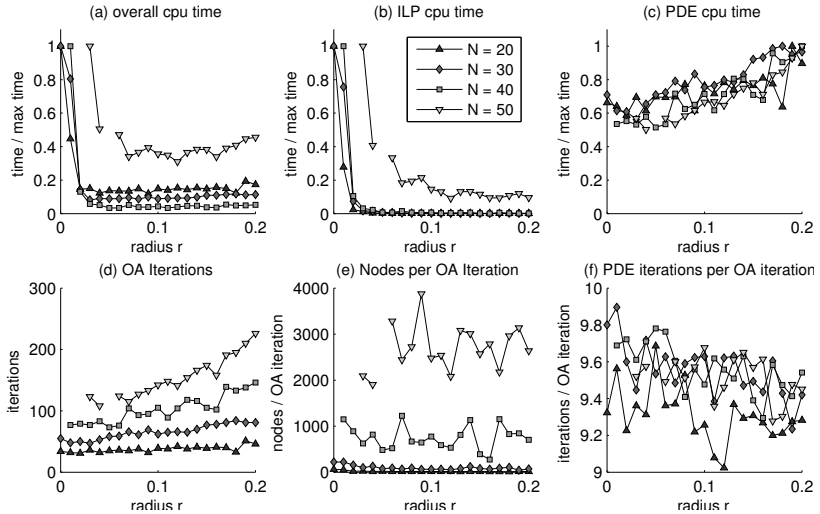


**Fig. 3** Comparison of different radii $r$ for the choice of cutting planes with $p = 3$.

the exponent $p$, in order to illustrate the influence of both the problem size and the nonlinearity. As already emphasized above, our approach is able to deal with rather general combinatorial constraints on the control vector $u$, as long as these can be modeled within the ILP solved in Step 2 of our algorithm. In the following, we set the set of admissible controls to

$$\mathcal{U} = \{u \in \{0,1\}^n \mid u_i - u_j \leq 0,\ (i,j) \in \mathcal{I}\}$$

for some index set $\mathcal{I}$. In other words, certain heat sources may be switched on only if others are switched on as well. In our experiments, the index set $\mathcal{I}$

contains at most $0.25n$ pairs $(i, j)$ with $i \neq j$ chosen at random. Note that this may induce chains of type $u_i \leq u_j \leq u_k \leq \ldots$ as well as constraints of type $u_i = u_j$.

The results are listed in Table 2, where we report the number of major iterations of the outer approximation algorithm, the number of added cutting planes, the number of nodes and the CPU time required by the ILP solver, as well as the number of iterations and the CPU time required by the PDE solver; all figures are summed up over the major iterations. The last column states the total running time for solving the instance to optimality.

We are able to solve the problem with up to 250 ($p = 1$), 120 ($p = 2$), 70 ($p = 3$) and 40 ($p = 4$) binary controls. For growing $n$ (and fixed $p$), the computation times of the ILP solves, the PDE solves and thus of the overall process increase. While the computation time for solving the PDE is dominant for small problems, the time for solving the ILPs becomes dominant for larger problems. In fact, while the size of the discretized PDE does not depend on $n$, the ILP solution time can be expected to grow exponentially in $n$ for reasons of complexity. Furthermore, the nonlinearity of the PDE – varied through the exponent $p$ – has a significant influence on the number of Newton iterations, as already known from Section 9.1.1, but also on the number of nodes within the ILP solves and major iterations of the outer approximation algorithm. Both increase with the exponent $p$. This leads to the conclusion that the cutting planes' quality is reduced for stronger nonlinearities.

When investigating the solution process over the major iterations of the outer approximation, it can be noticed that the number of nodes and computation time required by the ILP solver are not equally distributed. In fact, for most of the problems, the most difficult ILPs are those in the middle of the process, while the ILPs in the beginning and in the end are relatively easy to solve. This behavior is plotted in Figure 4 for $p = 3$, $n = 60$ and $p = 4$, $n = 30$. A possible explanation is the following: while in the beginning the growing number of cutting planes makes the problem harder to solve, the smaller number of remaining feasible solutions leads to a faster solution of the ILPs towards the end.

