DFG Deutsche Forschungsgemeinschaft
Priority Programme 1962

Numerical Solution of Optimal Control Problems with Switches, Switching Costs and Jumps

Christian Kirches, Ekaterina Kostina, Andreas Meyer, Matthias Schlöder

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

Preprint Number SPP1962-109

received on March 6, 2019

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/

Numerical Solution of Optimal Control Problems with Switches, Switching Costs and Jumps

Christian Kirches[1], Ekaterina A. Kostina[2], Andreas Meyer[3] and Matthias Schlöder*[2]

[1]Institute for Mathematical Optimization, Technische Universität Braunschweig, Germany
[2]Institute for Applied Mathematics, Heidelberg University, Germany
[3]Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Germany

March 5, 2019

Abstract

In this article, we present a framework for the numerical solution of optimal control problems, constrained by ordinary differential equations which can run in (finitely many) different modes, where a change of modes leads to additional switch cost in the cost function, and whenever the system changes its mode, jumps in the differential states are possible. In addition, for each mode there are certain constraints which shall only hold as long as the system stays in the respective mode. We present the problem class and represent the problem as a mixed-integer optimal control problem. We reformulate and relax the problem and discretize the control functions in the resulting problem. We present three different approaches for the treatment of switch costs and compare them with each other, whereat only one of them is suitable for the treatment of jumps in a general setting. We then take a direct approach ("first discretize, then optimize") to solve the resulting control-discretized problem numerically, where a direct method based on *hp*–adaptive collocation is used for the discretization. The resulting finite dimensional optimization problems are mathematical programs with vanishing constraints, and we suggest a numerical approach to solve sequences of this challenging problem class. In the end of the article, we present two examples: first an academic one concerning switch costs only, and second an example from mechanics, where also jumps occur. In the latter example, we generate a walking-like motion and discuss the modeling as well as the problem-specific configuration of our solution approach in detail.

Keywords: switched systems, switch costs, jumps in differential states, optimal control, mixed–integer optimal control, direct transcription methods, mathematical programs with vanishing constraints, walking–like motion

1 Introduction

Switched dynamic systems, which are a particular class of hybrid dynamic systems, have been extensively researched over the past decades and a lot of progress has been made in this field of applied mathematics both theoretically and computationally, cf. [39, 2, 17, 11, 36]. Recently, Meyer et al. [8] proposed a new approach for solving a class of switched dynamic systems in a numerically efficient way, based on generalized disjunctive programming, a direct approach to optimal control, and on solving the resulting Mathematical Programs with Vanishing Constraints

*Corresponding author. Email: schloeder@stud.uni-heidelberg.de

(MPVCs). However, their framework does not include switch costs and jumps in the differential states. In this article, we augment their approach in view of these aspects.
Hybrid systems are dynamic systems that involve continuous models as well as discrete event models. Applications of hybrid systems arise, amongst others, in the fields of industrial process control, power systems, and traffic control. Zhu and Antsaklis [39] provide a detailed survey on the topic. We are in particular interested in a medical application, namely model–based treatment planning of patients suffering from Cerebral Palsy (CP). These patients show a pathological gait, with common symptoms being internal rotation and so-called *pes equinus*, meaning the heel never touches the ground while walking. By applying orthopedic changes to the musculoskeletal system of patients, medical doctors aim at ameliorating this situation. Despite the impressive progress in this field [3], it is still challenging to predict the precise effects of interventions. Model-based treatment planning strives to design a computational testing environment for ex ante evaluation and assessment of potential surgery plans. In view of this application and phenomena like pes equinus, it is important *not* to assume a predefined order of modes-stages, but allow for changes which may result from a treatment.
In the context of this article, switched systems are Optimal Control Problems (OCPs) with possible discontinuities in the differential equations right hand side as well as in the differential states. In particular, all subsystems live in the same state space. Switched systems can be represented by an indexed set of differential equations

$$\dot{\mathbf{x}}(t) = \mathbf{F}^{w(t)}(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(t_0) = \mathbf{x}_0$$

and jump conditions

$$\mathbf{x}\left(t^+\right) = \boldsymbol{\Delta}_{w(t^-),w(t^+)}(\mathbf{x}(t^-))$$

which map the differential states before a switch $\mathbf{x}\left(t^-\right) = \lim_{\tau \nearrow t} \mathbf{x}(\tau)$ to the differential states after a switch $\mathbf{x}\left(t^+\right) = \lim_{\tau \searrow t} \mathbf{x}(\tau)$. At any point on the time horizon $\mathcal{T} = [t_0, t_f]$, a function $w : \mathcal{T} \to \{1, \ldots, n\}$ indicates the index of the applicable dynamic right hand side, and whenever this index changes from j_1 to j_2, a jump function $\boldsymbol{\Delta}_{j_1,j_2}(\cdot)$ specifically belonging to the (ordered) pair (j_1, j_2) acts on the differential states. In addition, each change of indices causes a contribution to the cost function.
Assuming that only a finite number of switching events occurs, the above switching function $w(\cdot)$ may be identified with a finite vector of tuples $s = [(t_0, j_0), (t_1, j_1), \ldots, (t_m, j_m)]$, where $0 \leq m < \infty$, $j_i \in \{1, \ldots, n\}$ for all $i = 0, \ldots, m$ and $t_0 \leq t_1 \leq \ldots \leq t_m \leq t_f$. Thus, the switching function is determined by the *switching sequence* $\sigma = \{j_i\}_{i=0}^m$ and the associated *switching times* $\mathcal{S}(w) = \{t_i\}_{i=1}^m$.
In generally one can distinguish two kinds of switches: Externally Forced Switches (EFSs) and Internally Forced Switches (IFSs). EFSs are also known as *controllable* or *explicit switches*. For problems involving EFSs the switchings are degrees of freedom. Conversely, switches in IFS problems depend on the states $\mathbf{x}(\cdot)$ and the current mode j. IFS systems arise for instance from ground contact of a robot leg or from a weir overflow of a distillation column. *Implicit switch* is another well known term for IFS. Most of the literature does not address combined EFS and IFS problems. However, Meyer et al. [8] handle EFS and IFS problems (without switch costs and jumps) in a unified framework, by combining the ideas of complementarity based formulations for EFS systems developed by Baumrucker and Biegler [4] with the idea of *embedding transformation*. Embedding transformation, developed independently by Sager [34] and Bengea and DeCarlo [5] for EFS systems, reformulates the switched dynamic system into the larger family of continuous systems

$$\dot{\mathbf{x}}(t) = \sum_{j=1}^{n} \boldsymbol{\alpha}_j(t)\, \mathbf{F}^j(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(t_0) = \mathbf{x}_0,$$

where $\boldsymbol{\alpha}_j(t) \in [0, 1]$ and $\sum_{j=1}^n \boldsymbol{\alpha}_j(t) = 1$. At the beginning of the optimization no assumptions about the number of switches, the switching sequence σ and the switching times $\mathcal{S}(w)$ are necessary. OCPs constrained by ordinary differential equations with implicitly defined, state-dependent discontinuities are notoriously difficult to solve. A common approach is to combine a modern simultaneous optimization method, e.g. Direct Multiple Shooting [9], with an appropriate switch detecting differential equation solver, see e.g. [7, 10, 25]. The main challenges of switch detecting solvers are the determination of the switching time and the sensitivity update process at discontinuities, cf.

[10, 25, 33, 20, 28]. Models where both the number and the sequence of arising switching points are known can be handled by multi-phase OCP, as in [35, 19]. In this case the implicit discontinuities do not have to be treated explicitly. Anyway, in many applications this knowledge is not available. Switch costs penalize switches by an additional term in the objective function. Kirches [26] and Jung [24] present approaches how to treat switch costs in a discretized context, meaning that there is a time grid $\mathbb{G} = \{t_i\}_{i=0}^{N} \subset \mathcal{T}$ and parameters $\mathbf{q}_j^i \in [0,1]$, such that

$$\boldsymbol{\alpha}_j(t) = \mathbf{q}_j^i \text{ for } t \in [t_i, t_{i+1})$$

for $i = 0, \ldots, N-2$ and $\boldsymbol{\alpha}_j(t) = \mathbf{q}_j^{N-1}$ for $t \in [t_{N-1}, t_N]$. Kirches [26] proposes to overestimate the number of switches in a discretized and relaxed problem. However, the stated approach suffers from the drawback, that it is only able to detect the modes j_1 and j_2 which are involved in a switch, but not their ordering, namely if the system switches from mode j_1 to j_2 or the other way around.

1.1 Contributions

In this article, we augment the method presented in [8] in view of switch costs and jumps in the differential states, and thus present a novel approach for the solution of OCPs with switch costs and jumps, where the number and order of model-stages is a priori unknown and is determined dynamically. This methodology is a step towards model-based treatment planning of CP, where the number and order of occurring model phased during the gait cycle may change after an intervention. Furthermore, we present two novel approaches for the treatment of switch costs in the context of OCPs, and compare them to an existing method.
We start with an Mixed-Integer Optimal Control Problem (MIOCP), to which we apply Partial Outer Convexification (POC) [34]. We introduce additional binary indicator functions similar to ideas reported [26] in the context of switch costs, and investigate and compare different tractable formulations for the numerical treatment of switch costs. The introduced binary indicators to can then be used to convexify the jump condition as well, leading to a new relaxation of the partially convexified MIOCP.

1.2 Structure

In Section 2 we introduce the class of switched OCPs with switch costs and jumps and represent a problem of this class as an MIOCP. In Section 3 we reformulate and relax the problem, and finally discretize the control functions in the resulting problem. Section 4 is dedicated to different approaches for the handling of switch costs in the context of OCPs and the comparison of those. In Section 5, we use a direct transcription method to transfer the relaxed and control-discretized OCP from Section 3 into an Nonlinear Programming Problem (NLP) belonging to the class of MPVCs. Section 6 then deals with the numerical treatment of this special type of NLPs. We demonstrate the merit of our approach in Section 7, where we generate a walking-like motion of the so-called simplest walker stick-man model, for which we present the Multi-Body System (MBS), the switched OCP and numerical results computed using a direct and all-at-once approach.

2 Problem formulation

In this section we describe the class of OCPs we are interested in and show, how a problem belonging to this class can be reformulated as aMIOCP.

2.1 OCPs with Switches, Switch Costs and Jumps

We take interest in OCPs, where the underlying dynamics can run in a finite number of different modes. Whenever the dynamic changes its mode, jumps in the differential states are possible. To state the problem we consider in this article, we first introduce some notation.
We consider the time horizon $\mathcal{T} = [t_0, t_f]$, where both t_0 and t_f are fixed without loss of generality. The dynamical system $\mathbf{x} \in W^{1,\infty}(\mathcal{T}, \mathbb{R}^{n_x})$ we deal with can run in the n different modes $\{1, \ldots, n\}$.

For every $t \in \mathcal{T}$, the mode our system runs in is reflected by the value of a control function $w : \mathcal{T} \to \{1, \dots, n\}$ such that

$$\text{System is in mode } j \text{ at time } t \iff w(t) = j\,.$$

We assume that the following assumption holds:

Assumption 2.1 (Strictly Positive Dwell Time) *The considered system has a strictly positive dwell time $\bar{\delta}$, i.e. the system does not change its mode in $[t_0, t_0 + \bar{\delta})$ and whenever the system changes its mode at a time point t_s, it stays in the respective mode for at least all $t \in (t_s, t_s + \bar{\delta}) \subseteq \mathcal{T}$.*

For $\mathcal{M} \subseteq \mathbb{R}^k$ we define

$$PC_{\bar{\delta}}(\mathcal{T}, \mathcal{M}) \overset{def}{=} \left\{ \boldsymbol{\omega} : \mathcal{T} \to \mathbb{R}^k \;\middle|\; \begin{array}{l} \bullet\ \forall t \in \mathcal{T} \setminus \{t_f\}\ \exists \tau_1, \tau_2 \in \mathcal{T} : \tau_2 - \tau_1 \geq \bar{\delta}, t \in [\tau_1, \tau_2) \\ \quad \text{and } \boldsymbol{\omega}(t) = \boldsymbol{\omega}(\tau_1)\ \forall t \in [\tau_1, \tau_2) \\ \bullet\ \boldsymbol{\omega}(t_f) = \boldsymbol{\omega}(t_f - \bar{\delta}) \\ \bullet\ \boldsymbol{\omega}(t) \in \mathcal{M}\ \forall t \in \mathcal{T} \end{array} \right\},$$

which are the *right–continuous piecewise constant functions* on $\mathcal{T}$ with values in $\mathcal{M}$ and dwell time $\bar{\delta}$. Because of Assumption 2.1 we demand $w \in PC_{\bar{\delta}}(\mathcal{T}, \{1, ..., n\})$ in the following. Here, the right–continuity is a choice we make without loss of generality.
For any right-continuous function $\mathbf{g} : \mathcal{T} \to \mathbb{R}^k$, for which also the left-hand side limits $\mathbf{g}(t^-) = \lim_{\tau \nearrow t_s} \mathbf{g}(\tau)$ exist for all $t \in \mathcal{T} \setminus \{t_0\}$, we define

$$\mathcal{S}(\mathbf{g}) = \left\{ t_s \in \mathcal{T} \setminus \{t_0\} \mid \mathbf{g}(t_s^-) \neq \mathbf{g}(t_s) \right\}.$$

Due to Assumption 2.1, $t_f \notin \mathcal{S}(w)$ and the set $\mathcal{S}(w)$ is finite. We denote its cardinality by $|\mathcal{S}(w)|$. The elements of $\mathcal{S}(w)$ are called *switching points* since the system's mode changes at these time points, and a change of modes is called a *switch. Instead of saying 'the system switches from mode j_1 to mode j_2', we simply write '$j_1 \to_w j_2$',* where the subscript emphasizes the dependency of the system's mode on the control function $w(\cdot)$.
When the considered dynamical system is in mode $j \in \{1, \dots, n\}$, it is governed by the – without loss of generality autonomous – Ordinary Differential Equation (ODE)

$$\dot{\mathbf{x}}(t) = \mathbf{f}^j(\mathbf{x}(t), \mathbf{u}(t))\,,$$

where $\mathbf{u} \in L^\infty(\mathcal{T}, \mathbb{R}^{n_u})$ is a control function. Whenever the system changes its mode – that means $w(\cdot)$ changes its value – at a switching point t_s, jumps in the differential states may occur. By

$$\mathbf{x}(t_s^-) = \lim_{\tau \nearrow t_s} \mathbf{x}(\tau) \quad \text{resp.} \quad \mathbf{x}(t_s^+) = \lim_{\tau \searrow t_s} \mathbf{x}(\tau)$$

we denote the value of the differential states before the jump resp. after the jump. We suppose that for every ordered pair $(j_1, j_2) \in \{1, \dots, n\}^2$ with $j_1 \neq j_2$, there is a function $\boldsymbol{\Delta}_{j_1, j_2} : \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$, mapping the differential states before the jump to the differential states after the jump:

$$\mathbf{x}(t_s^+) = \boldsymbol{\Delta}_{j_1, j_2}\left(\mathbf{x}(t_s^-)\right) \quad \text{if } j_1 \to_w j_2\,.$$

During the whole process, path constraints $\mathbf{0} \geq \mathbf{d}(\mathbf{x}(t), \mathbf{u}(t))$ must be satisfied, where $\mathbf{d} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_d}$, $\mathbf{0}$ is the zero vector of appropriate size, and all inequalities shall hold component-wise. Additionally, for each mode j there are path constraints $\mathbf{0} \geq \mathbf{c}^j(\mathbf{x}(t), \mathbf{u}(t))$ with $\mathbf{c}^j : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_{c_j}}$, which are only required to hold at time t if the system runs in the respective mode at time t. In practice the latter constraints can be used e. g. to determine the mode of the system. In addition, point constraints $\mathbf{0} \geq \mathbf{r}(\mathbf{x}(t_0), \mathbf{x}(t_f))$ with $\mathbf{r} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \to \mathbb{R}^{n_r}$ shall hold.
We set up an OCP to find controls $\mathbf{u}(\cdot) \in L^\infty(\mathcal{T}, \mathbb{R}^{n_u})$ as well as $w(\cdot) \in PC_{\bar{\delta}}(\mathcal{T}, \{1, ..., n\})$, which result in a dynamical process $\mathbf{x}(\cdot) \in W^{1,\infty}(\mathcal{T}, \mathbb{R}^{n_x})$, that satisfies all mentioned constraints and minimizes the value of a cost function. This cost function is built up by two contributions: The first contribution is without loss of generality given by a Mayer–term $\phi(\mathbf{x}(t_f))$ with $\phi : \mathbb{R}^{n_x} \to \mathbb{R}$, $\mathbf{x} \mapsto \mathbf{x}(t_f)$, and the second contribution is given by the (finite) number of switching points $|\mathcal{S}(w)|$, multiplied by a penalization parameter $\pi \geq 0$. We denote the second contribution by the term *switch costs.*

The resulting OCP we consider takes the following form:

$$\begin{aligned}
\min_{\mathbf{x}(\cdot),\mathbf{u}(\cdot),w(\cdot)} \quad & \phi(\mathbf{x}(t_f)) + \pi\,|\mathcal{S}(w)| & & \\
\text{s.t.} \quad & (\mathbf{x},\mathbf{u},w) \in W^{1,\infty}(\mathcal{T},\mathbb{R}^{n_x}) \times L^{\infty}(\mathcal{T},\mathbb{R}^{n_u}) \times PC_{\bar\delta}(\mathcal{T},\{1,...,n\}) & & \text{(1a)}\\
& \dot{\mathbf{x}}(t) = \mathbf{f}^j(\mathbf{x}(t),\mathbf{u}(t)) & \text{if } w(t)=j \text{ a.e. } t\in\mathcal{T} & \text{(1b)}\\
& \mathbf{x}(t_s^+) = \boldsymbol{\Delta}_{j_1,j_2}\left(\mathbf{x}(t_s^-)\right) & \text{if } j_1 \to_w j_2 \text{ at } t_s \in \mathcal{S}(w) & \text{(1c)}\\
& \mathbf{0} \geq \mathbf{c}^j(\mathbf{x}(t),\mathbf{u}(t)) & \text{if } w(t)=j \text{ a.e. } t\in\mathcal{T} & \text{(1d)}\\
& \mathbf{0} \geq \mathbf{d}(\mathbf{x}(t),\mathbf{u}(t)) & \text{a.e. } t\in\mathcal{T} & \text{(1e)}\\
& \mathbf{0} \geq \mathbf{r}(\mathbf{x}(t_0),\mathbf{x}(t_f)) & & \text{(1f)}
\end{aligned}$$

where we suppose, that the occurring functions are sufficiently smooth for our purposes. We make some remarks regarding problem (1):

- For numerical computations, the demanding for a dwell time $\bar\delta > 0$ is not restrictive, as we can imagine $\bar\delta$ to be the maximum possible granularity of the time grid.
- Though in the presented problem formulation switches arise explicitly from a change of values of the control function $w(\cdot)$, also systems with implicitly *and* explicitly forced switches can be treated using the above problem formulation, cf. [8].
- Consequently, by setting $\pi = 0$ and $\boldsymbol{\Delta}_{j_1,j_2}(\cdot) = \mathbf{Id}(\cdot)$ for all $(j_1,j_2) \in \{1,\dots,n\}^2$ with $j_1 \neq j_2$, the presented problem formulation also covers switched systems (with explicit *and* implicit switches) without switch costs and jumps as treated by Meyer et al. [8]. Hence the current framework can be seen as an extension of the framework presented in this reference.

