Numerical methods for mixed–integer optimal control problems

Sebastian Sager

Interdisciplinary Center for Scientific Computing,
Universität Heidelberg

This thesis was supervised by Professor Dr. Dr. h.c. Hans Georg Bock and Professor Dr. Gerhard Reinelt, Universität Heidelberg. It was handed in on December 21th, 2005 and defended on February 10th, 2006.
This thesis was published and can be ordered in bookform, see Sager (2005).

Contents

Zusammenfassung

Das Ziel der vorgelegten Arbeit ist die Entwicklung von numerischen Methoden zur Lösung gemischt–ganzzahliger optimaler Steuerungsprobleme. Sie führt dabei in die Grundlagen der optimalen Steuerung und der ganzzahligen Programmierung ein, um auf diesen aufbauend einen neuen Algorithmus zu entwickeln. Dieser ist durch theoretische Resultate motiviert und basiert auf Bocks direkter Mehrzielmethode, einer Konvexifizierung wie Relaxierung des Ausgangsproblemes, einer adaptiven Verfeinerung des unterliegenden Kontrolldiskretisierungsgitters und ganzzahligen Methoden heuristischer oder deterministischer Art. Seine Anwendbarkeit wird anhand einer Vielzahl von Referenzproblemen aus der Literatur und erstmals lösbaren Anwendungsproblemen aufgezeigt. Die in dieser Arbeit vorgestellten Neuerungen beinhalten

- einen rigorosen Beweis, dass die optimale Lösung eines konvexifizierten und relaxierten Steuerungsproblemes eine untere Schranke liefert, die beliebig genau durch eine ganzzahlige Lösung approximiert werden kann. Dieses wird für eine sehr allgemeine Problemklasse gezeigt, in der die rechte Seite nichtlinear von differentiellen und algebraischen Zuständen wie von Parametern und gewöhnlichen Steuerfunktionen abhängen kann.

- einen auf diesem Beweis beruhenden Algorithmus, der unter gewissen Bedingungen und Vorgabe einer Toleranz eine ganzzahlige Lösung mit einem Zielfunktionswert liefert, der dichter als diese Toleranz am optimal erreichbaren Wert liegt.

- neue Heuristiken, die eine Enumeration der ganzzahligen Variablen vermeiden und durch eine Kombination mehrerer Konzepte die Strukturen von optimalen Lösungen relaxierter Probleme ausnutzen.

- die Lösung mehrerer, aus unterschiedlichsten Gründen anspruchsvoller Optimierungsaufgaben. Die in dieser Arbeit gelösten Steuerungsprobleme beinhalten Transitionsstufen, gekoppelte und entkoppelte Innere Punkte Gleichungs– und Ungleichungsbeschränkungen, Pfad– und Steuerbeschränkungen, differentielle und algebraische Variablen, zeitunabhängige Parameter, freie Stufendauern und kontinuierliche wie ganzzahlige Steuerfunktionen. Es werden ferner Probleme behandelt, die extrem instabil sind oder zustandsabhängige Unstetigkeiten aufweisen.

- die Entwicklung eines Softwarepaketes, mit dem gemischt–ganzzahlige optimale Steuerungsprobleme effizient und generisch gelöst werden können, ohne analytische Vorarbeiten leisten zu müssen.

Ein wichtiges Ergebnis dieser Arbeit ist, dass gemischt–ganzzahlige optimale Steuerungsprobleme, trotz der hohen Komplexität der Problemklasse vom theoretischen Standpunkt aus, in der Praxis oft ohne exponentielle Laufzeiten lösbar sind.

Abstract

This thesis aims at developing numerical methods for mixed–integer optimal control problems. Based on the foundations of optimal control and integer programming a new algorithm is developed. This algorithm is motivated by theoretical results and based on Bock's direct multiple shooting method, a convexification and relaxation of the original problem, an adaptive refinement of the underlying control discretization grid and deterministic as well as heuristic integer methods. Its applicability is shown by a number of reference problems from the literature and applications that can be solved for the first time. The novelties presented in this thesis include

- a rigorous proof that the optimal solution of a convexified and relaxed control problem yields a lower bound that can be approximated arbitrarily close by an integer solution. This is shown for a very general problem class, in which the right hand side may depend nonlinear on differential and algebraic states as on parameters and ordinary control functions.

- an algorithm based upon this proof that, under certain conditions and given a tolerance, yields an integer solution that has an objective function value closer than the prescribed tolerance to the lower bound.

- novel heuristics that avoid an enumeration of the integer variables and exploit the structures of optimal solutions of relaxed problems by a combination of several concepts.

- the solution of several, for different reasons challenging optimization tasks. The control problems that are being solved in this work contain transition stages, coupled and decoupled interior point inequality and equality constraints, path and control constraints, differential and algebraic variables, time–independent parameters, free stage lengths and continuous as well as binary control functions. Furthermore we treat problems that are extremely unstable and contain state–dependent discontinuities.

- the development of a software package that solves efficiently and generically mixed–integer optimal control problems, without the need for analytic a priori work.

One important result of this work is that mixed–integer optimal control problems can, despite the high complexity of the problem class from a theoretical point of view, in practice often be solved without exponential running times.

Chapter 0

Introduction

Mathematical modeling, simulation and optimization techniques had a great impact in the history of mankind and helped to understand and improve many processes of different kinds. The term *process* in the broadest sense is understood as defined in the online encyclopedia Wikipedia (2005):

> *Process (lat. processus - movement) is a naturally occurring or designed sequence of operations or events, possibly taking up time, space, expertise or other resource, which produces some outcome. A process may be identified by the changes it creates in the properties of one or more objects under its influence.*

Since first pioneering works, most of them in the middle of the last century, more and more complex processes from economy, physics, engineering, chemistry and biology have been simulated, analyzed and optimized by mathematical methods. This work is meant as a contribution to the further extension of the class of processes that can be investigated in this sense with mathematical methods.
One very intuitive way of understanding what this work is about, is to think about a simple switch that can be either on or off. This switch is connected to a complex system and influences it in a certain way. For example a subway with a discrete gear is either accelerated if the switch is on — or it is not accelerated if the switch is off. In optimization one usually has an objective and constraints that one wants to be fulfilled. To stick to the above subway example, reaching a target station in a certain time is one constraint that has to be fulfilled and doing so with minimum energy is a possible objective. The question we want to answer for such systems is: given a mathematical model, constraints and an objective function, *how can we operate the switch in an optimal way?* This question arises whenever time–dependent yes–no decisions have to be made, e.g., for certain types of valves or pumps in engineering, investments in economics, or the choice of discrete operation modes in vehicles.
The optimization of processes that can be described by an underlying system of differential and algebraic equations with so–called control functions is referred to as optimal control. Whereas this expression is based upon common agreement, there are several names for optimal control problems containing binary or integer variables

in the literature. Sometimes it is referred to as *mixed–integer dynamic optimization* or *mixed-logic dynamic optimization* (MIDO or MLDO, see, e.g., Oldenburg *et al.* (2003)), sometimes as *hybrid optimal control* (e.g., Antsaklis & Koutsoukos (1998), Sussmann (1999) or Buss *et al.* (2002)), sometimes as a special case of *mixed–integer nonlinear program* (MINLP) optimization. As controls that take only values at their boundaries are known as *bang–bang controls* in the optimal control community, very often expressions containing bang–bang are used, too (e.g., Maurer & Osmolovskii (2004)). Although there may be good reasons for each of these names, we will use the expressions *mixed–integer optimal control* (MIOC) and *mixed–integer optimal control problem* (MIOCP) in this dissertation. The reason is that the expression *mixed–integer* describes very well the nature of the variables involved and is well–established in the optimization community, while *optimal control* is used for the optimization of control functions and parameters in dynamic systems, whereas the term dynamic optimization might also refer to *parameter estimation* or *optimal experimental design*, e.g., Körkel (2002).
As diverse as the names for the problem class are the ways to approach it. These approaches are typically based upon either one of two different points of view:

- MIOCPs can be seen as members of the mixed–integer problem family, as are mixed–integer linear programs (MILP), mixed–integer quadratic programs (MIQP), mixed–integer nonlinear programs (MINLP) or others. The difference between MIOCPs and other mixed–integer problems then is that the subproblems with fixed or relaxed integer functions resp. variables are optimal control problems instead of linear, quadratic or nonlinear programs.

- A MIOCP can be seen as a special kind of optimal control problem, where restrictions on the control functions are added.

The first point of view makes clear, why the problem class under consideration is so extremely difficult and no general purpose algorithms exist that yield acceptable results for all problem instances. Static, pure integer optimization problems that consist of a convex quadratic function and linear constraints are a subclass of the problem class under consideration here. Such problems and therefore the general class of MINLPs were proven to be $\mathcal{NP}$–hard, Garey & Johnson (1979), Murty (1987), Vavasis (1995). This means from a theoretical point of view, if $\mathcal{NP} \neq \mathcal{P}$, then there are problem instances which are not solvable in polynomial time.
For optimal control problems the *direct methods*, in particular *all–at–once approaches*, Bock & Plitt (1984), Bär (1984), Biegler (1984), have become the methods of choice for almost all practical control problems. These methods are based upon a discretization of the infinite–dimensional control space to a finite–dimensional one. For many problems a high accuracy in the approximation of the control space is necessary which results in a high number of binary control variables and, as mentioned above, to a high overall computing time.
From the second point of view it is well known that for certain systems, in particular linear ones, so–called bang–bang controls are optimal. On the other hand it is not clear what to do if this is not the case and the feasible set of the controls is a priori

restricted to two (or more) discrete values only. Here new methods are needed that determine automatically the optimal switching structure **and** the optimal switching times between the discrete values.

Although the first mixed–integer optimal control problems, namely the optimization of subway trains that are equipped with discrete acceleration stages, were already solved in the early eighties by Bock & Longman (1982) for the city of New York, the so–called *indirect methods* used there do not seem appropriate for generic large–scale optimal control problems with underlying nonlinear differential algebraic equation systems. Most progress after this pioneering work has been achieved in the fields of MINLP and recently also for the solution of optimal control problems with time–independent binary parameters respectively logical decisions.

Several authors treat optimal control problems in chemical engineering where binary parameters often occur as design alternatives, e.g., the location of the feed tray for distillation columns or a mode of operation. This is either done by assuming phase equilibrium, i.e., a steady state of the process, and solving a static optimization problem, e.g., Duran & Grossmann (1986), Grossmann *et al.* (2005), or by solving time–dependent dynamic subproblems, e.g., Schweiger & Floudas (1997) or Oldenburg *et al.* (2003). The algorithmic approaches are extensions of the algorithms developed for MINLPs, possibly in a form that is based on disjunctive (or logic–based) programming, see Turkay & Grossmann (1996) or Oldenburg (2005). A comparison between results from integer programming and from disjunctive programming is given in Grossmann *et al.* (2005).

As most practical optimization problems in engineering are nonconvex, several authors extended methods from static optimization that seek the global optimum, e.g., Esposito & Floudas (2000) and Papamichail & Adjiman (2004). Both present spatial Branch & Bound algorithms for dynamic systems. For spatial Branch & Bound schemes that are built upon an underestimation of the objective function and an overestimation of the feasible set by appropriate convex functions, Floudas *et al.* (2005) claim considerable progress. Barton & Lee (2004) and Lee *et al.* (2004) determine theoretical results on when optimal control problems are convex.

In the theory of hybrid systems one distinguishes between *state dependent* and *controllable switches*. For the first class, the switching between different models is caused by states of the optimization problem, e.g., ground contact of a robot leg or overflow of weirs in a distillation column. For the second class, which is the one we are interested in in this work, the switchings are degrees of freedom. Algorithms for the first class are given in Barton & Lee (2002) and Brandt-Pollmann (2004). For the second class the literature reports mainly on discrete time problems, for which the optimization problem is equivalent to a finite–dimensional one which can be solved by methods from MINLP. Introductions to the theory of hybrid systems are given in Antsaklis & Koutsoukos (1998) and Johansson *et al.* (2004). Recent research includes Zhang *et al.* (2001) and Stursberg *et al.* (2002).

Theoretical results on hybrid systems have been determined, e.g., by Sussmann (1999) and Shaikh (2004). Based on hybrid maximum principles or extensions of

Bellman's equation approaches to treat switched systems have been proposed, e.g., by Shaikh & Caines (2006), Attia *et al.* (2005) or Alamir & Attia (2004), that extend indirect methods or dynamic programming.

Direct methods have been applied only rarely to problems including discrete valued control functions so far. Burgschweiger *et al.* (2004) investigate a water distribution network in Berlin with on/off pumps, using a problem specific, nonlinear, continuous reformulation of the control functions. Terwen *et al.* (2004) treat powertrain control of heavy duty trucks with a rounding heuristics for the optimal gear choice on a fixed control discretization in a model predictive control context. Kaya & Noakes (1996), Kaya & Noakes (2003), Lee *et al.* (1999) and Rehbock & Caccetta (2002) use a switching time approach related to the one described in section 5.2. Buss *et al.* (2002) and Stryk & Glocker (2000) focus on problems in robotics, applying a combination of Branch and Bound and direct collocation.

Goals and results of this thesis

All named approaches to the problem class of mixed–integer optimal control problems and in particular to the treatment of binary control functions have drawbacks at one point or another that will be pointed out in the course of this thesis. The goal of this work is to derive methods that can be applied to a broad class of mixed–integer optimal control problems from different application areas with possibly completely different characteristics as dimension, stability or stiffness of the underlying dynamic system, involving algebraic variables, continuous control functions and parameters and path as well as interior point constraints. The methods are meant to work for systems regardless of the type of solution from a theoretical point of view, i.e., whether an optimal trajectory contains singular or bang–bang arcs resp. constraint–seeking or compromise–seeking arcs. The main contribution of this work consists of a development of algorithms that solve problems fitting into this problem class to optimality without any a priori assumptions on the solution structure.

We propose a novel approach that is based on an all–at–once approach, namely Bock's direct multiple shooting method, Bock & Plitt (1984), that has been applied successfully to a huge variety of challenging problems in industry and research and has advantages compared to other methods of optimal control as will be pointed out in this thesis. We treat the binary control functions by iterating on an adaptive refinement of the control discretization grid, making use of a convex relaxation of the original optimal control problem. We prove that this reformulated problem yields an objective value that can be reached up to any given $\varepsilon > 0$ by binary control functions. Upper bounds are obtained by solution of intermediate problems with fixed dimension on the given control discretization grids. Several methods, among them different rounding heuristics and deterministic approaches as Branch & Bound as well as a penalty term homotopy are presented to solve these intermediate problems and advantages resp. disadvantages are discussed.

The novelties presented in this thesis include

- a rigorous proof that the optimal solution of a convexified and relaxed control problem yields a lower bound that can be approximated arbitrarily close by an integer solution. This is shown for a very general problem class, in which the right hand side may depend nonlinear on differential and algebraic states as on parameters and ordinary control functions.
- an algorithm based upon this proof that, under certain conditions and given a tolerance, yields an integer solution that has an objective function value closer than the prescribed tolerance to the lower bound.
- novel heuristics that avoid an enumeration of the integer variables and exploit the structures of optimal solutions of relaxed problems by a combination of several concepts.
- the solution of several, for different reasons challenging optimization tasks. The control problems that are being solved in this work contain transition stages, coupled and decoupled interior point inequality and equality constraints, path and control constraints, differential and algebraic variables, time–independent parameters, free stage lengths and continuous as well as binary control functions. Furthermore we treat problems that are extremely unstable and contain state–dependent discontinuities.
- the development of a software package that solves efficiently and generically mixed–integer optimal control problems, without the need for analytic a priori work.

The theoretical results obtained allow a *decoupling* of the problems to find the optimal binary parameters and the optimal binary control functions, which will speed up the computing time for problems involving both types of discrete decisions significantly. To show the broad applicability of the developed methods, several case studies and applications are treated within this thesis. Case studies are examples from the literature resp. a new benchmark problem for which the solution structure is known and the behavior of our methods can be analyzed in detail. Three challenging applications from mechanics, cell biology and chemical engineering are presented and solved.
One important result of this work is that mixed–integer optimal control problems can, despite the high complexity of the problem class from a theoretical point of view, in practice often be solved without exponential running times. This is due to the fact that relaxed linear problems often have bang–bang arcs in the optimal solution for which the main task is to determine the switching structure and points. Both can be done efficiently by our proposed approach. For path–constrained or singular arcs the same approach of relaxation and adaptive refinement of the control discretization grid is the basis for a novel rounding heuristics that is tailored to the special ordered set type one structure of the convexified control functions and the intuitively clear fact that solutions in the interior have to be approximated by frequent switching.

Thesis overview

In this work we describe dynamic processes involving switches, but also other, continuous, control functions and time-independent variables in a mathematical way. In chapter 1 we will give a very general definition of multistage mixed–integer optimal control problems with underlying systems of differential–algebraic equations, state, control and interior point constraints. Three examples will be given to familiarize the reader with the problem class. In our novel problem formulation we distinguish between time–independent and time–dependent integer variables and consider cases when fixed time grids are given for the controls. In the latter case, the structure of the optimization problem corresponds more to time–independent variables and methods from integer programming have to be applied that avoid a complete enumeration, but still deliver the globally optimal solution.

In chapter 2 we will investigate optimal control problems without binary variables to create a basis for methods and theory to be presented in later chapters. First we present optimality conditions for optimal control problems, based on Pontryagin's maximum principle, and highlight the solution structure and how it depends on switching functions. In this context we explain the differences between constraint–seeking and compromise–seeking arcs on the one hand and bang–bang and singular arcs on the other hand.
Section 2.2 treats numerical solution methods. We will review indirect and direct methods and discuss respective advantages and disadvantages. It becomes clear why direct multiple shooting is the most promising approach for the optimization of practical and generic mixed–integer optimal control problems. Sequential quadratic programming and the concept of internal numerical differentiation to obtain derivative information are presented.
Section 2.4 gives a brief overview of global optimization of optimal control problems and discusses the question under which assumptions these problems are convex. Again we point out advantages of all–at–once approaches.

Methods for static MINLPs will be reviewed in chapter 3. These methods can and have to be applied for binary parameters and for problems on a fixed control discretization grid, as they will occur as intermediate subproblems in our algorithm.

In chapter 4 we will present a methodology to convexify optimal control problems with respect to the binary control functions. We will state several theorems that clarify the connection between the nonlinear and a convexified problem on the one hand and between binary and relaxed control problems on the other hand. In particular we will prove that, assumed there exists an optimal trajectory to the relaxed convexified problem with objective value Φ^{RL}, there also exists a feasible trajectory for the original, mixed–integer optimal control problem with an objective value $\Phi^{\mathrm{RL}} \leq \Phi^{\mathrm{BN}} \leq \Phi^{\mathrm{RL}} + \varepsilon$ for any given $\varepsilon > 0$.

We prove that binary parameters $\boldsymbol{v}^*$ that are optimal for the control problem with relaxed binary control functions will also be optimal for the integer problem. This allows to *decouple* the determination of the computationally expensive integer problems if parameters as well as control functions are present. This is very beneficial with respect to the overall run time of a solution procedure.

In section 4.3 we formulate an optimal control problem enriched by an additional penalty term in the objective functional and investigate some properties of such a control problem. In section 4.4 we discuss extensions to general multistage mixed–integer optimal control problems and occurring problems in the presence of path and control constraints.

In chapter 5 we will present our novel algorithm to solve mixed–integer optimal control problems. The algorithm is based on an interplay between the direct multiple shooting method, rigorous lower and upper bounds, adaptivity of the control discretization grid and the usage of either heuristics or deterministic methods to solve subproblems on a given grid.

In section 5.1 several rounding strategies are presented, among them specialized ones that take into account the fact that some variables are connected as they discretize the same control function. Furthermore rounding strategies for the multi–dimensional case with special ordered set restrictions on the control functions are given. The switching time optimization approach will be presented in section 5.2. This approach reformulates the optimal control problem as a multistage problem with fixed binary control function values. After an introduction of this approach we discuss its disadvantages and give an illustrative example for the most important one, the introduction of additional nonconvexities. In appendix B.4 we will present an example with multiple local minima for which the direct multiple shooting method converges to the global minimum while direct single shooting converges to a local minimum with bad objective value, although the stage lengths as the only independent degrees of freedom are initialized in both methods with the same values.

Our algorithm is based upon an adaptive refinement of the control discretization grid. In section 5.3 we motivate and present algorithms to obtain an estimation of the objective value corresponding to the optimal trajectory for the infinite–dimensional control problem and to refine a grid such that, under certain assumptions, the optimal trajectory of the relaxed problem can be approximated with a trajectory that is binary admissible. In section 5.4 we present a penalty term homotopy that adds quadratic penalty terms to the control problem on a given control discretization grid. This heuristics can be used to obtain integer values for the control discretization variables. Using a homotopy, we stay inside the convergence radius of the SQP method and we can detect when and where the underlying grid is too coarse.

This is used in the *multiple shooting based mixed–integer optimal control algorithm* (*MS MINTOC*) presented in section 5.5. Making use of the knowledge obtained in chapter 4 that the optimal binary solution of the nonlinear optimal control problem has a corresponding optimal binary solution of a convexified control problem for which we get an attainable lower bound by solving its relaxation, we first determine

this lower bound. We apply some heuristics, namely rounding and applying the switching time optimization, to get upper bounds and compare them with the lower bound. If the result is not satisfactory, we iterate on a refinement of the control grid and an application of the penalty term homotopy, until we end up with a binary admissible trajectory with objective value that is closer than a prescribed tolerance to the attainable optimum. We will prove that under certain theoretic assumptions the *MS MINTOC* algorithm will terminate with such a solution.

In chapter 6 we perform five case studies, i.e., applications of the *MS MINTOC* algorithm to optimal control problems for which the structure of the optimal trajectory $\mathcal{T}^*$ for the infinite–dimensional relaxed optimal control problem is known. In section 6.1 we will see that problems having a bang–bang structure in the relaxed binary control functions can be solved with only very little additional effort compared to the relaxed solution. For such problems the main problem is to find the switching points, the relaxed solution will then coincide with the binary admissible solution. In sections 6.2 and 6.3 we will investigate problems with chattering controls. They differ in a theoretical way, as example 6.2 does not possess an optimal solution although a limit of trajectories exists and example 6.3, the famous problem of Fuller, does have a solution that can be proven to be chattering. For both problems we obtain solutions with a finite number of switches that are closer than a prescribed tolerance ε to the globally optimal objective function value resp. an estimation of it. For such problems the main task is to first determine an adequate control discretization grid and then apply a method to obtain an integer solution on this grid. In section 6.4 we derive an approximation for an example with a singular arc and again get arbitrarily close to the optimal solution. Singular arcs are closely related to arcs that have a chattering solution, as they can be approximated by one. In section 6.5 we extend our study to the case where we need a global solution for a fixed control discretization grid. We demonstrate how a Branch & Bound algorithm can be applied to find such a global solution and point out which role heuristics play in such a scheme.

Applications that involve multistage processes, algebraic equations, interior point constraints, state–dependent discontinuities, path constraints and unstable dynamic systems are presented in chapter 7. In section 7.1 we solve the problem of an energy–optimal operation of a subway train with discrete operation modes in several scenarios, including for the first time point and path constraints. In section 7.2 the phase resetting of calcium concentration oscillations is investigated as an application of the *MS MINTOC* algorithm. This problem is of special mathematical interest as the underlying system is highly unstable in two components and only a very accurate refinement of the control grid yields a feasible solution with an acceptable impact on the system. A batch process with tray– and time–dependent reusage of waste cuts is optimized in section 7.3. By applying our novel method we obtain a new operation strategy for this process that improves the efficiency by more than 13% compared to the best solution given in the literature. The optimal control problem includes transition stages, coupled and decoupled interior point inequalities and equalities, differential and algebraic variables, free, time–independent parameters, free stage lengths and continuous as well as binary control functions. As to our knowledge,

for the first time an optimal control problem of this challenging type is solved to optimality.

In the appendix some basic definitions and theorems that are used throughout the thesis are stated for the convenience of the reader. The fishing benchmark problem is investigated in detail with respect to its optimal solution structure, to a reformulation with a bang–bang structure in its optimal solution and to occurring nonconvexities in the switching time optimization approach in contrast to an optimization of the relaxed original problem. Furthermore all parameter values for the applications and an overview of the notation used throughout the thesis are given.

Chapter 1

Mixed–integer optimal control problems

The scope of this chapter is to give a general formulation of mixed-integer optimal control problems. In section 1.1 the underlying system of differential equations is given, including assumptions. The step towards optimization is taken in section 1.2, the main keywords of this thesis are defined there. In section 1.3 an extension to multistage problems is undertaken. Some illustrating examples are delivered in section 1.4.

1.1 Differential–algebraic equation model

Many complex processes can be modeled by systems of differential–algebraic equations (DAE), e.g., Leineweber *et al.* (2003). In a fully–implicit form these systems read as

$$\mathbf{0} = \boldsymbol{f}^{\mathbf{impl}}(t, \dot{\boldsymbol{y}}(t), \boldsymbol{y}(t), \boldsymbol{u}(t), \boldsymbol{p}) \tag{1.1}$$

Here $t \in [t_0, t_f] \subset \mathbb{R}$ is the time. $\boldsymbol{p} \in \mathbb{R}^{n_p}$ is the *parameter* vector, including all time–independent degrees of freedom. The *control functions*[1] $\boldsymbol{u} : [t_0, t_f] \mapsto \mathbb{R}^{n_u}$ are assumed to be measurable, $\boldsymbol{u} \in \mathcal{U}_m = \{\boldsymbol{u} : [t_0, t_f] \mapsto \mathbb{R}^{n_u}, \boldsymbol{u}(\cdot) \text{ measurable}\}$. At this point $\boldsymbol{p}$ and $\boldsymbol{u}(\cdot)$ can be regarded as fixed. The *state variables* $\boldsymbol{y} : [t_0, t_f] \mapsto \mathbb{R}^{n_y}$ describe the state of the system, their respective time derivatives are given by $\dot{\boldsymbol{y}} = \frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t} : [t_0, t_f] \mapsto \mathbb{R}^{n_y}$. This formulation also covers models obtained by the method of lines from a system of partial differential–algebraic equations (PDAE), see, e.g., Schäfer (2005) or Toumi *et al.* (2006), and of course ordinary differential equations (ODE). Important subclasses of (1.1) are semi–explicit DAE systems of the form

$$\begin{aligned} \boldsymbol{B}(t, \boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{u}(t), \boldsymbol{p})\ \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(t, \boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{u}(t), \boldsymbol{p}), & \text{(1.2a)} \\ \mathbf{0} &= \boldsymbol{g}(t, \boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{u}(t), \boldsymbol{p}) & \text{(1.2b)} \end{aligned}$$

[1] or simply *controls*, in the engineering community sometimes referred to as *inputs* or *manipulated variables*

with a distinction between *differential variables* $\boldsymbol{x} : [t_0, t_f] \mapsto \mathbb{R}^{n_x}$ and *algebraic variables* $\boldsymbol{z} : [t_0, t_f] \mapsto \mathbb{R}^{n_z}$ without time derivative. We write $\boldsymbol{y} = (\boldsymbol{x}, \boldsymbol{z})$. Many chemical engineering problems are of semi–explicit form, see, e.g., Marquardt (1995) or Unger *et al.* (1995), and so are all the applications considered in this thesis. The matrix $\boldsymbol{B} \in \mathbb{R}^{n_x \times n_x}$ is assumed to be regular and could thus be inverted. Although an explicit inversion has to be avoided in practice, we will in the following leave away the matrix $\boldsymbol{B}$ for the sake of notational simplicity and as the applications in this thesis are of explicit form anyway. Furthermore we assume that the derivative of the algebraic right hand side function $\boldsymbol{g} : [t_0, t_f] \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_z} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p} \mapsto \mathbb{R}^{n_z}$ with respect to $\boldsymbol{z}$, namely $\partial \boldsymbol{g} / \partial \boldsymbol{z} \in \mathbb{R}^{n_z \times n_z}$, is regular. This guarantees that system (1.2) is of index 1 as differentiating once with respect to t will transform system (1.2) into an ODE. For the practical treatment of higher–order index systems and the exploitation of occurring invariants we refer to Schulz *et al.* (1998) for details. The right hand side function $\boldsymbol{f} : [t_0, t_f] \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_z} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p} \mapsto \mathbb{R}^{n_x}$ has to be piecewise Lipschitz continuous to ensure existence and uniqueness of a solution $\boldsymbol{y}$. Systems without an explicit dependence on time t are called *autonomous*. Non–autonomous systems can be transformed into autonomous systems by introduction of an additional differential variable and the equation

$$\dot{x}_{n_x+1} = 1 \tag{1.3}$$

with initial value $x_{n_x+1}(t_0) = t_0$. Therefore the argument t will be omitted in the following.

Definition 1.1 (Trajectory)
A trajectory (also referred to as solution) is given by

$$\mathcal{T} \;=\; (\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{p}) = \{\; (\boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{u}(t), \boldsymbol{p}) \;\mid\; t \in [t_0, t_f] \;\}$$

with functions $\boldsymbol{x} : [t_0, t_f] \mapsto \mathbb{R}^{n_x}$, $\boldsymbol{z} : [t_0, t_f] \mapsto \mathbb{R}^{n_z}$, $\boldsymbol{u} : [t_0, t_f] \mapsto \mathbb{R}^{n_u}$ *and a parameter vector* $\boldsymbol{p} \in \mathbb{R}^{n_p}$ *that satisfy (1.2). The components* $\boldsymbol{x}(\cdot)$ *and* $\boldsymbol{z}(\cdot)$ *will be referred to as state trajectories.*

To find trajectories that satisfy (1.2) one has to solve a DAE system. For this solution initial values may be subject to optimization or are given implicitly or explicitly. The latter can be achieved in a general way with the interior point equality constraints

$$\mathbf{0} = \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{x}(t_0), \boldsymbol{z}(t_0), \boldsymbol{x}(t_1), \boldsymbol{z}(t_1), \ldots, \boldsymbol{x}(t_f), \boldsymbol{z}(t_f), \boldsymbol{p}) \tag{1.4}$$

with t_i being interior points in the considered time interval $[t_0, t_f]$. In most cases the initial differential values are given explicitly by

$$\boldsymbol{x}(t_0) = \boldsymbol{x_0} \tag{1.5}$$

and initial algebraic values for given $(\boldsymbol{u}(\cdot), \boldsymbol{p})$ are determined as the solution of

$$\mathbf{0} \;=\; \boldsymbol{g}(\boldsymbol{x}(t_0), \boldsymbol{z}(t_0), \boldsymbol{u}(t_0), \boldsymbol{p}). \tag{1.6}$$

Another important special case of (1.4) are periodic processes with conditions that can be formulated as

$$\boldsymbol{y}(t_0) = \boldsymbol{y}(t_f). \tag{1.7}$$

For given parameters $\boldsymbol{p}$, continuous controls $\boldsymbol{u}(\cdot)$ and initial conditions we obtain an initial value problem (IVP). This problem can be solved with tailored numerical methods.
We would like to stress that the control functions $\boldsymbol{u}(\cdot)$ are assumed to be measurable only. This means that they can be discontinuous, which is also often the case if they are determined as optimal controls. In this thesis we will consider problems where some control functions take values from a disjunct feasible set exclusively. The solution of the DAE is still unique and well–defined if we only have finitely many discontinuities, the integration consists simply of several consecutive initial value problems. The integration has to be stopped and restarted whenever a discontinuity occurs in the controls.
The goal of optimal control is to determine control functions $\boldsymbol{u}(\cdot)$, parameters $\boldsymbol{p}$ and state trajectories $(\boldsymbol{x}(\cdot), \boldsymbol{z}(\cdot))$ that minimize a certain objective functional and for which the trajectory fulfills equations (1.2), (1.4) and additional constraints on the process.

1.2 Problem formulation

Before we come to the main point of this section and have a look at mixed-integer optimal control problems, we define what type of continuous optimal control problems we consider in this thesis and what integer variables are. Furthermore we will classify trajectories into admissible and non admissible ones.

Definition 1.2 (Continuous optimal control problem)
A continuous optimal control problem (OCP) is a constrained optimization problem of the following form:

$$\begin{aligned}
\min_{\boldsymbol{x},\boldsymbol{z},\boldsymbol{u},\boldsymbol{p}} \quad & \Phi[\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{p}] \\
s.t. \quad & \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{u}(t), \boldsymbol{p}), \quad t \in [t_0, t_f], \\
& \mathbf{0} = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{u}(t), \boldsymbol{p}), \quad t \in [t_0, t_f], \\
& \mathbf{0} \leq \boldsymbol{c}(\boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{u}(t), \boldsymbol{p}), \quad t \in [t_0, t_f], \\
& \mathbf{0} \leq \boldsymbol{r}^{\mathbf{ieq}}(\boldsymbol{x}(t_0), \boldsymbol{z}(t_0), \boldsymbol{x}(t_1), \boldsymbol{z}(t_1), \ldots, \boldsymbol{x}(t_f), \boldsymbol{z}(t_f), \boldsymbol{p}), \\
& \mathbf{0} = \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{x}(t_0), \boldsymbol{z}(t_0), \boldsymbol{x}(t_1), \boldsymbol{z}(t_1), \ldots, \boldsymbol{x}(t_f), \boldsymbol{z}(t_f), \boldsymbol{p}).
\end{aligned} \tag{1.8}$$

The variables t, $\boldsymbol{x}(\cdot)$, $\boldsymbol{z}(\cdot)$, $\boldsymbol{u}(\cdot)$ and $\boldsymbol{p}$ are as introduced in section 1.1. The parameter vector $\boldsymbol{p} \in \mathbb{R}^{n_p}$ includes all time–independent degrees of freedom, possibly also the stage length $h := t_f - t_0$ for problems with free end time. The right hand side function $\boldsymbol{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_z} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p} \mapsto \mathbb{R}^{n_x}$ is assumed to be piecewise Lipschitz and the derivative of the algebraic right hand side function $\boldsymbol{g} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_z} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p} \mapsto \mathbb{R}^{n_z}$ with respect to $\boldsymbol{z}$, namely $\partial \boldsymbol{g}/\partial \boldsymbol{z} \in \mathbb{R}^{n_z \times n_z}$, to be regular. The

objective functional $\Phi[\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{p}] := \int_{t_0}^{t_f} L(\boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{u}(t), \boldsymbol{p})dt + E(\boldsymbol{x}(t_f), \boldsymbol{z}(t_f), \boldsymbol{p})$ *is of Bolza-type, containing a Lagrange term* $\int_{t_0}^{t_f} L(\boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{u}(t), \boldsymbol{p})dt$ *and a Mayer term* $E(\boldsymbol{x}(t_f), \boldsymbol{z}(t_f), \boldsymbol{p})$*. Both* E *and* L *are assumed to be twice differentiable, as are the path constraints* $\boldsymbol{c} \in \mathbb{R}^{n_c}$*, the interior point inequality* $\boldsymbol{r}^{\mathbf{ieq}} \in \mathbb{R}^{n_{r^{\mathrm{ieq}}}}$ *and equality constraints* $\boldsymbol{r}^{\mathbf{eq}} \in \mathbb{R}^{n_{r^{\mathrm{eq}}}}$*.*

We will now introduce the concepts of admissibility and optimality of trajectories that are helpful in describing solutions of the OCP.

Definition 1.3 (Admissibility)
A trajectory is said to be admissible if $\boldsymbol{x}(\cdot)$ *is absolutely continuous,* $\boldsymbol{u}(\cdot)$ *is measurable and essentially bounded and the functions* $(\boldsymbol{x}(\cdot), \boldsymbol{z}(\cdot), \boldsymbol{u}(\cdot), \boldsymbol{p})$ *satisfy all constraints of problem (1.8). We say that a control function* $\hat{\boldsymbol{u}}(\cdot)$ *is feasible or admissible, if there exists at least one admissible trajectory* $(\boldsymbol{x}(\cdot), \boldsymbol{z}(\cdot), \hat{\boldsymbol{u}}(\cdot), \boldsymbol{p})$*.*

Definition 1.4 (Optimality)
A trajectory $(\boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ *is said to be globally optimal, if it is admissible and it holds*

$$\Phi[\boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{u}^*, \boldsymbol{p}^*] \leq \Phi[\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{p}] \tag{1.9}$$

for all admissible trajectories $(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{p})$*. A trajectory is said to be locally optimal, if it is admissible and there exists a* $\delta > 0$ *such that (1.9) holds for all admissible trajectories with*

$$||\boldsymbol{u}^*(t) - \boldsymbol{u}(t)|| \leq \delta \ \ \forall\, t \in [t_0, t_f], \quad ||\boldsymbol{p}^* - \boldsymbol{p}|| \leq \delta.$$

A control function is optimal, if it is a component of an optimal trajectory.

The problem class of OCPs, optimality conditions and algorithms will be investigated in detail in chapter 2. At this point we take all this as given and think about possible extensions to include discrete decisions. To do so, we need the following

Definition 1.5 (Integer and binary variables)
Let $\boldsymbol{w} : [t_0, t_f] \mapsto \mathbb{R}^{n_w}$ *be a measurable function and* $\boldsymbol{v} \in \mathbb{R}^{n_v}$ *a vector. A time–dependent or time–independent variable* $w_i(\cdot), 1 \leq i \leq n_w$ *resp.* $v_i, 1 \leq i \leq n_v$ *is called an integer variable, if it is restricted to values in* $\mathbb{Z}$*. If it is restricted to values in* $\{0, 1\}$*, it is called a binary variable or, in the case of* $w_i(\cdot)$*, also a binary control function.*

In the applications considered in this thesis the integer variables are restricted to values in a finite set. Therefore all integer variables $\hat{v} \in \{v_1, \ldots, v_{n_v}\}$ can be represented by $\lceil log_2\, n_v \rceil$ binary variables $\tilde{b}_i$ via a transformation to $\tilde{v} \in \{1, \ldots, n_v\}$ and

$$\tilde{v} = 1 + \sum_{i=1}^{\lceil log_2\, n_v \rceil} \tilde{b}_i\, 2^{i-1} \tag{1.10}$$

For the sake of notational simplicity we will use binary variables from here on exclusively, as they are mathematically (not computationally though!) equivalent. They will be referred to as binary or integer variables. We need some more definitions, before we can investigate mixed-integer optimal control problems. A possible limitation occurs when switching of the binary control functions can only take place at time points from a prefixed given set. This limitation is motivated by machines that can only switch in discrete time steps and by laws or investments that can only be applied resp. made at certain times, e.g., on the first of a month or year. Having this in mind we define

Definition 1.6 (Switching)
Let $\boldsymbol{w}(\cdot)$ be a binary control function. If we have a discontinuity in at least one component of $\boldsymbol{w}(\cdot)$ at time $\tilde{t}$, we say that the control function $\boldsymbol{w}(\cdot)$ switched or that a switching took place. The time point $\tilde{t}$ is called switching time.

Definition 1.7 (Feasible switching set)
The feasible switching set Ψ is the set of time points when a discontinuity in the binary control function vector $\boldsymbol{w}(\cdot)$ may occur. Ψ is either

$$\Psi_\tau = \{\tau_1, \tau_2, \ldots, \tau_{n_\tau}\} \tag{1.11}$$

a finite set of possible switching times or

$$\Psi_{\text{free}} = [t_0, t_f] \tag{1.12}$$

the whole time interval.

If $\Psi = \Psi_{\text{free}}$ there are no restrictions on the switchings and the controls can switch infinitely often, as $\boldsymbol{w}(\cdot)$ is only assumed to be measurable. A limitation on the number of switchings of the binary control functions must be taken into consideration for some problems, though, as an infinitely often occurring switching from one value to the other is not applicable in practice. This inhibition is achieved by a lower limit $\Psi_{\text{MIN}} \geq 0$ on the length of the time interval between two consecutive switching times.

Definition 1.8 (Binary admissibility)
$\boldsymbol{w}(\cdot)$ is called a binary admissible control function on $[t_0, t_f]$, if

$$\boldsymbol{w}(\cdot) \in \Omega(\Psi), \tag{1.13}$$

where $\Omega(\Psi)$ is defined as

$$\Omega(\Psi) := \{\boldsymbol{w} : [t_0, t_f] \mapsto \{0,1\}^{n_w}, \boldsymbol{w}(\cdot) \text{ piecewise constant with jumps only at times } \tau_i \in \Psi \text{ and } \tau_i - \tau_{i-1} \geq \Psi_{\text{MIN}}, \quad i > 1\}.$$

Remark 1.9 *Of course different components of $\boldsymbol{w}$ may have different grids or some might have no switching restrictions while others do. As we do not encounter such problems throughout this thesis and it would complicate the notation in an unnecessary manner, we restrict ourselves to cases where all components of $\boldsymbol{w}$ have the same restrictions.*

Remark 1.10 *The restrictions of the feasible switching set Ψ_τ may also enter the optimal control problem in a different way. In a multistage formulation, see section 1.3, stages could be introduced for each time horizon $[\tau_i, \tau_{i+1}]$ and with parameters $p_{n_p+i} := \tau_{i+1} - \tau_i$ that are fixed or free an equivalent optimal control problem can be formulated without the notation of feasible switching sets. The main difference in this formulation is the question whether the number of possible switches is finite and a priori known or not. As we follow different algorithmic approaches for problems with an a priori given switching structure, we will use the feasible switching sets as defined in definition 1.7. Stages in a multistage formulation of the original problem are then not caused by restrictions on the binary control functions.*

With the concept of binary admissible control functions we can now define mixed-integer optimal control problems:

Definition 1.11 (Mixed-integer optimal control problem)
A mixed-integer optimal control problem (MIOCP) is a constrained optimization problem of the following form:

$$\min_{\boldsymbol{x},\boldsymbol{z},\boldsymbol{w},\boldsymbol{u},\boldsymbol{v},\boldsymbol{p}} \Phi[\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{w}, \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{p}] \tag{1.14a}$$

subject to the DAE system

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{w}(t), \boldsymbol{u}(t), \boldsymbol{v}, \boldsymbol{p}), \quad t \in [t_0, t_f], & \text{(1.14b)} \\ \mathbf{0} &= \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{w}(t), \boldsymbol{u}(t), \boldsymbol{v}, \boldsymbol{p}), \quad t \in [t_0, t_f], & \text{(1.14c)} \end{aligned}$$

control and path constraints

$$\mathbf{0} \leq \boldsymbol{c}(\boldsymbol{x}(t), \boldsymbol{z}(t), \boldsymbol{w}(t), \boldsymbol{u}(t), \boldsymbol{v}, \boldsymbol{p}), \quad t \in [t_0, t_f], \tag{1.14d}$$

interior point inequalities and equalities

$$\begin{aligned} \mathbf{0} &\leq \boldsymbol{r}^{\mathbf{ieq}}(\boldsymbol{x}(t_0), \boldsymbol{z}(t_0), \boldsymbol{x}(t_1), \boldsymbol{z}(t_1), \ldots, \boldsymbol{x}(t_f), \boldsymbol{z}(t_f), \boldsymbol{v}, \boldsymbol{p}), & \text{(1.14e)} \\ \mathbf{0} &= \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{x}(t_0), \boldsymbol{z}(t_0), \boldsymbol{x}(t_1), \boldsymbol{z}(t_1), \ldots, \boldsymbol{x}(t_f), \boldsymbol{z}(t_f), \boldsymbol{v}, \boldsymbol{p}), & \text{(1.14f)} \end{aligned}$$

binary admissibility of $\boldsymbol{w}(\cdot)$

$$\boldsymbol{w}(\cdot) \in \Omega(\Psi), \tag{1.14g}$$

and integer constraints on some of the parameters

$$v_i \in \{0, 1\}, \quad i = 1 \ldots n_v. \tag{1.14h}$$

The designators $\Phi, L, E, \boldsymbol{f}, \boldsymbol{g}, \boldsymbol{c}, \boldsymbol{r}^{\mathbf{ieq}}, \boldsymbol{r}^{\mathbf{eq}}, \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{p}, t$ are as in definition 1.2 besides changes of dimension due to the additional integer variables $\boldsymbol{v} \in \{0, 1\}^{n_v}$ and the binary control functions $\boldsymbol{w} : [t_0, t_f] \mapsto \{0, 1\}^{n_w}$.

Remark 1.12 *While the concepts of admissibility and global optimality can be carried over to mixed–integer optimal control problems, this is not possible for local optima, as it makes no sense to speak about a δ neighborhood in the common sense. One possibility to define a neighborhood for binary variables is to use the Hamming distance, which is defined as the count of bits different in two given patterns of same length. This distance corresponds to the $||\cdot||_1$ norm in the space $\{0,1\}^n$. Nevertheless the concepts of continuous optimization as Karush–Kuhn–Tucker points, see section 2.3.1, are based upon the first definition of local optimality. If we use the expression local optimum or sometimes brief optimum for a trajectory $(\boldsymbol{x}^*, \boldsymbol{z}^*, \boldsymbol{u}^*, \boldsymbol{w}^*, \boldsymbol{p}^*, \boldsymbol{v}^*)$ it is meant as local optimality for fixed $\boldsymbol{w}^*$ and $\boldsymbol{v}^*$ in the usual sense and the binary variables $\boldsymbol{w}^*$ and $\boldsymbol{v}^*$ are not necessarily part of the global optimum.*

In chapters 3 and 5 we will often *relax* the mixed-integer optimal control problem to a continuous one, more precisely we will relax all constraints $w_i(\cdot) \in \{0,1\}$ resp. $v_i \in \{0,1\}$ to the supersets $w_i(\cdot) \in [0,1], v_i \in [0,1]$. The solution of the OCP obtained by this relaxation is then analyzed and used to obtain the solution of the MIOCP.

Definition 1.13 (Relaxed control problem)
The relaxation of a MIOCP is the OCP obtained by replacing the conditions (1.14g) and (1.14h) by

$$\boldsymbol{w}(\cdot) \in \bar{\Omega}(\Psi), \tag{1.15a}$$
$$v_i \in [0,1], \quad i = 1 \dots n_v \tag{1.15b}$$

and rewriting the controls $\boldsymbol{u}(\cdot)$ and $\boldsymbol{w}(\cdot)$ resp. the parameter vectors $\boldsymbol{p}$ and $\boldsymbol{v}$ into new variables $\boldsymbol{u}(\cdot)$ and $\boldsymbol{p}$. The relaxed function space $\bar{\Omega}(\Psi)$ is defined as

$$\bar{\Omega}(\Psi) := \{\boldsymbol{w} : [t_0, t_f] \mapsto [0,1]^{n_w}, \boldsymbol{w}(\cdot) \text{ piecewise constant with}$$
$$\text{jumps only at times } \tau_i \in \Psi \text{ and } \tau_i - \tau_{i-1} \geq \Psi_{\text{MIN}}, \quad i > 1\}.$$

As mentioned by Allgor (1997) such a relaxation may yield theoretical and practical difficulties. Examples for systems that have a unique solution for all integer variables, but none for relaxed values in the interval in between, can be constructed quite easily. Allgor (1997) gives the pathological example

$$\begin{pmatrix} -2v_1 t & 2v_2 t^2 \\ -1 & 2v_1 t \end{pmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} -x_1 \\ -x_2 \end{pmatrix}, \tag{1.16a}$$
$$v_1 + v_2 \leq 1. \tag{1.16b}$$

Brenan *et al.* (1996) showed that this DAE is not uniquely solvable for $\boldsymbol{v}$ fixed to $(0.5, 0.5)$. For every integer realization of $\boldsymbol{v}$ satisfying (1.16b) the solution is unique, though. Still the index of the DAE may change depending on the time t for such integer realizations, yielding a practical difficulty for generic DAE solvers.
We assume for the following that the DAE system is uniquely solvable for all realizations of the relaxation of the MIOCP, even if such a relaxation does not necessarily have to have an interpretable physical meaning as is the case for on–off pumps, valves or gears.

1.3 Multistage problems

In section 1.2 an optimal control problem was formulated with a constant number of variables and continuous differential states over the considered time horizon. For practical problems this is often not sufficient. Transitions as well as changes in the dynamics may occur that are best modeled by multistage optimal control problems, see, e.g., Leineweber (1999) or Diehl *et al.* (2002). To this end we introduce a finite number n_{mos} of intermediate time points into the set of time points t_i that were already used for interior point constraints, see (1.14e-1.14f). We obtain a set of ${n_{\mathrm{ms}}}^2$ ordered time points

$$t_0 \leq t_1 \leq \cdots \leq t_{n_{\mathrm{ms}}} = t_f \tag{1.17}$$

and an ordered subset $\{\tilde{t}_0, \tilde{t}_1, \ldots, \tilde{t}_{n_{\mathrm{mos}}}\}$ with time points, whenever a new model stage occurs and $\tilde{t}_0 = t_0$, $\tilde{t}_{n_{\mathrm{mos}}} = t_{n_{\mathrm{ms}}} = t_f$.

Definition 1.14 (Multistage mixed-integer optimal control problem)
A multistage mixed-integer optimal control problem (MSMIOCP) is a constrained optimization problem of the following form:

$$\min_{\boldsymbol{x_k}, \boldsymbol{z_k}, \boldsymbol{w_k}, \boldsymbol{u_k}, \boldsymbol{v}, \boldsymbol{p}} \sum_{k=0}^{n_{\mathrm{mos}}-1} \Phi_k[\boldsymbol{x_k}, \boldsymbol{z_k}, \boldsymbol{w_k}, \boldsymbol{u_k}, \boldsymbol{v}, \boldsymbol{p}] \tag{1.18a}$$

subject to the DAE model stages (from now on $k = 0 \ldots n_{\mathrm{mos}} - 1$)

$$\dot{\boldsymbol{x_k}}(t) = \boldsymbol{f_k}(\boldsymbol{x_k}(t), \boldsymbol{z_k}(t), \boldsymbol{w_k}(t), \boldsymbol{u_k}(t), \boldsymbol{v}, \boldsymbol{p}), \quad t \in [\tilde{t}_k, \tilde{t}_{k+1}] \tag{1.18b}$$

$$\mathbf{0} = \boldsymbol{g_k}(\boldsymbol{x_k}(t), \boldsymbol{z_k}(t), \boldsymbol{w_k}(t), \boldsymbol{u_k}(t), \boldsymbol{v}, \boldsymbol{p}), \quad t \in [\tilde{t}_k, \tilde{t}_{k+1}] \tag{1.18c}$$

control and path constraints

$$\mathbf{0} \leq \boldsymbol{c_k}(\boldsymbol{x_k}(t), \boldsymbol{z_k}(t), \boldsymbol{w_k}(t), \boldsymbol{u_k}(t), \boldsymbol{v}, \boldsymbol{p}), \quad t \in [\tilde{t}_k, \tilde{t}_{k+1}], \tag{1.18d}$$

interior point inequalities and equalities with k_i denoting the index of a model stage containing t_i, that is $t_i \in [\tilde{t}_{k_i}, \tilde{t}_{k_i+1}]$,

$$\mathbf{0} \leq \boldsymbol{r}^{\mathbf{ieq}}(\boldsymbol{y}_{\boldsymbol{k_0}}(t_0), \boldsymbol{y}_{\boldsymbol{k_1}}(t_1), \ldots, \boldsymbol{y}_{\boldsymbol{k_{n_{\mathrm{ms}}}}}(t_{n_{\mathrm{ms}}}), \boldsymbol{v}, \boldsymbol{p}), \tag{1.18e}$$

$$\mathbf{0} = \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{y}_{\boldsymbol{k_0}}(t_0), \boldsymbol{y}_{\boldsymbol{k_1}}(t_1), \ldots, \boldsymbol{y}_{\boldsymbol{k_{n_{\mathrm{ms}}}}}(t_{n_{\mathrm{ms}}}), \boldsymbol{v}, \boldsymbol{p}), \tag{1.18f}$$

binary admissibility of all $\boldsymbol{w_k}(\cdot)$

$$\boldsymbol{w_k}(\cdot) \in \Omega(\Psi), \tag{1.18g}$$

integer constraints on some of the parameters

$$v_i \in \{0, 1\}, \quad i = 1 \ldots n_v, \tag{1.18h}$$

[2]subscript ms because these points will later on correspond to multiple shooting nodes

and stage transition conditions

$$\boldsymbol{x_{k+1}}(\tilde{t}_{k+1}) \;=\; \boldsymbol{tr_k}(\boldsymbol{x_k}(\tilde{t}_{k+1}), \boldsymbol{z_k}(\tilde{t}_{k+1}), \boldsymbol{v}, \boldsymbol{p}). \tag{1.18i}$$

The objective Bolza functionals $\Phi_k := \int_{\tilde{t}_k}^{\tilde{t}_{k+1}} L_k(\boldsymbol{x_k}(t), \boldsymbol{z_k}(t), \boldsymbol{w_k}(t), \boldsymbol{u_k}(t), \boldsymbol{v}, \boldsymbol{p})dt + E_k(\boldsymbol{x_k}(\tilde{t}_{k+1}), \boldsymbol{z_k}(\tilde{t}_{k+1}), \boldsymbol{v}, \boldsymbol{p})$ *as well as the designators* L_k, E_k, $\boldsymbol{f_k}, \boldsymbol{g_k}$, $\boldsymbol{c_k}$, $\boldsymbol{r^{ieq}}, \boldsymbol{r^{eq}}$, $\boldsymbol{x_k}, \boldsymbol{z_k}$, $\boldsymbol{u_k}, \boldsymbol{p}, t$ *correspond to those in definition 1.11 without index k. The dimensions change towards not necessarily identical values* n_{x_k}, n_{z_k}, n_{u_k}, n_{w_k} *for each stage k.*
If no integer variables are present, that is $n_v = 0$ *and* $n_{w_k} = 0$ *for all k, then problem (1.18) is called a multistage optimal control problem (MSOCP).*

Note that consecutive time points may be identical, as stages with length zero do make sense in some cases. Clearly, regularization strategies must be used in a practical implementation to avoid singularities.
Definitions 1.1, 1.3 and 1.4 are extended in a straightforward way to include the multistage formulation and the integer variables. Then we can define

Definition 1.15 (Feasibility of binary control functions)
A vector $\boldsymbol{w}(\cdot)$ *of binary control functions is said to be feasible, if it is binary admissible (1.18g) and there exists an admissible trajectory for problem (1.18). A binary control function is called infeasible, if it is not feasible.*

1.4 Examples

Definition 1.14 includes several classes of optimal control problems. Compared to standard optimal control problems as, e.g., in Leineweber (1999) or Diehl *et al.* (2002), the additional integer restrictions (1.18g) and (1.18h) turn the problem class into a combinatorial one. Still, depending on Ψ and whether the integer variables are time–dependent or not, completely different problems fit into it that also require different tailored solution methods as will be shown in the following chapters. In this section we will formulate three examples to exemplify definition 1.14.

1.4.1 Time–independent integer variables

An illustrative example for time–independent 0-1 variables in the context of optimal control is given in Oldenburg *et al.* (2003). A distillation process used to separate components of a quaterny mixture is investigated. The energy necessary for the batch process to separate the mixture is to be minimized under purity constraints on the products. The model contains component concentrations and tray holdups as state variables and reflux–ratios of the three batch stages as control functions. The time–independent discrete decisions are whether each batch stage is operated

in regular or in inverse mode[3] and which withdrawal is fed to which batch, yielding altogether 40 different reasonable possibilities.
The algorithmic approach followed by Oldenburg *et al.* (2003) for the model study is based upon *outer approximation*, see section 3.4. Although 40 possibilities could still be enumerated, the number of possibilities grows exponentially with each additional logical decision and becomes prohibitive for extended models very soon.
As in the operation mode case study time–independent binary variables $\boldsymbol{v}$ are often logical decisions. Therefore, as proposed by Oldenburg *et al.* (2003), it often makes sense to formulate these problems as mixed–logic optimization problems and to neglect those parts of the model that are not active. This makes calculations more efficient.

1.4.2 Fixed time–grid

As D'Ancona and Volterra (1926) observed, due to an unexpected decrease in the fishing quota after World War I — everybody expected an increase as fishing was almost completely abandoned in the war years — there is a nontrivial interconnection between the evolution in time of biomasses of fish and fishing. In Sager *et al.* (2006) a simple model was presented as a benchmark problem with discrete decisions on a fixed control time–grid. The biomasses of two fish species — one predator, the other one prey — are the differential states of the model, the binary control is the operation of a fishing fleet. The mode of operation can only be changed at fixed time points, say the first of a month or a year. The optimization goal is to bring the system to a steady state, in this example to $\boldsymbol{x} = (1, 1)^T$. This is achieved by penalizing deviations from it over the whole time horizon with a Lagrange term. The Lotka–Volterra based optimal control problem reads as

$$\min_{\boldsymbol{x},w} \int_{t_0}^{t_f} (x_0(t) - 1)^2 + (x_1(t) - 1)^2 \, dt \tag{1.19a}$$

subject to the ODE

$$\dot{x}_0(t) = x_0(t) - x_0(t)x_1(t) - c_0 x_0(t)\, w(t), \tag{1.19b}$$
$$\dot{x}_1(t) = -x_1(t) + x_0(t)x_1(t) - c_1 x_1(t)\, w(t), \tag{1.19c}$$

initial values

$$\boldsymbol{x}(t_0) = \boldsymbol{x_0}, \tag{1.19d}$$

and the integer constraints

$$w(\cdot) \in \Omega(\Psi) \tag{1.19e}$$

[3] within this distillation column configuration the charge pot is located at the top of the column and the product is withdrawn at the bottom

with fixed $\boldsymbol{c}$ and $\boldsymbol{x_0}$ (see Appendix B) and $n_\tau = 60$ equidistant time points in $\Psi = \Psi_\tau = \{\tau_1, \tau_2, \ldots, \tau_{60}\}$, $[t_0, t_f] = [0, 12]$. Figure 1.1 shows one out of 2^{60} possible control realizations that fulfill (1.19e) and the corresponding states $x_0(\cdot)$ and $x_1(\cdot)$. In this example we do have a similar structure as in example 1.4.1, namely a fixed

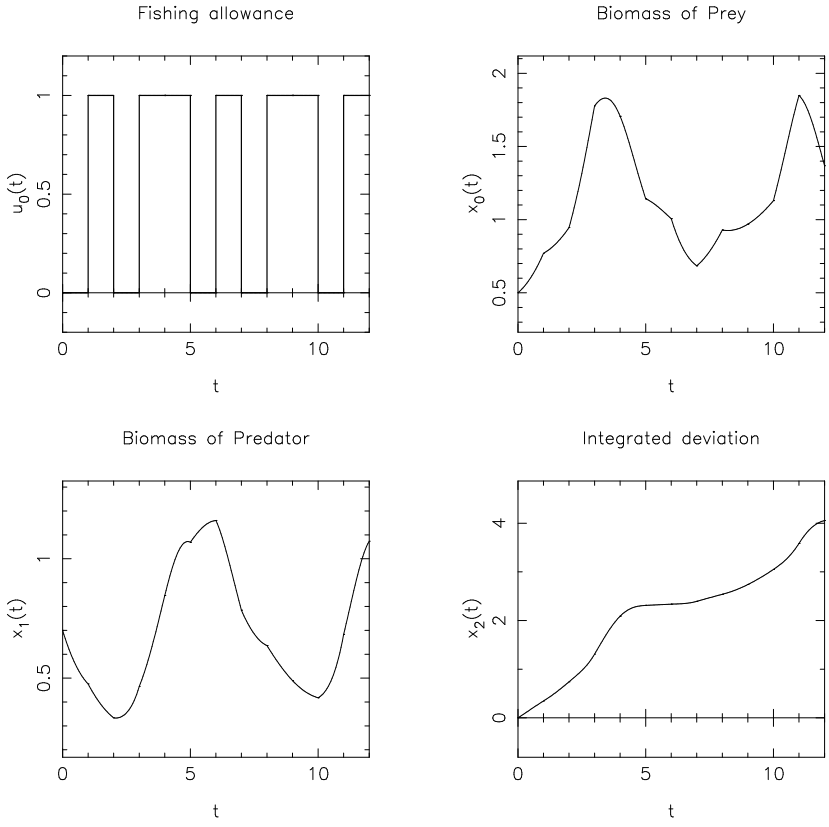

Figure 1.1: Random fishing control $w(\cdot)$, fulfilling (1.19e), the corresponding state trajectories $x_0(\cdot), x_1(\cdot)$ and the Lagrange term $L(\cdot)$. Note the nondifferentiabilities in the state trajectories whenever the control switches.

number of discrete variables. To find feasible and optimal solutions, methods from integer programming as Branch & Bound have to be applied. Such methods will be investigated in chapter 3, the fishing problem will be reviewed in section 6.5.

1.4.3 Free switching times

One of the easiest examples for a binary control function is that of a vehicle that can only be accelerated or decelerated by fixed values $\boldsymbol{w^{\max}}$ resp. $\boldsymbol{w^{\min}}$, see, e.g., Seguchi & Ohtsuka (2003). A model neglecting friction and aiming at bringing the vehicle in minimum time T to a point $\boldsymbol{x_T}$ reads as

$$\min_T \int_0^T 1 \, dt \tag{1.20a}$$

subject to the ODEs

$$\dot{x}_0(t) = x_1(t), \tag{1.20b}$$
$$\dot{x}_1(t) = w(t)\, w^{\max} + (1 - w(t))\, w^{\min}, \tag{1.20c}$$

initial values

$$\boldsymbol{x}(0) = \boldsymbol{x_0}, \tag{1.20d}$$

end point constraints

$$\boldsymbol{x}(T) = \boldsymbol{x_T}, \tag{1.20e}$$

and the integer constraint

$$w(\cdot) \in \Omega(\Psi) \tag{1.20f}$$

with $\Psi = \Psi_{\text{free}}$.
For this simple system the optimal control of bang–bang type can be calculated analytically from the necessary conditions of optimality of the relaxed problem, see chapter 2 or Bryson & Ho (1975) for details. With $\boldsymbol{x_0}$ as the origin $(0, 0)^T$, the terminal point $\boldsymbol{x_T} = (300, 0)^T$ and bounds $w^{\max} = 1$, $w^{\min} = -2$ we obtain acceleration with $w^{\max}$ until $t = 20$ and deceleration with $w^{\min}$ until $t = T = 30$. Figure 1.2 shows covered distance and velocity for this optimal binary control.

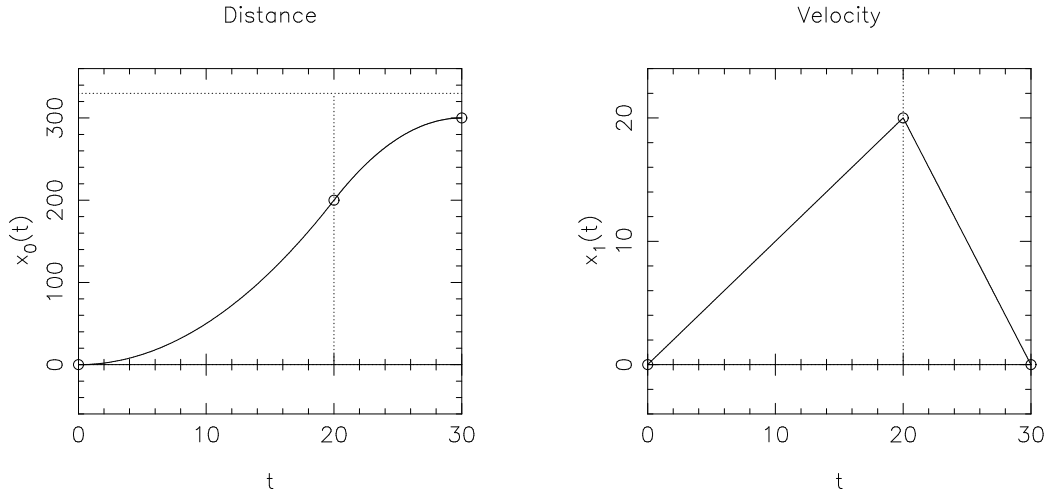

Figure 1.2: Covered distance and velocity of the vehicle for the time optimal control.

Examples 1.4.1 and 1.4.2 have a combinatorial nature — they have a finite though very large number of possible realizations of the integer variables and can thus in theory be enumerated. For binary control functions without a fixed time grid, that is with $\Psi = \Psi_{\text{free}}$, things are different. A control may switch at any given time on

the horizon, the number of possible realizations is thus infinite and other solution strategies have to be applied. In practice this will not be as easy as for example 1.4.3 if one wants to use direct methods (see chapter 2) and apply them to more complex problems.

On the other hand the rocket car example shows that there is structure in optimal solutions that has to be exploited. It clearly makes no sense to optimize this simple control problem by simply discretizing the control function and applying, e.g., a Branch & Bound algorithm to a huge number of binary variables that stamp from this discretization.

1.5 Summary

In this chapter we gave a very general definition of multistage mixed–integer optimal control problems with underlying systems of differential–algebraic equations, state, control and interior point constraints. Three examples are given to familiarize the reader with the problem class.
In our novel problem formulation we distinguish between time–independent and time–dependent integer variables and consider cases when fixed time grids are given for the controls. In the latter case, the structure of the optimization problem corresponds more to time–independent variables and methods from integer programming have to be applied that avoid a complete enumeration, but still deliver the globally optimal solution. Such methods will be investigated in chapter 3, while chapters 4 and 5 aim at developing numerical methods to solve problems involving time–dependent binary control functions without prefixed control grid.

Chapter 2

Optimal control

The problem class of mixed–integer optimal control problems has been defined in chapter 1. This chapter now aims at having a deeper look into optimal control theory and existing algorithms for the non–integer case. First we will review the necessary conditions of optimality, the maximum and the bang–bang principle.
The next step in sections 2.2 and 2.3 will be to investigate the different approaches to treat optimal control problems, namely indirect and direct methods. We investigate methodological extensions in the context of the direct multiple shooting method, therefore we will go more into detail in section 2.3 and also present the framework of sequential quadratic programming and of internal numerical differentiation.
In section 2.4 we will have a closer look at the questions of convexity and local or global optimality.

2.1 Optimality conditions

Most work that has been done in the area of optimality conditions for optimal control problems focuses on a special case of problem (1.18) with a single stage, no algebraic and time–independent variables $\boldsymbol{z}(\cdot)$ resp. $\boldsymbol{p}$ and special interior point conditions. We will follow this line of work and investigate the well known special case first, before we consider more general cases in section 2.1.3. We consider the problem

$$\min_{\boldsymbol{x},\boldsymbol{u}} E(\boldsymbol{x}(t_f)) \tag{2.1a}$$

subject to the ODE system

$$\dot{\boldsymbol{x}}(t) \;=\; \boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{u}(t)), \quad t \in [t_0,t_f], \tag{2.1b}$$

mixed path and control constraints

$$\mathbf{0} \;\leq\; \boldsymbol{c}(\boldsymbol{x}(t),\boldsymbol{u}(t)), \quad t \in [t_0,t_f], \tag{2.1c}$$

initial values

$$\boldsymbol{x}(t_0) \;=\; \boldsymbol{x_0}, \tag{2.1d}$$

and end point equalities

$$\mathbf{0} \;=\; \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{x}(t_f)) \tag{2.1e}$$

on a fixed time horizon $[t_0, t_f]$. Figure 2.1 illustrates this optimal control problem.

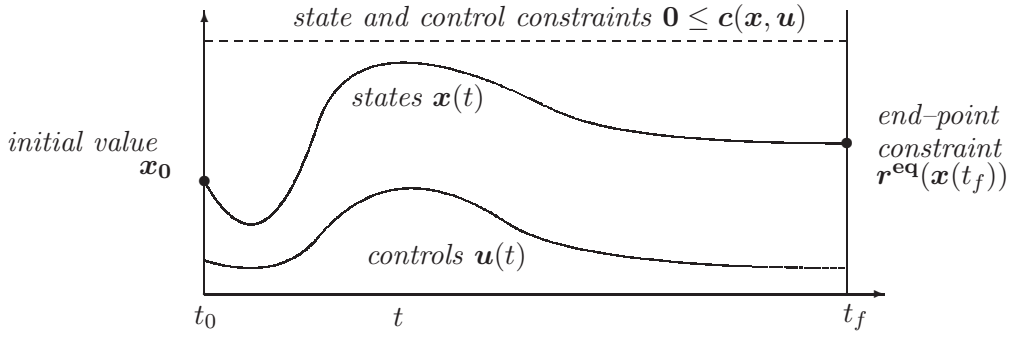

Figure 2.1: Schematic illustration of problem (2.1).

To state the maximum principle we will need the very important concept of the Hamiltonian.

Definition 2.1 (Hamiltonian, Lagrange multipliers)
The Hamiltonian of an optimal control problem (2.1) is given by

$$\mathcal{H}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \;:=\; \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\mu}^T \boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}). \tag{2.2}$$

The end–point Lagrangian function ψ is defined as

$$\psi(\boldsymbol{x}(t_f), \boldsymbol{\nu}) \;:=\; E(\boldsymbol{x}(t_f)) + \boldsymbol{\nu}^T \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{x}(t_f)). \tag{2.3}$$

$\boldsymbol{\lambda} : [t_0, t_f] \to \mathbb{R}^{n_x}$, $\boldsymbol{\mu} : [t_0, t_f] \to \mathbb{R}^{n_c}$ and $\boldsymbol{\nu} \in \mathbb{R}^{n_{r^{\mathrm{eq}}}}$ are called adjoint variables, co–states or Lagrange multipliers.

As in Definition 2.1 we will sometimes leave away the argument (t) in the following for the time dependent functions $\boldsymbol{u}, \boldsymbol{x}, \boldsymbol{\lambda}, \ldots$ for convenience.

2.1.1 Maximum principle

The *maximum principle* in its basic form, also sometimes referred to as *minimum principle*, goes back to the early fifties and the works of Hestenes, Boltyanskii, Gamkrelidze and of course Pontryagin. Precursors of the maximum principle as well as of the Bellman equation can already be found in Carathéodory's book of 1935, compare Pesch & Bulirsch (1994) for details.
The maximum principle states the existence of Lagrange multipliers $\boldsymbol{\lambda}^*(\cdot)$, $\boldsymbol{\mu}^*(\cdot)$ and $\boldsymbol{\nu}^*$ that satisfy adjoint differential equations and transversality conditions. The optimal control $\boldsymbol{u}^*(\cdot)$ is characterized as an implicit function of the states and the adjoint variables — a minimizer $\boldsymbol{u}^*(\cdot)$ of problem (2.1) also minimizes the Hamiltonian subject to additional constraints.