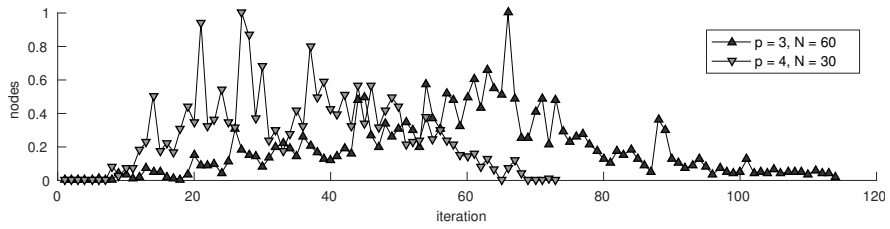


**Fig. 4** ILP nodes needed in each major iteration for the solution of (20) with $p = 3, n = 60$ and $p = 4, n = 30$. The number of nodes is scaled by the maximum number of nodes.

| | | | | | ILP solve | | PDE solve | | |
|---|---|---|---|---|---|---|---|---|---|
| $p$ | $n$ | obj | iter | #cuts | nodes | time [s] | iter | time [s] | time [s] |
| 1 | 10 | 10.00 | 2 | 394 | 0 | 0.91 | 3 | 0.47 | 4.37 |
| | 30 | 30.00 | 2 | 399 | 0 | 0.58 | 3 | 0.45 | 6.52 |
| | 50 | 32.00 | 4 | 406 | 21 | 0.88 | 7 | 1.30 | 25.71 |
| | 70 | 33.00 | 7 | 457 | 781 | 1.77 | 13 | 2.73 | 67.15 |
| | 90 | 36.00 | 9 | 457 | 15990 | 4.60 | 17 | 3.95 | 118.09 |
| | 110 | 36.00 | 7 | 445 | 4500 | 3.13 | 13 | 2.98 | 108.37 |
| | 130 | 39.00 | 8 | 461 | 10537 | 4.31 | 15 | 3.53 | 150.20 |
| | 150 | 40.00 | 5 | 451 | 35808 | 5.99 | 9 | 1.93 | 101.71 |
| | 170 | 43.00 | 10 | 484 | 110289 | 15.84 | 19 | 4.62 | 254.11 |
| | 190 | 44.00 | 12 | 496 | 242423 | 34.18 | 23 | 5.16 | 348.91 |
| | 210 | 48.00 | 19 | 550 | 3662067 | 485.92 | 37 | 7.93 | 1052.59 |
| | 230 | 48.00 | 23 | 565 | 4714413 | 746.99 | 45 | 9.65 | 1495.83 |
| | 250 | 49.00 | 14 | 580 | 11015773 | 2038.39 | 27 | 5.52 | 2504.50 |
| | 270 | - | - | - | - | - | - | - | $>$ 5h |
| 2 | 10 | 10.00 | 9 | 1624 | 18 | 2.75 | 68 | 16.43 | 34.40 |
| | 20 | 20.00 | 15 | 2815 | 92 | 4.01 | 120 | 30.23 | 83.56 |
| | 30 | 30.00 | 17 | 3358 | 395 | 5.12 | 141 | 34.90 | 119.99 |
| | 40 | 39.00 | 20 | 3587 | 723 | 8.36 | 158 | 39.29 | 168.74 |
| | 50 | 41.00 | 28 | 4956 | 3159 | 29.53 | 235 | 58.62 | 300.25 |
| | 60 | 42.00 | 29 | 5048 | 5757 | 40.14 | 253 | 62.17 | 361.85 |
| | 70 | 46.00 | 38 | 6868 | 15475 | 138.49 | 322 | 78.77 | 619.80 |
| | 80 | 52.00 | 46 | 7884 | 89042 | 393.17 | 376 | 88.81 | 1026.86 |
| | 90 | 48.00 | 45 | 6748 | 104637 | 313.71 | 381 | 89.65 | 1001.28 |
| | 100 | 50.00 | 73 | 9750 | 984609 | 2847.30 | 594 | 132.94 | 4020.34 |
| | 110 | 46.00 | 61 | 9430 | 593196 | 2216.06 | 495 | 107.43 | 3264.37 |
| | 120 | 47.00 | 67 | 10001 | 3500743 | 12183.54 | 553 | 121.23 | 13442.77 |
| | 130 | - | - | - | - | - | - | - | $>$ 5h |
| 3 | 10 | 10.00 | 13 | 2563 | 51 | 4.07 | 117 | 31.30 | 58.59 |
| | 20 | 20.00 | 27 | 5617 | 221 | 7.35 | 255 | 67.56 | 165.95 |
| | 30 | 30.00 | 44 | 8639 | 2019 | 28.70 | 398 | 104.16 | 349.30 |
| | 40 | 34.00 | 49 | 9675 | 5789 | 73.95 | 464 | 120.41 | 500.72 |
| | 50 | 33.00 | 89 | 17328 | 107817 | 841.38 | 872 | 218.74 | 1743.96 |
| | 60 | 30.00 | 114 | 20800 | 433807 | 2950.73 | 1086 | 250.90 | 4200.37 |
| | 70 | 30.00 | 137 | 26100 | 1347245 | 14775.53 | 1286 | 290.01 | 16454.80 |
| | 80 | - | - | - | - | - | - | - | $>$ 5h |
| 4 | 10 | 10.00 | 20 | 4014 | 94 | 5.73 | 207 | 55.39 | 96.48 |
| | 20 | 20.00 | 39 | 7960 | 621 | 13.18 | 414 | 109.44 | 251.83 |
| | 30 | 27.00 | 73 | 14554 | 4235 | 67.65 | 783 | 208.70 | 631.89 |
| | 40 | 40.00 | 110 | 21108 | 64513 | 450.43 | 1168 | 299.71 | 1428.49 |
| | 50 | - | - | - | - | - | - | - | $>$ 5h |