For the sake of a handy presentation, we omit writing the function spaces for $\mathbf{x}(\cdot)$ *and* $\mathbf{u}(\cdot)$ *as well as the constraints (1e) and (1f) in the following, though we keep them in mind.*

2.2 A Mixed-Integer Optimal Control Problem

We state a MIOCP, which is equivalent to Problem (1). In order to do so, we define

$$S^n_{\mathcal{F}} \overset{def}{=} \left\{ \boldsymbol{\omega} : \mathcal{T} \to [0,1]^n \;\middle|\; \textstyle\sum_{j=1}^n \boldsymbol{\omega}_j(t) = 1 \;\forall t \in \mathcal{T} \right\}$$

and consider the mapping

$$\varphi :\, S^n_{\mathcal{F}} \cap PC_{\bar\delta}(\mathcal{T},\{0,1\}^n) \longrightarrow PC_{\bar\delta}(\mathcal{T},\{1,...,n\})\ ,\ \boldsymbol{\omega}(\cdot) \longmapsto w(t) \overset{def}{=} \sum_{j=1}^n \boldsymbol{\omega}_j(t)\cdot j\ .$$

Lemma 2.2 *The mapping* φ *is a bijection.*

Proof See Appendix A.1 □

For a $\boldsymbol{\omega} \in S^n_{\mathcal{F}} \cap PC_{\bar\delta}(\mathcal{T},\{0,1\}^n)$, we set

$$j_1 \to_{\boldsymbol{\omega}} j_2 \iff j_1 \to_{\varphi(\boldsymbol{\omega})} j_2\ ,$$

and consider the following MIOCP:

$$\begin{aligned}
\min_{\mathbf{x}(\cdot),\mathbf{u}(\cdot),\boldsymbol{\omega}(\cdot)} \quad & \phi(\mathbf{x}(t_f)) + \pi\,|\mathcal{S}(\boldsymbol{\omega})| & & \\
\text{s.t.} \quad & \boldsymbol{\omega} \in S^n_{\mathcal{F}} \cap PC_{\bar\delta}(\mathcal{T},\{0,1\}^n) & & \text{(2a)}\\
& \dot{\mathbf{x}}(t) = \textstyle\sum_{j=1}^n \boldsymbol{\omega}_j(t)\cdot \mathbf{f}^j(\mathbf{x}(t),\mathbf{u}(t)) & \text{a.e. } t\in\mathcal{T} & \text{(2b)}\\
& \mathbf{x}(t_s^+) = \boldsymbol{\Delta}_{j_1,j_2}\left(\mathbf{x}(t_s^-)\right) & \text{if } j_1 \to_{\boldsymbol{\omega}} j_2 \text{ at } t_s \in \mathcal{S}(\boldsymbol{\omega}) & \text{(2c)}\\
& \mathbf{0} \geq \boldsymbol{\omega}_j(t)\cdot \mathbf{c}^j(\mathbf{x}(t),\mathbf{u}(t)) & \text{a.e. } t\in\mathcal{T}\ \forall j & \text{(2d)}
\end{aligned}$$

We have

Proposition 2.3 *Problem (2) and Problem (1) are equivalent in the following sense:* $(\mathbf{x},\mathbf{u},w)$ *is feasible for Problem (1) if and only if* $(\mathbf{x},\mathbf{u},\varphi^{-1}(w))$ *is feasible for Problem (2), and the values of the according cost functions coincide.*

Proof See Appendix A.2. □

3 Problem Reformulation

In this section we reformulate and relax Problem (2) using convexification techniques. In the end we discretize the controls in the resulting problem.

3.1 Reformulation and Relaxation

Let $\mathcal{P} \subset \mathcal{T} \setminus \{t_0, t_f\}$ be an arbitrary finite subset. For a given $\boldsymbol{\omega} \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, [0,1]^n)$, we define

$$V_{\mathcal{S}(\boldsymbol{\omega})\cup\mathcal{P}}(\mathcal{T}) = \{g : \mathcal{T} \to [0,1] \,|\, g(t) = 0 \text{ for } t \notin \mathcal{S}(w) \cup \mathcal{P}\},$$

which is the set of functions in $S^1_{\mathcal{F}}$ which vanish outside $\mathcal{S}(w) \cup \mathcal{P}$.
We consider the number of switching points for a given control function $\boldsymbol{\omega} \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, \{0,1\}^n)$. For every pair $(j_1, j_2) \in \{1, \dots, n\}^2$ with $j_1 \neq j_2$ we define a function $\boldsymbol{\theta}_{j_1,j_2} : \mathcal{T} \to \{0,1\}$ by

$$\boldsymbol{\theta}_{j_1,j_2}(t) = \begin{cases} \min\left(\boldsymbol{\omega}_{j_1}(t^-), \boldsymbol{\omega}_{j_2}(t^+)\right) & \text{if } t_0 < t < t_f \\ 0 & \text{else} \end{cases}.$$

Then

$$\boldsymbol{\theta}_{j_1,j_2}(t) = \begin{cases} 1 & \text{if } j_1 \to_{\boldsymbol{\omega}} j_2 \text{ at } t \\ 0 & \text{else} \end{cases},$$

which is why we call these functions *switching indicator functions*. We have $\boldsymbol{\theta}_{j_1,j_2} \in V_{\mathcal{S}(\boldsymbol{\omega})\cup\mathcal{P}}(\mathcal{T})$, and is easy to see that

$$|\mathcal{S}(\boldsymbol{\omega})| = \sum_{t \in \mathcal{S}(\boldsymbol{\omega})\cup\mathcal{P}} \sum_{\substack{j_1,j_2=1 \\ j_1 \neq j_2}}^{n} \boldsymbol{\theta}_{j_1,j_2}(t). \tag{3}$$

We define the aggregated jump function $\boldsymbol{\Delta} : \mathbb{R}^{n_x} \times [0,1]^{n\cdot(n-1)} \to \mathbb{R}^{n_x}$ by

$$\boldsymbol{\Delta}\left(\mathbf{z}, (\mathbf{a}_{j_1,j_2})_{j_1\neq j_2}\right) = \sum_{\substack{j_1,j_2=1 \\ j_1 \neq j_2}}^{n} \mathbf{a}_{j_1,j_2} \boldsymbol{\Delta}_{j_1,j_2}(\mathbf{z}) + \left(1 - \sum_{\substack{j_1,j_2=1 \\ j_1 \neq j_2}}^{n} \mathbf{a}_{j_1,j_2}\right) \mathbf{z}, \tag{4}$$

where the $\boldsymbol{\Delta}_{j_1,j_2}(\cdot)$ are the functions acting on the differential states $\mathbf{x}(t_s^-)$ in case $j_1 \to_{\boldsymbol{\omega}} j_2$ at the switching points $\mathcal{S}(\boldsymbol{\omega})$.
We set up the following problem

$$\begin{aligned}
\min_{\mathbf{x}(\cdot),\mathbf{u}(\cdot),\boldsymbol{\omega}(\cdot),\boldsymbol{\theta}_{j_1,j_2}(\cdot)} \quad & \phi(\mathbf{x}(t_f)) + \pi \sum_{t \in \mathcal{S}(\boldsymbol{\omega})\cup\mathcal{P}} \sum_{\substack{j_1,j_2=1 \\ j_1 \neq j_2}}^{n} \boldsymbol{\theta}_{j_1,j_2}(t) \\
\text{s.t.} \quad & \boldsymbol{\omega} \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, \{0,1\}^n) && \text{(5a)} \\
& \dot{\mathbf{x}}(t) = \textstyle\sum_{j=1}^{n} \boldsymbol{\omega}_j(t) \cdot \mathbf{f}^j(\mathbf{x}(t), \mathbf{u}(t)) \qquad a.e.\, t \in \mathcal{T} && \text{(5b)} \\
& \boldsymbol{\theta}_{j_1,j_2} \in V_{\mathcal{S}(\boldsymbol{\omega})\cup\mathcal{P}}(\mathcal{T}) && \text{(5c)} \\
& \boldsymbol{\theta}_{j_1,j_2}(t) = \min\left(\boldsymbol{\omega}_{j_1}(t^-), \boldsymbol{\omega}_{j_2}(t^+)\right) \quad \text{if } t \in \mathcal{S}(\boldsymbol{\omega}) \cup \mathcal{P} && \text{(5d)} \\
& \mathbf{x}(t^+) = \boldsymbol{\Delta}\left(\mathbf{x}(t^-), (\boldsymbol{\theta}_{j_1,j_2}(t))_{j_1,j_2}\right) \quad \text{if } t \in \mathcal{S}(\boldsymbol{\omega}) \cup \mathcal{P} && \text{(5e)} \\
& \mathbf{0} \geq \boldsymbol{\omega}_j(t) \cdot \mathbf{c}^j(\mathbf{x}(t), \mathbf{u}(t)) \qquad a.e.\, t \in \mathcal{T}\ \forall j && \text{(5f)}
\end{aligned}$$

Then we have

Proposition 3.1 *The Problems (2) and (5) are equivalent in the following sense: If $(\mathbf{x}(\cdot), \mathbf{u}(\cdot), \boldsymbol{\omega}(\cdot))$ is feasible for Problem (2), then there exist $\boldsymbol{\theta}_{j_1,j_2}(\cdot)$ such that $(\mathbf{x}(\cdot), \mathbf{u}(\cdot), \boldsymbol{\omega}(\cdot), (\boldsymbol{\theta}_{j_1,j_2}(\cdot))_{j_1\neq j_2})$ is feasible for Problem (5) and the values of the corresponding cost functions coincide. Vice versa, if $(\mathbf{x}(\cdot), \mathbf{u}(\cdot), \boldsymbol{\omega}(\cdot), (\boldsymbol{\theta}_{j_1,j_2}(\cdot))_{j_1\neq j_2})$ is feasible for Problem (5), then $(\mathbf{x}(\cdot), \mathbf{u}(\cdot), \boldsymbol{\omega}(\cdot))$ is feasible for Problem (2) and the values of the cost functions coincide.*

Proof See Appendix A.3. □

We now relax Problem (5), among others by replacing the discrete-valued control function $\boldsymbol{\omega}(\cdot)$ by a control function $\boldsymbol{\alpha}(\cdot)$, which allows for values in $[0,1]^n$. Let us consider the problem

$$\min_{\substack{\mathbf{x}(\cdot),\mathbf{u}(\cdot),\boldsymbol{\alpha}(\cdot),\\ \boldsymbol{\beta}_{j_1,j_2}(\cdot),\boldsymbol{\theta}_{j_1,j_2}(\cdot)}} \quad \phi(\mathbf{x}(t_f)) + \pi \sum_{t\in\mathcal{S}(\boldsymbol{\alpha})\cup\mathcal{P}} \sum_{\substack{j_1,j_2=1\\ j_1\neq j_2}}^{n} \boldsymbol{\theta}_{j_1,j_2}(t)$$

$$\text{s.t.} \quad \boldsymbol{\alpha} \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T},[0,1]^n) \tag{6a}$$

$$\dot{\mathbf{x}}(t) = \textstyle\sum_{j=1}^n \boldsymbol{\alpha}_j(t)\cdot \mathbf{f}^j(\mathbf{x}(t),\mathbf{u}(t)) \qquad a.e.\, t\in\mathcal{T} \tag{6b}$$

$$\boldsymbol{\beta}_{j_1,j_2}, \boldsymbol{\theta}_{j_1,j_2} \in V_{\mathcal{S}(\boldsymbol{\alpha})\cup\mathcal{P}}(\mathcal{T}) \tag{6c}$$

$$\boldsymbol{\theta}_{j_1,j_2}(t) \geq \boldsymbol{\beta}_{j_1,j_2}(t)\boldsymbol{\alpha}_{j_1}(t^-) + (1-\boldsymbol{\beta}_{j_1,j_2}(t))\,\boldsymbol{\alpha}_{j_2}(t^+) \quad \text{if } t\in\mathcal{S}(\boldsymbol{\alpha})\cup\mathcal{P} \tag{6d}$$

$$\mathbf{x}(t^+) = \boldsymbol{\Delta}\left(\mathbf{x}(t^-), (\boldsymbol{\theta}_{j_1,j_2}(t))_{j_1,j_2}\right) \quad \text{if } t\in\mathcal{S}(\boldsymbol{\alpha})\cup\mathcal{P} \tag{6e}$$

$$\mathbf{0} \geq \boldsymbol{\omega}_j(t)\cdot \mathbf{c}^j(\mathbf{x}(t),\mathbf{u}(t)) \qquad a.e.\, t\in\mathcal{T}\ \forall j \tag{6f}$$

Indeed, Problem (6) is a relaxation of Problem (2):

Proposition 3.2 *Let $(\mathbf{x}(\cdot),\mathbf{u}(\cdot),\boldsymbol{\omega}(\cdot))$ be feasible for Problem (2) and set $\boldsymbol{\alpha}(\cdot)=\boldsymbol{\omega}(\cdot)$. Then there exist functions $\boldsymbol{\beta}_{j_1,j_2}(\cdot)$ and $\boldsymbol{\theta}_{j_1,j_2}(\cdot)$, such that $(\mathbf{x}(\cdot),\mathbf{u}(\cdot),\boldsymbol{\alpha}(\cdot),(\boldsymbol{\beta}_{j_1,j_2}(\cdot))_{j_1\neq j_2},(\boldsymbol{\theta}_{j_1,j_2}(\cdot))_{j_1\neq j_2})$ is feasible for Problem (6) for every finite set $\mathcal{P}\subset\mathcal{T}\setminus\{t_0,t_f\}$ and the values of the corresponding cost functions coincide.*

Proof See Appendix A.4 □

In the context of optimal control, the technique of dropping the integrality constraint $\boldsymbol{\omega}(t)\in\{0,1\}^n$ $\forall t\in\mathcal{T}$ by replacing the controls $\boldsymbol{\omega}(\cdot)$ with controls $\boldsymbol{\alpha}(\cdot)$, which take values in $[0,1]^n$, is also known as POC, cf. [34]. Consider Problem (2) without switch costs. We relax the problem as follows: On the one hand, we replace the $\boldsymbol{\omega}(\cdot)$ with $\boldsymbol{\alpha}(\cdot)\in L^\infty(\mathcal{T},[0,1]^n)$ and hence allow for continuous values. On the other hand, we replace the $\mathbf{0}$ in the inequality constraints by $\delta\cdot\mathbf{1}$, where $\delta>0$ (and $\mathbf{1}$ is a vector of ones of appropriate size). One can show, that every feasible point $\boldsymbol{\alpha}(\cdot)$ of the relaxed problem can be approximated by binary feasible controls $\boldsymbol{\omega}(\cdot)\in L^\infty(\mathcal{T},\{0,1\}^n)$ again, and the smaller δ is, the better is the approximation. In particular, this holds for the optimal solution. Hence POC is a reasonable approach to solve MIOCPs. For details see [27, 29, 31].
Let us note, that we are not aware of an extension of the theoretical result mentioned above to MIOCPs in which the cost function depends on the integer controls, as for the switch costs. Anyway, we pursue the described approach, as it works out well in numerical experiments.

3.2 Control Discretization

We intend to develop strategies for the numerical solution of Problem (6) using a direct approach (*'first discretize, then optimize'*). Therefore we discretize the control functions first. To this aim, we introduce a time grid

$$\mathbb{G} = \{t_0 < t_1 < \cdots < t_N = t_f\}$$

with $\min_{i=1,\dots,N}|t_i - t_{i-1}| \geq \bar{\delta}$, and set $\mathcal{P} = \mathbb{G}\setminus\{t_0,t_f\}$. In accordance with Assumption (2.1), we restrict the control function $\boldsymbol{\alpha}(\cdot)$ to be locally constant on the grid intervals $[t_i,t_{i+1})$ resp. $[t_{N-1},t_N]$. Hence we can parameterize $\boldsymbol{\alpha}(\cdot)$ using vectors $\mathbf{q}^0,\dots,\mathbf{q}^{N-1}\in[0,1]^n$:

$$\boldsymbol{\alpha}(t) = \mathbf{q}^i \quad \text{ for all } t\in[t_i,t_{i+1}) \text{ resp. } [t_{N-1},t_N]\,.$$

Observe that due to this discretization, switches can only occur at the inner grid points, and therefore

$$\mathcal{S}(\boldsymbol{\alpha}) \subseteq \mathbb{G}\setminus\{t_0,t_f\} = \mathcal{P}\,.$$

The controls $\boldsymbol{\beta}_{j_1,j_2}(\cdot),\boldsymbol{\theta}_{j_1,j_2}(\cdot)\in V_{\mathcal{P}}(\mathcal{T})$ can be parameterized by $\boldsymbol{\beta}^i_{j_1,j_2},\boldsymbol{\theta}^i_{j_1,j_2}\in[0,1]$, $i=0,...,N$–2 such that

$$\boldsymbol{\beta}^{i-1}_{j_1,j_2} = \boldsymbol{\beta}_{j_1,j_2}(t_i) \ \text{ and } \ \boldsymbol{\theta}^{i-1}_{j_1,j_2} = \boldsymbol{\theta}_{j_1,j_2}(t_i) \ \text{ for all } i=1,\dots,N-1\,.$$

The $\boldsymbol{\theta}^i_{j_1,j_2}$ are called *switching indicators*. Since $\boldsymbol{\alpha}(t_i^-) = \mathbf{q}^{i-1}$ and $\boldsymbol{\alpha}(t_i^+) = \mathbf{q}^i$ for every inner grid point, the constraints (6d) can now be expressed in the following way:

$$\boldsymbol{\theta}^i_{j_1,j_2} \geq \boldsymbol{\beta}^i_{j_1,j_2}\mathbf{q}^i_{j_1} + \left(1 - \boldsymbol{\beta}^i_{j_1,j_2}\right)\mathbf{q}^{i+1}_{j_2} \text{ for } j_1 \neq j_2 \text{ and } i = 0, \dots, N-2\,.$$