Theorem 2.2 (Maximum principle)
Let problem (2.1) have a feasible optimal solution $\boldsymbol{u}^*(\cdot)$ with a system response $\boldsymbol{x}^*(\cdot)$. Then there exist Lagrange multipliers $\boldsymbol{\lambda}^*(\cdot)$, $\boldsymbol{\mu}^*(\cdot)$ and $\boldsymbol{\nu}^*$ such that for $t \in [t_0, t_f]$ it holds almost everywhere

$$\begin{aligned}
\dot{\boldsymbol{x}}^*(t) &= \mathcal{H}_{\boldsymbol{\lambda}}(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t), \boldsymbol{\lambda}^*(t), \boldsymbol{\mu}^*(t)) = \boldsymbol{f}(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t)), && \text{(2.4a)}\\
\dot{\boldsymbol{\lambda}}^{*T}(t) &= -\mathcal{H}_{\boldsymbol{x}}(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t), \boldsymbol{\lambda}^*(t), \boldsymbol{\mu}^*(t)), && \text{(2.4b)}\\
\boldsymbol{x}^*(t_0) &= \boldsymbol{x_0}, && \text{(2.4c)}\\
\boldsymbol{\lambda}^{*T}(t_f) &= -\boldsymbol{\psi}_{\boldsymbol{x}}(\boldsymbol{x}^*(t_f), \boldsymbol{\nu}^*), && \text{(2.4d)}\\
\mathbf{0} &\leq \boldsymbol{c}(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t)), && \text{(2.4e)}\\
\mathbf{0} &= \boldsymbol{r}^{\mathrm{eq}}(\boldsymbol{x}^*(t_f)), && \text{(2.4f)}\\
\boldsymbol{u}^*(t) &= \arg\min_{\boldsymbol{u}} \mathcal{H}(\boldsymbol{x}^*(t), \boldsymbol{u}(t), \boldsymbol{\lambda}^*(t), \boldsymbol{\mu}^*(t)), && \text{(2.4g)}\\
\mathbf{0} &= \boldsymbol{\mu}^{*T}(t)\ \boldsymbol{c}(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t)), && \text{(2.4h)}\\
\mathbf{0} &\leq \boldsymbol{\mu}^*(t). && \text{(2.4i)}
\end{aligned}$$

Here and in the following $\mathcal{H}_{\boldsymbol{\lambda}} = \frac{\partial \mathcal{H}}{\partial \boldsymbol{\lambda}}$ denotes the partial derivative of $\mathcal{H}$ with respect to $\boldsymbol{\lambda}$ and equivalently for other functions and variables. The complementarity conditions (2.4h) are to be understood componentwise, such that either $\mu_i^* = 0$ or $c_i = 0$ for all $i = 1 \ldots n_c$. For a proof of the maximum principle and further references see, e.g., Bryson & Ho (1975) or Pontryagin *et al.* (1962).
If the objective function in (2.1) is to be *maximized*, then (2.4g) is replaced by a pointwise maximization (this is of course where the name maximum principle comes from). As we chose the minimization formulation for the problems investigated in this thesis, we do the same for the Hamiltonian. In Appendix B.2 an example minimization problem is treated with a maximum formulation of the necessary conditions of optimality.

From the maximum principle first order necessary conditions can be deduced. In particular the pointwise minimization of the Hamiltonian almost everywhere requires that its derivative with respect to the control vector $\boldsymbol{u}$ vanishes almost everywhere

$$\mathbf{0}^T = \mathcal{H}_{\boldsymbol{u}}(\boldsymbol{x}^*, \boldsymbol{u}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*(t)) = \boldsymbol{\lambda}^{*T} \boldsymbol{f}_{\boldsymbol{u}}(\boldsymbol{x}^*, \boldsymbol{u}^*) + \boldsymbol{\mu}^{*T} \boldsymbol{c}_{\boldsymbol{u}}(\boldsymbol{x}^*, \boldsymbol{u}^*) \qquad \text{(2.5a)}$$

and that the *Legendre–Clebsch condition*

$$\mathcal{H}_{\boldsymbol{uu}} = \frac{\partial^2 \mathcal{H}}{\partial \boldsymbol{u}^2} \geq \mathbf{0}, \qquad \text{(2.5b)}$$

holds on unconstrained arcs, i.e., the Hessian of the Hamiltonian is a nonnegative–definite matrix. Note that equalities (2.5) are only necessary and not sufficient for optimality. Second order sufficient conditions have been derived and formulated, e.g., in Maurer & Osmolovskii (2004).

2.1.2 Solution structure

In this work we are particularly interested in finding optimal solutions that take the values 0 or 1 and are the extreme values of the relaxed interval $[0, 1]$. To investigate this further, we define

Definition 2.3 (Admissible region)
The admissible region $\mathcal{R}(t, \boldsymbol{x}(t))$ at time t and state $\boldsymbol{x}(t)$ is the union of all admissible control functions $\boldsymbol{u}(\cdot)$ evaluated at time t for which the path and state constraints (2.1c) hold.

Definition 2.4 (Boundary and interior of admissible region)
The surface $\partial\mathcal{R}(t, \boldsymbol{x}(t))$ limiting the admissible region is called boundary. The union of all controls $\boldsymbol{u}(t)$ with $\boldsymbol{c}(\boldsymbol{x}(t), \boldsymbol{u}(t)) < \boldsymbol{0}$ is called the interior of the admissible region, int $(\mathcal{R})$. A constraint c_i with $c_i < 0$ is called inactive and active if $c_i = 0$, $1 \leq i \leq n_c$.

The first order necessary conditions of optimality can be investigated in more detail to learn about the structure of an optimal solution $\boldsymbol{u}^*(\cdot)$[1]. This is also the path that indirect methods follow, compare section 2.2.1. As stated above, we are particularly interested whether a solution lies in the interior or on the boundary of the admissible region, thus we have to investigate when a constraint c_i, $1 \leq i \leq n_c$, is active.
Let us assume we have an optimal control function $\boldsymbol{u}(\cdot)$ and therewith (2.5a) and also point– and componentwise $0 = \mathcal{H}_{u_i}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{\lambda}(t), \boldsymbol{\mu}(t))$, $i = 1 \dots n_u$. Now two cases can be distinguished for each control $u_i(t)$: either $\boldsymbol{\lambda}^T \boldsymbol{f}_{u_i} \neq 0$ or $\boldsymbol{\lambda}^T \boldsymbol{f}_{u_i} = 0$. Based on this difference $\boldsymbol{\lambda}^T \boldsymbol{f}_u$ will be called switching function.

Definition 2.5 (Switching function)
The n_u-dimensional switching function is given by

$$\boldsymbol{\sigma}^T(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}) \;=\; \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}. \tag{2.6}$$

If the i-th entry of the switching function is not equal to zero at time t, then also $\boldsymbol{\mu}^T \boldsymbol{c}_{u_i}(\boldsymbol{x}, \boldsymbol{u}) \neq 0$. As $\boldsymbol{\mu} \geq \boldsymbol{0}$, at least one entry of $\boldsymbol{\mu}$ must be strictly positive. With the complementarity condition (2.4h) we have that one constraint which explicitly depends upon u_i (as its entry in $\boldsymbol{c}_{u_i}(\boldsymbol{x}, \boldsymbol{u})$ is non–zero) must be active. This constraint can be used to obtain an analytic expression for $u_i(t)$, we say that $u_i(t)$ is *constraint–seeking*, following the terminology and argumentation of Srinivasan *et al.* (2003). Please note that if numerical methods are based upon the necessary conditions of optimality as in section 2.2.1, special attention has to be given to the question whether t is a touch point or a boundary arc[2], that is if the component of

[1] we will leave away the asterisk from now on and assume $\boldsymbol{u}(\cdot)$ to be an optimal solution with corresponding Lagrange multipliers $\boldsymbol{\lambda}(\cdot)$, $\boldsymbol{\mu}(\cdot)$, $\boldsymbol{\nu}$ and system response $\boldsymbol{x}(\cdot)$

[2] Intervals $[t_{\text{entry}}, t_{\text{exit}}]$ with the same behavior are referred to as *arcs*

$\boldsymbol{c}(t)$ is an isolated zero or zero on a whole interval $[t_{\text{entry}}, t_{\text{exit}}]$ with $t_{\text{entry}} < t_{\text{exit}}$. See Bock (1978b), Pesch (1994) or Schulz *et al.* (1998) for details and efficient implementations. Here we will assume that this behavior is constant in a small neighborhood of t.

If the i-th entry of the switching function is equal to zero at time t, we have to further distinguish between two cases. If $\boldsymbol{\lambda}^T \boldsymbol{f}_{u_i}$ does explicitly depend upon u_i, the control can be determined from the equation $\boldsymbol{\lambda}^T \boldsymbol{f}_{u_i} = 0$. If this is not the case, we differentiate the Hamiltonian with respect to time. Since $\mathcal{H}_{u_i} = 0$ for all t, all time derivatives will vanish, too. Differentiating (2.5a) delivers

$$\begin{aligned} \frac{\mathrm{d}\mathcal{H}_{u_i}}{\mathrm{d}t} &= \dot{\boldsymbol{\lambda}}^T \boldsymbol{f}_{u_i} + \boldsymbol{\lambda}^T \left(\frac{\partial \boldsymbol{f}_{u_i}}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} + \frac{\partial \boldsymbol{f}_{u_i}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}} \right) + \dot{\boldsymbol{\mu}}^T \boldsymbol{c}_{u_i} + \boldsymbol{\mu}^T \left(\frac{\mathrm{d}\boldsymbol{c}_{u_i}}{\mathrm{d}t} \right) \\ &= \dot{\boldsymbol{\lambda}}^T \boldsymbol{f}_{u_i} + \boldsymbol{\lambda}^T \left(\frac{\partial \boldsymbol{f}_{u_i}}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} + \frac{\partial \boldsymbol{f}_{u_i}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}} \right) \end{aligned}$$

as $\boldsymbol{\mu}^T \boldsymbol{c}_{u_i} = 0$ over an interval and the complementarity conditions (2.4h) hold. Using (2.4a) and (2.4b) to replace $\dot{\boldsymbol{x}}$ and $\dot{\boldsymbol{\lambda}}$ one obtains

$$\frac{\mathrm{d}\mathcal{H}_{u_i}}{\mathrm{d}t} = \boldsymbol{\lambda}^T \left(\frac{\partial \boldsymbol{f}_{u_i}}{\partial \boldsymbol{x}} \boldsymbol{f} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} \boldsymbol{f}_{u_i} + \frac{\partial \boldsymbol{f}_{u_i}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}} \right) - \boldsymbol{\mu}^T \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}} \boldsymbol{f}_{u_i} \tag{2.7}$$

$$= \boldsymbol{\lambda}^T \Delta^1 \boldsymbol{f}_{u_i} - \boldsymbol{\mu}^T \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}} \boldsymbol{f}_{u_i} \tag{2.8}$$

where the operator Δ^1 is given by[3]

$$\Delta^1 \boldsymbol{F} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{x}} \boldsymbol{f} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} \boldsymbol{F} + \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}} \tag{2.9}$$

and represents the time differentiation of a vector function $\boldsymbol{F}$ along the trajectories of the dynamic system (2.4a, 2.4c). This operator is studied in the systems literature using tools of Lie algebra. Differentiating (2.8) $j-1$ more times in a similar manner leads to an expression consisting of two parts, a system dependent and a constraint dependent one:

$$\frac{\mathrm{d}^j \mathcal{H}_{u_i}}{\mathrm{d}t^j} = \boldsymbol{\lambda}^T \Delta^j \boldsymbol{f}_{u_i} - \boldsymbol{\mu}^T \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}} \Delta^{j-1} \boldsymbol{f}_{u_i}. \tag{2.10}$$

Here $\Delta^j := \Delta^1(\Delta^{j-1})$ with $\Delta^0 := id$ is recursively defined. The time differentiation (2.10) is repeated for increasing j until one of two cases occurs — either we have $\boldsymbol{\lambda}^T \Delta^j \boldsymbol{f}_{u_i} \neq 0$ and with the same argumentation as above it follows that the control is constraint–seeking or we have $\boldsymbol{\lambda}^T \Delta^j \boldsymbol{f}_{u_i} = 0$ and u_i appears explicitly in the expression, leading to a *compromise–seeking* control.

Figure 2.2 illustrates the logic behind the two different control types, constraint–seeking and compromise– (or sensitivity–) seeking controls.

[3] if $\boldsymbol{F}$ is not a function of the time derivatives of $\boldsymbol{u}$, else additional terms are needed

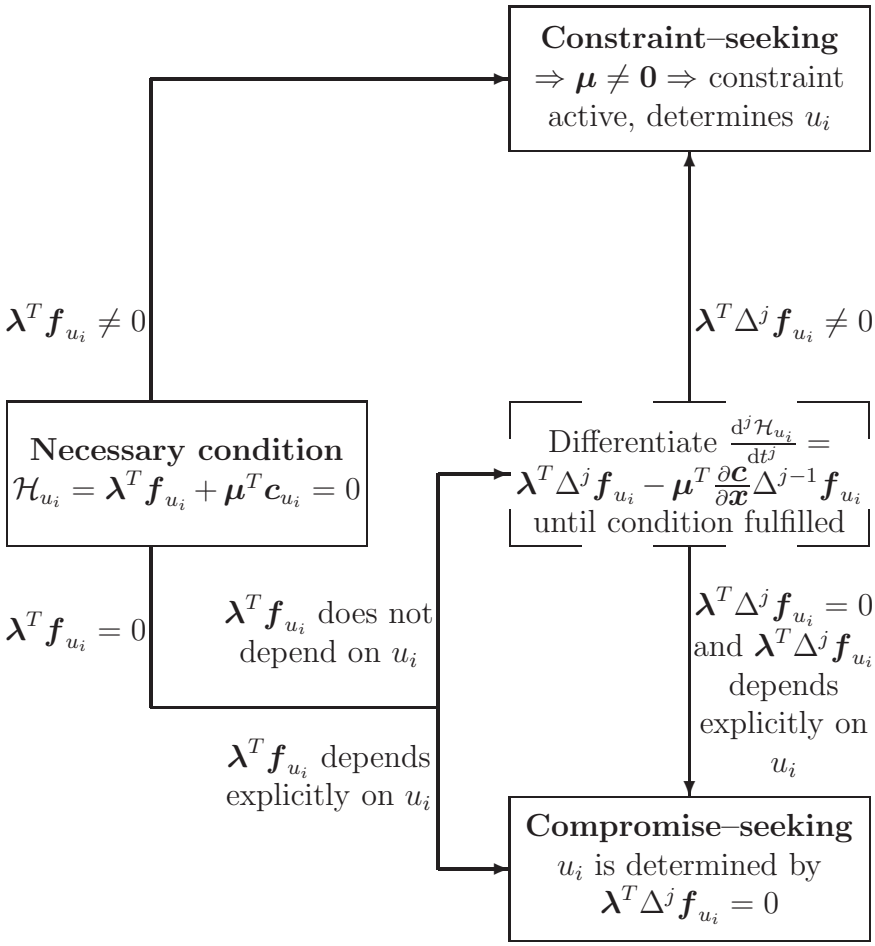

Figure 2.2: Constraint–seeking and compromise–seeking controls

It is worthwhile to look at the special case of control–affine systems. These systems have a linear entry of the controls $\boldsymbol{u}(\cdot)$ in the appearing functions, namely

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{F^0}(\boldsymbol{x}) + \boldsymbol{F^U}(\boldsymbol{x})\, \boldsymbol{u}, \tag{2.11a}$$

$$\boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{C^0}(\boldsymbol{x}) + \boldsymbol{C^U}(\boldsymbol{x})\, \boldsymbol{u}. \tag{2.11b}$$

For control–affine systems there is another important case differentiation between *singular* and *nonsingular* arcs. Those are defined by

Definition 2.6 (Singularity)
A control function $\boldsymbol{u}$ is said to be singular of rank r over a non–zero time interval $[t_{\text{entry}}, t_{\text{exit}}]$ with $t_{\text{entry}} < t_{\text{exit}}$, if r components of $\boldsymbol{u}$ cannot be determined from the condition $\mathcal{H}_{\boldsymbol{u}} = \mathbf{0}^T$ over this interval. If $r = 0$, $\boldsymbol{u}$ is called nonsingular. An input u_i is said to have a degree of singularity r if u_i appears for the first time in the $(r+1)$-th time derivative of $\mathcal{H}_{\boldsymbol{u}}$.

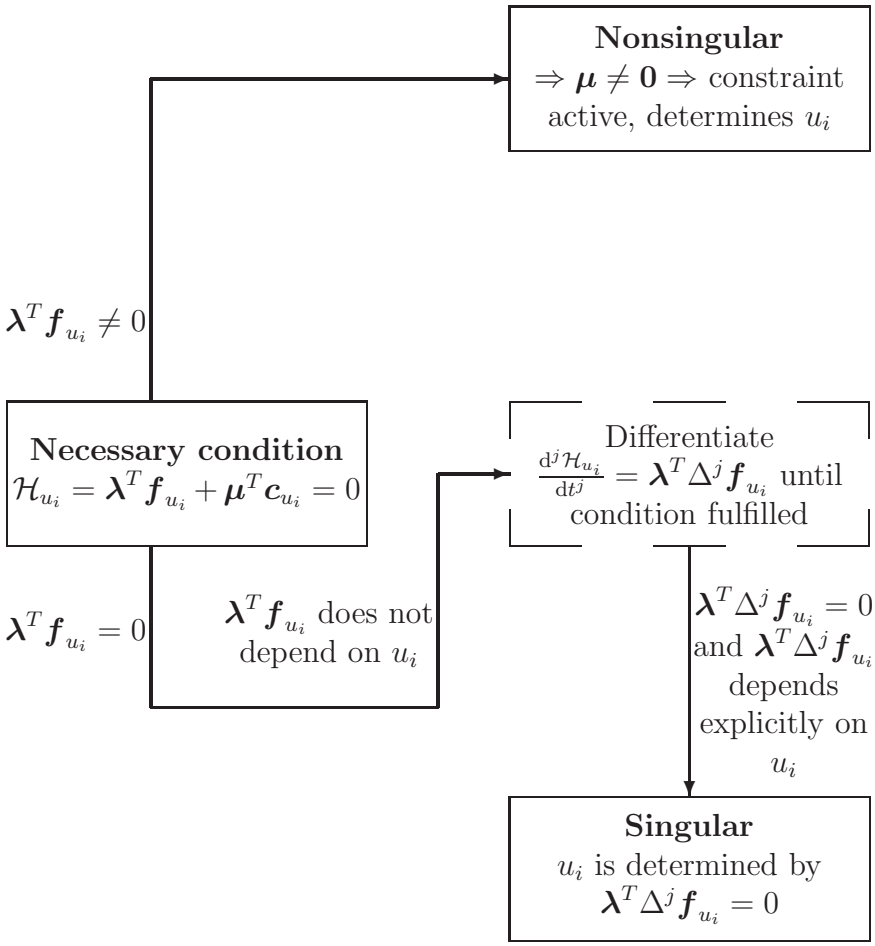

Figure 2.3: Nonsingular and singular controls in the case of a control–affine system and state–independent constraints. Compared to figure 2.2 two things changed: First $\mathcal{H}_{\boldsymbol{u}\boldsymbol{u}} = \mathbf{0}$, thus $\boldsymbol{\lambda}^T \boldsymbol{f}_{u_i}$ never depends on u_i and second $\frac{\partial \boldsymbol{c}}{\partial \boldsymbol{x}} = \mathbf{0}$, thus $\boldsymbol{\lambda}^T \Delta^j \boldsymbol{f}_{u_i} \stackrel{!}{=} 0$.

Definition 2.7 (Bang–bang controls)
If a nonsingular control is determined by a bound constraint $u_i = u_i^{\max}$ or $u_i = u_i^{\min}$, it is said to be of bang–bang type and called a bang–bang control.

Remark 2.8 *It was shown by Robbins (1967) that in the singular case the control functions u_i first appear in an even derivative of $\mathcal{H}_{\boldsymbol{u}}$ as a result of the second–order necessary conditions of optimality.*

Consider the case where $\boldsymbol{c}$ consists of bounds on the inputs only, say

$$\boldsymbol{c}(\boldsymbol{u}) \;=\; \begin{pmatrix} \boldsymbol{u} - \boldsymbol{u}^{\max} \\ \boldsymbol{u}^{\min} - \boldsymbol{u} \end{pmatrix}. \tag{2.12}$$

For a control–affine system the switching function of control u_i

$$\sigma_i(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}) \;=\; \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{f}}{\partial u_i} = \boldsymbol{\lambda}^T \; \boldsymbol{F}_{.i}^{\boldsymbol{u}}(\boldsymbol{x}) \tag{2.13}$$

plays an important role[4]. If it is strictly positive, the pointwise minimizer of the Hamiltonian function u_i must be as small as possible, thus at its lower bound. If it is strictly negative it holds $u_i = u_i^{\max}$, in both cases we do have a bang–bang arc. If $\sigma_i = 0$, we cannot deduce u_i from this expression as $\mathcal{H}_{\boldsymbol{u}\boldsymbol{u}} = \mathbf{0}$. In this singular case $\mathcal{H}_{u_i}$ has to be differentiated with respect to time until the degree of singularity of u_i is reached, assumed it is finite. The resulting singular control will lie in the interior of the admissible region (besides rare exceptional cases when the singular control takes exactly the value at the boundary).
In the general nonlinear case this differentiation between singular and nonsingular controls is not appropriate if emphasis is given to the question whether a control lies in the interior of the admissible region (compromise–seeking) or at its boundary (constraint–seeking). For control–affine systems, when the constraints are state–independent (e.g., they consist of bounds on the control functions only), a nonsingular arc implies that the control is on the boundary of the admissible region whereas a singular arc implies that it is in the interior of it as stated above. But in general singular controls can also be constraint–seeking in the case of state–dependent constraints and, for nonlinear systems, nonsingular controls can be compromise–seeking and lie in the interior of the admissible region. For control–affine systems the latter is not possible as the Hamiltonian is linear in $\boldsymbol{u}$, too. Therefore the switching function $\boldsymbol{\lambda}^T \boldsymbol{f}_{u_i}$ does not depend explicitly on u_i and we can neglect the bottommost case in Figure 2.2.
Figure 2.3 shows the the logic behind the two different control types, singular and nonsingular controls for the case of control–affine systems with state–independent constraints.

2.1.3 Extensions

Problem (2.1) is only a special case of the multistage optimal control problem class defined in Definition 1.14 (with $n_v = n_w = 0$). The maximum principle has to be extended to treat the more general problem class. In particular the following extensions are necessary to treat (1.18):

- *Lagrange term*

  If the objective functional does contain a Lagrange term $L(\boldsymbol{x}, \boldsymbol{u})$ (that could of course be included by an additional differential state and equation, too), the *Hamiltonian* is modified in the following way:

  $$\mathcal{H}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \;:=\; \lambda_L L(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\mu}^T \boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}) \tag{2.14}$$

  and theorem 2.2 holds as before. The multiplier $\lambda_L \in \mathbb{R}$ is typically scaled to the value 1 and neglected.

- *Parameters*

[4] $\boldsymbol{F}^{\boldsymbol{u}}_{\cdot i}(\boldsymbol{x})$ is the i–th column of matrix $\boldsymbol{F}^{\boldsymbol{u}}(\boldsymbol{x})$

Time–independent parameters $\boldsymbol{p}$ can be formally included by introducing additional differential state variables with $\dot{\boldsymbol{p}} = \boldsymbol{0}$, therefore they need not be considered explicitly.

- *Free end time*

  If the end time T is free for optimization, we obtain a second transversality condition besides (2.4d). If we add

  $$0 \;=\; \left(\mathcal{H} + \frac{\partial E}{\partial t} + \boldsymbol{\mu}^{*T}\,\frac{\partial \boldsymbol{r}^{\mathbf{eq}}}{\partial t}\right)_{t=T}$$

  to (2.4), theorem 2.2 holds as before.

- *Multistage problems*

  In optimal control often arcs have to be concatenated, for example singular and nonsingular or state–constrained and –unconstrained arcs. For them, as well as for multiple stages, the principle of optimality holds: the optimal trajectory $(\boldsymbol{x}(\cdot), \boldsymbol{u}(\cdot))$ over the whole time horizon is also optimal for every subinterval, that is in our case, arc resp. stage. Therefore the necessary conditions for all substages can be concatenated directly to obtain the necessary conditions of optimality for the multistage problem, if matching conditions are considered.

- *Algebraic variables*

  For the optimal control problems considered in this thesis we made the index 1 assumption that $\partial \boldsymbol{g}/\partial \boldsymbol{z} \in \mathbb{R}^{n_z \times n_z}$ is regular. By this assumption the algebraic variables are determined uniquely and can be neglected when necessary conditions of optimality are investigated.

- *More general boundary constraints*

  If the boundary constraints are given in a more general form than (2.4c, 2.4f), as might, e.g., be the case for periodic processes, as

  $$\boldsymbol{0} \;=\; \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f)), \tag{2.15}$$

  then the end–point Lagrangian function ψ has to be redefined as

  $$\psi(\boldsymbol{x}(t_f), \boldsymbol{\nu}) \;:=\; E(\boldsymbol{x}(t_f)) + \boldsymbol{\nu}^T \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_f)). \tag{2.16}$$

- *Interior point constraints*

  If we do have interior point equalities or inequalities, that is,

  $$\boldsymbol{0} \;=\; \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \ldots, \boldsymbol{x}(t_f)), \tag{2.17a}$$
  $$\boldsymbol{0} \;\leq\; \boldsymbol{r}^{\mathbf{ieq}}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \ldots, \boldsymbol{x}(t_f)), \tag{2.17b}$$

  then additional jump conditions on the co–states occur and the transversality conditions have to be modified. See Bryson & Ho (1975) for details.

Proofs of the maximum principle for these special cases can be found, e.g., in Bryson & Ho (1975).

2.1.4 Bang–bang principle

So far we considered implications of the necessary conditions of optimality and saw that an optimal solution may be on the boundary or in the interior of the admissible region. As we are interested in the case where binary control functions are binary feasible only if they are on the boundary of the unit cube $[0,1]^{n_w}$, we will next examine if and when controls in the interior may be replaced by controls on the boundary. We will stick here to a standard formulation of the bang–bang principle that can usually be found in the literature. In chapter 4 we will reinvestigate this principle in more detail and apply it to more general optimal control problems.
Following the line of investigation of the textbooks by Hermes & Lasalle (1969) and Macki & Strauss (1995), we consider the following linear control problem

$$\dot{\boldsymbol{x}}(t) \;=\; \boldsymbol{A^1}(t)\,\boldsymbol{x} + \boldsymbol{A^2}(t)\,\boldsymbol{u}, \quad t \in [t_0, t_f], \tag{2.18a}$$

with initial values

$$\boldsymbol{x}(t_0) \;=\; \boldsymbol{x_0}, \tag{2.18b}$$

and measurable control functions $\boldsymbol{u}(\cdot) \in \mathcal{U}_m$ that are bounded by

$$\boldsymbol{u}^{\min} \le \boldsymbol{u}(t) \le \boldsymbol{u}^{\max}, \quad t \in [t_0, t_f]. \tag{2.18c}$$

$\boldsymbol{A^1}$ and $\boldsymbol{A^2}$ are time–dependent, continuous matrices. We define

Definition 2.9 (Controllable set)
The controllable set at time T is the set of all points $\boldsymbol{x_0}$ that can be steered back to the origin in time T,

$$\mathbb{C}(T) \;=\; \{\boldsymbol{x_0} : \exists \boldsymbol{u}(\cdot) \in \mathcal{U}_m \text{ such that } \boldsymbol{x}(T; \boldsymbol{x_0}, \boldsymbol{u}) = \boldsymbol{0}\}.$$

The controllable set is the union of all sets $\mathbb{C}(T)$ for positive T,

$$\mathbb{C} \;=\; \cup_{T>0}\, \mathbb{C}(T).$$

Equivalently, we define $\mathbb{C}_{BB}(T)$ and $\mathbb{C}_{BB}$ for $\boldsymbol{u}(\cdot) \in \mathcal{U}_{BB}$, where

$$\mathcal{U}_{BB} \;=\; \{\boldsymbol{u} \in \mathcal{U}_m, u_i(t) = u_i^{\max} \text{ or } u_i(t) = u_i^{\min}\ \forall\ t \in [t_0, t_f], i = 1 \ldots n_u\},$$

as the controllable sets of bang–bang functions.

With the definitions made we can state the following theorem.

Theorem 2.10 (Bang–bang principle)
For the system (2.18) we have

$$\mathbb{C}(T) = \mathbb{C}_{BB}(T) \tag{2.19}$$

for all $T \ge 0$. This set is compact, convex and depends continuously on T.

Proofs can be found in Hermes & Lasalle (1969) or Macki & Strauss (1995). One very important conclusion of theorem 2.10 is that, if there is a solution at all to a linear time–optimal control problem, there is also a bang–bang solution that is optimal. For general nonlinear problems this is not true any more. Consider the one–dimensional example

$$\dot{x}(t) = u(t) + u^2(t) \tag{2.20}$$

with $u^{\min} = -1$ and $u^{\max} = 1$. Obviously $\dot{x}(t) \geq 0$ for $u(t) \in \{-1, 1\}$, while positive x_0 can be steered to the origin by controls in the interior, e.g, by $u(t) = -0.5$, ensuring $\dot{x}(t) < 0$.

2.2 Solution methods

There are several methods in the literature to solve optimal control problems of the kind (1.8). The first differentiation considers the optimization space. *Indirect methods* do optimize in an infinite dimensional function space, while *direct methods* do transform the problem to a finite–dimensional space first before the optimization takes place. *Dynamic programming* is based on the Hamilton–Jacobi–Bellman partial differential equations. Direct methods can be further distinguished based on the type of discretization that is used to transform the infinite–dimensional optimal control problem to a finite–dimensional nonlinear program. While *direct single shooting* discretizes the controls only and integrates the differential equations with a DAE solver to get corresponding states, *collocation* also discretizes the states and ensures continuity of the solution by additional constraints. A third direct method that combines the advantages of both approaches is *direct multiple shooting*. Figure 2.4 gives an overview of the mentioned methods. We will base all methods developed in this thesis on the direct multiple shooting method, therefore we will go more into detail in section 2.3 and comment on *sequential quadratic programming*, one way to solve the occurring nonlinear program, and on efficient methods to obtain derivatives in this section. All other methods are shortly described in the sequel. For a more detailed overview and comparison between indirect and direct methods, sequential and simultaneous approaches (in particular single shooting, multiple shooting and collocation) we refer to Binder *et al.* (2001).

2.2.1 Indirect methods

The classical approach to solving optimal control problems is based on Pontryagin's maximum principle, see theorem 2.2 in section 2.1.1. The necessary conditions of optimality are used to transform the optimization problem to a multipoint boundary value problem that is solved, e.g., by multiple shooting, see Osborne (1969), Bulirsch (1971) or Bock (1978b).
An optimal solution typically consists of several arcs. On each arc we do have constraint–seeking or compromise–seeking controls, as investigated in section 2.1.2. These controls are determined by the necessary conditions of optimality and have

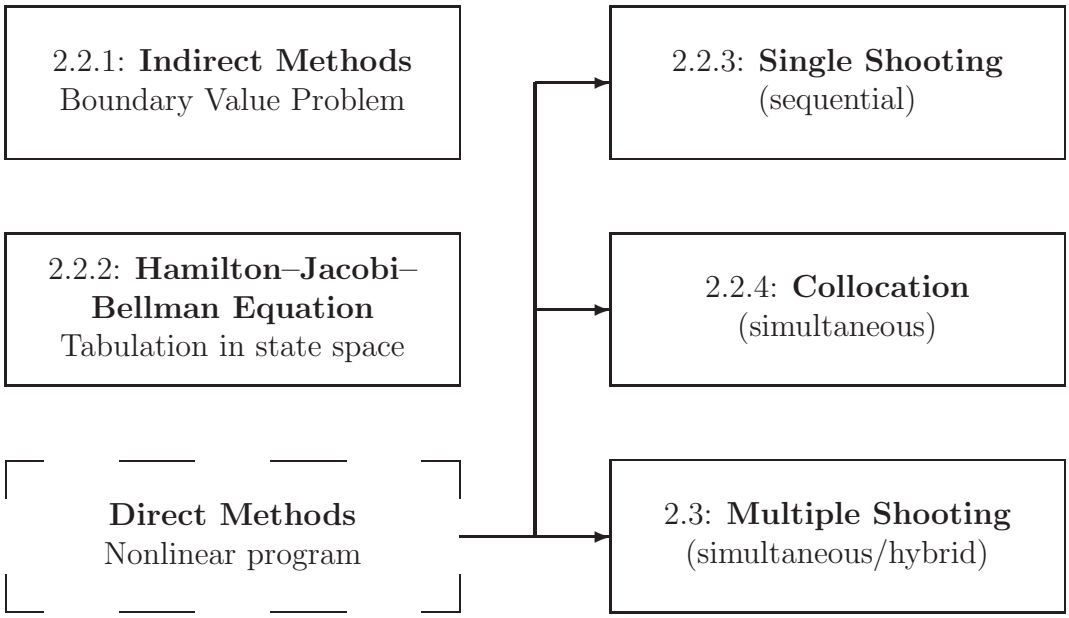

Figure 2.4: Optimal control solution methods described in this chapter

to be calculated analytically. Introduction of switching times τ_i as additional variables and switching conditions $\boldsymbol{S}(\boldsymbol{x}(\tau_i), \boldsymbol{\lambda}(\tau_i)) = \mathbf{0}$ for the transition from one arc to another leads to a multipoint boundary value problem with unknown parameters τ_i that can be determined by appropriate numerical methods. The derivation of correct and numerically stable switching conditions $\boldsymbol{S}(\boldsymbol{x}(\tau_i), \boldsymbol{\lambda}(\tau_i)) = \mathbf{0}$ is by no means trivial. Special cases have to be distinguished, compare section 2.1.2. Especially general path and control constraints $\boldsymbol{c}(\cdot)$ and interior point constraints $\boldsymbol{r}^{\mathbf{ieq}}(\cdot)$ lead to an a priori unknown switching structure. Constraints may get active or inactive, jumps in the adjoint variables may occur and special care has to be taken for active constraints whether touch points or boundary arcs are involved. See Bock (1978a), Hartl *et al.* (1995) or Pesch (1994) for details.

The disadvantages of indirect methods are quite obvious. The formulation of the boundary value problem in a numerically stable way requires a lot of know how and work. Furthermore already small changes in the value of a parameter or in the problem definition, e.g. an additional constraint, may change the switching structure completely.

The switching times have to stay in the multiple shooting intervals, otherwise convergence of Newton's method is not ensured anymore. Only if the switching structure is guessed correctly in advance and does not change during the iterations of the multiple shooting algorithm, it is possible to transform the problem onto fixed switching times.

Start values for all variables have to be delivered, which is often difficult especially for the adjoints. This is crucial, because one has to start inside the convergence region of Newton's method. In case of path constraints usually homotopies have to be applied to obtain such start values.

The main advantage of indirect methods is the high accuracy of the obtained solution, as the infinite–dimensional problem has been solved. In particular, no approximations of the controls have been undertaken, in contrast to direct methods, see sections 2.2.3, 2.2.4 and 2.3. Also, the resulting boundary value problem has a dimension of $2n_x$ only compared to dynamic programming, see section 2.2.2. As all degrees of freedom in the controls vanish, this approach seems appropriate for problems with a high number of control functions when compared to direct methods. If the number of states is large compared to the number of controls, direct methods are usually more efficient.
In general, an interactive iterative process involving the solution of several multipoint boundary value problems is necessary to get a solution using indirect methods. Both, insight into the problem and specific numerical knowledge are typically required for this task. Consequently, nowadays indirect methods are most often applied when high accuracy of the solution is crucial and enough time for obtaining the solution is available, e.g., in the aerospace domain, Pesch (1994) or Caillau *et al.* (2002). Typically, initial guesses for the variables are generated by applying direct methods, Bulirsch *et al.* (1991).
In Appendix B.3 the solution of example 1.4.2 is given in detail to illustrate the indirect approach.

2.2.2 Dynamic Programming and the HJB equation

Dynamic Programming is a discrete–time technique based on the principle of optimality, that is, any subarc of an optimal trajectory is also optimal. If we do have an optimal solution $(\boldsymbol{x}(\cdot), \boldsymbol{u}(\cdot))$ of problem (2.1) and an intermediate time point $\bar{t} \in [t_0, t_f]$, then the subarc $(\boldsymbol{x}(\cdot), \boldsymbol{u}(\cdot))$ on $[\bar{t}, t_f]$ is the optimal solution for the initial value $\bar{\boldsymbol{x}} = \boldsymbol{x}(\bar{t})$.

Definition 2.11 (Optimal–cost–to–go function)
The optimal–cost–to–go function on $[\bar{t}, t_f]$ is given by

$$J(\bar{\boldsymbol{x}}, \bar{t}) := \min_{\boldsymbol{x},\boldsymbol{u}} \int_{\bar{t}}^{t_f} L(\boldsymbol{x}, \boldsymbol{u})dt \; + \; E(\boldsymbol{x}(t_f)) \tag{2.21}$$

subject to $\boldsymbol{x}(\bar{t}) = \bar{\boldsymbol{x}}$ and equations (2.1b), (2.1c) and (2.1e).

With a time grid $t_0 < t_1 < \cdots < t_{n_{\mathrm{DP}}} = t_f$ the optimal-cost-to-go function at time t_k can be written as

$$J(\bar{\boldsymbol{x}}_k, t_k) \;\; = \;\; \min_{\boldsymbol{x},\boldsymbol{u}} \int_{t_k}^{t_{k+1}} L(\boldsymbol{x}, \boldsymbol{u})dt \; + \; J(\boldsymbol{x}(t_{k+1}), t_{k+1}) \tag{2.22}$$

subject to $\boldsymbol{x}(t_k) = \bar{\boldsymbol{x}}_k$ and equations (2.1b), (2.1c) and (2.1e). Now, starting from $J(\boldsymbol{x}, t_f) = E(\boldsymbol{x})$, the optimal-cost-to-go functions can be computed recursively backwards, $k = n_{\mathrm{DP}} - 1 \ldots 0$. The short horizon problems (2.22) have to be solved for all possible $\boldsymbol{x}_k$. These values are stored, a *tabulation in state space* is performed.

Dynamic Programming with infinitely small time steps leads to the *Hamilton–Jacobi–Bellman (HJB) equation*, see Bellman (1957) or Locatelli (2001), that can be used to determine the optimal control for continuous time systems:

$$-\frac{\partial J}{\partial t}(\boldsymbol{x},t) = \min_{\boldsymbol{u}} \left(L(\boldsymbol{x},\boldsymbol{u}) + \frac{\partial J}{\partial \boldsymbol{x}}(\boldsymbol{x},t)\ \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) \right) \tag{2.23}$$

This partial differential equation has to be solved backwards for $t \in [t_0, t_f]$, starting at the end of the horizon with

$$J(\boldsymbol{x}, t_f) = E(\boldsymbol{x}). \tag{2.24}$$

Remark 2.12 *Optimal controls for state $\boldsymbol{x}$ at time t are obtained from*

$$\boldsymbol{u}^*(\boldsymbol{x},t) = \arg\min_{\boldsymbol{u}} \left(L(\boldsymbol{x},\boldsymbol{u}) + \frac{\partial J}{\partial \boldsymbol{x}}(\boldsymbol{x},t)\ \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) \right) \tag{2.25}$$

subject to the constraints of problem (2.1). The optimal controls depend only on the derivative $\partial J/\partial \boldsymbol{x}$, but not on J itself. If adjoint variables $\boldsymbol{\lambda}(\cdot)$ are introduced as

$$\boldsymbol{\lambda}(t) \ = \ \frac{\partial J}{\partial \boldsymbol{x}}(\boldsymbol{x}(t),t)^T \in \mathbb{R}^{n_x}, \quad t \in [t_0, t_f], \tag{2.26}$$

then the connection to Pontryagin's maximum principle (see section 2.1.1) is obvious. The expression to be minimized in (2.25) is the Hamiltonian. The dynamic equations and transversality conditions for $\boldsymbol{\lambda}(\cdot)$ are obtained by differentiation of the HJB equation (2.23) resp. of the terminal condition (2.24) with respect to $\boldsymbol{x}$:

$$-\dot{\boldsymbol{\lambda}}^T(t) \ = \ \frac{\partial}{\partial \boldsymbol{x}} \left(\mathcal{H}(\boldsymbol{x}(t), \boldsymbol{u}^*(t,\boldsymbol{x},\boldsymbol{\lambda}), \boldsymbol{\lambda}(t)) \right), \quad t \in [t_0, t_f], \tag{2.27}$$

$$\boldsymbol{\lambda}(t_f)^T \ = \ \frac{\partial E}{\partial \boldsymbol{x}}(\boldsymbol{x}(t_f)). \tag{2.28}$$

Dynamic programming (resp. the solution of the partial differential HJB equation) has two advantages when compared to all other methods presented. First, the whole state space is searched, thus an optimal solution is also the global optimum (compare section 2.4). Second, all controls are precomputed once a solution is found – in online optimization the feedback controls can be readily applied. For some specific problems as *Riccati equations*, e.g., Locatelli (2001), analytic solutions can be derived.
In general this is not possible though. The main drawback is the so–called "curse of dimensionality", as a partial differential equation has to be solved in a high–dimensional state space. The HJB equation and dynamic programming are thus mainly used for small scale systems. See Locatelli (2001) for details.

2.2.3 Direct single shooting

In contrast to indirect methods or solution of the HJB equation, direct methods are based upon a transformation into a finite–dimensional optimization problem that

can be solved by nonlinear programming techniques. In direct single shooting, collocation and direct multiple shooting the control functions $\boldsymbol{u}(\cdot)$ are discretized. These methods differ in the way the corresponding state variables are treated, whether a sequential approach is used or a so–called all–at–once approach that solves the optimization problem and the integration of the system at the same time.
In direct single shooting, the sequential approach, the states $\boldsymbol{y}(\cdot)$ on $[t_0, t_f]$ are regarded as dependent variables. Numerical integration is used to obtain the states as functions $\boldsymbol{y}(\cdot; \boldsymbol{y_0}, \boldsymbol{q}, \boldsymbol{p})$ of finitely many control parameters

$$\boldsymbol{q} = (\boldsymbol{q_0}, \boldsymbol{q_1}, \ldots, \boldsymbol{q_{n_{ss}-1}})^T.$$

A piecewise approximation $\hat{\boldsymbol{u}}$ of the control functions $\boldsymbol{u}$ on a fixed grid is defined by

$$\hat{\boldsymbol{u}}(t) \;=\; \boldsymbol{\varphi_i}(t, \boldsymbol{q_i}), \quad t \in [t_i, t_{i+1}], \quad i = 0, \ldots, n_{\mathrm{ss}} - 1, \tag{2.29}$$

using control parameter vectors $\boldsymbol{q_i}$. In practice the functions $\boldsymbol{\varphi_i}$ are typically vectors of constant or linear functions. In each iteration of the solution procedure, an ODE

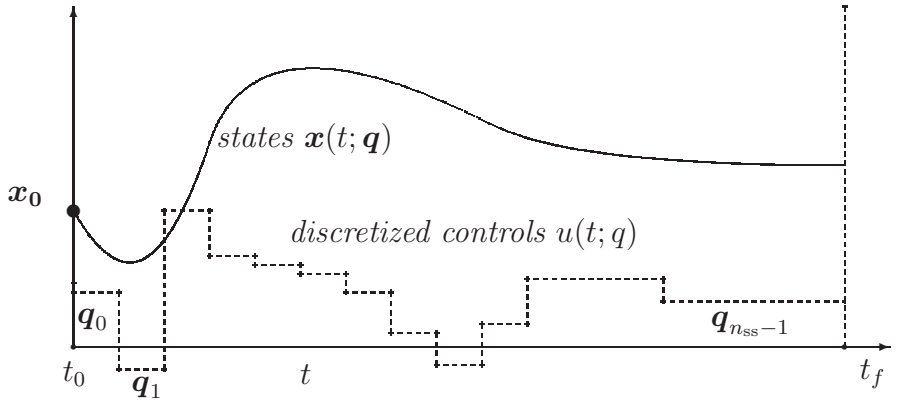

Figure 2.5: Illustration of direct single shooting. The controls are discretized, the corresponding states obtained by integration. The interval lengths do not have to be equidistant. E.g., the last interval may be larger than the preceding ones, as is typical for online optimization problems.

resp. a DAE has to be solved. The optimal control problem (1.14) is then a problem in the finite–dimensional variables $\boldsymbol{q}$, the initial values $\boldsymbol{y_0}$ and the parameters $\boldsymbol{p}$ only. If we write them in one n_ξ–dimensional vector

$$\boldsymbol{\xi} \;=\; (\boldsymbol{x_0}, \boldsymbol{z_0}, \boldsymbol{q_0}, \ldots, \boldsymbol{q_{n_{ss}-1}}, \boldsymbol{p})^T, \tag{2.30}$$

we obtain a finite–dimensional optimization problem

$$\min_{\boldsymbol{\xi}} \quad F(\boldsymbol{\xi}) \tag{2.31a}$$

$$\text{subject to} \quad \boldsymbol{G}(\boldsymbol{\xi}) = \boldsymbol{0}, \tag{2.31b}$$

$$\boldsymbol{H}(\boldsymbol{\xi}) \leq \boldsymbol{0}. \tag{2.31c}$$

Here the objective function $F(\boldsymbol{\xi})$ is given by

$$F(\boldsymbol{\xi}) \;=\; \Phi[\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{p}], \tag{2.32}$$

where the functions $\boldsymbol{x}(\cdot;\boldsymbol{\xi})$ and $\boldsymbol{z}(\cdot;\boldsymbol{\xi})$ are uniquely determined by $\boldsymbol{\xi}$ via integration. The interior point inequality resp. equality constraints as well as bounds on parameters and controls are subsumed into the finite–dimensional constraints $\boldsymbol{G}(\cdot)$ and $\boldsymbol{H}(\cdot)$. There are different ways to treat the infinite–dimensional control and path constraints, e.g., inclusion into the objective function by penalty terms or discretization to pointwise constraints that can be added to $\boldsymbol{G}(\cdot)$ and $\boldsymbol{H}(\cdot)$.
Problem (2.31) can be solved with a finite–dimensional optimization solver, e.g. by sequential quadratic programming, compare section 2.3.1.
Direct single shooting is an often–used method of optimization, in particular in engineering applications, as it is easily implemented if ODE/DAE solver and NLP solvers are available. Furthermore only few degrees of freedom are left in problem (2.31), if the number of controls is small and the initial values are fixed.
The drawback of direct single shooting is that only knowledge about the controls can be brought in, while no knowledge about the process itself, that is about $\boldsymbol{x}(\cdot)$, can be used for the initialization of the optimization problem. This is, e.g., crucial in tracking problems and whenever the states $\boldsymbol{y}(\cdot;\boldsymbol{\xi})$ depend nonlinearly on $\boldsymbol{q}$ or the system is unstable.

2.2.4 Collocation

Often the behavior of the process itself is well–known, while the controls $\boldsymbol{q}$ are what one is looking for. To take advantage of this fact, in collocation the states are not regarded as dependent variables any more, but discretized too. Collocation goes back to Tsang *et al.* (1975) and has been extended, e.g., by Bär (1984), Biegler (1984) and Schulz (1998).
In collocation not only the controls, but also the states are discretized on a fine grid with n_{col} time points and node values $\boldsymbol{s_i^x} \approx \boldsymbol{x}(t_i)$ resp. $\boldsymbol{s_i^z} \approx \boldsymbol{z}(t_i)$. The ODE

$$0 \;=\; \dot{\boldsymbol{x}}(t) - \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)), \quad t \in [0, t_f] \tag{2.33}$$

is replaced by finitely many equality constraints

$$\tilde{\boldsymbol{G}}_{\boldsymbol{i}}(\boldsymbol{q_i}, \boldsymbol{s_i^x}, \boldsymbol{s_{i+1}^x}) \;=\; \boldsymbol{0}, \quad i = 0 \ldots n_{\mathrm{col}} - 1, \tag{2.34a}$$

e.g., by the first–order approximation

$$\tilde{\boldsymbol{G}}_{\boldsymbol{i}}(\boldsymbol{q_i}, \boldsymbol{s_i^x}, \boldsymbol{s_{i+1}^x}) \;=\; \frac{\boldsymbol{s_{i+1}^x} - \boldsymbol{s_i^x}}{t_{i+1} - t_i} - \boldsymbol{f}\left(\frac{\boldsymbol{s_i^x} + \boldsymbol{s_{i+1}^x}}{2}, \boldsymbol{q_i}\right). \tag{2.34b}$$

A similar idea is used to treat algebraic variables. The Lagrange term is replaced by summation formulae. Path and control constraints are evaluated on the discretization grid. Subsuming all constraints one obtains a large scale, but sparse NLP in the n_ξ–dimensional vector

$$\boldsymbol{\xi} \;=\; (\boldsymbol{s_0^x}, \boldsymbol{s_0^z}, \boldsymbol{q_0}, \boldsymbol{s_1^x}, \boldsymbol{s_1^z}, \ldots, \boldsymbol{q_{n_{\mathrm{col}}-1}}, \boldsymbol{s_{n_{\mathrm{col}}}^x}, \boldsymbol{s_{n_{\mathrm{col}}}^z}, \boldsymbol{p})^T, \tag{2.35}$$

that is given by

$$\begin{aligned} \min_{\boldsymbol{\xi}} \quad & F(\boldsymbol{\xi}) && \text{(2.36a)} \\ \text{subject to} \quad & \boldsymbol{G}(\boldsymbol{\xi}) = \mathbf{0}, && \text{(2.36b)} \\ & \boldsymbol{H}(\boldsymbol{\xi}) \leq \mathbf{0}. && \text{(2.36c)} \end{aligned}$$

In contrast to program (2.31) we do not have any dependent variables $\boldsymbol{y}(\cdot)$ any more that have to be determined by integration in every iteration. This structured NLP can be solved with an appropriate nonlinear solver, e.g., with an interior point solver or a tailored sequential quadratic programming method for sparse problems.
As stated in the beginning of this section, collocation allows to use the knowledge about the process behavior in the initialization of the optimization problem. Therefore it is possible to treat highly nonlinear systems efficiently. Furthermore the algorithm is stable if the problem is well–posed, e.g., an unstable system with a terminal constraint, because small perturbations do not spread over the whole time horizon, but are damped out by the tolerance in the matching conditions. Sequential approaches are only stable, if the system itself is stable. Path and terminal constraints are handled in a more robust way than in direct single shooting. Although the optimization problem gets typically very large in the number of variables, it has been applied successfully to large–scale problems too, making use of structure–exploiting algorithms.
An adaptivity in time cannot be incorporated in a straightforward way, as it changes the dimensions of the underlying nonlinear program. This is crucial, though, especially for stiff systems. Stiff systems require small step sizes in the integration process, thus the underlying grid has to be chosen very fine – often too rigorous for the whole time horizon, if adaptivity is not used. Furthermore, such an adaptive scheme would be solver–specific, while integrator–based methods allow to use any available state–of–the–art DAE solver. This is a disadvantage of collocation and the reason why we chose direct multiple shooting as the basis for our methods. See Weiser & Deuflhard (2001) for a hybrid approach between indirect methods and collocation, based on an interior point algorithm.

2.3 Direct multiple shooting

Direct multiple shooting goes back to the diploma thesis of Plitt (1981) supervised by Georg Bock. It was first published in Bock & Plitt (1984) and has been extended and applied by different researchers over the years, recently e.g., by Santos *et al.* (1995), Franke *et al.* (2002), Leineweber *et al.* (2003), Brandt-Pollmann (2004), Terwen *et al.* (2004) or Schäfer (2005). It combines the advantages of direct single shooting and collocation. It is also a direct method and based upon a transformation of the infinite–dimensional problem to a finite–dimensional one by a discretization of the control functions, see (2.29). As in collocation, a time grid of *multiple shooting nodes*

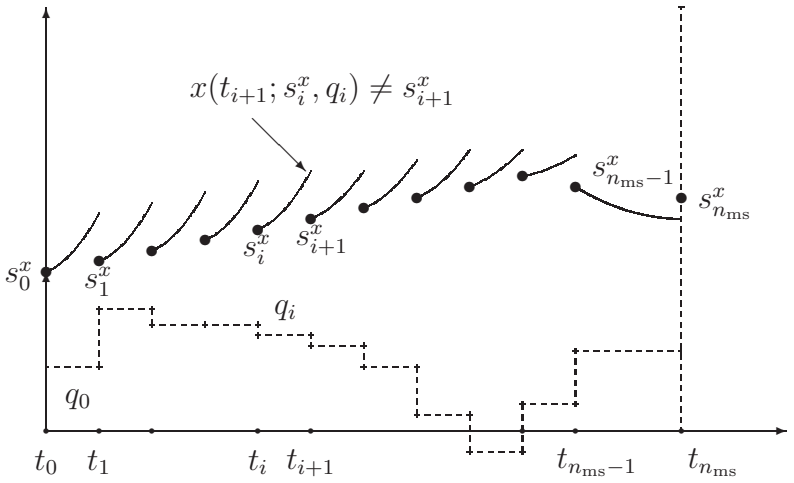

Figure 2.6: Illustration of direct multiple shooting. The controls are discretized, the corresponding states obtained by piecewise integration. The matching conditions are violated in this scheme — the overall trajectory is not yet continuous.

is introduced,

$$t_0 \leq t_1 \leq \cdots \leq t_{n_{\rm ms}} = t_f, \tag{2.37}$$

with corresponding node values $\boldsymbol{s_i^x} \approx \boldsymbol{x}(t_i)$ in $\mathbb{R}^{n_x}$ and $\boldsymbol{s_i^z} \approx \boldsymbol{z}(t_i)$ in $\mathbb{R}^{n_z}$, from now on $0 \leq i < n_{\rm ms}$. This grid is coarser though and all values $\boldsymbol{x}(t)$ in between are obtained by integration with an ODE/DAE solver. The DAE is solved independently on each of the multiple shooting intervals. On interval $[t_i, t_{i+1}]$ the initial values of differential and algebraic states are given by node values $\boldsymbol{s_i^x}$ and $\boldsymbol{s_i^z}$, respectively. The algebraic equations (1.18c) are relaxed (see Bock *et al.* (1988), Leineweber (1999)). They enter as conditions in t_i into the NLP. Continuity of the state trajectory at the multiple shooting grid points

$$\boldsymbol{s_{i+1}^x} \;=\; \boldsymbol{x}(t_{i+1}; \boldsymbol{s_i^x}, \boldsymbol{s_i^z}, \boldsymbol{q_i}, \boldsymbol{p}) \tag{2.38}$$

is incorporated by constraints into the optimization problem. Here $\boldsymbol{x}(\cdot)$ denotes the differential part of the DAE solution on interval $[t_i, t_{i+1}]$ with initial values $\boldsymbol{s_i^x}, \boldsymbol{s_i^z}$ at time t_i. These equations are not necessarily satisfied during the iterations of the nonlinear programming algorithm used to solve the NLP, but only when convergence has been achieved. Direct multiple shooting is therefore a so–called all–at–once approach that solves the dynamic equations and the optimization problem at the same time opposed to the sequential approach of single shooting that creates a continuous trajectory in every iteration. Figure 2.6 illustrates the concept of direct multiple shooting.

The control variables $\boldsymbol{q_i}$, the global parameters $\boldsymbol{p}$, that may include the time horizon lengths $h_i = \tilde{t}_{i+1} - \tilde{t}_i$, and the node values $\boldsymbol{s_i^x}, \boldsymbol{s_i^z}$ are the degrees of freedom of the discretized and parameterized optimal control problem. If we write them in one

n_ξ–dimensional vector

$$\boldsymbol{\xi} \;=\; (\boldsymbol{s_0^x}, \boldsymbol{s_0^z}, \boldsymbol{q_0}, \boldsymbol{s_1^x}, \boldsymbol{s_1^z}, \ldots, \boldsymbol{q_{n_{\mathrm{ms}}-1}}, \boldsymbol{s_{n_{\mathrm{ms}}}^x}, \boldsymbol{s_{n_{\mathrm{ms}}}^z}, \boldsymbol{p})^T, \tag{2.39}$$

similar to (2.35), but with less discretization points $n_{\mathrm{ms}} < n_{\mathrm{col}}$, rewrite the objective function as $F(\boldsymbol{\xi})$, subsume all equality constraints with the continuity conditions (2.38) into a function $\boldsymbol{G}(\boldsymbol{\xi})$ and all inequality constraints into a function $\boldsymbol{H}(\boldsymbol{\xi})$, then the resulting NLP can be written as

$$\min_{\boldsymbol{\xi}} \quad F(\boldsymbol{\xi}) \tag{2.40a}$$

$$\text{subject to} \quad \boldsymbol{G}(\boldsymbol{\xi}) = \boldsymbol{0}, \tag{2.40b}$$

$$\boldsymbol{H}(\boldsymbol{\xi}) \leq \boldsymbol{0}. \tag{2.40c}$$

This NLP can be solved with tailored iterative methods exploiting the structure of the problem, e.g., by sequential quadratic programming that will be highlighted in subsection 2.3.1. In this procedure special care has to be taken how derivative information is generated, as the model equations contain differential equations. This will be investigated further in subsection 2.3.2.

Due to the direct approach and the parameterization of the state space, direct multiple shooting is similar to collocation in many aspects and shares its advantages. In particular, knowledge about the process behavior may be used for the initialization of the optimization problem. The algorithm is stable, as mentioned already for collocation. This is important for the treatment of unstable and nonlinear systems, but also plays a role in the context of local or global optima, compare section 2.4. As in collocation, path and terminal constraints are handled in a more robust way than in direct single shooting. The main difference to the other all–at–once approach, collocation, lies in the fact that the differential equations are still solved by integration. This allows the usage of state–of–the–art error–controlled DAE integrators.

In addition to the conceptual advantages mentioned above, direct multiple shooting has a very beneficial structure that can be extensively exploited. The use of structure exploiting condensing algorithms for the Hessian as proposed in Plitt (1981) and Bock & Plitt (1984) reduces the dimensions of the matrices in the quadratic programs considerably to the size of those of the direct single shooting approach. Together with high–rank block–wise updates it reduces the computing time considerably. Other structure exploiting measures are the relaxed formulation of algebraic conditions and invariants that allows inconsistent iterates, Bock *et al.* (1988), Schulz *et al.* (1998), and the projection onto an invariant manifold to improve convergence and reduce the degrees of freedom, Schlöder (1988), Schulz *et al.* (1998) and Schäfer (2005). Furthermore the intrinsic parallel structure with decoupled problems can be used for an efficient parallelization, Gallitzendörfer & Bock (1994).
For more details on direct multiple shooting, see one of the aforementioned works or in particular Bock & Plitt (1984), Leineweber (1999) or Leineweber *et al.* (2003). An efficient implementation of the described method is the software package `MUSCOD-II`, see Diehl *et al.* (2001).

2.3.1 Sequential Quadratic Programming

For all direct methods a NLP of the form (2.40) has to be solved. For general theory and algorithms for finite–dimensional constrained optimization problems we refer to standard textbooks in the field, e.g., Fletcher (1987) or Nocedal & Wright (1999). At this point we will focus on one very efficient way to solve NLP (2.40), namely on sequential quadratic programming (SQP), first proposed by Wilson (1963), and only mention very shortly definitions and results from the general optimization theory. To state necessary and sufficient conditions of optimality we need the concepts of the Lagrangian, the active set and linear independence constraint qualification.

Definition 2.13 (Lagrangian)
The Lagrangian $\mathcal{L}$ of a constrained nonlinear program is defined by

$$\mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \mathbf{F}(\boldsymbol{\xi}) + \boldsymbol{\lambda}^{\mathbf{T}} \boldsymbol{G}(\boldsymbol{\xi}) + \boldsymbol{\mu}^{\mathbf{T}} \boldsymbol{H}(\boldsymbol{\xi}), \tag{2.41}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^{n_G}$ and $\boldsymbol{\mu} \in \mathbb{R}^{n_H}$ are the Lagrange multipliers of the systems equality resp. inequality constraints.

Definition 2.14 (Active set)
The active set $\mathcal{A}(\boldsymbol{\xi})$ of an inequality–constrained NLP at a point $\boldsymbol{\xi}$ is the set of all indices $1 \leq i \leq n_H$ for which the corresponding equality constraint is active,

$$\mathcal{A}(\boldsymbol{\xi}) = \{i : H_i(\boldsymbol{\xi}) = 0, \;\; 1 \leq i \leq n_H\}. \tag{2.42}$$

Definition 2.15 (Linear independence constraint qualification)
If the gradients of the equality constraints $\frac{\partial \boldsymbol{G}_i}{\partial \boldsymbol{\xi}}$, $i = 1 \ldots n_G$ and of the active inequality constraints $\frac{\partial \boldsymbol{H}_i}{\partial \boldsymbol{\xi}}$, $i \in \mathcal{A}(\boldsymbol{\xi})$ at a point $\boldsymbol{\xi}$ are linearly independent, we say that the linear independence constraint qualification (LICQ) holds.

First order necessary conditions of optimality are given by the following theorem of Karush (1939), Kuhn & Tucker (1951).

Theorem 2.16 (First order necessary conditions of optimality)
Let $\boldsymbol{\xi}^*$ be a local minimizer of NLP (2.40) for which (LICQ) holds. Then there exist unique Lagrange multipliers $\boldsymbol{\lambda}^* \in \mathbb{R}^{n_G}$ and $\boldsymbol{\mu}^* \in \mathbb{R}^{n_G}$ such that at $(\boldsymbol{\xi}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ the following conditions hold:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\xi}}(\boldsymbol{\xi}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \mathbf{0} \tag{2.43a}$$

$$\boldsymbol{G}(\boldsymbol{\xi}^*) = \mathbf{0} \tag{2.43b}$$

$$\boldsymbol{H}(\boldsymbol{\xi}^*) \leq \mathbf{0} \tag{2.43c}$$

$$\boldsymbol{\mu}^* \geq \mathbf{0} \tag{2.43d}$$

$$\boldsymbol{\mu}^{*T} \, \boldsymbol{H}(\boldsymbol{\xi}^*) = \mathbf{0} \tag{2.43e}$$

$(\boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ is called a Karush Kuhn Tucker (KKT) point, if conditions (2.43) hold. A proof for theorem 2.16 can be found in Fletcher (1987). Second order necessary conditions are given by

Theorem 2.17 (Second order necessary conditions of optimality)
Let $(\boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ be a KKT point and assume that (LICQ) holds.
For every vector $\boldsymbol{\Delta\xi} \in \mathbb{R}^{n_\xi}$ with

$$\frac{\partial \boldsymbol{G}_i}{\partial \boldsymbol{\xi}}(\boldsymbol{\xi})\ \boldsymbol{\Delta\xi} \;=\; \mathbf{0}, \quad i = 1 \ldots n_G \tag{2.44a}$$

$$\frac{\partial \boldsymbol{H}_i}{\partial \boldsymbol{\xi}}(\boldsymbol{\xi})\ \boldsymbol{\Delta\xi} \;=\; \mathbf{0}, \quad i \in \mathcal{A}(\boldsymbol{\xi}) \tag{2.44b}$$

it holds that

$$\boldsymbol{\Delta\xi}^T\ \frac{\partial^2 \mathcal{L}}{\partial \boldsymbol{\xi}^2}(\boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu})\ \boldsymbol{\Delta\xi} \;\geq\; \mathbf{0}. \tag{2.45}$$

Sufficient conditions are given by

Theorem 2.18 (Second order sufficient conditions of optimality)
Let $(\boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ be a KKT point and assume that (LICQ) holds.
For every vector $\boldsymbol{\Delta\xi} \in \mathbb{R}^{n_\xi}$ with

$$\frac{\partial \boldsymbol{G}_i}{\partial \boldsymbol{\xi}}(\boldsymbol{\xi})\ \boldsymbol{\Delta\xi} \;=\; \mathbf{0}, \quad i = 1 \ldots n_G \tag{2.46a}$$

$$\frac{\partial \boldsymbol{H}_i}{\partial \boldsymbol{\xi}}(\boldsymbol{\xi})\ \boldsymbol{\Delta\xi} \;=\; \mathbf{0}, \quad i \in \mathcal{A}(\boldsymbol{\xi}) \text{ and } \mu_i > 0 \tag{2.46b}$$

$$\frac{\partial \boldsymbol{H}_i}{\partial \boldsymbol{\xi}}(\boldsymbol{\xi})\ \boldsymbol{\Delta\xi} \;\geq\; \mathbf{0}, \quad i \in \mathcal{A}(\boldsymbol{\xi}) \text{ and } \mu_i = 0 \tag{2.46c}$$

$$\tag{2.46d}$$

it holds that

$$\boldsymbol{\Delta\xi}^T\ \frac{\partial^2 \mathcal{L}}{\partial \boldsymbol{\xi}^2}(\boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu})\ \boldsymbol{\Delta\xi} \;>\; \mathbf{0}. \tag{2.47}$$

A proof for these theorems can be found, e.g., in the textbook of Nocedal & Wright (1999).
In the following we will also use the operator ∇ for derivatives with respect to $\boldsymbol{\xi}$ and leave away the argument for notational convenience. Now we do want to sketch the algorithm we use to solve the occurring nonlinear programs. The general form of an SQP algorithm is the following.

Algorithm 2.1 (SQP)

1. Set the iteration counter $k = 0$ and start with guesses $\boldsymbol{\xi^0}$ for the unknown parameters (2.39) and $\boldsymbol{\lambda^0}$, $\boldsymbol{\mu^0}$ for the Lagrange multipliers.

2. Evaluate $F(\boldsymbol{\xi^k})$, $\boldsymbol{G}(\boldsymbol{\xi^k})$, $\boldsymbol{H}(\boldsymbol{\xi^k})$ and derivatives $\nabla F(\boldsymbol{\xi^k})$, $\nabla \boldsymbol{G}(\boldsymbol{\xi^k})$ and $\nabla \boldsymbol{H}(\boldsymbol{\xi^k})$ with respect to $\boldsymbol{\xi}$ by solution of DAEs. Calculate $\mathbf{H^k}$.

3. Compute correction term $\boldsymbol{\Delta\xi}$ and Lagrange multipliers $\tilde{\boldsymbol{\lambda}}$, $\tilde{\boldsymbol{\mu}}$ by solution of the quadratic program

$$\begin{aligned} \min_{\boldsymbol{\Delta\xi}} \quad & \nabla F(\boldsymbol{\xi^k})^T \boldsymbol{\Delta\xi} + \frac{1}{2} \Delta\boldsymbol{\xi}^T \mathbf{H^k} \boldsymbol{\Delta\xi} && \text{(2.48a)} \\ \text{subject to} \quad & \\ & \boldsymbol{G}(\boldsymbol{\xi^k}) + \nabla \boldsymbol{G}(\boldsymbol{\xi^k})^T \boldsymbol{\Delta\xi} = \mathbf{0}, && \text{(2.48b)} \\ & \boldsymbol{H}(\boldsymbol{\xi^k}) + \nabla \boldsymbol{H}(\boldsymbol{\xi^k})^T \boldsymbol{\Delta\xi} \leq \mathbf{0}. && \text{(2.48c)} \end{aligned}$$

4. Perform steps

$$\begin{aligned} \boldsymbol{\xi^{k+1}} &= \boldsymbol{\xi^k} + \alpha\, \Delta\boldsymbol{\xi} && \text{(2.49a)} \\ \boldsymbol{\lambda^{k+1}} &= \boldsymbol{\lambda^k} + \alpha \left(\tilde{\boldsymbol{\lambda}} - \boldsymbol{\lambda^k} \right) && \text{(2.49b)} \\ \boldsymbol{\mu^{k+1}} &= \boldsymbol{\mu^k} + \alpha \left(\tilde{\boldsymbol{\mu}} - \boldsymbol{\mu^k} \right) && \text{(2.49c)} \end{aligned}$$

   with step length α.

5. If terminal condition is not fulfilled, increase k and `GOTO 2`.

The SQP algorithm is based on a series of quadratic approximations of the nonlinear program. There are different versions of it that differ mainly in the way how the approximation of the Hessian of the Lagrangian,

$$\mathbf{H^k} \approx \frac{\partial^2 \mathcal{L}}{\partial \boldsymbol{\xi}^2}(\boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \qquad (2.50)$$

is calculated, how the step length $\alpha \in (0, 1]$ for the globalization is determined and which terminal condition is used. The most common step length methods are based upon *line search*, *trust region* or *watchdog* techniques.
The termination criterion is typically either the value of a *merit function*, the KKT conditions being inside a certain tolerance or a small increment of the Lagrangian in the search direction, $||\nabla \mathcal{L} \boldsymbol{\Delta\xi}|| \leq \varepsilon$. $\mathbf{H^k}$ can be the exact Hessian or an approximation obtained, e.g., by update techniques. The QP can be solved by active set strategies, interior point methods or crossover techniques (e.g., Huber (1998)). For details and further references see the aforementioned textbooks.

The quadratic program (2.48) has been chosen such that, if $\mathbf{H^k}$ is the exact Hessian, the original Lagrangian and the Lagrangian of the QP are identical up to second order as we have

$$\frac{\partial \mathcal{L}^{\mathrm{QP}}}{\partial \Delta \boldsymbol{\xi}} = \nabla F + \mathbf{H^k} \Delta \boldsymbol{\xi} + \boldsymbol{\lambda}^T \nabla \boldsymbol{G} + \boldsymbol{\mu}^T \nabla \boldsymbol{H}. \tag{2.51}$$

This guarantees that an optimal solution of the nonlinear program is also a minimizer of the quadratic program.

Remark 2.19 *If $\Delta\boldsymbol{\xi}$ fulfills conditions (2.44), it will be orthogonal to the gradients of the equality and active inequality constraints. Furthermore for inactive constraints with $H_i < 0$ we have $\mu_i = 0$ because of (2.43e). Thus we have*

$$\nabla \mathcal{L} \;=\; \nabla F \tag{2.52}$$

and the minimization of QP (2.48) corresponds to minimizing the second order approximation of the Lagrangian.

The convergence rates of SQP algorithms are deduced from corresponding Newton methods. We have

Theorem 2.20 (Equivalence between SQP and Newton's method) *If $\mathbf{H^k} = \frac{\partial^2 \mathcal{L}}{\partial \boldsymbol{\xi}^2}(\boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ and $\alpha = 1$, the SQP algorithm is equivalent to Newton's method for the KKT conditions of the nonlinear optimization problem.*

and therewith locally quadratic convergence of the SQP method with exact Hessian. In general $\mathbf{H^k}$ will be an approximation of the Hessian only, as evaluations of the second derivatives would be computationally expensive. Nevertheless superlinear convergence can be achieved making use of appropriate update schemes. Again, we refer to the textbooks above for a general treatment and to Boggs *et al.* (1982) or Boggs & Tolle (1995) for details. The methods available in our software implementation are described in detail in Leineweber (1999). Further techniques exploiting special structures of the QP arising from the direct multiple shooting parameterization are described in Schlöder (1988) and Schäfer *et al.* (2003).

2.3.2 Derivatives

For derivative–based optimization methods derivatives with respect to the variables $\boldsymbol{\xi}$ are needed. When the NLP (2.40) is obtained from the direct multiple shooting method, the derivatives $\nabla F(\boldsymbol{\xi})$, $\nabla \boldsymbol{G}(\boldsymbol{\xi})$ and $\nabla \boldsymbol{H}(\boldsymbol{\xi})$ depend upon the *variational trajectories*

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{q}}, \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{s}}, \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{p}} \tag{2.53}$$

that have to be calculated together with the solution $\boldsymbol{y} = (\boldsymbol{x}, \boldsymbol{z})$ of the system (1.14b,1.14c). $\boldsymbol{y}$ is also referred to as *nominal trajectory*. The derivatives have to

be calculated with a certain accuracy required by the optimization algorithm, this typically takes most of the overall computing time to solve optimal control problems.