**Table 2** Results for different numbers $n$ of binary controls and different exponents $p$ with precedence constraints on the control vector $u$. The number of added cuts (#cuts), the number of B&B nodes and the time for ILP solution, as well as the number of iterations and the time for the PDE solution are summed up over the major iterations. All times are CPU times in seconds.

### *9.1.4 Example 2: Randomly Distributed Costs and Precedence Constraints*

In addition to Example 1, we consider a problem with randomly distributed costs $c$. Therefore, for every problem size $n$ one vector $c$ is selected where each $c_i$, $i = 1, \ldots, n$, is chosen independently in the interval $(0, 1)$ using a uniformly distributed random number generator. The results are shown in Table 3 for different $p$ and increasing $n$, starting at the largest possible $n$ of Table 2.

| | | | | | ILP solve | | PDE solve | | |
|---|---|---|---|---|---|---|---|---|---|
| p | $n$ | obj | iter | #cuts | nodes | time [s] | iter | time [s] | time [s] |
| 1 | 250 | 15.92 | 7 | 492 | 12524 | 10.54 | 13 | 2.73 | 242.36 |
| | 300 | 15.42 | 8 | 523 | 25321 | 12.53 | 15 | 3.47 | 338.29 |
| | 400 | 16.64 | 9 | 565 | 340830 | 88.67 | 17 | 4.04 | 572.88 |
| | 500 | 15.11 | 5 | 556 | 52982 | 20.31 | 9 | 1.99 | 325.33 |
| | 600 | 15.30 | 6 | 589 | 18563 | 18.79 | 11 | 2.44 | 472.41 |
| | 700 | 14.62 | 8 | 624 | 308315 | 112.85 | 15 | 3.27 | 846.90 |
| | 800 | 13.47 | 9 | 682 | 1147875 | 461.01 | 17 | 3.72 | 1411.75 |
| | 900 | 14.70 | 9 | 710 | 2717080 | 1317.79 | 17 | 3.85 | 2388.09 |
| | 1000 | 14.38 | 9 | 748 | 5122292 | 2633.98 | 17 | 3.45 | 3814.96 |
| | 1100 | 12.83 | 8 | 742 | 635094 | 340.30 | 15 | 3.04 | 1475.15 |
| | 1200 | 12.70 | 7 | 789 | 11111739 | 5740.24 | 13 | 2.70 | 6810.35 |
| | 1300 | - | - | - | - | - | - | - | $> 5$h |
| | 1400 | 13.06 | 7 | 838 | 22358640 | 13415.77 | 13 | 2.82 | 14663.01 |
| | 1500 | - | - | - | - | - | - | - | $> 5$h |
| 2 | 120 | 17.62 | 42 | 6011 | 68987 | 381.66 | 333 | 78.52 | 1195.04 |
| | 130 | 14.14 | 33 | 4999 | 23254 | 229.55 | 259 | 61.68 | 907.80 |
| | 140 | 18.88 | 51 | 7681 | 523347 | 1973.46 | 409 | 92.08 | 3075.82 |
| | 150 | 21.72 | 57 | 7326 | 1036400 | 4532.09 | 445 | 99.27 | 5851.86 |
| | 160 | 21.60 | 51 | 6842 | 611097 | 2298.98 | 419 | 90.34 | 3529.53 |
| | 170 | 19.19 | 57 | 8405 | 631267 | 3209.97 | 452 | 98.60 | 4674.73 |
| | 180 | 20.06 | 56 | 7413 | 1236444 | 5500.25 | 445 | 97.70 | 7036.16 |
| | 190 | - | - | - | - | - | - | - | $> 5$h |
| 3 | 70 | 11.47 | 82 | 14678 | 212073 | 1792.73 | 776 | 192.66 | 2860.89 |
| | 80 | 12.63 | 89 | 12727 | 142892 | 1616.76 | 792 | 185.36 | 2848.11 |
| | 90 | 13.24 | 116 | 16904 | 1334989 | 12125.44 | 1026 | 234.02 | 13860.52 |
| | 100 | - | - | - | - | - | - | - | $> 5$h |
| 4 | 40 | 8.55 | 88 | 14878 | 15121 | 253.94 | 896 | 229.65 | 1023.19 |
| | 50 | 5.96 | 104 | 20137 | 61407 | 1085.66 | 1079 | 272.96 | 2138.60 |
| | 60 | 9.10 | 168 | 26601 | 1001795 | 10341.12 | 1769 | 426.89 | 12235.70 |
| | 70 | - | - | - | - | - | - | - | $> 5$h |