Accordingly, we replace the control function $\mathbf{u}(\cdot)$ by some function $\mathbf{U}(\cdot)$, which can be parameterized by a finite number of parameters. Let $\mathbb{S}^n = \{\mathbf{v} \in \{0,1\}^n \mid \sum_{j=1}^n \mathbf{v}_j = 1\}$ and $\operatorname{conv}(\mathbb{S}^n) = \{\mathbf{v} \in [0,1]^n \mid \sum_{j=1}^n \mathbf{v}_j = 1\}$ its convex hull. After the control discretization, the resulting problem takes the following form:

$$\begin{aligned}
\min_{\substack{\mathbf{x}(\cdot),\mathbf{U}(\cdot),\boldsymbol{\alpha}(\cdot),\\ \boldsymbol{\beta}^i_{j_1,j_2},\boldsymbol{\theta}^i_{j_1,j_2}}} \quad & \phi(\mathbf{x}(t_f)) + \pi \sum_{i=0}^{N-2} \sum_{\substack{j_1,j_2=1\\ j_1\neq j_2}}^{n} \boldsymbol{\theta}^i_{j_1,j_2} & & \\
\text{s.t.} \quad & \mathbf{q}^i \in \operatorname{conv}(\mathbb{S}^n) & i = 0,\dots,N-1 \quad & (7a)\\
& \boldsymbol{\alpha}(t) = \mathbf{q}^i \quad \text{for } t \in [t_i, t_{i+1}) & i = 0,\dots,N-1 \quad & (7b)\\
& \dot{\mathbf{x}}(t) = \textstyle\sum_{j=1}^n \boldsymbol{\alpha}_j(t) \cdot \mathbf{f}^j(\mathbf{x}(t), \mathbf{U}(t)) & a.e.\, t \in \mathcal{T} \quad & (7c)\\
& \boldsymbol{\beta}^i_{j_1,j_2}, \boldsymbol{\theta}^i_{j_1,j_2} \in [0,1] & i = 0,\dots,N-2 \quad & (7d)\\
& \boldsymbol{\theta}^i_{j_1,j_2} \geq \boldsymbol{\beta}^i_{j_1,j_2}\mathbf{q}^i_{j_1} + \left(1 - \boldsymbol{\beta}^i_{j_1,j_2}\right)\mathbf{q}^{i+1}_{j_2} & i = 0,\dots,N-2 \quad & (7e)\\
& \mathbf{x}(t^+_{i+1}) = \boldsymbol{\Delta}\left(\mathbf{x}(t^-_{i+1}), \left(\boldsymbol{\theta}^i_{j_1,j_2}\right)_{j_1,j_2}\right) & i = 0,\dots,N-2 \quad & (7f)\\
& \mathbf{0} \geq \boldsymbol{\alpha}_j(t) \cdot \mathbf{c}^j(\mathbf{x}(t), \mathbf{U}(t)) & a.e.\, t \in \mathcal{T}\ \forall j \quad & (7g)
\end{aligned}$$

Observe, that for binary valued $\boldsymbol{\alpha}(\cdot)$, i.e. $\mathbf{q}^i \in \mathbb{S}^n$, we have

$$|\mathcal{S}(\boldsymbol{\alpha})| \leq \sum_{i=0}^{N-2} \sum_{\substack{j_1,j_2=1\\ j_1\neq j_2}}^{n} \boldsymbol{\theta}^i_{j_1,j_2}\,, \tag{8}$$

and if the switching indicators $\boldsymbol{\theta}^i_{j_1,j_2}$ take their smallest possible value, (8) even holds with equality. The presented approach has its pros and cons. On the one hand, in Problem (1), switches can happen at any time $t \in (t_0 + \bar{\delta}, t_f - \bar{\delta})$ and need to be detected in some way. In Problem (7) however, switches can only occur in the inner grid point, which relieves us from classical switch detection. On the other hand, the presented approach suffers from two drawbacks. The number of variables goes quadratically with the number of modes, which can quickly result in a huge number of variables. Another drawback is the situation, when the switching indicators $\boldsymbol{\theta}^i_{j_1,j_2}$ do not take binary values in the solution of Problem (7). The *relaxed switch costs* $\pi \sum_i \sum_{j_1\neq j_2} \boldsymbol{\theta}^i_{j_1,j_2}$ as well as the aggregated jump function $\boldsymbol{\Delta}$ might have no physical meaning in this case. Hence, if we find such solutions, one should think of additional strategies, e.g. penalty terms, to enforce binary values and consequently also meaningful jump functions in the solution.

4 Switch Costs

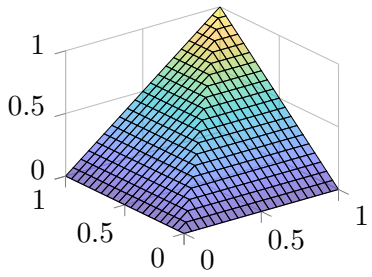
In the previous section we reformulated the switch costs in Problem (2) in a numerically useful manner. In this section, we give two alternatives and compare the expressions with each other. Both approaches are generalizations of an idea by Kirches [26].
Let $\boldsymbol{\alpha} \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, [0,1]^n)$ and $t \in \mathcal{T}$. In accordance with our previous notation, we say

$$\text{System is in mode } j \text{ at } t \iff \boldsymbol{\alpha}_j(t) = 1 \text{ at } t\,.$$

If there is an index j with $\boldsymbol{\alpha}_j(t) \in (0,1)$, we speak of a *fractional mode*. Furthermore we expand our notation by

$$j_1 \to_{\boldsymbol{\alpha}} j_2 \text{ at } t_s \overset{def}{\iff} \boldsymbol{\alpha}_{j_1}(t_s^-) = \boldsymbol{\alpha}_{j_2}(t_s^+) = 1\,,$$

which again means, the system switches its mode at t_s.
Since for now we are only interested in switch costs, we assume $\boldsymbol{\Delta}(\cdot) = \mathbf{Id}(\cdot)$ for the remainder of this section, and therefore consider systems *without* jumps in the differential states. However, we comment on the suitability of the subsequent reformulations in presence of jumps.



$$\mathbf{q}^i_{j_1} = \boldsymbol{\alpha}_{j_1}(t_i) \qquad \mathbf{q}^{i+1}_{j_2} = \boldsymbol{\alpha}_{j_2}(t_{i+1})$$

Figure 1: Minimal possible value of the 'omnipotent' switching indicator $\boldsymbol{\theta}^i_{j_1,j_2}$ due to inequality (7e).

4.1 Reformulation 'Omnipotent'

This reformulation was already explained in Section 3.1. In the resulting control-discretized problem of Section 3.2, it reads as follows: Let $\boldsymbol{\alpha} \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, [0,1]^n)$. For every distinct pair of modes (j_1, j_2), we consider parameters $\boldsymbol{\theta}^i_{j_1,j_2} \in [0,1]$ with the property

$$\boldsymbol{\theta}^i_{j_1,j_2} \geq \min(\boldsymbol{\alpha}_{j_1}(t_i), \boldsymbol{\alpha}_{j_1}(t_{i+1})) \quad \text{for } i = 0, \dots, N-2\,, \tag{9}$$

(as a consequence of (7e)) and the term

$$\pi \sum_{i=0}^{N-2} \sum_{\substack{j_1,j_2=1 \\ j_1 \neq j_2}}^{n} \boldsymbol{\theta}^i_{j_1,j_2} \tag{10}$$

is added to the cost function of the considered OCP for some $\pi > 0$. Figure 1 displays the minimal possible values of $\boldsymbol{\theta}^i_{j_1,j_2}$ for given $\boldsymbol{\alpha}(\cdot)$.
Let us assume that an optimal $\boldsymbol{\alpha}^*(\cdot)$ is binary-valued, i.e. $\boldsymbol{\alpha}^* \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, \{0,1\}^n)$. Then

$$\min(\boldsymbol{\alpha}^*_{j_1}(t_i), \boldsymbol{\alpha}^*_{j_1}(t_{i+1})) = \begin{cases} 1 & \text{if } j_1 \to_{\boldsymbol{\alpha}^*} j_2 \text{ at } t_{i+1} \\ 0 & \text{else} \end{cases},$$

the inequalities (9) are active due to minimization, and (10) indeed equals the penalized number of switches $\pi|\mathcal{S}(\boldsymbol{\alpha}^*)|$. Therefore the according set of optimal switching indicators is *omnipotent* in the sense, that for every inner grid point $t_{i+1} \in \mathbb{G}$, the family $\left(\boldsymbol{\theta}^i_{j_1,j_2}\right)_{j_1 \neq j_2}$ contains the information whether a switch occurred and if so, which modes are involved in the switch as in which order, i.e. if $j_1 \to_{\boldsymbol{\alpha}^*} j_2$ or $j_2 \to_{\boldsymbol{\alpha}^*} j_1$ at t_{i+1}. The resulting (control-discretized) OCP finally takes the form

$$\begin{aligned}
\min_{\substack{\mathbf{x}(\cdot),\mathbf{U}(\cdot),\boldsymbol{\alpha}(\cdot), \\ \boldsymbol{\beta}^i_{j_1,j_2},\boldsymbol{\theta}^i_{j_1,j_2}}} \quad & \phi(\mathbf{x}(t_f)) + \pi \sum_{i=0}^{N-2} \sum_{\substack{j_1,j_2=1 \\ j_1 \neq j_2}}^{n} \boldsymbol{\theta}^i_{j_1,j_2} && \text{(OCP-Omnipotent)} \\
\text{s.t.} \quad & \mathbf{q}^i \in \operatorname{conv}(\mathbb{S}^n) && i = 0, \dots, N-1 \\
& \boldsymbol{\alpha}(t) = \mathbf{q}^i \quad \text{for } t \in [t_i, t_{i+1}) && i = 0, \dots, N-1 \\
& \dot{\mathbf{x}}(t) = \textstyle\sum_{j=1}^n \boldsymbol{\alpha}_j(t) \cdot \mathbf{f}^j(\mathbf{x}(t), \mathbf{U}(t)) && a.e.\, t \in \mathcal{T} \\
& \boldsymbol{\beta}^i_{j_1,j_2}, \boldsymbol{\theta}^i_{j_1,j_2} \in [0,1] && i = 0, \dots, N-2 \\
& \boldsymbol{\theta}^i_{j_1,j_2} \geq \boldsymbol{\beta}^i_{j_1,j_2} \mathbf{q}^i_{j_1} + \left(1 - \boldsymbol{\beta}^i_{j_1,j_2}\right) \mathbf{q}^{i+1}_{j_2} && i = 0, \dots, N-2 \\
& \mathbf{0} \geq \boldsymbol{\alpha}_j(t) \cdot \mathbf{c}^j(\mathbf{x}(t), \mathbf{U}(t)) && a.e.\, t \in \mathcal{T}\ \forall j
\end{aligned}$$

The properties of the switching indicators make them suitable for the treatment of jumps, see Problem (7). However, because of the jump condition (7f), for the solution of the problem it is not true anymore, that binary valued $\boldsymbol{\alpha}(\cdot)$ imply binary valued switching indicators.

4.2 Reformulation 'Involved'

Now we present the original idea by Kirches [26]. Let $\boldsymbol{\omega} \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, \{0,1\}^n)$ and $t_s \in \mathcal{S}(\boldsymbol{\omega})$. Then

$$\min\left(\boldsymbol{\omega}_j(t_s^-) + \boldsymbol{\omega}_j(t_s^+), 2 - \boldsymbol{\omega}_j(t_s^-) - \boldsymbol{\omega}_j(t_s^+)\right) = \begin{cases} 1 & \text{if } j \to_{\boldsymbol{\omega}} j' \text{ or } j' \to_{\boldsymbol{\omega}} j \text{ for some } j' \neq j \\ 0 & \text{else} \end{cases}$$

for all j. Furthermore, we have

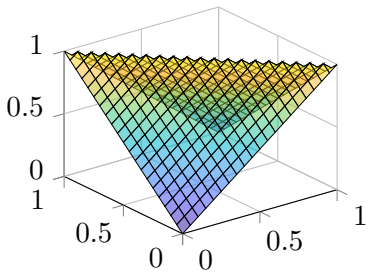
$$|\mathcal{S}(\boldsymbol{\omega})| = \frac{1}{2} \sum_{t_s \in \mathcal{S}(\boldsymbol{\omega})} \min\left(\boldsymbol{\omega}_j(t_s^-) + \boldsymbol{\omega}_j(t_s^+), 2 - \boldsymbol{\omega}_j(t_s^-) - \boldsymbol{\omega}_j(t_s^+)\right) .$$

Using this idea and processing Problem (2) similarly as in the Sections 3.1 and 3.2, we receive the control-discretized problem

$$\min_{\substack{\mathbf{x}(\cdot),\mathbf{U}(\cdot),\boldsymbol{\alpha}(\cdot),\\ \boldsymbol{\beta}^i_j,\boldsymbol{\Theta}^i_j}} \quad \phi(\mathbf{x}(t_f)) + \pi \sum_{i=0}^{N-2} \frac{1}{2} \sum_{j=1}^{n} \boldsymbol{\Theta}^i_j \qquad \text{(OCP-Involved)}$$

$$\begin{aligned}
\text{s.t.} \quad \mathbf{q}^i &\in \operatorname{conv}(\mathbb{S}^n) && i = 0, \dots, N-1 \\
\boldsymbol{\alpha}(t) &= \mathbf{q}^i \quad \text{for } t \in [t_i, t_{i+1}) && i = 0, \dots, N-1 \\
\dot{\mathbf{x}}(t) &= \textstyle\sum_{j=1}^n \boldsymbol{\alpha}_j(t) \cdot \mathbf{f}^j(\mathbf{x}(t), \mathbf{U}(t)) && a.e.\, t \in \mathcal{T} \\
\boldsymbol{\beta}^i_j, \boldsymbol{\Theta}^i_j &\in [0,1] && i = 0, \dots, N-2 \\
\boldsymbol{\Theta}^i_j &\geq \boldsymbol{\beta}^i_j \left(\mathbf{q}^i_j + \mathbf{q}^{i+1}_j\right) + \left(1 - \boldsymbol{\beta}^i_j\right)\left(2 - \mathbf{q}^i_j - \mathbf{q}^{i+1}_j\right) && i = 0, \dots, N-2 \qquad (11) \\
\mathbf{0} &\geq \boldsymbol{\alpha}_j(t) \cdot \mathbf{c}^j(\mathbf{x}(t), \mathbf{U}(t)) && a.e.\, t \in \mathcal{T}\ \forall j
\end{aligned}$$

The variables $\boldsymbol{\Theta}^i_j$ are also called *switching indicators*, and Figure 2 displays the minimal possible values of $\boldsymbol{\theta}^i_{j_1,j_2}$ for a feasible $\boldsymbol{\alpha} \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, [0,1]^n)$.



$\mathbf{q}^i_j = \boldsymbol{\alpha}_j(t_i)$ $\qquad$ $\mathbf{q}^{i+1}_j = \boldsymbol{\alpha}_j(t_{i+1})$

Figure 2: Minimal possible value of the 'involved' switching indicator $\boldsymbol{\Theta}^i_j$ if inequality (11) holds.

Let us again assume, that an optimal control $\boldsymbol{\alpha}^*(\cdot)$ takes binary values, and let $\boldsymbol{\Theta}^i_j$ be the according optimal variables. Then similarly to Section 4.1, because of the inequalities (11), we have

$$|\mathcal{S}(\boldsymbol{\alpha}^*)| = \sum_{i=0}^{N-2} \frac{1}{2} \sum_{j=1}^{n} \boldsymbol{\Theta}^i_j \, .$$

For every inner grid point $t_{i+1} \in \mathbb{G}$, the family $\left(\boldsymbol{\Theta}^i_j\right)_j$ contains the information, whether a switch occurred or not. In contrast to the 'omnipotent' switching indicators, this time we only receive the information, which modes are *involved* in a switch, but the order of modes remains hidden. Therefore, using our approach the 'involved' switching indicators are only suited for the treatment of jumps in special cases, e.g. if all jump functions $\boldsymbol{\Delta}_{j_1,j_2}(\cdot)$ coincide.

4.3 Reformulation 'Subsequent'

Let again $\boldsymbol{\omega} \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, \{0,1\}^n)$ and $t_s \in \mathcal{S}(\boldsymbol{\omega})$. Then

$$\min\left(\boldsymbol{\omega}_j(t_s^+), 1-\boldsymbol{\omega}_j(t_s^-)\right) = \begin{cases} 1 & \text{if } j' \to_{\boldsymbol{\omega}} j \text{ for some } j' \neq j \\ 0 & \text{else} \end{cases}$$

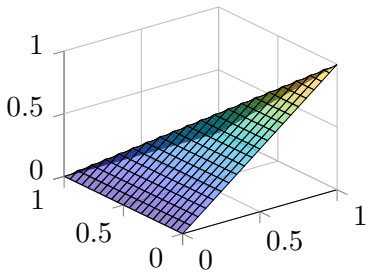
for all j and we have

$$|\mathcal{S}(\boldsymbol{\omega})| = \sum_{t_s \in \mathcal{S}(\boldsymbol{\omega})} \min\left(\boldsymbol{\omega}_j(t_s^+), 1-\boldsymbol{\omega}_j(t_s^-)\right) .$$

If we use this idea and again process Problem (2) as in the Sections 3.1 and 3.2, we get

$$\begin{aligned}
\min_{\substack{\mathbf{x}(\cdot),\mathbf{U}(\cdot),\boldsymbol{\alpha}(\cdot),\\ \boldsymbol{\beta}^i_j,\boldsymbol{\theta}^i_j}} \quad & \phi(\mathbf{x}(t_f)) + \pi \sum_{i=0}^{N-2} \frac{1}{2} \sum_{j=1}^{n} \boldsymbol{\theta}^i_j && && \text{(OCP-Subsequent)} \\
\text{s.t.} \quad & \mathbf{q}^i \in \operatorname{conv}(\mathbb{S}^n) && i = 0,\dots,N-1 \\
& \boldsymbol{\alpha}(t) = \mathbf{q}^i \quad \text{for } t \in [t_i, t_{i+1}) && i = 0,\dots,N-1 \\
& \dot{\mathbf{x}}(t) = \textstyle\sum_{j=1}^n \boldsymbol{\alpha}_j(t) \cdot \mathbf{f}^j(\mathbf{x}(t), \mathbf{U}(t)) && a.e.\, t \in \mathcal{T} \\
& \boldsymbol{\beta}^i_j, \boldsymbol{\theta}^i_j \in [0,1] && i = 0,\dots,N-2 \\
& \boldsymbol{\theta}^i_j \geq \boldsymbol{\beta}^i_j \mathbf{q}^{i+1}_j + \left(1-\boldsymbol{\beta}^i_j\right)\left(1-\mathbf{q}^i_j\right) && i = 0,\dots,N-2 && (12) \\
& \mathbf{0} \geq \boldsymbol{\alpha}_j(t) \cdot \mathbf{c}^j(\mathbf{x}(t), \mathbf{U}(t)) && a.e.\, t \in \mathcal{T}\ \forall j
\end{aligned}$$

The variables $\boldsymbol{\theta}^i_j$ are again called *switching indicators*, and for a feasible $\boldsymbol{\alpha} \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, [0,1]^n)$, their minimal possible value is displayed in Figure 3.