The classical way to obtain an approximation for the variational trajectories is called *external numerical differentiation* (END) and based upon the difference quotient. One approximates

$$\frac{\partial \boldsymbol{x}(t,\boldsymbol{\xi})}{\partial \boldsymbol{\xi}}\,\boldsymbol{\Delta\xi} \;=\; \frac{\boldsymbol{x}(t,\boldsymbol{\xi}+\varepsilon\boldsymbol{\Delta\xi})-\boldsymbol{x}(t,\boldsymbol{\xi})}{\varepsilon}+\mathcal{O}(\varepsilon) \tag{2.54}$$

by neglecting the ε–dependent terms. $\boldsymbol{x}(t,\boldsymbol{\xi}+\varepsilon\boldsymbol{\Delta\xi})$ is calculated for perturbed variables $\boldsymbol{\xi}$ in a direction $\boldsymbol{\Delta\xi}$ with a given factor ε.

External numerical differentiation gets its name from the fact that the trajectories are differentiated outside the discretization scheme of the DAE integrator. This is a severe disadvantage. Typically already for small perturbations of $\boldsymbol{\xi}$ one gets a different step size, order and error control. These adaptive components cannot be differentiated, though, as was already stressed by Ortega & Rheinboldt (1966) and later on by Gear & Vu (1983). If all adaptive components are fixed, the accuracy of nominal and varied trajectories has to be increased dramatically, leading to an augmented overall computing time. As a rule of thumb the accuracy of the derivative is at best about half the digits of the accuracy of the nominal trajectory.

A more sophisticated approach to obtain approximations of variational trajectories is based upon a differentiation inside the discretization scheme of the integrator and is therefore called *internal numerical differentiation* (IND). Internal numerical differentiation goes back to Bock (1981) and is also stable for low integration accuracies, Bock (1987). If one differentiates the parameterized DAE system on a multiple shooting interval,

$$\begin{aligned}
\dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{z}(t),\boldsymbol{q},\boldsymbol{p}), && \text{(2.55a)}\\
\boldsymbol{0} &= \boldsymbol{g}(\boldsymbol{x}(t),\boldsymbol{z}(t),\boldsymbol{q},\boldsymbol{p}), && \text{(2.55b)}\\
\boldsymbol{x}(t_i) &= \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}, && \text{(2.55c)}\\
\boldsymbol{z}(t_i) &= \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}, && \text{(2.55d)}
\end{aligned}$$

where $t\in[t_i,t_{i+1}]$, $i\in\{1,\dots n_{\mathrm{ms}-1}\}$, with respect to $\boldsymbol{\xi}=(\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}},\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}},\boldsymbol{q},\boldsymbol{p})^T$, one obtains the *variational DAE* system consisting of a differential

$$\begin{pmatrix} \dot{W}_{\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}}^{\boldsymbol{x}} \\ \dot{W}_{\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}}^{\boldsymbol{x}} \\ \dot{W}_{\boldsymbol{q}}^{\boldsymbol{x}} \\ \dot{W}_{\boldsymbol{p}}^{\boldsymbol{x}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_{\boldsymbol{x}} & \boldsymbol{f}_{\boldsymbol{z}} & \boldsymbol{f}_{\boldsymbol{q}} & \boldsymbol{f}_{\boldsymbol{p}} \end{pmatrix} \cdot \begin{pmatrix} W_{\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}}^{\boldsymbol{x}} & W_{\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}}^{\boldsymbol{x}} & W_{\boldsymbol{q}}^{\boldsymbol{x}} & W_{\boldsymbol{p}}^{\boldsymbol{x}} \\ W_{\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}}^{\boldsymbol{z}} & W_{\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}}^{\boldsymbol{z}} & W_{\boldsymbol{q}}^{\boldsymbol{z}} & W_{\boldsymbol{p}}^{\boldsymbol{z}} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{pmatrix}, \tag{2.56a}$$

(2.56b)

and an algebraic part

$$\mathbf{0} \;=\; \begin{pmatrix} \boldsymbol{g_x} & \boldsymbol{g_z} & \boldsymbol{g_q} & \boldsymbol{g_p} \end{pmatrix} \cdot \begin{pmatrix} W^{\boldsymbol{x}}_{\boldsymbol{s}^{\boldsymbol{x}}_i} & W^{\boldsymbol{x}}_{\boldsymbol{s}^{\boldsymbol{z}}_i} & W^{\boldsymbol{x}}_{\boldsymbol{q}} & W^{\boldsymbol{x}}_{\boldsymbol{p}} \\ W^{\boldsymbol{z}}_{\boldsymbol{s}^{\boldsymbol{x}}_i} & W^{\boldsymbol{z}}_{\boldsymbol{s}^{\boldsymbol{z}}_i} & W^{\boldsymbol{z}}_{\boldsymbol{q}} & W^{\boldsymbol{z}}_{\boldsymbol{p}} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} \end{pmatrix} \tag{2.56c}$$

with initial values

$$\begin{pmatrix} W^{\boldsymbol{x}}_{\boldsymbol{s}^{\boldsymbol{x}}_i} \\ W^{\boldsymbol{x}}_{\boldsymbol{s}^{\boldsymbol{z}}_i} \\ W^{\boldsymbol{x}}_{\boldsymbol{q}} \\ W^{\boldsymbol{x}}_{\boldsymbol{p}} \end{pmatrix} (t_i) \;=\; \begin{pmatrix} \boldsymbol{I} \\ \boldsymbol{I} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}. \tag{2.56d}$$

Here $W^{\cdot}_{\cdot}$ denote the *Wronskian* matrices of the system trajectories,

$$W^{\boldsymbol{x}}_{\boldsymbol{s}^{\boldsymbol{x}}_i} \;=\; \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{s}^{\boldsymbol{x}}_i}(t; \boldsymbol{s}^{\boldsymbol{x}}_i, \boldsymbol{s}^{\boldsymbol{z}}_i, \boldsymbol{q}, \boldsymbol{p}), \quad W^{\boldsymbol{z}}_{\boldsymbol{s}^{\boldsymbol{x}}_i} = \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{s}^{\boldsymbol{x}}_i}(t; \boldsymbol{s}^{\boldsymbol{x}}_i, \boldsymbol{s}^{\boldsymbol{z}}_i, \boldsymbol{q}, \boldsymbol{p}), \tag{2.57}$$

and equivalently for subscripts $\boldsymbol{s}^{\boldsymbol{z}}_{\boldsymbol{i}}$, $\boldsymbol{q}$ and $\boldsymbol{p}$. Note that for a relaxation of the algebraic equations, see Bock *et al.* (1988), Bauer (1999) or Leineweber (1999), an additional term has to be included in (2.55b).

The variational DAE system can be solved by two different IND techniques. The first one, called *varied trajectories*, uses finite differences to approximate the solution. An initial value problem is solved for the nominal system and for a perturbed $\boldsymbol{\xi}$ yielding a first order approximation of the derivative. In contrast to external numerical differentiation it is possible to vary the adaptive components of the integrator, as long as these are identical for nominal and variational trajectory. The main drawback this approach shares with external numerical differentiation is the difficulty to choose an appropriate step size ε and the reduced accuracy due to the first order approximation. These are also the reasons why we do not follow this approach.

The second one is the solution of the linear equation system obtained by a discretization of system (2.56). This can be done by either Newton's method or by a direct solution of the linear equation system. The discretization of system (2.56) can be performed very efficiently with the *backward differentiation formulae* (BDF). For an overview of multi–step integration techniques see, e.g., Ascher & Petzold (1998). The main advantage of this approach is that iteration matrices can be used for the nominal as well as for the variational solutions.

It can be shown that formulation of the BDF scheme for the variational DAEs and differentiation of the BDF scheme for the nominal DAEs lead to the same equations. This principle, the commutativity of integration scheme and differentiation operator, leads to an exact numerical derivative of the numerical solution of the DAE if exact derivatives of the model equations $\boldsymbol{f}$ and $\boldsymbol{g}$ are supplied. This is referred to as the *analytical limit* of IND, Bock (1983).

The accuracy of this approach and the efficient computation of required iteration matrices lead to the choice to use internal numerical differentiation for derivative generation in our work. A detailed discussion and details, e.g., about computation of directional derivatives, can be found in Bauer (1999). In Brandt-Pollmann (2004) one finds a detailed discussion on how to exploit sparsity and efficiently combine these concepts with *automatic differentiation* to reach the analytical limit.

2.4 Global optimization

Both the conditions of optimality given by the maximum principle for the infinite–dimensional optimal control problem, see section 2.1.1, as the necessary and sufficient conditions for the finite–dimensional discretized nonlinear program (2.40), see section 2.3.1, are conditions that guarantee a *local* minimum only. As discussed before, the solution of the Hamilton–Jacobi–Bellman equation is the only approach guaranteeing a global optimum. This approach is prohibitive though for problems involving more than only few variables because of the curse of dimensionality. The question is risen, what can be said about the relation between local and global minima in discretized optimal control problems.

It is a well–known fact that a sufficient condition for a local optimum to be also globally optimal is convexity of the objective function as well as of the feasible set[5]. For NLPs of the form (2.40) the objective function F has to be convex. The feasible set is determined by equalities and inequalities. It is clear that on the one hand equalities $\boldsymbol{G} = \boldsymbol{0}$ may yield disconnected and thus non–convex feasible sets, if they are nonlinear, no matter if they are convex or non–convex. Inequalities $\boldsymbol{H} \leq \boldsymbol{0}$ on the other hand may yield disconnected feasible sets only if they are non–convex as illustrated in the one–dimensional example $H(x) = -x^2 + 1 \leq 0$ with feasible set $[-\infty, -1] \cup [1, \infty]$. Convexity of F and $\boldsymbol{H}$ and linearity of $\boldsymbol{G}$ do suffice for convexity of the optimization problem (2.40).

An interesting question is when optimal control problems with dynamic equations and Mayer term are convex. Recently, Barton & Lee (2004), Lee *et al.* (2004) investigated this. One important result of Barton & Lee (2004) is that a function $F(\boldsymbol{x}(t;\boldsymbol{\xi}), \boldsymbol{\xi}, t)$ is convex on $\mathbb{R}^{n_\xi}$ if two conditions hold. First the function F has to be convex in both arguments $\boldsymbol{x}$ and $\boldsymbol{\xi}$. Second, $\boldsymbol{x}(t;\boldsymbol{\xi})$ has to be affine with respect to ξ. The basic assumption besides convexity of F and $\boldsymbol{H}$ and linearity of $\boldsymbol{G}$ to guarantee convexity then is that the system can be described by a linear time–variant DAE. If one assumes that the algebraic variables can be computed from (1.14c) and substituted into (1.14b) or that there are no algebraic variables at all and if (1.14b) has the form

$$\dot{\boldsymbol{x}}(t) \;=\; \boldsymbol{A^1}(t)\,\boldsymbol{x}(t) + \boldsymbol{A^2}(t)\,\boldsymbol{u}(t) + \boldsymbol{A^3}(t)\,\boldsymbol{p}, \quad t \in [t_0, t_f] \tag{2.58}$$

with time–dependent matrices $\boldsymbol{A^1}(\cdot)$, $\boldsymbol{A^2}(\cdot)$ and $\boldsymbol{A^3}(\cdot)$, then it is possible to derive analytic properties of the solution $\boldsymbol{x}(\cdot)$. In particular it can be shown that $\boldsymbol{x}(\cdot)$ is

[5] See A.1 in the appendix for a definition of convexity

affine in the controls $\boldsymbol{u}(\cdot)$, the parameters $\boldsymbol{p}$ and the initial value $\boldsymbol{x_0}$. If the controls are discretized according to (2.29) it is also affine in the control parameterization vector $\boldsymbol{q}$ and therefore in $\boldsymbol{\xi}$ as defined by (2.39). The differential state can be expressed by

$$\boldsymbol{x}(\boldsymbol{\xi}, t) \;=\; W^{\boldsymbol{x}}\boldsymbol{\xi} + \boldsymbol{a}(t), \quad t \in [t_0, t_f], \tag{2.59}$$

with the Wronskian $W^{\boldsymbol{x}}$ (2.57) and a time–dependent vector $\boldsymbol{a}(\cdot)$ defined by the solution of the linear time–variant dynamic system. Although it will in general not be possible to obtain an explicit analytic expression for $\boldsymbol{x}(\cdot)$, it is of great help to know that it is affine in the optimization variable to apply the aforementioned theorem.

Remark 2.21 *The above considerations can be applied to the objective function as well as to the constraints. As the sum of convex functions is again convex multistage problems with objective functions of the type (1.18a) can be treated, too. But this holds only for fixed stage lengths* $(\tilde{t}_{i+1} - \tilde{t}_i)$ *and explicit discontinuities, as the affinity property does no longer hold for a time transformation of the DAE.*

Remark 2.22 *A very interesting result, Oldenburg (2005), concerning direct methods of optimal control is that the conditions for convexity lead to exactly the same set of restrictions for direct single shooting, collocation and direct multiple shooting. Thus there is no advantage of one method over the other with respect to guaranteed convexity of the optimal control problem.*

For general optimal control problems, convexity cannot be assumed, though, and methods of global optimization have to be applied if globality of the solution is a practical issue. In nonlinear programming the methodology is well developed by now. Progress has been made in rigorous resp. complete methods that are typically based upon *spatial Branch & Bound* schemes, underestimation of the objective function and overestimation of the feasible set (see figure 2.7) by appropriate convex functions, Falk & Soland (1969), Tawarmalani & Sahinidis (2002) or Floudas *et al.* (2005), as well as additional techniques, e.g., *interval arithmetic* or *constraint propagation*.
Besides the deterministic methods there are several asymptotically complete or incomplete methods in use, sometimes also referred to as heuristics, that are in most cases based upon *random search*, *tabu search* or biological resp. physical archetypes, such as melting iron for *simulated annealing*, behavior of insects for *ant colony optimization* and *particle swarm optimization* and of course evolutionary concepts for *genetic algorithms*. See Neumaier (2004) for an excellent overview of methods in global optimization and further references.
For optimal control problems such methods are rarely applied as the computational effort is prohibitive. Esposito & Floudas (2000) and Papamichail & Adjiman (2004) do present spatial Branch & Bound algorithms for dynamic systems. Their algorithms are based upon a series of upper bounds obtained from local solutions of the non–convex problem and lower bounds obtained by solution of a convexified problem. The convexification of occurring functions and in particular of the dynamic

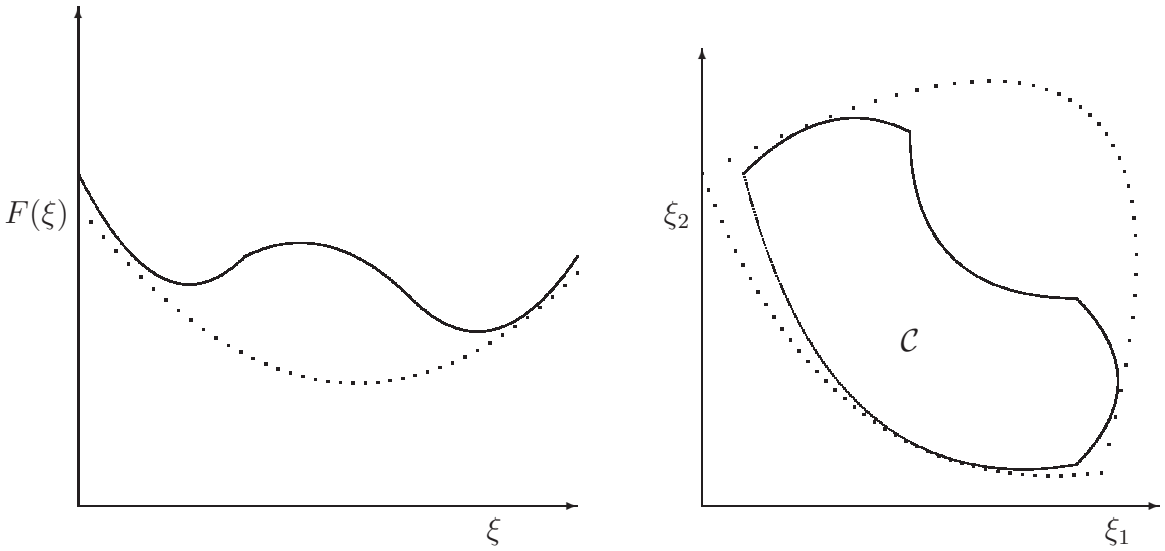

Figure 2.7: Convex underestimation (dotted) of an objective function $F(\xi)$ (solid) and convex overestimation (dotted) of a feasible region $\mathcal{C}$ (solid).

system is based upon convex envelopes, e.g., for bilinear functions first proposed by McCormick (1976). This approach seems to be computationally expensive, though, and not appropriate for the solution of real–life problems yet.
Besides theoretical results on globality issues a lot of experience has been gained with the solution of practical optimal control problems. If multiple local solutions exist, they are often due to (near–) symmetry as in robotics and mechanics. Anyway, in contrast to time–independent optimization problems, one has in most cases additional knowledge or at least an idea about the optimal behavior of the system (not necessarily about the controls and parameters, though). The direct multiple shooting method and collocation allow to exploit this additional information and incorporate this knowledge into the optimization problem, avoiding local minima far away from the optimal solution. This advantage of the simultaneous approaches proved to be very successful in the past for real–life applications.

2.5 Summary

In this chapter we investigated optimal control problems without binary variables to create a basis for methods and theory to be presented in later chapters. First we presented optimality conditions for optimal control problems, based on Pontryagin's maximum principle, and highlighted the solution structure and how it depends on switching functions. In this context we explained the differences between constraint–seeking and compromise–seeking arcs on the one hand and bang–bang and singular arcs on the other hand. In section 2.1.4 we stated the bang–bang principle which ensures that for linear systems, if a solution exists, there is always a bang–bang solution that is optimal.

Section 2.2 treats numerical solution methods. An overview is given about indirect and direct methods and they are presented with a short description of respective advantages and disadvantages. It becomes clear why direct multiple shooting is the most promising approach for the optimization of practical and generic mixed–integer optimal control problems. The most important concepts including sequential quadratic programming and the concept of internal numerical differentiation to obtain derivative information are presented.
Section 2.4 gives a brief overview of global optimization of optimal control problems and discusses the question under which assumptions these problems are convex.

Before the concepts of this section can be used in the progress of this work, we shall review mixed–integer techniques for nondynamic optimization problems first. This is the topic of the following chapter.

Chapter 3

Mixed–integer nonlinear programming

Finite–dimensional static optimization problems that involve continuous as well as integer variables are referred to as mixed–integer nonlinear programs (MINLPs). This problem class has received growing interest over the past twenty years. While enormous progress has been made in the field of mixed–integer linear programming, see, e.g., Johnson *et al.* (2000), Jünger & Reinelt (2004), Wolsey & Nemhauser (1999) or Bixby *et al.* (2004) for progress reports and further references, it turns out to be extremely challenging to bring together concepts from (linear) integer programming and nonlinear optimization. Pure integer optimization problems without continuous variables that consist of a convex quadratic function and linear constraints are a subclass of the problem class under consideration here. Such problems and therefore the general class of MINLPs were proven to be $\mathcal{NP}$–hard, Garey & Johnson (1979), Murty (1987), Vavasis (1995). This means from a theoretical point of view, if it is true that $\mathcal{NP} \neq \mathcal{P}$, then there are problem instances which are not solvable in polynomial time.
Several approaches have been proposed to solve MINLPs. The aim of this chapter is to give an overview of these methods and give indications where to find additional information. Excellent surveys are given by Grossmann & Kravanja (1997), Grossmann (2002), Bussieck & Prüssner (2003) and recently in Nowak (2005). Kallrath (2002) focuses on modeling issues in practice. The GAMSWORLD home page, Bussieck (2005), has a lot of further references, including a list of available MINLP solvers and benchmark problem collections.
The optimization problems we consider in this chapter are of the form

$$\begin{aligned} \min_{\boldsymbol{x},\boldsymbol{y}} \quad & F\left(\boldsymbol{x},\boldsymbol{y}\right) \\ \text{s.t.} \quad & \mathbf{0} \geq \boldsymbol{H}\left(\boldsymbol{x},\boldsymbol{y}\right), \\ & \boldsymbol{x} \in \boldsymbol{X}, \;\; \boldsymbol{y} \in \boldsymbol{Y}, \end{aligned} \tag{3.1}$$

with $F : \boldsymbol{X} \times \boldsymbol{Y} \mapsto \mathbb{R}$ and $\boldsymbol{H} : \boldsymbol{X} \times \boldsymbol{Y} \mapsto \mathbb{R}^{n_H}$ twice differentiable functions. $\boldsymbol{X}$ is a convex, compact set and $\boldsymbol{Y}$ corresponds to a polyhedral set of integer points, e.g., $\boldsymbol{Y} = \{0,1\}^{n_y}$. Please note that the meaning of the variables changes in this chapter in

comparison to the rest of the thesis and that we restrict our presentation on problems of type (3.1) and do not discuss logic–based problem formulations. For such an overview and further references we refer to Lee *et al.* (1999), Grossmann *et al.* (2005) or Oldenburg (2005). Equality constraints are not considered explicitely for the conceptual presentation of MINLP algorithms, as they can be formally transformed into inequality constraints.
We will need the notion of special ordered set restrictions in this thesis.

Definition 3.1 (Special ordered set property)
The variables $y_1, \ldots, y_{n_y}$ are said to fulfill special ordered set restrictions of type one (SOS1), if they fulfill the constraints

$$\sum_{i=1}^{n_y} y_i = 1, \tag{3.2a}$$

$$y_1, \ldots, y_{n_y} \in \{0, 1\}. \tag{3.2b}$$

If they fulfill

$$\sum_{i=1}^{n_y} y_i = 1, \tag{3.3a}$$

$$y_1, \ldots, y_{n_y} \in [0, 1], \tag{3.3b}$$

and at most two of the y_i are nonzero and if so, they are consecutive, then $\boldsymbol{y}$ is said to have the SOS type two property (SOS2).

SOS1 restrictions will be very important in this work, as they occur automatically after the convexifications of chapter 4. SOS2 restrictions typically occur when nonlinear functions are approximated by piecewise linear functions.

We will shortly describe MINLP methods in the literature, in particular reformulation techniques in section 3.1, Branch & Bound in section 3.2, Branch & Cut in section 3.3, Outer Approximation in section 3.4, Generalized Benders decomposition in section 3.5, Extended cutting planes in section 3.6, and LP/NLP based Branch & Bound in section 3.7. We mention extensions to treat nonconvex problems in section 3.8 and sum up in section 3.9.

3.1 Reformulations

The best way to avoid the complexity of integer variables is to avoid integer variables in the first place. For some problems it is indeed possible to replace integer variables by continuous variables and additional constraints. The first idea to replace an integer variable $y \in \{0, 1\}$ by a continuous variable $x \in [0, 1]$ is to add the constraint

$$x\,(1 - x) = 0 \tag{3.4}$$

to the problem formulation. Unfortunately this equality constraint is nonconvex with a disjoint feasible set and optimization solvers perform badly on such equations, as the constraint qualification (compare definition 2.15) is violated. There are a couple of approaches to enlargen the feasible set. A homotopy with a parameter $\beta \geq 0$ going towards zero and a constraint

$$x\,(1-x) \leq \beta \tag{3.5}$$

is a well known regularization technique, especially in the field of mathematical programs with equilibrium constraints (MPECs), which are optimization problems with complementarity conditions in the constraints, see Luo *et al.* (1996), Leyffer (2003), Raghunathan & Biegler (2003).
Continuous reformulations of discrete sets, e.g., of $(y_1, y_2) \in \{(0,1),(1,0)\}$, are typically motivated geometrically. Raghunathan & Biegler (2003) propose to use

$$\begin{aligned} y_1\, y_2 &= 0, && \text{(3.6a)}\\ y_1, y_2 &\geq 0, && \text{(3.6b)}\\ y_1 + y_2 &= 1, && \text{(3.6c)} \end{aligned}$$

which corresponds to the intersection of the line from $(0,1)$ to $(1,0)$ with the positive parts of the axes. (3.6a) can of course be regularized as above as $y_1\, y_2 \leq \beta$. Stein *et al.* (2004) propose to use a circle instead of the axes with the kink at the origin and replace (3.6a) with

$$\left(y_1 - \frac{1}{2}\right)^2 + \left(y_2 - \frac{1}{2}\right)^2 = \frac{1}{2},$$

respectively in a regularized form to replace (3.6) with

$$\begin{aligned} \left(y_1 - \frac{1}{2}\right)^2 + \left(y_2 - \frac{1}{2}\right)^2 &\leq \frac{1}{2}, && \text{(3.7a)}\\ \left(y_1 - \frac{1}{2}\right)^2 + \left(y_2 - \frac{1}{2}\right)^2 &\geq \left(\frac{1}{\sqrt{2}} - \beta\right)^2, && \text{(3.7b)}\\ y_1 + y_2 &= 1. && \text{(3.7c)} \end{aligned}$$

A third reformulation is based on the Fischer–Burmeister function

$$F^{\mathrm{FB}}(y_1, y_2) = y_1 + y_2 - \sqrt{y_1^2 + y_2^2} \tag{3.8}$$

which is zero when y_1, y_2 are binary, as this implies $y_1 = y_1^2$ and $y_2 = y_2^2$. Leyffer (2003) shows how to overcome the nondifferentiability of the Fischer–Burmeister function at the origin and successfully applies an SQP algorithm with only minor modifications to the solution of MPECs.

Another target for reformulations are the nonlinearities. The basic idea to use underestimating and overestimating linear functions is best exemplified by replacing a

bilinear term xy by a new variable z and additional constraints. This reformulation was proposed by McCormick (1976). For the new variable z we obtain the linear constraints

$$\begin{array}{rcccl} y^{\mathrm{lo}}x + x^{\mathrm{lo}}y - x^{\mathrm{lo}}y^{\mathrm{lo}} & \leq & z & \leq & y^{\mathrm{lo}}x + x^{\mathrm{up}}y - x^{\mathrm{up}}y^{\mathrm{lo}}, \\ y^{\mathrm{up}}x + x^{\mathrm{up}}y - x^{\mathrm{up}}y^{\mathrm{up}} & \leq & z & \leq & y^{\mathrm{up}}x + x^{\mathrm{lo}}y - x^{\mathrm{lo}}y^{\mathrm{up}}, \end{array} \tag{3.9}$$

for given bounds on x and y, i.e., $x \in [x^{\mathrm{lo}}, x^{\mathrm{up}}]$ and $y \in [y^{\mathrm{lo}}, y^{\mathrm{up}}]$. The inequalitites follow from $(x - x^{\mathrm{lo}})(y - y^{\mathrm{lo}}) \geq 0$ and three similar equations. The snag is of course that very tight bounds are needed for a successful optimization, which is not the case in the presence of strong nonlinearities. See Tawarmalani & Sahinidis (2002) or Floudas *et al.* (2005) for references on general under– resp. overestimation of functions.

3.2 Branch & Bound

Branch & Bound is a general framework that was developed to solve integer and combinatorial problems. The LP–based Branch & Bound algorithm for integer programming was developed in the sixties by Land & Doig (1960) and Dakin (1965). It was later on also applied to MINLPs, e.g., Gupta & Ravindran (1985), Leyffer (1993) or Borchers & Mitchell (1994), as well as to global optimization, see e.g., Tawarmalani & Sahinidis (2002) for an overview. We will assume in this section that F and $\boldsymbol{H}$ are convex functions. Extensions for the nonconvex case are discussed in section 3.8.
Branch & Bound performs a tree search in the space of the binary (integer) variables, with NLPs resp. LPs on every node of the search tree. The root consists of the original problem with all binary variables relaxed. All nodes in the tree are sons of this father node with additional inequalities. This principle is repeated in every subtree. The inequalities partition recursively the full integer problem into small subproblems, based on the fact that the minimum of all solutions of these subproblems is identical to the optimal solution of the full problem.

For the case of binary variables, we first solve the relaxed problem with $\boldsymbol{y} \in [0,1]^{n_y}$ and decide on which of the variables with non–integral value we shall *branch*, say y_i. Two new subproblems are then created with y_i fixed to 0 and 1, respectively. These new subproblems are added to a list and the father problem is removed from it. This procedure is repeated for all problems of the list until none is left. There are three exceptions to this rule, when a node is not branched on, but abandoned directly:

1. The relaxed solution is an integer solution. Then we have found a feasible solution of the MINLP resp. MILP and can compare the objective value with the current upper bound (and update it, if possible).

2. The problem is infeasible. Then all problems on the subtree will be infeasible, too.

3. The objective value is higher than the current upper bound. As it is a lower bound on the objective values of all problems on the subtree, they can be abandoned from the tree search.

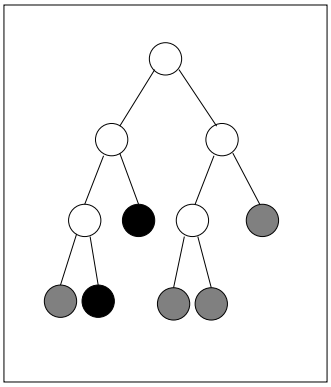

Figure 3.1: Branch and Bound concept. Nodes in the search tree are fathomed when they are infeasible or bounded from below (grey) or yield an integer solution (black). Otherwise they are split up in smaller subproblems (white).

Although it is in theory possible that all leaves of the tree have to be visited, which corresponds to a complete enumeration of all 2^{n_y} possible binary assignments, Branch & Bound does perform quite well for MILPs, Johnson *et al.* (2000), and also for MINLPs that comprise more costly subproblems in form of NLPs, Leyffer (2001). This performance depends crucially on several aspects.

- Good primal heuristics are important, as more subtrees can be fathomed right away with a good upper bound on the optimal value. A heuristics is everything that tries to come up with a feasible, probably suboptimal solution, e.g., by combinatorial or purely LP–based means. There are problem specific heuristics, e.g., for machine scheduling, Hall *et al.* (1997), or the Traveling Salesman Problem, the iterated Lin–Kerningham heuristics of Mak & Morton (1993), as well as heuristics for general integer programs. The Pivot–and–Complement heuristics from Balas & Martin (1980) for binary variables is based on the fact that nonbasis variables are fixed at their respective bounds. Pivoting slack variables into the basis, one obtains a solution with all variables fixed to either 0 or 1. This heuristics has later been extended to Pivot–and–Shift that can also treat mixed–integer problems. A completely different approach aims at an enumeration of 0-1 vectors in the neighborhood of the relaxed solution. Balas *et al.* (2001) propose an algorithm (OCTANE) that sorts all 0-1 vectors

in the order as they are intersected by a hyperplane, that is orthogonal to a search direction $\boldsymbol{d}$, when this hyperplane is moved from the solution $\boldsymbol{x}$ of the relaxed LP in direction $\boldsymbol{d}$. The main drawback of this approach seems to be the difficulty to find a search direction $\boldsymbol{d}$ with a high probability of yielding feasible solutions. An alternative approach that also enumerates 0-1 vectors is mentioned in Johnson *et al.* (2000). This approach uses the intersections of the rays of the cone generated by the LP basic solution with the hyperplanes of the unit cube to determine candidate solutions.

- Typically more than one variable will be fractional. Therefore a decision has to be made, which variable is chosen to branch on. This choice may depend on a priori given user preferences, on the value of the fractional variables, e.g., in *most–violation–branching*, or on the impact of a variable on the objective function. In *strong branching* the son problems for all open nodes are solved, before a decision is taken dependent on, e.g., the objective value of these problems. State–of–the–art Branch & Bound codes exploit structural information about the variables (such as special ordered set constraints) and tailored branching rules are applied. A survey of branching rules is presented by Linderoth & Savelsbergh (1999).

- Another important issue is the decision in which order the subproblems will be proceeded, with the extreme options *depth–first* search, i.e., the newly created subproblems are proceeded first with the hope of obtaining an upper bound as early as possible deep down in the search tree and the possibility of efficient warm starts, and *breadth–first* search, i.e., one of the subproblems on the highest level in the tree is proceeded first. Dür & Stix (2005) investigate different rules with respect to their performance from a stochastical point of view.

Branch & Bound techniques have been so successful in linear programming, as the relaxation from MILPs to LPs is extremely beneficial regarding computation times. LPs are in $\mathcal{P}$ (which was first shown by Khachiyan's ellipsoid method), whereas MILPs are $\mathcal{NP}$–complete. Furthermore LP–solvers are nowadays so competitive that the solution of underlying subproblems is not really an obstacle any more. For MINLPs and their relaxation to NLPs this is not true yet. On each node of the search tree a NLP has to be solved, which may be very costly. A more efficient way of integrating the Branch and Bound scheme and SQP has thus been proposed by Borchers & Mitchell (1994) and Leyffer (2001). Here branching is allowed after each iteration of the NLP solver. In this way, the nonlinear part of the MINLP problem is solved at the same time as the tree is being searched to solve the integer part. The problem that nodes cannot be fathomed any more, as no guaranteed lower bound is available when no convergence has been achieved is overcome by Borchers & Mitchell (1994) by solving dual problems. As this is computationally expensive, Leyffer (2001) proposes to include an additional inequality in the SQP procedure that cuts off all solutions that are worse than the current upper bound. The fathoming rule can then be replaced by a feasibility test of the nonlinear program. This nice concept helps to speed up the performance of the Branch & Bound method by a given factor,

approximately the quotient of the mean number of iterations needed for the solution of the nonlinear programs and the number of iterations after which a new branching is performed. This factor, approximately 2 to 5, is unfortunately fixed and does not help to move into a new dimension of solvable mixed–integer nonlinear programs.

As for linear programs, e.g., Wunderling (1996), primal–dual active set–based methods will typically outperform interior point methods due to their excellent restart possibilities after adding inequalities resp. variables. Note that the Branch & Bound method is computationally attractive in the MINLP context, whenever the integer part is more dominant than the nonlinear part, that is, the relaxed NLPs are not too costly to solve.

3.3 Branch & Cut

In the Branch & Bound scheme presented above binary variables have been fixed on each node. This fixation of variables can be seen as the introduction of an additional inequality

$$y_i \leq 0$$

respectively

$$y_i \geq 1$$

for the relaxed variable $y_i \in [0, 1]$. For general integer programs these equations read as

$$y_i \leq \lfloor y_i \rfloor$$

respectively

$$y_i \geq \lceil y_i \rceil,$$

where $\lfloor y_i \rfloor$ means rounding down and $\lceil y_i \rceil$ rounding up the value of y_i to the next integer value. These inequalities *cut* off a part of the feasible domain of the problem relaxation that does not contain any integer solution. These inequalities are only locally valid, i.e., they hold in a certain part of the search tree. This concept of adding cutting planes can be extended to more general inequalities that are globally valid.

Let us first consider the linear case. The main idea of a Branch & Cut algorithm is based on the fact that the optimal solution of a linear optimization problem lies on a vertex of the feasible region. If we had a description of the convex hull of the integer solutions, its solution would be on a vertex and therefore integer and the optimal solution to the original problem. The Branch & Cut algorithm aims at adding more and more cuts to a basic problem, until the convex hull is approximated at least

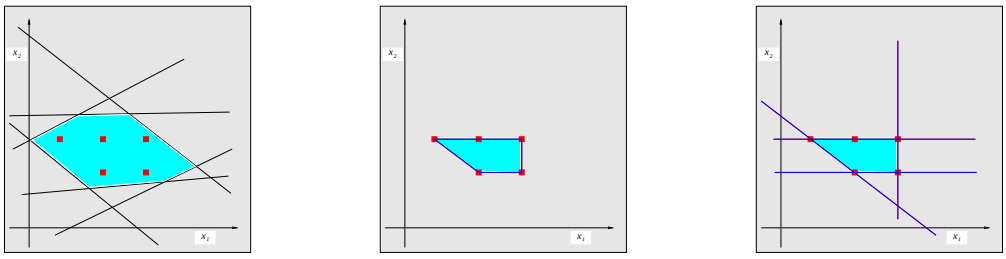

Figure 3.2: Feasible set of a LP relaxation and included integer points (left). Convex hull of these integer points (middle) and corresponding cuts (right).

locally well enough such that the optimal solution of the relaxed problem is integer. This concept is shown in figure 3.2.
Much effort in integer programming has been spent on the question how to efficiently determine strong cuts that cut off as much as possible from the relaxed feasible set. The task to determine such a constraint that is fulfilled by all feasible integer solutions, but not for all points of the relaxed set is referred to as *separation problem*, compare Grötschel *et al.* (1988).
According to Johnson *et al.* (2000) there are three different types of valid inequalities at a very high level that can be used to achieve integrality.

- Type I – No structure: these inequalities are based only on variables being integral or binary. Therefore they can always be used to separate a fractional point. The earliest and best known class of Type I inequalities are the Gomory–Chvatal cuts introduced by Gomory (1958) and Chvatal (1973). For integer variables $y_i \geq 0$ and an inequality

$$\sum_{i=1}^{n_y} a_{ji}\, y_i \quad \leq \quad b_j$$

  a Gomory–Chvatal cut is given by

$$\sum_{i=1}^{n_y} \lfloor a_{ji} \rfloor\, y_i \quad \leq \quad \lfloor b_j \rfloor.$$

  This concept can be extended to include continuous variables as well. Another type of type I inequalies are lift–and–project inequalities. An LP has to be solved for every inequality, which may turn out to be expensive. However, given any fractional solution, a violated inequality can always be found. See, e.g., Balas *et al.* (1993), Körkel (1995) or Balas & Perregaard (1999) for details.

- Type II – Relaxed structure: these inequalities are derived from relaxations of the problem, for example by considering a single row of the constraint set. Therefore, they can at best only separate fractional points that are infeasible to the convex hull of the relaxation. However, these inequalities are usually

facets of the convex hull of the relaxation and may therefore be stronger than type I inequalities.

- Type III – Problem specific structure: these inequalities are typically derived from the full problem structure, or a substantial part of it. They are usually very strong in that they may come from known classes of facets of the convex hull of feasible solutions (compare rightmost illustration in figure 3.2). Their application is limited to the particular problem class and the known classes of inequalities for that problem class.

Remark 3.2 *Another concept that is very important in the solution of large–scale MILPs is Branch & Price resp. Branch, Cut & Price. If a large number of integer variables, say millions, is involved, it is beneficial to work on a small subset of these variables only and add resp. remove dynamically variables to this subset. Recently, Hoai* et al. *(2005) could solve an airline scheduling problem with a large number of binary variables with such an approach. For MINLPs this approach seems to be somewhat premature as the number of binary variables is in the range of tens or hundreds instead of millions, compare the reference problems in Bussieck (2005). Still, a Branch, Cut & Price algorithm for MINLPs can already be found in Nowak (2005).*

Branch & Cut techniques are well developped for MILPs. The work to transfer the methodology to the nonlinear case has just started, though. This is not straightforward, as the separation problems that have to be solved to determine cutting planes may be nonconvex and therefore hard to solve. Stubbs & Mehrotra (1999), Stubbs & Mehrotra (2002) use nonlinear cuts for convex problems of the form (3.1). Iyengar (2001) proposes quadratic cuts for mixed 0-1 quadratic programs. A Branch-Cut-and-Price algorithm that is based on Lagrangian cuts, a decomposition of the MINLP and MINLP heuristics is presented in Nowak (2005), together with a framework that can also handle nonconvex problems by polynomial underestimation of the nonconvex functions.

3.4 Outer approximation

While Branch & Bound and Branch & Cut methods were first developped for MILPs and then transferred to the solution of MINLPs, the outer approximation algorithm of Duran & Grossmann (1986) was explicitely developped for MINLPs. Again we assume that the functions F and $\boldsymbol{H}$ are convex and treat extensions later.
Outer approximation is motivated by the idea to avoid a huge number of NLPs, that may be very costly to solve, and instead to use available well–advanced MILP solvers. It is based on a decoupling of the integer and the nonlinear part, by an alternating sequence of MILPs and NLPs. The solution of the linear integer problem yields an integer assignment and a lower bound to the convex MINLP. The solution of the NLP, with the integer variables fixed to the result of the last MILP, gives (maybe) a feasible solution and therefore an upper bound and a point around which to linearize

and modify the MILP. This iteration is persued until the lower bound is close enough to the upper bound or infeasibility is detected.
Let k denote the index of an outer iteration. The NLP mentioned above is derived from (3.1) by fixing the binary variables $\boldsymbol{y}$ to $\boldsymbol{y^k}$:

$$\begin{array}{ll} \min\limits_{\boldsymbol{x}} & F\left(\boldsymbol{x}, \boldsymbol{y^k}\right) \\ \text{s.t.} & \boldsymbol{0} \geq \boldsymbol{H}\left(\boldsymbol{x}, \boldsymbol{y^k}\right), \\ & \boldsymbol{x} \in \boldsymbol{X}. \end{array} \tag{3.10}$$

The MILP is referred to as *master problem* and given by

$$\begin{array}{lll} \min\limits_{\boldsymbol{x},\boldsymbol{y},\alpha} & \alpha & \\ \text{s.t.} & \alpha \geq F(\boldsymbol{x^k}, \boldsymbol{y^k}) + \nabla F(\boldsymbol{x^k}, \boldsymbol{y^k})^T \begin{pmatrix} \boldsymbol{x} - \boldsymbol{x^k} \\ \boldsymbol{y} - \boldsymbol{y^k} \end{pmatrix} & \forall\, (\boldsymbol{x^k}, \boldsymbol{y^k}) \in K, \\ & \boldsymbol{0} \geq \boldsymbol{H}\left(\boldsymbol{x^k}, \boldsymbol{y^k}\right) + \nabla \boldsymbol{H}\left(\boldsymbol{x^k}, \boldsymbol{y^k}\right)^T \begin{pmatrix} \boldsymbol{x} - \boldsymbol{x^k} \\ \boldsymbol{y} - \boldsymbol{y^k} \end{pmatrix} & \forall\, (\boldsymbol{x^k}, \boldsymbol{y^k}) \in K, \\ & \boldsymbol{x} \in \boldsymbol{X},\ \ \boldsymbol{y} \in \boldsymbol{Y}. & \end{array} \tag{3.11}$$

The set K contains a certain number of points $(\boldsymbol{x^k}, \boldsymbol{y^k})$ that are used for linear inequalities. In the outer approximation algorithm they are chosen as solution points of the convex NLPs (3.10) and lie therefore on the boundary of these problems – by adding a linear constraint at such points the feasible convex set is approximated from the outside, motivating the name of the algorithm. Duran & Grossmann (1986) and Fletcher & Leyffer (1994) showed that, assumed the set K contains all feasible solutions $(\boldsymbol{x^k}, \boldsymbol{y^k})$ of (3.10) (resp. the solution of a proposed feasibility problem in case the NLP has no solution), the solutions of the MILP (3.11) and the MINLP (3.1) are identical.

Motivated by this theorem one iterates between the two subproblems, adding linearization points to K until the lower bound provided by the MILPs reaches the upper bound provided by the NLPs. Note that we obtain a series of nondecreasing optimal objective function values of the MILPs that are all lower bounds, no matter how many inequalities are added, compare figure 3.3.
To avoid to get stuck, additional care has to be taken to make the integer assignments $\boldsymbol{y^k}$ that were already chosen infeasible. This guarantees the finiteness of the outer approximation algorithm, if $\boldsymbol{Y}$ is finite, and can be achieved by the cuts

$$\sum_{i \in B^k} y_i - \sum_{i \in N^k} y_i \leq |B^k| - 1, \ \ \forall\, (\boldsymbol{x^k}, \boldsymbol{y^k}) \in K \tag{3.12}$$

with index sets

$$\begin{aligned} B^k &= \left\{ i \mid y_i^k = 1 \right\}, \\ N^k &= \left\{ i \mid y_i^k = 0 \right\}. \end{aligned}$$

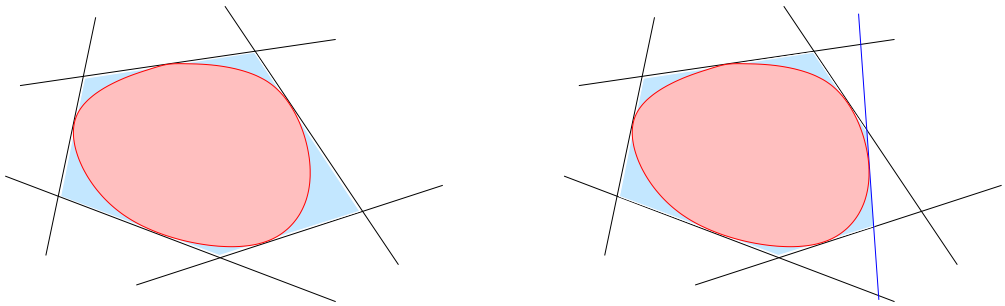

Figure 3.3: Adding a cut at one linearization point reduces the feasible set of the linear program (3.11), but does not cut off any feasible points of (3.1).

These are known to be weak cuts, but achieve exactly what one wants: all $(\cdot, \boldsymbol{y^k}) \in K$ are infeasible while all other binary assignments of $\boldsymbol{y}$ fulfill (3.12).
The outer approximation algorithm has been extended to disjunctive programming and applied to mixed–integer optimal control problems with time–independent binary variables, see Grossmann (2002), Oldenburg (2005) for further references.

3.5 Generalized Benders decomposition

Generalized Benders decomposition was proposed by Geoffrion (1972) and is therefore older than the outer approximation algorithm. Nevertheless we presented outer approximation first, as Generalized Benders decomposition is more conveniently derived starting from the outer approximation master problem (3.11)[1].
For convenience we write $F^k = F(\boldsymbol{x^k}, \boldsymbol{y^k})$ and $\boldsymbol{H^k} = \boldsymbol{H}(\boldsymbol{x^k}, \boldsymbol{y^k})$. As every single $(\boldsymbol{x^k}, \boldsymbol{y^k}) \in K$ is the optimal solution of an NLP, the Karush–Kuhn–Tucker conditions (2.43) are fulfilled with a Lagrange multiplier $\boldsymbol{\mu^k} \geq 0$ and it holds

$$\nabla_x F^k + \nabla_x \boldsymbol{H^k} \boldsymbol{\mu^k} = \boldsymbol{0}. \tag{3.13}$$

If, for every $(\boldsymbol{x^k}, \boldsymbol{y^k}) \in K$, we multiply the n_H lower inequalities in (3.11) with the Lagrange multiplier vector $\boldsymbol{\mu^k} \geq 0$ and add the first inequality of (3.11), we obtain the inequality

$$\alpha \geq F^k + \boldsymbol{\mu^k}^T \boldsymbol{H^k} + (\nabla F^k + \nabla \boldsymbol{H^k} \boldsymbol{\mu^k})^T \begin{pmatrix} \boldsymbol{x} - \boldsymbol{x^k} \\ \boldsymbol{y} - \boldsymbol{y^k} \end{pmatrix} \tag{3.14}$$

for all $(\boldsymbol{x^k}, \boldsymbol{y^k}) \in K$. By applying (3.13) we can eliminate the variables $\boldsymbol{x}$ completely and obtain

$$\alpha \geq F^k + \boldsymbol{\mu^k}^T \boldsymbol{H^k} + (\nabla_y F^k + \nabla_y \boldsymbol{H^k} \boldsymbol{\mu^k})^T (y - \boldsymbol{y^k}). \tag{3.15}$$

[1] this derivation is based on a personal communication of Sven Leyffer

Remembering the definition of the Lagrangian on page 42,

$$\mathcal{L}^{\mathbf{k}} \;=\; F^k + \boldsymbol{\mu}^{\boldsymbol{k}^T}\boldsymbol{H}^{\boldsymbol{k}},$$

and noting that the expression

$$\nabla_y F^k + \nabla_y \boldsymbol{H}^{\boldsymbol{k}}\boldsymbol{\mu}^{\boldsymbol{k}}$$

is nothing but the multipliers $\boldsymbol{\lambda}^{\boldsymbol{k}}$ of the condition $\boldsymbol{y} = \boldsymbol{y}^{\boldsymbol{k}}$ in NLP (3.10), we write (3.15) as

$$\alpha \geq \mathcal{L}^{\mathbf{k}} + \boldsymbol{\lambda}^{\boldsymbol{k}^{\mathbf{T}}}(\boldsymbol{y} - \boldsymbol{y}^{\boldsymbol{k}}),$$

for all $(\boldsymbol{x}^{\boldsymbol{k}}, \boldsymbol{y}^{\boldsymbol{k}}) \in K$. Benders master problem is therefore given by

$$\begin{array}{lll} \min\limits_{\boldsymbol{y},\alpha} & \alpha & \\ \text{s.t.} & \alpha \;\geq\; \mathcal{L}^{\mathbf{k}} + \boldsymbol{\lambda}^{\boldsymbol{k}^{\mathbf{T}}}(\boldsymbol{y} - \boldsymbol{y}^{\boldsymbol{k}}) & \forall\, (\boldsymbol{x}^{\boldsymbol{k}}, \boldsymbol{y}^{\boldsymbol{k}}) \in K, \\ & \boldsymbol{y} \in \boldsymbol{Y}. & \end{array} \tag{3.16}$$

Generalized Benders decomposition is identical to outer approximation with the only difference that problem (3.16) instead of (3.11) is used as a master program. This makes it fairly easy to compare the two methods. Obviously, (3.16) has less constraints and variables, as the continuous variables $\boldsymbol{x}$ have been eliminated. It is almost a pure integer program with α as the only continuous variable. Therefore (3.16) can typically be solved much easier. But, as is clear from the above derivation, where inequalities were summed up, the formulation is weaker than (3.11). This is also the reason why Generalized Benders decomposition is hardly used anymore today and outer approximation is more popular.

3.6 Extended cutting planes

The Extended cutting planes method does not solve any nonlinear programs. It is an extension of Kelley's cutting plane method, Kelley (1960), and only iterates on mixed–integer linear programs, completely ignoring the nonlinear part. The main idea is to linearize the original function around the solution of the linear master problem and to add the most violated constraint to it. The extended cutting plane method is described in detail in Westerlund & Pettersson (1995). It has also been extended to pseudo–convex functions. As both the nonlinear as the integer part are solved in the same MILP of type (3.11), the method may need a large number of iterations and typically shows slow nonlinear convergence.

3.7 LP/NLP based Branch & Bound

Quesada & Grossmann proposed a new method to solve MINLPs in 1992 with the catchy title LP/NLP based Branch & Bound. The main idea of this approach is to

reduce the number of MILPs that have to be solved in the outer approximation algorithm. Instead, only one single master program of type (3.11) is solved by a Branch & Bound approach. Every time a new $\boldsymbol{y^k}$ is found in this procedure, compare the first exception on page 55, the Branch & Bound algorithm is stopped. The variable $\boldsymbol{y^k}$ is fixed and a NLP of type (3.10) is solved, just as in the outer approximation algorithm. Also, we add the solution $(\boldsymbol{x^k}, \boldsymbol{y^k})$ to the set K and linearizations around this point to the master program (3.11). These linearizations are included dynamically into the stopped Branch & Bound method by updating all open nodes and the tree search is then continued, avoiding the need to restart.

The advantage of the LP/NLP based Branch & Bound compared to outer approximation is considerable. Experience shows that about the same number of NLPs has to be solved, but only one MILP. Leyffer (1993) reports substantial savings with this method. The drawback of the method seems to be on the practical side. One needs access to a state–of–the–art MILP solver to realize the necessary changes, i.e., the stop of the algorithm and the modification of the underlying constraints. The fact that this is not possible for the best (commercial) MILP solvers as CPLEX or XPRESS, leads to the fact that no fair comparison has been made and no commercial implementation of the method exists. A recent implementation is available in the `COIN-OR` environment and gives hope to further developments in the near future, see Bonami *et al.* (2005).

The general idea can of course be transfered to General Benders decomposition and cutting plane methods. Akrotirianakis *et al.* (2001) propose to use Gomory–Chvatal cuts in the tree search and report a speedup of approximately 3 when compared to a standard Branch & Bound.

3.8 Nonconvex problems

We assumed so far that the functions $F(\cdot)$ and $\boldsymbol{H}(\cdot)$ are convex or even linear. If they are nonconvex, local minima are not necessarily global minima any more and the issue of global optimization arises again that was already addressed in section 2.4. There are two main problems, when nonconvexities occur:

- The solution of an NLP may have several local minima.
- Linearizations do not yield a valid lower bound any more, as parts of the feasible set may be cut off. Compare the left picture in figure 3.4.

To overcome this problem, methods of global optimization have to be used that were already mentioned in section 2.4, most importantly the concepts of underestimation resp. overestimation of the nonconvex functions, compare figure 2.7, and application of spatial Branch & Bound, going back to Falk & Soland (1969). See Neumaier (2004), Tawarmalani & Sahinidis (2002), Floudas *et al.* (2005) or Nowak (2005) for

more information.

For Branch & Bound and outer approximation some heuristics were proposed, too. As the nodes cannot be fathomed any more without risking to neglect feasible, better solutions, a Branch & Bound approach should not be applied to nonconvex optimization problems. A heuristics to overcome this problem is proposed in Leyffer (2001).

For outer approximation an augmented penalty approach has been proposed by Viswanathan & Grossmann (1990). This approach is based on including a *security distance* by slack variables $\boldsymbol{\beta}$. The master program (3.11) is modified to

$$\begin{array}{lll} \min\limits_{\boldsymbol{x},\boldsymbol{y},\alpha,\boldsymbol{\beta}} & \alpha + \sum\limits_{i=1}^{n_H} w_i \beta_i & \\ \text{s.t.} & \alpha \geq F(\boldsymbol{x^k},\boldsymbol{y^k}) + \nabla F(\boldsymbol{x^k},\boldsymbol{y^k})^T \begin{pmatrix} \boldsymbol{x}-\boldsymbol{x^k} \\ \boldsymbol{y}-\boldsymbol{y^k} \end{pmatrix} & \forall\, (\boldsymbol{x^k},\boldsymbol{y^k}) \in K, \\ & \boldsymbol{\beta} \geq \boldsymbol{H}\,(\boldsymbol{x^k},\boldsymbol{y^k}) + \nabla \boldsymbol{H}\,(\boldsymbol{x^k},\boldsymbol{y^k})^T \begin{pmatrix} \boldsymbol{x}-\boldsymbol{x^k} \\ \boldsymbol{y}-\boldsymbol{y^k} \end{pmatrix} & \forall\, (\boldsymbol{x^k},\boldsymbol{y^k}) \in K, \\ & |B^k| - 1 \geq \sum\limits_{i\in B^k} y_i - \sum\limits_{i\in N^k} y_i, & \forall\, (\boldsymbol{x^k},\boldsymbol{y^k}) \in K, \\ & \boldsymbol{x} \in \boldsymbol{X},\ \ \boldsymbol{y} \in \boldsymbol{Y},\ \ \boldsymbol{\beta} \geq \boldsymbol{0}. & \end{array} \tag{3.17}$$

with penalty parameters w_i sufficiently large. This concept is illustrated in figure 3.4.

Figure 3.4: Outer approximation augmented penalty method. While the outer approximation method may cut off parts of the feasible set (left), introduction of slack variables $\boldsymbol{\beta}$ may avoid this effect (right).

Other concepts that are of interest include quadratic master problems that use second order information, Fletcher & Leyffer (1994), and an efficient handling of constraints, Grossmann (2002), in outer approximation.

3.9 Summary

In this chapter we gave an overview of methods to solve mixed–integer nonlinear programs. We started by presenting reformulation techniques and the general frameworks of Branch & Bound respectively Branch & Cut.

We introduced outer approximation, Generalized Benders decomposition and extended cutting planes, that have a strong focus on the integer part of MINLPs and perform typically well when the functions do not show strong nonlinear behavior, but the NLPs are expensive to solve. To deal with nonlinear effects, one might use quadratic master problems (MIQPs) that use second order information, Fletcher & Leyffer (1994).

If the NLPs are not too costly to solve and the bottleneck is the MILP, a Branch & Bound approach will probably be the better choice, Fletcher & Leyffer (1994). To reduce the high costs of solving NLPs on every node of the search tree, an algorithm to integrate SQP and Branch & Bound should be applied.

See Skrifvars *et al.* (1998) for a performance comparison of these algorithms when applied to model structure determination and parameter estimation problems.

The most promising way seems to be a consequent integration of nonlinear and integer parts, either by the presented LP/NLP based Branch & Bound algorithm or by a combination of global optimization and Branch, Cut & Price methods with a set of rounding, combinatorial or dual heuristics, as proposed in Nowak (2005).

Chapter 4

Binary control functions

The algorithms of chapter 3 are based upon a fixed discretization of the controls, yielding a constant number of variables. This is typically not sufficient to determine an optimal solution, if the switching points are free. Before we come to methods that take this into account, we will investigate some properties that motivate the algorithms of chapter 5.

For our approach it is crucial to obtain lower and upper bounds on the objective value to judge the quality of an obtained solution. While upper bounds are obtained from any binary feasible solution, it is not as straightforward to get a good estimate for the obtainable objective value. As illustrated by a one–dimensional example in section 2.1.4, the reachable sets of generic measurable control functions $\boldsymbol{u} \in \mathcal{U}_m$ and bang–bang functions $\boldsymbol{u} \in \mathcal{U}_{BB}$ not necessarily coincide for nonlinear optimal control problems. Therefore it is not correct to assume that the objective value obtained by a relaxed control can also be obtained by a bang–bang control. As this is different for convex problems, we will *convexify* the optimal control problem and use the obtained solution as a *reachable* lower bound for the *nonlinear binary* problem. The different optimal control problems considered in this chapter are described in section 4.1 and bounds are derived in section 4.2.

To obtain a bang–bang solution we will also use penalty terms. In section 4.3 aspects concerning such penalizations are highlighted. Extensions to the simplified control problems in the first sections will be discussed in section 4.4 with a special focus on constraint and path constraints.

4.1 Convexification

As formulated in definitions 1.11 and 1.14 in chapter 1, our goal is to find an optimal solution of general MIOCPs. Here we will first consider a less general problem to state our ideas. In particular we restrict our investigations to singlestage problems without constraints, algebraic variables and time–independent binary variables $\boldsymbol{v}$ to avoid an unnecessary complication of the notation.

Definition 4.1 (Nonlinear problem in binary and relaxed form)
Problem (BN) is given by

$$\min_{\boldsymbol{x},\boldsymbol{w},\boldsymbol{u},\boldsymbol{p}} \Phi[\boldsymbol{x},\boldsymbol{w},\boldsymbol{u},\boldsymbol{p}] \tag{4.1a}$$

subject to the ODE system

$$\dot{\boldsymbol{x}}(t) \;=\; \boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{w}(t),\boldsymbol{u}(t),\boldsymbol{p}), \quad t\in[t_0,t_f], \tag{4.1b}$$

with initial values

$$\boldsymbol{x}(t_0) \;=\; \boldsymbol{x_0}, \tag{4.1c}$$

and binary admissibility of $\boldsymbol{w}(\cdot)$*,*

$$\boldsymbol{w}(\cdot) \;\in\; \Omega(\Psi), \tag{4.1d}$$

with $\Psi=\Psi_{\text{free}}$*. We write* Φ^{BN} *for the objective value obtained by an admissible solution. The relaxed problem obtained by replacing constraint (4.1d) with (1.15a) will be denoted as problem (RN) with corresponding optimal objective value* Φ^{RN}*.*

We will then need a convexification with respect to the binary control functions $\boldsymbol{w}(\cdot)$. Again we consider both, the binary feasible and the relaxed case.

Definition 4.2 (Convexified linear problem in binary and relaxed form)
Problem (BL) is given by

$$\min_{\boldsymbol{x},\tilde{\boldsymbol{w}},\boldsymbol{u},\boldsymbol{p}} \sum_{i=1}^{2^{n_w}} \Phi[\boldsymbol{x},\boldsymbol{w^i},\boldsymbol{u},\boldsymbol{p}]\;\tilde{w}_i(\cdot), \tag{4.2a}$$

subject to the ODE system

$$\dot{\boldsymbol{x}}(t) \;=\; \sum_{i=1}^{2^{n_w}} \boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{w^i},\boldsymbol{u}(t),\boldsymbol{p})\;\tilde{w}_i(t), \quad t\in[t_0,t_f], \tag{4.2b}$$

with initial values

$$\boldsymbol{x}(t_0) \;=\; \boldsymbol{x_0}, \tag{4.2c}$$

binary admissibility of the new control function vector $\tilde{\boldsymbol{w}}=(\tilde{w}_1,\ldots,\tilde{w}_{2^{n_w}})^T$*,*

$$\tilde{\boldsymbol{w}}(\cdot) \;\in\; \Omega(\Psi), \tag{4.2d}$$

again with $\Psi=\Psi_{\text{free}}$*, and the special ordered set property*

$$\sum_{i=1}^{2^{n_w}} \tilde{w}_i(t) \;=\; 1, \quad t\in[t_0,t_f]. \tag{4.2e}$$

The vectors $\boldsymbol{w^i}\in\mathbb{R}^{n_w}$ *are fixed and enumerate all possible binary assignments of* $\boldsymbol{w}$*,* $i=1\ldots 2^{n_w}$*. We write* Φ^{BL} *for the objective value obtained by an admissible solution. The relaxed problem with constraint (1.15a) instead of (4.2d) will be denoted as problem (RL) with corresponding optimal objective value* Φ^{RL}*.*

Remark 4.3 *Equality constraint (4.2e) allows the direct elimination of one control function, e.g., of $\tilde{w}_{2^{n_w}}(\cdot)$. For $t \in [t_0, t_f]$ the term*

$$\boldsymbol{f}\left(\boldsymbol{x}(t), \boldsymbol{w}^{\boldsymbol{2^{n_w}}}, \boldsymbol{u}(t), \boldsymbol{p}\right)\ \tilde{w}_{2^{n_w}}(t), \tag{4.3}$$

can be replaced by

$$\boldsymbol{f}\left(\boldsymbol{x}(t), \boldsymbol{w}^{\boldsymbol{2^{n_w}}}, \boldsymbol{u}(t), \boldsymbol{p}\right)\left(1 - \sum_{i=1}^{2^{n_w}-1} \tilde{w}_i(t)\right), \tag{4.4}$$

if equality constraint (4.2e) is replaced by

$$\sum_{i=1}^{2^{n_w}-1} \tilde{w}_i(t) \ \leq\ 1. \tag{4.5}$$

The same holds for the term related to $\tilde{w}_{2^{n_w}}(\cdot)$ in the objective functional.

Remark 4.4 *The convexification yields an exponentially growing number of control functions with respect to n_w. The number of control functions of the convexified problem is $n_{\tilde{w}} = 2^{n_w} - 1$. The convexification approach in connection with a direct method to solve the relaxed problem (RL) is therefore not suited for problems with a high number of nonconvex binary control functions, say $n_w > 8$.*

Remark 4.5 *For one–dimensional binary controls $w(\cdot)$ with $n_w = 1$, the convexification yields no increase in the number of binary control variables, $n_w = n_{\tilde{w}} = 2^{n_w} - 1$. If furthermore the right hand side and the objective functional are affine in the control $w(\cdot)$, then problems (BN) and (BL) resp. (RN) and (RL) are identical. In other words: a one–dimensional, control–affine optimal control problem is already in the convex form (4.2).*

The connection between the problem classes introduced above for general nonlinear and multidimensional binary control functions will be the topic of the next section.

4.2 Bounds

We defined four problem classes in the preceding section, namely binary and relaxed optimal control problems that are either nonlinear or linear in the control functions $\boldsymbol{w}$ resp. $\tilde{\boldsymbol{w}}$. We will now investigate how optimal objective values correlate to each other, assuming an optimal solution exists.

Theorem 4.6 (Comparison of binary solutions)
If problem (BL) has an optimal solution $(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ with objective value Φ^{BL}, then there exists an n_w–dimensional control function $\boldsymbol{w}^*$ such that $(\boldsymbol{x}^*, \boldsymbol{w}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ is an optimal solution of problem (BN) with objective value Φ^{BN} and

$$\Phi^{\mathrm{BL}} = \Phi^{\mathrm{BN}}.$$

The converse holds as well.

Proof. Assume $(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ is a minimizer of (BL). As it is feasible, we have the special ordered set property (4.2e) and with $\tilde{w}_i^*(\cdot) \in \{0,1\}$ for all $i = 1 \dots 2^{n_w}$ it follows that there exists one index $1 \leq j(t) \leq 2^{n_w}$ for all $t \in [t_0, t_f]$ such that $\tilde{w}_{j(t)}^* = 1$ and $\tilde{w}_i^* = 0$ for all $i \neq j(t)$.
The binary control function

$$\boldsymbol{w}^*(t) \;:=\; \boldsymbol{w}^{\boldsymbol{j(t)}}, \quad t \in [t_0, t_f]$$

is therefore well–defined and yields for fixed $(\boldsymbol{x}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ an identical right hand side function value

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}^*(t), \boldsymbol{w}^*(t), \boldsymbol{u}^*(t), \boldsymbol{p}^*) &= \boldsymbol{f}(\boldsymbol{x}^*(t), \boldsymbol{w}^{\boldsymbol{j(t)}}, \boldsymbol{u}^*(t), \boldsymbol{p}^*) \\ &= \sum_{i=1}^{2^{n_w}} \boldsymbol{f}(\boldsymbol{x}^*(t), \boldsymbol{w}^{\boldsymbol{i}}, \boldsymbol{u}^*(t), \boldsymbol{p}^*)\; \tilde{w}_i^*(t) \end{aligned}$$

and an identical objective function

$$\begin{aligned} \Phi(\boldsymbol{x}^*(t), \boldsymbol{w}^*(t), \boldsymbol{u}^*(t), \boldsymbol{p}^*) &= \Phi(\boldsymbol{x}^*(t), \boldsymbol{w}^{\boldsymbol{j(t)}}, \boldsymbol{u}^*(t), \boldsymbol{p}^*) \\ &= \sum_{i=1}^{2^{n_w}} \Phi(\boldsymbol{x}^*(t), \boldsymbol{w}^{\boldsymbol{i}}, \boldsymbol{u}^*(t), \boldsymbol{p}^*)\; \tilde{w}_i^*(t) \end{aligned}$$

compared to the feasible and optimal solution $(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ of (BL). Thus the vector $(\boldsymbol{x}^*, \boldsymbol{w}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ is a feasible solution of problem (BN) with objective value Φ^{BL}. Now assume there was a feasible solution $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{w}}, \hat{\boldsymbol{u}}, \hat{\boldsymbol{p}})$ of (BN) with objective value $\hat{\Phi}^{\mathrm{BN}} < \Phi^{\mathrm{BL}}$. As the set $\{\boldsymbol{w}^{\boldsymbol{1}}, \dots, \boldsymbol{w}^{\boldsymbol{2^{n_w}}}\}$ contains all feasible assignments of $\hat{\boldsymbol{w}}$, one has again an index function $\hat{j}(\cdot)$ such that $\hat{\boldsymbol{w}}$ can be written as

$$\hat{\boldsymbol{w}}(t) \;:=\; \boldsymbol{w}^{\hat{\boldsymbol{j}}\boldsymbol{(t)}}, \quad t \in [t_0, t_f].$$

With the same argument as above $\tilde{\boldsymbol{w}}$ defined as

$$\tilde{w}_i(t) = \begin{cases} 1 & i = \hat{j}(t) \\ 0 & \text{else} \end{cases} \qquad i = 1, \dots, 2^{n_w}, \quad t \in [t_0, t_f],$$

is feasible for (BL) with objective value $\hat{\Phi}^{\mathrm{BN}} < \Phi^{\mathrm{BL}}$ which contradicts the optimality assumption. Thus $(\boldsymbol{x}^*, \boldsymbol{w}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ is an optimal solution of problem (BN).
The converse of the statement is proven with the same argumentation starting from an optimal solution of (BN). ■

Theorem 4.6 holds only for controls $\tilde{w}_i(t) \in \{0,1\}$, not for the relaxed problems (RN) and (RL) with $\tilde{w}_i(t) \in [0,1]$. This can be seen in a simple one–dimensional example. Consider the (BN) problem

$$\min_{x,w} -x(t_f) \tag{4.6a}$$

subject to the ODE

$$\dot{x}(t) \;=\; \frac{1}{2} - 4\left(w(t) - \frac{1}{2}\right)^2, \quad t \in [t_0, t_f], \tag{4.6b}$$

with given initial value x_0 and time horizon $[t_0, t_f]$, where $w(\cdot)$ is restricted to values in $\{0, 1\}$. Binary feasible assignments are therefore $w^0 = 0$ and $w^1 = 1$, yielding the convexified problem (BL)

$$\min_{x,\tilde{w}_1,\tilde{w}_2} -x(t_f) \tag{4.7a}$$

subject to the ODE

$$\dot{x}(t) \;=\; -\frac{1}{2}\tilde{w}_1(t) - \frac{1}{2}\tilde{w}_2(t), \quad t \in [t_0, t_f], \tag{4.7b}$$

with given initial value x_0 and time horizon $[t_0, t_f]$ and

$$\tilde{w}_1(t) + \tilde{w}_2(t) = 1. \tag{4.7c}$$

$\tilde{w}_1(t)$ and $\tilde{w}_2(t)$ are restricted to values in $\{0, 1\}$.
Clearly, for both (BN) and (BL) $\dot{x}(t) = -\frac{1}{2}$ holds for all binary feasible choices of $w(t)$ resp. $\tilde{\boldsymbol{w}}(t)$. Objective function as well as right hand side of the ODE coincide as stated by theorem 4.6. For relaxed $w(t) \in [0, 1]$ one has $\dot{x}(t) \in [-\frac{1}{2}, \frac{1}{2}]$, while a relaxation of $\tilde{\boldsymbol{w}}$ does not change the value of $\dot{x}(t) = -\frac{1}{2}$.
This example allows some generalizations. As the reachable sets are different for (RN) and (RL), the optimal objective values of optimal control problems will typically be different, too, $\Phi^{\mathrm{RN}} \leq \Phi^{\mathrm{RL}}$. Furthermore we saw that a relaxation of (BN) to (RN) may indeed largen the reachable set. Theorem 4.7 investigates whether this is also the case for (RL) and (BL). For the proof of this theorem we will need the famous theorem of Krein–Milman and the Gronwall lemma. Both, as well as some basic definitions, are given in appendix A.

Theorem 4.7 (Comparison of solutions of the convexified problem)
If problem (RL) has an optimal solution $(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ *with objective value* Φ^{RL}*, then for any given* $\varepsilon > 0$ *there exists a binary feasible control function* $\bar{\boldsymbol{w}}$ *and a state trajectory* $\bar{\boldsymbol{x}}$ *such that* $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}, \boldsymbol{u}^*, \boldsymbol{p}^*)$ *is an admissible solution of problem (BL) with objective value* Φ^{BL} *and*

$$\Phi^{\mathrm{BL}} \leq \Phi^{\mathrm{RL}} + \varepsilon.$$

Proof. The proof can be split up in several elementary steps.