**Table 3** Results for different numbers $n$ of binary controls and different exponents $p$ with randomly distributed costs $c$. The number of added cuts (#cuts), the number of B&B nodes and the time for ILP solution, as well as the number of iterations and the time for the PDE solution are summed up over the major iterations. All times are CPU times in seconds.

It turns out that with randomly distributed costs problem (20) can be solved much more efficiently than with uniform costs, resulting in less computation time for the same number of binary controls $n$, e.g., 242.36s instead of 2504.50s ($p = 1, n = 250$) or 1023.19s instead of 1428.49s ($p = 4, n = 40$). On the one hand – which is the main factor here – the ILPs can be solved more efficiently. In fact, in discrete optimization, uniform objective functions often lead to harder problems in practice as they admit more feasible solutions with similar or equal objective function values. On the other hand, the outer approximation algorithm needs less major iterations compared to the uniform case, e.g., 42 instead of 67 (for $p = 2$ and $n = 120$) or 88 instead of 110 (for $p = 4$ and $n = 40$). In summary, we are able to solve problems with up to 1400 ($p = 1$), 180 ($p = 2$), 90 ($p = 3$) and 60 ($p = 4$) binary controls in this example.

### 9.2 Test Instances with Tracking-Type Objective

We next consider combinatorial optimal control problems with $L^\infty$-tracking-type objective functions; see Section 7. We use the same partial differential equation with the same partition of heat sources as in Section 9.1. As desired temperature we choose $y_d(x) = \frac{1}{2}(x_1 + x_2)S(1)$ for $x \in \Omega$, which creates a gradient temperature distribution. In case of the $L^\infty$-tracking type objective function, we can calculate a cut for all $x \in \Omega$, but as for the concave cuts in Section 9.1.2 we filter these options with the same parameter choices to reduce the number of cuts, making sure however that the point $x$ yielding the strongest bound on $z$ is taken into account. To obtain comparable figures for different exponents $p$, we scale the objective function by $\|S(1)\|_{L^\infty}$.

Table 4 presents the results for different number of heat sources $n$ and different exponents $p$. In contrast to the previous tables, we separate the time into the ILP solve time, the state $y$ computation time as well as the concave and submodular cut generation time. For $p = 1$ and $p = 2$, we can solve the problem up to $n = 500$ within the limit and for $p = 3$ and $p = 4$ up to $n = 400$. The dominating runtime factor here is the generation of the submodular cuts, since for every cut $n$ nonlinear PDEs have to be solved. The reoptimization strategies, which are even more beneficial here, cannot prevent this trend.

Nevertheless, for the nonlinear cases $p \in \{2, 3, 4\}$, much larger instances can be solved to optimality, compared to the previous results with only concave cuts. This is due to a slower increase in major iterations of the outer approximation algorithm, which in turn can be explained by the fact that the cutting planes derived from submodularity are much tighter than the tangent inequalities: when produced at a point $u^*$, the latter inequality touches the graph of $S(.)(x)$ only in $u^*$ in general, while the former touches the graph in $n + 1$ binary points by construction. Moreover, contrary to the tangent based cuts, the strength of submodularity-based cuts does not deteriorate with increasing exponent $p$.