$\mathbf{q}^i_j = \boldsymbol{\alpha}_j(t_i)$ $\qquad$ $\mathbf{q}^{i+1}_j = \boldsymbol{\alpha}_j(t_{i+1})$

Figure 3: Minimal possible value of the 'subsequent' switching indicator $\boldsymbol{\theta}^i_j$ in presence of inequality (12).

Let us again assume, that an optimal control $\boldsymbol{\alpha}^*(\cdot)$ is binary-valued, and let $\boldsymbol{\theta}^i_j$ be the according optimal variables. Then for every inner grid point $t_{i+1} \in \mathbb{G}$, the switching indicators $\left(\boldsymbol{\theta}^i_j\right)_j$ contain the information, whether a switch occurred or not, and if so, what is the mode in the *subsequent* interval $[t_{i+1}, t_{i+2})$. The mode in the interval $[t_i, t_{i+1})$ stays hidden.
In the presence of jumps, using our approach the 'subsequent' switching indicators are therefore only suitable in special cases, for instance if the jump functions $\boldsymbol{\Delta}_{j_1,j_2}(\cdot)$ only depend on the mode after a switch, i.e. $\boldsymbol{\Delta}_{j_1,j_2}(\cdot) = \boldsymbol{\Delta}_{j_1',j_2}(\cdot)$ for all $j_1, j_1' \neq j_2$.

4.4 Comparison of the Reformulations

In this section, we compare the three types of switching indicators $\boldsymbol{\Theta}^i_j, \boldsymbol{\theta}^i_j$ and $\boldsymbol{\theta}^i_{j_1,j_2}$ introduced in the last sections, resp. their minimal possible values and review some of their properties. For this purpose, we define

$$\phi_{inv},\ \phi_{subs},\ \phi_{omni} : \operatorname{conv}(\mathbb{S}^n) \times \operatorname{conv}(\mathbb{S}^n) \longrightarrow \mathbb{R}$$

by

$$\phi_{inv}(\mathbf{a},\mathbf{b}) = \frac{1}{2}\sum_{j=1}^{n} \min(\mathbf{a}_j + \mathbf{b}_j, 2 - \mathbf{a}_j - \mathbf{b}_j)\,,$$
$$\phi_{subs}(\mathbf{a},\mathbf{b}) = \sum_{j=1}^{n} \min(\mathbf{b}_j, 1 - \mathbf{a}_j)\,,$$
$$\phi_{omni}(\mathbf{a},\mathbf{b}) = \sum_{\substack{j_1,j_2=1 \\ j_1\neq j_2}}^{n} \min(\mathbf{a}_{j_1}, \mathbf{b}_{j_2})\,.$$

The minimal contributions to the cost functions of the Problems (OCP-Involved), (OCP-Subsequent) and (OCP-Omnipotent), which belong to the grid point t_{i+1}, are then given by

$$\phi_{inv}\left(\mathbf{q}^i, \mathbf{q}^{i+1}\right) \quad \text{resp.} \quad \phi_{subs}\left(\mathbf{q}^i, \mathbf{q}^{i+1}\right) \quad \text{resp.} \quad \phi_{omni}\left(\mathbf{q}^i, \mathbf{q}^{i+1}\right)\,.$$

We first investigate upper bounds of the three functions.

Proposition 4.1 *Let $\mathbf{a},\mathbf{b} \in conv(\mathbb{S}^n)$. We have $\phi_{inv}(\mathbf{a},\mathbf{b}),\ \phi_{subs}(\mathbf{a},\mathbf{b}) \leq 1$. If $\mathbf{a}_j + \mathbf{b}_j \leq 1$ for every component j, we even get $\phi_{inv}(\mathbf{a},\mathbf{b}) = \phi_{subs}(\mathbf{a},\mathbf{b}) = 1$. For ϕ_{omni} we have*

$$\sup_{\mathbf{a},\mathbf{b}\in conv(\mathbb{S}^n)} \phi_{omni}(\mathbf{a},\mathbf{b}) = n - 1\,.$$

Proof See Appendix A.5 □

Second, we investigate lower bounds.

Proposition 4.2 *We have $\phi_{inv}(\mathbf{a},\mathbf{b}),\ \phi_{subs}(\mathbf{a},\mathbf{b}),\ \phi_{omni}(\mathbf{a},\mathbf{b}) \geq 0$ for all $\mathbf{a},\mathbf{b} \in conv(\mathbb{S}^n)$ and*

$$\phi_{inv}(\mathbf{a},\mathbf{b}) = \phi_{inv}(\mathbf{a},\mathbf{b}) = \phi_{omni}(\mathbf{a},\mathbf{b}) = 0 \quad \Longleftrightarrow \quad \mathbf{a},\mathbf{b} \in \mathbb{S}^n \text{ and } \mathbf{a} = \mathbf{b}\,.$$

Proof See Appendix A.6 □

Consider the Problems (OCP-Involved), (OCP-Subsequent) and (OCP-Omnipotent). The last proposition states, that in view of the (relaxed) switch costs, it is optimal to avoid fractional modes and to stay in the same mode for the whole time horizon. Nevertheless, due to constraints or the Mayer–term contribution in the cost functions, switches are unavoidable or desirable.
Next, we investigate the incurring switch costs in two neighbored intervals.

Proposition 4.3 *Let $\mathbf{a},\mathbf{b},\mathbf{c} \in conv(\mathbb{S}^n)$. For $i \in \{inv, subs\}$, the 'triangle inequality'*

$$\phi_i(\mathbf{a},\mathbf{c}) \leq \phi_i(\mathbf{a},\mathbf{b}) + \phi_i(\mathbf{b},\mathbf{c}) \tag{13}$$

holds. For ϕ_{omni} this is in general not true. Nevertheless, if $\mathbf{a},\mathbf{c} \in \mathbb{S}^n$, then (13) also holds for $i = omni$.

Proof See Appendix A.7 □

Consider the Problems (OCP-Involved), (OCP-Subsequent) and (OCP-Omnipotent) again. Assume, the system is in mode j_1 at time t_i and in mode j_2 at time t_{i+2}. The last proposition states, that – in view of the relaxed switch costs – it is at least not advantageous for the system to switch into some fractional mode at time t_{i+1}. Unfortunately, there are relevant cases, in which is also not disadvantageous, as the next proposition shows.

Proposition 4.4 *Let $\mathbf{a},\mathbf{c} \in \mathbb{S}^n$, such that $\mathbf{a}_l = \mathbf{c}_k = 1$ for some $l \neq k$, and $\mathbf{b} \in conv(\mathbb{S}^n)$. Then for $i \in \{inv, subs, omni\}$, we have*

$$\phi_i(\mathbf{a},\mathbf{c}) = \phi_i(\mathbf{a},\mathbf{b}) + \phi_i(\mathbf{b},\mathbf{c}) \quad \Longleftrightarrow \quad \mathbf{b}_l + \mathbf{b}_k = 1\,. \tag{14}$$

Proof See Appendix A.8. □

Summing up the results of Section 4.4 so far, in view of switch costs, it is advantageous for our system to stay in one (non-fractional) mode j_1 on the whole time horizon, see Proposition 4.2. If the system switches to mode j_2 for any reason, then there are fractional modes such that in view of the switch costs it does not make a difference if the system switches to mode j_2 directly or uses the fractional mode as transition mode for one time interval, see Proposition 4.3 and Proposition 4.4.
As a last step, we consider the special case $n = 2$. Here the choice of switching indicators makes no difference in view of the switch costs:

Proposition 4.5 *Let $n = 2$. Then for all $\mathbf{a}, \mathbf{b} \in conv(\mathbb{S}^n)$ we have*

$$\phi_{inv}(\mathbf{a}, \mathbf{b}) = \phi_{subs}(\mathbf{a}, \mathbf{b}) = \phi_{omni}(\mathbf{a}, \mathbf{b}).$$

Proof See Appendix A.9.

5 Discretization of the Infinite Dimensional OCP

We aim to solve Problem (7) numerically using a direct approach (*'first discretize, then optimize'*). For the direct approach, methods like Direct Multiple Shooting [9] and Direct Collocation [6] have been established as the methods of choice. Similarly to [8], in this paper we use the latter. Collocation methods transcribe the OCP to an NLP by parameterizing the states and controls using polynomials and collocating the differential equations using nodes obtained from a Gaussian quadrature. Since we allow for jumps in the differential states, our framework differs from the one presented by Meyer et al. [8], and therefore we give a detailed description of the discretization again.
As we are dealing with switched systems including jumps, we choose piecewise defined polynomials over the *finite elements* $[t_i, t_{i+1}]$ to discretize differential states and controls. We have already introduced the time grid $\mathbb{G} = \{t_0 < t_1 < \cdots < t_N = t_f\}$ in Section 3.2. For each finite element we choose Lagrange basis polynomials $\left\{\mathcal{L}_k^{(i)}\right\}_{k=0}^{K_i}$ and $\left\{\bar{\mathcal{L}}_m^{(i)}\right\}_{m=1}^{\bar{K}_i}$, given by

$$\mathcal{L}_k^{(i)}(t) = \prod_{\substack{l=0 \\ l\neq k}}^{K_i} \frac{t - t_l^{(i)}}{t_k^{(i)} - t_l^{(i)}}, \quad \bar{\mathcal{L}}_m^{(i)}(t) = \prod_{\substack{l=1 \\ l\neq m}}^{\bar{K}_i} \frac{t - \bar{t}_l^{(i)}}{\bar{t}_m^{(i)} - \bar{t}_l^{(i)}}, \quad i = 0, \ldots, N-1.$$

Depending of the concrete method the *collocation points* $t_k^{(i)}, \bar{t}_m^{(i)} \in \mathbb{R}$ ($k = 1, \ldots, K_i$, $m = 1, \ldots, \bar{K}_i$, $i = 0, \ldots, N-1$) are obtained from the roots of an orthogonal polynomial and/or linear combinations of the polynomial and its derivatives. Due to their good computational efficiency (see e.g. [22, 16]) we choose flipped Legendre-Gauss-Radau (LGR) points in this contribution. If l is the number of collocation points and $\mathcal{P}_l$ denotes the l^{th}-degree Legendre polynomial, LGR points are the roots of $\mathcal{P}_{l-1}(\tau) + \mathcal{P}_l(\tau)$. LGR points lie on the half open interval $t \in [-1, 1)$. One obtains the flipped LGR points by flipping the LGR points about the origin. The affine transformations

$$t^{(i)}(\tau) = \frac{t_{i+1} + t_i}{2} + \tau \, \frac{t_{i+1} - t_i}{2}, \quad i = 0, \ldots, N-1,$$

map flipped LGR points to the finite elements $\mathcal{T}_i = [t_i, t_{i+1}]$ and yield the collocation points $t_k^{(i)}$, $\bar{t}_m^{(i)}$. In addition we set $t_0^{(i)} = t_i$.

The differential states are approximated element-wise as

$$\mathbf{X}^{(i)}(t) = \sum_{k=0}^{K_i} \mathbf{x}_k^{(i)} \mathcal{L}_k^{(i)}(t), \quad t \in \mathcal{T}_i, \quad i = 0, \ldots, N-1,$$

where K_i is the number of collocation points and $\mathbf{x}_k^{(i)} \in \mathbb{R}^{n_x}$ are the nodal values. The derivative with respect to time of the differential state approximations are given by

$$\dot{\mathbf{X}}^{(i)} = \sum_{k=0}^{K_i} \mathbf{x}_k^{(i)} \dot{\mathcal{L}}_k^{(i)}(t), \quad t \in \mathcal{T}_i, \quad i = 0, \dots, N-1.$$

Analogously to the state approximations, the controls $\mathbf{U}$ are given element-wise by

$$\mathbf{U}^{(i)}(t) = \sum_{m=1}^{\bar{K}_i} \mathbf{u}_m^{(i)} \bar{\mathcal{L}}_m^{(i)}(t), \quad t \in \mathcal{T}_i, \quad i = 0, \dots, N-1.$$

Here we have the nodal values $\mathbf{u}_m^{(i)} \in \mathbb{R}^{n_u}$. The controls $\boldsymbol{\alpha}$ resp. their representation $\mathbf{A}$ are piecewise constant functions

$$\mathbf{A}^{(i)}(t) = \mathbf{q}^i, \quad t \in \mathcal{T}_i, \quad i = 0, \dots, N-1.$$

To end up with an NLP we discretize the Mayer–type cost function as $\phi\left(\mathbf{x}_{K_{N-1}}^{(N-1)}\right)$ and the differential equations by means of element–wise collocation

$$\begin{aligned} \mathbf{0} &= \dot{\mathbf{X}}^{(i)}\left(t_k^{(i)}\right) - \sum_{j=1}^{n} \mathbf{A}_j^{(i)}\left(t_k^{(i)}\right) \mathbf{f}^j\left(\mathbf{X}^{(i)}\left(t_k^{(i)}\right), \mathbf{U}^{(i)}\left(t_k^{(i)}\right)\right), \quad k = 1, \dots, K_i, \quad i = 0, \dots, N-1, \\ \iff \quad \mathbf{0} &= \sum_{l=0}^{K_i} \mathbf{x}_l^{(i)} \dot{\mathcal{L}}_l^{(i)}\left(t_k^{(i)}\right) - \sum_{j=1}^{n} \mathbf{q}_j^i \, \mathbf{f}^j\left(\mathbf{x}_k^{(i)}, \mathbf{U}^{(i)}\left(t_k^{(i)}\right)\right), \qquad k = 1, \dots, K_i, \quad i = 0, \dots, N-1. \end{aligned}$$

The jump condition is encoded in the constraints

$$\mathbf{x}_1^{(i+1)} = \boldsymbol{\Delta}\left(\mathbf{x}_{K_i}^{(i)}, (\boldsymbol{\theta}_{j_1,j_2}^i)_{j_1 \neq j_2}\right) \quad i = 0, \dots, N-2$$

where the switching indicators $\boldsymbol{\theta}_{j_1,j_2}^i$ were introduced in Section 3.2, and for the definition of $\boldsymbol{\Delta}(\cdot)$ see (4). Recall the boundary constraints (1f) and the path constraints (1e), that we omitted for notational reason. The discretization of the boundary constraints leads to the NLP constraints

$$\mathbf{0} \geq \mathbf{r}\left(\mathbf{x}_0^{(0)}, \mathbf{x}_{K_{N-1}}^{(N-1)}\right).$$

Path constraints are enforced to hold at collocation points $t_k^{(i)}$ and the constraints (7g) shall hold at all grid points:

$$\begin{aligned} &\mathbf{0} \geq \mathbf{d}\left(\mathbf{x}_k^{(i)}, \mathbf{U}^{(i)}\left(t_k^{(i)}\right)\right), && k \in \{1, \dots, K_i\}, \quad i = 0, \dots, N-1, \\ &\mathbf{0} \geq \mathbf{q}_j^i \cdot \mathbf{c}_j\left(\mathbf{x}_0^{(i)}, \mathbf{U}^{(i)}\left(t_0^{(i)}\right)\right), && j = 1, \dots, n, \quad i = 0, \dots, N-1, \qquad (15a) \\ &\mathbf{0} \geq \mathbf{q}_j^{N-1} \cdot \mathbf{c}_j\left(\mathbf{x}_{K_{N-1}}^{(N-1)}, \mathbf{U}^{(N-1)}\left(t_{K_{N-1}}^{(N-1)}\right)\right), && j = 1, \dots, n. \qquad (15b) \end{aligned}$$

Constraints of the form (15a) resp. (15b) are called *vanishing constraints*, and thus, the resulting NLP is a MPVC. The numerical treatment of MPVCs is addressed in the next section.

6 Numerical Treatment of MPVCs

This section is concerned with a strategy for the numerical treatment of MPVCs, and in particular with the treatment of the collocation NLP from the last last section. In general, an NLP of the form

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^n} \quad & f(\mathbf{x}) && (16a) \\ \text{s.t.} \quad & \mathbf{g}_i(\mathbf{x}) \leq 0 \quad i = 1, \dots, m && (16b) \\ & \mathbf{h}_j(\mathbf{x}) = 0 \quad j = 1, \dots, p && (16c) \\ & \mathbf{H}_i(\mathbf{x}) \geq 0 \quad i = 1, \dots, l && (16d) \\ & \mathbf{H}_i(\mathbf{x})\mathbf{G}_i(\mathbf{x}) \leq 0 \quad i = 1, \dots, l && (16e) \end{aligned}$$

with $\mathcal{C}^1$-functions $f, \mathbf{g}_i, \mathbf{h}_j, \mathbf{H}_i, \mathbf{G}_i : \mathbb{R}^n \to \mathbb{R}$ is called an MPVC [1, 21]. MPVCs arise in many applications, e.g. in truss topology [1], and in discretized MIOCPs like in this paper. In particular, the collocation NLP from the last section is a MPVC.
Due to the structure of the constraints (16e), an MPVC is in general a non-convex problem. Furthermore, constraint qualifications may be violated: Let $\mathbf{x}^*$ be a feasible point for Problem (16). If $\{i \mid \mathbf{H}_i(\mathbf{x}^*) = 0\} \neq \emptyset$, the Linear Independence Constraint Qualification is violated at $\mathbf{x}^*$, and if $\{i \mid \mathbf{H}_i(\mathbf{x}^*) = 0 \text{ and } \mathbf{G}_i(\mathbf{x}^*) \geq 0\} \neq \emptyset$, also the weaker Mangasarian Fromowitz Constraint Qualification [30] is violated [21]. The latter results in an unbounded set of Lagrange multipliers [37] and therefore numerical problems are to be expected. Anyway, it is reasonable to assume that the Guignard Constraint Qualification [18] is satisfied [1]. Hence, stationary points are Karush-Kuhn-Tucker points.

6.1 Relaxation Strategy

In view of the expected numerical difficulties, Izmailov and Solodov [23] propose to embed Problem (16) into a family of perturbed problems (which is also a known approach for Mathematical Programs with Complementarity Constraints):

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & f(\mathbf{x}) & & \\ \text{s.t.} \quad & \mathbf{g}_i(\mathbf{x}) \leq 0 & & i = 1, \dots, m \\ & \mathbf{h}_j(\mathbf{x}) = 0 & & j = 1, \dots, p \\ & \mathbf{H}_i(\mathbf{x}) \geq 0 & & i = 1, \dots, l \\ & \mathbf{H}_i(\mathbf{x})\mathbf{G}_i(\mathbf{x}) \leq \gamma & & i = 1, \dots, l \end{aligned} \tag{17}$$

for $\gamma > 0$. For $\gamma \to 0$, the feasible set of (17) approaches the one of (16), and under mild assumptions, the relaxed problem has advantageous properties in view of holding constraint qualifications. More details and convergence results can also be found in [21].