1. We reformulate (RL) by transforming the Lagrange term to a Mayer term by introduction of an additional differential variable as described in chapter 2. The right hand side of the differential equations can still be written as in (RL).

2. Assume we have a feasible solution $(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ of (RL) that is optimal and in particular fulfills

$$\tilde{\boldsymbol{w}} \in \Omega = \left\{ \boldsymbol{w} : [t_0, t_f] \mapsto [0,1]^{n_{\tilde{w}}} \text{ with } \sum_{i=1}^{n_{\tilde{w}}} w_i(t) = 1, \quad t \in [t_0, t_f] \right\}. \tag{4.8}$$

We fix $(\boldsymbol{x}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)^T$ and regard $\boldsymbol{f}$ as a function of $\tilde{\boldsymbol{w}}$ only:

$$\tilde{\boldsymbol{f}}(\tilde{\boldsymbol{w}}) \;:=\; \boldsymbol{f}(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}, \boldsymbol{u}^*, \boldsymbol{p}^*) = \sum_{i=1}^{n_{\tilde{w}}} \boldsymbol{f}(\boldsymbol{x}^*, \boldsymbol{w^i}, \boldsymbol{u}^*, \boldsymbol{p}^*)\ \tilde{w}_i,$$

pointwise, all functions evaluated almost everywhere in $[t_0, t_f]$. We define the sets

$$\Gamma_N \;=\; \left\{ \tilde{\boldsymbol{w}} \in \Omega \;:\; \int_{t_k}^{t_{k+1}} \tilde{\boldsymbol{f}}(\tilde{\boldsymbol{w}})\ \mathrm{d}t = \int_{t_k}^{t_{k+1}} \tilde{\boldsymbol{f}}(\tilde{\boldsymbol{w}}^*)\ \mathrm{d}t, \;\; k = 0 \ldots N-1 \right\}$$

where the time points t_k depend on N and are given by

$$t_{k+1} = t_k + \frac{t_f - t_0}{N}, \;\; k \;=\; 0 \ldots N-1.$$

3. The linear operators T_k defined by

$$T_k \tilde{\boldsymbol{w}} = \int_{t_k}^{t_{k+1}} \sum_{i=1}^{n_{\tilde{w}}} \boldsymbol{f}(\boldsymbol{x}^*, \boldsymbol{w^i}, \boldsymbol{u}^*, \boldsymbol{p}^*)\ \tilde{w}_i\ \mathrm{d}t$$

are continuous. Since for a continuous operator the inverse image of a closed set is closed and the intersection of finitely many closed sets is closed, Γ_N is closed. Furthermore it is convex and nonempty for all N, as $\tilde{\boldsymbol{w}}^* \in \Gamma_N$. Hence all Γ_N are compact in a weak $*$ topology.

4. The nonemptyness and compactness of Γ_N in a Hausdorff topology allows the application of the Krein–Milman theorem A.7. Hence, Γ_N has an extreme point $\bar{\boldsymbol{w}}_{\boldsymbol{N}} = (\bar{w}_{N,1}, \ldots, \bar{w}_{N,n_{\tilde{w}}})$.

5. The functions $\bar{w}_{N,i} : [t_0, t_f] \mapsto [0,1]$ take values almost everywhere in $\{0,1\}$. Otherwise there is a contradiction to $\bar{\boldsymbol{w}}_{\boldsymbol{N}}$ being an extreme point as one can construct two functions in Γ_N of which $\bar{\boldsymbol{w}}_{\boldsymbol{N}}$ is a nontrivial convex combination, as follows.

   Suppose $\bar{\boldsymbol{w}}_{\boldsymbol{N}} \in \{0,1\}^{n_{\tilde{w}}}$ almost everywhere was not true. In this case there exists a set $E_1 \subset [t_k, t_{k+1}]$ for an index $0 \le k < N$ and a function $\boldsymbol{\zeta(\cdot)}$ nonzero on E_1 and zero elsewhere on $[t_0, t_f]$ with

$$\int_{E_1} \sum_{i=1}^{n_{\tilde{w}}} \boldsymbol{f}(\boldsymbol{x}^*, \boldsymbol{w^i}, \boldsymbol{u}^*, \boldsymbol{p}^*)\ \zeta_i(\tau)\ \mathrm{d}\tau \;=\; 0, \tag{4.9}$$

and $\bar{\boldsymbol{w}}_{\boldsymbol{N}} \pm \boldsymbol{\zeta}$ fulfills (4.8).

The proof of this statement will be by induction on the dimension n_x of $\boldsymbol{f}(\cdot)$ (the dimension of $\boldsymbol{x}$ is kept fixed, though). Let us first consider the case $n_x = 1$. We write $f_j^i = f_j(\boldsymbol{x}^*, \boldsymbol{w}^{\boldsymbol{i}}, \boldsymbol{u}^*, \boldsymbol{p}^*)$ for the j–th entry of the function vector $\boldsymbol{f}$.

As $\bar{\boldsymbol{w}}_{\boldsymbol{N}} \in \{0,1\}^{n_{\tilde{w}}}$ almost everywhere is not true, there is at least one index $0 \leq k < N$, one set $E_1 \subset [t_k, t_{k+1}]$ with positive measure and a $\delta > 0$ such that

$$||\bar{\boldsymbol{w}}_{\boldsymbol{N}}(t) - \boldsymbol{\sigma}^{\boldsymbol{i}}||_2 > \delta > 0, \quad t \in E_1, \quad i = 1 \ldots 2^{n_{\tilde{w}}}. \tag{4.10}$$

Here the $\boldsymbol{\sigma}^{\boldsymbol{i}}$ enumerate all vertices of the polytope $[0,1]^{n_{\tilde{w}}}$. Let $E_2 \subset E_1$ be such that both E_2 and its complement $E_3 := E_1 - E_2$ have positive measure. This is possible for a nonatomic measure as the Lebesgue measure.

We partition the set E_2 into $2^{n_{\tilde{w}}}$ sets $E_{2,i}$ by defining

$$E_{2,i} = \{t \in E_2 \text{ with } i = \arg\min |\bar{\boldsymbol{w}}_{\boldsymbol{N}}(t) - \boldsymbol{\sigma}^{\boldsymbol{i}}|, \text{ smallest index if not unique}\}.$$

Obviously $\bigcup_i E_{2,i} = E_2$, $E_{2,i} \cap E_{2,j} = \{\}$ for $i \neq j$ and each $E_{2,i}$ is measurable. Next we define a function $\boldsymbol{\zeta_2}(\cdot) : [t_0, t_f] \mapsto [0,1]^{n_{\tilde{w}}}$ by

$$\boldsymbol{\zeta_2}(t) = \begin{cases} \mathbf{0} & t \in [t_0, t_f] - E_2 \\ \frac{1}{2}(\bar{\boldsymbol{w}}_{\boldsymbol{N}}(t) - \boldsymbol{\sigma}^{\boldsymbol{i}}) & t \in E_{2,i} \end{cases}$$

Because of (4.10) $\boldsymbol{\zeta_2} \neq \mathbf{0}$. Furthermore $\bar{\boldsymbol{w}}_{\boldsymbol{N}} \pm \boldsymbol{\zeta_2}$ fulfill by construction (4.8). We define similarly a function $\boldsymbol{\zeta_3}(\cdot)$ on E_3 and $\boldsymbol{\zeta}(t) = \alpha_2 \boldsymbol{\zeta_2}(t) + \alpha_3 \boldsymbol{\zeta_3}(t)$. Now it is clearly possible to choose α_2 and α_3 such that

$$|\alpha_2| \leq 1, |\alpha_3| \leq 1, |\alpha_2| + |\alpha_3| > 0 \tag{4.11}$$

and

$$\begin{aligned} \int_{E_1} \sum_{i=1}^{n_{\tilde{w}}} f_1^i \, \zeta_i(\tau) \, \mathrm{d}\tau &= \alpha_2 \int_{E_2} \sum_{i=1}^{n_{\tilde{w}}} f_1^i \, \zeta_{2,i}(\tau) \, \mathrm{d}\tau + \alpha_3 \int_{E_3} \sum_{i=1}^{n_{\tilde{w}}} f_1^i \, \zeta_{3,i}(\tau) \, \mathrm{d}\tau \\ &= 0. \end{aligned} \tag{4.12}$$

The induction step is performed in a similar way. By induction hypothesis (4.9) with E_1 replaced by E_2 resp. E_3 we have nonzero measurable functions $\boldsymbol{\zeta_2}(\cdot)$ and $\boldsymbol{\zeta_3}(\cdot)$ such that

$$\int_{E_2} \sum_{i=1}^{n_{\tilde{w}}} f_j^i \, \zeta_{2,i}(\tau) \, \mathrm{d}\tau = 0, \tag{4.13}$$

$$\int_{E_3} \sum_{i=1}^{n_{\tilde{w}}} f_j^i \, \zeta_{3,i}(\tau) \, \mathrm{d}\tau = 0, \tag{4.14}$$

for $j = 1 \ldots n_x - 1$, $\boldsymbol{\zeta_2}(\cdot)$ and $\boldsymbol{\zeta_3}(\cdot)$ are identical zero on $[t_0, t_f] - E_2$ resp. $[t_0, t_f] - E_3$ and $\bar{\boldsymbol{w}}_{\boldsymbol{N}} \pm \boldsymbol{\zeta_2}$, $\bar{\boldsymbol{w}}_{\boldsymbol{N}} \pm \boldsymbol{\zeta_3}$ fulfill (4.8). Again we define $\boldsymbol{\zeta}(t) =$

$\alpha_2\boldsymbol{\zeta_2}(t)+\alpha_3\boldsymbol{\zeta_3}(t)$ and choose α_2 and α_3 such that (4.11) and the integral of the last component vanishes over E_1

$$\begin{aligned}\int_{E_1}\sum_{i=1}^{n_{\tilde{w}}} f^i_{n_x}\,\zeta_i(\tau)\,\mathrm{d}\tau &= \alpha_2\int_{E_2}\sum_{i=1}^{n_{\tilde{w}}} f^i_{n_x}\,\zeta_{2,i}(\tau)\,\mathrm{d}\tau+\alpha_3\int_{E_3}\sum_{i=1}^{n_{\tilde{w}}} f^i_{n_x}\,\zeta_{3,i}(\tau)\,\mathrm{d}\tau\\ &= 0.\end{aligned}$$

Because of (4.8) and

$$\int_{t_k}^{t_{k+1}}\sum_{i=1}^{n_{\tilde{w}}}\boldsymbol{f^i}\,(\bar{w}_{N,i}(\tau)\pm\zeta_i(\tau))\,\mathrm{d}\tau = \int_{t_k}^{t_{k+1}}\sum_{i=1}^{n_{\tilde{w}}}\boldsymbol{f^i}\,\bar{w}_{N,i}(\tau)\,\mathrm{d}\tau$$

we have $\bar{\boldsymbol{w}}_{\boldsymbol{N}}\pm\boldsymbol{\zeta}\in\Gamma_N$. This is a contradiction to $\bar{\boldsymbol{w}}_{\boldsymbol{N}}$ being an extreme point. Therefore the functions $\bar{w}_{N,i}:[t_0,t_f]\mapsto[0,1]$ take values in $\{0,1\}$ almost everywhere.

6. With fixed $(\bar{\boldsymbol{w}}_{\boldsymbol{N}},\boldsymbol{u}^*,\boldsymbol{p}^*)^T$ we define $\bar{\boldsymbol{x}}_{\boldsymbol{N}}(\cdot)$ as the unique solution of the ODE (4.2b-4.2c). We write $\boldsymbol{f}(\boldsymbol{x},\boldsymbol{w})$ for $\boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{w}(t),\boldsymbol{u}^*(t),\boldsymbol{p}^*)$ and $|\cdot|$ for the euclidian norm $||\cdot||_2$. It remains to show that $|\bar{\boldsymbol{x}}_{\boldsymbol{N}}(t_f)-\boldsymbol{x}^*(t_f)|$ gets arbitrarily small for increasing N as this ensures that the continuous Mayer term does so, too. We have

$$\begin{aligned}|\boldsymbol{x}^*(t)-\bar{\boldsymbol{x}}_{\boldsymbol{N}}(t)| &= \left|\int_{t_0}^{t}\boldsymbol{f}(\boldsymbol{x}^*,\tilde{\boldsymbol{w}}^*)-\boldsymbol{f}(\bar{\boldsymbol{x}}_{\boldsymbol{N}},\bar{\boldsymbol{w}}_{\boldsymbol{N}})\,\mathrm{d}\tau\right|\\ &= \left|\int_{t_0}^{t}\boldsymbol{f}(\boldsymbol{x}^*,\tilde{\boldsymbol{w}}^*)-\boldsymbol{f}(\boldsymbol{x}^*,\bar{\boldsymbol{w}}_{\boldsymbol{N}})+\boldsymbol{f}(\boldsymbol{x}^*,\bar{\boldsymbol{w}}_{\boldsymbol{N}})-\boldsymbol{f}(\bar{\boldsymbol{x}}_{\boldsymbol{N}},\bar{\boldsymbol{w}}_{\boldsymbol{N}})\,\mathrm{d}\tau\right|\\ &\le \left|\int_{t_0}^{t}\boldsymbol{f}(\boldsymbol{x}^*,\tilde{\boldsymbol{w}}^*)-\boldsymbol{f}(\boldsymbol{x}^*,\bar{\boldsymbol{w}}_{\boldsymbol{N}})\,\mathrm{d}\tau\right|\\ &\quad+\left|\int_{t_0}^{t}\boldsymbol{f}(\boldsymbol{x}^*,\bar{\boldsymbol{w}}_{\boldsymbol{N}})-\boldsymbol{f}(\bar{\boldsymbol{x}}_{\boldsymbol{N}},\bar{\boldsymbol{w}}_{\boldsymbol{N}})\,\mathrm{d}\tau\right| \qquad (4.15)\end{aligned}$$

For a fixed N and a given t we define $0\le k^*<N$ as the unique index such that $t_{k^*}\le t<t_{k^*+1}$. The first term of (4.15) can then be written as

$$\begin{aligned}&\left|\int_{t_0}^{t}\boldsymbol{f}(\boldsymbol{x}^*,\tilde{\boldsymbol{w}}^*)-\boldsymbol{f}(\boldsymbol{x}^*,\bar{\boldsymbol{w}}_{\boldsymbol{N}})\,\mathrm{d}\tau\right|\\ &\quad=\left|\int_{t_0}^{t_{k^*}}\boldsymbol{f}(\boldsymbol{x}^*,\tilde{\boldsymbol{w}}^*)-\boldsymbol{f}(\boldsymbol{x}^*,\bar{\boldsymbol{w}}_{\boldsymbol{N}})\,\mathrm{d}\tau+\int_{t_{k^*}}^{t}\boldsymbol{f}(\boldsymbol{x}^*,\tilde{\boldsymbol{w}}^*)-\boldsymbol{f}(\boldsymbol{x}^*,\bar{\boldsymbol{w}}_{\boldsymbol{N}})\,\mathrm{d}\tau\right|\\ &\quad=\left|\underbrace{\int_{t_0}^{t_{k^*}}\tilde{\boldsymbol{f}}(\tilde{\boldsymbol{w}}^*)-\tilde{\boldsymbol{f}}(\bar{\boldsymbol{w}}_{\boldsymbol{N}})\,\mathrm{d}\tau}_{=0}+\int_{t_{k^*}}^{t}\tilde{\boldsymbol{f}}(\tilde{\boldsymbol{w}}^*)-\tilde{\boldsymbol{f}}(\bar{\boldsymbol{w}}_{\boldsymbol{N}})\,\mathrm{d}\tau\right|\\ &\quad\le\sqrt{n_x}\int_{t_{k^*}}^{t}\left|\tilde{\boldsymbol{f}}(\tilde{\boldsymbol{w}}^*)\right|+\left|\tilde{\boldsymbol{f}}(\bar{\boldsymbol{w}}_{\boldsymbol{N}})\right|\,\mathrm{d}\tau\\ &\quad\le\sqrt{n_x}\,2M\,(t_f-t_0)\,/\,N.\end{aligned}$$

M is the supremum of $|\boldsymbol{f}(\cdot)|$ on the compact set $[0,1]^{n_{\tilde{w}}}$ with all other arguments fixed to $(\boldsymbol{x}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$. As N is free, it can be chosen such that

$$\left| \int_{t_0}^{t} \boldsymbol{f}(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}^*) - \boldsymbol{f}(\boldsymbol{x}^*, \bar{\boldsymbol{w}}_{\boldsymbol{N}}) \, \mathrm{d}\tau \right| \leq \delta e^{-\sqrt{n_x} K |t_f - t_0|} \tag{4.16}$$

for any given $\delta > 0$, where K is the Lipschitz constant of $\boldsymbol{f}(\cdot)$ with respect to the state variable $\boldsymbol{x}$. The second term of (4.15), by Lipschitz continuity

$$\left| \int_{t_0}^{t} \boldsymbol{f}(\boldsymbol{x}^*, \bar{\boldsymbol{w}}_{\boldsymbol{N}}) - \boldsymbol{f}(\bar{\boldsymbol{x}}_{\boldsymbol{N}}, \bar{\boldsymbol{w}}_{\boldsymbol{N}}) \, \mathrm{d}\tau \right| \quad \leq \quad \sqrt{n_x} \, K \int_{t_0}^{t} |\boldsymbol{x}^* - \bar{\boldsymbol{x}}_{\boldsymbol{N}}| \, \mathrm{d}\tau \tag{4.17}$$

depends on an estimation of $|\boldsymbol{x}^* - \bar{\boldsymbol{x}}_{\boldsymbol{N}}|$. With (4.16) we have

$$|\boldsymbol{x}^*(t) - \bar{\boldsymbol{x}}_{\boldsymbol{N}}(t)| \leq \delta e^{-\sqrt{n_x} K |t_f - t_0|} + \sqrt{n_x} \, K \int_{t_0}^{t} |\boldsymbol{x}^*(\tau) - \bar{\boldsymbol{x}}_{\boldsymbol{N}}(\tau)| \, \mathrm{d}\tau. \tag{4.18}$$

An application of the Gronwall inequality A.8 gives

$$|\boldsymbol{x}^*(t) - \bar{\boldsymbol{x}}_{\boldsymbol{N}}(t)| \leq \delta e^{-\sqrt{n_x} K |t_f - t_0|} \; e^{\sqrt{n_x} K |t - t_0|} = \delta \tag{4.19}$$

for all $t \in [t_0, t_f]$.

7. The Mayer term $E(\boldsymbol{x}(t_f))$ is a continuous function of $\boldsymbol{x}$, hence for all $\varepsilon > 0$ we can find a $\delta > 0$ such that

$$E(\bar{\boldsymbol{x}}(t_f)) \leq E(\boldsymbol{x}^*(t_f)) + \varepsilon$$

for all $\bar{\boldsymbol{x}}$ with $|\bar{\boldsymbol{x}}(t_f) - \boldsymbol{x}^*(t_f)| < \delta$. For this δ we find an N sufficiently large such that there is a binary feasible function $\bar{\boldsymbol{w}} = \bar{\boldsymbol{w}}_{\boldsymbol{N}}$ and a state trajectory $\bar{\boldsymbol{x}} = \bar{\boldsymbol{x}}_{\boldsymbol{N}}$ with $|\bar{\boldsymbol{x}}(t_f) - \boldsymbol{x}^*(t_f)| < \delta$ and $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}, \boldsymbol{u}^*, \boldsymbol{p}^*)$ is an admissible trajectory. As there is no Lagrange term left after the reformulation of the objective functional, the proof is complete.

■

One of the main ideas of the proof is the approximation of the optimal state trajectory $\boldsymbol{x}^*(\cdot)$. As shown in the proof, $\boldsymbol{x}^*(\cdot)$ can be approximated arbitrarily close, uniformly. Figure 4.1 illustrates this approximation for a one–dimensional example. It is possible though that the state trajectory of a singular solution cannot be obtained by a bang–bang solution, although the state trajectories obtained by bang–bang controls lie dense in the space of state trajectories obtained by relaxed controls. We will review an example in chapter 6.

In our proof we used a form of extension to the bang–bang principle, as we need the fact that the state trajectories can be approximated arbitrarily close, allowing thus to transfer the results of the linear system investigated in section 2.1.4 to the more general control–affine case needed for the applications under consideration in

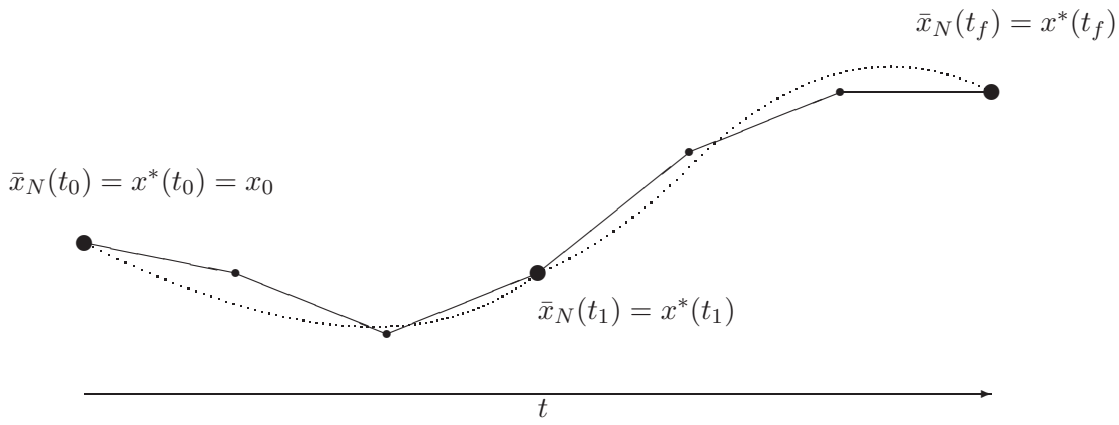

Figure 4.1: Approximation of a state trajectory $x^*(\cdot)$ with $\bar{x}_N(\cdot) \in \Gamma_N$ for $N = 2$. Note that both states $x^*(t_1)$ and $x^*(t_f)$ are reached exactly, which may take place if the right hand side function $\boldsymbol{f}(\cdot)$ does not depend on $\boldsymbol{x}$. As N goes to infinity, the difference $|\bar{\boldsymbol{x}}_{\boldsymbol{N}}(t) - \boldsymbol{x}^*(t)|$ will fall under any given tolerance δ, regardless of $\boldsymbol{f}(\cdot)$.

this work. The fact that the reachable sets of bang–bang and relaxed controls in a convexified system of the form (4.2) coincide, can be further generalized. In fact, Aumann (1965) showed that this holds true for the convex hull of a function. We state his theorem without proof in the appendix on page 172 for the convenience of the reader, but it is of no practical impact for the following. Let us now dwell on the question how the maximum principle relates to the relaxed convex problem (RL).

Remark 4.8 *Assume we have a solution $(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ of (RL) that is optimal. For this solution, the maximum principle must hold. Therefore we have the condition (2.4g) on the controls $\tilde{\boldsymbol{w}}^*$ almost everywhere in $[t_0, t_f]$:*

$$\tilde{\boldsymbol{w}}^*(t) \;=\; \arg\min_{\boldsymbol{w}} \mathcal{H}(\boldsymbol{x}^*(t), \boldsymbol{w}, \boldsymbol{u}^*(t), \boldsymbol{p}^*, \boldsymbol{\lambda}^*(t)). \tag{4.20}$$

As the Hamiltonian of the convexified system reads as

$$\begin{aligned}
\mathcal{H}(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}, \boldsymbol{u}^*, \boldsymbol{p}^*, \boldsymbol{\lambda}^*) &= \boldsymbol{\lambda}^{*T} \boldsymbol{f}(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}, \boldsymbol{u}^*, \boldsymbol{p}^*) \\
&= \boldsymbol{\lambda}^{*T} \left(\sum_{i=1}^{2^{n_w}} \boldsymbol{f}(\boldsymbol{x}^*, \boldsymbol{w^i}, \boldsymbol{u}^*, \boldsymbol{p}^*)\ \tilde{w}_i \right) \\
&= \sum_{i=1}^{2^{n_w}} \underbrace{\boldsymbol{\lambda}^{*T} \boldsymbol{f}(\boldsymbol{x}^*, \boldsymbol{w^i}, \boldsymbol{u}^*, \boldsymbol{p}^*)}_{\alpha_i :=}\ \tilde{w}_i \\
&\geq \sum_{i=1}^{2^{n_w}} \min_j \{\alpha_j\}\ \tilde{w}_i \\
&= \min_j \{\alpha_j\} \sum_{i=1}^{2^{n_w}} \tilde{w}_i \\
&= \min_j \{\alpha_j\},
\end{aligned}$$

it follows

$$\min_{\tilde{\boldsymbol{w}}} \mathcal{H}(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}, \boldsymbol{u}^*, \boldsymbol{p}^*, \boldsymbol{\lambda}^*) \;=\; \min_j \{\alpha_j\}$$

and the minimum of α_j *with* $1 \le j \le 2^{n_w}$ *determines the vector* $\tilde{w}$*. If*

$$k \;=\; \arg\min \{\alpha_j, 1 \le j \le 2^{n_w}\}$$

is unique, then the pointwise minimization of the Hamiltonian requires

$$\tilde{w}_i = \begin{cases} 1 & i = k \\ 0 & i \neq k \end{cases}$$

and the optimal solution is purely bang–bang.
If the minimum of the α_j*'s is not unique, things are more complicated. Consider a one–dimensional (*$n_w = 1$*), control–affine optimal control problem with equality* $\alpha_1 = \alpha_2$*. The Hamiltonian of the convexified problem reads as*

$$\begin{aligned} \mathcal{H} &= \boldsymbol{\lambda}^T \left(\boldsymbol{f}(\boldsymbol{x}, 0)\, \tilde{w}_2 + \boldsymbol{f}(\boldsymbol{x}, 1)\, \tilde{w}_1 \right) \\ &= \boldsymbol{\lambda}^T \left(\boldsymbol{f}(\boldsymbol{x}, 0)\, (1 - \tilde{w}_1) + \boldsymbol{f}(\boldsymbol{x}, 1)\, \tilde{w}_1 \right) \end{aligned}$$

therefore one has for the switching function

$$\begin{aligned} \mathcal{H}_{\tilde{w}_1} &= \frac{\partial}{\partial \tilde{w}_1} \left(\boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, 0)\, (1 - \tilde{w}_1) + \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, 1)\, \tilde{w}_1 \right) \\ &= \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, 1) - \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, 0) \\ &= \alpha_1 - \alpha_2. \end{aligned}$$

It follows that if we have $\boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, 0) = \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, 1)$ *on an interval, the control on this interval is singular (characterized by* $H_w = \boldsymbol{\lambda}^T \boldsymbol{f}_w = 0$ *independent of* w*, compare definition 2.6).*

Subsuming the results obtained so far, we can now state the final result of this section.

Theorem 4.9 (Comparison of solutions)
If problem (RL) has an optimal solution $(\boldsymbol{x}^*, \tilde{\boldsymbol{w}}^*, \boldsymbol{u}^*, \boldsymbol{p}^*)$ *with objective value* Φ^{RL}*, then for any given* $\varepsilon > 0$ *there exists a binary feasible control function* $\bar{\boldsymbol{w}}$ *and a state trajectory* $\bar{\boldsymbol{x}}$ *such that* $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{w}}, \boldsymbol{u}^*, \boldsymbol{p}^*)$ *is an admissible solution of problem (BL) with objective value* Φ^{BL} *and a* n_w*–dimensional control function* $\boldsymbol{w}$ *such that* $(\bar{\boldsymbol{x}}, \boldsymbol{w}, \boldsymbol{u}^*, \boldsymbol{p}^*)$ *is an admissible solution of problem (BN) with objective value* Φ^{BN} *and it holds*

$$\Phi^{\mathrm{RN}} \le \Phi^{\mathrm{RL}} \le \Phi^{\mathrm{BL}} = \Phi^{\mathrm{BN}} \le \hat{\Phi}^{\mathrm{BN}}$$

and

$$\Phi^{\mathrm{BN}} = \Phi^{\mathrm{BL}} \le \Phi^{\mathrm{RL}} + \varepsilon,$$

where $\hat{\Phi}^{\mathrm{BN}}$ *is the objective function value of any feasible solution to problem (BN).*

Proof. Admissibility follows from the fact that $\bar{\boldsymbol{w}}$ is constructed as an extreme point of a set Γ_N with values in $\{0,1\}$ and is therefore binary feasible. The corresponding state trajectory is determined such as to guarantee admissibility. These results transfer directly to the solution $(\bar{\boldsymbol{x}}, \boldsymbol{w}, \boldsymbol{u}^*, \boldsymbol{p}^*)$ of problem (BN), see theorem 4.6.
$\Phi^{\mathrm{RL}} \leq \Phi^{\mathrm{BL}}$ holds as the feasible set of the relaxed problem (RL) is a superset of the feasible set of problem (BL). The equality $\Phi^{\mathrm{BN}} = \Phi^{\mathrm{BL}}$ is given by theorem 4.6. The global minimum Φ^{BN} is not larger by definition than any feasible solution $\hat{\Phi}^{\mathrm{BN}}$. Theorem 4.7 states that $\Phi^{\mathrm{BL}} \leq \Phi^{\mathrm{RL}} + \varepsilon$ for any given $\varepsilon > 0$. It remaines to show that $\Phi^{\mathrm{RN}} \leq \Phi^{\mathrm{RL}}$. Assume $\Phi^{\mathrm{RN}} > \Phi^{\mathrm{RL}}$. Set $\varepsilon = (\Phi^{\mathrm{RN}} - \Phi^{\mathrm{RL}})/2$, then we have

$$\Phi^{\mathrm{BN}} = \Phi^{\mathrm{BL}} \leq \Phi^{\mathrm{RL}} + \varepsilon < \Phi^{\mathrm{RN}},$$

which contradicts $\Phi^{\mathrm{RN}} \leq \Phi^{\mathrm{BN}}$ as the feasible set of problem (RN) is a superset of the one of problem (BN). ■

Theorem 4.7 is a theoretical result. If an optimal control problem has singular arcs, a bang–bang solution may have to switch infinitely often in a finite time interval to approximate this singular solution. This behavior is referred to as *chattering* in the optimal control community, Zelikin & Borisov (1994). The first example of an optimal control problem exhibiting chattering behavior was given by Fuller (1963). This example and another control problem with chattering control will be investigated in chapter 6.
In the engineering community chattering behavior is called *Zeno's phenomenon*[1], e.g., Zhang *et al.* (2001).
For our purposes we do not have to care about chattering resp. Zeno's phenomenon too much, as we are interested in an approximate, near–optimal solution on a finite control grid only. Knowing the best objective value that can be achieved with a bang–bang control, we can stop an iterative process to adapt the control grid (to be

[1]This refers to the great ancient philosopher Zeno of Elea. Zeno of Elea was a pre–Socratic Greek philosopher of southern Italy and a pupil of Parmenides, see Vlastos (1967). He is mostly known for his 40 paradoxes, among which the most famous are

- *The Dichotomy*

  Motion is impossible since "that which is in locomotion must arrive at the half-way stage before it arrives at the goal."

- *The Arrow*

  "If everything when it occupies an equal space is at rest, and if that which is in locomotion is always occupying such a space at any moment, the flying arrow is therefore motionless."

- *The Achilles*

  "In a race, the quickest runner can never overtake the slowest, since the pursuer must first reach the point whence the pursued started, so that the slower must always hold a lead."

These paradoxes can be found, e.g., in *Physics* of Aristotle (350 B.C.), VI:9, 239. Zeno of Elea was the first to draw attention to the apparent interpretational problems occurring whenever an infinite number of events has to take place in a finite time interval.

presented in the next chapter) when we get closer than a prescribed small tolerance to this optimal value, obtaining a control with a finite number of switches only.

4.3 Penalty terms

As proven in the preceding section, one can always find a bang–bang solution to a convexified problem with the same objective function value up to ε, if there is a solution at all. In practice this solution is not necessarily unique and is, when applying a direct method, based on an approximation of the control space. Therefore a solution will typically contain values $\tilde{w}_i(t) \in (0,1)$ and will not be bang–bang. In this section we are interested in manipulating the optimal control problem such that its optimal solution is purely bang–bang, even in the space of approximated controls. This is achieved by adding a penalty term to the objective functional.

Definition 4.10 (Problem with penalty terms)
Problem (PRN) is given by

$$\min_{\boldsymbol{x},\boldsymbol{w},\boldsymbol{u},\boldsymbol{p}} \Phi[\boldsymbol{x},\boldsymbol{w},\boldsymbol{u},\boldsymbol{p}] + \sum_{i=1}^{n_w} \beta_i \int_{t_0}^{t_f} w_i(t)\,(1-w_i(t))dt \tag{4.21a}$$

subject to the ODE system

$$\dot{\boldsymbol{x}}(t) \;=\; \boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{w}(t),\boldsymbol{u}(t),\boldsymbol{p}), \quad t \in [t_0,t_f], \tag{4.21b}$$

with initial values

$$\boldsymbol{x}(t_0) \;=\; \boldsymbol{x_0}, \tag{4.21c}$$

and relaxed binary admissibility of $\boldsymbol{w}(\cdot)$,

$$\boldsymbol{w}(\cdot) \;\in\; \bar{\Omega}(\Psi), \tag{4.21d}$$

with $\Psi = \Psi_{\text{free}}$. *The parameters* $\beta_i \in \mathbb{R}$ *are nonnegative,* $\beta_i \geq 0$.

If we assume that problem (PRN) has an optimal trajectory $(\boldsymbol{x}^*,\boldsymbol{w}^*,\boldsymbol{u}^*,\boldsymbol{p}^*)$ for

$$(\beta_1,\ldots,\beta_{n_w})^T = \mathbf{0},$$

then there will also be an optimal trajectory for any other choice of

$$\beta_i \geq 0, \;\; i = 1 \ldots n_w,$$

as $(\boldsymbol{x}^*,\boldsymbol{w}^*,\boldsymbol{u}^*,\boldsymbol{p}^*)$ is feasible for the modified problem and yields an upper bound Φ^{PRN} and a lower bound Φ^{RN} at the same time. We see furthermore that if problem (BN) has an optimal trajectory, this solution will also be optimal for (PRN), if all β_i are chosen sufficiently large. Otherwise, as for at least one $1 \leq i \leq n_w$

$$\int_{t_0}^{t_f} w_i(t)\,(1-w_i(t)) > \delta > 0,$$

and β_i can be chosen such that the objective value

$$\Phi^{BN} < \Phi^{RN} + \beta_i \delta \leq \Phi^{PRN}$$

contradicts the nonoptimality assumption.

Remark 4.11 *As investigated in section 4.2, for convex systems there exists always a trajectory with a bang–bang control $\boldsymbol{w}(\cdot)$ and an objective value*

$$\Phi^{RL} \leq \Phi^{BL} \leq \Phi^{RL} + \varepsilon$$

if there is an optimal trajectory for problem (RL). By adding a penalty term with

$$(\beta_1, \ldots, \beta_{n_w})^T > \mathbf{0}$$

any solution that is not bang–bang almost everywhere has an increased objective value with respect to every bang–bang solution. Therefore there may be no optimal solutions any more that are not purely bang–bang. This, of course, is only a theoretical result, as chattering controls cannot be represented exactly by numerical methods.

Assume we have two different penalty parameter vectors $\boldsymbol{\beta^k}$ and $\boldsymbol{\beta^l}$ with

$$\beta_i^k < \beta_i^l, \;\; i = 1 \ldots n_w,$$

and an optimal trajectory of problem $(\mathrm{PRN})^k$, defined as problem (PRN) with $\boldsymbol{\beta} = \boldsymbol{\beta^k}$. Now let us consider the Hamiltonian $\mathcal{H}^l$ of problem $(\mathrm{PRN})^l$, evaluated at the optimal trajectory of problem $(\mathrm{PRN})^k$,

$$\begin{aligned} \mathcal{H}^l &= L + \boldsymbol{\lambda}^T \boldsymbol{f} + \sum_{i=1}^{n_w} \beta_i^l \; w_i(t) \; (1 - w_i(t)) && (4.22) \\ &= \mathcal{H}^k + \sum_{i=1}^{n_w} (\beta_i^l - \beta_i^k) \; w_i(t) \; (1 - w_i(t)). && (4.23) \end{aligned}$$

The derivative with respect to w_i reads as

$$\mathcal{H}_{w_i}^l = \mathcal{H}_{w_i}^k + \underbrace{(\beta_i^l - \beta_i^k)}_{>0} \, (1 - 2w_i(t)). \qquad (4.24)$$

We have $\mathcal{H}_{w_i}^l > \mathcal{H}_{w_i}^k$ for $w_i(t) < 0.5$ and $\mathcal{H}_{w_i}^l < \mathcal{H}_{w_i}^k$ for $w_i(t) > 0.5$. For singular controls with $\mathcal{H}_{w_i}^k = 0$ an augmentation of $\boldsymbol{\beta^k}$ thus leads to a movement of the minimizer of the Hamiltonian. This principle is depicted in figure 4.2 for a two–dimensional static optimization example with $\beta = \beta_1 = \beta_2$ ranging from 0 to 25.

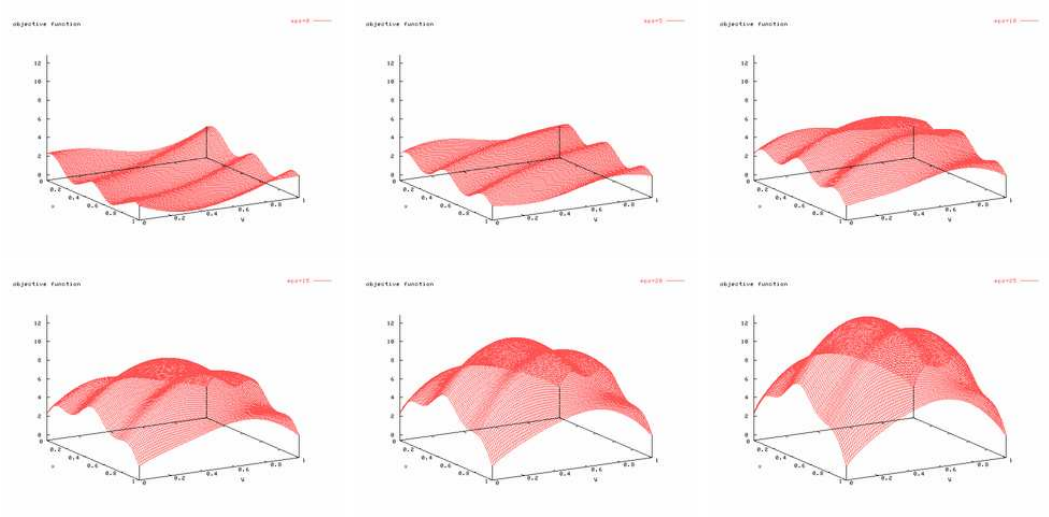

Figure 4.2: Function values of a two–dimensional example function given by $f(x_1, x_2) = 0.7\ sin(0.5 + 15\ x_1 + 6\ x_2) + 3\ (x_1 - 0.7)^2 + 3\ (x_2 - 0.4)^2$ (top left) and function values for $f(x_1, x_2) + \beta(x_1 - x_1^2) + \beta(x_2 - x_2^2)$ with $\beta = 0, 5, 10, 15, 20, 25$ from top left to right bottom.

4.4 Constraints and other extensions

In the preceding sections we investigated optimal control problems that are a special case of definition 1.14. With one exception, the extensions to general multistage mixed–integer optimal control problems are done in a straightforward way, though. This exception is general path and control constraints $\boldsymbol{c}(\cdot)$. Assume we have a feasible solution of the relaxed problem (RL). Here two cases can be differentiated: for constraints that depend only on $\boldsymbol{x}(\cdot)$, $\boldsymbol{u}(\cdot)$ and $\boldsymbol{p}$, but not on $\boldsymbol{w}(\cdot)$, the inequalities can be fulfilled to a certain tolerance with a bang–bang control as the states $\boldsymbol{x}(\cdot)$ can be approximated arbitrarily close (see proof of theorem 4.7). This is different, if $\boldsymbol{c}(\cdot)$ depends explicitly upon $\boldsymbol{w}(\cdot)$. Consider the pathological one–dimensional example with control constraints given by

$$\mathbf{0} \leq \boldsymbol{c}(\boldsymbol{w}) = \begin{pmatrix} 1 - 10^{-n} - w(t) \\ w(t) - 10^{-n} \end{pmatrix}, \ n \geq 1. \tag{4.25}$$

These constraints exclude all binary solutions $w(t) \in \{0, 1\}$, while singular controls might still be feasible. Thus it is obvious that no general bang–bang theorems are possible for general path and control constraints $\boldsymbol{c}(\cdot)$ and open questions remain that may be the topic of future research. We will restrict ourselves in the following to an assumption concerning the control constraints and replace the general path and control constraint (1.18d) by

$$-\boldsymbol{\varepsilon_c} \ \leq \ \boldsymbol{c_k}(\boldsymbol{x_k}(t), \boldsymbol{z_k}(t), \boldsymbol{u_k}(t), \boldsymbol{v}, \boldsymbol{p}), \quad t \in [\tilde{t}_k, \tilde{t}_{k+1}], \tag{4.26}$$

for a given tolerance $\boldsymbol{\varepsilon_c} > \mathbf{0}$.

As investigated in section 2.1.3, algebraic variables, due to the index 1 assumption, and multiple stages do not yield any theoretical difficulties and the results of this chapter can be transformed to such problems.
Having this in mind we can now state the following theorem for the general multistage case that allows us to *decouple* the determination of the optimal binary parameters from the determination of the optimal binary functions. Let (BN) denote the multistage mixed–integer optimal control problem (1.18), where the constraints (1.18d) are replaced by (4.26). Let (RL) denote the same problem in relaxed (with respect to the binary control functions, not to the binary parameters) and convexified form.

Theorem 4.12 (Comparison of solutions with binary parameters)
Let $k = 0 \dots n_{\mathrm{mos}} - 1$ denote the model stage index. If problem (RL) has an optimal solution $\mathcal{T}^* = (\boldsymbol{x}_k^*, \boldsymbol{z}_k^*, \tilde{\boldsymbol{w}}_k^*, \boldsymbol{u}_k^*, \boldsymbol{v}^*, \boldsymbol{p}^*)$ with objective value Φ^{RL}, then for any given $\varepsilon > 0$, $\boldsymbol{\varepsilon_c} > \mathbf{0}$ there exist binary feasible control functions $\bar{\boldsymbol{w}}_{\boldsymbol{k}}$ and state trajectories $\bar{\boldsymbol{x}}_{\boldsymbol{k}}, \bar{\boldsymbol{z}}_{\boldsymbol{k}}$ such that $(\bar{\boldsymbol{x}}_{\boldsymbol{k}}, \bar{\boldsymbol{z}}_{\boldsymbol{k}}, \bar{\boldsymbol{w}}_{\boldsymbol{k}}, \boldsymbol{u}_k^*, \boldsymbol{v}^*, \boldsymbol{p}^*)$ is an admissible solution of problem (BL) with objective value Φ^{BL} and n_w–dimensional control functions $\boldsymbol{w}_k$ such that $(\bar{\boldsymbol{x}}_{\boldsymbol{k}}, \bar{\boldsymbol{z}}_{\boldsymbol{k}}, \boldsymbol{w}_k, \boldsymbol{u}_k^*, \boldsymbol{v}^*, \boldsymbol{p}^*)$ is an admissible solution of problem (BN) with objective value Φ^{BN} and it holds

$$\Phi^{\mathrm{RN}} \leq \Phi^{\mathrm{RL}} \leq \Phi^{\mathrm{BL}} = \Phi^{\mathrm{BN}} \leq \hat{\Phi}^{\mathrm{BN}}$$

and

$$\Phi^{\mathrm{BN}} = \Phi^{\mathrm{BL}} \leq \Phi^{\mathrm{RL}} + \varepsilon,$$

where $\hat{\Phi}^{\mathrm{BN}}$ is the objective function value of any feasible solution to problem (BN).

Proof. We show that an admissible trajectory $(\bar{\boldsymbol{x}}_{\boldsymbol{k}}, \bar{\boldsymbol{z}}_{\boldsymbol{k}}, \bar{\boldsymbol{w}}_{\boldsymbol{k}}, \boldsymbol{u}_k^*, \boldsymbol{v}^*, \boldsymbol{p}^*)$ of (BL) exists with objective value $\Phi^{\mathrm{BL}} \leq \Phi^{\mathrm{RL}} + \varepsilon$. The other claims follow with the same arguments as in theorem 4.9. We only state the necessary extensions to this proof.
We fix $\boldsymbol{v}^*$, that fulfills by assumption the binary constraint (1.18h), as we do with all $\boldsymbol{u}_k^*$ and with $\boldsymbol{p}^*$. Then we notice that by the index 1 assumption an additional differentiation of the algebraic constraints (1.18c) will formally transform the algebraic variables into differential ones. Therefore all algebraic variables can be determined such that for any given δ (1.18b,1.18c) hold and $|\bar{\boldsymbol{z}}_{\boldsymbol{k}}(t) - \boldsymbol{z}^*(t)| < \delta$ for all $t \in [t_0, t_f]$, if this is possible for differential variables.
The singlestage case can be transferred directly to the multistage one, as the stage transition conditions are continuous and depend only on the states that can be approximated arbitrarily close and on the parameters $(\boldsymbol{v}^*, \boldsymbol{p}^*)$ that are fixed. Leaves to consider the path constraints (4.26) and the interior point constraints (1.18e, 1.18f) that were temporarily neglected in theorem 4.9. For the path constraints we can choose $\delta > 0$ in

$$|(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{z}}_k)(t) - (\boldsymbol{x}^*, \boldsymbol{z}^*)(t)| < \delta, \quad t \in [t_0, t_f]$$

as a minimum of the values necessary to ensure that the continuous objective function and the continuous path controls are within the prescribed tolerances ε resp. $\boldsymbol{\varepsilon_c}$. For

the interior point constraints we can proceed in a similar manner. We prescribe a tolerance $\boldsymbol{\varepsilon_r}$ and can fulfill all interior point inequality and equality constraints up to this precision by an adequate choice of δ. This completes the proof. ■

Theorem 4.12 has one very important consequence. To determine optimal binary parameters $\boldsymbol{v}^*$ it is sufficient to solve an associated control problem with relaxed binary control functions. For $\boldsymbol{v}^*$ fixed we may then in a second step find the optimal binary admissible control functions $\bar{\boldsymbol{w}}_{\boldsymbol{k}}$. This decoupling of the computationally expensive integer problems to determine binary parameters and binary control functions is beneficial with respect to the overall run time of a solution procedure.

For penalty terms constraints pose additional difficulties, too. In the presence of path constraints the derivative of the Hamiltonian with respect to w_i reads as

$$\mathcal{H}^l_{w_i} \;=\; \mathcal{H}^k_{w_i} + \boldsymbol{\mu}^T \boldsymbol{c}_{\boldsymbol{w_i}} + (\beta^l_i - \beta^k_i)\,(1 - 2w_i(t)). \tag{4.27}$$

Now a penalty increase $(\beta^l_i - \beta^k_i)$ not necessarily leads to a different trajectory. The derivative of the Hamiltonian evaluated for an optimal trajectory of $(\mathrm{PRN})^k$ with $w_i(t)$ being a singular control, has to vanish as stated by the maximum principle. For unconstrained systems the penalty increase lead to a "movement" of the optimal trajectory of $(\mathrm{PRN})^k$ towards an optimal solution of $(\mathrm{PRN})^l$ such that the maximum principle is again satisfied. If path constraints are active, the additional term

$$(\beta^l_i - \beta^k_i)\,(1 - 2w_i(t))$$

may also be "compensated" by an augmented Lagrange multiplier $\boldsymbol{\mu}$. In other words, the solution is "pushed" against constraints that are cutting off a binary solution from the feasible region, the trajectory may get stuck in this point.

Figure 4.3 illustrates this situation for the two–dimensional nondynamic example given in figure 4.2. Iterative descent–based optimization methods will get stuck in such a point. If no additional degrees of freedom in continuous variables are available to change the position of the constraint with respect to the binary variables, multistart or backtracking techniques have to be applied. This is more likely in pure integer than in mixed–integer problems. This possibility to get stuck is indeed the reason, why penalty term techniques are not very popular in integer programming and why we treat binary parameters $\boldsymbol{v}$ as described in chapter 3 instead of applying the same convexification and penalization techniques as will be presented in chapter 5.
But, for optimization problems resulting from discretized optimal control problems, compare chapter 2, there exists a remedy to overcome this problem. By modifying the control discretization grid, i.e., by refining the control approximation, the optimization problem is transformed into a related one with additional degrees of freedom that may be used to fulfill the constraint. Details will be given in chapter 5.

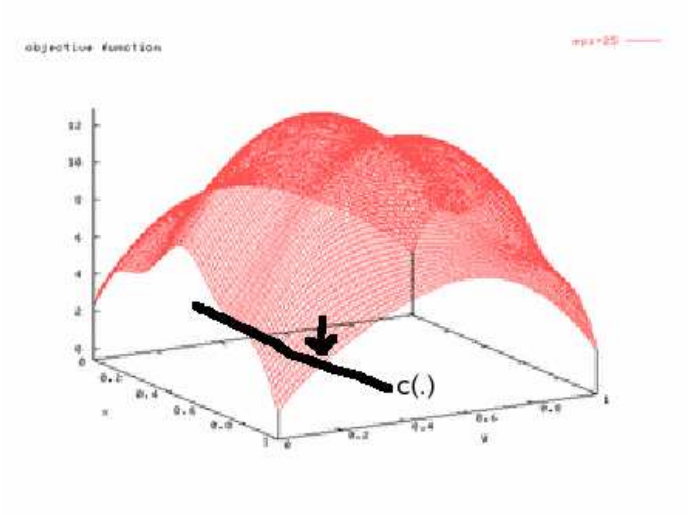

Figure 4.3: Two–dimensional example function of figure 4.2 with a constraint cutting off the bottom left binary solution $(0, 0)$.

4.5 Summary

In this chapter we presented a methodology to convexify optimal control problems with respect to the binary control functions. We stated several theorems that clarify the connection between the nonlinear and a convexified problem on the one hand and between binary and relaxed control problems on the other hand. In particular we proved that, assumed there exists an optimal trajectory to the relaxed convexified problem with objective value Φ^{RL}, there also exists a feasible trajectory for the original, mixed–integer optimal control problem with an objective value $\Phi^{\mathrm{RL}} \leq \Phi^{\mathrm{BN}} \leq \Phi^{\mathrm{RL}} + \varepsilon$ for any given $\varepsilon > 0$. This fact will be exploited in the following chapter, where the solution of the relaxed convexified problem serves as a (reachable!) lower bound and stopping criterion of an iterative solving procedure.
We proved in theorem 4.12 that binary parameters $\boldsymbol{v}^*$ that are optimal for the control problem with relaxed binary control functions will also be optimal for the integer problem. This allows to *decouple* the determination of the computationally expensive integer problems if parameters as well as control functions are present. This is very beneficial with respect to the overall run time of a solution procedure.
In section 4.3 we formulated an optimal control problem enriched by an additional penalty term in the objective functional and investigated some properties of such a control problem. In particular we saw that the optimal trajectory of it will be binary admissible if the penalty parameter vector $\boldsymbol{\beta}$ is chosen sufficiently large. In section 4.4 we discussed extensions to general multistage mixed–integer optimal control problems and occurring problems in the presence of path and control constraints.
The theoretical results obtained in this chapter will be used to motivate the numerical methods that are presented in chapter 5.

Chapter 5

Numerical methods for binary control functions

As shown in the last chapter, the determination of optimal binary parameters $\boldsymbol{v}^*$ can be decoupled from the determination of optimal binary control functions $\boldsymbol{w}^*(\cdot)$ to make solution algorithms more efficient. In this chapter we are going to define algorithms to solve mixed–integer optimal control problems of the form (1.14) without resp. with fixed binary parameters $\boldsymbol{v}$, i.e.,

$$\min_{\boldsymbol{x},\boldsymbol{z},\boldsymbol{w},\boldsymbol{u},\boldsymbol{p}} \Phi[\boldsymbol{x},\boldsymbol{z},\boldsymbol{w},\boldsymbol{u},\boldsymbol{p}] \tag{5.1a}$$

subject to the DAE system

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{z}(t),\boldsymbol{w}(t),\boldsymbol{u}(t),\boldsymbol{p}), \quad t \in [t_0,t_f], & \text{(5.1b)} \\ \boldsymbol{0} &= \boldsymbol{g}(\boldsymbol{x}(t),\boldsymbol{z}(t),\boldsymbol{w}(t),\boldsymbol{u}(t),\boldsymbol{p}), \quad t \in [t_0,t_f], & \text{(5.1c)} \end{aligned}$$

control and path constraints

$$\boldsymbol{0} \leq \boldsymbol{c}(\boldsymbol{x}(t),\boldsymbol{z}(t),\boldsymbol{w}(t),\boldsymbol{u}(t),\boldsymbol{p}), \quad t \in [t_0,t_f], \tag{5.1d}$$

interior point inequalities and equalities

$$\begin{aligned} \boldsymbol{0} &\leq \boldsymbol{r}^{\mathbf{ieq}}(\boldsymbol{x}(t_0),\boldsymbol{z}(t_0),\boldsymbol{x}(t_1),\boldsymbol{z}(t_1),\ldots,\boldsymbol{x}(t_f),\boldsymbol{z}(t_f),\boldsymbol{p}), & \text{(5.1e)} \\ \boldsymbol{0} &= \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{x}(t_0),\boldsymbol{z}(t_0),\boldsymbol{x}(t_1),\boldsymbol{z}(t_1),\ldots,\boldsymbol{x}(t_f),\boldsymbol{z}(t_f),\boldsymbol{p}), & \text{(5.1f)} \end{aligned}$$

and binary admissibility of $\boldsymbol{w}(\cdot)$

$$\boldsymbol{w}(\cdot) \in \Omega(\Psi). \tag{5.1g}$$

Please note that we restrict our investigation to the singlestage case only because of notational simplicity. All algorithms can be applied to multistage problems as well. We will base all algorithms that are presented in the sequel of this work on the direct multiple shooting method. We discretize the control space as described in chapter 2

with constant functions. We restrict the optimization space thus to functions that can be written as

$$\boldsymbol{w}(t) \;=\; \boldsymbol{q_i}, \quad t \in [t_i, t_{i+1}], \quad i = 0, \ldots, n_{\mathrm{ms}} - 1, \tag{5.2}$$

compare (2.29). The constant functions $\boldsymbol{q_i} \in \mathbb{R}^{n_w}$ have to take values $\boldsymbol{q_i} \in \{0,1\}^{n_w}$ or, for the relaxed problem, $\boldsymbol{q_i} \in [0,1]^{n_w}$ to be admissible. The continuous control functions $\boldsymbol{u}(\cdot)$ are discretized in a similar manner, not necessarily with constant functions, but in this chapter $\boldsymbol{q_i}$ will refer to a discretization of $\boldsymbol{w}(\cdot)$ exclusively for the sake of notational simplicity. The underlying control discretization grid depends upon the number n_{ms} and positions t_i of possible changes in the constant control function values. We will refer to it as

$$\mathcal{G} = \{t_0, t_1, \ldots, t_{n_{\mathrm{ms}}}\}.$$

If the feasible switching set is free, i.e. $\Psi = \Psi_{\mathrm{free}}$, the number of multiple shooting nodes n_{ms} and the time points t_i are free parameters of the direct multiple shooting method and $\mathcal{G}$ can be chosen by a user or appropriate methods, see section 5.3. If $\Psi = \Psi_\tau$, then we set

$$n_{\mathrm{ms}} := n_\tau \text{ and } t_i := \tau_i, \quad i = 0 \ldots n_{\mathrm{ms}}$$

to guarantee that the jump conditions in (1.13) are satisfied. In this case we lose some degrees of freedom. The methods presented in sections 5.2, 5.3 and 5.5 are based upon the freedom to determine switching points, thus these methods can only be applied when $\Psi = \Psi_{\mathrm{free}}$.
We will first mention *rounding strategies*, followed by an investigation of the *switching time approach* in section 5.2. The latter is based on a reformulation of the single stage model to a multistage model with free stage lengths. In section 5.3 we will examine adaptivity issues in the control discretization and propose an algorithm to refine the discretization wherever controls are not at their respective bounds. In section 5.4 we present a penalty term homotopy. In section 5.5 we will finally formulate our new algorithm to solve mixed–integer optimal control problems.

5.1 Rounding strategies

Rounding strategies are based upon a fixed discretization $\mathcal{G}$ of the control space. This discretization may be enforced by a Ψ_τ or may result from a users choice resp. an adaptive refinement procedure. Despite the fact that we have a finite–dimensional binary optimization problem, there is a difference to generic static integer optimization problems of the form (3.1), because there is a "connection" between some of the $n_w \cdot n_{\mathrm{ms}}$ variables. More precisely we have n_w sets of n_{ms} variables that discretize the same control function, only at different times.
The rounding approach to solve problem (5.1) consists of relaxing the integer requirements $\boldsymbol{q_i} \in \{0,1\}^{n_w}$ to $\tilde{\boldsymbol{q}}_{\boldsymbol{i}} \in [0,1]^{n_w}$ and to solve a relaxed problem first. The

obtained solution $\tilde{\boldsymbol{q}}$ can then be investigated – in the best case it is an integer feasible bang-bang solution and we have found an optimal solution for the integer problem. In case the relaxed solution is not integer, one of the following rounding strategies can be applied. The constant values $q_{j,i}$ of the control functions $w_j(t)$, $j = 1 \dots n_w$ and $t \in [t_i, t_{i+1}]$, are fixed to

- Rounding strategy SR (standard rounding)

$$q_{j,i} = \begin{cases} 1 & \text{if } \tilde{q}_{j,i} \geq 0.5 \\ 0 & \text{else} \end{cases} .$$

- Rounding strategy SUR (sum up rounding)

$$q_{j,i} = \begin{cases} 1 & \text{if } \sum_{k=0}^{i} \tilde{q}_{j,k} - \sum_{k=0}^{i-1} q_{j,k} \geq 1 \\ 0 & \text{else} \end{cases} .$$

- Rounding strategy SUR-0.5 (sum up rounding with a different threshold)

$$q_{j,i} = \begin{cases} 1 & \text{if } \sum_{k=0}^{i} \tilde{q}_{j,k} - \sum_{k=0}^{i-1} q_{j,k} \geq 0.5 \\ 0 & \text{else} \end{cases} .$$

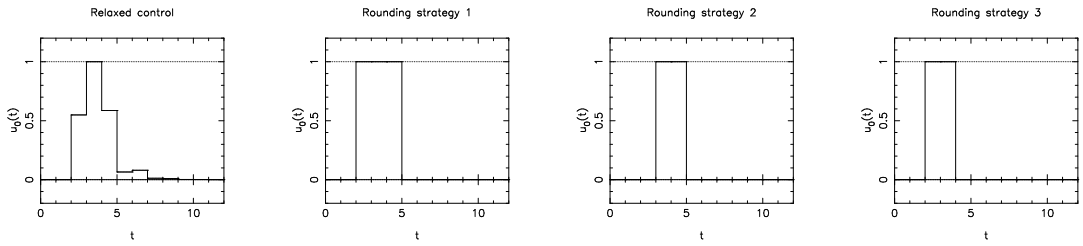

Figure 5.1: One–dimensional example of the rounding strategies. From left to right the relaxed solution $\tilde{\boldsymbol{q}}$ and solutions $\boldsymbol{q}$ obtained by rounding strategy SR, SUR and SUR-0.5.

Figure 5.1 shows an illustrative example of the effect of the different rounding strategies. For strategies SUR and SUR-0.5 the values of the $\tilde{q}_{j,i}$ are summed up over the intervals to have

$$\int_{t_0}^{t_f} w_j(\tau)\, \mathrm{d}\tau \approx \int_{t_0}^{t_f} \tilde{w}_j(\tau)\, \mathrm{d}\tau$$

for all $j = 1 \dots n_w$.
Special care has to be taken if the control functions have to fulfill the special ordered set type one restriction (see page 53) as it arises from a convexification, compare chapter 4. Many rounded solutions will violate (3.2b). Rounding strategy SR preserves this property if and only if exactly one value $\tilde{q}_{j,i} \geq 0.5$ on each interval i. For

the sum up rounding strategies this is not enough, the sum of several controls may show similar behavior over the multiple shooting intervals. For problems with the SOS1 property we therefore propose to use one of the following rounding strategies that guarantee (3.2b). We fix the constant values $q_{j,i}$ of the control functions $w_j(t)$, $j = 1 \dots n_w$ and $t \in [t_i, t_{i+1}]$, to

- Rounding strategy SR-SOS1 (standard)

$$q_{j,i} = \begin{cases} 1 & \text{if } \tilde{q}_{j,i} \geq \tilde{q}_{k,i} \; \forall \;\; k \neq j \text{ and } j < k \; \forall \;\; k : \tilde{q}_{j,i} = \tilde{q}_{k,i} \\ 0 & \text{else} \end{cases} .$$

- Rounding strategy SUR-SOS1 (sum up rounding)

$$\begin{aligned} \hat{q}_{j,i} &= \sum_{k=0}^{i} \tilde{q}_{j,k} - \sum_{k=0}^{i-1} q_{j,k} \\ q_{j,i} &= \begin{cases} 1 & \text{if } \hat{q}_{j,i} \geq \hat{q}_{k,i} \; \forall \;\; k \neq j \text{ and } j < k \; \forall \;\; k : \hat{q}_{j,i} = \hat{q}_{k,i} \\ 0 & \text{else} \end{cases} . \end{aligned}$$

Rounding strategies yield trajectories that fulfill the integer requirements, but are typically not optimal and often not even admissible. Nevertheless rounding strategies may be applied successfully to obtain upper bounds in a Branch and Bound scheme, to get a first understanding of a systems behavior or to yield initial values for the switching time optimization approach. Rounding strategy SUR-SOS1 is specifically tailored to the special ordered set restrictions that stem from the convexification and works well for a suitably chosen discretization grid, as it reflects the typical switching behavior for singular resp. on constrained arcs.

5.2 Switching time optimization

One possibility to solve problem (5.1) is motivated by the idea to optimize the switching structure and to take the values of the binary controls fixed on given intervals, as is done for bang-bang arcs in indirect methods. Let us consider the one–dimensional case, $n_w = 1$, first. Instead of the control $w(\cdot) : [t_0, t_f] \mapsto \{0, 1\}$ we do get n_{mos} fixed constant control functions

$$w_k : [\tilde{t}_k, \tilde{t}_{k+1}] \mapsto \{0, 1\}$$

defined by

$$w_k(t) = \begin{cases} 0 & \text{if k even} \\ 1 & \text{if k odd} \end{cases} , \quad t \in [\tilde{t}_k, \tilde{t}_{k+1}] \tag{5.3}$$

with $k = 0 \dots n_{\text{mos}} - 1$ and $t_0 = \tilde{t}_0 \leq \tilde{t}_1 \leq \dots \leq \tilde{t}_{n_{\text{mos}}} = t_f$.

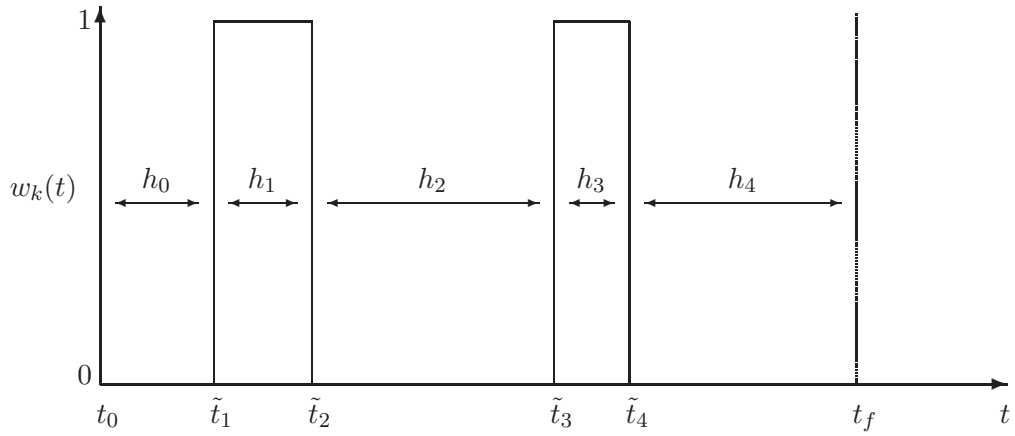

Figure 5.2: Switching time optimization, one–dimensional example with $n_{\text{mos}} = 5$.

These control functions will enter a multistage optimal control problem of the form (1.18). If we assume that an optimal binary control function $\boldsymbol{w}(\cdot)$ switches only finitely often, then problem (5.1) is equivalent to optimizing n_{mos} and the time vector $\tilde{\boldsymbol{t}}$, respectively the vector $\boldsymbol{h}$ of stage lengths $h_k := \tilde{t}_{k+1} - \tilde{t}_k$, in a multistage formulation

$$\min_{\boldsymbol{x_k}, \boldsymbol{z_k}, \boldsymbol{u_k}, \boldsymbol{p}, \boldsymbol{h}, n_{\text{mos}}} \sum_{k=0}^{n_{\text{mos}}-1} \Phi_k[\boldsymbol{x_k}, \boldsymbol{z_k}, \boldsymbol{w_k}, \boldsymbol{u_k}, \boldsymbol{p}] \tag{5.4a}$$

subject to the DAE model stages (from now on $k = 0 \ldots n_{\text{mos}} - 1$)

$$\begin{aligned} \dot{\boldsymbol{x}}_{\boldsymbol{k}}(t) &= \boldsymbol{f_k}(\boldsymbol{x_k}(t), \boldsymbol{z_k}(t), \boldsymbol{w_k}(t), \boldsymbol{u_k}(t), \boldsymbol{p}), \quad t \in [\tilde{t}_k, \tilde{t}_{k+1}] && \text{(5.4b)} \\ \mathbf{0} &= \boldsymbol{g_k}(\boldsymbol{x_k}(t), \boldsymbol{z_k}(t), \boldsymbol{w_k}(t), \boldsymbol{u_k}(t), \boldsymbol{p}), \quad t \in [\tilde{t}_k, \tilde{t}_{k+1}] && \text{(5.4c)} \end{aligned}$$

control and path constraints

$$\mathbf{0} \leq \boldsymbol{c_k}(\boldsymbol{x_k}(t), \boldsymbol{z_k}(t), \boldsymbol{w_k}(t), \boldsymbol{u_k}(t), \boldsymbol{p}), \quad t \in [\tilde{t}_k, \tilde{t}_{k+1}], \tag{5.4d}$$

interior point inequalities and equalities with k_i denoting the index of a model stage containing t_i, that is $t_i \in [\tilde{t}_{k_i}, \tilde{t}_{k_i+1}]$,

$$\begin{aligned} \mathbf{0} &\leq \boldsymbol{r}^{\mathbf{ieq}}(\boldsymbol{y}_{\boldsymbol{k_0}}(t_0), \boldsymbol{y}_{\boldsymbol{k_1}}(t_1), \ldots, \boldsymbol{y}_{\boldsymbol{k_{n_{\text{ms}}}}}(t_{n_{\text{ms}}}), \boldsymbol{p}), && \text{(5.4e)} \\ \mathbf{0} &= \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{y}_{\boldsymbol{k_0}}(t_0), \boldsymbol{y}_{\boldsymbol{k_1}}(t_1), \ldots, \boldsymbol{y}_{\boldsymbol{k_{n_{\text{ms}}}}}(t_{n_{\text{ms}}}), \boldsymbol{p}), && \text{(5.4f)} \end{aligned}$$

and for positive $h_k \geq 0$ the constraint

$$\sum_{k=0}^{n_{\text{mos}}-1} h_k = t_f - t_0. \tag{5.4g}$$

In (5.4) all $w_k(t)$ are fixed to either 0 or 1. This approach is visualized in figure 5.2 with $n_{\text{mos}} = 5$.

For fixed n_{mos} we have an optimal control problem that fits into the definition of problem (1.18) and can be solved with standard methods, where the stage lengths h_k take the role of parameters that have to be determined. The approach can be extended in a straightforward way to a n_w–dimensional binary control function $\boldsymbol{w}(\cdot)$. Instead of (5.3) one defines $\boldsymbol{w_k}$ as

$$\boldsymbol{w_k}(t) = \boldsymbol{w^i} \quad \text{if } k = j\, 2^{n_w} + i - 1, \quad t \in [\tilde{t}_k, \tilde{t}_{k+1}] \tag{5.5}$$

for some $j \in \mathbb{N}_0$ and some $1 \le i \le 2^{n_w}$. The $\boldsymbol{w^i}$ enumerate all 2^{n_w} possible assignments of $\boldsymbol{w}(\cdot) \in \{0,1\}^{n_w}$, compare chapter 4.

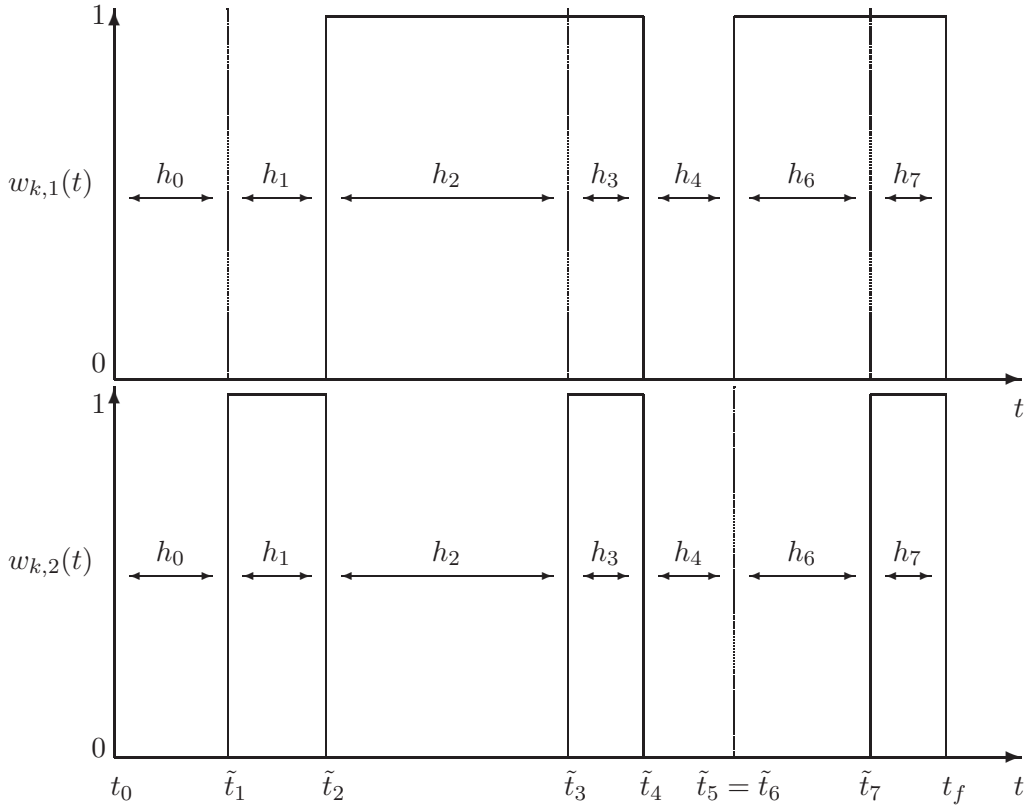

Figure 5.3: Switching time optimization, two–dimensional example with $n_{\text{mos}} = 8$ and $h_5 = 0$.