| p | $n$ | obj | iter | #cuts | ILP time [s] | state time [s] | concave time [s] | submod. time [s] | overall time [s] |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 100 | 0.53 | 4 | 1219 | 1.49 | 0.88 | 45.41 | 99.30 | 152.06 |
| | 200 | 0.51 | 11 | 1264 | 11.28 | 4.78 | 303.79 | 800.47 | 1125.42 |
| | 300 | 0.49 | 18 | 1408 | 31.14 | 8.96 | 770.72 | 2334.60 | 3151.52 |
| | 400 | 0.47 | 31 | 1597 | 131.92 | 17.16 | 1810.17 | 6281.18 | 8247.64 |
| | 500 | 0.44 | 27 | 1703 | 507.70 | 15.42 | 1965.14 | 7003.79 | 9499.46 |
| | 600 | - | - | - | - | - | - | - | > 5h |
| 2 | 100 | 0.56 | 10 | 1862 | 10.20 | 18.58 | 131.51 | 1069.28 | 1235.60 |
| | 200 | 0.55 | 12 | 2166 | 32.13 | 23.70 | 323.65 | 2714.25 | 3101.05 |
| | 300 | 0.55 | 21 | 2276 | 67.59 | 44.88 | 891.39 | 6901.66 | 7913.19 |
| | 400 | 0.54 | 14 | 2286 | 93.87 | 27.22 | 765.99 | 6052.96 | 6948.04 |
| | 500 | 0.54 | 20 | 2354 | 73.91 | 39.12 | 1413.02 | 11145.32 | 12680.29 |
| | 600 | - | - | - | - | - | - | - | > 5h |
| 3 | 100 | 0.57 | 9 | 2046 | 8.97 | 17.14 | 114.69 | 1233.40 | 1380.75 |
| | 200 | 0.56 | 13 | 2618 | 33.29 | 29.22 | 349.35 | 3504.50 | 3924.20 |
| | 300 | 0.55 | 21 | 2550 | 80.07 | 49.64 | 884.31 | 8503.28 | 9526.40 |
| | 400 | 0.55 | 21 | 2591 | 114.90 | 46.57 | 1176.92 | 11120.18 | 12467.81 |
| | 500 | - | - | - | - | - | - | - | > 5h |
| 4 | 100 | 0.57 | 11 | 2666 | 13.34 | 30.92 | 143.36 | 1807.75 | 2003.85 |
| | 200 | 0.56 | 21 | 3148 | 67.70 | 57.46 | 583.70 | 6775.95 | 7494.99 |
| | 300 | 0.57 | 16 | 2698 | 98.37 | 41.38 | 652.75 | 7615.56 | 8417.17 |
| | 400 | 0.55 | 25 | 3122 | 264.74 | 70.24 | 1412.66 | 15582.11 | 17341.00 |
| | 500 | - | - | - | - | - | - | - | > 5h |

**Table 4** Results for different numbers $n$ of binary controls and different exponents $p$ for $L^\infty$-tracking type objective function. The number of added cuts (#cuts), the time for ILP solution, the PDE solution, the concave cuts and the submodular cuts are summed up over the major iterations. All times are CPU times in seconds.

### 9.3 Stationary Heating of a Metallic Workpiece

We finally show some results for the application mentioned in Section 2, i.e., the stationary heating of a metallic workpiece. We solve the problem in two dimensions, which requires to adapt the Boltzmann type radiation boundary condition to

$$\kappa \nabla y \cdot n + \sigma\, y^3 = \sigma\, y_0^3$$

with $\sigma = 1.92 \cdot 10^{-10}$ and $\kappa = 16$. The workpiece is given as $[0,1]^2$ and the heat sources correspond to 0.02 by 0.02 squares arranged on a $k \times k$-grid regularly covering the workpiece, each one equipped with a power of 2500 W. The surrounding temperature is chosen as 293 K.

In the first part, we consider uniform costs again, thus aiming at a minimal number of sources switched on. In Figure 5, optimal temperature distributions are depicted for $y_{\min} \equiv 1000\,\mathrm{K}$, for the cases $k = 5$ (12 sources needed, 10 iterations, 217 CPU seconds) and $k = 10$ (11/16/1793 s). Figure 6 shows optimal solutions for $k = 15$ with $y_{\min} \equiv 750\,\mathrm{K}$ (5/10/1076 s) and $y_{\min} \equiv 1250\,\mathrm{K}$