6.2 Relaxation Homotopy, Backtracking, Adaptive Refinement

Now we specify, how the relaxation strategy from the last section is used to solve the collocation NLP resulting from Problem (7). We choose an approach similar to the one used by Meyer et al. [8]. As stated therein, it is important to couple the discretization accuracy with the value of the homotopy parameter γ. We therefore propose to solve a sequence of NLPs, where γ is driven to zero while the grid is adapted successively.
Let γ_0 the chosen initial parameter and γ_k the relaxation parameter in the k-th iteration. The relaxation parameter is driven to zero by

$$\gamma_{k+1} = \rho\,\gamma_k \quad \text{for } k \geq 0 \tag{18}$$

with some $\rho \in (0, 1)$. After each iteration k, we pursue the following strategy: If the k-th NLP was infeasible, the grid is refined adaptively and we try to solve the resulting NLP again with the same relaxation parameter. In the other case, in addition to the grid refinement we diminish the relaxation parameter using (18).
Meyer [8] proposes a strategy how to refine the grid and how to use the NLP-solvers output from the previous iteration to warm-start the solver in the next iteration. However, our numerical experiments have shown that this strategy does not work properly for our augmented framework. Therefore, the concrete strategy for refining the grid as well as for warm-starting the solver needs to be assigned to each problem individually.
As mentioned in Section 3, we are in general interested in binary-valued variables $\boldsymbol{\theta}^i_{j_1,j_2}$ which satisfy (9) with equality, in particular if jumps are involved. One way to enforce this is to augment the cost function with an additional term

$$\pi_2 \sum_{j=1}^{n} \int_{t_0}^{t_f} \boldsymbol{\alpha}_j(t)(1 - \boldsymbol{\alpha}_j(t))\, dt\,, \tag{19}$$

cf. Sager [34]. Choosing π_2 appropriately this yields binary-valued $\boldsymbol{\alpha}(\cdot)$, and due to the cost function also binary-valued $\boldsymbol{\theta}^i_{j_1,j_2}$ satisfying (9) with equality in turn (if the penalty parameter π

is chosen large enough). Hence, if fractional values of $\boldsymbol{\alpha}(\cdot)$ are detected in the solution of the k-th NLP, we add a discretized version of the term (19) resp. raise the penalty parameter π_2 in the objective belonging to the next iteration's NLP.
This procedure is repeated until the optimal solution of a feasible NLP with prescribed termination tolerance $\gamma \leq \gamma_{\text{acc}}$ is found, $\boldsymbol{\alpha}(\cdot)$ are binary valued and the inequality constraints (9) resp. the corresponding version for the used switching-indicators hold with equality. *We denote this solution by the optimal solution of MIOCP (5).* Numerical problems are to be expected when γ gets too small. A suitable value for γ_{acc} however strongly depends on the constraints (7g), which are problem-specific, and hence needs to be assigned to each problem individually.

7 Numerical experiments

In this section, we present the results of two numerical experiments – an academic example concerned with switch costs only, and an example from robotics, in which we generate a walking-like motion. In the latter example, jumps in the differential states occur.

7.1 An Academic Example Involving Switch Costs

The first numerical experiment addresses switch costs. We introduce a simple OCP which can run in two different modes. In this problem, it is optimal to stay in a fractional mode for the whole time horizon. We then extend the problem by switch costs. As shown in Proposition 4.5, the choice of switching indicators does not matter in case of two possible modes. We therefore decide for the 'omnipotent' switching indicators from Section 4.1. Other numerical experiments regarding switch costs in more complex applications can be found in works of Jung [24] and Kirches [26]. In these references the 'involved' switching indicators from Section 4.2 are used.
The (continuous) problem we consider looks as follows:

$$\begin{aligned} \min_{\mathbf{x}(\cdot),\boldsymbol{\omega}(\cdot)} \quad & \int_0^5 \mathbf{x}(t)^2 \, dt + \pi |\mathcal{S}(\boldsymbol{\omega})| \\ \text{s.t.} \quad & \boldsymbol{\omega} \in S^2_{\mathcal{F}} \cap PC^{\bar{\delta}}(\mathcal{T}, \{0,1\}^2) \\ & \dot{\mathbf{x}}(t) = \begin{cases} +1 & \text{if } \boldsymbol{\omega}_1(t) = 1 \\ -1 & \text{if } \boldsymbol{\omega}_2(t) = 1 \end{cases} \qquad t \in \mathcal{T} \text{ a.e.} \\ & \mathbf{x}(0) = 0 \end{aligned} \tag{20}$$

where $\mathcal{T} = [0,5]$ and $\pi = 0$, and the relaxed problem resulting from POC is given by

$$\begin{aligned} \min_{\mathbf{x}(\cdot),\boldsymbol{\alpha}(\cdot)} \quad & \mathbf{x}_2(5) + \pi |\mathcal{S}(\boldsymbol{\alpha})| \\ \text{s.t.} \quad & \boldsymbol{\alpha} \in S^2_{\mathcal{F}} \cap PC^{\bar{\delta}}(\mathcal{T}, [0,1]^2) \\ & \begin{pmatrix} \dot{\mathbf{x}}_1(t) \\ \dot{\mathbf{x}}_2(t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{\alpha}_1(t) - \boldsymbol{\alpha}_2(t) \\ \mathbf{x}_1(t)^2 \end{pmatrix} \qquad t \in \mathcal{T} \text{ a.e.} \\ & \mathbf{x}(0) = \mathbf{0} \end{aligned} \tag{21}$$

where we replaced the Lagrange term in the cost function by a Mayer–term. Recall that $\boldsymbol{\alpha}_1(t) + \boldsymbol{\alpha}_2(t) = 1$ for all $\boldsymbol{\alpha} \in S^2_{\mathcal{F}}$. Since $\pi = 0$, the optimal control for this relaxed Problem (21) is obviously given by $\boldsymbol{\alpha}_j(t) \equiv \frac{1}{2}$ for $j = 1, 2$ and yields the cost function value 0. For the original Problem (20), the optimal solution depends on the dwell time $\bar{\delta}$ and includes a high-frequency switching.
Now we consider $\pi > 0$. After executing the steps described in Section 3.1, we end up with a relaxed and control-discretized problem with the cost function

$$\mathbf{x}_2(5) + \pi \sum_{i=0}^{N-2} \sum_{\substack{j_1, j_2 = 1 \\ j_1 \neq j_2}}^{2} \boldsymbol{\theta}^i_{j_1, j_2} \, . \tag{22}$$

Depending on the discretization, the choice of π, as well as on the choice of initial values, we find different (local) solutions of the resulting collocation NLP (see Section 5). Among these, there

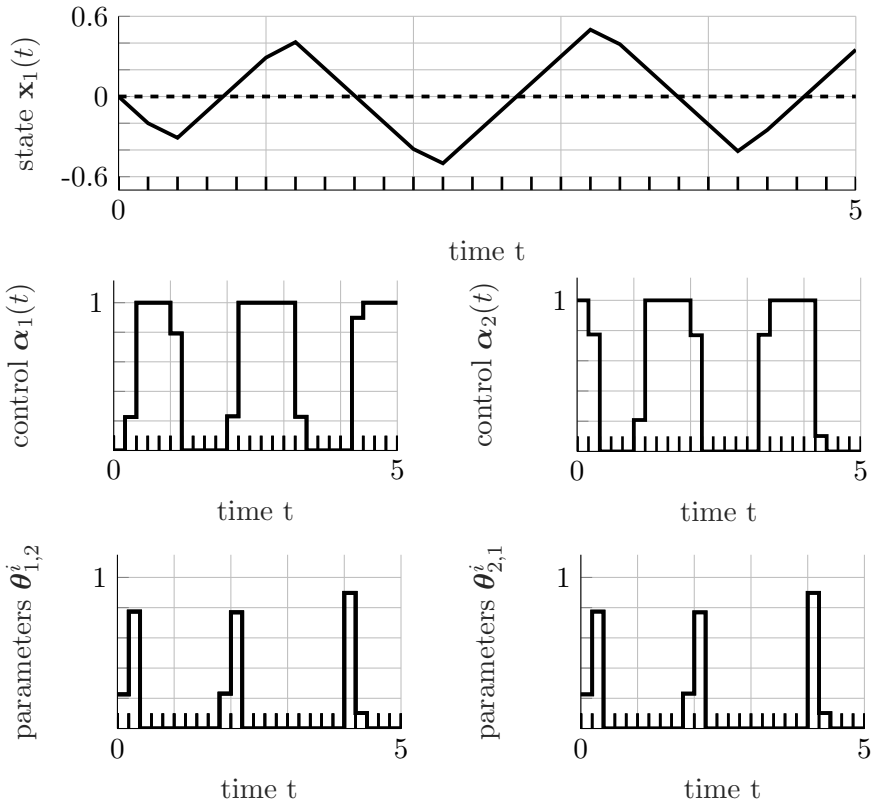


Figure 4: Control profiles for the $\boldsymbol{\alpha}_j(\cdot)$ as well as the $\boldsymbol{\theta}^i_{j_1,j_2}$ – displayed as locally constant functions – in the initial NLP's solution of Problem (21). We choose $\pi = 0.1$ and solve the NLP with IPOPT [38]. The cost function assumes the value ≈ 0.8488.

are trajectories of the shape as displayed in Figure 4. In these cases, the variables $\boldsymbol{\theta}^i_{j_1,j_2}$ take the value 0, except for pairs of neighbored intervals, in which we have $\boldsymbol{\theta}^i_{j_1,j_2} + \boldsymbol{\theta}^{i+1}_{j_1,j_2} = 1$. This reminds of Proposition 4.4. Indeed, we are in the same situation here. Following the proposition, in view of the switch costs it makes no difference, if the system changes its mode directly, or adopts an intermediate value in between. The reason why the system chooses the latter way therefore is to be found in the contribution $\mathbf{x}_2(5)$ in the cost function.
As already mentioned by Kirches [26], unfortunately one cannot observe a direct connection between the parameter π and the number of switches in the found solution. Nevertheless this is not a contradiction, since the used NLP solvers are only able to detect local minima.

7.2 A Walking-Like Motion

Assuming that, as a consequence of nature's evolutionary process, natural gaits are optimal with respect to a certain performance criterion depending on individual trait parameters, we follow [13, 19, 32] and introduce an OCP, solutions of which are the desired optimal walking-like motions. Different from the examples cited, we refrain from using a multi-stage formulation with a predefined order of phases but use our free-phase approach, which is of interest for model-based treatment planning in CP.
In this example we consider the simplest walker model as in [15], a rigid MBS that consists of three point masses, connected by massless rods as displayed in Fig. 5. We refer to the bottom point masses as *feet*, and to the top point mass as *head*, such that the MBS can be seen as a *walker*, a stick-man with two *legs*. Individual trait parameters are normalized to $1kg$ for the head and both feet, and $1m$ length for the rods, of which we neglect the masses.

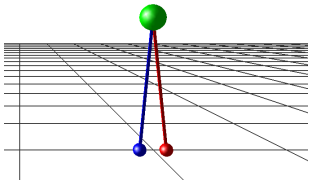


Figure 5: The "Simplest Walker" modeled by a rigid multi-body system. Illustration created using MeshUp [13].

Simplest Walker Dynamics

We allow for movements in only two dimensions. The MBS has four degrees of freedom, comprising the head's position in 2D and the two legs' rotations around the head pivot. The system can be described by means of four *generalized coordinates*, summarized in $\mathbf{q}(t)$, cf. Table 1, and their time derivatives, to which we refer as *generalized velocities*.

Table 1: Generalized coordinates of the "Simplest Walker" MBS.

$\mathbf{q}_1(\cdot)$	horizontal position of the head
$\mathbf{q}_2(\cdot)$	vertical position of the head
$\mathbf{q}_3(\cdot)$	angle between the left leg and its resting position
$\mathbf{q}_4(\cdot)$	angle between the right leg and its resting position

The resting position of a leg is reached if it hangs straight down. The walker is able to accelerate its feet by controlling rotational torques $\boldsymbol{\tau}_1(\cdot)$ and $\boldsymbol{\tau}_2(\cdot)$ applied to the two legs, which we summarize in $\mathbf{u}(\cdot)$.
As we consider walking-like motions, we are interested in the equations of motion of the MBS for varying external contacts, indexed with j. These contacts can be expressed in terms of $\mathbf{q}(t)$ by

$$\mathbf{g}^j(\mathbf{q}(t)) = \mathbf{0}, \tag{23}$$

where j indicates the holding contact at time t. The governing MBS dynamics can then be expressed by a switched Differential Algebraic Equation (DAE) of index 3

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau} + \mathbf{G}^j(\mathbf{q})^T\boldsymbol{\lambda}, \tag{24a}$$

$$\mathbf{g}^j(\mathbf{q}) = \mathbf{0}, \tag{24b}$$

where $\mathbf{H}(\mathbf{q})$ is the generalized inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ the generalized bias force, $\boldsymbol{\tau}$ the vector of applied generalized forces, $\mathbf{G}^j(\mathbf{q}) = \frac{\partial}{\partial \mathbf{q}}\mathbf{g}^j(\mathbf{q})$ the contact Jacobian and $\boldsymbol{\lambda}$ the contact force. After reducing the index to 1, the resulting system reads as

$$\begin{pmatrix} \mathbf{H}(\mathbf{q}) & \mathbf{G}^j(\mathbf{q})^T \\ \mathbf{G}^j(\mathbf{q}) & \mathbf{0} \end{pmatrix} \begin{pmatrix} \ddot{\mathbf{q}} \\ -\boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\tau} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \\ -\dot{\mathbf{G}}^j(\mathbf{q})\dot{\mathbf{q}} \end{pmatrix}, \tag{25}$$

and to ensure equivalence of (25) and (24), the constraints

$$\mathbf{g}^j(\mathbf{q}) = \mathbf{0}, , \tag{26a}$$

$$\mathbf{G}^j(\mathbf{q})\dot{\mathbf{q}} = \mathbf{0}, \tag{26b}$$

need to be satisfied correspondingly.
Whenever the external contact changes, e.g. if the a foot hits the ground after swinging freely before, a *collision impact* occurs and transfers the generalized velocities before the collision, $\dot{\mathbf{q}}(t^-)$, to those after the collision, $\dot{\mathbf{q}}(t^+)$. In our model, the impact is assumed to be perfectly inelastic and can be expressed by

$$\begin{pmatrix} \mathbf{H}(\mathbf{q}) & \mathbf{G}^j(\mathbf{q})^T \\ \mathbf{G}^j(\mathbf{q}) & \mathbf{0} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{q}}(t^+) \\ -\boldsymbol{\Lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{H}(\mathbf{q})\dot{\mathbf{q}}(t^-) \\ \mathbf{0} \end{pmatrix}, \tag{27}$$

where j corresponds to the external contact *after* the impact, and $\mathbf{\Lambda}$ is the contact impulse.
By activating equation (25) towards $\ddot{\mathbf{q}}$ resp. $\boldsymbol{\lambda}$ and differentiating once more, the system can be transferred to a switched ODE system $\dot{\mathbf{x}} = \mathbf{f}^j(\mathbf{x}, \mathbf{u})$, and the jump condition (27) can be transferred to $\mathbf{x}$. In particular, for every switch from mode j_1 to mode j_2 with $(j_1, j_2) \in \{(1,2),(2,1)\}$, there is a function $\mathbf{\Delta}_{j_1,j_2}$, mapping the differential states before the switch to the differential states after the switch and thus reflecting the corresponding jump.
Hence our approach, which was designed for OCPs constrained by switched ODEs, is applicable. Anyway, though the reformulation of the DAE as an ODE is needed for theoretical purposes, in practice the index-1 formulation (25) can already be treated as an ODE, where for every evaluation of the ODE's right-hand side, the linear system needs to be solved. This enables us to work with generalized coordinates and velocities instead of the differential states.
Details on rigid MBS dynamics and algorithms to compute related quantities can be found in [13] and, more extensively, in [12]. The software library RBDL [14] is used in this study and provides all computations required for the purpose of optimal control.

An Optimal Control Problem for a Walking-Like Motion

We first propose a set of constraints imposed to model the process of walking. Let $p_x^{\mathrm{l,r,h}}(t)$ resp. $p_y^{\mathrm{l,r,h}}(t)$ denote the horizontal and vertical position of the left foot, the right foot, and the head, respectively. The walking motion spans a time interval $\mathcal{T} = [0, t_f]$ where $t_{\mathrm{f}} \geq 0$ is a free end time subject to optimization. At time $t = 0$, the constraints

$$p_x^{\mathrm{h}}(0) = 0, \qquad p_y^{\mathrm{l}}(0) = p_y^{\mathrm{r}}(0) = 0, \tag{28a}$$

$$0.2 \leq p_x^{\mathrm{l}}(0) - p_x^{r}(0) \leq 0.8, \qquad 0 \leq p_y^{\mathrm{h}}(0), \tag{28b}$$

$$-\pi \leq \mathbf{q}_3(0), \mathbf{q}_4(0) \leq \pi, \qquad -5 \cdot \mathbf{1} \leq \dot{\mathbf{q}}(0) \leq 5 \cdot \mathbf{1} \tag{28c}$$

force the walker to start in a fixed (28a), unambiguous (28c) position, and with some freedom left for optimizing the initial posture (28c) and velocities (28b). At the end of the time horizon, the constraints

$$1.8 \leq p_x^{\mathrm{h}}(t_{\mathrm{f}}), \quad p_y^{\mathrm{l}}(t_{\mathrm{f}}), p_y^{\mathrm{r}}(t_{\mathrm{f}}) \leq 0.1, \tag{29}$$

impose a posture with both feet at least close to the ground while the prescribed final position (29) forces the walker to move at all. In order to generate a "realistic" walking-like motion, we demand the head of the walker to stay above a certain level, and we would like the feet not to penetrate the ground. Since a stick-man with stiff legs is however not able to walk in a reasonable way without penetrating the ground, we set up a tolerance $\varepsilon_{tol} = 0.1$, and demand

$$-\varepsilon_{tol} \leq p_y^l(t), p_y^r(t), \quad 0.8 \leq p_y^h(t), \tag{30}$$

for all times t, where for the initial time, this is already ensured by (28). Furthermore, we want the resulting movement to be *cyclic* up to a certain accuracy, which means that the posture of the walker in the beginning and the end of the observed interval should not differ too much, and the same shall hold for the velocities of the segments. To achieve this, we demand

$$-\varepsilon_{tol} \leq \mathbf{q}_j(0) - \mathbf{q}_j(t_f) \leq \varepsilon_{tol}, \tag{31}$$

for $j = 3, 4$, and

$$-\varepsilon_{tol}\mathbf{1} \leq \dot{\mathbf{q}}(0) - \dot{\mathbf{q}}(t_f) \leq \varepsilon_{tol}\mathbf{1}. \tag{32}$$

In a walking motion, either one of the two feet must be fixed to the ground to avoid jumping motions. The movement can hence be realized by alternating between two possible contact configurations of the switched MBS, which we call *modes*:

- Mode 1: the *left* foot is fixed to the ground
- Mode 2: the *right* foot is fixed to the ground

A third mode, in which both feet are fixed to the ground, arises only momentarily as an isolated point of transition between modes 1 and 2. The two modes of interest are characterized by

$$\mathbf{0} = \mathbf{c}^1(\mathbf{x}(t)) = (p_y^{\mathrm{l}}(t), v_x^{\mathrm{l}}(t))^T, \tag{33a}$$

$$\mathbf{0} = \mathbf{c}^2(\mathbf{x}(t)) = (p_y^{\mathrm{r}}(t), v_x^{\mathrm{r}}(t))^T, \tag{33b}$$

meaning (33a) resp. (33b) holds, when the system is in mode 1 resp. 2 at time t.
It is reasonable to assume that the switched MBS has a strictly positive dwell-time $\bar{\delta}$, such that only finitely many switches occur, but not in 0 or t_f. The overall holding differential equation together with the mode-characterizing constraints (33) can then be written in term of the differential states

$$\boldsymbol{\omega} \in \mathcal{S}^2_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, \{0,1\}^2) \tag{34a}$$

$$\dot{\mathbf{x}}(t) = \textstyle\sum_{j=1}^2 \boldsymbol{\omega}_j(t) \cdot \mathbf{f}^j(\mathbf{x}(t), \mathbf{u}(t)) \qquad a.e.\, t \in \mathcal{T} \tag{34b}$$

$$\mathbf{0} \geq \pm\boldsymbol{\omega}_j(t) \cdot \mathbf{c}^j(\mathbf{x}(t)) \text{ for } j = 1,2, \quad a.e.\, t \in \mathcal{T}\ \forall j \tag{34c}$$

where $\boldsymbol{\omega}(\cdot)$ are the mode-indicator functions, meaning $\boldsymbol{\omega}_j(t) = 1 \iff$ system is in mode j, and $\mathbf{u}(\cdot)$ are the controllable rotational torques accelerating the feet of the walker.
The cost function which shall be minimized is given by

$$\int_{t_0}^{t_f} \mathbf{u}_1(t)^2 + \mathbf{u}_2(t)^2 dt + t_f + \pi \left|\mathcal{S}(\boldsymbol{\omega})\right| , \tag{35}$$

and encodes a compromise between walking speed, energy consumption and the number of steps. The MIOCP we propose to solve is finally given by

$$\min_{\substack{t_{\mathrm{f}}, \mathbf{x}(\cdot), \mathbf{u}(\cdot), \\ \boldsymbol{\omega}(\cdot), \boldsymbol{\theta}_{j_1,j_2}(\cdot)}} \quad \int_{t_0}^{t_f} \mathbf{u}_1(t)^2 + \mathbf{u}_2(t)^2 dt + t_f + \pi \left|\mathcal{S}(\boldsymbol{\omega})\right| \tag{36a}$$

$$\text{s.t.} \qquad \boldsymbol{\omega} \in \mathcal{S}^2_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, \{0,1\}^2) \tag{36b}$$

$$\dot{\mathbf{x}}(t) = \textstyle\sum_{j=1}^n \boldsymbol{\omega}_j(t) \mathbf{f}^j(\mathbf{x}(t), \mathbf{u}(t)), \tag{36c}$$

$$\boldsymbol{\theta}_{j_1,j_2}(t) = \min\left(\boldsymbol{\omega}_{j_1}(t^-), \boldsymbol{\omega}_{j_2}(t^+)\right), \tag{36d}$$

$$\mathbf{x}(t^+) = \boldsymbol{\Delta}\left(\mathbf{x}(t^-), (\boldsymbol{\theta}_{j_1,j_2}(t))_{j_1,j_2}\right)\ \forall t \in \mathcal{S}(\boldsymbol{\omega}), \tag{36e}$$

$$\mathbf{0} \geq \pm\boldsymbol{\omega}_j(t) \mathbf{c}^j(\mathbf{x}(t)) \text{ for } j = 1,2, \tag{36f}$$

$$\mathbf{0} \geq \mathbf{c}(\mathbf{x}(t)), \tag{36g}$$

$$\mathbf{0} \geq \mathbf{r}(\mathbf{x}(t_0), \mathbf{x}(t_{\mathrm{f}})), \tag{36h}$$

where $\boldsymbol{\theta}_{j_1,j_2}$ are the switching indicators introduced in Section 3.1, $\boldsymbol{\Delta}$ is the aggregated jump-function, see (4), and (36g) and (36h) summarize the constraints (28a)-(32). The solution of Problem (36) describes the optimal gait of the walker w.r.t. the objective function (35). To solve the problem numerically we apply the methods described in the previous sections.

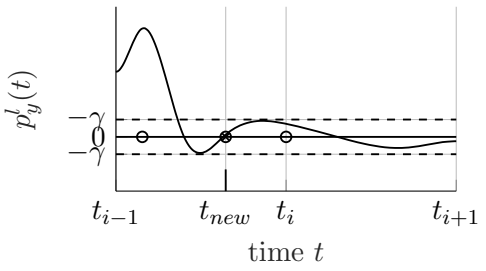
Homotopy, Adaptive Refinement, Warm-start and Problem Adaption

In Section 6.2 we stated, that strategies for homotopy, adaptive refinement and warm-start should be set up for each problem individually. In the present example, we proceed as follows: The factor for diminishing the homotopy parameter according to (18) is set to $\rho = 0.7$. For the adaptive refinement, we set up three rules 1–3. For each interval $[t_i, t_{i+1}]$ we proceed according to the following logic:

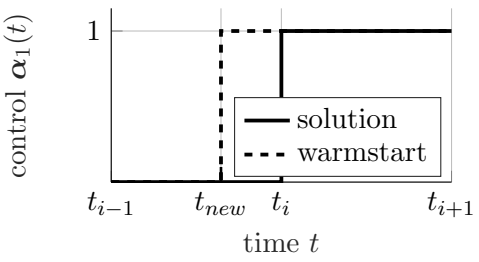
- Shall the interval be refined according to rule 1? If yes, do so. If not, check
- Shall the interval be refined according to rule 2? If yes, do so. If not, check
- Shall the interval be refined according to rule 3? If yes, do so. If not, do not refine.

The rules are given as follows:

Rule 2: Set up a reasonable tolerance ε_{tol} . If either



(a) Exemplary trajectory of the left foot's vertical position in the k-th NLP solution.



(b) Exemplary trajectories of $\boldsymbol{\alpha}_1(\cdot)$ in the k-th NLP solution as well as after the warm-start for iteration $k+1$. Since in the left foot entered the regularized ground (37) already at time t_{new} in the k-th NLP solution, we warm-start the NLP with values of $\boldsymbol{\alpha}_1(\cdot)$, which reflect this behavior.

Figure 6: Exemplary visualization: warm-start of the control $\boldsymbol{\alpha}_1(t)$ in the problem, which is described in Section 7.2. According to rule 1, the new grid point is the first collocation point belonging to the discretization of differential states $\mathbf{x}(\cdot)$, at which the respective foot is located in the regularized ground.

- $\min_j \boldsymbol{\alpha}_j(t_i) > \varepsilon_{tol}$ (i.e. the system is in a fractional mode) or
- $\boldsymbol{\theta}^i_{1,2} > \varepsilon_{tol}$ or $\boldsymbol{\theta}^i_{2,1} > \varepsilon_{tol}$ (i.e. a jump occurs after the according interval)

the interval shall be bisected.
Rule 3: Treating index-reduced DAEs numerically can result in a drift due to numerical errors (if the chosen grid is not fine enough), meaning that the constraints (23) get violated (too strong) after some time. In our example, according to (36f) and its relaxation, for every interval we want one foot to be at least inside the *regularized ground*, meaning one of the conditions

$$-\gamma \leq p_l^y(t) \leq \gamma \quad \text{or} \quad -\gamma \leq p_r^y(t) \leq \gamma \tag{37}$$

should hold at every collocation point inside $[t_i, t_{i+1}]$ belonging to the discretization of the differential states. Since after relaxing and discretizing the problem, we only demand (36f) to hold at the grid points for our optimization problem, in general (37) needs not to be true, even if the controls $\boldsymbol{\alpha}_j(\cdot)$ take binary values. Nevertheless, if it is not true, we refine the interval as well as its precursor in order to achieve a higher accuracy. Both intervals shall be subdivided in the middle again.
Rule 1: Whenever the system changes its mode at t_{i+1}, according to our model we would expect either foot to enter the regularized ground in the interval $[t_i, t_{i+1}]$. If so, we detect the first collocation point in this interval belonging to discretization of the differential states, at which this is the case. We refine this interval and choose the described collocation point as the point for subdivision, unless it coincides with t_{i+1}. In the latter case, we bisect the interval again. For an illustration of rule 1, see Figure 6.

The warm-start then works as follows:

- The differential states $\mathbf{x}(\cdot)$ and the controls $\mathbf{u}_1(\cdot)$ and $\mathbf{u}_2(\cdot)$ are interpolated.
- If an interval is bisected, $\boldsymbol{\alpha}_1(\cdot)$ and $\boldsymbol{\alpha}_2(\cdot)$ are interpolated as well.
- If an interval is subdivided at another point (which can only happen if rule 1 is applied), we use the information about the differential states at the collocation points in order to initialize $\boldsymbol{\alpha}_1(\cdot)$ and $\boldsymbol{\alpha}_2(\cdot)$ reasonably.
- The control parameters $\boldsymbol{\beta}^i_{j_1,j_2}$ and $\boldsymbol{\theta}^i_{j_1,j_2}$ are initialized in a way, that

  $$\boldsymbol{\theta}^i_{j_1,j_2} = \min\left(\boldsymbol{\alpha}_{j_1}(t_i), \boldsymbol{\alpha}_{j_2}(t_{i+1})\right)$$

  holds.

In addition, we adapt our problem whenever non-binary values of $\boldsymbol{\alpha}(\cdot)$ occur upon the accuracy of the regularization term γ. In this case we either add the term (19) to the objective function if it was not already present, or raise π_2 (see (19)) by the factor 10. With this, we aim to achieve binary-valued $\boldsymbol{\alpha}_j(\cdot)$, which together with the penalization of the switching indicators finally yield binary-valued switching indicators (if the penalization parameter π for the switch costs is big enough).

Results

We use IPOPT [38] with standard settings except for the accuracy, which is set to 10^{-6}, in order to solve the arising NLPs. All generalized coordinates are approximated by polynomials of degree 3 and both controls $\mathbf{u}_j(t)$ by piecewise linear functions. We choose $\pi = 5$ and $\gamma_0 = 10^{-3}$. In this problem, we find $\gamma_{acc} = 2 \cdot 10^{-4}$ to be a suitable value. For the chosen initial values we receive the optimal solution depicted in Fig. 7 and Fig. 8 after $k = 6$ iterations. The solver determines the end time to be $t_f \approx 5.334$. A visualization of the postures of the walker is seen in Fig. 9.
Observe, that one cannot assume the chosen switching structure to be optimal. Depending on the initial values, the chosen NLP solver finds a feasible switching structure by means of the $\boldsymbol{\alpha}_j(\cdot)$ and determines optimal controls $\mathbf{u}_j(\cdot)$ for this specific structure.

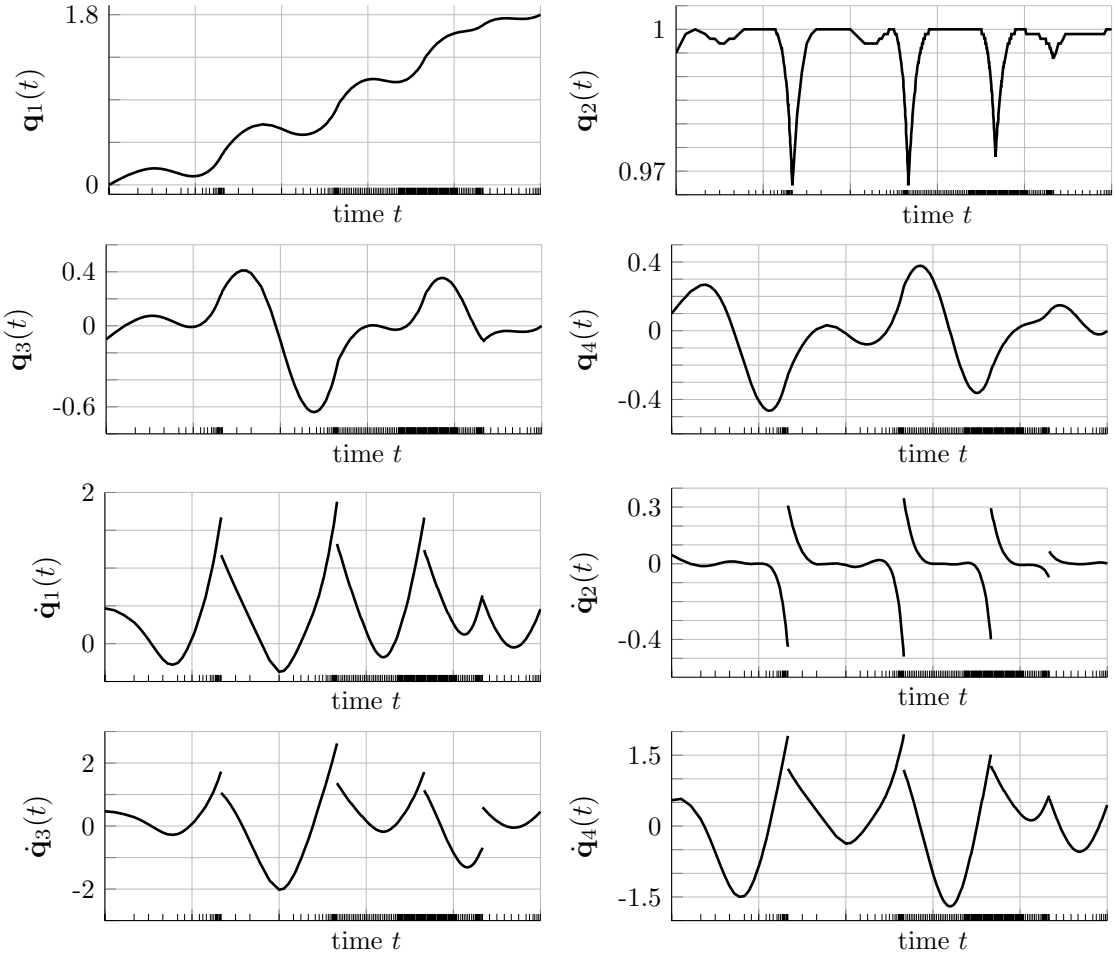
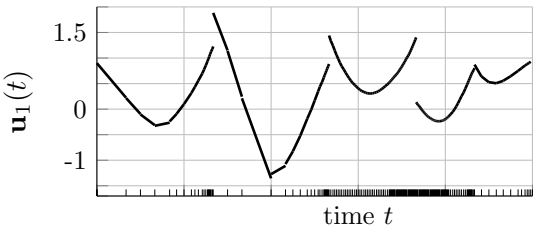
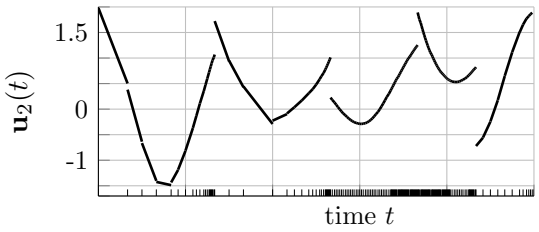


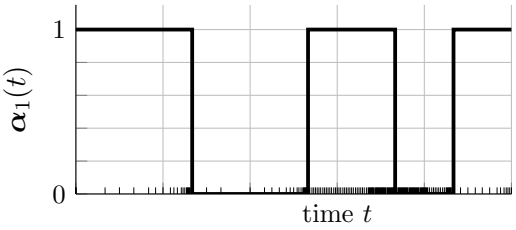
Figure 7: Generalized coordinates and velocities in the optimal solution, cf. Tab. 1 for their meaning. Jumps occur in the generalized velocities $\dot{\mathbf{q}}$ whenever one foot hits the ground. An accumulation of grid points in the vicinity of collision impact time points is produced by the grid adaption strategy to precisely locate the impact times in the collocation system.



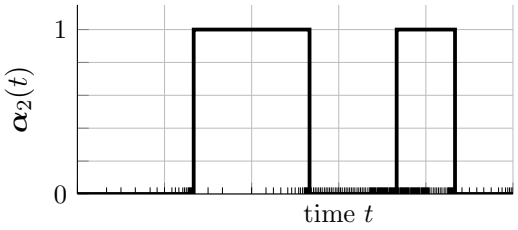
(a) Left foot torque $\boldsymbol{\tau}_1(t)$.



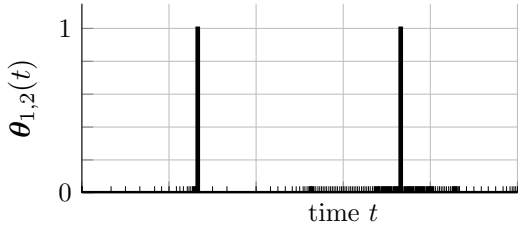
(b) Right foot torque $\boldsymbol{\tau}_2(t)$.



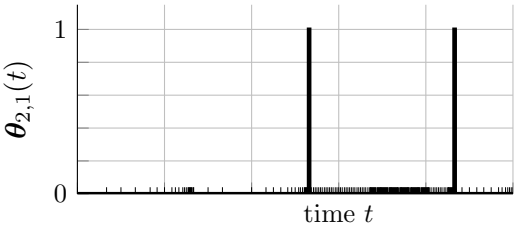
(c) Indicator $\boldsymbol{\alpha}_1(\cdot)$ for mode 1; left foot on the ground.



(d) Indicator $\boldsymbol{\alpha}_2(\cdot)$ for mode 2; right foot on the ground.



(e) $\boldsymbol{\theta}_{1,2}(\cdot)$ indicating switches from mode 1 to mode 2.

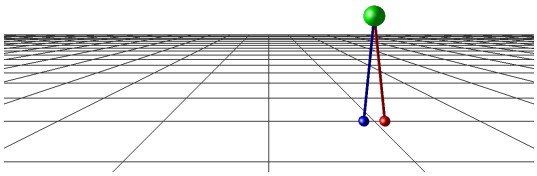


(f) $\boldsymbol{\theta}_{2,1}(\cdot)$ indicating switches from mode 2 to mode 1.

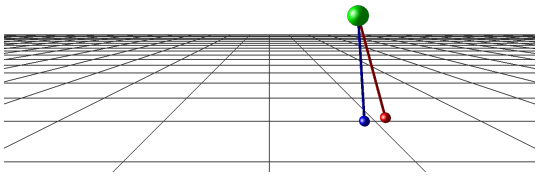
Figure 8: Trajectories of controls $\mathbf{u}_j(\cdot) = \boldsymbol{\tau}_j(\cdot)$, mode-indicators $\boldsymbol{\alpha}(\cdot)$ and mode-change-indicators $\boldsymbol{\theta}(\cdot)$ in the optimal solution.