Figure 5.3 shows a two–dimensional example. For $n_w = 2$ we have

$$\boldsymbol{w^1} = (0,0)^T, \boldsymbol{w^2} = (0,1)^T, \boldsymbol{w^3} = (1,0)^T, \boldsymbol{w^4} = (1,1)^T.$$

If we choose $n_{\text{mos}} = 8$ we obtain eight stages with

$$\boldsymbol{w_0}(t) = \boldsymbol{w^1}, \boldsymbol{w_1}(t) = \boldsymbol{w^2}, \boldsymbol{w_2}(t) = \boldsymbol{w^3}, \boldsymbol{w_3}(t) = \boldsymbol{w^4},$$
$$\boldsymbol{w_4}(t) = \boldsymbol{w^1}, \boldsymbol{w_5}(t) = \boldsymbol{w^2}, \boldsymbol{w_6}(t) = \boldsymbol{w^3}, \boldsymbol{w_7}(t) = \boldsymbol{w^4}.$$

Depicted is an example where $h_5 = 0$, the assignment $\boldsymbol{w^2} = (0,1)^T$ enters effectively only once on the second stage $[\tilde{t}_1, \tilde{t}_2]$.

This two–dimensional example already indicates some intrinsic problems of the switching time approach. First, the number of model stages grows exponentially not only in the number of control functions, but also in the number of expected switches of the binary control functions. Starting from a given number of stages, as depicted in figure 5.3, allowing a small change in one of the control functions requires additional 2^{n_w} stages. If it is indeed exactly one function $w_i(\cdot)$ that changes while all others stay fixed, $2^{n_w} - 1$ of the newly introduced stages will have length 0. This leads to a second drawback, namely a nonregular situation that may occur when stage lengths are reduced to zero. Consider the situation depicted in figure 5.4. The length of an intermediate stage corresponding to control $w_2(t) = 0$ has been reduced to zero by the optimizer. Therefore the sensitivity of the optimal control problem with respect to h_1 and h_3 is given by the value of their sum $h_1 + h_3$ only. Thus special care has to be taken to treat the case where stage lengths diminish during the optimization procedure. Kaya & Noakes (1996, 2003) and Maurer *et al.* (2005) propose an algorithm to eliminate such stages. This is possible, still the stage cannot be reinserted, as the time when to insert it is undetermined.

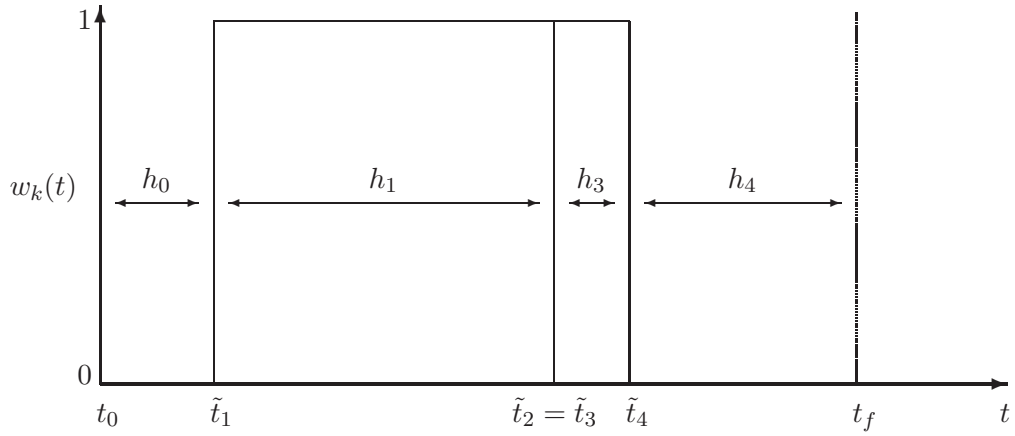

Figure 5.4: Switching time optimization, one–dimensional example with diminishing interior stage and occurring nonregularity. $\tilde{t}_2 = \tilde{t}_3$ can take any value in the interval $[\tilde{t}_1, \tilde{t}_4]$ without any influence on the optimal control problem.

The third drawback is that the number of switches is typically not known, left alown the precise switching structure. Some authors propose to iterate on n_{mos} until there is no further decrease in the objective function of the corresponding optimal solution, Rehbock & Caccetta (2002) and Kaya & Noakes (1996, 2003). But it should be stressed that this can only be applied to more complex systems, if initial values for the location of the switching points that are close to the optimum are available, as they are essential for the convergence behavior of the underlying method. This is closely connected to the fourth and most important drawback of the switching time approach. The reformulation yields additional nonconvexities in the optimization space. Even if the optimization problem is convex in the optimization variables resulting from a constant discretization of the control function $\boldsymbol{w}(\cdot)$, the reformulated problem may be nonconvex.

To demonstrate this effect we will investigate the fishing problem introduced in section 1.4.2. We will not prove nonconvexity in mathematical rigor, but instead show some results obtained by simulation that give an insight into the issue. The optimization problem that results from a switching time approach with $n_{\text{mos}} = 5$ stages reads as

$$\min_{\boldsymbol{x_k},\boldsymbol{h}} \int_{t_0}^{t_f} (x_0(t)-1)^2 + (x_1(t)-1)^2 \, dt \tag{5.6a}$$

subject to the ODE system

$$\dot{x}_0(t) = x_0(t) - x_0(t)x_1(t), \tag{5.6b}$$
$$\dot{x}_1(t) = -x_1(t) + x_0(t)x_1(t), \tag{5.6c}$$

for $t \in [t_0, \tilde{t}_1] \cup [\tilde{t}_2, \tilde{t}_3] \cup [\tilde{t}_4, t_f]$ and to

$$\dot{x}_0(t) = x_0(t) - x_0(t)x_1(t) - c_0 x_0(t), \tag{5.6d}$$
$$\dot{x}_1(t) = -x_1(t) + x_0(t)x_1(t) - c_1 x_1(t), \tag{5.6e}$$

for $t \in [\tilde{t}_1, \tilde{t}_2] \cup [\tilde{t}_3, \tilde{t}_4]$. The initial values are given by

$$\boldsymbol{x}(t_0) = \boldsymbol{x_0} \tag{5.6f}$$

and we have the constraint

$$\sum_{k=0}^{4} h_k = t_f - t_0 \tag{5.6g}$$

on the model stages with

$$h_k = \tilde{t}_{k+1} - \tilde{t}_k \geq 0, \tag{5.6h}$$

$k = 0 \ldots 4$. Note that the binary control assignments $w^1 = 0$ and $w^2 = 1$ have been inserted directly into the formulation. A local optimum of problem (5.6) with initial values given in appendix B is

$$\boldsymbol{h}^* = (2.46170, 1.78722, 0.89492, 0.31169, 6.54447)^T \tag{5.7}$$

with a switching structure similar to that of figure 5.2 and an objective value of $\Phi = 1.34967$.

We fix some variables[1] to obtain an objective landscape by varying h_2, h_3 and h_4 and integrating system (5.6). The length of the third stage, given by h_2, can take all values in $[0, h_2^* + h_3^* + h_4^*]$. The length of the fourth stage, i.e. h_3, can take all values in $[0, h_2^* + h_3^* + h_4^* - h_2]$. The length of the terminal stage h_4 is set to $t_f - t_0 - h_0 - h_1 - h_2 - h_3$ to satisfy constraint (5.6g). The chosen step size for

[1] $h_0 = h_0^* = 2.46170$ and $h_1 = h_1^* = 1.78722$.

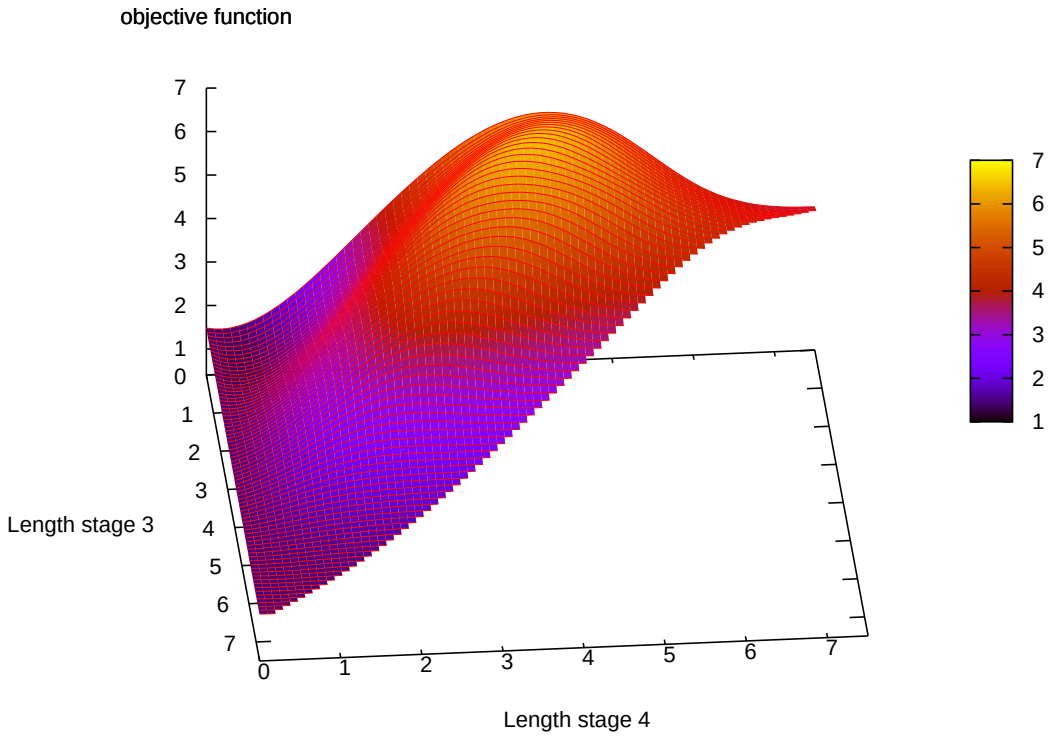

Figure 5.5: Objective function value of the fishing problem in switching time formulation, dependent on $\tilde{t}_3$ and $\tilde{t}_4$, the begin respectively end of the fourth stage.

the visualization is 0.1. Figure 5.5 shows the landscape obtained by a simulation. Obviously the objective function is nonconvex and contains multiple local minima. The nonconvexity is even more apparent, when the case $n_{\text{mos}} = 3$ is considered, as we are not close to the global minimum by fixing some of the variables to their optimal values. Figure B.2 in appendix B shows a simulation for it and a comparison with the optimal control problem in another formulation that is less nonconvex.

Despite the mentioned drawbacks of the switching time optimization approach, it can be applied to practical problems, if it is combined with a bunch of other concepts. This includes rigorous lower and upper bounds, good initial values, a strategy to deal with diminishing stage lengths and the direct multiple shooting method that helps when dealing with nonconvexities, compare the comments in section 2.4 and the explicit example in the appendix. In B.4 it is shown how the initialization of the multiple shooting node values for the differential states can help to let the solution converge towards the local optimum one is looking for.

An advantage of the switching time approach is that solutions can be formulated in a compact way that we will define here. We will use this formulation for notational brevity in the proceeding of this thesis.

Definition 5.1 (Stage lengths solution)
The stage lengths solution

$$\mathcal{S}(q; h_0, h_1, \dots h_{n_{\text{mos}}})$$

with $q \in \{0,1\}, h_i \geq 0$ *denotes the one–dimensional control function* $w(\cdot)$ *mapping* $[t_0, t_f] \mapsto \{0,1\}$ *given by*

$$w(t) = \begin{cases} q & t \in [\tilde{t}_{2\,k}, \tilde{t}_{2\,k+1}] \\ 1-q & t \in (\tilde{t}_{2\,k+1}, \tilde{t}_{2\,k+2}] \end{cases}, \quad k = 0 \dots \left\lceil \frac{n_{\text{mos}}}{2} \right\rceil$$

with $\tilde{t}_k := t_0 + \sum_{i=1}^{k} h_{i-1}$, $k = 0 \dots n_{\text{mos}}+1$. *If* $\boldsymbol{q}$ *is a vector, then* $\mathcal{S}(\boldsymbol{q}; h_0, h_1, \dots h_{n_{\text{mos}}})$ *denotes the solution* $w(t) = q_k$ *for* $t \in [\tilde{t}_k, \tilde{t}_{k+1}]$.

In other words, a stage lengths solution yields the value of a one–dimensional binary control function on the very first interval $[\tilde{t}_0, \tilde{t}_1]$ and all interval lengths, thus all information necessary to reconstruct $w(\cdot)$ on $[t_0, t_f]$. To give an example, the solution corresponding to (5.7) can be written as

$$w(\cdot) = \mathcal{S}(0; 2.46170, 1.78722, 0.89492, 0.31169, 6.54447).$$

5.3 Adaptive control grid

When control functions are discretized with piecewise constant functions (5.2), we restrict the search for an optimal admissible trajectory to a subspace. In this space there may be no admissible trajectory at all. If an admissible optimal solution exists, it typically has a higher objective value than the optimal trajectory of the full, relaxed, infinite–dimensional control space that will be denoted by $\mathcal{T}^*$ in the following. But, as was shown in chapter 4, the trajectories with piecewise constant controls, being a superset of the trajectories with bang–bang controls, lie dense in the space of all trajectories. In other words, given a tolerance ε, one can always find a control discretization $t_1 \dots t_{n_{\text{ms}}}$ such that the Euclidean distance between the corresponding optimal trajectory and $\mathcal{T}^*$ is less than ε for each time $t \in [t_0, t_f]$. The goal of this section is to describe adaptivity in the control discretization grid $\mathcal{G}$ that serves two purposes: first, we can use it to obtain an estimation for the optimal objective function value of $\mathcal{T}^*$ via *extrapolation* and second, we can use it to get a grid on which we may approximate $\mathcal{T}^*$ arbitrarily close with a bang–bang solution.
The control grid can be modified in two different ways to get a better objective function value. The first one would be to change the position of the time points t_i where jumps in the controls may occur. This approach corresponds to the switching time approach presented in section 5.2. The second way we will follow here is to insert additional time points.
When we add a time point where a control may change its constant value, we enlarge the reachable set. In fact, the insertion of an additional time point $\tau \in [t_i, t_{i+1}]$ is equivalent to leaving away the restriction

$$\boldsymbol{w}(\tau^-) = \boldsymbol{w}(\tau^+)$$

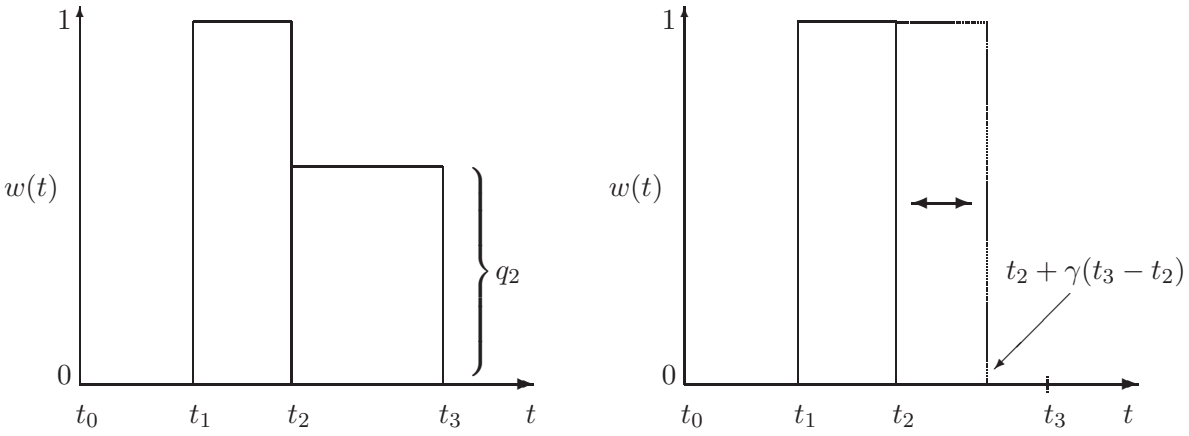

Figure 5.6: The main idea of an adaptive control grid. By inserting an additional time point $t_2 + \gamma(t_3 - t_2)$ where $w(\cdot)$ may change its value, the noninteger control $0 < q_2 < 1$ is transformed to two binary controls $\in \{0, 1\}$ and the optimal objective value is reduced.

that enforces continuity of the constant control $\boldsymbol{w}(\cdot)$ on $[t_i, t_{i+1}]$.

To show that uniform convergence towards trajectory $\mathcal{T}^*$ is possible, we used an equidistant control parameterization with an increasing number $N \approx n_{\text{ms}}$ of intervals in section 4.2. For practical purposes this is not a good approach for two reasons. First, information from the previous solution cannot be reused directly as the time points change in every iteration. Second, we lose computational efficiency as the control discretization grid may be too fine in regions where it is not necessary, e.g., where the control is at its upper bound for a considerable time interval.

Let us consider two control discretization grids $\mathcal{G}^k$ and $\mathcal{G}^{k+1}$. If we keep all time points when changing the grid $\mathcal{G}^k$ to a finer grid $\mathcal{G}^{k+1}$, i.e. $\mathcal{G}^k \subseteq \mathcal{G}^{k+1}$, and if we insert time points only in intervals $[t_i^k, t_{i+1}^k]$ if $0 < \tilde{q}_i^k < 1$, where $\tilde{\boldsymbol{q}}^{\boldsymbol{k}}$ is an optimal solution of the relaxed problem with control discretization grid $\mathcal{G}^k$, both drawbacks are avoided.

In the following we will use the assumptions

$$\tilde{q}_i = 0 \;\Rightarrow\; w^*(t) = 0 \text{ almost everywhere in } [t_i, t_{i+1}] \tag{5.8a}$$

$$\tilde{q}_i = 1 \;\Rightarrow\; w^*(t) = 1 \text{ almost everywhere in } [t_i, t_{i+1}] \tag{5.8b}$$

$$0 < \tilde{q}_i < 1 \;\Rightarrow\; \text{neither } w^*(t) = 0 \text{ a.e. nor } w^*(t) = 1 \text{ a.e. in } [t_i, t_{i+1}] \tag{5.8c}$$

that correlate the value of $\tilde{\boldsymbol{q}}$ to the optimal trajectory $\mathcal{T}^*$. This allows us to formulate an algorithm to determine an estimation for the objective function value corresponding to $\mathcal{T}^*$.

Algorithm 5.1 (Estimation of Φ^*)

1. Set $k := 0$. Choose an initial control discretization grid $\mathcal{G}^0$.
2. Solve the relaxed optimization problem for the control discretization grid $\mathcal{G}^k$. Obtain objective function value Φ^k.
3. Set
$$\mathcal{G}^{k+1} := \mathcal{G}^k \cup \left\{ \frac{t_i^k + t_{i+1}^k}{2} : \quad 0 < \tilde{q}_i < 1, \quad i = 0 \dots n_{\mathrm{ms}}^k - 1 \right\}.$$
4. Increment k. Extrapolate the values $(2^{-k}, \Phi^k)$ to obtain $(0, \Phi^*)$.
5. If $\Phi^k \approx \Phi^*$ set $n_{\mathrm{ext}} = k$, STOP.
6. Go to step 2.

For a description of extrapolation see a standard textbook on numerical analysis, e.g., Stoer & Bulirsch (1992). While bisection is a good choice for an extrapolation, the grids that are created in algorithm 5.1 are not necessarily suited for a bang–bang solution. It remains to answer the question how many time points are to be inserted in an interval $[t_i^k, t_{i+1}^k]$ and *where* to insert them. This answer depends very much on the structure of $\mathcal{T}^*$, more precisely on the question whether $\mathcal{T}^*$ contains arcs with singular resp. bang–bang controls. If $w^*(\cdot) \in \mathcal{T}^*$ contains bang–bang arcs in the interval $[t_i^k, t_{i+1}^k]$, the points where the switchings take place are the optimal choice to insert the new time points. But if $\mathcal{T}^*$ is a chattering or singular solution, there may be infinitely many switching points. As the structure of $\mathcal{T}^*$ is furthermore a priori unknown in direct methods, we consider a homotopy $\{\mathcal{G}^k\}$ and only insert one or two additional time points per interval in each iteration. The solution on grid $\mathcal{G}^{k+1}$ is then used to determine grid $\mathcal{G}^{k+2}$ and so on until a stopping criterion is fulfilled. Let us first consider a single control $w(\cdot)$ with value $0 < \tilde{q}_i < 1$ on an interval $[t_i, t_{i+1}]$, see the left diagram of figure 5.6. If, as in the figure, $\tilde{q}_{i-1} = 1$ and $\tilde{q}_{i+1} = 0$, it is possible that $\mathcal{T}^*$ consists of two bang–bang arcs on $[t_{i-1}, t_{i+2}]$ with the switching point

$$\tau = t_i + \gamma(t_{i+1} - t_i), \quad 0 < \gamma < 1 \tag{5.9}$$

somewhere in the interval $[t_i, t_{i+1}]$. To determine γ, we write

$$\boldsymbol{f}(w) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{z}(t), w(t), \boldsymbol{u}(t), \boldsymbol{p}).$$

We would like to have

$$\int_{t_i}^{t_{i+1}} \boldsymbol{f}(\tilde{q}_i)\, \mathrm{d}t = \int_{t_i}^{\tau} \boldsymbol{f}(1)\, \mathrm{d}t + \int_{\tau}^{t_{i+1}} \boldsymbol{f}(0)\, \mathrm{d}t.$$

on $[t_i, t_{i+1}]$, compare the right diagram of figure 5.6. A first order approximation, which is exact for linear systems, yields

$$\int_{t_i}^{t_{i+1}} \boldsymbol{f}(0) + \boldsymbol{f_w}\, \tilde{q}_i \, \mathrm{d}t = \int_{t_i}^{\tau} \boldsymbol{f}(0) + \boldsymbol{f_w}\, 1 \, \mathrm{d}t + \int_{\tau}^{t_{i+1}} \boldsymbol{f}(0) \, \mathrm{d}t$$

which is equivalent to

$$\tilde{q}_i \int_{t_i}^{t_{i+1}} \boldsymbol{f_w} \, \mathrm{d}t = \int_{t_i}^{\tau} \boldsymbol{f_w} \, \mathrm{d}t. \tag{5.10}$$

τ can thus be determined by integration of $\boldsymbol{f_w}$. For our purposes it turned out that a further simplification yields good results. If we assume $\boldsymbol{f_w} \approx const.$ on $[t_i, t_{i+1}]$ for small $t_{i+1} - t_i$, we obtain an estimate

$$\gamma \approx \tilde{q}_i \tag{5.11}$$

for τ from (5.10) that can be readily inserted without any additional calculations. This is the motivation for a choice of γ based on an estimated $1-0$ structure. If we assume that the structure of $\mathcal{T}^*$ is first 0 and then 1, (5.11) becomes

$$\gamma \approx 1 - \tilde{q}_i. \tag{5.12}$$

For a structure $0-1-0$ the integral equality (5.10) reads as

$$\tilde{q}_i \int_{t_i}^{t_{i+1}} \boldsymbol{f_w} \, \mathrm{d}t = \int_{\tau_1}^{\tau_2} \boldsymbol{f_w} \, \mathrm{d}t$$

and with $\tau_1 - t_i = t_{i+1} - \tau_2$ we have $\gamma_1 = \frac{1-\tilde{q}_i}{2}$ and $\gamma_2 = \tilde{q}_i + \frac{1-\tilde{q}_i}{2}$.
If the structure contains multiple bang–bang arcs or at least one singular arc in $[t_i, t_{i+1}]$, we cannot a priori estimate which location will yield the optimal improvement in the objective value. For these cases we have to rely on the bisection effect. Led by these considerations, we propose different *adaptive modes* to insert time points τ into a control discretization grid $\mathcal{G}^k$. For each interval $[t_i^k, t_i^{k+1}]$ with control $\tilde{q}_i$ we proceed in one of the following modes.

- Adaptive mode 1 (Bisection)

  If $\tilde{q}_i \notin \{0, 1\}$, insert one additional point

  $$\tau = t_i^k + \gamma(t_{i+1}^k - t_i^k) \tag{5.13}$$

  with $\gamma = 0.5$ into $\mathcal{G}^{k+1}$.

- Adaptive mode 2 (Bang–bang mode)

  Again we insert one point τ. The location depends on $\tilde{q}_{i-1}$, $\tilde{q}_i$ and $\tilde{q}_{i+1}$. If $\tilde{q}_i \notin \{0, 1\}$, insert the additional point

  $$\tau = t_i^k + \gamma(t_{i+1}^k - t_i^k) \tag{5.14}$$

into $\mathcal{G}^{k+1}$. γ is determined by

$$\tau = \begin{cases} \tilde{q}_i & \text{if } \tilde{q}_{i-1} > \tilde{q}_{i+1} \\ 1 - \tilde{q}_i & \text{if } \tilde{q}_{i-1} < \tilde{q}_{i+1} \\ 0.5 & \text{if } \tilde{q}_{i-1} = \tilde{q}_{i+1} \end{cases} . \tag{5.15}$$

- Adaptive mode 3 (Structure determining mode)
  If $\tilde{q}_i \notin \{0, 1\}$ we insert two points $\tau_1, \tau_2 \in [t_i^k, t_{i+1}^k]$, depending on $\tilde{q}_i$,
  $$\tau_1 = t_i^k + \tilde{q}_i(t_{i+1}^k - t_i^k), \;\; \tau_2 = t_i^k + (1 - \tilde{q}_i)(t_{i+1}^k - t_i^k) \tag{5.16}$$
  into $\mathcal{G}^{k+1}$.
- Adaptive mode 4 (Pulse)
  If $\tilde{q}_i \notin \{0, 1\}$ we insert two points $\tau_1, \tau_2 \in [t_i^k, t_{i+1}^k]$,
  $$\tau_1 = t_i^k + \frac{1 - \tilde{q}_i}{2}(t_{i+1}^k - t_i^k), \;\; \tau_2 = t_i^k + \frac{1 + \tilde{q}_i}{2}(t_{i+1}^k - t_i^k) \tag{5.17}$$
  into $\mathcal{G}^{k+1}$.

The choices of τ in the adaptive modes are based on the considerations above and guesses for the switching structure in the interval $[t_i^k, t_i^{k+1}]$. While adaptive mode 1 is a simple bisection of the control discretization grid, adaptive mode 2 aims at inserting time points that are close to possible switching times of a simple bang–bang solution. It is assumed that if the control value $\tilde{q}_{i-1}$ on the previous interval is higher than the value on the following one, $\tilde{q}_{i+1}$, we have a $1 - 0$ structure on $[t_i^k, t_i^{k+1}]$. We derived (5.11) for this structure, compare also figure 5.6. If $\tilde{q}_{i-1} < \tilde{q}_{i+1}$, e.g., $\tilde{q}_{i-1} = 0, \tilde{q}_{i+1} = 1$, adaptive mode 2 guesses a $0 - 1$ structure of $\mathcal{T}^*$ on $[t_i^k, t_i^{k+1}]$ and takes (5.12) to determine τ. Figure 5.7 illustrates this idea. If the values are identical, adaptive mode 2 divides into two equidistant intervals as in mode 1.
The assumptions taken in adaptive mode 2 may be wrong for certain trajectories $\mathcal{T}^*$. To handle the case where the structure is $0 - 1$ although $\tilde{q}_{i-1} > \tilde{q}_{i+1}$ resp. $1 - 0$ although $\tilde{q}_{i-1} < \tilde{q}_{i+1}$, adaptive mode 3 enters *both* points (5.11) and (5.12) into $\mathcal{G}^{k+1}$. The price for the flexibility is of course an additional, possibly redundant time point. Adaptive mode 4 allows "pulses" in the middle of interval $[t_i^k, t_{i+1}^k]$, see the illustration in picture 5.8.
For the case $n_w > 1$ we have to extend the algorithms presented. There are two possible ways to determine adequate τ's for an interval $[t_i^k, t_{i+1}^k]$, if several $\tilde{q}_{j,i} \notin \{0, 1\}$. The first would be to add several τ's by applying one of the adaptive modes presented above to each control function. The second is to apply it only to a control function $w_{j^*}(\cdot)$, if

$$\min(\tilde{q}_{j^*,i}, 1 - \tilde{q}_{j^*,i}) = \max_j \min(\tilde{q}_{j,i}, 1 - \tilde{q}_{j,i}),$$

i.e., it has the maximum integer violation of all j. As the introduction of additional time points is part of an iterative procedure, the other functions are treated in future iterations. The latter approach is the one we prefer for our method.

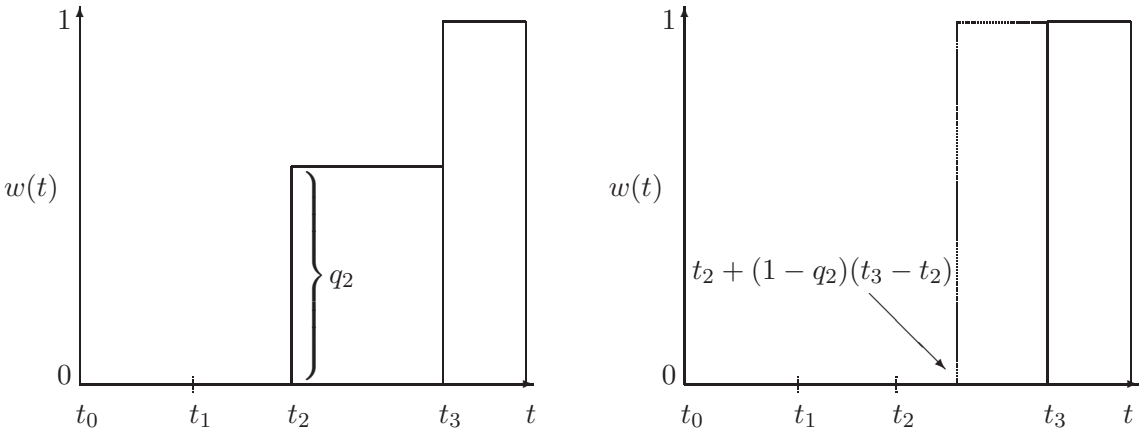

Figure 5.7: Adaptive mode 2, case where $0 = \tilde{q}_{i-1} < \tilde{q}_{i+1} = 1$. Time point τ is chosen as (5.11) assuming a $0-1$ structure as depicted in the right hand side.

Remark 5.2 *The choice of the time points is based upon assumptions and may yield an unnecessary high number of iterations with respect to an optimal choice. As this choice depends on a combinatorial choice of possible switching structures in each interval, one may as well apply deterministic methods as Branch and Bound or enumeration to find the switching times. This may be topic of a future research study and is beyond the scope of this thesis for which the presented adaptive modes worked sufficiently well.*

Remark 5.3 *In an iterative procedure there will be more and more redundant time points. By redundant we mean time points that are very close to their neighbors, not needed for a beneficial behavior of the multiple shooting method and in which no change in the binary control functions occurs. In practice, one may detect such points and remove them from the problem formulation to improve computational efficiency. From a theoretical point of view we still need these points to guarantee convergence of our algorithm.*

Remark 5.4 *The adaptive scheme is built upon a homotopy, as the optimization problem with control discretization grid $\mathcal{G}^{k+1}$ is initialized with values of the optimal trajectory of grid $\mathcal{G}^k$. For state variables and continuous controls on newly inserted time points we use integration resp. interpolation. For the relaxed binary control functions $\boldsymbol{w}(\cdot)$ we either set the values on each subinterval of $[t_i, t_{i+1}]$ to $\tilde{\boldsymbol{q}}_{\boldsymbol{i}}$ and try to ensure binary admissibility by an outer loop, or we fix the values to an assumed structure coherent with the choice of the τ's. If the latter approach is chosen, we fix lower and upper bounds of the control variables on the corresponding intervals both to* 0 *or* 1*, but keep the value $\tilde{\boldsymbol{q}}_{\boldsymbol{i}}$ for the initialization of the optimization problem to stay inside the convergence region of the SQP method. This idea is closely related to* initial value embedding*, compare Diehl (2001).*

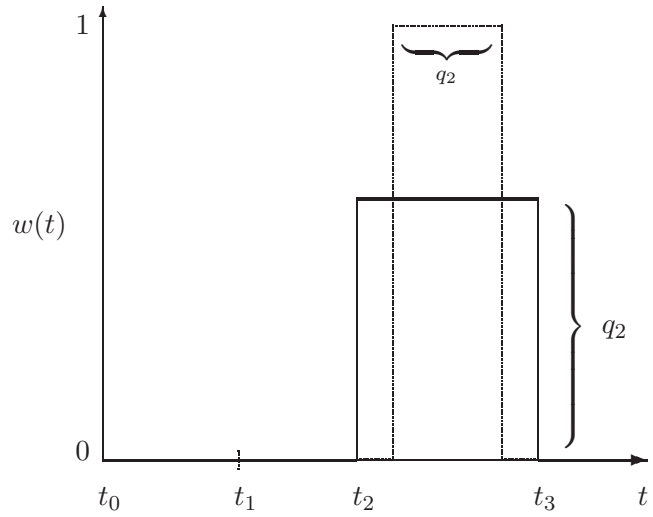

Figure 5.8: Adaptive mode 4. The bang–bang control with $0-1-0$ structure and switching time points $\tau_1 = t_i^k + \frac{1-\tilde{q}_i}{2}(t_{i+1}^k - t_i^k)$ and $\tau_2 = t_i^k + \frac{1+\tilde{q}_i}{2}(t_{i+1}^k - t_i^k)$ is depicted with a dotted line.

5.4 Penalty term homotopy

We consider an optimal control problem $P^k = P(\boldsymbol{\beta^k}), k \in \mathbb{N}_0$ defined as in (4.21) and dependent on the penalty parameter vector $\boldsymbol{\beta^k}$. Remember that problems of the form (4.21) are identical to the relaxed versions of the mixed–integer optimal control problem (5.1), but also penalize all measurable violations of the integer requirements with a concave quadratic penalty term.
The proposed penalty term homotopy consists of solving a series of continuous optimal control problems $\{P(\boldsymbol{\beta^k})\}, k \in \mathbb{N}_0$ with relaxed $w(t)$. Problem P^{k+1} is initialized with the solution of P^k and $\beta_i^0 = 0$ so that P^0 is the relaxed version of problem (5.1). The penalty parameters β_i^k are raised monotonically until all $w_j(\cdot) \in \{0,1\}$, $j = 1 \dots n_w$, or a stopping criterion is fulfilled.

Algorithm 5.2 (Penalty term homotopy)

1. Set $k := 0$ and $\boldsymbol{\beta^k} = \mathbf{0}$.
2. Solve the relaxed optimization with penalty parameter vector $\boldsymbol{\beta^k}$.
3. Increment k, choose $\boldsymbol{\beta^k} \geq \boldsymbol{\beta^{k-1}}$.
4. If solution integer or stopping criterion fulfilled STOP else go to step 2.

As shown in section 4.3 the solution of problem $P(\boldsymbol{\beta})$ will be integer, if β is chosen sufficiently large. Still, for a given grid the optimal solution is not necessarily admissible, as constraints $\boldsymbol{c}(\cdot)$ may cut off an integer solution. If it is feasible, it may have a very bad objective value with respect to $\mathcal{T}^*$. Therefore we stop algorithm 5.2, if

one of the following stopping criteria is fulfilled:

Definition 5.5 (Stopping criteria of penalty term homotopy)
The stopping criteria of algorithm 5.2 are given by

- *The objective value Φ^k of the optimal solution $\boldsymbol{w^k}(\cdot)$ of problem P^k is much worse than the solution of problem P^{k-1}, when the penalty term is neglected:*

$$\begin{aligned} \Phi^k \quad & -\sum_{i=1}^{n_w} \beta_i^k \int_{t_0}^{t_f} w_i^k(t)\,(1-w_i^k(t))\,\mathrm{d}t \\ \gg \quad \Phi^{k-1} \quad & -\sum_{i=1}^{n_w} \beta_i^{k-1} \int_{t_0}^{t_f} w_i^{k-1}(t)\,(1-w_i^{k-1}(t))\,\mathrm{d}t. \end{aligned} \tag{5.18}$$

  As one reasonable choice for "$\gg$" we choose the tolerance given by a user for the gap between integer and relaxed solution, ε.

- *For more than n_{stuck} iterations the optimal trajectory $\mathcal{T}^k$ has not moved further than a certain tolerance.*

- *The maximum number of iterations has been reached, $k \geq n_{\text{pen}}$.*

If one of the first two criteria is met, the control discretization grid $\mathcal{G}$ is probably too coarse. When we use the penalty term homotopy as a part of an outer algorithm, it makes sense to stop the penalty homotopy to first refine the grid, see section 5.5.
If the first criterion is met, the optimal trajectory of the secondlast problem solved, P^{k-1}, should be used for a further refinement of the control grid. In practice this requires the storage of the secondlast optimal trajectory.
The third criterion is used to guarantee finiteness of the algorithm.
For the presented algorithm a good choice for the β_i^k is crucial for the behavior of the method. A too fast increase in the penalty parameters results in less accuracy and is getting closer to simple rounding, while a slow increase leads to an augmentation in the number of QPs that have to be solved. We choose β_i^k according to

$$\beta_i^0 = 0, \quad \beta_i^k = \beta_{\text{init}} \cdot \beta_{\text{inc}}^{k-1} \geq 0 \quad k = 1, \ldots, n_k. \tag{5.19}$$

For a system with all variables scaled to 1.0 we made good experiences with a choice of $\beta_{\text{init}} \approx 10^{-4}$ and $\beta_{\text{inc}} \approx 2$.

Remark 5.6 *Another possibility to penalize the nonintegrality is proposed by Stein* et al. *(2004), compare section 3.1. The authors introduce additional inequalities, prohibiting nonintegral domains of the optimization space. For our purposes we prefer to penalize nonintegrality instead of forbidding it, as a nonintegral solution will be used to further refine the control grid discretization.*

Figure 5.9 shows an example, the solution of the relaxed fishing problem on a fixed grid that will be investigated in more detail in section 6.5.

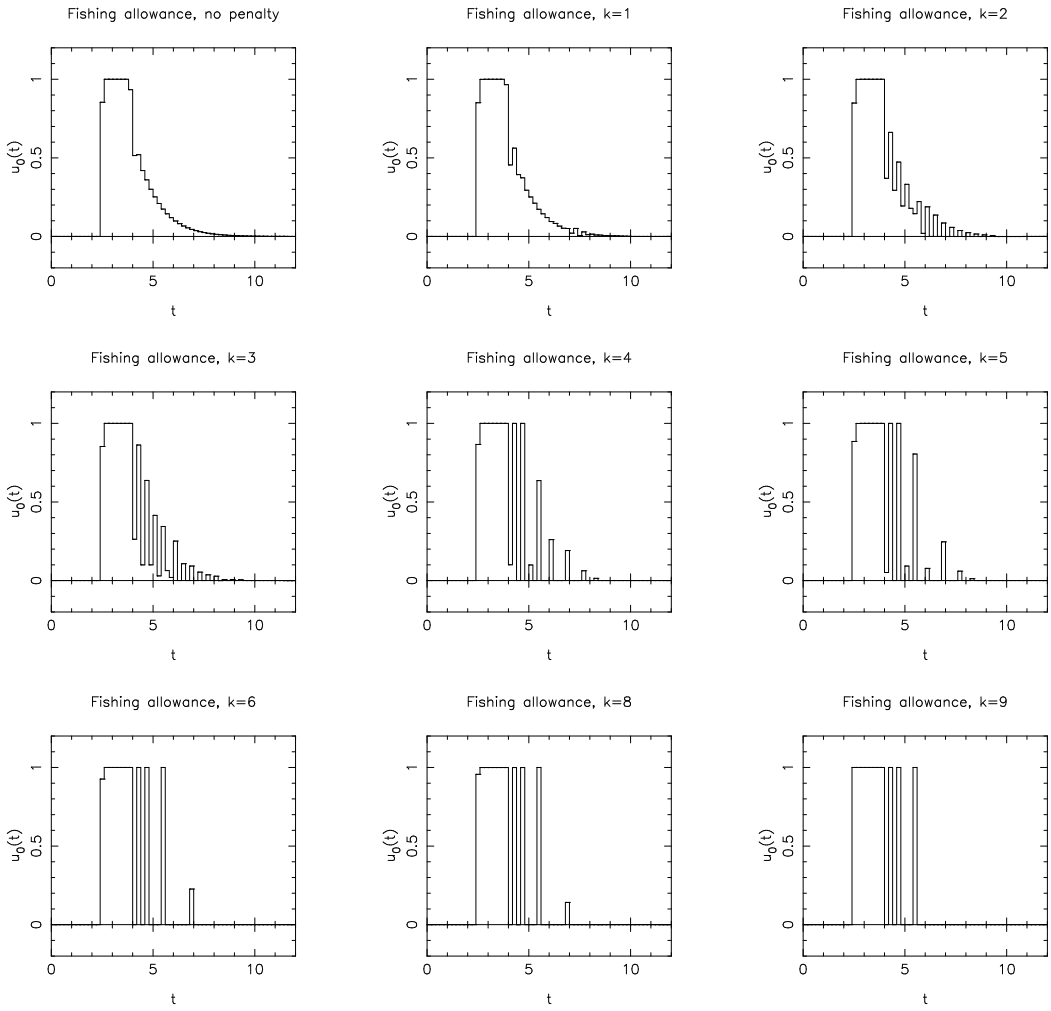

Figure 5.9: The fishing problem of section 6.5 as an example for the penalty homotopy applied on a fixed, equidistant control discretization grid with $n_{\mathrm{ms}} = 60$. The penalty parameter β is chosen as $\beta = 10^{-4}\ 2^k$. The top left plot shows the solution to the unaltered problem. The other plots show, from top middle to bottom right, the solutions for augmented k until binary admissibility is achieved at $k = 9$, neglecting $k = 7$ for lack of space. The objective value of the relaxed problem is $\Phi^0 = 1.34465$. It rises up to $\Phi^9 = 1.34996$. The largest gap is between the solutions for $k = 8$ and $k = 9$, as $\Phi^8 = 1.34693$, penalty terms neglected.

5.5 MS MINTOC

In this section we will bring together the concepts presented so far in this thesis and formulate our novel algorithm to solve mixed–integer optimal control problems. We will call this algorithm *multiple shooting based mixed–integer optimal control algorithm*, in short *MS MINTOC* . The algorithm gets a user specified tolerance $\varepsilon > 0$ as problem specific input. ε determines how large the gap between relaxed and binary solution may be. Furthermore an initial control discretization grid $\mathcal{G}^0$ with n_{ms} stages is supplied.

Algorithm 5.3 (*MS MINTOC*)

1. Convexify problem (5.1) as described in section 4.1.
2. Relax this problem to $\tilde{\boldsymbol{w}}(\cdot) \in [0,1]^{n_{\tilde{w}}}$.
3. Solve this problem for control discretization $\mathcal{G}^0$, obtain the grid–dependent optimal value $\Phi^{\text{RL}}_{\mathcal{G}^0}$.
4. Apply algorithm 5.1 for n_{ext} steps and obtain $\Phi^{\text{RL}}_{\mathcal{G}^{n_{\text{ext}}}}$ as the objective function value on the finest grid $\mathcal{G}^{n_{\text{ext}}}$. Set $\Phi^{\text{RL}} = \Phi^{\text{RL}}_{\mathcal{G}^{n_{\text{ext}}}}$ to this upper bound on Φ^*.
5. If the optimal trajectory on $\mathcal{G}^{n_{\text{ext}}}$ is binary admissible then STOP.
6. Apply a rounding or penalty heuristics, see section 5.1. If the trajectory is binary admissible, obtain upper bound Φ^{ROU}. If $\Phi^{ROU} < \Phi^{\text{RL}} + \varepsilon$ then STOP.
7. Use switching time optimization, see section 5.2, initialized with the rounded solution of the previous step. If the obtained trajectory is binary admissible, obtain upper bound Φ^{STO}. If $\Phi^{STO} < \Phi^{\text{RL}} + \varepsilon$ then STOP.
8. Reconsider the optimal trajectory $\mathcal{T} = \mathcal{T}^0$ of the relaxed, convexified problem with the initial control discretization $\mathcal{G}^0$, set the counter $k = 0$.
9. REPEAT
   (a) Refine the control grid $\mathcal{G}^k$ by a method described in section 5.3, based on the control values of trajectory $\mathcal{T}$.
   (b) Apply a penalty term homotopy given by algorithm 5.2, see section 5.4. If $\Phi \leq \Phi^{\text{RL}} + \varepsilon$, update trajectory $\mathcal{T} = \mathcal{T}^k$, else define $\mathcal{T}$ as the initial trajectory of the homotopy.
   (c) $k = k + 1$.
10. UNTIL ($\tilde{\boldsymbol{w}}(\cdot) \in \{0,1\}^{n_{\tilde{w}}}$ AND $\Phi \leq \Phi^{\text{RL}} + \varepsilon$)

In the first steps of the algorithm we try to obtain a binary admissible trajectory by heuristic methods. If this is not successful, we iterate in a loop on the control

problem with a stepwise refined control discretization grid $\mathcal{G}^k$. The trajectories obtained from the last control problem (resp. the secondlast, compare section 5.4) are used to initialize the following control problem and to set up the refined control grid $\mathcal{G}^{k+1}$. The penalty parameters β are reset to $\mathbf{0}$ in each iteration. This aims at avoiding local minima caused by the nonconvex concave penalty terms.
Algorithm 5.3 may end with different results. If no convergence can be achieved for any of the subproblems, different initializations or different algorithmic settings have to be tried. For example one might consider changing from a line search to a trust box or a watchdog technique. We do assume here that convergence up to a given tolerance can be achieved for the relaxed problem. All subproblems thereafter are part of a homotopy, they start thus with an admissible trajectory. This strategy aims at "staying" inside the convergence radius of Newton's method respectively the SQP algorithm 2.1.

Theorem 5.7 (Behavior of algorithm 5.3)
If

- *the relaxed control problem on grid $\mathcal{G}^0$ possesses an admissible optimal trajectory*
- *all considered problems can be solved precisely to global optimality in a finite number of iterations*
- *bisection is used to adapt the control grid on all intervals (independent of the values $\tilde{q}_i$'s)*

then for all $\varepsilon > 0$ algorithm 5.3 will terminate with a trajectory that is binary admissible and a corresponding objective value Φ such that

$$\Phi \leq \Phi^{\mathrm{RL}}_{\mathcal{G}^{n_{\mathrm{ext}}}} + \varepsilon$$

where $\Phi^{\mathrm{RL}}_{\mathcal{G}^{n_{\mathrm{ext}}}}$ is the objective value of the optimal trajectory for the relaxed problem with the grid $\mathcal{G}^{n_{\mathrm{ext}}}$ of the last iteration in the estimation of Φ^{RL}.

Proof. There are several possibilities where algorithm 5.3 may stop. The first one is that one of the optimal control problems that have to be solved is infeasible or no convergence can be achieved for its solution. As we assume that there is an admissible trajectory for the relaxed problem on the coarsest grid $\mathcal{G}^0$, this trajectory will be admissible for all other occurring problems, too, as all other control discretization grids are supersets of $\mathcal{G}^0$ and a modification of the objective function does not concern admissibility. In addition we assume that all considered problems can be solved to global optimality.
A second possibility is that the algorithm gets stuck in the inner loop 9.-10. and does not terminate at all. As shown in chapter 4 there exists a number N such that the exit condition 10. is fulfilled for all optimal bang–bang trajectories with a finer discretization than an equidistant control discretization with N time points. As a bisection of all intervals is performed, we have

$$t_{i+1} - t_i \leq \frac{t_f - t_0}{2^k}, \quad i = 0 \ldots n^k_{\mathrm{ms}}$$

where k is the iteration counter of the loop 9.-10. There exists a k such that the maximum distance of all $t_{i+1} - t_i < 1/N$. For this control grid we still have to find the optimal binary admissible solution. By adding a penalty term that is sufficiently large, the optimal binary admissible solution is the optimal solution for the relaxed problem on this control discretization grid and will therefore by assumption be found. This yields the wanted contradiction to the assumption condition 10. could rest unfulfilled in an infinite loop.
Therefore algorithm 5.3 must stop in either step 5., 6., 7. or 10. In all cases we find a trajectory which fulfills the integer requirement, is admissible and which has an objective value which is closer than the prescribed tolerance ε, completing the proof. ■

Theorem 5.7 needs some strong assumptions. While the first, the existence of an admissible optimal trajectory for the relaxed optimal control problem on a user specified grid is an absolute must before wanting to solve a mixed–integer problem, the second assumption does not hold for optimal control solvers looking for local minima. Typically the optimal control problems under considerations are nonconvex. By adding a concave penalty term the number of local minima may reach 2^{n_w}, compare figure 4.2. One possibility to overcome this problem is to use an optimal control solver that can handle nonconvex problems and determine a global solution in each step of the iteration. We will give an example in section 6.5 for a globally optimal solution on a fixed control grid. For all control problems we treated so far, the penalty term homotopy was sufficiently good. First of all, as we see in an example in section 6.2, the adaptivity can compensate the fact that a solution is only local. If it is chosen fine enough, even a local solution will satisfy the terminal condition. Another issue is related to the multiple shooting method. As already pointed out in section 2.4 and exemplified in appendix B.4, all–at–once approaches have advantages with respect to avoiding local minima. One important reason why we prefer the penalty term homotopy is that one can deduce from its solution the regions where the control discretization grid is too coarse. Applying, e.g., a Branch&Bound method would deliver the globally optimal integer solution, but in case the corresponding objective value was not good enough, one has no information on where to refine the grid. Still there is place for future research in this direction and the penalty term homotopy should be regarded as only one possible approach to solve the optimal control problem on a given grid. In fact, for the applications and case studies presented in this thesis, the rounding heuristics SUR-SOS1 showed to be a very good alternative if it is included in a grid refinement sequence.
Furthermore, as all calculations are performed on a computer, there are numerical issues as the machine accuracy that gives a lower bound on the distance of two switching points that have to be considered. For this reason the theoretical theorems do not hold for arbitrarily small ε. As stated in section 5.3 we will not only use bisection for practical problems but also try to get closer to supposed switching points by a nonequidistant partition of the intervals $[t_i, t_{i+1}]$. This might lead to a series of monotonically decreasing interval lengths that is bounded by $1/N$ from below. This

destroys the theoretical argument and it may cause numerical problems as extinction because time points are accumulated. Furthermore we rely on assumption (5.8) and only adapt the grid where the binary control functions of the optimal trajectory on the current control grid take noninteger values.
All these issues prevent the *MS MINTOC* algorithm from being used as a deterministic black–box algorithm. Tuning with respect to the initial grid, termination criteria, the adaptivity mode, penalty parameters and initial values is necessary to obtain the wanted results. On the other hand, this holds true for continuous control problems, as well.

5.6 Summary

In this chapter we presented our novel algorithm to solve mixed–integer optimal control problems. The algorithm is based on an interplay between the direct multiple shooting method, rigorous lower and upper bounds, the usage of heuristics, adaptivity of the control discretization grid and a penalty term homotopy to obtain a trajectory fulfilling integer requirements.
In section 5.1 several rounding strategies were presented, among them specialized ones that take into account the fact that some variables are connected as they discretize the same control function. Furthermore rounding strategies for the multi–dimensional case with special ordered set restrictions on the control functions were given. Rounding strategies yield trajectories that fulfill the integer requirements, but are typically not optimal and often not even admissible. Nevertheless rounding strategies may be applied successfully to obtain upper bounds in a Branch and Bound scheme, to get a first understanding of the behavior of a system or to yield initial values for the switching time optimization approach presented in section 5.2. This approach reformulates the optimal control problem as a multistage problem with fixed binary control function values. After an introduction of this approach we discussed its disadvantages and gave an illustrative example for the most important one, the introduction of additional nonconvexities. In appendix B.4 we will present an example with multiple local minima and show that the direct multiple shooting method may converge to the global minimum while direct single shooting converges to a local minimum with bad objective value, although the stage lengths as the only independent degrees of freedom in both methods are initialized with the same values. Our algorithm is based upon an adaptive refinement of the control discretization grid. In section 5.3 we motivated and presented algorithms to obtain an estimation of the objective value corresponding to the optimal trajectory for the infinite–dimensional control problem and to refine a grid such that, under certain assumptions, the optimal trajectory of the relaxed problem can be approximated with a trajectory that is binary admissible.
In section 5.4 we presented a penalty term homotopy that adds quadratic penalty terms to the control problem on a given control discretization grid. This heuristics is used to obtain integer values for the control discretization variables. Using a homotopy, we stay inside the convergence radius of the SQP method and we can

detect when and where the underlying grid is too coarse.
This is used in the *MS MINTOC* algorithm 5.3 presented in section 5.5. Making use of the knowledge obtained in chapter 4 that the optimal binary solution of the nonlinear optimal control problem has a corresponding optimal binary solution of a convexified control problem for which we get an attainable lower bound by solving its relaxation, we first determine this lower bound. We apply some heuristics, namely rounding and applying the switching time optimization, to get upper bounds and compare them with the lower bound. If the result is not satisfactory, we iterate on a refinement of the control grid and an application of the penalty term homotopy, until we end up with a binary admissible trajectory with objective value that is closer than a prescribed tolerance to the attainable optimum. We proved that under certain theoretic assumptions algorithm 5.3 will terminate with such a solution.

Chapter 6

Case studies

Our algorithm is based on direct multiple shooting and needs therefore no a priori analysis of the structure of the optimal control problem to be solved. In this chapter we will apply it to problems for which this structure is already known to show the applicability of our method and to investigate typical behavior of the algorithm.
Five case studies will be presented. Case studies are applications of the *MS MINTOC* algorithm presented on page 104, to optimal control problems for which the structure of the optimal trajectory $\mathcal{T}^*$ for the infinite–dimensional relaxed optimal control problem is known. All problems presented contain differential states and a single binary control function only to focus on the structure of this function $w(\cdot)$. We will investigate the behavior of the algorithm on different kinds of trajectories,in particular we will treat one example with bang–bang structure, section 6.1, two examples where different kinds of chattering controls occur in the optimal trajectory, sections 6.2 and 6.3, and one example containing a singular arc, section 6.4.
For the last example we will furthermore investigate the case where the switching times are fixed and algorithm *MS MINTOC* can not be applied. In section 6.5 we will present a Branch and Bound method for optimal control problems which is an extension to the Branch and Bound method for mixed–integer nonlinear problems, compare section 3.2.
All calculations of this thesis were done on a PC with AMD Athlon 3000+ processor under Linux. If not otherwise stated, we used a trust box technique, a finite difference approximation of the Hessian, the DAE solver DAESOL with the BDF method for the solution of the DAEs and the generation of derivatives and accuracies of 10^{-6} for the KKT condition and 10^{-7} for the integration. The times that will be given for the solution of optimal control problems include graphics, initializations and so on. We will furthermore state the number of QPs that had to be solved as an indication of the number of iterations in the SQP algorithm.

6.1 F–8 aircraft

First we will consider an example with a bang–bang structure in the optimal control function $w^*(\cdot)$. Kaya & Noakes (2003) discuss the time–optimal control problem of

an F–8 aircraft. The model they consider goes back to Garrard & Jordan (1977) and is given by

$$\min_{\boldsymbol{x},w,T} T \tag{6.1a}$$

subject to the ODE

$$\begin{aligned}
\dot{x}_0 &= -0.877\, x_0 + x_2 - 0.088\, x_0\, x_2 + 0.47\, x_0^2 - 0.019\, x_1^2 - x_0^2\, x_2 \\
&\quad +3.846\, x_0^3 - 0.215\, w + 0.28\, x_0^2\, w + 0.47\, x_0\, w^2 + 0.63\, w^3 && \text{(6.1b)} \\
\dot{x}_1 &= = x_2 && \text{(6.1c)} \\
\dot{x}_2 &= -4.208\, x_0 - 0.396\, x_2 - 0.47\, x_0^2 - 3.564\, x_0^3 \\
&\quad -20.967\, w + 6.265\, x_0^2\, w + 46\, x_0\, w^2 + 61.4\, w^3 && \text{(6.1d)}
\end{aligned}$$

with initial condition

$$\boldsymbol{x}(0) = (0.4655, 0, 0)^T, \tag{6.1e}$$

terminal constraint

$$\boldsymbol{x}(T) = (0, 0, 0)^T, \tag{6.1f}$$

and a restriction of the control to values in

$$w(t) \in \{-0.05236, 0.05236\}, \quad t \in [0, T]. \tag{6.1g}$$

We write $x_i = x_i(t)$ for the three differential states of the system. x_0 is the angle of attack in radians, x_1 is the pitch angle, x_2 is the pitch rate in rad/s, and the control function $w = w(t)$ is the tail deflection angle in radians. See Kaya & Noakes (2003) for further references and details for this problem.

Applying algorithm 5.3, we first convexify problem (6.1). We introduce a control function $\tilde{w}(\cdot)$ at the place of $w(\cdot)$, replace (6.1g) by

$$\tilde{w}(\cdot) \in \{0, 1\} \tag{6.2a}$$

and (6.1b-6.1d) by

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, -0.05236)\,(1 - \tilde{w}) + \boldsymbol{f}(\boldsymbol{x}, 0.05236)\,\tilde{w} \tag{6.2b}$$

as described in section 4.1.

We solve the obtained linear optimal control problem with a relaxation of (6.2a), in short problem (RL), on an equidistant grid $\mathcal{G}^0$ with $n_{\text{ms}} = 25$. We require the terminal condition to be fulfilled to an accuracy of 10^{-5} in each component. The optimal trajectory for this problem is shown in figure 6.1, the corresponding objective value in seconds is $\Phi^{\text{RL}}_{\mathcal{G}^0} = T = 5.86255$.

We now apply algorithm 5.1 to get an estimate for the objective function value of $\mathcal{T}^*$. We get a series of control grids $\mathcal{G}^k$ with corresponding trajectories and objective function values $\Phi^{\text{RL}}_{\mathcal{G}^k}$. The grid $\mathcal{G}^{k+1}$ is obtained from grid $\mathcal{G}^k$ by bisection (adaptive mode 1). The optimal binary control functions $\tilde{w}(\cdot)$ are plotted in figure 6.2 for

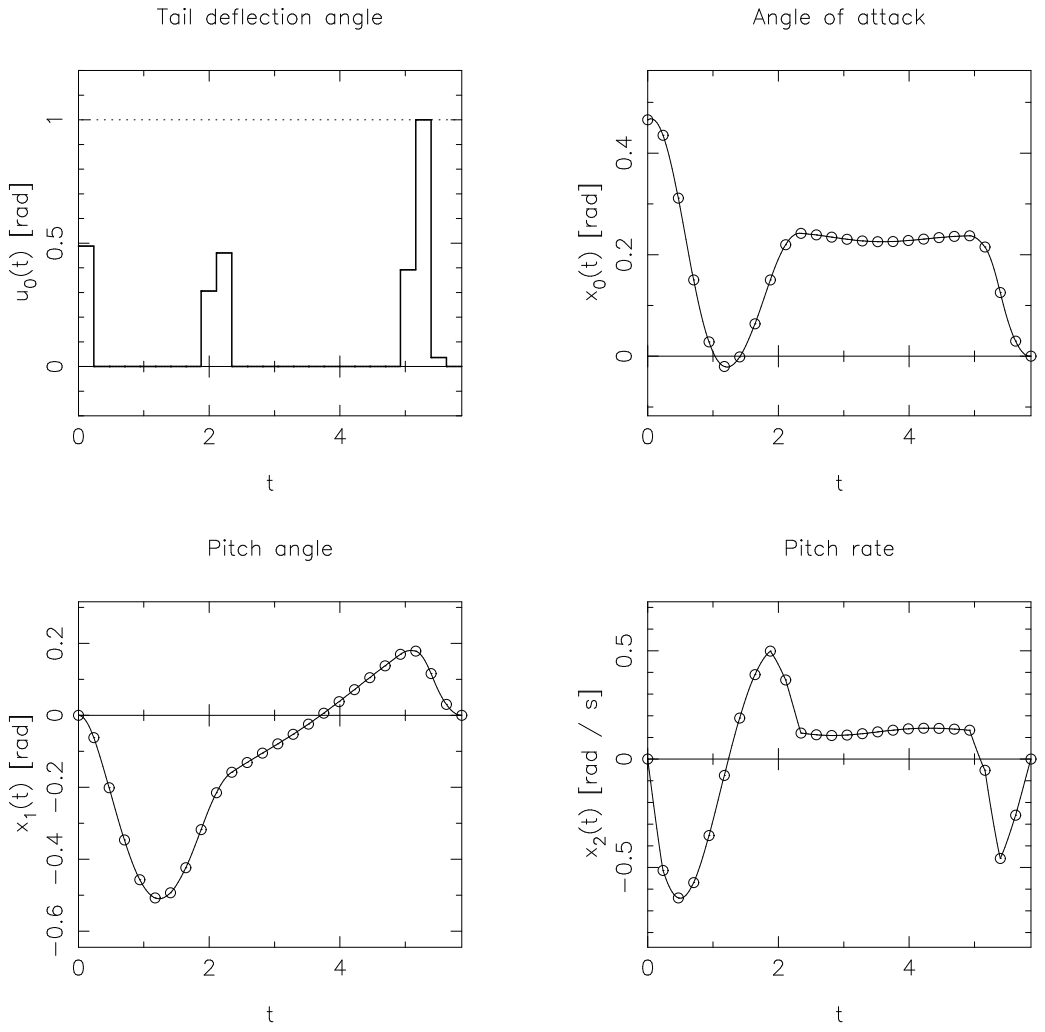

Figure 6.1: Trajectory of optimal solution of the relaxed F–8 aircraft problem on an equidistant control discretization grid with $n_{\mathrm{ms}} = 25$.

$n_{\mathrm{ext}} = 6$. The objective values are

$$\Phi^{\mathrm{RL}}_{\mathcal{G}^0} = 5.86255, \tag{6.3a}$$
$$\Phi^{\mathrm{RL}}_{\mathcal{G}^1} = 5.73275, \tag{6.3b}$$
$$\Phi^{\mathrm{RL}}_{\mathcal{G}^2} = 5.73202, \tag{6.3c}$$
$$\Phi^{\mathrm{RL}}_{\mathcal{G}^3} = 5.73174, \tag{6.3d}$$
$$\Phi^{\mathrm{RL}}_{\mathcal{G}^4} = 5.73157, \tag{6.3e}$$
$$\Phi^{\mathrm{RL}}_{\mathcal{G}^5} = 5.73147. \tag{6.3f}$$

The objective values yield a monotonically falling series. The convergence is very slow, still. If we do not a bisection, but apply adaptive mode 2, compare section 5.3,

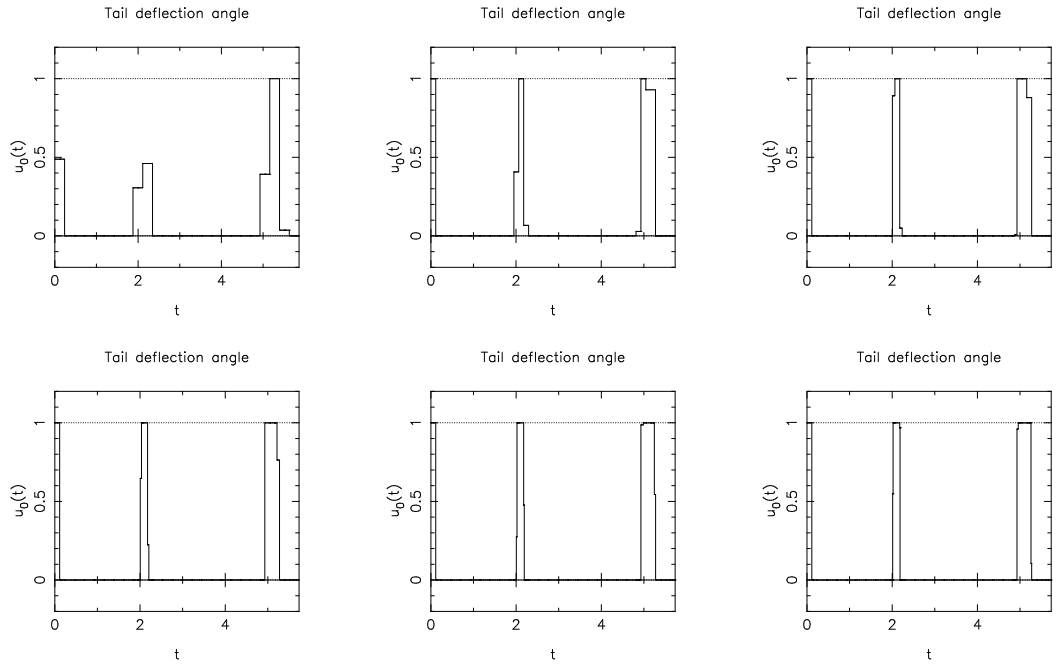

Figure 6.2: Optimal control functions $\tilde{w}(\cdot)$ of the relaxed F–8 aircraft problem for control grids $\mathcal{G}^0$ (top left) to $\mathcal{G}^5$ (bottom right).

we obtain the values

$$\Phi^{\mathrm{RL}}_{\mathcal{G}^0} = 5.86255, \tag{6.4a}$$
$$\Phi^{\mathrm{RL}}_{\mathcal{G}^1} = 5.73187, \tag{6.4b}$$
$$\Phi^{\mathrm{RL}}_{\mathcal{G}^2} = 5.73049, \tag{6.4c}$$
$$\Phi^{\mathrm{RL}}_{\mathcal{G}^3} = 5.73046, \tag{6.4d}$$
$$\Phi^{\mathrm{RL}}_{\mathcal{G}^4} = 5.73046, \tag{6.4e}$$
$$\Phi^{\mathrm{RL}}_{\mathcal{G}^5} = 5.73046. \tag{6.4f}$$

For the last three grid refinements we did not get any progress in the objective function up to 10^{-5} and the objective function looks almost binary admissible at a first glance, see the left plot in figure 6.3. But if we apply a rounding strategy to this solution, we obtain[1]

$$\tilde{w} = \mathcal{S}(1; 0.11200, 1.90169, 0.16773, 2.74897, 0.33019, 0.46988), \tag{6.5}$$

a solution which fulfills terminal constraint (6.1f) only with an accuracy of 10^{-4} instead of 10^{-5} and is therefore not admissible. Applying the switching time optimization algorithm initialized with (6.5), we get

$$\tilde{w} = \mathcal{S}(1; 0.10235, 1.92812, 0.16645, 2.73071, 0.32994, 0.47107) \tag{6.6}$$

which is admissible and has an objective function value of $\Phi^{\mathrm{STO}} = 5.72864$ seconds. This control function is plotted on the right hand side of figure 6.3. The optimal

[1] compare definition 5.1 of $\mathcal{S}$

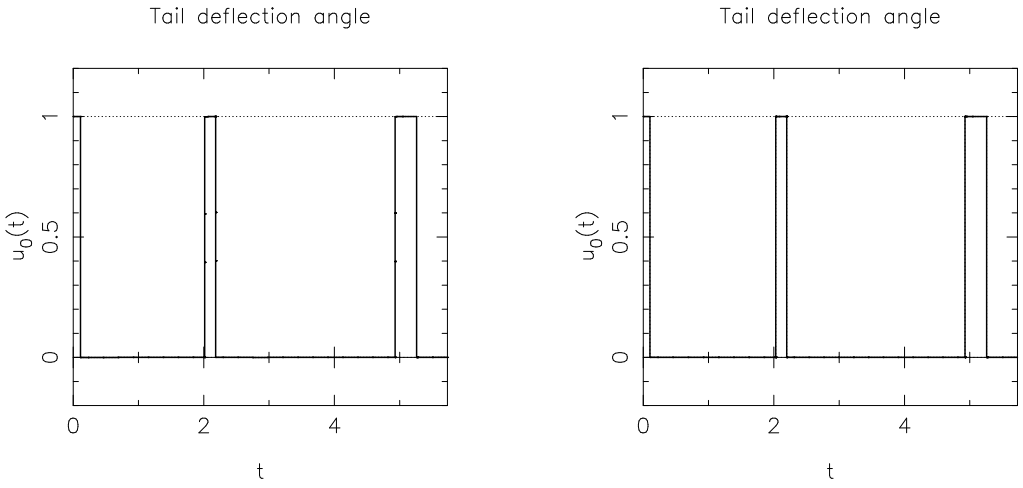

Figure 6.3: Left: Optimal control function $\tilde{w}(\cdot)$ of the relaxed F–8 aircraft problem for control grid $\mathcal{G}^5$ (adaptive mode 2). The solution is not completely integer and rounding leads to a violation of the terminal constraint. Right: optimal control function (6.6) obtained by a switching time optimization, initialized with the rounded solution of the left plot.

trajectory is slightly better than the one given in Kaya & Noakes (2003),

$$\tilde{w}^{\text{Kaya}} = \mathcal{S}(1; 0.10292, 1.92793, 0.16687, 2.74338, 0.32992, 0.47116) \tag{6.7}$$

with $\Phi^{\text{Kaya}} = 5.74217$. Solution (6.7) violates furthermore the terminal constraint (6.1f) by more than 10^{-3}. This poor quality is probably due to an inaccurate integration of Kaya & Noakes (2003) as they state

> *The efficiency of TOS can be increased further by implementing . . . , using a more efficient ODE solver such as the Runge–Kutta–Fehlberg solver . . .*

while our calculations were done using an error–controlled BDF method with an accuracy of 10^{-7}.
Coming back to our intention to solve problem (6.1), we can easily transform solution (6.6) to a solution $w(\cdot)$ of (6.1) and notice that algorithm 5.3 stops in step 7., yielding an objective function value $\Phi^{\text{STO}} = 5.72864$ even smaller than the estimation from the relaxed problems.
The calculation of the relaxed solution takes 1.5 seconds and 19 QPs. For the refined grid we have 5 seconds and 43 QPs. If we add the switching time optimization, it is 6 seconds and 47 QPs.