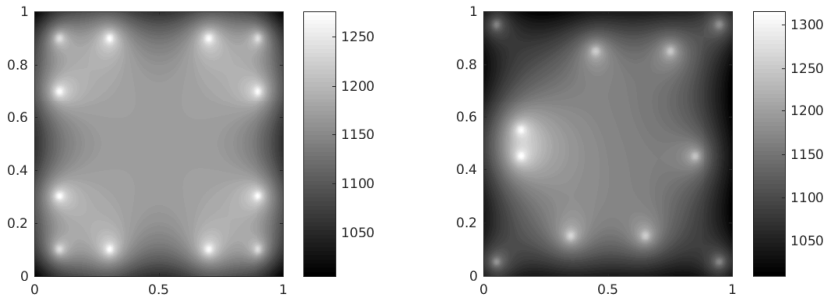


**Fig. 5** Optimal states for $k = 5$ (left) and $k = 10$ (right) with $y_{\min} \equiv 1000\,\mathrm{K}$.

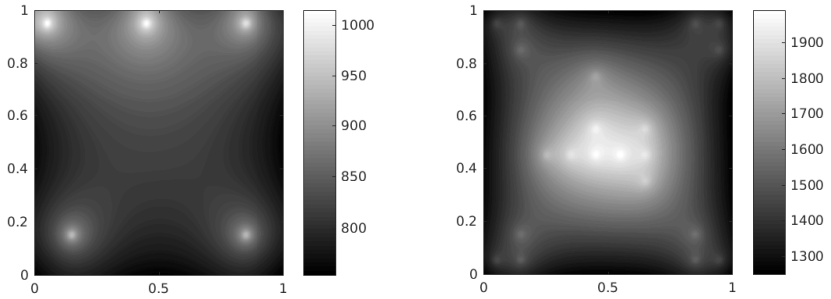


**Fig. 6** Optimal states for $k = 10$ with $y_{\min} \equiv 750\,\mathrm{K}$ (left) and $y_{\min} \equiv 1250\,\mathrm{K}$ (right).

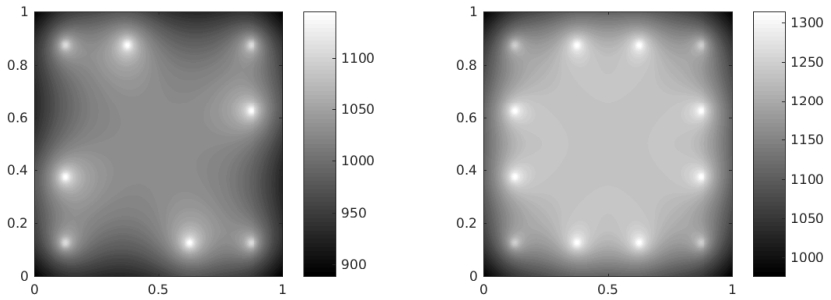


**Fig. 7** Optimal states for $k = 4$ with $y_d \equiv 1000\,\mathrm{K}$ (left) and $y_d \equiv 1250\,\mathrm{K}$ (right).

(21/19/2262 s). Note that an optimal solution may be symmetric, but does not necessarily have to be. Moreover, in the case of uniform costs, there are many different optimal solutions in general.

In the second part, we consider an $L^1$-tracking type objective function. In Figure 7, optimal temperature distributions are shown for $k = 4$ and a desired temperature of $y_d \equiv 1000\,\mathrm{K}$ (8/572/27895 s) and $y_d \equiv 1250\,\mathrm{K}$ (12/197/11132 s).

## 10 Conclusions and Future Directions

We have presented an outer approximation approach for solving a large class of semilinear elliptic optimal control problems with static combinatorial controls, yielding optimal solutions in finitely many iterations. The algorithm exploits the pointwise concavity and submodularity of the solution operator in terms of the control variables in order to generate valid linear cutting planes.

Some ideas of the algorithm can be extended to the case of mixed-integer controls. Indeed, the cutting planes derived from concavity in Section 3 are valid globally, so that lower bounds on the states can be handled as before. However, in the presence of continuous control variables, we cannot expect finite convergence anymore, and more care is needed in the selection of cutting planes to ensure any kind of convergence.

In fact, also the cutting planes derived from submodularity in Section 5 remain valid for all $u \in [0,1]^n$, this follows from submodularity and concavity together. But, unfortunately, it is not true any more that every infeasible control vector can be cut off by an appropriate cutting plane. Consequently, we cannot easily extend this part of our approach to mixed-integer problems. This reflects the fact that non-convexity is problematic for continuous variables, while for binary variables the situation is rescued by submodularity.