8 Conclusion and Outlook

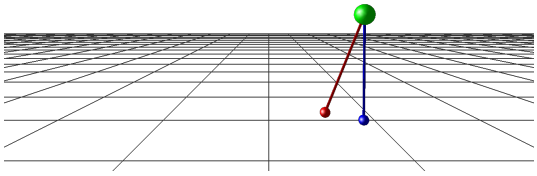
In this article, we have presented a novel approach for solving OCPs constrained by ODEs with discontinuities in the differential equations right hand side as well as in the differential states, and switch costs, which permits the dynamical identification of number and order of model-stages. Our approach is based on binary indicator functions, on the one hand to mark the mode of the system, and on the other hand to mark a change of modes. To solve the problem numerically, we use a direct and simultaneous adaptive collocation approach to optimal control, which results in an MPVC. We have applied our approach in a mechanical example to generate a walking-like motion and have shown, that it indeed leads to physically reasonable results.
In future research, the following questions need to be addressed: can the used approach be extended in order to treat jumps and switch costs in case of inconsistent switching and Filippov solutions? Can one develop a generalized strategy for adaptive refinement and warm-start orchestrated with the vanishing constraint homotopy? It would also be appealing to derive convergence results in order to justify our approach theoretically. Furthermore, the approach calls for application in a more complex setting in order to test its suitability for model–based treatment planning of CP in praxis, as this was one of the key motivations for this project.



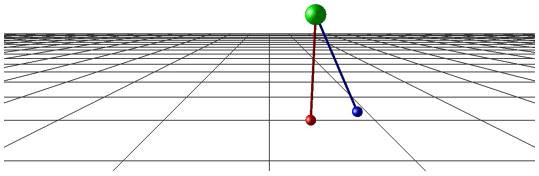
(a) $t = t_0$: walker in initial position.



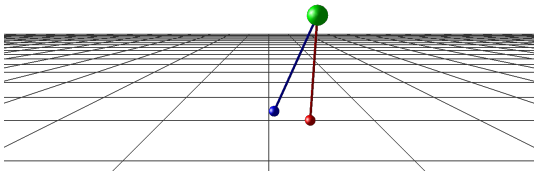
(b) $t \approx 0.437$: walker in mode 1. The left foot is fixed to the ground.



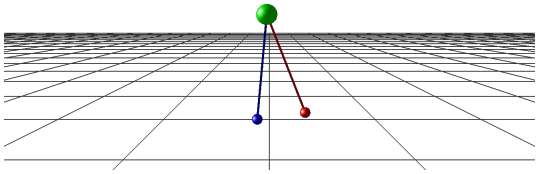
(c) $t \approx 1.038$.



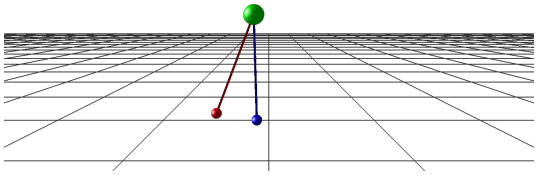
(d) $t \approx 1.662$: walker in mode 2 after a collision impact took place. Now, the right foot is fixed to the ground.



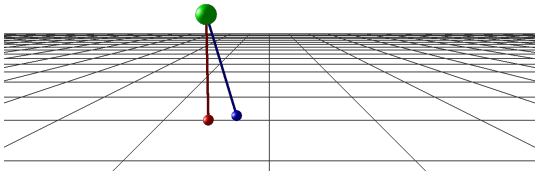
(e) $t \approx 2.301$: again, a change of modes occurred.



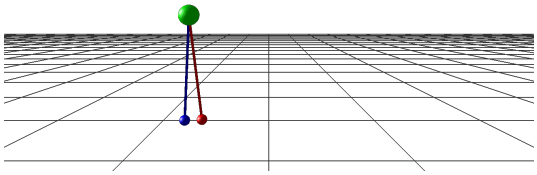
(f) $t \approx 3.004$



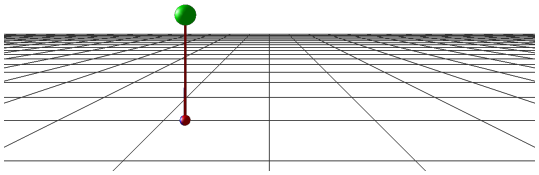
(g) $t \approx 3.733$



(h) $t \approx 4.206$



(i) $t \approx 4.868$



(j) $t = t_f$: walker in terminal position.

Figure 9: Postures of the walker at various time points. Visualization created with MeshUp [13].

Acknowledgments

C. Kirches and E. Kostina were supported by the German Federal Ministry of Education and Research, grants n° 05M17MBA-MoPhaPro (both), 05M18MBA-MOReNet and 61210304-ODINE (C. Kirches), 05M18VHA-MOReNet (E. Kostina). A. Meyer acknowledges funding by the European Research Council Adv. Inv. Grant MOBOCON 291 458 (H.G. Bock). C. Kirches, E. Kostina, and M. Schlöder acknowledge funding by Deutsche Forschungsgemeinschaft through Priority Programme 1962 "Non-smooth and Complementarity-based Distributed Parameter Systems: Simulation and Hierarchical Optimization".

References

[1] W. Achtziger and C. Kanzow, *Mathematical programs with vanishing constraints: optimality conditions and constraint qualifications*, Mathematical Programming Series A **114** (2008), 69–99.

[2] P. Antsaklis and X. Koutsoukos, *On hybrid control of complex systems: A survey*, In 3rd International Conference ADMP'98, Automation of Mixed Processes: Dynamic Hybrid Systems, March 1998, pp. 1–8.

[3] Stéphane Armand, Geraldo Decoulon, and Alice Bonnefoy-Mazure, *Gait analysis in children with cerebral palsy*, EFORT Open Reviews **1** (2016), no. 12, 448–460.

[4] B.T. Baumrucker and L.T. Biegler, *MPEC strategies for optimization of a class of hybrid dynamic systems*, Journal of Process Control **19** (2009), no. 8, 1248–1256, Special Section on Hybrid Systems: Modeling, Simulation and Optimization.

[5] S.C. Bengea and R.A. Decarlo, *Optimal control of switching systems*, Automatica **41** (2005), no. 1, 11–27.

[6] L.T. Biegler, *Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation*, Computers & Chemical Engineering **8** (1984), 243–248.

[7] H.G. Bock, *Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*, Bonner Mathematische Schriften, vol. 183, Rheinische Friedrich–Wilhelms–Universität Bonn, Bonn, 1987.

[8] H.G. Bock, C. Kirches, A. Meyer, and A. Potschka, *Numerical solution of optimal control problems with explicit and implicit switches*, Optimization Methods and Software **33** (2018), no. 3, 450–474.

[9] H.G. Bock and K.J. Plitt, *A Multiple Shooting algorithm for direct solution of optimal control problems*, Proceedings of the 9th IFAC World Congress (Budapest), Pergamon Press, 1984, pp. 242–247.

[10] U. Brandt-Pollmann, *Numerical solution of optimal control problems with implicitly defined discontinuities with applications in engineering*, Dissertation, Heidelberg University, 2004.

[11] S. Engell, G. Frehse, and E. Schnieder (eds.), *Modelling, analysis, and design of hybrid systems*, Springer Berlin Heidelberg, 2002.

[12] R. Featherstone, *Rigid body dynamics algorithms*, Springer US, 2008.

[13] M.L. Felis, *Modeling emotional aspects in human locomotion*, Dissertation, Heidelberg University, 2016.

[14] ———, *RBDL: An efficient rigid-body dynamics library using recursive algorithms*, Autonomous Robots **41** (2016), no. 2, 495–511.

[15] Mariano Garcia, *The simplest walking model: Stability, complexity, and scaling*, Journal of Biomechanical Engineering **120** (1998), no. 2, 281.

[16] D. Garg, M.A. Patterson, W.W. Hager, A.V. Rao, D.A. Benson, and G.T. Huntington, *An Overview of Three Pseudospectral Methods for the Numerical Solution of Optimal Control Problems*, Advances in the Astronautical Sciences **135** (2009), no. 1, 475–487.

[17] R. Goebel, R. Sanfelice, and A.R. Teel, *Hybrid dynamical systems*, IEEE Control Systems Magazine **29** (2009), no. 2, 28–93.

[18] M. Guignard, *Generalized Kuhn–Tucker conditions for mathematical programming problems in a Banach space*, SIAM Journal on Control **7** (1969), no. 2, 232–241.

[19] K. Hatz, *Efficient numerical methods for hierarchical dynamic optimization with application to cerebral palsy gait modeling*, Dissertation, Heidelberg University, 2014.

[20] I.A. Hiskens and M.A. Pai, *Trajectory sensitivity analysis of hybrid systems*, IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications **47** (2000), no. 2, 204–220.

[21] T. Hoheisel, *Mathematical Programs with Vanishing Constraints*, Dissertation, Julius–Maximilians–Universität Würzburg, 2009.

[22] G.T. Huntington, D. Benson, and A.V. Rao, *A comparison of accuracy and computational efficiency of three pseudospectral methods*, Proceedings of the AIAA Guidance, Navigation, and Control Conference, 2007, pp. 840–864.

[23] A.F. Izmailov and M.V. Solodov, *Mathematical Programs with Vanishing Constraints: Optimality Conditions, Sensitivity, and a Relaxation Method*, Journal of Optimization Theory and Applications **142** (2009), 501–532.

[24] M. Jung, *Relaxations and Approximations for Mixed-Integer Optimal Control*, Dissertation, Heidelberg University, 2013.

[25] C. Kirches, *A Numerical Method for Nonlinear Robust Optimal Control with Implicit Discontinuities and an Application to Powertrain Oscillations*, Diploma Thesis, Heidelberg University, October 2006.

[26] ———, *Fast numerical methods for mixed-integer nonlinear model-predictive control*, Dissertation, Heidelberg University, 2010.

[27] C. Kirches, F. Lenders, and P. Manns, *Approximation properties and tight bounds for constrained mixed-integer optimal control*, Optimization Online Preprint **5404** (2016), submitted.

[28] R.I. Leine, D.H. Van Campen, and B.L. Van De Vrande, *Bifurcations in nonlinear discontinuous systems*, Nonlinear Dynamics **23** (2000), no. 2, 105–164.

[29] F. Lenders, *Numerical methods for mixed-integer optimal control with combinatorial constraints*, Dissertation, Heidelberg University, 2017.

[30] O.L. Mangasarian and S. Fromovitz, *Fritz John necessary optimality conditions in the presence of equality and inequality constraints*, Journal of Mathematical Analysis and Applications **17** (1967), 37–47.

[31] P. Manns and C. Kirches, *Improved regularity assumptions for partial outer convexification of mipdecos*, Optimization Online Preprint **6585** (2018), (submitted to ESAIM: Control, Optimisation and Calculus of Variations).

[32] Katja Mombaur, *Optimal control for applications in medical and rehabilitation technology: Challenges and solutions*, Springer Optimization and Its Applications, Springer International Publishing, 2016, pp. 103–145.

[33] A. Saccon, N.V.D. Wouw, and H. Nijmeijer, *Sensitivity analysis of hybrid systems with state jumps with application to trajectory tracking*, IEEE Conference on Decision and Control (2014), no. 27, 3065–3070.

[34] S. Sager, *Numerical methods for mixed–integer optimal control problems*, Dissertation, Heidelberg University, 2006.

[35] O. von Stryk and M. Glocker, *Decomposition of Mixed-Integer Optimal Control Problems Using Branch and Bound and Sparse Direct Collocation*, Proc. ADPM 2000 – The 4th International Conference on Automatisation of Mixed Processes: Hybrid Dynamical Systems, 2000, pp. 99–104.

[36] P. Tabuada, *Verification and control of hybrid systems: a symbolic approach*, Springer Science & Business Media, 2009.

[37] Gerd Wachsmuth, *On LICQ and the uniqueness of lagrange multipliers*, Operations Research Letters **41** (2013), no. 1, 78–80.

[38] A. Wächter and L.T. Biegler, *On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming*, Mathematical Programming **106** (2006), no. 1, 25–57.

[39] F. Zhu and P.J. Antsaklis, *Optimal control of hybrid switched systems: A brief survey*, Discrete Event Dynamic Systems **25** (2015), no. 3, 345–364.

Appendix A Proofs of Lemmata and Propositions

A.1 Proof of Lemma 2.2

Let $w \in PC_{\bar{\delta}}(\mathcal{T}, \{1, ..., n\})$. We set $\boldsymbol{\omega}(t) = \left(\delta_{j\, w(t)}\right)_j$ using the *Kronecker delta*. Then obviously we have $\boldsymbol{\omega}(\cdot) \in PC_{\bar{\delta}}(\mathcal{T}, \{0,1\}^n)$. Let $t \in \mathcal{T}$ and $w(t) = j'$. Then $\sum_{j=1}^n \boldsymbol{\omega}_j(t) = \boldsymbol{\omega}_{j'}(t) = 1$, ergo $\boldsymbol{\omega}(\cdot) \in S^n_{\mathcal{F}}$, and $\sum_{j=1}^n \boldsymbol{\omega}_j(t) \cdot j = \boldsymbol{\omega}_{j'}(t) \cdot j' = j' = w(t)$. Hence φ is surjective.
To show the injectivity, let $\boldsymbol{\omega}^1(\cdot), \boldsymbol{\omega}^2(\cdot) \in S^n_{\mathcal{F}} \cap PC_{\bar{\delta}}(\mathcal{T}, \{0,1\}^n)$ with $\boldsymbol{\omega}^1(\cdot) \neq \boldsymbol{\omega}^2(\cdot)$. Then there is a $t \in \mathcal{T}$ and distinct indices $j_1, j_2 \in \{1, ..., n\}$ such that $\boldsymbol{\omega}^1_{j_1}(t) = 1 = \boldsymbol{\omega}^2_{j_2}(t)$, and all other entries are zero respectively. Hence

$$\varphi(\boldsymbol{\omega}^1)(t) = \sum_{j=1}^n \boldsymbol{\omega}^1_j(t) \cdot j = j_1 \neq j_2 = \varphi(\boldsymbol{\omega}^2)(t)\,,$$

which finishes the proof.

A.2 Proof of Proposition 2.3

For the first direction, let $(\mathbf{x}, \mathbf{u}, w)$ be feasible for Problem (1). We set $\boldsymbol{\omega}(t) = \left(\delta_{j\, w(t)}\right)_j$. By the proof of Lemma 2.2 we know $\boldsymbol{\omega} = \varphi^{-1}(w)$, and by construction we have $\mathcal{S}(w) = \mathcal{S}(\boldsymbol{\omega})$. Thus the values of both cost functions coincide. Since $\boldsymbol{\omega}_j(t) = \delta_{j\, w(t)}$ for all $t \in \mathcal{T}$ we have

$$\mathbf{f}^j(\mathbf{x}(t), \mathbf{u}(t)) = \sum_{j=1}^n \boldsymbol{\omega}_j(t) \cdot \mathbf{f}^j(\mathbf{x}(t), \mathbf{u}(t)) \quad \text{if } w(t) = j\,,$$

and the differential right sides of both problems coincide almost everywhere. Since the omitted constraints (1e) and (1f) are not affected by the reformulation, it remains to show that (2d) holds. Let $t \in \mathcal{T}$ such that $w(t) = j'$ and $\mathbf{0} \geq \mathbf{c}^{j'}(\mathbf{x}(t), \mathbf{u}(t))$. Then $\boldsymbol{\omega}_{j'}(t) = 1$ and $\boldsymbol{\omega}_j(t) = 0$ for all $j \neq j'$. Therefore

$$\mathbf{0} \geq \boldsymbol{\omega}_j(t) \cdot \mathbf{c}^j(\mathbf{x}(t), \mathbf{u}(t))$$

indeed holds for all $j \in \{1, \ldots, n\}$. The proof for reverse direction works similarly.

A.3 Proof of Proposition 3.1

We take a look at the first statement. Let $(\mathbf{x}(\cdot), \mathbf{u}(\cdot), \boldsymbol{\omega}(\cdot))$ be feasible for Problem (2) and define the $\boldsymbol{\theta}_{j_1,j_2}(\cdot)$ by (5c) and (5d). For each $t_s \in \mathcal{S}(\boldsymbol{\omega})$, we have

$$\boldsymbol{\Delta}\left(\mathbf{x}(t_s^-), (\boldsymbol{\theta}_{j_1,j_2}(t_s))_{j_1,j_2}\right) = \boldsymbol{\Delta}_{j_1,j_2}\left(\mathbf{x}(t_s^-)\right) \quad \text{if } j_1 \to_{\boldsymbol{\omega}} j_2 \text{ at } t_s\,. \tag{38}$$

Indeed, if $j_1 \to_{\boldsymbol{\omega}} j_2$, we have $\boldsymbol{\omega}_{j_1}(t_s^-) = \boldsymbol{\omega}_{j_2}(t_s^+) = 1$, $\boldsymbol{\omega}_{j'}(t_s^-) = 0$ for all $j' \neq j_1$ and $\boldsymbol{\omega}_{j'}(t_s^+) = 0$ for all $j' \neq j_2$. Due to (5d), we therefore have

$$\boldsymbol{\theta}_{j_1',j_2'}(t_s) = \begin{cases} 1 & \text{if } j_1' = j_1 \text{ and } j_2' = j_2 \\ 0 & \text{else} \end{cases}\,,$$

and (38) holds as one can easily verify. For $t \in \mathcal{P} \setminus \mathcal{S}(\boldsymbol{\omega})$ on the other hand, we have $\boldsymbol{\theta}_{j_1,j_2}(t) = 0$ for all $j_1 \neq j_2$ according to (5d). This yields

$$\boldsymbol{\Delta}\left(\mathbf{x}(t^-), (\boldsymbol{\theta}_{j_1,j_2}(t))_{j_1,j_2}\right) = \mathbf{x}(t^-)\,,$$

and no jump in the differential states occurs, as desired. Therefore $(\mathbf{x}(\cdot), \mathbf{u}(\cdot), \boldsymbol{\omega}(\cdot), (\boldsymbol{\theta}_{j_1,j_2}(\cdot))_{j_1 \neq j_2})$ is feasible for Problem (5), and the cost function values coincide because of (3).
The second statement can be proven in a similar fashion.

A.4 Proof of Proposition 3.2

For every $\alpha_1, \alpha_2 \in \mathbb{R}$, there exists a $\beta \in [0,1]$ such that

$$\min(\alpha_1, \alpha_2) = \beta\alpha_1 + (1-\beta)\alpha_2\,.$$

Using this and Proposition 3.1, the statement follows.