6.2 Sliding mode chattering control

In chapter 4 we introduced chattering controls as a special type of bang–bang controls that switch infinitely often in a finite time interval. We will now and in section 6.3

investigate how algorithm 5.3 performs on problems for which chattering controls are known to be the best solution. Both problems do not exactly fit into the problem class we are interested in, as they are already in relaxed form. Nevertheless we apply our method to these problems assuming an integer requirement would hold.
Two different cases of chattering controls can be distinguished. The first one to be considered in this section is related to problems for which a minimizing trajectory does *not* exist. Consider the following example of Bolza that can be found in Zelikin & Borisov (1994).

$$\min_{x,w} \int_0^{\sqrt{2}} x^2 + (1 - (-1 + 2\,w)^2)^2 \,\mathrm{d}t \tag{6.8a}$$

subject to the ODE

$$\dot{x} \;=\; 1 - 2\,w \tag{6.8b}$$

with initial condition

$$x(0) = 0, \tag{6.8c}$$

terminal constraint

$$x(\sqrt{2}) = 0, \tag{6.8d}$$

path constraints

$$x(t) \geq 0, \tag{6.8e}$$

and a restriction of the control to values in

$$w(t) \in [0, 1], \quad t \in [0, \sqrt{2}]. \tag{6.8f}$$

Obviously every control function given by

$$w = \mathcal{S}\left(1; \underbrace{\frac{1}{n_{\mathrm{ms}}}, \ldots, \frac{1}{n_{\mathrm{ms}}}}_{n_{\mathrm{ms}}}\right), \tag{6.9}$$

with alternating values 1 and 0 on an equidistant grid yields an admissible trajectory for (6.8) if n_{ms} is an even number. The objective function value falls strictly monotonic with rising n_{ms}. States and the objective function are plotted in figure 6.4 for some values of n_{ms}. The objective function values converge towards 0, a function value which would require $x \equiv 0$. This state cannot be obtained by a bang–bang control, though. Therefore no optimal trajectory $\mathcal{T}^*$ exists. Nevertheless, as stated by theorem 4.9, for each $\varepsilon > 0$ there exists a bang–bang control with an objective value smaller ε. In fact, (6.9) is such a solution if n_{ms} is chosen sufficiently large.
We do not want to plug in an analytical solution, though, but rather test our algorithm 5.3. The first step will be to convexify the problem under consideration. In

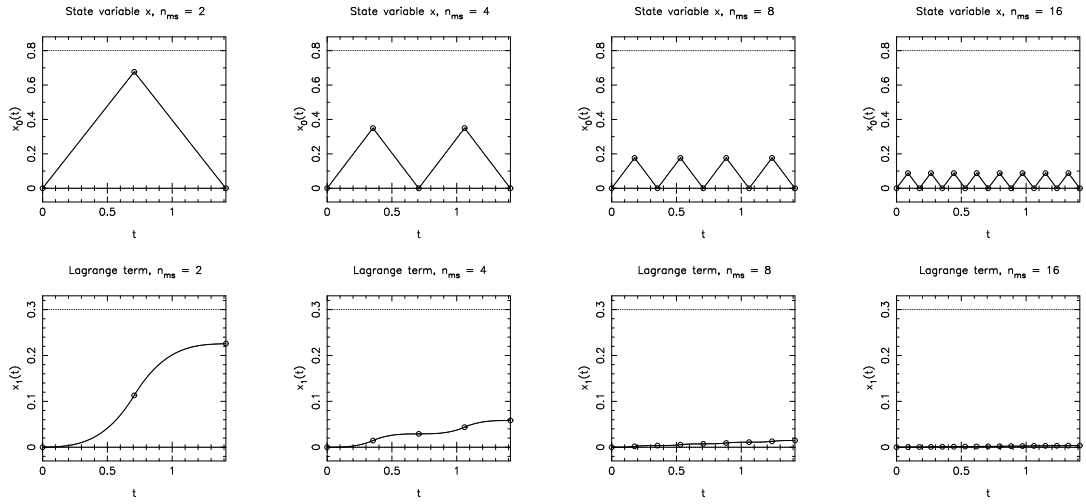

Figure 6.4: State $x(\cdot)$ (top row) and Lagrange term $\int_0^t L(x,w)\ \mathrm{d}\tau$ (bottom row) of trajectories corresponding to (6.9) with values $n_{\mathrm{ms}} = 2, 4, 8, 16$ from left to right.

this case the linear ODE is left unchanged and the objective function (6.8a) simplifies to

$$\min_{x,w} \int_0^{\sqrt{2}} x^2 \ \mathrm{d}t \tag{6.10}$$

as $(1-(-1+2\cdot 1)^2) = (1-(-1+2\cdot 0)^2) = 0$. The optimal solution of the relaxed problem on any grid $\mathcal{G}$ will of course be $w \equiv 0.5, x \equiv 0$. A rounding strategy will be useful, if the control is rounded up on all odd and rounded down on all even intervals. This is indeed achieved with rounding strategy SUR-0.5 for $w \equiv 0.5$. If for a given ε we refine the grid, apply the rounding heuristics, compare the obtained objective value with ε to either refine the grid further in a next iteration or to stop the iteration, we obtain trajectories as plotted in figure 6.4. For $\varepsilon = 10^{-3}$, e.g., we have to iterate three times starting with $n_{\mathrm{ms}} = 4$. On the grid

$$\mathcal{G}^3 = \left\{0, \frac{1}{32}\sqrt{2}, \ldots, \frac{31}{32}\sqrt{2}, \sqrt{2}\right\}$$

we get the objective function value $\Phi = 9.29\ 10^{-4}$, the values on $\mathcal{G}^0$, $\mathcal{G}^1$ and $\mathcal{G}^2$ are shown in figure 6.4.

The rounding approach is critical, though. First, numerical round off errors may have a large effect. In this example much depends on the question whether the control on the first interval is rounded up or down as $x \geq 0$ is required. As the value is exactly 0.5, no guarantee can be given. Furthermore it is very unlikely that the optimal relaxed control is exactly 0.5 as in this case, if sliding mode chattering occurs in practical problems.

Therefore we will apply steps 9 to 10 of algorithm 5.3. Again we have a peculiar situation, the extreme case where $w \equiv 0.5$ is the exact maximum of the concave penalty function. Starting from a nonequidistant grid, to make things a little harder,

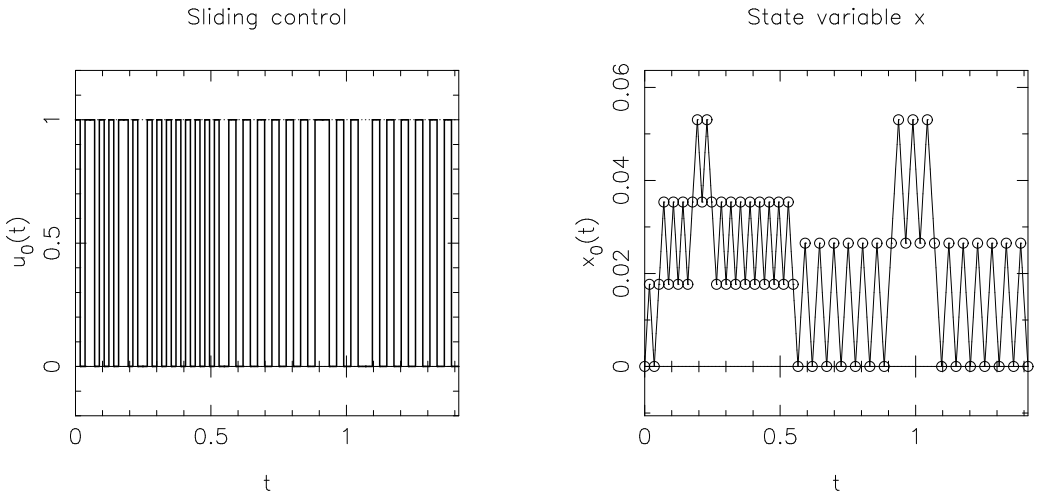

Figure 6.5: Control $w(\cdot)$, state $x(\cdot)$ and Lagrange term $\int_0^t L(x,w)\ \mathrm{d}\tau$ of the trajectory obtained by *MS MINTOC* . Although the solution is obviously a local minimum on the given grid, is suffices the terminal condition.

with $n_{\mathrm{ms}} = 4$, we get four outer iterations. As the solutions obtained in the first iterations from an application of the penalty term homotopy do not yield an objective value smaller than ε, we have to refine the grid using the relaxed solution to determine the switching points. The result is again a bisection as $w \equiv 0.5$, the resulting trajectory with objective value $8.79\ 10^{-4}$ is shown in figure 6.5. The solution is only a local minimum, as one readily sees in the behavior of the state variable $x(\cdot)$. This is also the reason why the grid has to be finer than the one for the global optimal solution (6.9). Let us see it the other way around: this implies that by a refinement of the control grid we can compensate the fact that we do only find a local solution with the penalty heuristics.
The calculation of the solution takes 5 seconds and 27 QPs.

6.3 Fuller's problem

The first control problem with an optimal chattering solution was given by Fuller (1963). This problem reads as follows.

$$\min_{\boldsymbol{x},w} \int_0^1 x_0^2\ \mathrm{d}t \tag{6.11a}$$

subject to the ODE

$$\dot{x_0} = x_1 \tag{6.11b}$$
$$\dot{x_1} = 1 - 2\,w \tag{6.11c}$$

with initial condition

$$\boldsymbol{x}(0) = \boldsymbol{x_0}, \tag{6.11d}$$

terminal constraint

$$\boldsymbol{x}(1) = \boldsymbol{x_T}, \tag{6.11e}$$

and a restriction of the control to values in

$$w(t) \in [0,1], \quad t \in [0,1]. \tag{6.11f}$$

In contrast to the chattering control example presented in section 6.2, an optimal trajectory does exist for all $\boldsymbol{x_0}$ and $\boldsymbol{x_T}$ in a vicinity of the origin. As Fuller showed, this optimal trajectory $\mathcal{T}^*$ contains a bang–bang control function that switches infinitely often in the interval $[0,1]$. An extensive analytical investigation of this problem and a discussion of the ubiquity of Fuller's problem can be found in Zelikin & Borisov (1994), a recent investigation of chattering controls in relay feedback systems in Johansson *et al.* (2002). We will use $\boldsymbol{x_0} = \boldsymbol{x_T} = (0.01, 0)^T$ for our calculations. The optimal trajectory for the relaxed control problem on an equidistant grid $\mathcal{G}^0$ with $n_{\text{ms}} = 19$ is shown in the top row of figure 6.6. Note that this solution is not bang–bang due to the discretization of the control space. Even if this discretization is made very fine, a trajectory with $w(t) = 0.5$ on an interval in the middle of $[0,1]$ will be found as a minimum. To obtain a bang–bang solution, we have to apply our algorithm. The trajectory obtained by a switching time optimization, initialized with the result of rounding strategy SR, is shown in the middle row of figure 6.6. The switching time optimization method yields an objective value of $1.89\ 10^{-4}$. As the objective function value of the relaxed problem is smaller, $\Phi^{\text{RL}} = 1.53\ 10^{-5}$, one might want to reduce the function value further, e.g. closer than $\varepsilon = 10^{-6}$ to Φ^{RL}. If we apply algorithm 5.3, we obtain the trajectory shown in the bottommost row of figure 6.6 that yields an objective function value of $1.52\ 10^{-5}$ and switches 35 times. The calculation of the optimal trajectory of the relaxed problem takes approximately 1 second and 16 QPs, the switching time optimization 5 seconds and 42 QPs and the *MS MINTOC* algorithm takes 36 seconds and 163 QP solutions.

6.4 Fishing problem

An optimal trajectory of a control problem may contain compromise–seeking (singular) arcs, compare chapter 2. In this section we will review the fishing example introduced in section 1.4.2. We will replace the switching restriction $w(\cdot) \in \Omega(\Psi_\tau)$ in problem formulation (1.19) by $w(\cdot) \in \Omega(\Psi_{\text{free}})$. In appendix B it is shown that the optimal trajectory of the control problem given by

$$\min_{\boldsymbol{x},w} \int_{t_0}^{t_f} (x_0(t) - 1)^2 + (x_1(t) - 1)^2 \, dt \tag{6.12a}$$

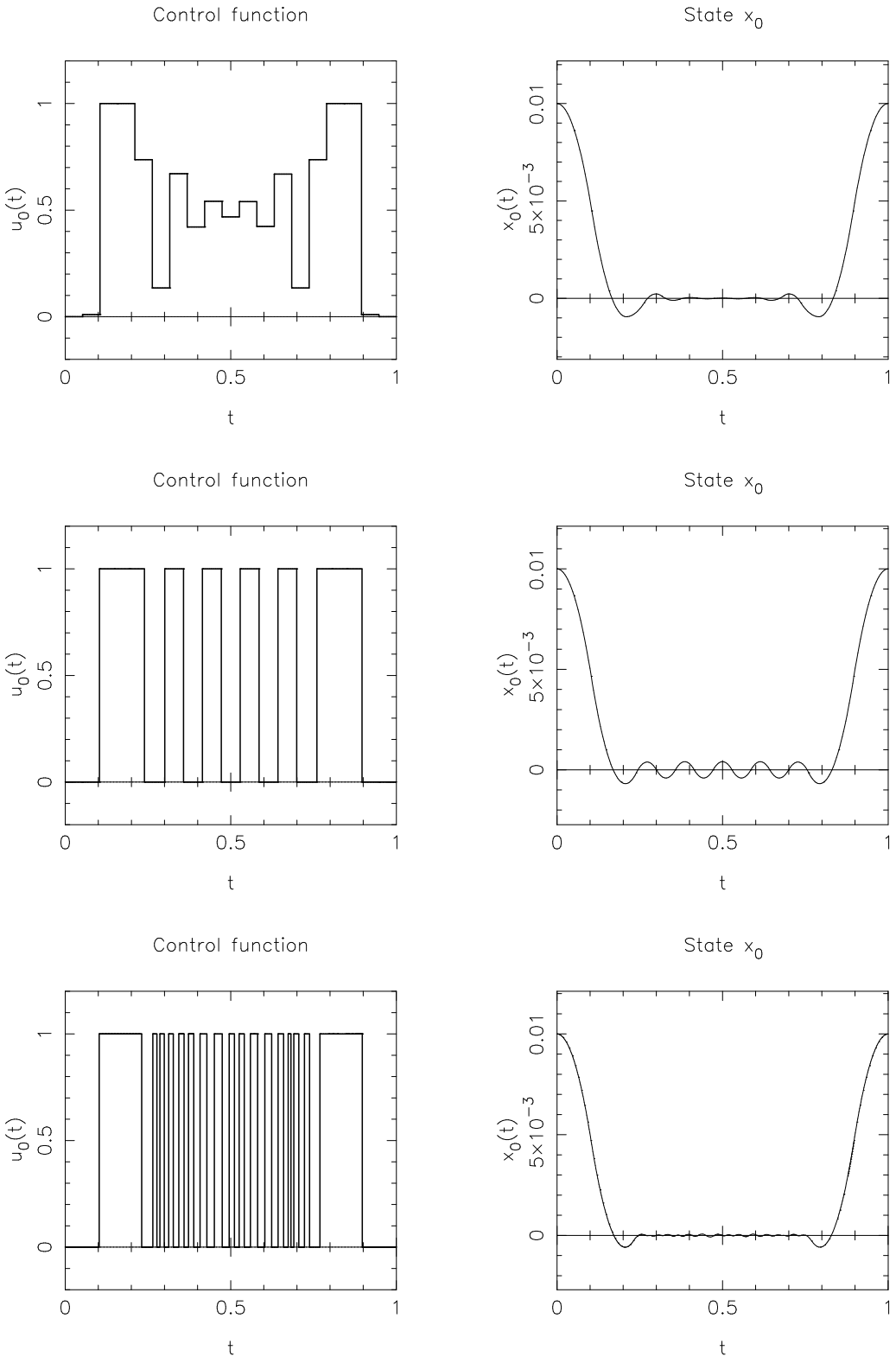

Figure 6.6: Control $w(\cdot)$ and states $\boldsymbol{x}(\cdot)$ for Fuller's problem. Top: optimal trajectory for the relaxed problem on grid $\mathcal{G}^0$. Middle: trajectory obtained by switching time optimization. Bottom: trajectory obtained by *MS MINTOC* with higher accuracy.

subject to the ODE

$$\dot{x}_0(t) = x_0(t) - x_0(t)x_1(t) - c_0 x_0(t)\, w(t), \tag{6.12b}$$
$$\dot{x}_1(t) = -x_1(t) + x_0(t)x_1(t) - c_1 x_1(t)\, w(t), \tag{6.12c}$$

initial values

$$\boldsymbol{x}(t_0) = \boldsymbol{x_0}, \tag{6.12d}$$

and the integer constraints

$$w(\cdot) \in \Omega(\Psi_{\text{free}}) \tag{6.12e}$$

contains a singular arc, compare figures B.1 and 6.7.

Remark 6.1 *The occurring singular arc is caused by the objective function. If problem (6.12) is replaced by problem (B.9) to minimize the time to get into the steady state* $\boldsymbol{x_T} = (1,1)^T$*, the optimal solution will be entirely bang–bang. See section B.6 in the appendix for further details and a plot of the optimal trajectory.*

When singular arcs are present, it is not sufficient to refine the grid to obtain a bang–bang solution as, e.g., for the F–8 aircraft problem in section 6.1 or the switching point between the first two arcs in this example. But, as a first step, an adaptation of the control grid is useful to give insight into the switching structure and to obtain an estimation for Φ^{RL}. Figure 6.7 shows optimal control functions on different control discretization grids. We obtain an objective value of $\Phi^{\text{RL}}_{\mathcal{G}^5} = 1.34409$ which is indeed a good estimate for the value of the infinite–dimensional problem solved in the appendix, $\Phi^* = 1.34408$. Figure 6.8 shows the states that correspond to this solution.

To obtain a binary admissible solution we proceed as follows. We apply a penalty homotopy on either of the grids $\mathcal{G}^0 \ldots \mathcal{G}^5$ and use the obtained solution to initialize the switching time optimization. In algorithm *MS MINTOC* , we start with a given tolerance ε and check in each step whether we are close enough. Here we will present all values up to the grid $\mathcal{G}^5$ with $n_{\text{ms}} = 264$ grid points to illustrate typical behavior for systems comprising singular arcs. We write $\tilde{\Phi}^k$ for the solution of the relaxed problem, $\hat{\Phi}^k$ for the objective value obtained after termination of the penalty homotopy and Φ^k for the objective value after the switching time optimization, everything on the fixed grid $\mathcal{G}^k$. Table 6.1 then gives a comparison of the obtained objective values. For the grid $\mathcal{G}^5$ numerical problems occur, as some time points are too close to each other. This leads to *cycling* phenomena in the active set based QP solution procedure. Therefore we have to be content with the solution on grid $\mathcal{G}^4$, which yields the objective function value of 1.34424 and is therefore closer than $2\ 10^{-4}$ to $\tilde{\Phi}^4$. Note that *MS MINTOC* decides on its own whether a grid has to be further refined or not. The determination of Φ^4 took 135 seconds and the solution of 75

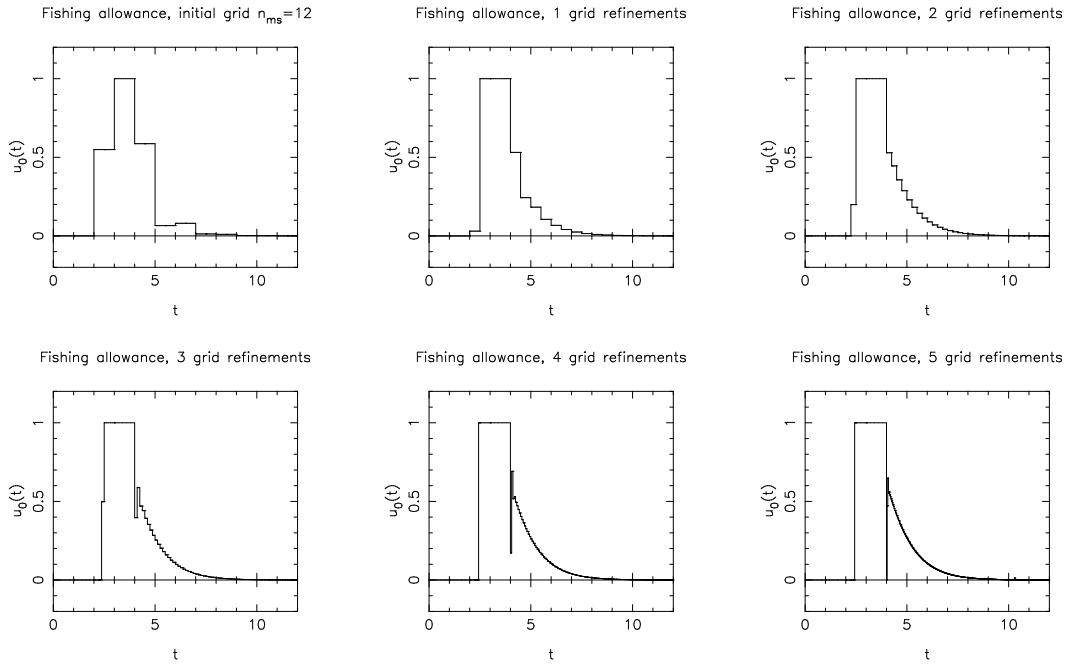

Figure 6.7: Optimal control functions $w(\cdot)$ of the relaxed fishing problem on different grids $\mathcal{G}^k$. The initial grid $\mathcal{G}^0$ is chosen equidistantly with $n_{\mathrm{ms}} = 12$, all other grids are obtained by bisection of the preceding grid, wherever a control is not at its bounds. The final grid $\mathcal{G}^5$ consists of $n_{\mathrm{ms}} = 264$ intervals.

QPs. The corresponding control function switches 22 times, is given by

$$\begin{aligned} w(\cdot) = \mathcal{S}(0;\quad & 2.43738, 1.57402, 0.0897238, 0.118878, \\ & 0.109465, 0.101335, 0.13225, 0.088103, \\ & 0.159216, 0.0772873, 0.193265, 0.0679251, \\ & 0.238052, 0.0593931, 0.301416, 0.0511773, \\ & 0.398989, 0.0428478, 0.57489, 0.0336509, \\ & 1.01055, 0.0211654, 4.11901). \end{aligned} \tag{6.13}$$

and depicted in figure 6.9. Note that the stage lengths where $w(\cdot) = 1$ are monotonically decreasing as the stage lengths where $w(\cdot) = 0$ are monotonically increasing on the singular arc.

6.5 Fishing problem on a fixed grid

The main strength of our method is the adaptivity in the control discretization. It may happen though that no solution can be found that is close enough to the optimal value of the relaxed problem, Φ^{RL}. This might either be because the control grid is

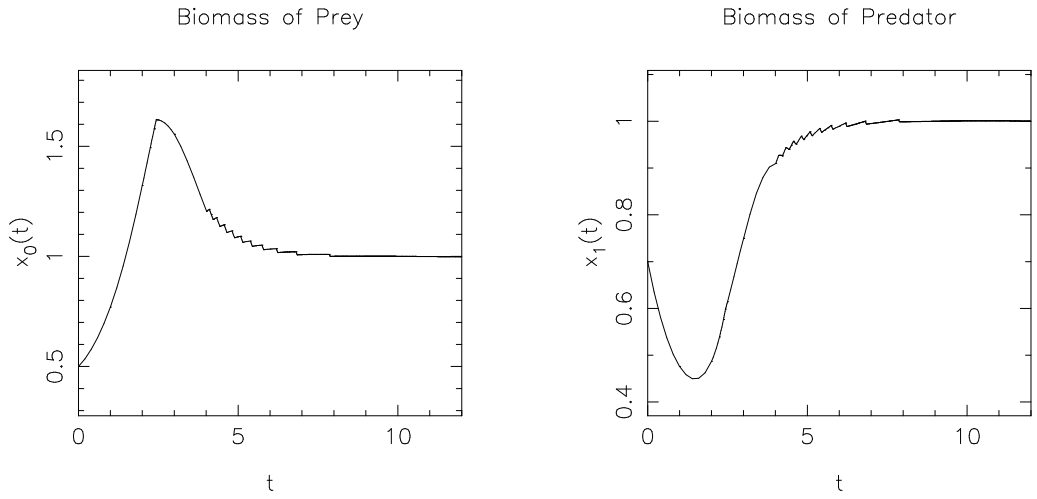

Figure 6.8: States $\boldsymbol{x}(\cdot)$ from the best binary admissible trajectory (6.13) for the fishing problem. They yield an objective function value of $\Phi^4 = 1.34424$. Note the non-differentiabilities in the states caused by the switching of the control function.

Grid	$\tilde{\Phi}^k$	$\hat{\Phi}^k$	Φ^k
$\mathcal{G}^0$	1.38568	1.38276	1.38276
$\mathcal{G}^1$	1.34751	1.42960	1.38276
$\mathcal{G}^2$	1.34563	1.35750	1.34632
$\mathcal{G}^3$	1.34454	1.35384	1.34461
$\mathcal{G}^4$	1.34409	1.34613	1.34424
$\mathcal{G}^5$	1.34409	–	–

Table 6.1: Objective values $\tilde{\Phi}^k$ of the relaxed problem, $\hat{\Phi}^k$ after termination of the penalty homotopy and Φ^k of the resulting trajectories after a switching time optimization, performed on several grids $\mathcal{G}^k$.

fixed, $\Psi = \Psi_\tau$, or because the heuristics do not yield a sufficient solution for a given discretization.

In these cases we have to apply a global optimization approach, compare section 2.4 and chapter 3. In this section we will consider problem (1.19) on a fixed grid and apply a Branch & Bound strategy to find the global solution on a given discretization of the control grid. We will consider the case where $\mathcal{G}$ is given by an equidistant discretization with $n_{\text{ms}} = 60$ and choose the control parameterization intervals $[t_i, t_{i+1}]$ such that they coincide with the intervals $[\tau_i, \tau_{i+1}]$, $i = 0 \ldots n_{\text{ms}} = n_\tau$.

Optimal trajectories of relaxed problems have already been determined in section 6.4 for different grids. For $\mathcal{G}$ the objective function value is $\Phi^{\text{RL}} = 1.34465$, the optimal control function $w(\cdot)$ looks very similar to those plotted in figure 6.7. 19 QPs have to be solved to obtain this solution.

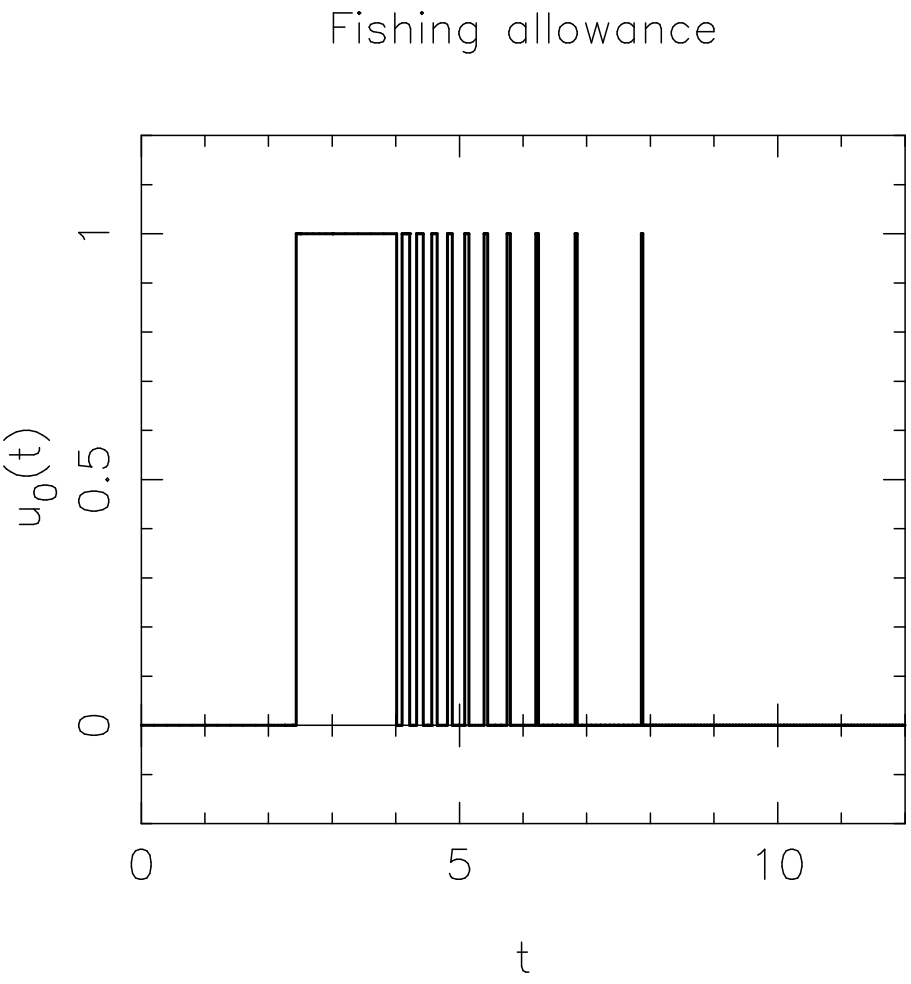

Figure 6.9: Best binary admissible control function $w(\cdot)$ of the fishing problem, given by (6.13).

Table 6.2 gives an overview of the objective function values that correspond to binary admissible trajectories found by different heuristics. As it indicates, the trajectories

Heuristics	# QPs	Objective value
$w(\cdot) \equiv 0$	0	6.06187
$w(\cdot) \equiv 1$	0	9.40231
Rounding SR	19	1.51101
Rounding SUR	19	1.45149
Rounding SUR-0.5	19	1.34996
Penalty homotopy	47	1.34996

Table 6.2: Number of QPs to be solved and obtained objective value for several heuristics to get a binary admissible solution.

for rounding strategy SUR-0.5 and the penalty homotopy are indeed identical for the chosen parameters and this example. This solution is given by

$$w(t) = \begin{cases} 1 & t \in [\tau_i, \tau_{i+1}] \text{ and } i \in I_{\text{on}} = \{13, 14, \ldots, 20, 22, 24, 28\} \\ 0 & t \in [\tau_i, \tau_{i+1}] \text{ and } i \in \{1, \ldots, 60\} \,/\, I_{\text{on}} \end{cases} \tag{6.14}$$

As the optimal solution of the relaxed problem is a lower bound on the optimal solution of the binary problem, the difference between the relaxed solution, that is, $\Phi^{\text{RL}} = 1.34465$ and a binary admissible solution $\hat{\Phi}$ gives an indication about how good this heuristic solution is. At runtime one could determine if the relative gap of

$$\frac{1.34996 - 1.34465}{1.34465} \approx 0.004$$

is sufficient. We will now assume this is not the case and apply a Branch & Bound algorithm to find the global optimal solution of problem (1.19). Branch & Bound only works for convex problems, compare section 3.2. In the following we assume that both the objective function and the feasible set of the fishing problem are convex. That this assumption is justified at least in the vicinity of the solution is shown in appendix B.5. Using depth-first search and most violation branching we obtain the global optimal solution

$$w(t) = \begin{cases} 1 & t \in [\tau_i, \tau_{i+1}] \text{ and } i \in I_{\text{on}} = \{13, 14, \ldots, 20, 22, 25, 28\} \\ 0 & t \in [\tau_i, \tau_{i+1}] \text{ and } i \in \{1, \ldots, 60\} \,/\, I_{\text{on}} \end{cases} . \tag{6.15}$$

after 6571 SQP iterations and about 15 minutes[2], yielding an objective function value of $\Phi^{\text{GO}} = 1.34898$.

The large number of necessary SQP iterations can be reduced significantly by making use of the obtained binary admissible solutions as upper bounds. Making use of the solution obtained by the penalty homotopy, the number of iterations can be reduced

[2]A large part of this computing time goes into graphics and I/O. Note that the number of iterations is much smaller than given in Sager *et al.* (2006) for this problem as a trust box technique and no updates for the Hessian have been used here

by 20%. For the calculations done in Sager *et al.* (2006) this number was even up to 50%.
Rounding strategy SUR-0.5 and the penalty homotopy give a result close to the optimal solution (close in the sense that the *Hamming distance*, the count of bits different in two patterns, is only 2), differing only on intervals $22, 23$ and $27, 28$. If we start with the global solution as an upper bound, further 756 nodes have to be visited and 3705 SQP iterations are needed to verify the globality of the solution in our Branch and Bound implementation. See figure 6.10 for a plot the optimal trajectory.

6.6 Summary

In this chapter we presented several case studies to illustrate the broad applicability of our methods. We saw in section 6.1 that problems having a bang–bang structure in the relaxed binary control functions can be solved with only very few additional effort compared to the relaxed solution. For such problems the main problem is to find the switching points, the relaxed solution will then coincide with the binary admissible solution.
In sections 6.2 and 6.3 we investigated problems with chattering controls. They differ in a theoretical way, as example 6.2 does not possess an optimal solution although a limit of trajectories exists and example 6.3, the famous problem of Fuller, does have a solution that can be proven to be chattering. For both problems we obtain solutions with a finite number of switches that are closer than a prescribed tolerance ε to the globally optimal objective function value resp. an estimation of it. For such problems the main task is to first determine an adequate control discretization grid and than apply a method to obtain an integer solution on this grid.
In section 6.4 we derive an approximation for an example with a singular arc and again get very close to the optimal solution (that is derived in the appendix for completeness). Singular arcs are closely related to arcs that have a chattering solution, as they can be approximated by one.
In section 6.5 we extend our study to the case where we need a global solution for a fixed control discretization grid. We demonstrate how a Branch & Bound algorithm can be applied to find such a global solution and point out which role heuristics play in such a scheme.
The various control problems are, from an optimal control point of view, completely different in their solution structure. We showed that the *MS MINTOC* algorithm solves all of them up to a prescribed tolerance — without any a priori guesses on the switching structure or values of Lagrange multipliers. Therefore we may and indeed do assume that our methods are applicable to a broad class of optimal control problems.

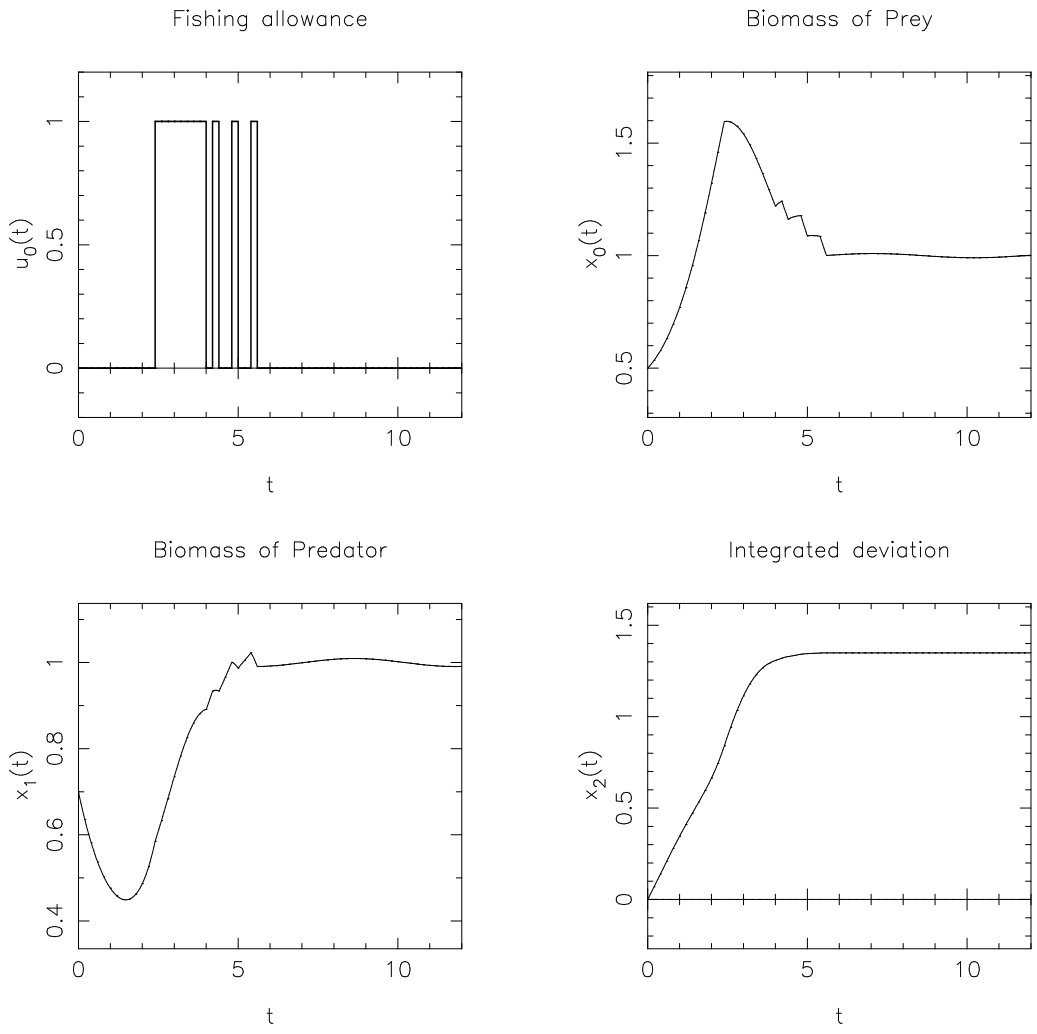

Figure 6.10: Optimal trajectory on the fixed grid $\mathcal{G}$ of the fishing problem. The solution is given by (6.5). Note again the approximation by switching of the optimal state trajectory by a nondifferentiable one, compare figure 4.1.

Chapter 7

Applications

In this chapter we present mixed–integer optimal control problems of the form (1.18) and give numerical results. The applications are from three different fields, namely mechanics, systems biology and chemical engineering.
In section 7.1 we solve a minimum energy problem related to the New York subway system. A subway train can be operated in discrete stages only and several constraints have to be fulfilled. This example is particularly challenging as state–dependent nondifferentiabilities in the model functions occur.
In section 7.2 a model to influence a signalling pathway in cells, based on the concentration of calcium ions, is presented. Although the model consists of an ODE only and has no difficult path constraints, it is very hard to find an integer solution. This is due to the fact that the system is extremly unstable. The results of this section have partly been published in Lebiedz *et al.* (2005).
In section 7.3 we present the example of a batch distillation process with so–called waste cuts. Recycling of the waste cuts is formulated as a cyclically repeated batch distillation process, where the waste cuts are recycled at intermediate points in time. We solve the resulting multipoint boundary value optimization problem including binary control functions and derive a time– and tray–dependent reusage of these cuts that improves the objective functional significantly and helps to give insight into the process.
Please note that the notation partly changes in this chapter. Explanations are given in the text and in the appendix.

7.1 Subway optimization

The optimal control problem we treat in this section goes back to work of Bock & Longman (1982) for the city of New York. It aims at minimizing the energy used for a subway ride from one station to another, taking into account boundary conditions and a restriction on the time. It is given by

$$\min_{\boldsymbol{x},w,T} \int_0^T L(\boldsymbol{x}(t), w(t))\ \mathrm{d}t \tag{7.1a}$$

subject to the ODE system

$$\dot{\boldsymbol{x}}(t) \;=\; \boldsymbol{f}(\boldsymbol{x}(t), w(t)), \quad t \in [t_0, T], \tag{7.1b}$$

path constraints

$$\mathbf{0} \;\leq\; \boldsymbol{x}(t), \quad t \in [t_0, T], \tag{7.1c}$$

interior point inequalities and equalities

$$\mathbf{0} \;\leq\; \boldsymbol{r}^{\mathbf{ieq}}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \ldots, \boldsymbol{x}(T), T), \tag{7.1d}$$
$$\mathbf{0} \;=\; \boldsymbol{r}^{\mathbf{eq}}(\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \ldots, \boldsymbol{x}(T), T), \tag{7.1e}$$

and binary admissibility of $w(\cdot)$

$$w(t) \;\in\; \{1, 2, 3, 4\}. \tag{7.1f}$$

The terminal time T denotes the time of arrival of a subway train in the next station. The differential states $x_0(\cdot)$ and $x_1(\cdot)$ describe position resp. velocity of the train. The train can be operated in one of four different modes,

$$w(t) \;=\; \begin{cases} 1 & \text{Series} \\ 2 & \text{Parallel} \\ 3 & \text{Coasting} \\ 4 & \text{Braking} \end{cases} \tag{7.1g}$$

that influences the acceleration resp. deceleration of the train and therewith the energy consumption. The latter is to be minimized and given by the Lagrange term

$$L(\boldsymbol{x}(t), 1) \;=\; \begin{cases} e\, p_1 & \text{for } x_1(t) \leq v_1 \\ e\, p_2 & \text{for } v_1 < x_1(t) \leq v_2 \\ e\, \sum_{i=0}^{5} c_i(1) \left(\frac{1}{10}\gamma\, x_1(t)\right)^{-i} & \text{for } x_1(t) > v_2 \end{cases}, \tag{7.1h}$$

$$L(\boldsymbol{x}(t), 2) \;=\; \begin{cases} 0 & \text{for } x_1(t) \leq v_2 \\ e\, p_3 & \text{for } v_2 < x_1(t) \leq v_3 \\ e\, \sum_{i=0}^{5} c_i(2) \left(\frac{1}{10}\gamma\, x_1(t) - 1\right)^{-i} & \text{for } x_1(t) > v_3 \end{cases}, \tag{7.1i}$$

$$L(\boldsymbol{x}(t), 3) \;=\; 0, \tag{7.1j}$$

$$L(\boldsymbol{x}(t), 4) \;=\; 0. \tag{7.1k}$$

The right hand side function $\boldsymbol{f}(\cdot)$ is dependent on the mode $w(\cdot)$ and on the state variable $x_1(\cdot)$. For all $t \in [0, T]$ we have

$$\dot{x_0}(t) \;=\; x_1(t). \tag{7.1l}$$

For operation in series, $w(t) = 1$, we have

$$\dot{x_1}(t) \;=\; f_1(\boldsymbol{x}, 1) = \begin{cases} f_1^{1A}(\boldsymbol{x}) & \text{for } x_1(t) \leq v_1 \\ f_1^{1B}(\boldsymbol{x}) & \text{for } v_1 < x_1(t) \leq v_2 \\ f_1^{1C}(\boldsymbol{x}) & \text{for } x_1(t) > v_2 \end{cases}, \tag{7.1m}$$

with

$$\begin{aligned} f_1^{1A}(\boldsymbol{x}) &= \frac{g\ e\ a_1}{W_{\text{eff}}}, \\ f_1^{1B}(\boldsymbol{x}) &= \frac{g\ e\ a_2}{W_{\text{eff}}}, \\ f_1^{1C}(\boldsymbol{x}) &= \frac{g\ (e\ T(x_1(t),1) - R(x_1(t))}{W_{\text{eff}}}. \end{aligned}$$

For operation in parallel, $w(t) = 2$, we have

$$\dot{x_1}(t) = f_1(\boldsymbol{x}, 2) = \begin{cases} f_1^{2A}(\boldsymbol{x}) & \text{for } x_1(t) \le v_2 \\ f_1^{2B}(\boldsymbol{x}) & \text{for } v_2 < x_1(t) \le v_3 \\ f_1^{2C}(\boldsymbol{x}) & \text{for } x_1(t) > v_3 \end{cases}, \tag{7.1n}$$

with

$$\begin{aligned} f_1^{2A}(\boldsymbol{x}) &= 0, \\ f_1^{2B}(\boldsymbol{x}) &= \frac{g\ e\ a_3}{W_{\text{eff}}}, \\ f_1^{2C}(\boldsymbol{x}) &= \frac{g\ (e\ T(x_1(t),2) - R(x_1(t))}{W_{\text{eff}}}. \end{aligned}$$

For coasting, $w(t) = 3$, we have

$$\dot{x_1}(t) = f_1(\boldsymbol{x}, 3) = -\frac{g\ R(x_1(t))}{W_{\text{eff}}} - C \tag{7.1o}$$

and for braking, $w(t) = 4$,

$$\dot{x_1}(t) = f_1(\boldsymbol{x}, 4) = -u(t) = u_{\max}. \tag{7.1p}$$

The braking deceleration $u(\cdot)$ can be varied between 0 and a given $u_{\max}$. It can be shown easily that for the problem at hand only maximal braking can be optimal, hence we fix $u(\cdot)$ to $u_{\max}$ without loss of generality.
The occurring forces are given by

$$R(x_1(t)) = ca\ \gamma^2 x_1(t)^2 + bW\gamma x_1(t) + \frac{1.3}{2000}W + 116, \tag{7.1q}$$

$$T(x_1(t),1) = \sum_{i=0}^{5} b_i(1) \left(\frac{1}{10}\gamma x_1(t) - 0.3\right)^{-i}, \tag{7.1r}$$

$$T(x_1(t),2) = \sum_{i=0}^{5} b_i(2) \left(\frac{1}{10}\gamma x_1(t) - 1\right)^{-i}. \tag{7.1s}$$

The interior point equality constraints $\boldsymbol{r}^{\mathbf{eq}}(\cdot)$ are given by initial and terminal constraints on the state trajectory,

$$\boldsymbol{x}(0) = (0,0)^T, \quad \boldsymbol{x}(T) = (S,0)^T. \tag{7.1t}$$

The interior point inequality constraints $\boldsymbol{r}^{\mathbf{ieq}}(\cdot)$ consist of a maximal driving time $T^{\max}$ to get from $\boldsymbol{x}(0) = (0,0)^T$ to $\boldsymbol{x}(T) = (S,0)^T$,

$$T \leq T^{\max}. \tag{7.1u}$$

In the equations above the parameters e, p_1, p_2, p_3, $b_i(w)$, $c_i(w)$, γ, g, a_1, a_2, a_3, W_{eff}, C, c, b, W, $u_{\max}$, $T^{\max}$, v_1, v_2 and v_3 are fixed. They are given in appendix C. Details about the derivation of this model and the assumptions made can be found in Bock & Longman (1982) or in Krämer-Eis (1985).
Bock & Longman (1982) solved the problem at hand for different values of S and W already in the early eighties by the *Competing Hamiltonians* approach. This approach computes the values of Hamiltonian functions for each possible mode of operation and compares them in every time step. As the maximum principle holds also for disjoint control sets, the maximum of these Hamiltonians determines the best possible choice. This approach is based on indirect methods, therefore it suffers from the disadvantages named in chapter 2 — in particular from the need to supply very accurate initial values for the Lagrange multipliers and the switching structure. Our direct approach does not require such guesses. We transform the problem with the discrete–valued function $w(\cdot)$ to a convexified one with a four–dimensional control function $\tilde{\boldsymbol{w}} \in [0,1]^4$ and $\sum_{i=1}^4 \tilde{w}_i(t) = 1$ for all $t \in [0,T]$ as described in chapter 4. This allows us to write the right hand side function $\tilde{\boldsymbol{f}}$ and the Lagrange term $\tilde{L}$ as

$$\tilde{\boldsymbol{f}}(\boldsymbol{x}, \tilde{\boldsymbol{w}}) \;=\; \sum_{i=1}^{4} \tilde{w}_i(t)\; \boldsymbol{f}(\boldsymbol{x}, i)$$

respectively as

$$\tilde{L}(\boldsymbol{x}, \tilde{\boldsymbol{w}}) \;=\; \sum_{i=1}^{4} \tilde{w}_i(t)\; L(\boldsymbol{x}, i).$$

Both functions still contain state–dependent discontinuities. Recent work in the area of such implicit discontinuities has been performed by Brandt-Pollmann (2004), who proposes a monitoring strategy combined with switching point determination and Wronskian update techniques. Fortunately the order of the different areas is quite clear in our case. As the distance S that has to be covered in time $T^{\max}$, a certain minimum velocity greater than v_3 is required for a given time and any admissible solution has to accelerate at the beginning, keep a certain velocity and decelerate by either coasting or braking towards the end of the time horizon. Therefore we assume that every optimal admissible trajectory fits into the structure of the multistage problem

- Stage 0, $[\tilde{t}_0, \tilde{t}_1] : 0 \leq x_1(\cdot) \leq v_1$, only series, $\tilde{w}_2 = \tilde{w}_3 = \tilde{w}_4 = 0$
- Stage 1, $[\tilde{t}_1, \tilde{t}_2] : v_1 \leq x_1(\cdot) \leq v_2$, only series, $\tilde{w}_2 = \tilde{w}_3 = \tilde{w}_4 = 0$
- Stage 2, $[\tilde{t}_2, \tilde{t}_3] : v_2 \leq x_1(\cdot) \leq v_3$

- Stage 3, $[\tilde{t}_3, \tilde{t}_4] : v_3 \leq x_1(\cdot)$
- Stage 4, $[\tilde{t}_4, \tilde{t}_5] : v_3 \leq x_1(\cdot)$
- Stage 5, $[\tilde{t}_5, \tilde{t}_6] : 0 \leq x_1(\cdot) \leq v_3$, only coasting or braking, $\tilde{w}_1 = \tilde{w}_2 = 0$

with $\tilde{t}_0 = t_0 = 0$ and $\tilde{t}_6 = T \leq T^{\max}$. The fourth stage has been split up in two stages, because we will insert additional constraints later on. The first two stages are pure acceleration stages. As $f_2(\boldsymbol{x}, 2) \equiv 0$ on the first two stages, we fix $\tilde{w}_1 = 1$ and $\tilde{w}_2 = \tilde{w}_3 = \tilde{w}_4 = 0$ on both. This allows us to compute the exact switching times $\tilde{t}_1$ and $\tilde{t}_2$ between these stages and fix them. On the sixth stage we assume that no further acceleration is necessary once the threshold velocity v_3 has been reached and allow only further deceleration by coasting or braking. Therefore no discontinuity will occur on this stage any more. As the constraint $v_3 \leq x_1(\cdot)$ avoids discontinuities, the only switching point to determine is $\tilde{t}_3$. We determine $\tilde{t}_3$ by the addition of an interior point constraint

$$x_1(\tilde{t}_3) = v_3,$$

although this approach may yield numerical difficulties as the model is only accurate when this condition is fulfilled. If, on the other hand, we obtain an admissible solution that fulfills the conditions on $x_1(\cdot)$ given above, the model restrictions are also fulfilled and the discontinuities take place at times where the model stages change and all derivative information is updated. For this reason all given solutions are indeed local optima that are admissible, also in the sense that the model discontinuities are treated correctly. Within our approach we use a line search instead of a trust box or watchdog technique to globalize convergence.
We will now investigate problem (7.1) with the same parameters used in Krämer-Eis (1985), namely $n = 10$, $W = 78000$, $S = 2112$, $T^{\max} = 65$, $e = 1.0$ and all other parameters as given in the appendix. We obtain a multistage problem with six model stages on which different interior point and path constraints are given. For these parameters we determine the switching times of the series mode in stages 0 and 1 as

$$\tilde{t}_1 = 0.631661, \quad \tilde{t}_2 = 2.43955. \tag{7.2}$$

We will first have a look at a trajectory of a relaxation of this problem. This solution is optimal on a given grid $\mathcal{G}^0$ with $n_{\text{ms}} = 34$ intervals. This grid is not equidistant, due to the multitude of stages that partly have fixed stage lengths. The obtained solutions for the binary control functions $\tilde{w}_i(\cdot)$ on this and a refined grid are shown in figure 7.1. The corresponding trajectories yield objective values of 1.15086 resp. of 1.14611. Applying a second refinement the solution is almost completely integer with $\Phi = 1.14596$. We round this solution and initialize a switching time optimization with it. The obtained trajectory, including the differential states distance $x_0(\cdot)$ and velocity $x_1(\cdot)$ is plotted in figure 7.2. The solution is given by

$$w(t) = \mathcal{S}(1, 2, 1, 3, 4;\ 3.64338, 8.96367, 33.1757, 11.3773, 7.84002). \tag{7.3}$$

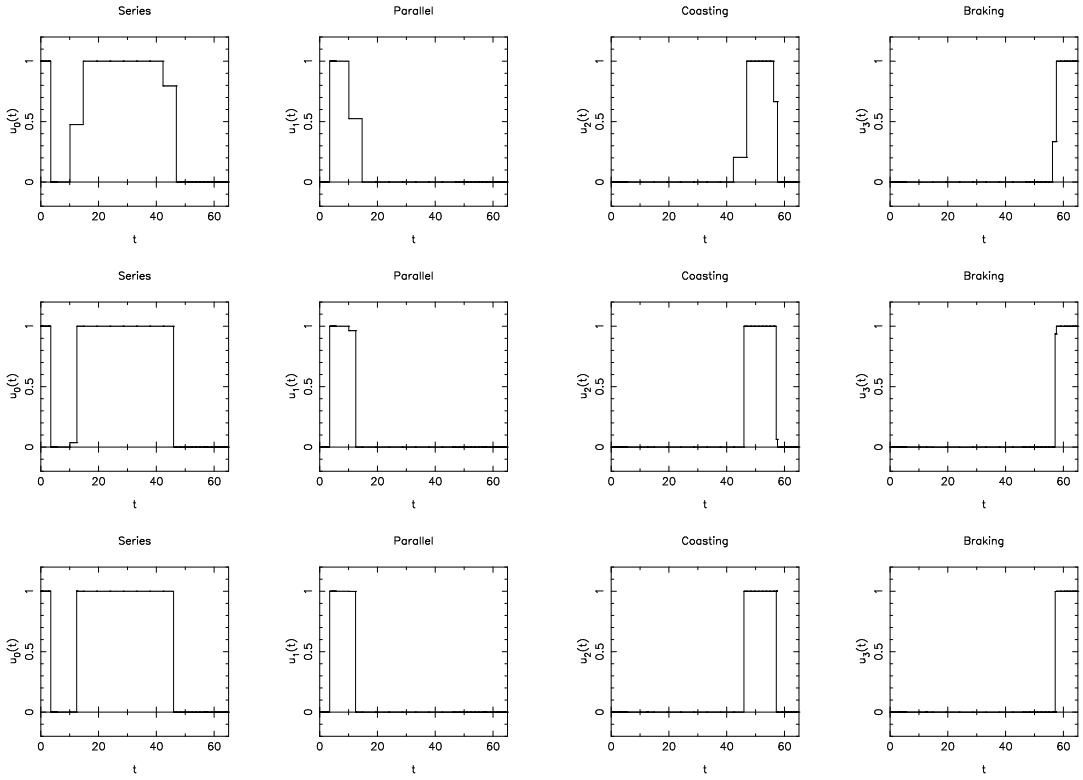

Figure 7.1: The controls for operation in series, $\tilde{w}_1(\cdot)$, in parallel, $\tilde{w}_2(\cdot)$, coasting, $\tilde{w}_3(\cdot)$ and braking, $\tilde{w}_4(\cdot)$, from left to right. The upper solution is optimal for the relaxed problem on a given grid $\mathcal{G}^0$, the middle one for a grid $\mathcal{G}^1$ obtained from $\mathcal{G}^0$ by adaptive mode 2. The lowest row shows the optimal solution on grid $\mathcal{G}^2$ that is used to initialize the switching time optimization algorithm.

Time t	$Mode$	$f_1 =$	$x_0(t)[ft]$	$x_1(t)[mph]$	$x_1(t)[ft/s]$	Energy
0.0	S	f_1^{1A}	0.0	0.0	0.0	0.0
0.631661	S	f_1^{1B}	0.453711	0.979474	1.43656	0.0186331
2.43955	S	f_1^{1C}	10.6776	6.73211	9.87375	0.109518
3.64338	P	f_1^{2B}	24.4836	8.65723	12.6973	0.147387
5.59988	P	f_1^{2C}	57.3729	14.2658	20.9232	0.339851
12.607	S	f_1^{1C}	277.711	25.6452	37.6129	0.93519
45.7827	C	$f_1(3)$	1556.5	26.8579	39.3915	1.14569
46.8938	C	$f_1(3)$	1600	26.5306	38.9115	1.14569
57.16	B	$f_1(4)$	1976.78	23.5201	34.4961	1.14569
65.00	-	–	2112	0.0	0.0	1.14569

Table 7.1: Trajectory corresponding to the optimal solution (7.3). The rows of the table give typical values for the different arcs.

In other words, first we operate in series until $\hat{t}_1 = 3.64338 \in [\tilde{t}_2, \tilde{t}_3]$ with state–dependent changes of the right hand side function at $\tilde{t}_1$ and $\tilde{t}_2$ as given by (7.2), then we operate in parallel mode until $\hat{t}_2 = 12.607 \in [\tilde{t}_3, \tilde{t}_5]$, then again in series until $\hat{t}_3 = 45.7827 \in [\tilde{t}_3, \tilde{t}_5]$. At $\hat{t}_4 = 57.16 \in [\tilde{t}_3, \tilde{t}_5]$ we stop coasting and brake until $T = T^{\max} = 65$. All results are given as an overview in table 7.1.
This solution is identical in structure to the one given in Krämer-Eis (1985). The switching times are a little bit different, though. This may be connected to the phenomenon of multiple local minima that occur when applying a switching time approach, compare section 5.2. The trajectory given in Krämer-Eis (1985) yields an energy consumption of $\Phi = 1.14780$. If we use either this solution or the rounded solution of the relaxed solution without adaptive refinement of the control grid as an initialization of the switching time approach, we obtain the local minimum

$$w(t) = \mathcal{S}(1, 2, 1, 3, 4;\ 3.6415, 8.82654, 34.5454, 10.0309, 7.95567),$$

which switches earlier into the parallel mode, has an augmented runtime in series and a shorter coasting arc. The objective function value of $\Phi = 1.14661$ is worse than the one given above, but still close enough to the relaxed value that serves as an estimate for Φ^*.
Our algorithm has therefore the ability to reproduce the optimal results of Bock & Longman (1982) and Krämer-Eis (1985). But we can go further, as we can apply our algorithm also to extended problems with additional constraints. To illustrate this, we will add constraints to problem (7.1). First we consider the point constraint

$$x_1(t) \leq v_4 \text{ if } x_0(t) = S_4 \tag{7.4}$$

for a given distance $0 < S_4 < S$ and velocity $v_4 > v_3$. Note that the state $x_0(\cdot)$ is strictly monotonically increasing with time, as $\dot{x}_0(t) = x_1(t) > 0$ for all $t \in (0, T)$. We include condition (7.4) by additional interior point constraints

$$0 \leq r^{\text{ieq}}(\boldsymbol{x}(\tilde{t}_4)) = v_4 - x_1(\tilde{t}_4), \tag{7.5a}$$
$$0 = r^{\text{eq}}(\boldsymbol{x}(\tilde{t}_4)) = S_4 - x_0(\tilde{t}_4), \tag{7.5b}$$

assuming that the point of the track S_4 will be reached within the stage $[\tilde{t}_3, \tilde{t}_5]$. For a suitable choice of (S_4, v_4) this holds of course true. We do not change anything in the initialization resp. in the parameters of our method and obtain for $S_4 = 1200$ and $v_4 = 22/\gamma$ the optimal solution for problem (7.1) with the additional interior point constraints (7.5) as

$$\begin{aligned} w(t) \;=\; & \mathcal{S}(1,2,1,3,4,2,1,3,4; \\ & 2.86362, 10.722, 15.3108, 5.81821, \\ & 1.18383, 2.72451, 12.917, 5.47402, 7.98594). \end{aligned} \tag{7.6}$$

The corresponding trajectory with $\Phi = 1.3978$ is plotted in figure 7.3. Compared to (7.3), solution (7.6) has changed the switching structure. To meet the point constraint, the velocity has to be reduced by an additional coasting and braking arc. After this track point S_4, the parallel mode speeds up as soon as possible and the series mode guarantees that the velocity is high enough to reach the next station in time.
Not only the additional constraint influences the optimal switching structure, but also the values of the parameters. For a speed limit at a track point in the first half of the way, say $S_4 = 700$, we obtain the solution

$$\begin{aligned} w(t) \;=\; & \mathcal{S}(1,2,1,3,2,1,3,4; \\ & 2.98084, 6.28428, 11.0714, 4.77575, \\ & 6.0483, 18.6081, 6.4893, 8.74202) \end{aligned} \tag{7.7}$$

that is plotted in figure 7.4. For this solution there is only one braking arc ($w(t) = 4$) left. The reason is that the speed limit comes early enough such that the main distance can be covered afterwards and no high speed at the beginning, followed by braking, which is very energy consuming, is necessary. On the other hand, the braking arc at the end of the time horizon is longer, as we have an increased velocity with respect to solution (7.6) for all $t \geq 40$. This can be seen in a direct comparison in figure 7.9. The energy consumption is $\Phi = 1.32518$, thus lower than for the constraint at $S_4 = 1200$.
Point constraints like (7.4) may be quite typical for practical subway problems, e.g., when parts of the track are not in the best shape. Another typical restriction would be path constraints on subsets of the track. We will consider a problem with additional path constraints

$$x_1(t) \leq v_5 \;\text{ if }\; x_0(t) \geq S_5. \tag{7.8}$$

We include condition (7.8) by one additional path and one additional interior point constraint

$$0 \;\leq\; c(\boldsymbol{x}, t) = v_5 - x_1(t), \quad t \in [\tilde{t}_4, T] \tag{7.9a}$$

$$0 \;=\; r^{\text{eq}}(\boldsymbol{x}(\tilde{t}_4)) = S_5 - x_0(\tilde{t}_4), \tag{7.9b}$$

assuming again that the point of the track S_5 will be reached within the stage $[\tilde{t}_3, \tilde{t}_5]$. The additional path constraint changes the qualitative behavior of the relaxed solution. While all solutions considered this far were bang–bang and the main work consisted in finding the switching points, we now have a constraint–seeking arc. Figure 7.5 shows the relaxed solution. The path constraint (7.9) is active on a certain arc and determines the values of series mode and coasting. The sum of these two yields $\dot{x}_1 \equiv 0$, ensuring $x_1(t) = v_5$. Any optimal solution will look similar on this arc, no matter how often we refine the grid. We showed in preceding chapters that it is possible to approximate this singular solution arbitrarily close. This implies a fast switching between the two operation modes, though, which is not suited for practical purposes. Our algorithm allows to define a tolerance ε such that a compromise is found between a more energy–consuming operation mode which needs only few switches and is therefore more convenient for driver and passengers and an operation mode consuming less energy but switching more often to stay closer to the relaxed optimal solution.

By a refinement of the grid we get an estimate for Φ^*. The optimal solutions for refined grids yield a series of monotonically decreasing objective function values

$$1.33108, 1.31070, 1.31058, 1.31058, \ldots \tag{7.10}$$

We use the different grids to use rounding strategy SUR-SOS1 on them and initialize a switching time optimization with it. On the coarsest grid we obtain a solution that may only switch once between acceleration in series mode and coasting. The optimal solution looks thus as depicted in figure 7.6 — the velocity is reduced by braking strictly below the velocity constraint, such that it touches the constraint exactly once before the final coasting and braking to come to a hold begins. This solution is given by

$$\begin{aligned} w(t) \;=\; & \mathcal{S}(1,2,1,3,4,1,3,4; \\ & 2.68054, 13.8253, 12.2412, 4.03345, \\ & 1.65001, 15.3543, 7.99192, 7.22329) \end{aligned} \tag{7.11}$$

and yields an energy consumption of $\Phi = 1.38367$. This value is quite elevated compared to (7.10). If we use the same approach on refined grids we obtain

$$\begin{aligned} w(t) \;=\; & \mathcal{S}(1,2,1,3,4,1,3,1,3,1,3,4; \\ & 2.74258, 12.7277, 13.6654, 4.57367, \\ & 1.08897, 1.77796, 1.35181, 6.41239, \\ & 1.34993, 6.40379, 5.43439, 7.47134) \end{aligned} \tag{7.12}$$

with $\Phi = 1.32763$ depicted in figure 7.7 respectively

$$\begin{aligned} w(t) \;=\; & \mathcal{S}(1,2,1,3,4,1,3,1,3,1,3,1,3,1,3,1,3,4; \\ & 2.74458, 12.5412, 13.5547, 5.08831, \\ & 0.964007, 0.0571219, 0.739212, 3.56618, \\ & 0.744176, 3.58963, 0.745454, 3.59567, \\ & 0.71566, 3.45484, 0.111917, 0.549478, \\ & 4.69464, 7.54318) \end{aligned} \tag{7.13}$$

with $\Phi = 1.31822$ depicted in figure 7.8. An additional refinement yields a solution with 51 switches and $\Phi = 1.31164$ which is already quite close to the limit of (7.10). The results show the strength of our approach. Neglecting numerical problems when stage lengths become to small, we may approximate the singular solution arbitrarily close. As this often implies a large number of switchings, one may want to obtain a solution that switches less. Our approach allows to generate candidate solutions with a very precise estimation of the gap between this candidate and an optimal solution.

7.2 Phase resetting of calcium oscillations

Biological rhythms as impressing manifestations of self–organized dynamics associated with the phenomenon life have been of particular interest since quite a long time, see, e.g., Goldbeter (1996). Even before the mechanistic basis of certain biochemical oscillators was elucidated by molecular biology techniques, their investigation and the issue of perturbation by external stimuli has attracted much attention. Reasoning that limit cycle attractors are topologically equivalent to the circle

$$S^1 = \{\boldsymbol{x} \in \mathbb{R}^n \mid \; ||\boldsymbol{x}|| = 1\},$$

Winfree could rigorously prove via topological arguments that for particular phase resetting behavior there exists a critical stimulus, depending on strength and timing, which annihilates the oscillations completely. Without further stimuli, oscillations will finally be regained due to the instability of the steady state, but with an indefinite phase relation to the original oscillation. This can be seen as a kind of phase resetting. This situation corresponds to a stimulus–timing–phase singularity, see Winfree (2001) for a comprehensive overview and in–depth discussion. By determining families of phase resetting curves depicting the phase shift as a function of stimulus strength and timing, it is sometimes possible to identify such singularities either experimentally or theoretically by model–based numerical simulations. This can be done by finding the transition from so called type-1 to type-0 phase resetting curves depending on the stimulus strength, see Winfree (2001).
However, in complex multi–component systems occurring in cell biology the overwhelming variety of the kind, strength and timing of possible stimuli make simulation–based approaches via phase resetting curves impractical. For this reason, a systematic

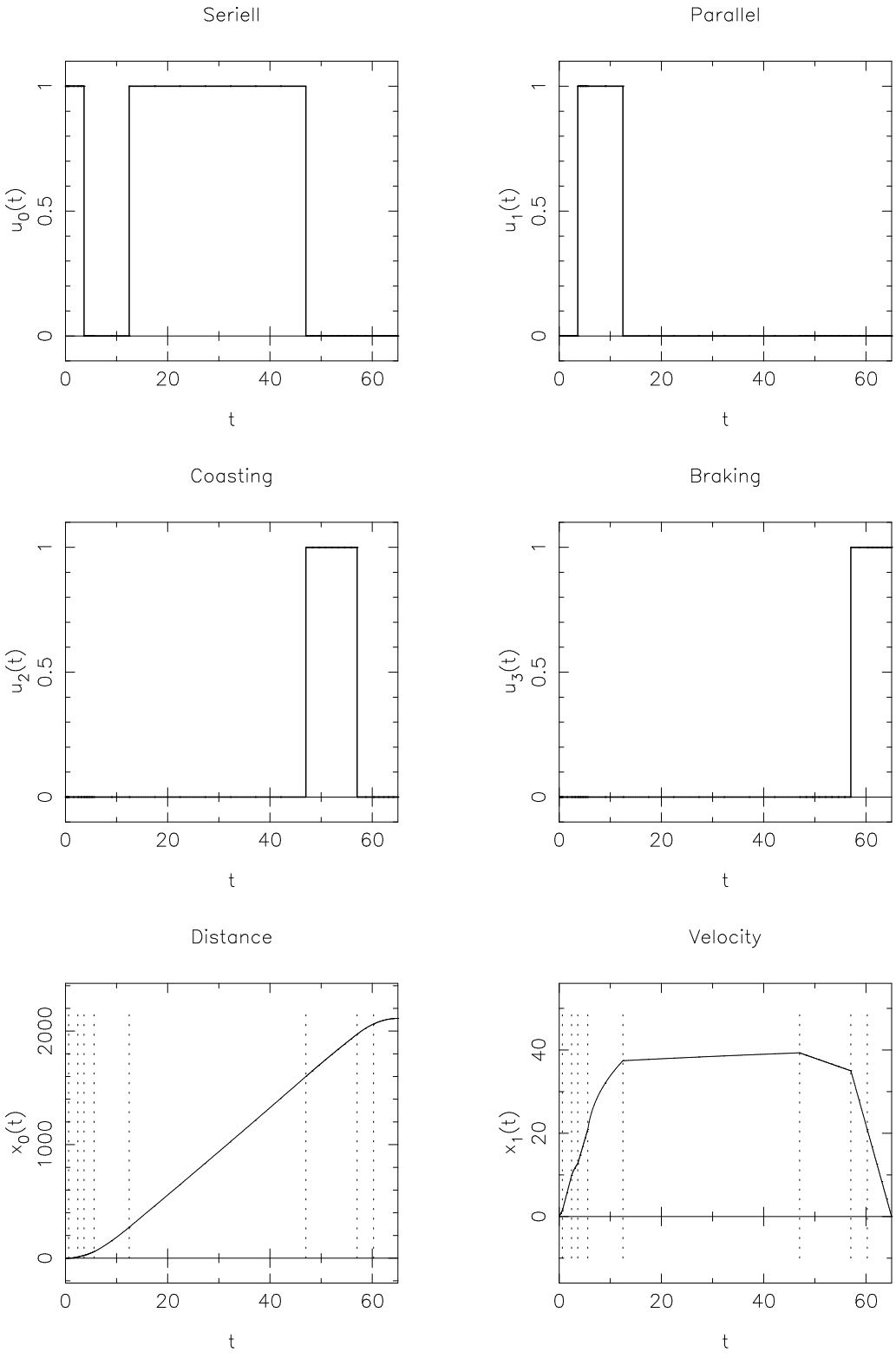

Figure 7.2: The controls for operation in series, $\tilde{w}_1(\cdot)$, in parallel, $\tilde{w}_2(\cdot)$, coasting, $\tilde{w}_3(\cdot)$ and braking, $\tilde{w}_4(\cdot)$. The last row shows covered distance $x_0(\cdot)$ and velocity $x_1(\cdot)$. This is the optimal trajectory for problem (7.1), given by (7.3).