As observed in our experiments, most of our algorithm's running time is spent for solving ILPs, particularly for larger instances. A significant speed-up can thus be expected from a more sophisticated solution strategy for these ILPs, exploiting the iterative structure of the algorithm again. For this, one could use general ideas discussed for outer approximation algorithms such as branch-and-cut-based outer approximation [32,8]. Our convexity result could also be exploited in many other ways, e.g., we could use an iterative method for nonlinear optimization using first-order information in order to compute a local (and hence global) optimizer. However, this would only solve the continuous relaxation of our problem, so that such an approach needs to be embedded into a branch-and-bound scheme again. Moreover, submodularity cannot be integrated easily into such an approach.

Finally, we would like to mention that we can also deal with pointwise concave (instead of convex) functions $g$ and $b$. In this case, the solution operator $S$ is pointwise convex (instead of concave) and supermodular (instead of submodular). Our approach can thus be applied in a symmetric way.

## References

1. J.-J. Alibert and J.-P. Raymond, *Boundary control of semilinear elliptic equations with discontinuous leading coefficients and unbounded controls*, Numerical Functional Analysis and Optimization, 18 (1997), pp. 235–250.
2. M. Avraam, N. Shah, and C. Pantelides, *Modelling and optimisation of general hybrid systems in the continuous time domain*, Computers & Chemical Engineering, 22, Supplement 1 (1998), pp. S221–S228.

3. S. Balakrishna and L. T. Biegler, *A unified approach for the simultaneous synthesis of reaction, energy, and separation systems*, Industrial & Engineering Chemistry Research, 32 (1993), pp. 1372–1382.
4. V. Bansal, V. Sakizlis, R. Ross, J. D. Perkins, and E. N. Pistikopoulos, *New algorithms for mixed-integer dynamic optimization*, Computers & Chemical Engineering, 27 (2003), pp. 647–668.
5. F. Baumann, S. Berckey, and C. Buchheim, *Facets of Combinatorial Optimization – Festschrift for Martin Grötschel*, Springer-Verlag, 2013, ch. Exact Algorithms for Combinatorial Optimization Problems with Submodular Objective Functions, pp. 271–294.
6. P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, *Mixed-integer nonlinear optimization*, Acta Numerica, 22 (2013), pp. 1–131.
7. T. J. Boehme, M. Schori, B. Frank, M. Schultalbers, and B. Lampe, *Solution of a hybrid optimal control problem for parallel hybrid vehicles subject to thermal constraints*, in 2013 IEEE 52nd Annual Conference on Decision and Control (CDC), IEEE, 2013, pp. 2220–2226.
8. P. Bonami, L. Biegler, A. Conn, G. Cornuéjols, I. Grossmann, C. Laird, J. Lee, A. Lodi, F. Margot, N. Sawaya, and A. Wächter, *An algorithmic framework for convex mixed integer nonlinear programs*, Discrete Optimization, 5 (2008), pp. 186–204.
9. E. Casas, *Control of an elliptic problem with pointwise state constraints*, SIAM Journal on Control and Optimization, 24 (1986), pp. 1309–1318.
10. ———, *Boundary control of semilinear elliptic equations with pointwise state constraints*, SIAM Journal on Control and Optimization, 31 (1993), pp. 993–1006.
11. R. Chandra, *Partial differential equations constrained combinatorial optimization on an adiabatic quantum computer*, master's thesis, Purdue University, 2013.
12. K. Deckelnick and M. Hinze, *Convergence of a finite element approximation to a state constrained elliptic control problem*, SIAM Journal on Numerical Analysis, 45 (2007), pp. 1937–1953.
13. V. D. Dimitriadis and E. N. Pistikopoulos, *Flexibility analysis of dynamic systems*, Industrial & Engineering Chemistry Research, 34 (1995), pp. 4451–4462.
14. M. A. Duran and I. E. Grossmann, *An outer-approximation algorithm for a class of mixed-integer nonlinear programs*, Mathematical Programming, 36 (1986), pp. 307–339.
15. J. Edmonds, *Submodular functions, matroids, and certain polyhedra*, in Combinatorial Optimization – Eureka, You Shrink!, M. Jünger, G. Reinelt, and G. Rinaldi, eds., vol. 2570 of LNCS, Springer, 2003, pp. 11–26.
16. A. Fügenschuh, B. Geissler, A. Martin, and A. Morsi, *The transport PDE and mixed-integer linear programming*, in Models and Algorithms for Optimization in Logistics, C. Barnhart, U. Clausen, U. Lauther, and R. H. Möhring, eds., no. 09261 in Dagstuhl Seminar Proceedings, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany, 2009.
17. S. Fujishige, *Submodular functions and optimization*, Annals of Discrete Mathematics, Elsevier, 1991.
18. B. Geissler, O. Kolb, J. Lang, G. Leugering, A. Martin, and A. Morsi, *Mixed integer linear models for the optimization of dynamical transport networks*, Mathematical Methods of Operations Research, 73 (2011), pp. 339–362.
19. M. Gerdts, *A variable time transformation method for mixed-integer optimal control problems*, Optimal Control Applications and Methods, 27 (2006), pp. 169–182.
20. P. Grisvard, *Elliptic Problems in Nonsmooth Domains*, Classics in Applied Mathematics, SIAM, Philadelphia, 1985.
21. R. Haller-Dintelmann, C. Meyer, J. Rehberg, and A. Schiela, *Hölder continuity and optimal control for nonsmooth elliptic problems*, Applied Mathematics and Optimization, 60 (2009), pp. 397–428.
22. F. M. Hante and S. Sager, *Relaxation methods for mixed-integer optimal control of partial differential equations*, Computational Optimization and Applications, 55 (2013), pp. 197–225.
23. M. Hintermüller and K. Kunisch, *PDE-constrained optimization subject to pointwise constraints on the control, the state, and its derivative*, SIAM Journal on Optimization, 20 (2009), pp. 1133–1156.