A.5 Proof of Proposition 4.1

Take a look at ϕ_{subs}. Then

$$\sum_{j=1}^{n} \min(\mathbf{b}_j, 1-\mathbf{a}_j) \leq \sum_{j=1}^{n} \mathbf{b}_j = 1\,,$$

since $\mathbf{b} \in \mathrm{conv}\,(\mathbb{S}^n)$. Now define

$$J_1 := \{j \in \{1,\dots,n\} \mid \mathbf{a}_j + \mathbf{b}_j \leq 1\} \quad \text{and} \quad J_2 := \{1,\dots,n\} \setminus J_1\,.$$

Then we find

$$\phi_{subs}(\mathbf{a},\mathbf{b}) = \sum_{j=1}^{n} \min(\mathbf{b}_j, 1-\mathbf{a}_j) = \sum_{j\in J_1} \mathbf{b}_j + \sum_{j\in J_2} (1-\mathbf{a}_j)\,.$$

We see: if $\mathbf{a}_j + \mathbf{b}_j \leq 1$ for all j, i.e. $J_1 = \{1,\dots,n\}$ and $J_2 = \emptyset$, then $\phi_{subs}(\mathbf{a},\mathbf{b}) = 1$. For ϕ_{inv}, the proof works similar.
For ϕ_{omni} we have

$$\phi_{omni}(\mathbf{a},\mathbf{b}) = \sum_{\substack{j_1,j_2=1\\ j_1\neq j_2}}^{n} \min(\mathbf{a}_{j_1}, \mathbf{b}_{j_2}) \leq \sum_{\substack{j_1,j_2=1\\ j_1\neq j_2}}^{n} \mathbf{a}_{j_1} = (n-1)\sum_{j_1=1}^{n} \mathbf{a}_{j_1} = n-1 \tag{39}$$

and if we set $\mathbf{a}_j = \mathbf{b}_j = \frac{1}{n}$ for all j, the inequality in (39) becomes an equality, which closes the proof.

A.6 Proof of Proposition 4.2

The first statement is obviously true. For the proof of the second statement, we first take a look at '$\Leftarrow$': If $\mathbf{a},\mathbf{b} \in \mathbb{S}^n$ and $\mathbf{a} = \mathbf{b}$, for every j we have either $\mathbf{a}_j = \mathbf{b}_j = 1$ or $\mathbf{a}_j = \mathbf{b}_j = 0$. Therefore $\phi_{inv}(\mathbf{a},\mathbf{b}) = \phi_{subs}(\mathbf{a},\mathbf{b}) = 0$. Also for every pair (j_1, j_2) with $j_1 \neq j_2$ either $\mathbf{a}_{j_1} = 0$ or $\mathbf{b}_{j_2} = 0$, which is why $\phi_{omni}(\mathbf{a},\mathbf{b}) = 0$.
'$\Rightarrow$': We first take a look at ϕ_{inv}. If $\phi_{inv}(\mathbf{a},\mathbf{b}) = 0$, then $\min(\mathbf{a}_j + \mathbf{b}_j, 2 - \mathbf{a}_j - \mathbf{b}_j) = 0$ for all j. Since $\mathbf{a}_j, \mathbf{b}_j \in [0,1]$, this is only possible if $\mathbf{a}_j = \mathbf{b}_j \in \{0,1\}$ for all j. Therefore $\mathbf{a} = \mathbf{b}$, and since $\mathbf{a},\mathbf{b} \in \mathrm{conv}\,(\mathbb{S}^n)$, the statement is true for ϕ_{inv}.
Next we consider ϕ_{subs}. Similar as before, if $\phi_{subs}(\mathbf{a},\mathbf{b}) = 0$ then $\min(\mathbf{b}_j, 1-\mathbf{a}_j) = 0$ for all j. Thus for every j either $\mathbf{b}_j = 0$ or $\mathbf{a}_j = 1$. Since $\mathbf{b} \in \mathrm{conv}\,(\mathbb{S}^n)$, there must be a j with $\mathbf{b}_j > 0$, and hence $\mathbf{a}_j = 1$. Since $\mathbf{a} \in \mathrm{conv}\,(\mathbb{S}^n)$, it follows $\mathbf{a}_{j'} = 0$ for all $j' \neq j$. In particular, $\mathbf{a}_{j'} \neq 1$ for all $j' \neq j$, and therefore $\mathbf{b}_{j'} = 0$ for all $j' \neq j$, ergo $\mathbf{b}_j = 1$, which shows the result for ϕ_{subs}.
If $\phi_{omni}(\mathbf{a},\mathbf{b}) = 0$, one has $\min(\mathbf{a}_{j_1}, \mathbf{b}_{j_2}) = 0$ for all (j_1, j_2) with $j_1 \neq j_2$. Since $\mathbf{a} \in \mathrm{conv}\,(\mathbb{S}^n)$, there is a j with $\mathbf{a}_j > 0$. This yields $\mathbf{b}_{j'} = 0$ for all $j' \neq j$ and therefore $\mathbf{b}_j = 1$. Now using the same arguments again, we can conclude $\mathbf{a}_j = 1$ and $\mathbf{a}_{j'} = 0$ for all $j' \neq j$, in particular $\mathbf{a} = \mathbf{b} \in \mathbb{S}^n$.

A.7 Proof of Proposition 4.3

To proof the statement, we take a look at several distinct cases. Lets first consider $i = inv$. It is sufficient to show

$$\min(\mathbf{a}_j + \mathbf{c}_j, 2 - \mathbf{a}_j - \mathbf{c}_j) \leq \min(\mathbf{a}_j + \mathbf{b}_j, 2 - \mathbf{a}_j - \mathbf{b}_j) + \min(\mathbf{b}_j + \mathbf{c}_j, 2 - \mathbf{b}_j - \mathbf{c}_j)$$

for all j.
Case i) Let $\min(\mathbf{a}_j + \mathbf{c}_j, 2 - \mathbf{a}_j - \mathbf{c}_j) = \mathbf{a}_j + \mathbf{c}_j$, i.e. $\mathbf{a}_j + \mathbf{c}_j \leq 2 - \mathbf{a}_j - \mathbf{c}_j$. Then

$$\begin{aligned}
&\mathbf{a}_j + \mathbf{c}_j \leq (\mathbf{a}_j + \mathbf{b}_j) + (\mathbf{b}_j + \mathbf{c}_j)\,,\\
&\mathbf{a}_j + \mathbf{c}_j \leq \mathbf{a}_j + 2 - \mathbf{c}_j = (\mathbf{a}_j + \mathbf{b}_j) + (2 - \mathbf{b}_j - \mathbf{c}_j)\,,\\
&\mathbf{a}_j + \mathbf{c}_j \leq 2 - \mathbf{a}_j + \mathbf{c}_j = (2 - \mathbf{a}_j - \mathbf{b}_j) + (\mathbf{b}_j + \mathbf{c}_j)\,,\\
&\mathbf{a}_j + \mathbf{c}_j \leq 2 - \mathbf{a}_j - \mathbf{c}_j \leq 2 - \mathbf{a}_j - \mathbf{c}_j + 2 - 2\mathbf{b}_j = (2 - \mathbf{a}_j - \mathbf{b}_j) + (2 - \mathbf{b}_j - \mathbf{c}_j)\,.
\end{aligned}$$

Case ii) Let $\min(\mathbf{a}_j + \mathbf{c}_j, 2 - \mathbf{a}_j - \mathbf{c}_j) = 2 - \mathbf{a}_j - \mathbf{c}_j$. We get

$$\begin{aligned}
&2 - \mathbf{a}_j - \mathbf{c}_j \leq \mathbf{a}_j + \mathbf{c}_j \leq (\mathbf{a}_j + \mathbf{b}_j) + (\mathbf{b}_j + \mathbf{c}_j)\,,\\
&2 - \mathbf{a}_j - \mathbf{c}_j \leq \mathbf{a}_j + 2 - \mathbf{c}_j = (\mathbf{a}_j + \mathbf{b}_j) + (2 - \mathbf{b}_j - \mathbf{c}_j)\,,\\
&2 - \mathbf{a}_j - \mathbf{c}_j \leq 2 - \mathbf{a}_j + \mathbf{c}_j = (2 - \mathbf{a}_j - \mathbf{b}_j) + (\mathbf{b}_j + \mathbf{c}_j)\,,\\
&2 - \mathbf{a}_j - \mathbf{c}_j \leq 2 - \mathbf{a}_j - \mathbf{c}_j + 2 - 2\mathbf{b}_j = (2 - \mathbf{a}_j - \mathbf{b}_j) + (2 - \mathbf{b}_j - \mathbf{c}_j)\,.
\end{aligned}$$

Altogether, we see

$$\min(\mathbf{a}_j + \mathbf{c}_j, 2 - \mathbf{a}_j - \mathbf{c}_j) \leq \min(\mathbf{a}_j + \mathbf{b}_j, 2 - \mathbf{a}_j - \mathbf{b}_j) + \min(\mathbf{b}_j + \mathbf{c}_j, 2 - \mathbf{b}_j - \mathbf{c}_j)\,,$$

which proves the first statement for $i = inv$.
Now consider $i = subs$. It is sufficient to show

$$\min(\mathbf{c}_j, 1 - \mathbf{a}_j) \leq \min(\mathbf{b}_j, 1 - \mathbf{a}_j) + \min(\mathbf{c}_j, 1 - \mathbf{b}_j)$$

for all j.
Case i) Let $\min(\mathbf{c}_j, 1 - \mathbf{a}_j) = \mathbf{c}_j$. Then

$$\begin{aligned}
&\mathbf{c}_j \leq \mathbf{b}_j + \mathbf{c}_j,\\
&\mathbf{c}_j \leq 1 = \mathbf{b}_j + (1 - \mathbf{b}_j)\,,\\
&\mathbf{c}_j \leq (1 - \mathbf{a}_j) + \mathbf{c}_j\,,\\
&\mathbf{c}_j \leq 1 - \mathbf{a}_j \leq (1 - \mathbf{a}_j) + (1 - \mathbf{b}_j)\,.
\end{aligned}$$

Case ii) Now let $\min(\mathbf{c}_j, 1 - \mathbf{a}_j) = 1 - \mathbf{a}_j$. We find

$$\begin{aligned}
&1 - \mathbf{a}_j \leq \mathbf{c}_j \leq \mathbf{b}_j + \mathbf{c}_j,\\
&1 - \mathbf{a}_j \leq 1 = \mathbf{b}_j + (1 - \mathbf{b}_j)\,,\\
&1 - \mathbf{a}_j \leq (1 - \mathbf{a}_j) + \mathbf{c}_j\,,\\
&1 - \mathbf{a}_j \leq (1 - \mathbf{a}_j) + (1 - \mathbf{b}_j)\,.
\end{aligned}$$

Altogether

$$\min(\mathbf{c}_j, 1 - \mathbf{a}_j) \leq \min(\mathbf{b}_j, 1 - \mathbf{a}_j) + \min(\mathbf{c}_j, 1 - \mathbf{b}_j)\,,$$

which shows (13) for $i = subs$.

For $i = omni$, the triangle inequality does not hold in general. As a counterexample, consider

$$\mathbf{a} = \begin{pmatrix} \frac{1}{3} \\ \frac{3}{3} \\ \frac{2}{3} \\ 0 \end{pmatrix}\,,\ \mathbf{b} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}\,,\ \mathbf{c} = \begin{pmatrix} \frac{2}{5} \\ \frac{3}{5} \\ \frac{2}{5} \\ \frac{1}{5} \end{pmatrix}\,.$$

We get

$$\begin{aligned}
\phi_{omni}(\mathbf{a}, \mathbf{c}) &= \sum_{\substack{j_1, j_2 = 1 \\ j_1 \neq j_2}}^{3} \min(\mathbf{a}_{j_1}, \mathbf{c}_{j_2}) = \left(\frac{1}{3} + \frac{1}{5}\right) + \left(\frac{2}{5} + \frac{1}{5}\right) + 0 > \frac{1}{3} + \frac{3}{5}\,,\\
\phi_{omni}(\mathbf{a}, \mathbf{b}) &= \sum_{\substack{j_1, j_2 = 1 \\ j_1 \neq j_2}}^{3} \min(\mathbf{a}_{j_1}, \mathbf{b}_{j_2}) = \frac{1}{3} + 0 + 0 = \frac{1}{3}\,,\\
\phi_{omni}(\mathbf{b}, \mathbf{c}) &= \sum_{\substack{j_1, j_2 = 1 \\ j_1 \neq j_2}}^{3} \min(\mathbf{b}_{j_1}, \mathbf{c}_{j_2}) = 0 + \left(\frac{2}{5} + \frac{1}{5}\right) + 0 = \frac{3}{5}\,,
\end{aligned}$$

and the triangle inequality does not hold.
To prove the last statement, let $\mathbf{a}, \mathbf{c} \in \mathbb{S}^n$ with $\mathbf{a}_l = \mathbf{c}_k = 1$ and $\mathbf{b} \in \operatorname{conv}(\mathbb{S}^n)$. Then

$$\phi_{omni}(\mathbf{a}, \mathbf{b}) = \sum_{j_1 \neq j_2} \min(\mathbf{a}_{j_1}, \mathbf{b}_{j_2}) = \sum_{j_2 \neq l} \min(1, \mathbf{b}_{j_2}) = 1 - \mathbf{b}_l \tag{40}$$

and similar $\phi_{omni}(\mathbf{b},\mathbf{c}) = 1 - \mathbf{b}_k$. For $\mathbf{a} = \mathbf{c}$, we have $\phi_{omni}(\mathbf{a},\mathbf{c}) = 0$ according to Proposition 4.2, and for $\mathbf{a} \neq \mathbf{c}$ obviously $\phi_{omni}(\mathbf{a},\mathbf{c}) = 1$. This yields

$$\phi_{omni}(\mathbf{a},\mathbf{c}) \leq 1 \leq 2 - (\mathbf{b}_l + \mathbf{b}_k) = (1 - \mathbf{b}_l) + (1 - \mathbf{b}_k) \overset{(40)}{=} \phi_{omni}(\mathbf{a},\mathbf{b}) + \phi_{omni}(\mathbf{b},\mathbf{c}) \tag{41}$$

which finally closes the proof.

A.8 Proof of Proposition 4.4

For $i \in \{inv, subs\}$ we have $\phi_i(\mathbf{a},\mathbf{c}) = 1$ by Proposition 4.1, and for $i = omni$, this is also true, as one directly verifies.
Let us consider $i = omni$ first. Due to (41), statement (14) is equivalent to the statement

$$1 = 2 - (\mathbf{b}_l + \mathbf{b}_k) \quad \Longleftrightarrow \quad \mathbf{b}_l + \mathbf{b}_k = 1\,, \tag{42}$$

which is obviously true.
Next we take a look at $i = inv$. We find

$$\begin{aligned} 2\,\phi_{inv}(\mathbf{a},\mathbf{b}) &= \sum_{j=1}^{n} \min(\mathbf{a}_j + \mathbf{b}_j, 2 - \mathbf{a}_j - \mathbf{b}_j) \\ &= \min(1 + \mathbf{b}_l, 1 - \mathbf{b}_l) + \sum_{j \neq l} \min(\mathbf{b}_j, 2 - \mathbf{b}_j) = 1 - \mathbf{b}_l + \sum_{j \neq l} \mathbf{b}_j \\ &= 1 - 2\mathbf{b}_l + \sum_{j=1}^{n} \mathbf{b}_j = 2 - 2\mathbf{b}_l\,, \end{aligned}$$

hence $\phi_{inv}(\mathbf{a},\mathbf{b}) = 1 - \mathbf{b}_l$, and similarly $\phi_{inv}(\mathbf{b},\mathbf{c}) = 1 - \mathbf{b}_k$. Again, statement (14) reduces to the valid statement (42).
For $i = subs$, the proof works similar.

A.9 Proof of Proposition 4.5

Since $n = 2$, we have $\mathbf{a}_1 + \mathbf{a}_2 = 1 = \mathbf{b}_1 + \mathbf{b}_2$. Therefore

$$\phi_{subs}(\mathbf{a},\mathbf{b}) = \min(\mathbf{b}_1, 1 - \mathbf{a}_1) + \min(\mathbf{b}_2, 1 - \mathbf{a}_2) = \min(\mathbf{b}_1, \mathbf{a}_2) + \min(\mathbf{b}_2, \mathbf{a}_1) = \phi_{omni}(\mathbf{a},\mathbf{b})\,.$$

Since $\mathbf{a}_2 + \mathbf{b}_2 = 2 - \mathbf{a}_1 - \mathbf{b}_1$ and $2 - \mathbf{a}_2 - \mathbf{b}_2 = \mathbf{a}_1 + \mathbf{b}_1$, we furthermore have

$$\begin{aligned} \phi_{inv}(\mathbf{a},\mathbf{b}) &= \frac{1}{2}\left[\min(\mathbf{a}_1 + \mathbf{b}_1, 2 - \mathbf{a}_1 - \mathbf{b}_1) + \min(\mathbf{a}_2 + \mathbf{b}_2, 2 - \mathbf{a}_2 - \mathbf{b}_2)\right] \\ &= \min(\mathbf{a}_1 + \mathbf{b}_1, 2 - \mathbf{a}_1 - \mathbf{b}_1) = \begin{cases} \mathbf{a}_1 + \mathbf{b}_1 & \text{if } \mathbf{a}_1 + \mathbf{b}_1 \leq 1 \\ 2 - \mathbf{a}_1 - \mathbf{b}_1 & \text{if } \mathbf{a}_1 + \mathbf{b}_1 > 1 \end{cases}. \end{aligned}$$

On the other hand, we also find

$$\begin{aligned} \phi_{omni}(\mathbf{a},\mathbf{b}) &= \min(\mathbf{a}_1, \mathbf{b}_2) + \min(\mathbf{a}_2, \mathbf{b}_1) = \min(\mathbf{a}_1, 1 - \mathbf{b}_1) + \min(1 - \mathbf{a}_1, \mathbf{b}_1) \\ &= \min(\mathbf{a}_1 + \mathbf{b}_1, 1) - \mathbf{b}_1 + \min(1, \mathbf{b}_1 + \mathbf{a}_1) - \mathbf{a}_1 \\ &= 2\min(\mathbf{a}_1 + \mathbf{b}_1, 1) - (\mathbf{a}_1 + \mathbf{b}_1) = \begin{cases} \mathbf{a}_1 + \mathbf{b}_1 & \text{if } \mathbf{a}_1 + \mathbf{b}_1 \leq 1 \\ 2 - \mathbf{a}_1 - \mathbf{b}_1 & \text{if } \mathbf{a}_1 + \mathbf{b}_1 > 1 \end{cases}, \end{aligned}$$

and hence $\phi_{inv}(\mathbf{a},\mathbf{b}) = \phi_{omni}(\mathbf{a},\mathbf{b})$, which completes the proof.