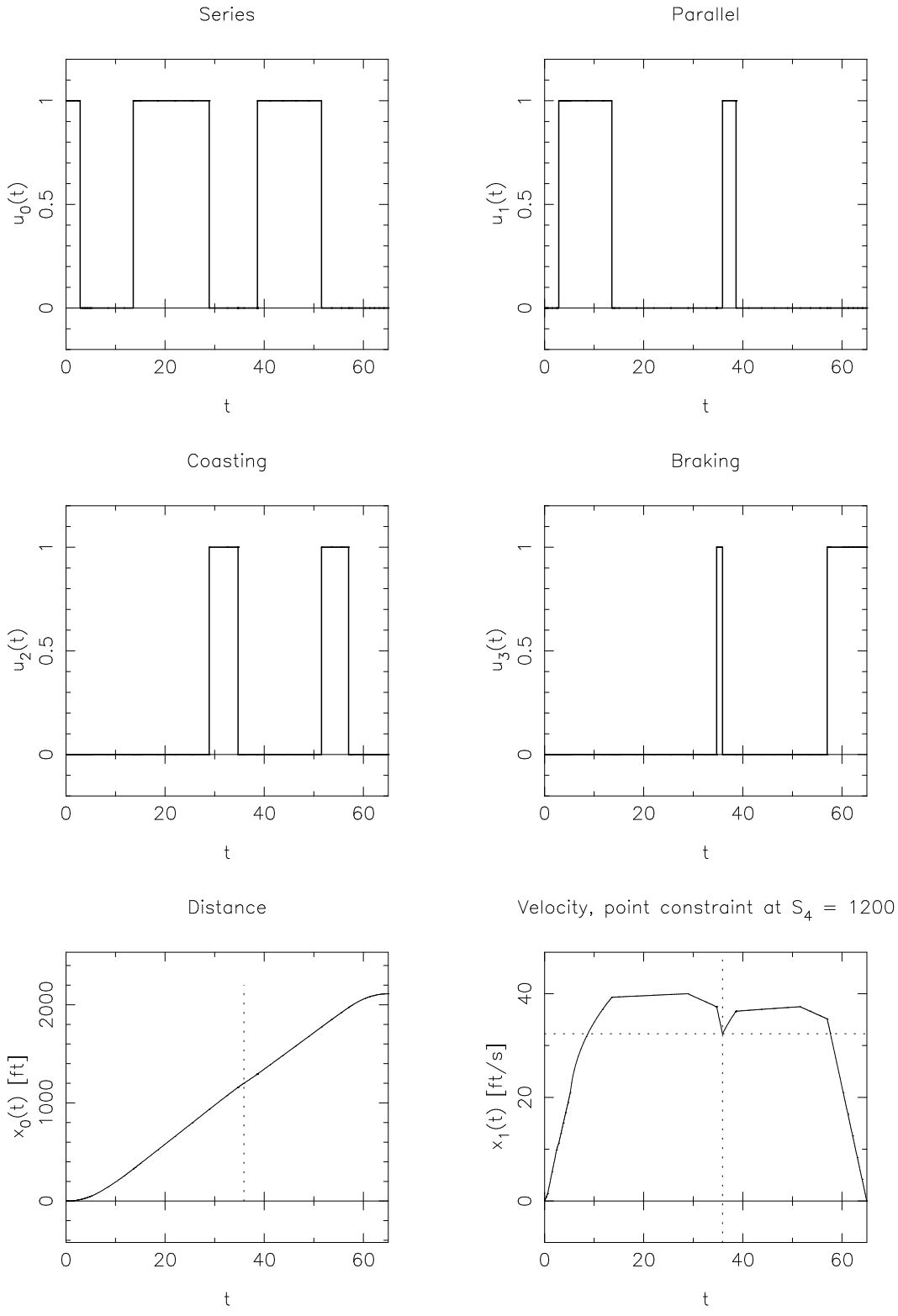

Figure 7.3: As in figure 7.2. This is the optimal trajectory for problem (7.1) with the additional **point constraints** (7.5), given by (7.6). The vertical dotted line indicates the time $\tilde{t}_4$ when the track point $x_0(\tilde{t}_4) = S_4 = 1200$ is reached. The horizontal dotted line shows the constraint velocity $v_4 = 22/\gamma$.

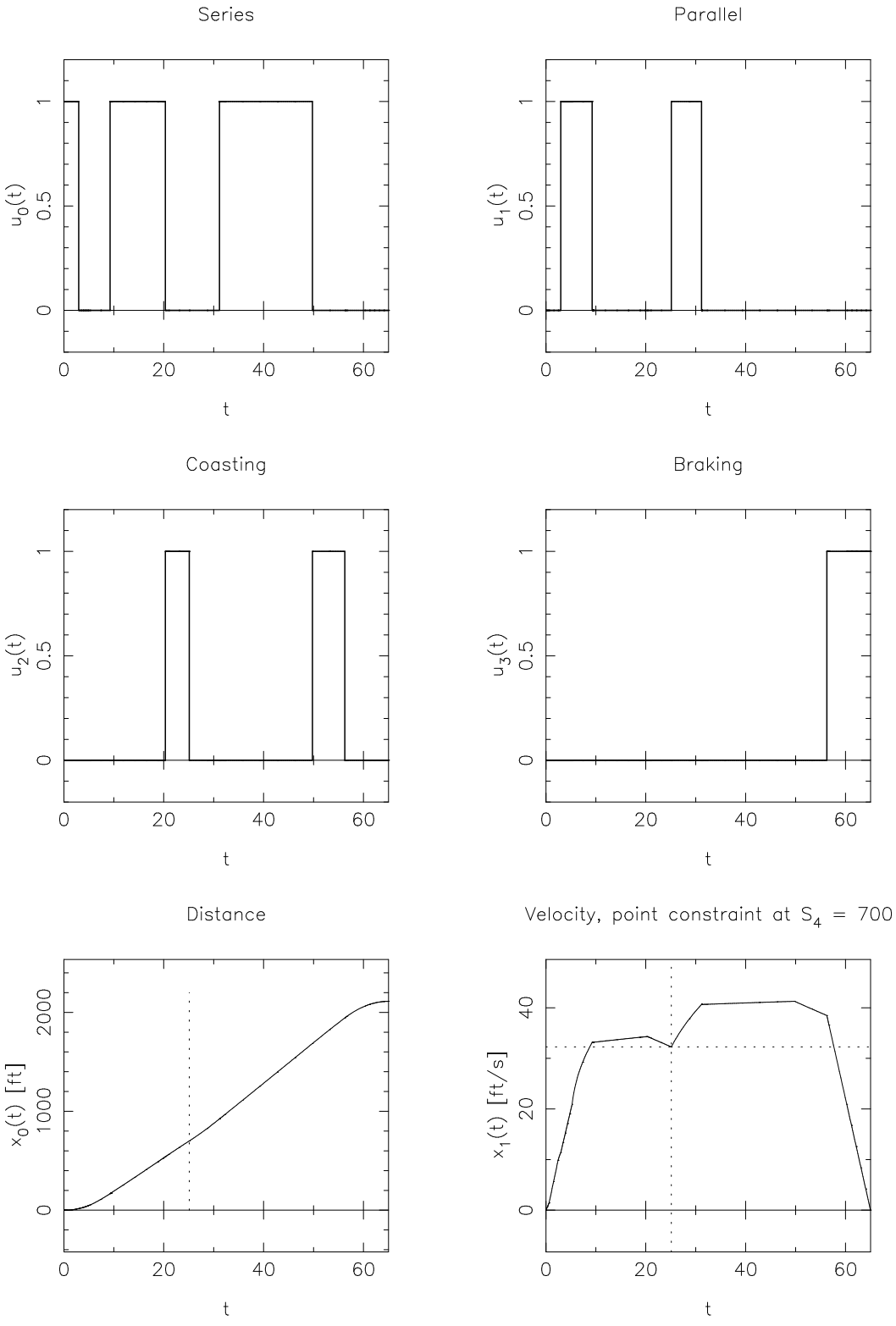

Figure 7.4: As in figure 7.3, but this time with a restriction at the track point $x_0(\tilde{t}_4) = S_4 = 700$ instead of $S_4 = 1200$. The first braking arc disappears, therefore the braking arc at the end of the track has to be longer.

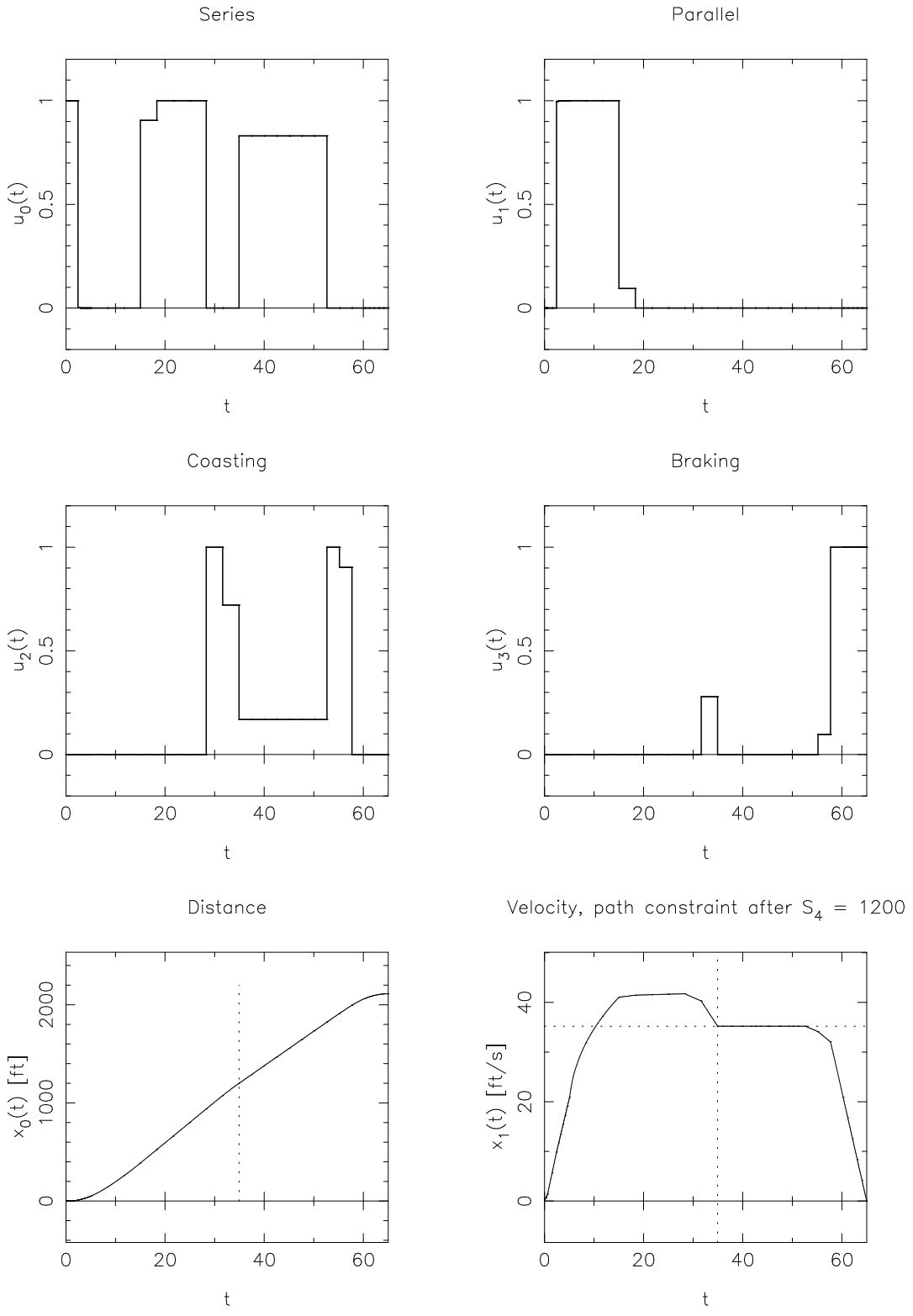

Figure 7.5: The plots are as before. This is the optimal trajectory for the *relaxed* problem (7.1) with the additional **path constraints** (7.9). Note that this constraint is active on a certain arc and determines the values of series mode and coasting. The sum of these two yields $\dot{x}_1 \equiv 0$. Any optimal solution will look similar on this arc, no matter how often we refine the grid. The energy consumption is $\Phi = 1.33108$. After one refinement it is $\Phi = 1.31070$, after two refinements $\Phi = 1.31058$.

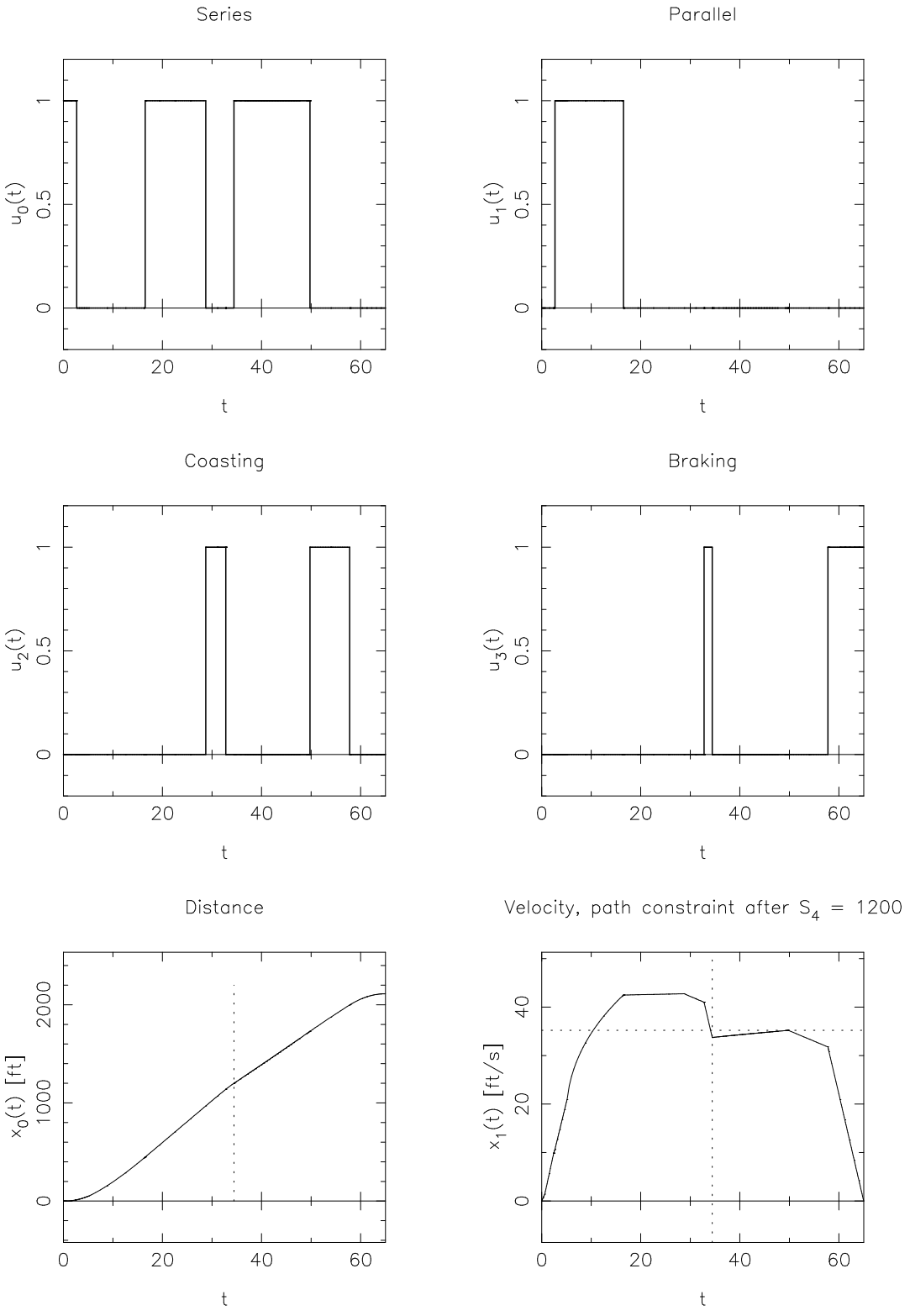

Figure 7.6: The plots are as before. This is an admissible trajectory for the *integer* problem (7.1) with the additional constraints (7.9). Note that the path constraint is active on only one touch point. The controls are chosen such that the state x_1 is below the constraint at the beginning of the arc and only touches it once at its end. The energy consumption is $\Phi^1 = 1.38367$.

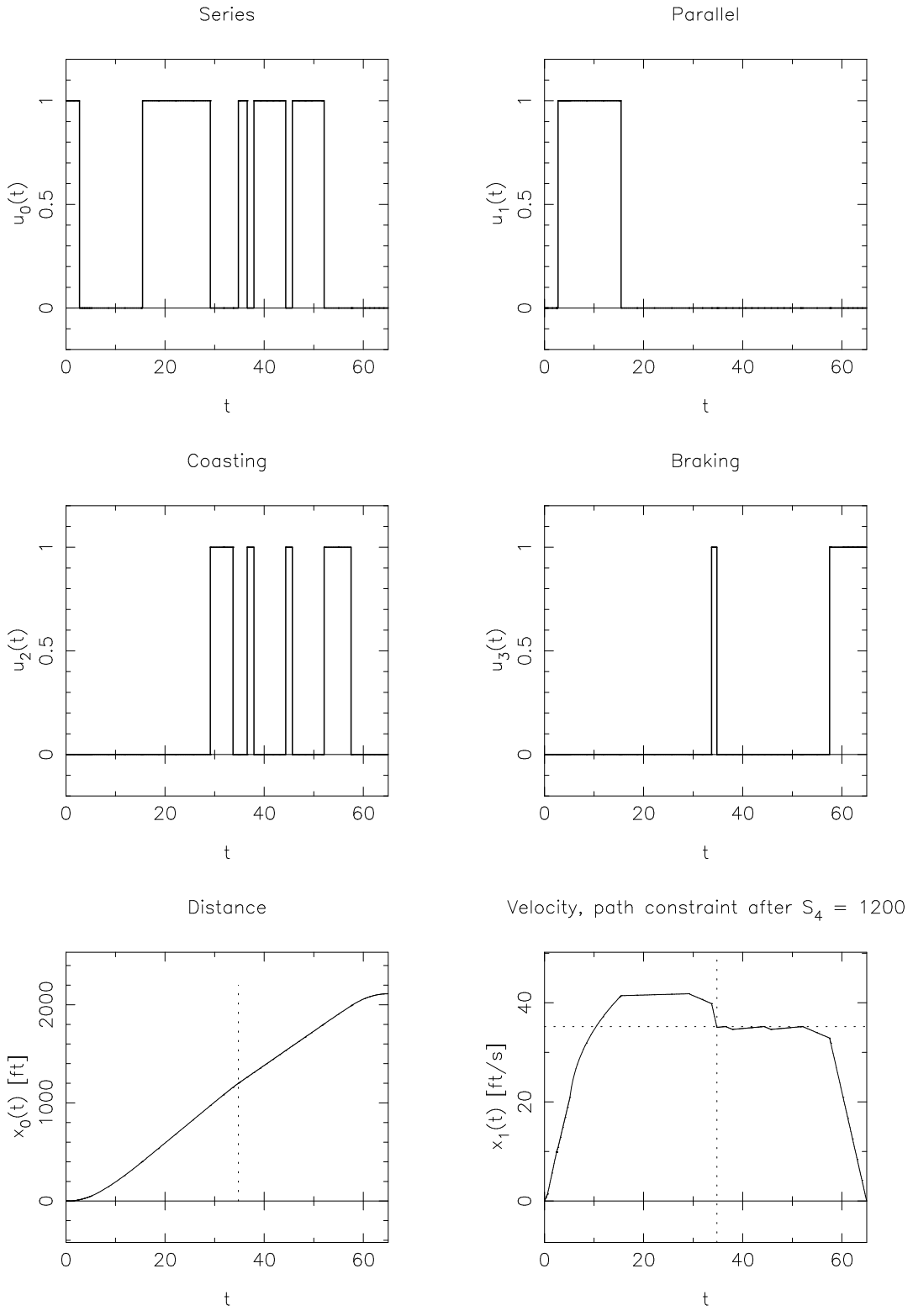

Figure 7.7: As in figure 7.6. Note that the path constraint is now active on three touch points. This arc is better approximated, therefore the energy consumption $\Phi^2 = 1.32763$ is better than $\Phi^1 = 1.38367$.

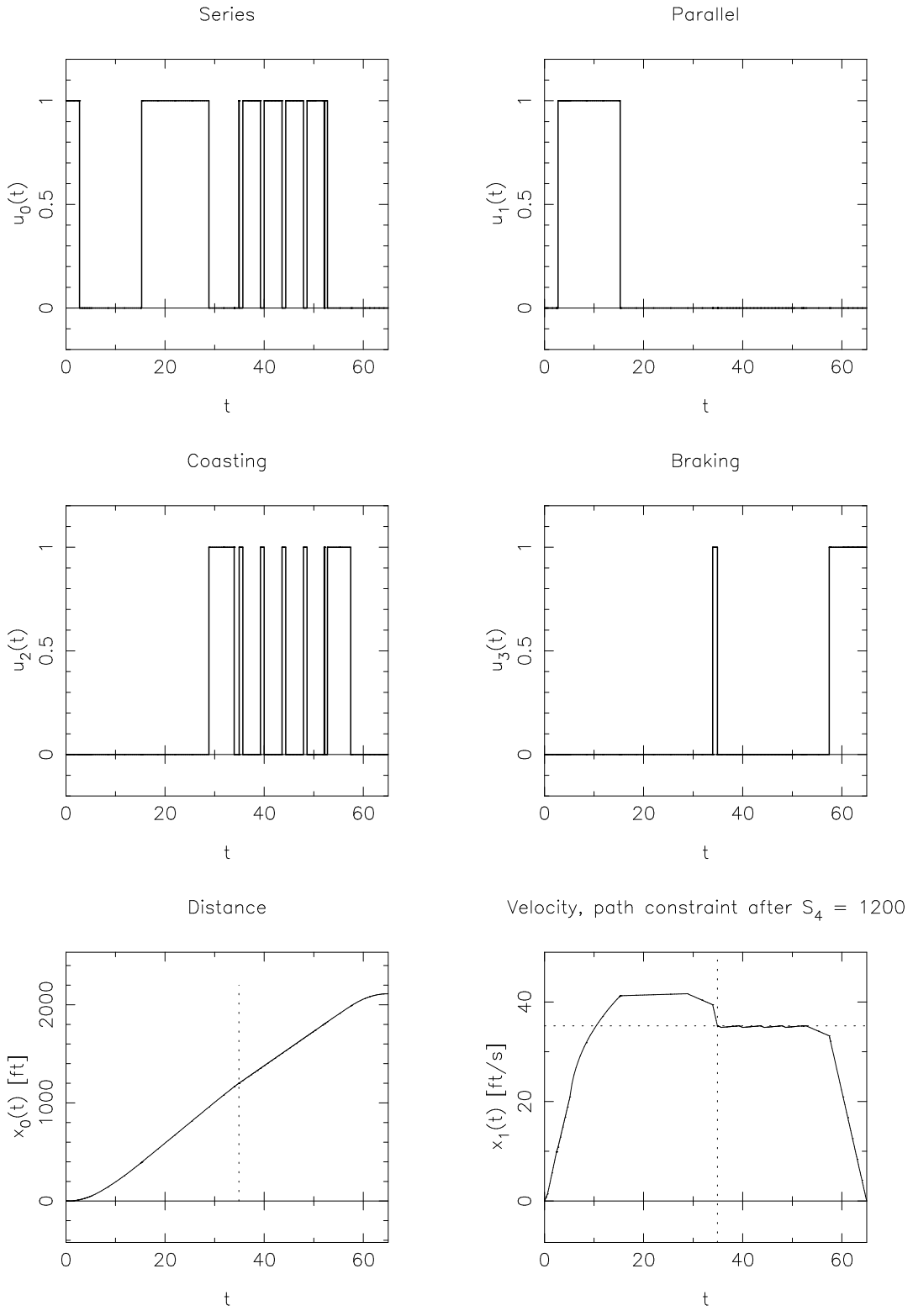

Figure 7.8: As in figure 7.6. Note that the path constraint is now active on six touch points. The constraint–arc is even better approximated than before, therefore the energy consumption $\Phi^3 = 1.31822$ is better than $\Phi^2 = 1.32763$.

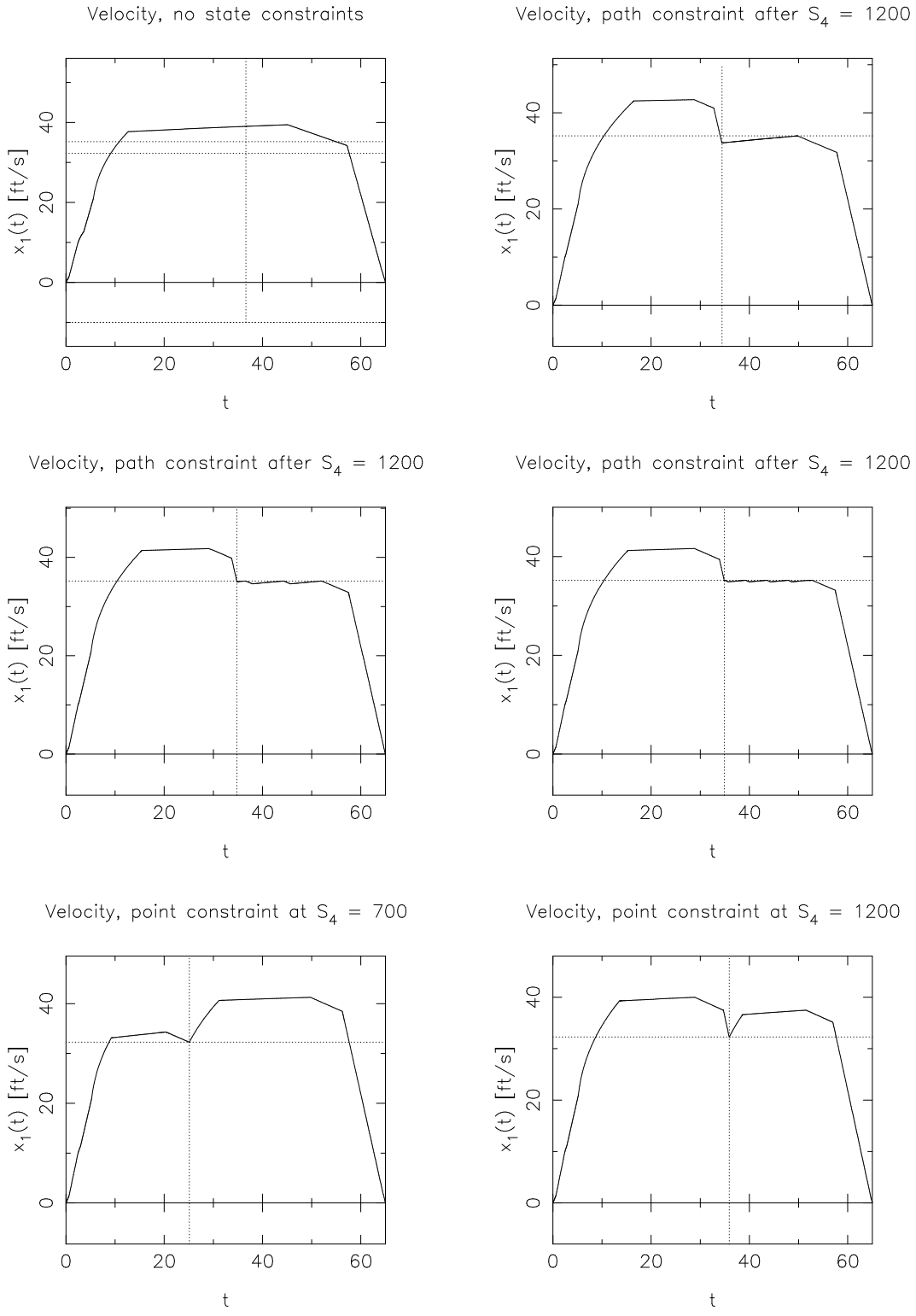

Figure 7.9: Final comparison of the different states $x_1(\cdot)$. Top left: the state trajectory for problem (7.1) without constraints on the velocity. Top right and two plots in the middle: solutions for the problem with path constraint, with increasing accuracy of the approximation of the singular solution. Bottommost plots: optimal trajectories for point constraint. The vertical dotted lines show when $x_0 = 1200$ resp. $x_0 = 700$ are reached. The horizontal lines show the velocities v_4 resp. v_5.

and automatic algorithmic procedure for identification of the phase singularities is attractive. We demonstrate how our methods can be applied to limit cycle oscillations coexisting with stable or unstable steady states. This is possible in any kind of physical or chemical system, if a kinetic model is available.
As an important biochemical example, we choose a calcium oscillator model describing intracellular calcium spiking in hepatocytes induced by an extracellular increase in adenosine triphosphate (ATP) concentration to demonstrate the performance of our optimal control method. The calcium signaling pathway is initiated via a receptor activated G–protein inducing the intracellular release of inositol triphoshate (IP_3) by phospholipase C. The IP_3 triggers the opening of endoplasmic reticulum (ER) and plasma membrane calcium channels and a subsequent inflow of calcium ions from intracellular and extracellular stores leading to transient calcium spikes. The ODE model for the calcium oscillator consists of four variables describing the time–dependent concentrations of activated G-protein $x_0(\cdot)$, active phospholipase C represented by $x_1(\cdot)$, intracellular calcium $x_2(\cdot)$ and intra–ER calcium $x_3(\cdot)$ respectively. The model takes into account known feedback–regulations of the pathway, in particular CICR (calcium induced calcium release), and active transport of calcium from the cytoplasm across both ER–membrane and plasma membrane via SERCA (sarco–endoplasmic reticulum Ca^{2+}–ATPase) and PMCA (plasma membrane Ca^{2+}–ATPase) pumps. We leave away the argument (t) for notational convenience. The dynamics are then described by the following ODE system $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$

$$\dot{x}_0 = k_1 + k_2 x_0 - \frac{k_3 x_0 x_1}{x_0 + K_4} - \frac{k_5 x_0 x_2}{x_0 + K_6} \tag{7.14a}$$

$$\dot{x}_1 = (1 - u_2) \cdot k_7 x_0 - \frac{k_8 x_1}{x_1 + K_9} \tag{7.14b}$$

$$\dot{x}_2 = \frac{k_{10} x_1 x_2 x_3}{x_3 + K_{11}} + k_{12} x_1 + k_{13} x_0 - \frac{k_{14} x_2}{u_1 x_2 + K_{15}} - \frac{k_{16} x_2}{x_2 + K_{17}} + \frac{x_3}{10} \tag{7.14c}$$

$$\dot{x}_3 = -\frac{k_{10} x_1 x_2 x_3}{x_3 + K_{11}} + \frac{k_{16} x_2}{x_2 + K_{17}} - \frac{x_3}{10} \tag{7.14d}$$

with initial values $\boldsymbol{x}(0) = \boldsymbol{x_0}$ and fixed parameters $\boldsymbol{p} = (k_1, \ldots, K_{17})^T$ that are given in appendix D and two external controls u_1 and u_2. Modeling details including a comprehensive discussion of parameter values and the dynamical behavior observed in simulations with a comparison to experimental observations can be found in Kummer *et al.* (2000). In our study the model is identical to the one derived there, except for an additional first–order leakage flow of calcium from the ER back to the cytoplasm, which is modeled by $\pm\frac{x_3}{10}$ in equations 3 and 4 of system (7.14). It reproduces well experimental observations of cytoplasmic calcium oscillations as well as bursting behavior and in particular the frequency encoding of the triggering stimulus strength, which is a well known mechanism for signal processing in cell biology, see Berridge (1997).
As a source of external control we additionally introduced a temporally varying concentration u_1 of an uncompetitive inhibitor of the PMCA ion pump and an inhibitor u_2 of PLC activation by the G-protein. The inhibitor influence u_1 is modeled ac-

cording to standard Michaelis-Menten kinetics for uncompetitive enzyme inhibition in (7.14c), compare Bisswanger (2002). The influence on PLC activation is modeled by a multiplication with $(1 - u_2)$ in (7.14b). According to Kummer *et al.* (2000)) u_2 indicates the blocking extent of the PLC activation, where $u_2 \equiv 1$ corresponds to full inhibition. In practice, La^{3+} is often used as a potent PMCA inhibitor, Morgan & Jacob (1998), and RGS proteins (regulators of G-protein signaling) are known as inhibitors of activated G-protein effects like PLC activation.
The aim of our control approach is to identify strength and timing of inhibitor stimuli (u_1, u_2) that lead to a phase singularity which annihilates the intracellular calcium oscillations. We address this problem by formulating an objective function that aims at minimizing the state deviation from the desired steady state integrated over time. We are interested in particular control functions (u_1, u_2) that switch between the value $u_1 \equiv 1$ (zero PMCA inhibitor concentration) and $u_1 = u_1^{\max} \geq 1$ corresponding to a maximum PMCA inhibitor concentration and $u_2 = 0$ and $u_2 = 1$ for zero and full RGS inhibition of PLC activation respectively. By setting $u_1 \equiv 1 + w_1(u_1^{\max} - 1)$ and $u_2 \equiv w_2$ this can be formulated with the help of two binary control functions (w_1, w_2). We add two terms $p_1 w_1$ and $p_2 w_2$ with $p_1, p_2 \geq 0$ given in the appendix to the objective functional for two purposes. First, we want to favor solutions that use small total amounts of inhibitors. Second, this leads to a regularization of the unstable problem in the sense that solutions with short stimuli at the end of the control horizon, that are local minima, are avoided. Leaving the intensity $u_{\max}$ as a time–independent degree of freedom, i.e., as an additional free parameter, we finally obtain the mixed–integer optimal control problem to minimize

$$\min_{\boldsymbol{x},\boldsymbol{w}} \quad \int_0^T \sum_{i=0}^{3} \left(x_i(t) - x_i^s\right)^2 + p_1 w_1(t) + p_2 w_2(t) \, \mathrm{d}t \tag{7.15a}$$

subject to the ODE

$$\dot{x}_0 = k_1 + k_2 x_0 - \frac{k_3 x_0 x_1}{x_0 + K_4} - \frac{k_5 x_0 x_2}{x_0 + K_6} \tag{7.15b}$$

$$\dot{x}_1 = (1 - w_2) \cdot k_7 x_0 - \frac{k_8 x_1}{x_1 + K_9} \tag{7.15c}$$

$$\dot{x}_2 = \frac{k_{10} x_1 x_2 x_3}{x_3 + K_{11}} + k_{12} x_1 + k_{13} x_0 - \frac{k_{16} x_2}{x_2 + K_{17}} + \frac{x_3}{10} - \frac{k_{14} x_2}{(1 + w_1(u_1^{\max} - 1))x_2 + K_{15}} \tag{7.15d}$$

$$\dot{x}_3 = -\frac{k_{10} x_1 x_2 x_3}{x_3 + K_{11}} + \frac{k_{16} x_2}{x_2 + K_{17}} - \frac{x_3}{10} \tag{7.15e}$$

initial values

$$\boldsymbol{x}(t_0) = \boldsymbol{x_0}, \tag{7.15f}$$

bounds

$$1 \leq u_1^{\max} \leq 1.3, \tag{7.15g}$$

$$0 \leq x_0, x_1, x_2, x_3 \tag{7.15h}$$

and the integer constraints

$$w_1, w_2 \quad \in \quad \Omega(\Psi_{\text{free}}). \tag{7.15i}$$

The fixed parameter values $k_1, \ldots, K_{17}, p_1, p_2$, the initial values $\boldsymbol{x_0}$ and the reference values $\boldsymbol{x^s}$ are given in appendix D. The latter have been determined by using the XPPAUT software, Ermentrout (2002), by path-following of a Hopf-bifurcation through variation of the parameter k_2,

Remark 7.1 *Another possibility to formulate the objective function includes a scaling of the deviations, i.e.,*

$$\min_{\boldsymbol{x},\boldsymbol{w}} \quad \int_0^T \sum_{i=0}^{3} \left(\frac{x_i(t) - x_i^s}{x_i^s} \right)^2 + p_1 w_1(t) + p_2 w_2(t) \, \mathrm{d}t$$

instead of (7.15a).

The set of parameters and initial values gives rise to bursting–type limit cycle oscillations. Figure 7.10 shows this behavior in simulation results. Three scenarios are plotted, the first one for no inhibition at all, i.e., $w_1 \equiv 0$ and $w_2 \equiv 0$. The middle column shows the differential states $x_0(\cdot), x_1(\cdot), x_2(\cdot)$ and $x_3(\cdot)$ for $w_1 \equiv 1$, $u_1^{\max} = 1.3$ and $w_2 \equiv 0$, i.e., constant maximum inhibition of the PMCA ion pump and no channel blocking. The dotted horizontal lines show the reference state $\boldsymbol{x^s}$.
To solve problem (7.15) we proceed as follows. First we convexify the system with respect to the binary control functions as shown in section 4.1. w_2 enters linearly. The convexification with respect to w_1 gives

$$\begin{aligned} \dot{x}_2 \quad = \quad & \frac{k_{10} x_1 x_2 x_3}{x_3 + K_{11}} + k_{12} x_1 + k_{13} x_0 - \frac{k_{16} x_2}{x_2 + K_{17}} + \frac{x_3}{10} \\ & - \left(\tilde{w}_1 \frac{k_{14} x_2}{u_1^{\max} x_2 + K_{15}} + (1 - \tilde{w}_1) \frac{k_{14} x_2}{x_2 + K_{15}} \right) \end{aligned} \tag{7.16}$$

The next step is then to solve a relaxed problem on a fixed grid $\mathcal{G}^0$. We choose a grid with $n_{\text{ms}} = 25$. We will first fix w_2 to zero and investigate the optimal trajectory for a one–dimensional control. Figure 7.11 shows the optimal control functions for the relaxed system on the grid $\mathcal{G}^0$ and grids $\mathcal{G}^i$ obtained by an iterative refinement with adaptive mode 2. The objective function values Φ^i on grid $\mathcal{G}^i$ decrease as

$$\begin{aligned} \Phi^0 \quad &= \quad 1760.13, \Phi^1 = 1744.77, \Phi^2 = 1725.49, \Phi^3 = 1719.27, \Phi^4 = 1715.38, \\ \Phi^5 \quad &= \quad 1713.72, \Phi^6 = 1712.60, \Phi^7 = 1712.14, \Phi^8 = 1711.81, \Phi^9 = 1711.68. \end{aligned}$$

The iterative procedure to determine the optimal trajectory on grid $\mathcal{G}^9$ takes 35 seconds and 84 SQP iterations. Having a closer look at the control functions for the relaxed problem, we decide to apply a simple rounding heuristics to obtain a binary admissible trajectory from the relaxed solutions. Rounding yields a considerable gap

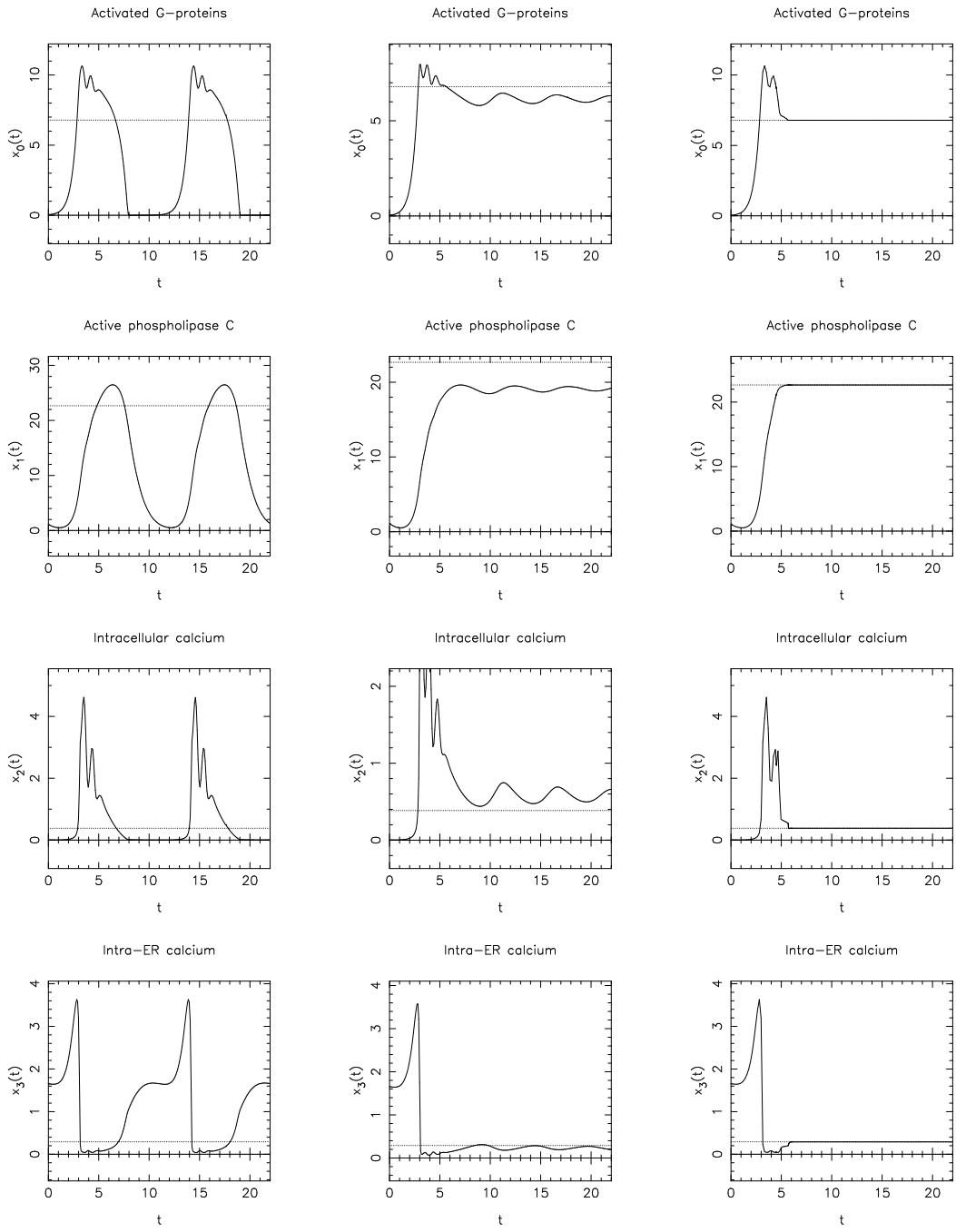

Figure 7.10: Simulation of system (7.15). The rows show the four differential states $x_0, \ldots, x_3$. The dotted horizontal lines indicate the reference state $\boldsymbol{x^s}$. The leftmost column shows the states for no inhibition, $w_1 \equiv 0$ and $w_2 \equiv 0$. The middle column shows a simulation for $w_1 \equiv 1$, $u_1^{\max} = 1.3$ and $w_2 \equiv 0$, i.e., a constant maximum inhibition of the PMCA ion pump. The rightmost column shows the states corresponding to the optimal trajectory (7.17).

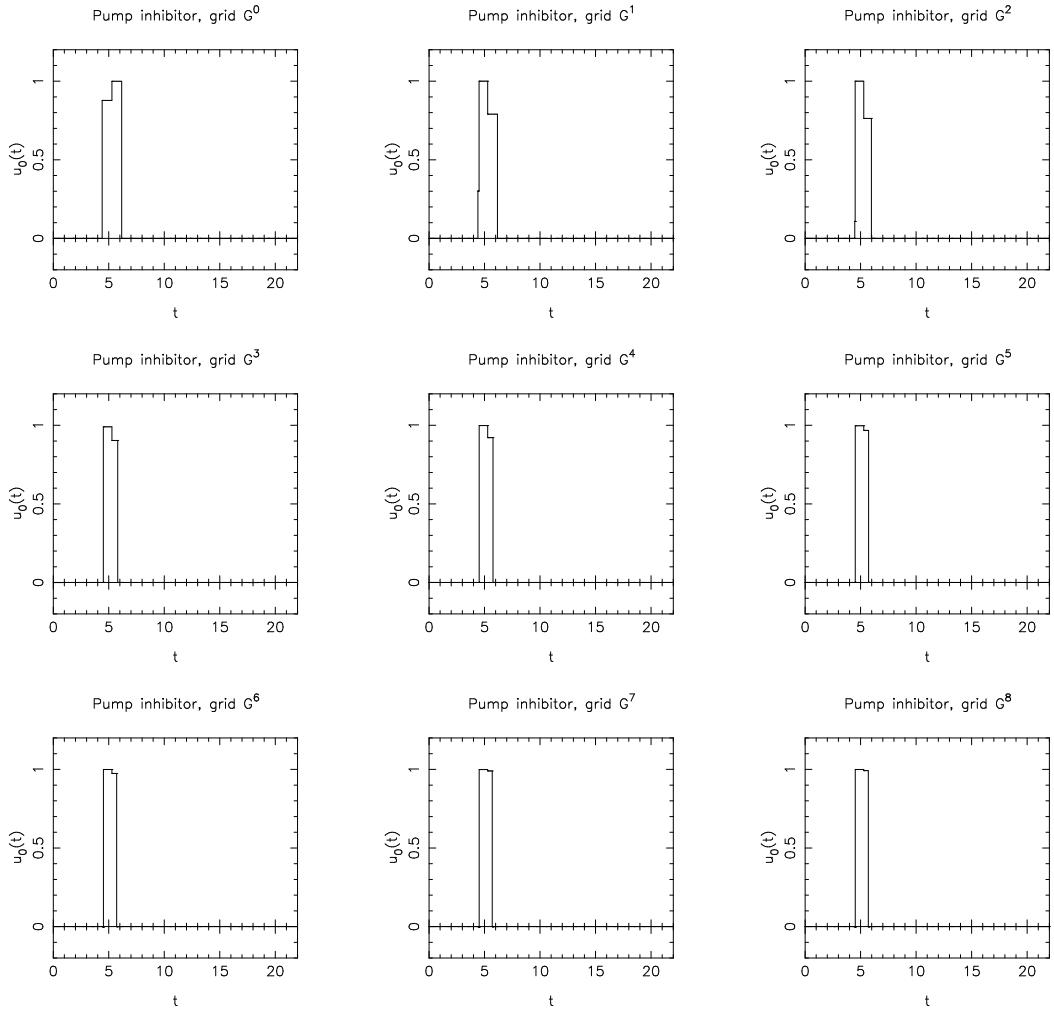

Figure 7.11: Relaxed solutions on different adaptive grids $\mathcal{G}^0$ to $\mathcal{G}^8$.

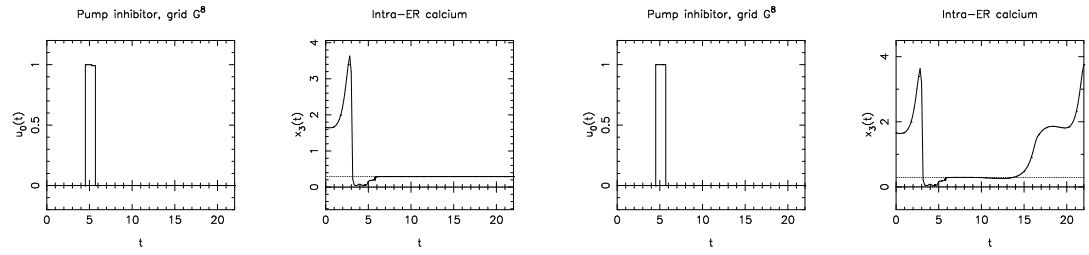

Figure 7.12: Relaxed (left) and rounded (right) control function $w_1(\cdot)$ and corresponding state $x_3(\cdot)$ on the grid $\mathcal{G}^8$.

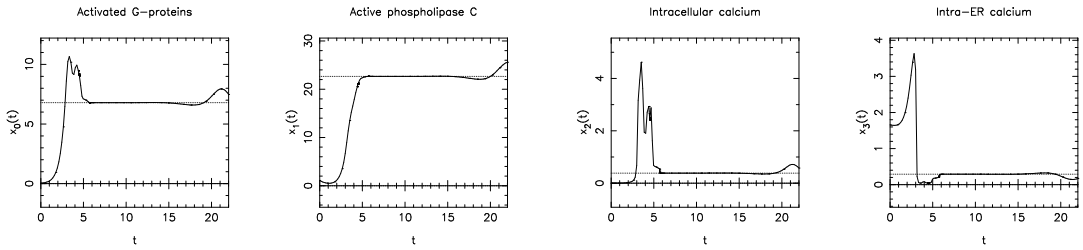

Figure 7.13: Corresponding states $x_0, \ldots, x_3$ of solution (7.17), integrated over the whole time horizon after rounding to six digits.

in the objective function values, though. Figure 7.12 shows the relaxed and a rounded control function on the grid $\mathcal{G}^8$ with one corresponding differential state trajectory. The objective value of this rounded, binary admissible trajectory is $\bar{\Phi} = 4143.78$, which corresponds to a high integrated deviation from the steady state.
This is also the reason, why so many adaptive iterations are necessary to come close to a binary admissible solution with an acceptable system response. Bisection, i.e., adaptive mode 0, is even slower than adaptive mode 2. After six refinements the objective value is still up at 1726.87, worse than after two refinements with adaptive mode 2. The switching time approach yields local minima with an objective value way above 4000 for all initializations with the rounded solutions on the grids $\mathcal{G}^0$ to $\mathcal{G}^7$. In iteration 8 we obtain finally the binary admissible result

$$w_1(t) = \mathcal{S}(0, 1, 0; 4.51958, 1.16726, 16.31317), \tag{7.17a}$$
$$w_2(t) = 0 \tag{7.17b}$$

with $\Phi = 1711.49$ and $u_1^{\max} = 1.11880$. The corresponding state trajectories are plotted in the rightmost column of figure 7.10.
Rounding to six digits in (7.17) and a single shooting integration, that is, with no matching tolerances at the multiple shooting nodes, leads to the trajectory depicted in figure 7.13. This trajectory leaves the unstable steady state earlier than in picture 7.10 and has an augmented objective function value. This is caused by the extreme sensitivity of the system to small perturbations in the stimulus.
Simulating the solution further in time, figure 7.14, we see that the limit cycle oscillations restart at about the end of the control horizon. This is different, if at least two stable steady states exist. In Lebiedz *et al.* (2006) our methods are applied not only to find a stimulus to switch from one limit cycle resp. steady state to another, but also to identify a second one, within the same optimal control problem, to switch back to the original periodic limit cycle.
The simple structure of solution (7.17) allows a more detailed investigation of the objective function landscape, as already performed in section 5.2. We vary the begin and the length of the stimulus over the feasible time domain, determine all other variables such that the trajectory is admissible and integrate the resulting ODE. Figure 7.16 shows the corresponding objective function values in two– respectively

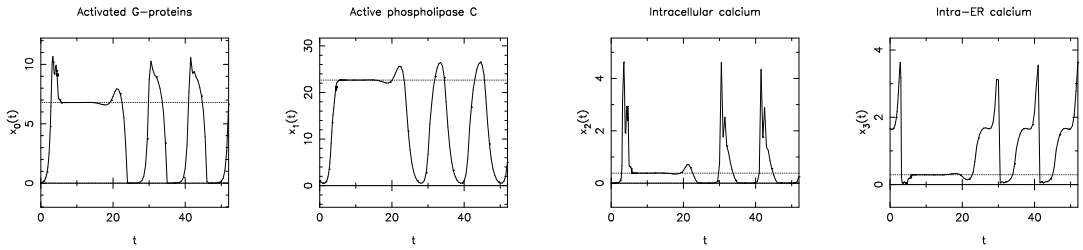

Figure 7.14: Restart of limit cycle oscillations, integration of (7.17) over the prolonged time interval $[0, 52]$.

three–dimensional plots.
Obviously the PMCA inhibitor suffices to reset the phase of the limit cycle oscillations. Still we apply our algorithm also to the two–dimensional case, where also an inhibition of the PLC activation by the G-protein is possible ($w_2(\cdot)$). With the same procedure as above, i.e., an iterative refinement of the grid, rounding and a switching time optimization until we reach the lower bound, we obtain the binary admissible result

$$\begin{aligned} \boldsymbol{w}(t) \;&=\; \mathcal{S}(\,(0,1),(0,0),(1,0),(0,0); \\ &\qquad 1.60631, 2.78085, 0.71340, 16.8994\,). \end{aligned} \tag{7.18}$$

with $\Phi = 1538.00$ and $u_1^{\max} = 1.13613$. This objective value neglects the term $p_1 w_1 + p_2 w_2$. Compared with the objective function value of $\Phi = 1604.13$ obtained by (7.17), again neglecting $p_1 w_1$, this is a considerable improvement from a mathematical point of view. The corresponding trajectory is plotted in figure 7.15.
The results obtained here for the calcium oscillator example demonstrate that we can successfully identify critical phase resetting stimuli leading to the (transient) annihilation of limit cycle oscillations by applying *MS MINTOC*. Based on detailed kinetic models such control strategies for complex self–organizing systems may turn out to be of great benefit in various applications ranging from physicochemical systems in technical processes, Kiss & Hudson (2003), to drug development and biomedical treatment strategies of dynamical diseases, see Walleczek (2000) or Petty (2004).
Despite this possibly still far ahead practical applicability of optimal control methods to biological systems, it may give an insight into underlying principles by analysis of optimal trajectories. Furthermore such models have interesting properties from a mathematical point of view. The instability of the steady state makes it comparatively hard to determine the switching points, although the switching structure is known in this case. We cannot make use of any switching time optimization approach without a very accurate initialization, otherwise we will run into one of the local minima connected to an early restart of the oscillation.
As already pointed out in section 5.2 and appendix B.4, typically a very high number of local minima occurs in a switching time problem formulation. An *a posteriori* analysis of the present case study shows that an optimization of the switching times

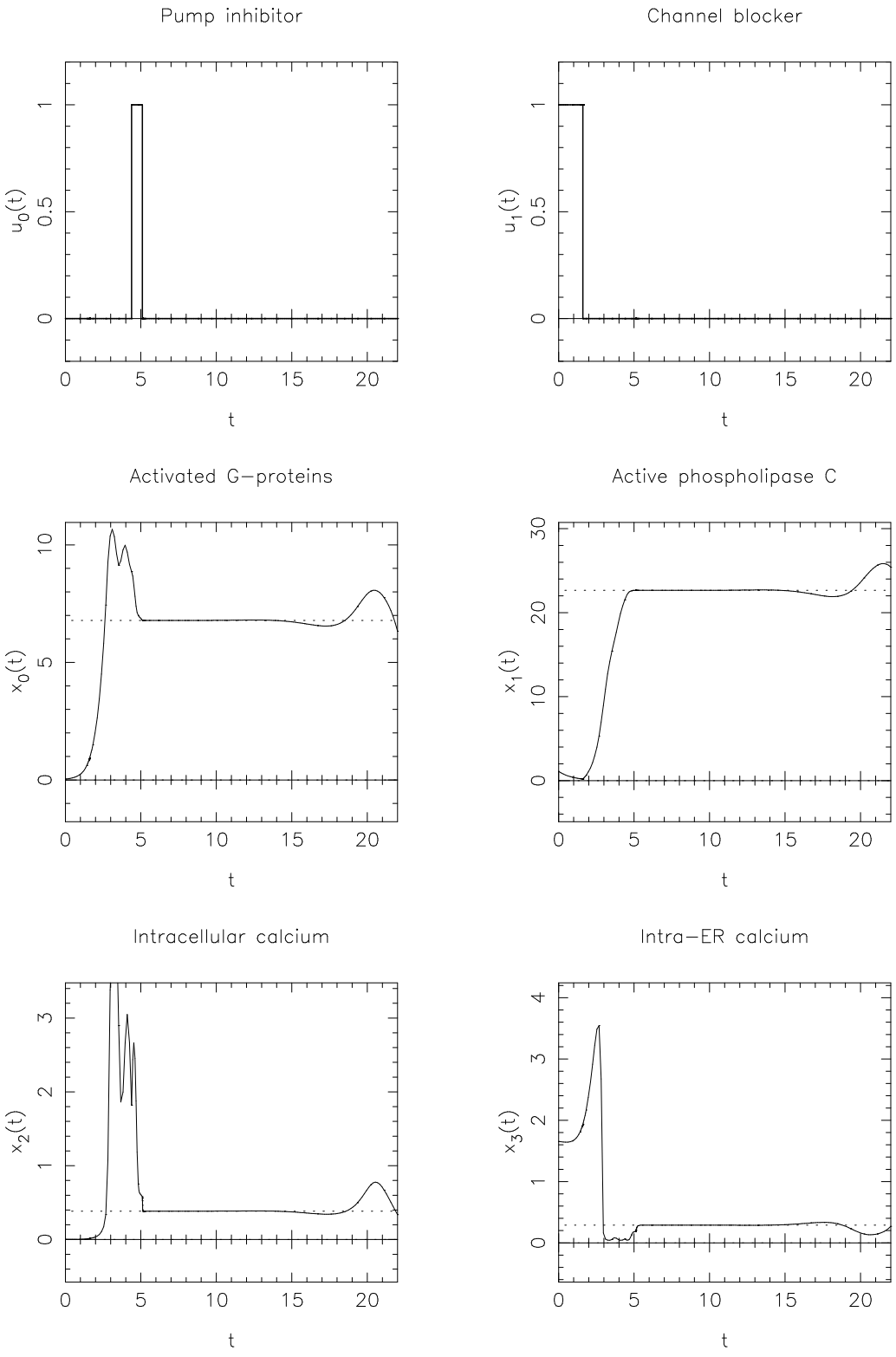

Figure 7.15: Optimal trajectory with two control functions.

of w_1 only yields acceptable results if the initial values are closer than 10^{-3} time units to the optimal values. For the behavior of our algorithm it is crucial that we use the underlying direct multiple shooting method, a method that has shown to be suitable for other unstable dynamical systems with complex self–organizing behavior before, Lebiedz & Brandt-Pollmann (2003) or Brandt-Pollmann *et al.* (2005).

7.3 Batch distillation with recycled waste cuts

In this section we will treat a mixed–integer optimal control problem from chemical engineering, namely a batch distillation process with recycled waste cuts. Waste cut recycling problems are treated, e.g., by Mayur *et al.* (1970) and Christensen & Jorgensen (1987) for binary batch distillation, where it is possible to find time optimal reflux policies in the framework of Pontryagin's maximum principle. Luyben (1988, 1990) treats a ternary mixture, but does not solve an optimal control problem. In Diehl *et al.* (2002) the authors optimize stage durations, recycling ratios and controls simultaneously for a ternary mixture, using a cost function that incorporates energy costs, product prices and possible waste cut disposal costs. The model presented there will be the basis for our study. We extend this model by additional degrees of freedom – the waste cuts may not only be reinserted as batch at the beginning of the production cuts to the reboiler, but distributed over time to any tray.
We will proceed as follows. We will first present the model of the process, followed by a review of the results of Diehl *et al.* that will serve as reference solutions. Then we formulate the mixed–integer optimal control problem and apply *MS MINTOC* to solve it.

The multiple–fraction batch distillation process has as its goal the separation of a mixture of n_{comp} components into different fractions of prespecified purity levels. We will refer to this mixture as *feed* in the following.
In this study we treat the complete separation of a ternary mixture, which is accomplished by four consecutive distillation phases: the lightest component is separated in a first production cut, remaining traces of it are removed in a following *waste cut*[1], which helps to attain the desired purity of the second lightest fraction in the following second production cut. The purity of the third and heaviest fraction is achieved by a last waste cut that removes remaining impurities from the bottoms product.
The process is controlled by the *reflux ratio* R, which can be varied over time. A sketch of the distillation column and the process under investigation is shown in Figure 7.17.
The mathematical model of the distillation column, developed, e.g., by Farhat *et al.* (1990), is based on the following simplifying assumptions:

- Ideal trays
- No liquid holdup

[1] also referred to as *slop cut*

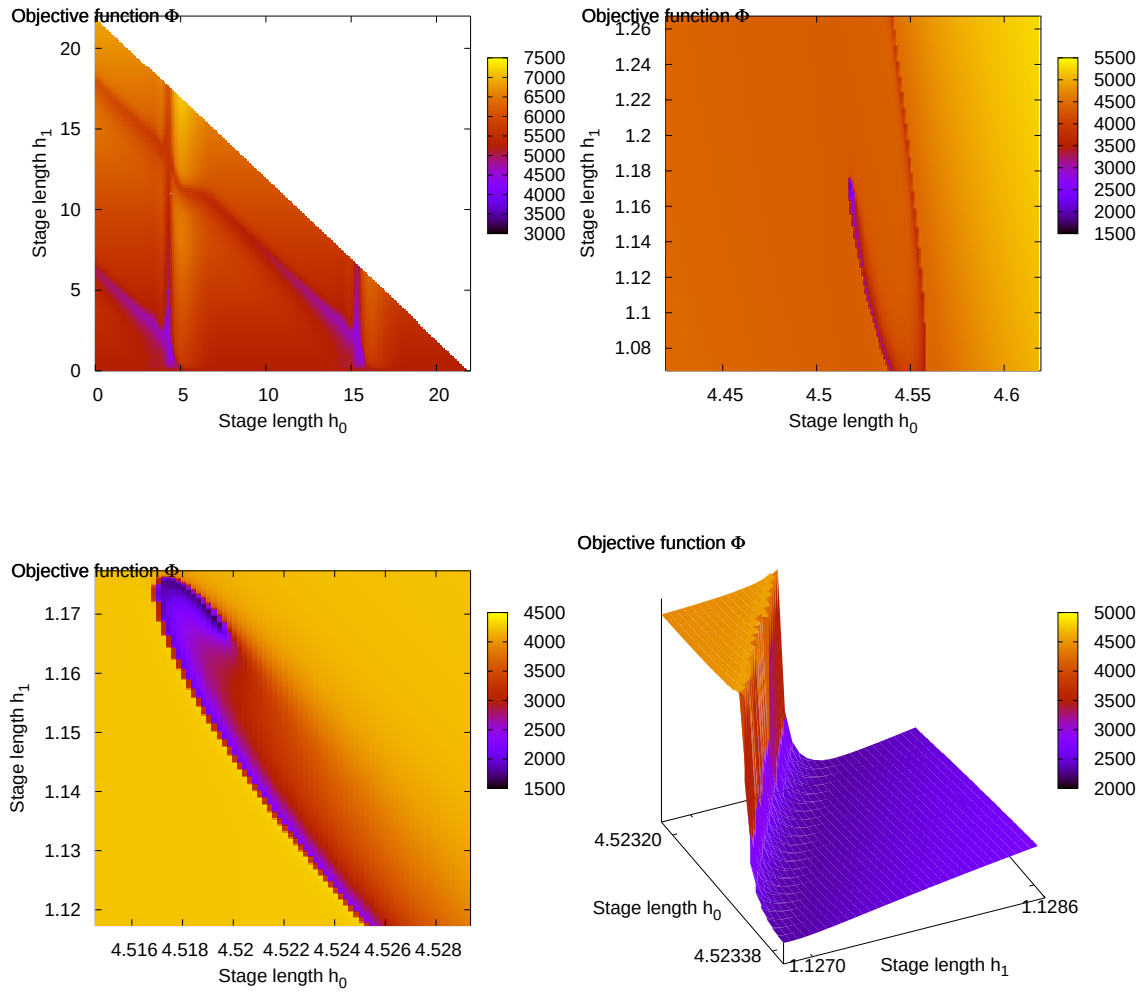

Figure 7.16: Objective function landscape of the calcium example, obtained by brute–force simulation of the ODE by variation of h_0, h_1 and $h_2 = 22 - h_0 - h_1$. The x-axis gives the length h_0 before the stimulus begins, the y-axis the length h_1 of this stimulus. In the top left plot the whole time domain $[0, 22]$ is shown. Note the periodic nature of the landscape with a period of ≈ 11 time units. Of course the second valley at $h_0 \approx 15$ is not so deep as the one at $h_0 \approx 4.5$, as the resetting takes place a whole period later. The top right and bottom left plots show a zoom into the vicinity of the optimal values $h_0^* = 4.51958$, $h_1^* = 1.16726$. The bottom right plot illustrates the topography of the border of the narrow channel depicted in the bottom left plot. The main problem to optimize this problem is the fact that the border line consists of local maxima on the lines orthogonal to the border. Therefore all descent–based methods will iterate away from the channel, instead of following it to its deepest point.

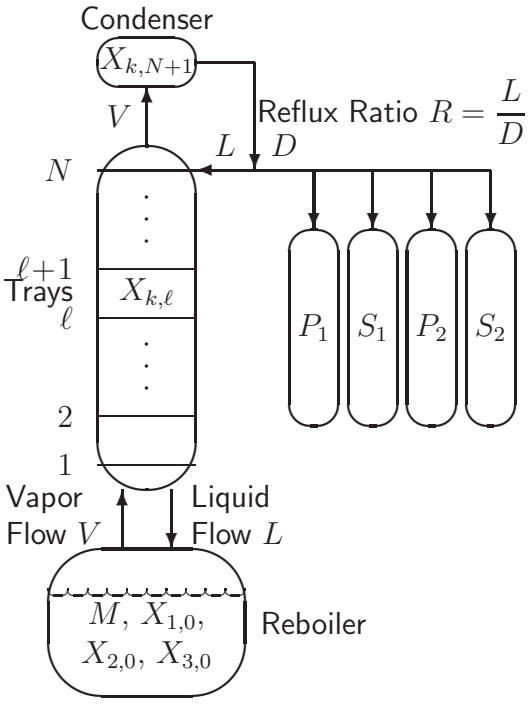

Figure 7.17: Ternary batch distillation with two production cuts P_1, P_2 and two waste cuts S_1, S_2 in a column with $N = n_{\text{tray}}$ ideal trays.

- No pressure drop
- Total condenser
- Constant molar overflow for vapor V and liquid L
- Tray phase equilibria as for ideal mixture, with partial pressures determined by the Antoine-equation

We assume that the heating power is kept constant, so that the vapor flow V is not a control, but a fixed parameter. Using the reflux ratio R, L and D can directly be eliminated,

$$L = V\frac{R}{R+1} \quad \text{and} \quad D = V\frac{1}{R+1}. \tag{7.19}$$

The concentrations in the reboiler are denoted by $X_{k,0}$ and those in the condenser by $X_{k,N+1}$, analogously to the tray concentrations ($X_{k,\ell}$).
The only non-neglected mass in the system is the molar reboiler content M. During the distillation process, M is reduced by the distillate flow D,

$$\frac{dM}{dt} \;=\; -D \;=\; \frac{-V}{R+1}. \tag{7.20a}$$

The mass conservation of the different components k requires analogously

$$\frac{d(MX_{k,0})}{dt} \;=\; -DX_{k,N+1}$$

which is equivalent to

$$\frac{d(X_{k,0})}{dt} \;=\; \frac{V}{M(R+1)}(X_{k,0} - X_{k,N+1}) \tag{7.20b}$$

for $k = 1, 2$. The conservation of the third component is implicitly given by (7.20a), (7.20b). Therefore, the three dynamic equations for the differential variables M, $X_{1,0}$, and $X_{2,0}$ are specified.
As algebraic equations we first require the componentwise mass conservation in the column section above the $(\ell + 1)$st tray,

$$V K_k(T_\ell)\, X_{k,\ell} - L X_{k,\ell+1} - D X_{k,N+1} \;=\; 0$$

or equivalently

$$K_k(T_\ell) X_{k,\ell} - \frac{R}{R+1} X_{k,\ell+1} - \frac{1}{R+1} X_{k,N+1} = 0 \tag{7.20c}$$

for $k = 1, 2 \quad \ell = 0, 1, \ldots, N$. The first term corresponds to the vapor flow entering into the $(\ell + 1)$st tray, the second to the outflowing liquid, and the third to the distillate flow D. The vapor concentrations on the ℓth tray are calculated as the product of the equilibrium values $K_k(T_\ell)$ with the liquid concentration $X_{k,\ell}$.
The concentration of the third component is determined by the closing condition

$$1 - \sum_{k=1}^{3} X_{k,\ell} = 0, \quad \ell = 0, 1, \ldots, N+1. \tag{7.20d}$$

The tray temperature T_ℓ is implicitly defined by the closing condition for the vapor concentrations,

$$1 - \sum_{k=1}^{3} K_k(T_\ell)\, X_{k,\ell} = 0, \;\; \ell = 0, 1, \ldots, N. \tag{7.20e}$$

Assuming an ideal mixture, the equilibrium values $K_k(T_\ell)$ are determined according to Raoult's law,

$$K_k(T_\ell) = \frac{\rho_k^s(T_\ell)}{\rho}, \quad k = 1, 2, 3. \tag{7.20f}$$

Here, ρ is the total pressure assumed to be constant over the whole column, and $\rho_k^s(T_\ell)$ are the partial pressures of the undiluted components – they are determined by the Antoine equation for $k = 1, 2, 3$,

$$\rho_k^s(T_\ell) = \exp_{10}\left(A_k - \frac{B_k}{T_\ell + C_k}\right). \tag{7.20g}$$

The Antoine coefficients used in the presented example are given in appendix E. The total number of algebraic variables ($X_{3,0}$, $X_{k,\ell}$ for $k = 1, 2, 3$ and $\ell = 1, \ldots, N+1$ as well as T_ℓ for $\ell = 0, \ldots, N$) in the given formulation is $4N + 5$. By resolving (7.20d),

the concentrations $X_{3,\ell}$ for $\ell = 1, 2 \ldots, N$ can be eliminated directly, so that only $3N + 5$ algebraic states remain. In the computations presented in this paper we have used a model with $N = 5$ trays.
The described model, with holdups neglected on trays and in the reflux drum, is similar to models considered by Diwekar *et al.* (1987) or Logsdon *et al.* (1990). It has to be noted, however, that the inclusion of holdups in the model may lead to quite different optimal profiles, as demonstrated e.g. by Logsdon & Biegler (1993) and Mujtaba & Macchietto (1998). Other approaches towards batch distillation modeling and optimization have been developed by Diwekar (1995), and by Mujtaba & Macchietto (1992, 1996).

Diehl *et al.* (2002) show that it is profitable to recycle the waste cuts instead of removing them from the distillation process under certain assumptions. For the separation of an amount of feedstock considerably larger than a single batch, carried out by a sequence of identical batches, this could be achieved by adding the waste material of one batch to the feed of the following one. Although this reduces the amount of *fresh* feedstock which can be processed in a single batch (due to limited reboiler holdup) and thus results in a longer overall distillation time T for a given total amount of feedstock, the product outputs P_1, P_2 and P_3 are increased and disposal of the slop cuts S_1 and S_2 becomes unnecessary.
If the product prices are specified by constants c^i_{price}, and the costs for the energy consumption per time unit by c_{energy} and for slop cut disposal by s_j, the profit P for the whole process is given by

$$P = \sum_{i=1}^{3} c^i_{\text{price}} P_i \; - \; \sum_{j=1}^{2} s_j S'_j \; - \; c_{\text{energy}} T, \tag{7.21}$$

where S'_j are the amounts of slop cut material that are *not* recycled. The costs for the feed purchase are constant for a fixed amount of feedstock and are left out of consideration in the problem formulation.
To treat the recycling problem, we consider the limiting case of a total amount of feed that is considerably larger than the reboiler holdup; this would result in a large number of consecutive, nearly identical batches: here, one batch produces the amount of slop cut material that is available for recycling in the following batch. If we assume that all batches are identical, we need to treat only *one single* batch which produces exactly the same slop cut material as its output which was previously given to it as an input for recycling. This quasi–periodic process formulation leads to a coupled multipoint boundary value problem.
The first slop cut is added at the beginning of the first production cut, as it contains mainly components 1 and 2, while the second slop cut is recycled in the second production cut, because it mainly consists of components 2 and 3.
To allow for *partial* recycling of the slop cuts the recycling ratios R_1, R_2 are introduced as *free* optimization parameters. Thus, the non–recycled parts of the slop cuts become

$$S'_j = (1 - R_j) S_j, \quad j = 1, 2. \tag{7.22}$$

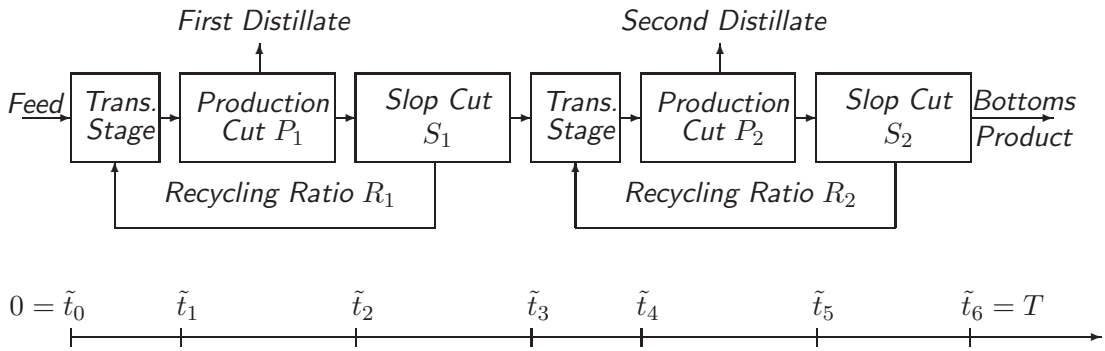

Figure 7.18: Quasi–periodic ternary batch distillation with waste cut recycling.