24. F. Incropera and D. D. Witt, *Fundamentals of Heat and Mass Transfer*, Wiley, Chichester, 1985.
25. D. Kinderlehrer and G. Stampacchia, *An introduction to variational inequalities and their applications*, vol. 31, SIAM, 2000.
26. C. Kirches and F. Lenders, *Approximation properties and tight bounds for constrained mixed-integer optimal control*, tech. rep., Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, 2016.
27. C. Kirches, S. Sager, H. G. Bock, and J. P. Schlöder, *Time-optimal control of automobile test drives with gear shifts*, Optimal Control Applications and Methods, 31 (2010), pp. 137–153.
28. L. Lovász, *Submodular functions and convexity*, in Mathematical programming: the state of the art (Bonn, 1982), Springer, Berlin, 1983, pp. 235–257.
29. C. Meyer, *Error estimates for the finite-element approximation of an elliptic control problem with pointwise state and control constraints*, Control and Cybernetics, 37 (2008), pp. 51–85.
30. C. Meyer, U. Prüfert, and F. Tröltzsch, *On two numerical methods for state-constrained elliptic control problems*, Optimization Methods and Software, 22 (2007), pp. 871–899.
31. M. J. Mohideen, J. D. Perkins, and E. N. Pistikopoulos, *Optimal design of dynamic systems under uncertainty*, AIChE Journal, 42 (1996), pp. 2251–2272.
32. I. Quesada and I. Grossmann, *An LP/NLP based branched and bound algorithm for convex MINLP optimization problems*, Computers and Chemical Engineering, 16 (1992), pp. 937–947.
33. S. Sager, H. Bock, and M. Diehl, *The integer approximation error in mixed-integer optimal control*, Mathematical Programming, 133 (2012), pp. 1–23.
34. S. Sager, M. Jung, and C. Kirches, *Combinatorial integral approximation*, Mathematical Methods of Operations Research, 73 (2011), pp. 363–380.
35. A. Schiela and W. Wollner, *Barrier methods for optimal control problems with convex nonlinear gradient state constraints*, SIAM J. Optim, 21 (2011), pp. 269–286.
36. J. Till, S. Engell, S. Panek, and O. Stursberg, *Applied hybrid system optimization: An empirical investigation of complexity*, Control Engineering Practice, 12 (2004), pp. 1291–1303.
37. F. Tröltzsch, *Optimal Control of Partial Differential Equations: Theory, Methods, and Applications*, Graduate studies in mathematics, American Mathematical Society, 2010.
38. O. von Stryk and M. Glocker, *Decomposition of mixed-integer optimal control problems using branch and bound and sparse direct collocation*, in Proc. ADPM 2000 – The 4th International Conference on Automation of Mixed Processes: Hybrid Dynamic Systems, S. Engell, S. Kowalewski, and J. Zaytoon, eds., Dortmund, sep 2000, pp. 99–104.
39. P. Zhang, D. Romero, J. Beck, and C. Amon, *Solving wind farm layout optimization with mixed integer programming and constraint programming*, in Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems, C. Gomes and M. Sellmann, eds., vol. 7874 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2013, pp. 284–299.