For the determination of the total distillation time we proceed as follows: instead of determining the increased number of batches due to the slop cut recycling, we artificially increase the initial reboiler filling of a single batch by the amount of the recycled material from slop cut S_1 – this leaves the total amount of batches fixed but increases the time for one batch by exactly the same factor, by which the amount of batches should have been increased. This is due to the fact that the differential equations (7.20a), (7.20b) are invariant with respect to simultaneous rescaling of time t and reboiler content M. Note also that the vapor flow V enters all differential equations linearly and only leads to a rescaling of the time axis. Therefore, by varying V as a control over time (instead of keeping it constant), no additional gains would be produced in the considered cost model.
If we neglect the cost of the preparation time needed between two batches (during which no energy costs occur), we can restrict the objective formulation to material– and energy–costs of one single batch only. We note that experimentation with alternate problem formulations, e.g., the fixing of production and optimization of raw material, are easy to accomplish in the framework of our approach. For the sake of simplicity, however, we will restrict ourselves to one single formulation in the following.

Formulation as a multipoint boundary value problem

The process under consideration is formulated as a periodic multistage optimal control problem. The idea is illustrated in figure 7.18.
By conceptionally introducing intermediate *transition stages* to describe the addition of the slop cuts, we can formulate interior point constraints of type (1.18f) on the value of the reboiler holdup M as a linear coupling between the value of M at the interior points $0 = \tilde{t}_0, \tilde{t}_1, \ldots, \tilde{t}_6 = T$ in the following way:

$$\begin{aligned} M|_{\tilde{t}_1} - M|_{\tilde{t}_0} &= R_1 S_1 \\ &= R_1 (M|_{\tilde{t}_2} - M|_{\tilde{t}_3}), \end{aligned} \tag{7.23a}$$

$$\begin{aligned} M|_{\tilde{t}_4} - M|_{\tilde{t}_3} &= R_2 S_2 \\ &= R_2 (M|_{\tilde{t}_5} - M|_{\tilde{t}_6}). \end{aligned} \tag{7.23b}$$

Remark 7.2 *Note that the transition stages are artificial and have zero length in our formulation. This leads to, e.g.,* $\tilde{t}_0 = \tilde{t}_1$*, but* $M|_{\tilde{t}_0} \neq M|_{\tilde{t}_1}$ *resp.* $x_0(\tilde{t}_0) \neq x_0(\tilde{t}_1)$*.*

Analogously, the transition equations for the two other differential variables, the reboiler concentrations $X_{k,0}$ $(k = 1, 2)$, are formulated by using the conservation of the component quantities $X_{k,0}M$,

$$X_{k,0}M|_{\tilde{t}_1} - X_{k,0}M|_{\tilde{t}_0} = R_2(X_{k,0}M|_{\tilde{t}_2} - X_{k,0}M|_{\tilde{t}_3}), \tag{7.24a}$$
$$X_{k,0}M|_{\tilde{t}_4} - X_{k,0}M|_{\tilde{t}_3} = R_2(X_{k,0}M|_{\tilde{t}_5} - X_{k,0}M|_{\tilde{t}_6}), \tag{7.24b}$$

for $k = 1, 2$. Altogether, we obtain six coupled interior point constraints that determine the jumps of the three differential states in the two transition stages.
The purity requirements for the three product fractions are imposed by additional interior point constraints of the form (1.18e),

$$\frac{X_{1,0}M|_{\tilde{t}_1} - X_{1,0}M|_{\tilde{t}_2}}{M|_{\tilde{t}_1} - M|_{\tilde{t}_2}} \geq X_{P_1}, \tag{7.25a}$$
$$\frac{X_{2,0}M|_{\tilde{t}_4} - X_{2,0}M|_{\tilde{t}_5}}{M|_{\tilde{t}_4} - M|_{\tilde{t}_5}} \geq X_{P_2}, \tag{7.25b}$$
$$X_{3,0}|_{\tilde{t}_6} \geq X_{P_3}. \tag{7.25c}$$

Here, $X_{P_1} = 98\%$, $X_{P_2} = 96\%$, and $X_{P_3} = 99\%$ are the required minimum purities of the main component in the product fractions.

	TS	P1	S1	TS	P2	S2	End
nr	$\tilde{t}_0$	$\tilde{t}_1$	$\tilde{t}_2$	$\tilde{t}_3$	$\tilde{t}_4$	$\tilde{t}_5$	$\tilde{t}_6$
0		x_0	$-x_0 - pr_0$				
1		x_0x_1	$-x_0x_1 - pr_1$				
2					x_0	$-x_0 - pr_0$	
3					x_0x_1	$-x_0x_1 - pr_1$	
4	x_0	$-x_0$	$x_0\ p_2$	$-x_0\ p_2$			
5	x_0x_1	$-x_0x_1$	$x_0x_1\ p_2$	$-x_0x_1\ p_2$			
6	x_0x_2	$-x_0x_2$	$x_0x_2\ p_2$	$-x_0x_2\ p_2$			
7				x_0	$-x_0$	$x_0\ p_3$	$-x_0\ p_3$
8				x_0x_1	$-x_0x_1$	$x_0x_1\ p_3$	$-x_0x_1\ p_3$
9				x_0x_2	$-x_0x_2$	$x_0x_2\ p_3$	$-x_0x_2\ p_3$
dc			$\frac{pr_1}{pr_0} \geq X_{P_1}$			$\frac{pr_1}{pr_0} \geq X_{P_2}$	$1 - x_1 - x_2 \geq X_{P_3}$

Table 7.2: Interior point constraints overview for time–independent slop cuts to reboiler.

Table 7.2 shows all interior point constraints in an overview. The differential states are $x_0(\cdot)$ as the still pot holdup M, $x_1(\cdot)$ and $x_2(\cdot)$ as the mole fractions $X_{1,0}$ respectively $X_{2,0}$. The time independent parameters are p_0 as the vapor flow rate V, p_1 as the system pressure ρ, and p_2 and p_3 as the recycling ratios R_1 and R_2 of the first respectively second slop cut. The table is to be read in the following way. The columns correspond to time points $\tilde{t}_i$ when a new model stage begins, i.e., for the first six columns to the begin of either transition stage (TS), production stage (P1 or P2) or slop stage (S1 or S2). The last time point is the end of the last slop stage at $\tilde{t}_6 = T$. The entries x_i in column j are to be understood as $x_i|_{\tilde{t}_j}$ (remember remark 7.2). The rows of the table correspond to nine coupled and three decoupled interior point constraints, a differentiation that makes computations more efficient, see Leineweber (1999). The coupled constraints 0–9 are equality constraints. For each row the sum of all columns has to be zero. While constraints 0 to 3 determine the local parameters pr_0, pr_1 that are needed to calculate the purity of the components in the production stages, constraints 4 to 9 are the mass conservation laws given above. Constraints 4 and 7 are reformulations of (7.23), constraints 5, 6 and 8,9 are the mass conservation laws (7.24) for the two slop cuts and two components. The three decoupled constraints in row *dc* are the inequality constraints (7.25) that guarantee the prescribed purity of the product.

Time–dependent slopcut inflow to any tray

We extend the optimal control problem by introducing additional degrees of freedom. We allow a recycling of the slop cuts S1 and S2 of the previous batch process not only at the stage transition times into the reboiler, but at any given time t to any of the N trays. To this end we introduce additional differential variables, namely x_3 which gives the content of slop cut reservoir S1 and x_4 for the one of reservoir S2. Additional time–independent parameters are p_4 and p_5 for the mole fractions of components 1 resp. 2 in the first reservoir and p_6 and p_7 for those in the second. The control functions $u_i(\cdot), i = 1 \ldots N+1$ describe the inflow of reservoir S1 to tray $i-1$, where tray 0 is the reboiler. The functions $u_i(\cdot), i = N+2 \ldots 2N+2$ describe the inflow from reservoir S2 to tray $i-(N+2)$. We will first consider the case where these functions can take any continuous value, i.e., there exists a pump for every single connection. This scenario is illustrated in comparison to the preceding ones in figure 7.19.

The differential equations have to be modified. Using $L = V\frac{R}{R+1}$ and $D = \frac{V}{R+1}$ we derive again from mass conservation

$$\dot{x}_0 = \frac{\mathrm{d}M}{\mathrm{d}t} \quad = \quad -D + \sum_{i=1}^{2N+2} u_i = \frac{-V}{R+1} + \sum_{i=1}^{2N+2} u_i \tag{7.26a}$$

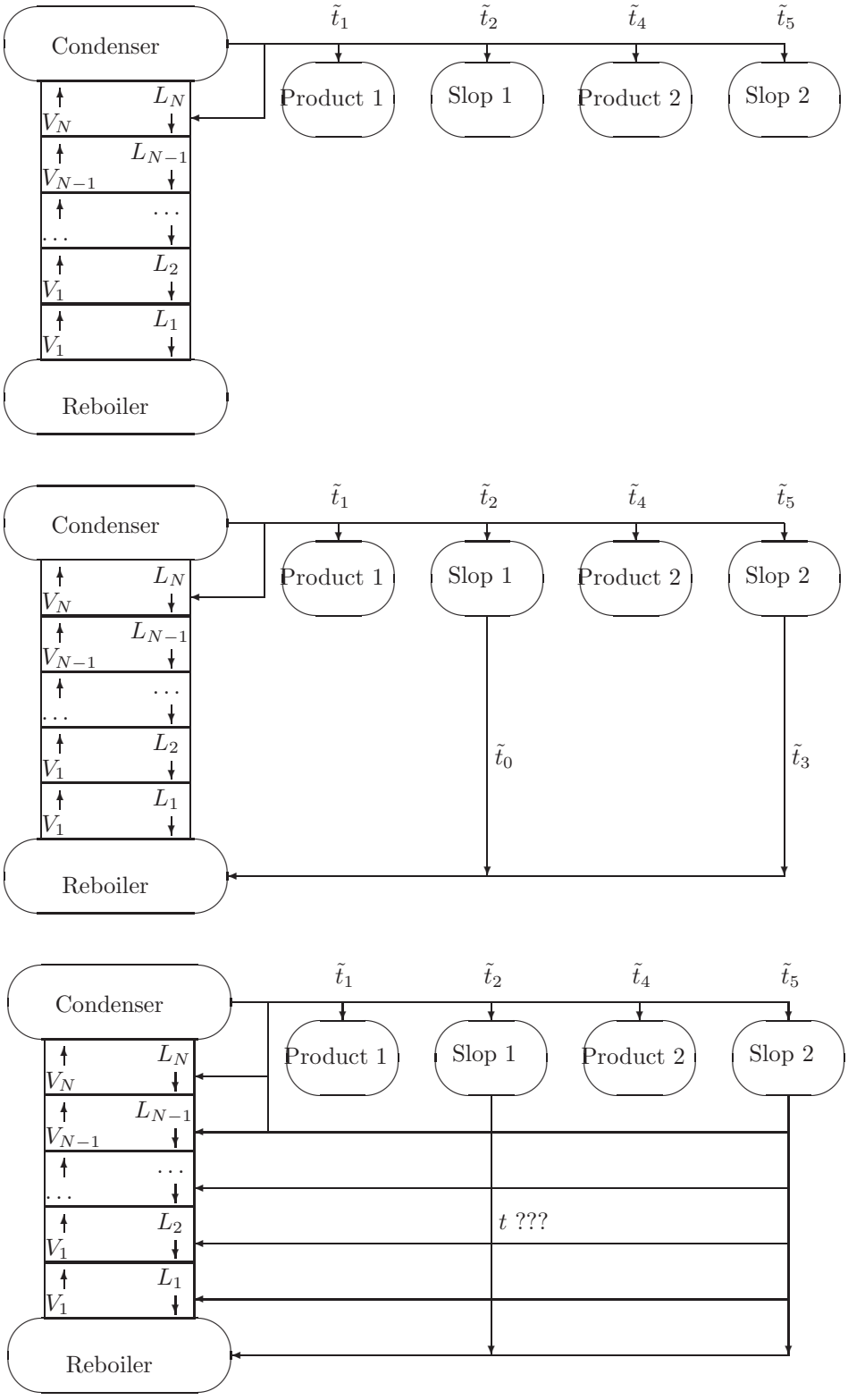

Figure 7.19: Illustration of the three different scenarios – without recycling (A, top), with recycling in transition stages only (B, middle) and with time– and tray–dependent reusage (C, bottom).

for the liquid holdup in the column, compare (7.20a). From

$$\begin{aligned}\frac{\mathrm{d}(MX_{k,0})}{\mathrm{d}t} &= X_{k,0}\frac{\mathrm{d}M}{\mathrm{d}t} + M\frac{\mathrm{d}X_{k,0}}{\mathrm{d}t} \\ &= -DX_{k,N+1} + \sum_{i=1}^{N+1} u_i\, p_{3+k} + \sum_{i=N+2}^{2N+2} u_i\, p_{5+k}\end{aligned}$$

it follows that

$$\begin{aligned}\dot{x}_k = \frac{\mathrm{d}X_{k,0}}{\mathrm{d}t} &= \frac{V}{M(R+1)}\,(X_{k,0} - X_{k,N+1}) - \frac{X_{k,0}}{M}\sum_{i=1}^{2N+2} u_i \\ &\quad + \frac{1}{M}\sum_{i=1}^{N+1} u_i\, p_{3+k} + \frac{1}{M}\sum_{i=N+2}^{2N+2} u_i\, p_{5+k} \\ &= \frac{V\,(X_{k,0} - X_{k,N+1})}{M(R+1)} + \frac{1}{M}\sum_{i=1}^{N+1} u_i(p_{3+k} - X_{k,0}) \\ &\quad + \frac{1}{M}\sum_{i=N+2}^{2N+2} u_i(p_{5+k} - X_{k,0}) \qquad (7.26b)\end{aligned}$$

for $k = 1, 2$, with additional terms when compared to (7.20b). The differential equations for the slop reservoirs are given by

$$\dot{x}_3 = -\sum_{i=1}^{N+1} u_i, \qquad (7.26c)$$

$$\dot{x}_4 = -\sum_{i=N+2}^{2N+2} u_i. \qquad (7.26d)$$

No additional algebraic variables are needed for the extended model. The existing equations (7.20c) to (7.20g) have to be modified, though. We use the same approach as above and require the componentwise mass conservation in the column section above the $(\ell+1)$st tray,

$$\begin{aligned}0 &= VK_k(T_\ell)X_{k,\ell} - L_\ell X_{k,\ell+1} - DX_{k,N+1} \\ &\quad + \sum_{i=\ell+1}^{N} (u_{i+1}\, p_{3+k} + u_{i+N+2}\, p_{5+k}) \qquad (7.27)\end{aligned}$$

with L_ℓ being the liquid flow entering tray ℓ, given by

$$L_\ell = L + \sum_{i=\ell+1}^{N} (u_{i+1} + u_{i+N+2}).$$

Dividing (7.27) by V yields

$$\begin{aligned} 0 \;=\; & K_k(T_\ell)X_{k,\ell} - \frac{R}{R+1}X_{k,\ell+1} - \frac{X_{k,N+1}}{R+1} \\ & + \frac{(p_{3+k} - X_{k,\ell+1})}{V}\sum_{i=\ell+1}^{N} u_{i+1} + \frac{(p_{5+k} - X_{k,\ell+1})}{V}\sum_{i=\ell+1}^{N} u_{i+N+2}, \end{aligned} \tag{7.28}$$

for $k = 1, 2$ and $\ell = 0, 1, \ldots, N$, compare (7.20c). The other algebraic equations, (7.20d) to (7.20g), remain unaltered, as the objective function (7.21).

The interior point constraints are modified and extended. Table 7.3 gives an overview that can be compared directly with table 7.2.

	TS	P1	S1	TS	P2	S2	End
nr	$\tilde{t}_0$	$\tilde{t}_1$	$\tilde{t}_2$	$\tilde{t}_3$	$\tilde{t}_4$	$\tilde{t}_5$	$\tilde{t}_6$
0		x_0 $+x_3 + x_4$	$-x_0 - pr_0$ $-x_3 - x_4$				
1		x_0x_1+ $p_4x_3 + p_6x_4$	$-x_0x_1 - pr_1$ $-p_4x_3 - p_6x_4$				
2					x_0 $+x_3 + x_4$	$-x_0 - pr_0$ $-x_3 - x_4$	
3					x_0x_2 $p_5x_3 + p_7x_4$	$-x_0x_2 - pr_1$ $-p_5x_3 - p_7x_4$	
4	x_0	$-x_0$	$x_0\ p_2$	$-x_0\ p_2$			
5	x_0x_1	$-x_0x_1$	$x_0x_1\ p_2$	$-x_0x_1\ p_2$			
6	x_0x_2	$-x_0x_2$	$x_0x_2\ p_2$	$-x_0x_2\ p_2$			
7				x_0	$-x_0$	$x_0\ p_3$	$-x_0\ p_3$
8				x_0x_1	$-x_0x_1$	$x_0x_1\ p_3$	$-x_0x_1\ p_3$
9				x_0x_2	$-x_0x_2$	$x_0x_2\ p_3$	$-x_0x_2\ p_3$
10		x_3	$-(1-p_2)x_0$	$(1-p_2)x_0$			
11		x_4				$-(1-p_3)x_0$	$(1-p_3)x_0$
12			$x_0x_1 - p_4x_0$	$p_4x_0 - x_0x_1$			
13			$x_0x_2 - p_5x_0$	$p_5x_0 - x_0x_2$			
14						$x_0x_1 - p_6x_0$	$p_6x_0 - x_0x_1$
15						$x_0x_2 - p_7x_0$	$p_7x_0 - x_0x_2$
dc			$\frac{pr_1}{pr_0}$ $\geq X_{P_1}$			$\frac{pr_1}{pr_0}$ $\geq X_{P_2}$	$1 - x_1 - x_2$ $\geq X_{P_3}$

Table 7.3: Interior point constraints overview for time– and tray–dependent slop cuts to reboiler. The bold constraint numbers indicate modifications of the constraints given in table 7.2.

The first four constraints 0 to 3 have been altered, as the mass that is distilled in the two production cuts is not the difference between the mass in the reboiler at the begin and at the end of a production cut any more, but may include mass from one or both of the slop reservoirs that has been fed during the production cut. Therefore the differences between the slop reservoir contents x_3 resp. x_4, also componentwise,

are contained in the new interior point constraints

$$\begin{aligned} pr_0 &= x_0(\tilde{t}_1) - x_0(\tilde{t}_2) + x_3(\tilde{t}_1) - x_3(\tilde{t}_2) + x_4(\tilde{t}_1) - x_4(\tilde{t}_2) \\ pr_1 &= x_0x_1(\tilde{t}_1) - x_0x_1(\tilde{t}_2) + p_4\,(x_3(\tilde{t}_1) - x_3(\tilde{t}_2)) + p_6\,(x_4(\tilde{t}_1) - x_4(\tilde{t}_2)) \end{aligned}$$

(and equivalently for $\tilde{t}_4, \tilde{t}_5$) that determine the local parameters pr_0, pr_1 that yield the purities pr_1/pr_0 for the decoupled interior point inequalities.
Additional interior point conditions are needed to determine the initial values of the two slop reservoirs that contain the fraction of the two waste cuts that is refed dynamically. These equations read as

$$\begin{aligned} x_3(\tilde{t}_1) &= (1-p_2)\,(x_0(\tilde{t}_2) - x_0(\tilde{t}_3)) \\ x_4(\tilde{t}_1) &= (1-p_3)\,(x_0(\tilde{t}_5) - x_0(\tilde{t}_6)) \end{aligned}$$

and can be found in rows 10 and 11 in table 7.3. Recall that $0 \leq p_2, p_3 \leq 1$ are the fractions of the waste cuts that are fed in the transition stages directly to the reboiler, as in the previous scenario. The right hand side describes thus what is left of the produced waste cuts in the reservoirs after this fill–in.
We need to determine the parameters p_4, p_5, p_6 and p_7, i.e., the fractions of components 1 and 2 in each of the two slop reservoir tanks. These parameters are given by

$$\begin{aligned} p_4 &= \frac{x_0x_1(\tilde{t}_2) - x_0x_1(\tilde{t}_3)}{x_0(\tilde{t}_2) - x_0(\tilde{t}_3)}, \quad p_5 = \frac{x_0x_2(\tilde{t}_2) - x_0x_2(\tilde{t}_3)}{x_0(\tilde{t}_2) - x_0(\tilde{t}_3)}, \\ p_6 &= \frac{x_0x_1(\tilde{t}_5) - x_0x_1(\tilde{t}_6)}{x_0(\tilde{t}_5) - x_0(\tilde{t}_6)}, \quad p_7 = \frac{x_0x_5(\tilde{t}_5) - x_0x_5(\tilde{t}_6)}{x_0(\tilde{t}_5) - x_0(\tilde{t}_6)} \end{aligned}$$

and can be found in rows 12 to 15.

The extensions we made require an additional pump for each possible connection between the reservoirs and the trays, i.e., a number of $2N+2$ pumps. As pumps are comparatively expensive, we consider the case where we only have one pump at hand for each slop reservoir plus a valve that determines to which tray the recycled waste cut is fed. The control functions $u_i(\cdot)$ are thus replaced by the product between the continuous inflow controls and binary control functions, i.e.,

$$u_i = \hat{u}_1\, w_i, \quad i = 1 \ldots N+1 \tag{7.29a}$$

$$u_i = \hat{u}_2\, w_i, \quad i = N+2 \ldots 2N+2 \tag{7.29b}$$

and constraints

$$w_i \in \{0,1\}, \quad i = 1 \ldots 2N+2 \tag{7.29c}$$

$$\sum_{i=1}^{N+1} w_i = \sum_{i=N+2}^{2N+2} w_i = 1. \tag{7.29d}$$

The resulting optimal control problem is of type (1.18) and includes transition stages, coupled and decoupled interior point inequalities and equalities, differential and algebraic variables, free, time–independent parameters, free stage lengths and continuous as well as binary control functions.

Optimization Results

We make the same assumptions for costs and prices as in Diehl *et al.* (2002). Corresponding to the authors, these are made as careful as possible not to encourage the recycling of slop cuts. In particular no costs at all are assigned to slop cut disposal and the prices for the products are chosen so low that in the non–recycling case the gains are only marginally higher than the energy consumption costs. The parameters s_1, s_2, c^1_{price}, c^2_{price}, c^3_{price} and c_{energy} are given together with the other parameters in the appendix.

Before we apply *MS MINTOC* to obtain a solution of the mixed–integer optimal control problem, we will review and reproduce the two scenarios presented in Diehl *et al.* (2002). These are the optimal control without any recycling of the slop cuts, i.e., with recycling ratios R_1, R_2 fixed to be zero (from now on scenario A), and a second one following the description given above, with full freedom to choose the recycling ratios between 0 and 1, but no recycling at any time during a stage or to a tray other than the reboiler (scenario B).

The optimization for scenario A yields process duration (energy) costs of 3.51 units, and a product sale income of 3.96 units, i.e., $P = 0.45$ units of profit. The slop cut outputs are small, with relatively short durations and high reflux ratios, as these outputs are lost for the process.

When a time–independent slop cut recycling is allowed, the optimization of scenario B results in a full recycling, i.e., $R_1 = R_2 = 1$ with an increased total time resp. energy cost of 3.74 units and a product sale income of 4.50 units. The net profits of 0.76 units are increased by 69% compared to the non–recycling value of 0.45.

Parts of the optimal trajectories are depicted in the top rows of figure 7.20. The reflux ratio is generally smaller than before, as the purity constraints are satisfied by taking bigger slop cuts, that are no longer lost for the process.

In Diehl *et al.* (2002) further optimization studies are given, e.g., an investigation for which product prices recycling becomes unprofitable. They claim that, when one product price, c^1_{price}, was reduced to lower values, starting from the nominal value $c^1_{\text{price}} = 4.50$ and stepwise decreased by 0.25, recycling of the first slop cut becomes unprofitable for $c^1_{\text{price}} < 2.25$, and no recycling of the first slop cut is recommended by the optimizer. But we are more interested in the question, whether we can improve the performance of the batch process by a recycling of the waste cuts during the cuts, not only in the transition stages.

We allow a reflow into any of the five trays or the reboiler at any time, also during the slop cuts. Scenario B is included in scenario C, as we will call it, as a recycling in the transition stages is still possible. We will even use scenario B, i.e., $R_1 = R_2 = 1$ and $\hat{u}_1 = \hat{u}_2 = 0$, as initialization of the optimization.

The algorithm proceeds in the usual way. First a relaxed problem is solved, then the control discretization grid is refined and finally we perform a switching time optimization. We obtain a solution to the relaxed problem without any recycling in the transition stages any more, $R^*_1 = R^*_2 = 0$. Parts of the optimal trajectory are depicted in figure 7.21. The corresponding objective value is $\Phi^0 = 0.86016$

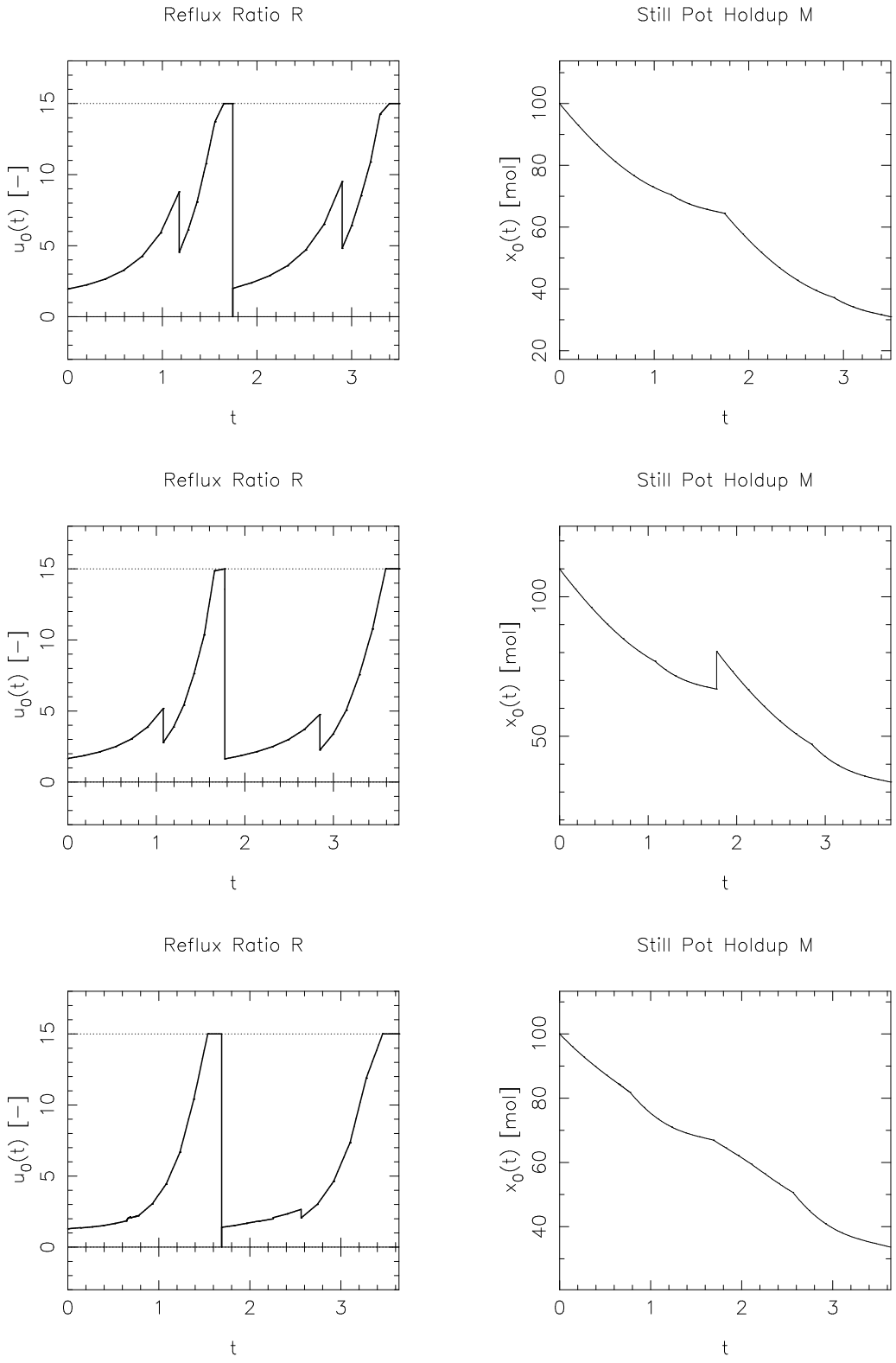

Figure 7.20: Reflux ratio and still pot holdup for the batch process without waste cuts, scenario A (top), with waste cuts, scenario B (middle) and with time– and tray–dependent waste cuts, scenario C (bottom). The parameterization of the control for the reflux ratio $R(\cdot)$ is chosen to be continuous piecewise linear.

units. Refining the grid we obtain $\Phi^1 = 0.86024$ and $\Phi^2 = 0.86105$ with a solution that is almost integer. We round this solution with strategy SR-SOS1 and perform a switching time optimization. The optimal solution yields an objective value of $\Phi = 0.86312$, with a final time of $T = 3.63688$, and is thus 13.6% more profitable than the optimal solution of scenario B. The same amount of feed, i.e., all of it, is recycled. The process is more profitable, because it is faster and needs less energy.
Parts of the optimal trajectory are shown in figures 7.20 (reflux and still pot holdup, bottom line), 7.22 (some binary control functions and the reflux controls $\hat{u}_1$ and $\hat{u}_2$), 7.23 (slop cut reservoir contents) and in figures E.2, E.4 and E.6 (Temperature and mole fractions on all trays) in the appendix.
As can be seen in figure 7.22, there is neither any reflux during the slop cuts nor from reservoir S2 during the first resp. of reservoir S1 during the second production cut. This is clear, as the concentrations of component 2 in the first reservoir, p_5^*, and of the first component in the second reservoir, p_6^*, are low,

$$\begin{aligned} p_4^* &= 0.83955, \\ p_5^* &= 0.16025, \\ p_6^* &= 0.000123, \\ p_7^* &= 0.76551. \end{aligned}$$

The two valves switch once in the considered time interval. The times and chronology of the events are

Time	Event
0.0	Start P1, flux from reservoir S1 to tray 3
0.69091	Start flux from reservoir S1 to tray 4
0.77524	Start S1
1.68985	Start P2, flux from reservoir S2 to tray 2
2.10154	Start flux from reservoir S2 to tray 3
2.56252	Start S2
3.63688	End of the process

It is interesting to investigate, how much is gained by the introduction of the valves. If we fix

$$w_4 = 1, \quad w_9 = 1, \quad w_i = 0 \text{ for } i \in \{1, 2, 3, 5, 6, 7, 8, 10, 11, 12\},$$

i.e., we direct the slop reservoir S1 flux to tray 3 and the slop reservoir S2 flux to tray 2, we do not need valves any more. An optimization with these variables fixed yields an overall process time of $T = 3.64208$ with a corresponding profit of $P = 0.85792$. It can be expected that for columns with more trays, the solution will switch more often and the effect will be more considerable, as for different parameters considering the objective function. The strength of our approach is to predict exactly the gains that can be achieved by introducing valves and to calculate if the investment and maintenance costs of the valves exceed the gains of the process.

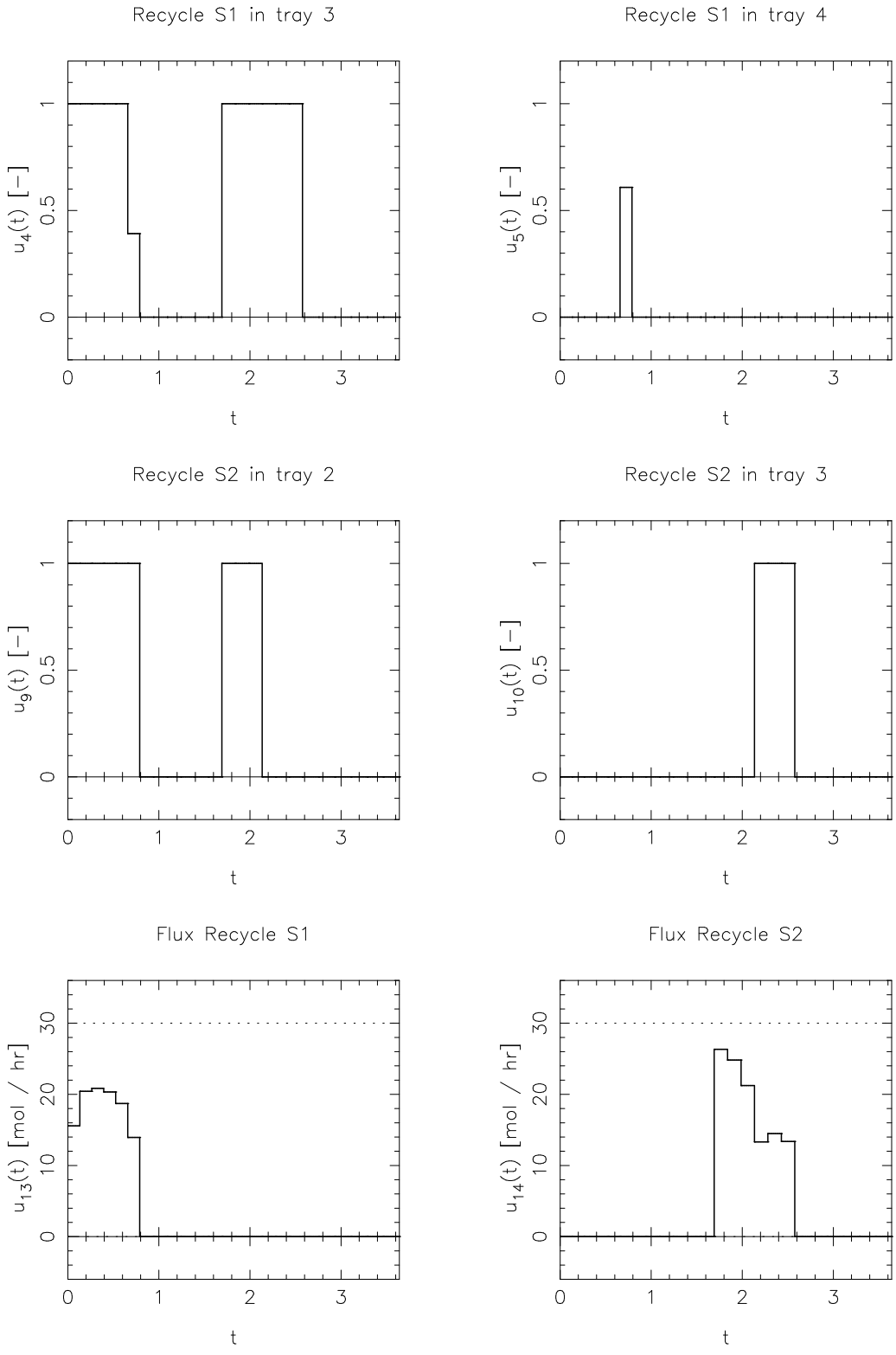

Figure 7.21: Parts of the solution of the relaxed problem C. Only four of the relaxed binary control functions are shown, all other 8 are identical zero over the whole time horizon. Note that the binary control functions have no impact, whenever the corresponding fluxes $\hat{u}_1$ and $\hat{u}_2$ are zero.

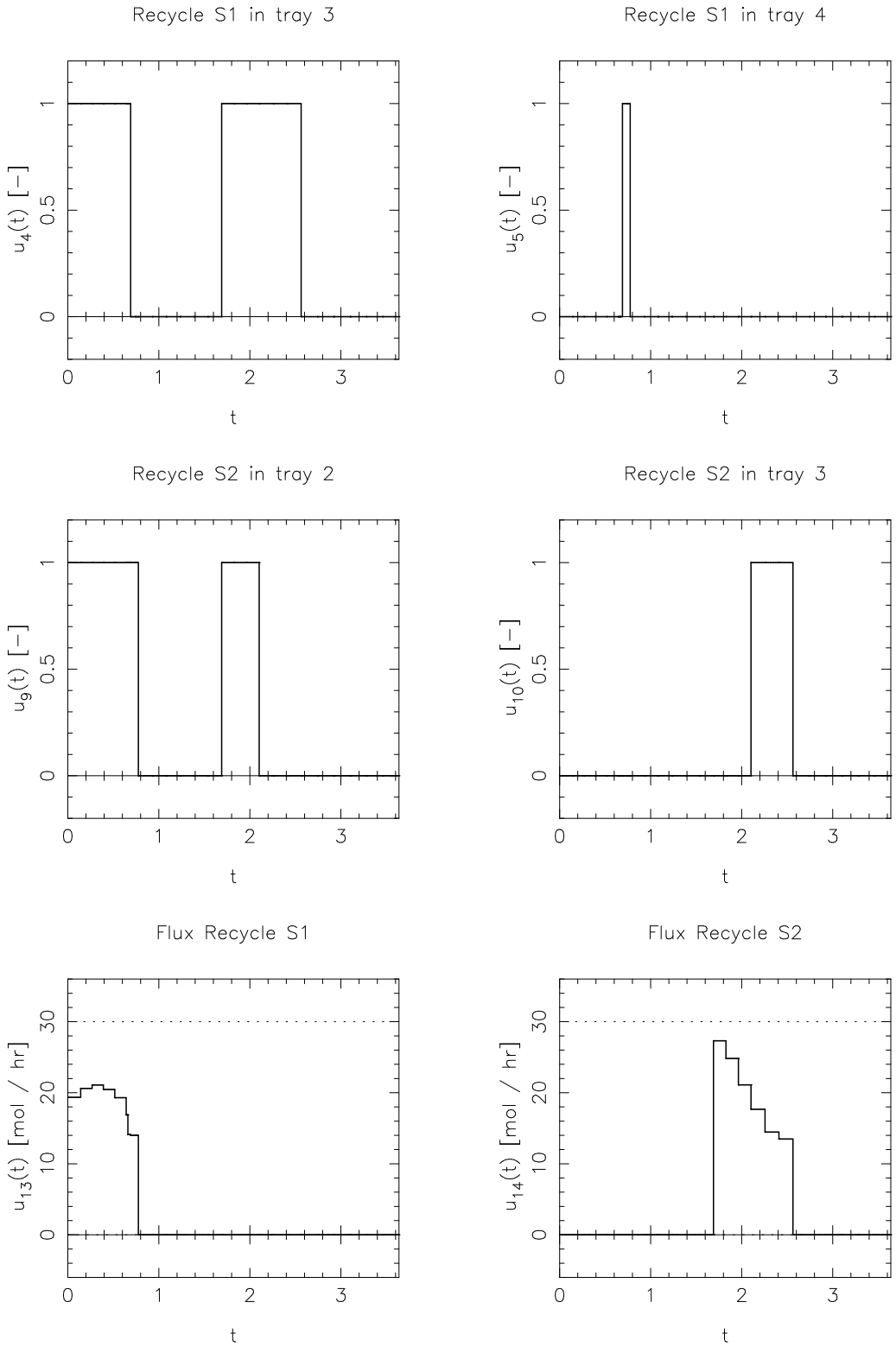

Figure 7.22: Parts of the binary admissible solution of scenario C. The flux from reservoir S1 goes to tray 3 and 4, with one switch in between(top). The flux from reservoir S2 is directed to trays 2 and 3, also with one switching (middle). The bottom line shows the corresponding fluxes.

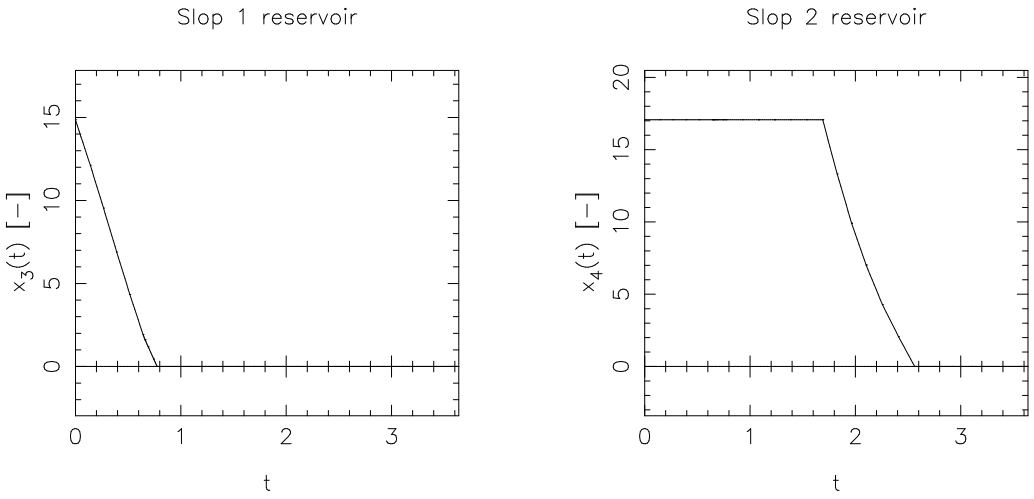

Figure 7.23: Content of the slop cut reservoirs for the optimal solution. The content is reduced by a flux shown in figure 7.22.

Remark 7.3 *For our calculations we chose pumps for the reservoirs that have an upper limit of*

$$\hat{u}_1, \hat{u}_2 \leq 30.$$

If we use the same type of pump as for the reflux with an upper limit of 15 mol per hour, the process runs longer and the profit decreases to 0.849732. *From an optimal control point of view the solution of this problem is structurally different than the one obtained above, as the flux controls are constraint–seeking instead of compromise–seeking (compare chapter 2). Within our approach nothing has to be changed, though.*

The run time of the algorithm is within the range of two or three minutes, depending on the initialization of the variables.

7.4 Summary

Three challenging applications have been presented and solved by the *MS MINTOC* algorithm. The first one includes nondifferentiabilities in the model functions and point resp. path constraints that cause severe changes in the switching structure of optimal solutions. We know from the theoretic work in chapter 4 that the optimal solution for the velocity–constrained case that consists of an infinite switching to stay as close as possible to the limit on the constrained arc, can be approximated arbitrarily close. In section 7.1 we calculate such solutions explicitely. Furthermore, by prespecifying a higher tolerance on the energy consumption, our method delivers

a solution with less switches, making driving more comfortable for driver and passengers.

The second application treats a highly unstable system. Our method is used to identify strength and length of phase resetting stimuli. The combination of Bock's direct multiple shooting method, a convexification and relaxation and an adaptive refinement of the control grid allows the determination of the optimal stimulus, although the objective function landscape is highly nonconvex with the optimal solution "hidden" in a very narrow channel. This was demonstrated by an a posteriori analysis of the objective function landscape by a computationally expensive simulation.

The third application includes transition stages, coupled and decoupled interior point inequalities and equalities, differential and algebraic variables, free, time–independent parameters, free stage lengths and continuous as well as binary control functions. As to our knowledge, for the first time an optimal control problem of this challenging type is solved to optimality. The process under consideration is more efficient by more than 13% compared to the best known solution in the literature, when the proposed time– and tray–dependent recycling of the slop cuts is applied.

Appendix A

Mathematical definitions and theorems

We state some fundamental definitions and theorems for the convenience of the reader.

A.1 Definitions

Definition A.1 (Convexity)
A function $\boldsymbol{F} : \mathbb{R}^n \mapsto \mathbb{R}^m$ *is convex, if for all* $\boldsymbol{x_1}, \boldsymbol{x_2} \in \mathbb{R}^n$ *it holds for all* $\alpha \in [0, 1]$ *that*

$$\alpha \boldsymbol{F}(\boldsymbol{x_1}) + (1 - \alpha)\boldsymbol{F}(\boldsymbol{x_2}) \quad \geq \quad \boldsymbol{F}(\alpha \boldsymbol{x_1} + (1 - \alpha)\boldsymbol{x_2}). \tag{A.1}$$

A subset $\mathcal{K} \subseteq \mathcal{X}$ *of a real linear space* $\mathcal{X}$ *is convex, if for all* $\boldsymbol{x_1}, \boldsymbol{x_2} \in \mathcal{K}$ *and* $\alpha \in [0, 1]$ *also* $\alpha \boldsymbol{x_1} + (1 - \alpha)\boldsymbol{x_2} \in \mathcal{K}$.

Definition A.2 (Convex combination)
Let $\mathcal{X}$ *be some real linear space and* $\mathcal{K}$ *a set in* $\mathcal{X}$. *Then a convex combination of pairwise different elements from* $\mathcal{K}$ *is a linear combination of the form*

$$x = \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_n x_n$$

for some $n > 0$, *where each* $x_i \in \mathcal{K}$, *each* $\alpha_i \in \mathbb{R}$, $\alpha_i \geq 0$ *and* $\sum_i \alpha_i = 1$, $i = 1 \ldots n$.

Definition A.3 (Convex hull)
Let $\mathrm{conv}(\mathcal{K})$ *be the set of all convex combinations from* $\mathcal{K}$, *subset of some real linear space* $\mathcal{X}$. *We call* $\mathrm{conv}(\mathcal{K})$ *the convex hull of* $\mathcal{K}$.

Remark A.4 *The convex hull is sometimes also called convex envelope or convex closure of* $\mathcal{K}$. *The convex hull is a convex set, and is the smallest convex set which contains* $\mathcal{K}$. *A set* $\mathcal{K}$ *is convex if and only if* $\mathcal{K} = \mathrm{conv}(\mathcal{K})$.

Definition A.5 (Extreme point)
Let $\mathcal{X}$ be a real linear space and $\mathcal{K}$ a set in $\mathcal{X}$. A point $x \in \mathcal{K}$ is an extreme point of $\mathcal{K}$, if whenever

$$x = \alpha x_1 + (1-\alpha)x_2, \quad x_1, x_2 \in \mathcal{K}, \quad 0 < \alpha < 1,$$

then $x_1 = x_2$.

A.2 Theorems

Theorem A.6 (Zorn's lemma)
Let $\mathcal{X}$ be a nonempty partially ordered set with the property that every completely ordered subset has an upper bound in $\mathcal{X}$. Then $\mathcal{X}$ contains at least one maximal element.

Theorem A.7 (Krein–Milman)
Let $\mathcal{X}$ be a real linear topological space with the property that for any two distinct points x_1 and x_2 of $\mathcal{X}$ there is a continuous linear functional x' with

$$x'(x_1) \neq x'(x_2).$$

Then each nonempty compact set $\mathcal{K}$ of $\mathcal{X}$ has at least one extreme point.

The proof is based upon Zorn's lemma and can be found in, e.g., Hermes & Lasalle (1969) or any standard textbook on functional analysis.

Theorem A.8 (Gronwall inequality)
Let $x(\cdot) : [t_0, t_f] \mapsto \mathbb{R}$ be a continuous function, $t_0 \leq t \leq t_f$, $\alpha, \beta \in \mathbb{R}$ and $\beta > 0$. If

$$x(t) \leq \alpha + \beta \int_{t_0}^{t} x(\tau)\, \mathrm{d}\tau \tag{A.2}$$

then

$$x(t) \leq \alpha\, e^{\beta(t-t_0)} \tag{A.3}$$

for all $t \in [t_0, t_f]$.

A proof is given, e.g., in Walter (1993).

Theorem A.9 (Aumann 1965)
Let $\boldsymbol{F}$ be a measurable function defined on the interval $[t_0, t_f]$ with values $\boldsymbol{F}(t)$ nonempty compact subsets of a fixed compact set in $\mathbb{R}^n$. We write

$$\int_{t_0}^{t_f} \boldsymbol{F}(\tau)\, \mathrm{d}\tau := \left\{ \int_{t_0}^{t_f} \boldsymbol{f}(\tau)\, \mathrm{d}\tau : \boldsymbol{f} \text{ measurable, } \boldsymbol{f}(\tau) \in \boldsymbol{F}(\tau), \tau \in [t_0, t_f] \right\}$$

and $\mathrm{conv}(\boldsymbol{F})$ for the function with values $\mathrm{conv}(\boldsymbol{F}(t))$, the convex hull of $\boldsymbol{F}(t)$. Then we have

$$\int_{t_0}^{t_f} \boldsymbol{F}(\tau)\, \mathrm{d}\tau = \int_{t_0}^{t_f} \mathrm{conv}(\boldsymbol{F}(\tau))\, \mathrm{d}\tau$$

and both are convex, compact subsets of $\mathbb{R}^n$.

Appendix B

Details of the Fishing problem

B.1 Parameter values

The parameters and initial values used for the fishing problems (1.19) resp. (6.12) are as follows:

$$
\begin{aligned}
c_0 &= 0.4, && \text{(B.1a)}\\
c_1 &= 0.2, && \text{(B.1b)}\\
x_{00} &= 0.5, && \text{(B.1c)}\\
x_{01} &= 0.7, && \text{(B.1d)}\\
t_0 &= 0, && \text{(B.1e)}\\
t_f &= 12. && \text{(B.1f)}
\end{aligned}
$$

B.2 First order necessary conditions of optimality

We will consider the continuous optimal control problem obtained by a relaxation of problem (6.12), see page 119. We will write $\boldsymbol{x}$, $\boldsymbol{\lambda}$ and w for $\boldsymbol{x}(t)$, $\boldsymbol{\lambda}(t)$ and $w(t)$, respectively. The Hamiltonian, compare page 24, is given by

$$
\begin{aligned}
\mathcal{H} &= -L(\boldsymbol{x}) + \lambda^T f(\boldsymbol{x}, w)\\
&= -(x_0 - 1)^2 - (x_1 - 1)^2 \qquad \text{(B.2)}\\
&\quad + \lambda_0(x_0 - x_0x_1 - c_0x_0w) + \lambda_1(-x_1 + x_0x_1 - c_1x_1w),
\end{aligned}
$$

as we transform the minimization to a maximization problem ($\min \int L \mapsto \max \int -L$). There is no end–point Lagrangian $\boldsymbol{\psi}(\boldsymbol{x}(t_f), \boldsymbol{\nu})$, as no Mayer–term and end–point constraints are present. Furthermore we do not have any path constraints $\boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u})$.

The maximum principle thus reads as

$$\begin{aligned}
\dot{\boldsymbol{x}}(t) &= \mathcal{H}_{\boldsymbol{\lambda}}(\boldsymbol{x}(t), w(t), \boldsymbol{\lambda}(t)) = \boldsymbol{f}(\boldsymbol{x}(t), w(t)), && \text{(B.3a)}\\
\dot{\boldsymbol{\lambda}}^T(t) &= -\mathcal{H}_{\boldsymbol{x}}(\boldsymbol{x}(t), w(t), \boldsymbol{\lambda}(t)), && \text{(B.3b)}\\
\boldsymbol{x}(t_0) &= \boldsymbol{x_0}, && \text{(B.3c)}\\
\boldsymbol{\lambda}^T(t_f) &= \boldsymbol{0}, && \text{(B.3d)}\\
w(t) &= \arg\max_{0\le w\le 1} \mathcal{H}(\boldsymbol{x}(t), w(t), \boldsymbol{\lambda}(t)), && \text{(B.3e)}
\end{aligned}$$

B.3 Solution via an indirect method

The differential equations for the adjoint variables read as

$$\begin{aligned}
\dot{\lambda_0} &= -\mathcal{H}_{x_0} = 2(x_0-1) - \lambda_0(1 - x_1 - c_0 w) - \lambda_1 x_1 && \text{(B.4a)}\\
\dot{\lambda_1} &= -\mathcal{H}_{x_1} = 2(x_1-1) + \lambda_0 x_0 - \lambda_1(-1 + x_0 - c_1 w) && \text{(B.4b)}
\end{aligned}$$

with the transversality conditions

$$\lambda_0(t_f) = \lambda_1(t_f) = 0. \qquad \text{(B.4c)}$$

We define the switching function, see page 26, as

$$\begin{aligned}
\sigma(\boldsymbol{x}, w, \boldsymbol{\lambda}) &= \mathcal{H}_w = \boldsymbol{\lambda}^T \frac{\partial \boldsymbol{f}}{\partial w}\\
&= -c_0\lambda_0 x_0 - c_1\lambda_1 x_1 && \text{(B.5)}
\end{aligned}$$

and obtain the following boundary value problem from the first order necessary conditions of optimality (B.3)

$$\begin{aligned}
\dot{x_0}(t) &= x_0 - x_0 x_1 - c_0 x_0\, w && \text{(B.6a)}\\
\dot{x_1}(t) &= -x_1 + x_0 x_1 - c_1 x_1\, w && \text{(B.6b)}\\
x_0(t_0) &= x_{00} && \text{(B.6c)}\\
x_1(t_0) &= x_{01} && \text{(B.6d)}\\
\dot{\lambda_0} &= 2(x_0-1) - \lambda_0(1 - x_1 - c_0 w) - \lambda_1 x_1 && \text{(B.6e)}\\
\dot{\lambda_1} &= 2(x_1-1) + \lambda_0 x_0 - \lambda_1(-1 + x_0 - c_1 w) && \text{(B.6f)}\\
\lambda_0(t_f) &= 0 && \text{(B.6g)}\\
\lambda_1(t_f) &= 0 && \text{(B.6h)}\\
w(t) &= \arg\max_w \mathcal{H} = \begin{cases} 1 & \text{for } \sigma(\boldsymbol{x}, w, \boldsymbol{\lambda}) > 0 \\ 0 & \text{for } \sigma(\boldsymbol{x}, w, \boldsymbol{\lambda}) < 0 \\ w_{sing} & \text{for } \sigma(\boldsymbol{x}, w, \boldsymbol{\lambda}) = 0 \end{cases} && \text{(B.6i)}
\end{aligned}$$

We differentiate $\sigma(\boldsymbol{x}, w, \boldsymbol{\lambda}) = 0$ with respect to time as explained in section 2.1.2 to obtain an explicit representation of the singular control $w_{sing}(\cdot)$. The switching function is given by

$$\sigma(\boldsymbol{x}, w, \boldsymbol{\lambda}) \;=\; -c_0\lambda_0 x_0 - c_1\lambda_1 x_1.$$

Differentiation with respect to t yields

$$\begin{aligned}
\frac{\mathrm{d}\sigma}{\mathrm{d}t}(\boldsymbol{x}, w, \boldsymbol{\lambda}) &= -\left(c_0\left(\dot{\lambda_0}x_0 + \lambda_0\dot{x_0}\right) + c_1\left(\dot{\lambda_1}x_1 + \lambda_1\dot{x_1}\right)\right) \\
&= -\left(c_0((2(x_0-1) - \lambda_0(1 - x_1 - c_0u) - \lambda_1 x_1)x_0\right. \\
&\quad + \lambda_0(x_0 - x_0x_1 - c_0x_0u)) \\
&\quad + c_1((2(x_1-1) + \lambda_0x_0 - \lambda_1(-1 + x_0 - c_1u))x_1 \\
&\quad \left. + \lambda_1(-x_1 + x_0x_1 - c_1x_1u))\right) \\
&= -c_0 2(x_0-1)x_0 - c_1 2(x_1-1)x_1 + c_0\lambda_1 x_0x_1 - c_1\lambda_0 x_0x_1 \\
&= -2c_0x_0^2 + 2c_0x_0 + (c_0\lambda_1 - c_1\lambda_0)x_0x_1 - 2c_1x_1^2 + 2c_1x_1
\end{aligned}$$

This expression still does not depend explicitly upon w (as expected, see remark 2.8), therefore we differentiate once more and obtain

$$\begin{aligned}
\frac{\mathrm{d}^2\sigma}{\mathrm{d}^2t}(\boldsymbol{x}, w, \boldsymbol{\lambda}) &= -4c_0\dot{x_0}x_0 + 2c_0\dot{x_0} - 4c_1\dot{x_1}x_1 + 2c_1\dot{x_1} \\
&\quad + c_0\dot{\lambda_1}x_0x_1 - c_1\dot{\lambda_0}x_0x_1 + (c_0\lambda_1 - c_1\lambda 1)(x_0\dot{x_1} + \dot{x_0}x_1) \\
&= -4c_0x_0^2 + 4c_0x_0^2x_1 + 4c_0^2x_0^2w + 2c_0x_0 - 2c_0x_0x_1 - 2c_0^2x_0w \\
&\quad + 4c_1x_1^2 - 4c_1x_0x_1^2 + 4c_1^2x_1^2w - 2c_1x_1 + 2c_1x_0x_1 - 2c_1^2x_1w \\
&\quad + c_0x_0x_1\left(2(x_1-1) + \lambda_0x_0 + \lambda_1 - \lambda_1x_0 + c_1\lambda_1w\right) \\
&\quad - c_1x_0x_1\left(2(x_0-1) - \lambda_1x_1 - \lambda_0 + \lambda_0x_1 + c_0\lambda_0w\right) \\
&\quad + c_0\lambda_1x_0\left(-x_1 + x_0x_1 - c_1x_1w\right) \\
&\quad + c_0\lambda_1x_1\left(+x_0 - x_0x_1 - c_0x_0w\right) \\
&\quad - c_1\lambda_0x_0\left(-x_1 + x_0x_1 - c_1x_1w\right) \\
&\quad - c_1\lambda_0x_1\left(+x_0 - x_0x_1 - c_0x_0w\right) \\
&= w\left(4c_0^2x_0^2 - 2c_0^2x_0 - 2c_1^2x_1 + 4c_1^2x_1^2 - c_0^2\lambda_1x_0x_1 + c_1^2\lambda_0x_0x_1\right) \\
&\quad - 4c_0x_0^2 + 4c_1x_1^2 + 2c_0x_0 - 2c_1x_1 + (-4c_0 + 4c_1 + c_0\lambda_1 + c_1\lambda_0)x_0x_1 \\
&\quad + (4c_0 - 2c_1 + c_0\lambda_0 - c_1\lambda_0)x_0^2x_1 + (-4c_1 + 2c_0 + c_1\lambda_1 - c_0\lambda_1)x_0x_1^2
\end{aligned}$$

The second time derivative of the switching function $\frac{\mathrm{d}^2\sigma}{\mathrm{d}^2t}(\boldsymbol{x}, w, \boldsymbol{\lambda}) = 0$ does depend explicitly upon the control variable $w(\cdot)$, therefore we can determine w_{sing} as

$$\begin{aligned}
w_{sing}(\boldsymbol{x}, \boldsymbol{\lambda}) &= -\big(-4c_0x_0^2 + 4c_1x_1^2 + 2c_0x_0 - 2c_1x_1 + (-4c_0 + 4c_1 + c_0\lambda_1 + c_1\lambda_0)x_0x_1 \\
&\quad + (4c_0 - 2c_1 + c_0\lambda_0 - c_1\lambda_0)x_0^2x_1 + (-4c_1 + 2c_0 + c_1\lambda_1 - c_0\lambda_1)x_0x_1^2\big) \\
&\quad / \left(4c_0^2x_0^2 - 2c_0^2x_0 - 2c_1^2x_1 + 4c_1^2x_1^2 - c_0^2\lambda_1x_0x_1 + c_1^2\lambda_0x_0x_1\right)
\end{aligned}$$

$w_{sing}(\boldsymbol{x}, \boldsymbol{\lambda})$ does depend explicitly upon the state variables $\boldsymbol{x}$ and on the Lagrange multipliers $\boldsymbol{\lambda}$. In this case it is possible though to eliminate the Lagrange multipliers by exploiting the invariants $\sigma = \frac{\mathrm{d}\sigma}{\mathrm{d}t} = 0$. From

$$\sigma(\boldsymbol{x}, w, \boldsymbol{\lambda}) \;=\; -c_0\lambda_0 x_0 - c_1\lambda_1 x_1 = 0$$

we can deduce

$$\lambda_0 \;=\; -c_1\lambda_1 x_1 / c_0 x_0 \tag{B.7}$$

and insert it into the first time derivative of the switching function

$$\begin{aligned}
\frac{\mathrm{d}\sigma}{\mathrm{d}t}(\boldsymbol{x}, w, \boldsymbol{\lambda}) &= -2c_0x_0^2 + 2c_0x_0 + (c_0\lambda_1 - c_1\lambda_0)x_0x_1 - 2c_1x_1^2 + 2c_1x_1 \\
&= -2c_0x_0^2 + 2c_0x_0 + (c_0\lambda_1 + c_1c_1\lambda_1x_1/c_0x_0)x_0x_1 - 2c_1x_1^2 + 2c_1x_1 \\
&= -2c_0x_0^2 + 2c_0x_0 + (c_0 + c_1^2x_1/c_0x_0)\lambda_1x_0x_1 - 2c_1x_1^2 + 2c_1x_1 = 0.
\end{aligned}$$

From this expression we can deduce λ_1 as a function of $\boldsymbol{x}$,

$$\begin{aligned}
\lambda_1 &= (-2c_0x_0^2 + 2c_0x_0 - 2c_1x_1^2 + 2c_1x_1)\frac{-1}{(c_0 + c_1^2x_1/c_0x_0)x_0x_1} \\
&= \frac{2(c_0x_0^2 - c_0x_0 + c_1x_1^2 - c_1x_1)}{(c_0 + c_1^2x_1/c_0x_0)x_0x_1}
\end{aligned}$$

which yields λ_0 when inserted in (B.7) as

$$\begin{aligned}
\lambda_0 &= -c_1\frac{2(c_0x_0^2 - c_0x_0 + c_1x_1^2 - c_1x_1)}{(c_0 + c_1^2x_1/c_0x_0)x_0x_1c_0x_0}x_1 \\
&= \frac{-2c_1(c_0x_0^2 - c_0x_0 + c_1x_1^2 - c_1x_1)}{(c_0^2x_0 + c_1^2x_1)x_0}.
\end{aligned}$$

The singular control can now be expressed as a *feedback control* $w_{sing}(\boldsymbol{x})$ depending on $\boldsymbol{x}$ only:

$$\begin{aligned}
w_{sing}(\boldsymbol{x}) &= -\big(-4c_0x_0^2 + 4c_1x_1^2 + 2c_0x_0 - 2c_1x_1 + (-4c_0 + 4c_1 + c_0\lambda_1 + c_1\lambda_0)x_0x_1 \\
&\quad + (4c_0 - 2c_1 + c_0\lambda_0 - c_1\lambda_0)x_0^2x_1 + (-4c_1 + 2c_0 + c_1\lambda_1 - c_0\lambda_1)x_0x_1^2\big) \\
&\quad / \big(4c_0^2x_0^2 - 2c_0^2x_0 - 2c_1^2x_1 + 4c_1^2x_1^2 - c_0^2\lambda_1x_0x_1 + c_1^2\lambda_0x_0x_1\big) \\
&= (\, c_0^3x_0^3 - c_1^3x_1^3 + c_0^3x_0^2x_1 - c_1^3x_0x_1^2 + 2c_0x_0x_1^2c_1^2 \\
&\quad - 2c_1x_0^2x_1c_0^2 - 4c_0^2x_0c_1x_1^2 + 2c_0^2x_0c_1x_1 \\
&\quad + 4c_1^2x_1c_0x_0^2 - 2c_1^2x_1c_0x_0 - x_0^3x_1c_0^3 \\
&\quad + x_0^2x_1^2c_1^3 + x_0x_1^3c_1^3 - 2x_0^2x_1^2c_1^2c_0 + x_0^3x_1c_1c_0^2 \\
&\quad - x_0x_1^3c_0c_1^2 - x_0^3x_1c_1^2c_0 - x_0^2x_1^2c_0^3 + 2x_0^2x_1^2c_0^2c_1 \\
&\quad + x_0x_1^3c_0^2c_1 \,) \\
&\quad / \,(\, c_0^4x_0^3 + 2c_0^2x_0^2c_1^2x_1 \\
&\quad - 2c_0^2x_0c_1^2x_1 + 2c_1^2x_1^2c_0^2x_0 + c_1^4x_1^3 - c_0^3x_0c_1x_1^2 \\
&\quad + c_0^3x_0c_1x_1 - c_1^3x_1c_0x_0^2 + c_1^3x_1c_0x_0 \,)
\end{aligned}$$

With an explicit representation of the singular control, either in the form $w_{sing}(\boldsymbol{x}, \boldsymbol{\lambda})$ or in feedback form, the boundary value problem is complete. The trajectory obtained by a multiple shooting solution with 23 intervals of problem (B.6) yields an objective value of $\Phi[\boldsymbol{x}, w] = 1.34408$. The corresponding trajectories, the switching function and the control function $w(\cdot)$ are depicted in picture B.1 for an assumed bang–bang–singular structure.

Remark B.1 *This structure is not necessarily optimal. Indeed, by investigating the relaxed solutions show in figure 6.7 there is also the possibility that a very short additional bang–bang arc with value* 0 *occurs between the* 1*–arc and the singular arc. Because of its shortness it is not relevant for the objective value up to a precision of* 10^{-6}*, though.*

B.4 Nonconvexity of the switching time approach

In this section we want to have a look at convexity issues in the switching time approach, in the special case of the fishing problem. We will show how the direct multiple multiple shooting method is superior to direct single shooting with respect to convergence to local minima.
Let us consider the problem in switching time formulation as in section 5.2, but this time for $n_{\text{mos}} = 3$ stages with $w_0(t) = 0$, $w_1(t) = 1$ and $w_2(t) = 0$. The stage lengths h_0 and h_1 are varied in $[0, 12]$ with a step length of 0.1. The final stage length h_2 is chosen to satisfy constraint (5.4g). Figure B.2 shows the corresponding objective function landscape.
There are at least two local minima in this formulation, one at

$$\boldsymbol{h}^* = (2.61213, 1.72876, 7.65911)^T$$

with objective value $\Phi^* = 1.38275$ and one at

$$\hat{\boldsymbol{h}} = (8.98983, 2.02772, 0.98245)^T$$

with objective value $\hat{\Phi} = 4.55123$. It depends on the initialization of the optimization variables which minimum is reached by the SQP method. Here the direct multiple shooting approach shows one of its advantages compared to direct single shooting. If the multiple shooting nodes are initialized close to an expected behavior, as shown in figure B.4, the optimization iteration converges towards $\boldsymbol{h}^*$ also from initializations of $\boldsymbol{h}$ for which the direct single shooting approach will converge towards $\hat{\boldsymbol{h}}$. This behavior is depicted in figure B.3. In both figures only one differential state is shown, the biomass of the predator species looks similar.
The given example indicates strongly not to use the switching time approach without very accurate initial values for the stage lengths and only in combination with an all–at–once optimal control method, **not** with direct single shooting.

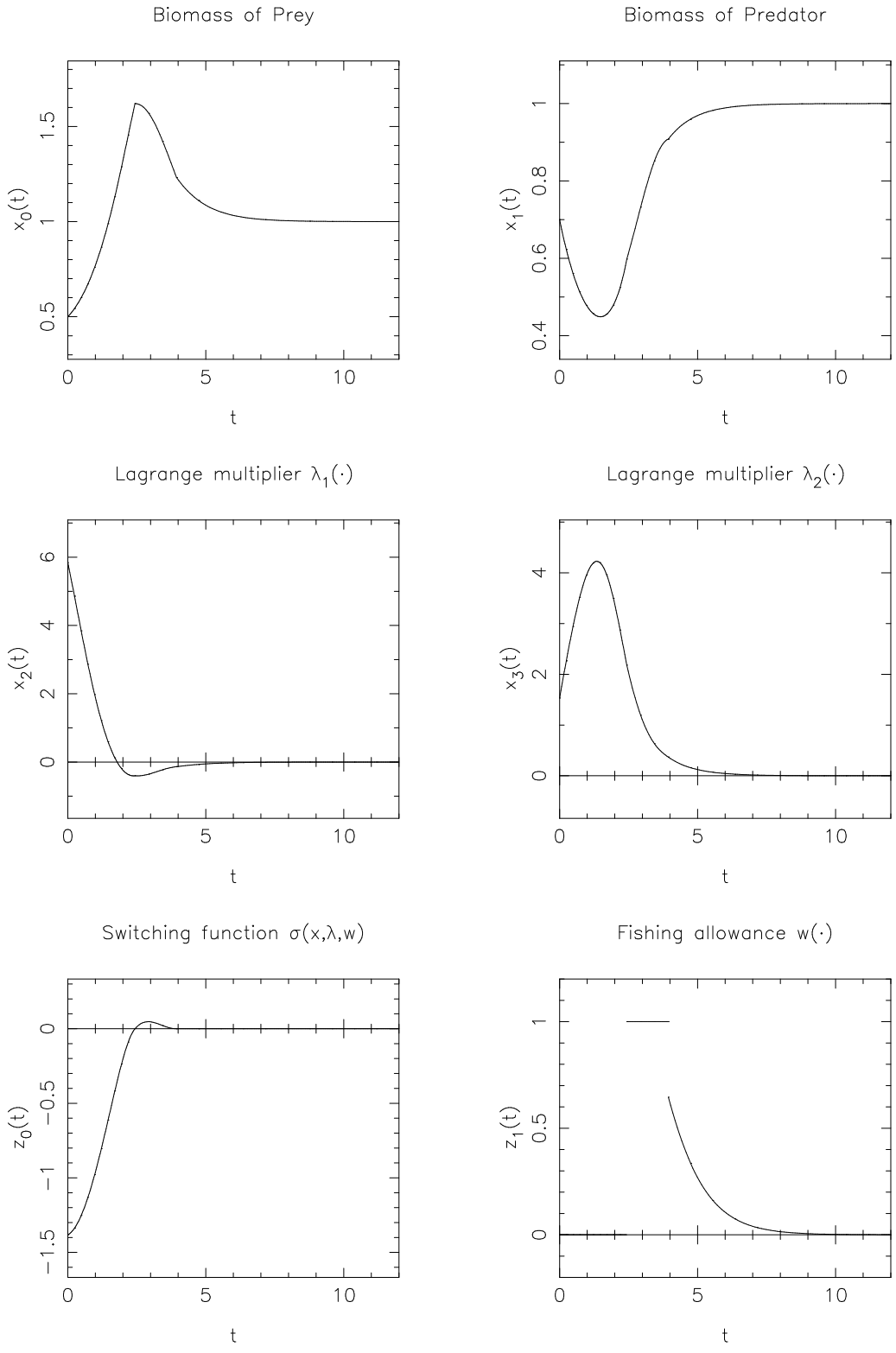

Figure B.1: The corresponding state and costate trajectories, the switching function and the control function $w(\cdot)$ for the relaxed fishing problem.

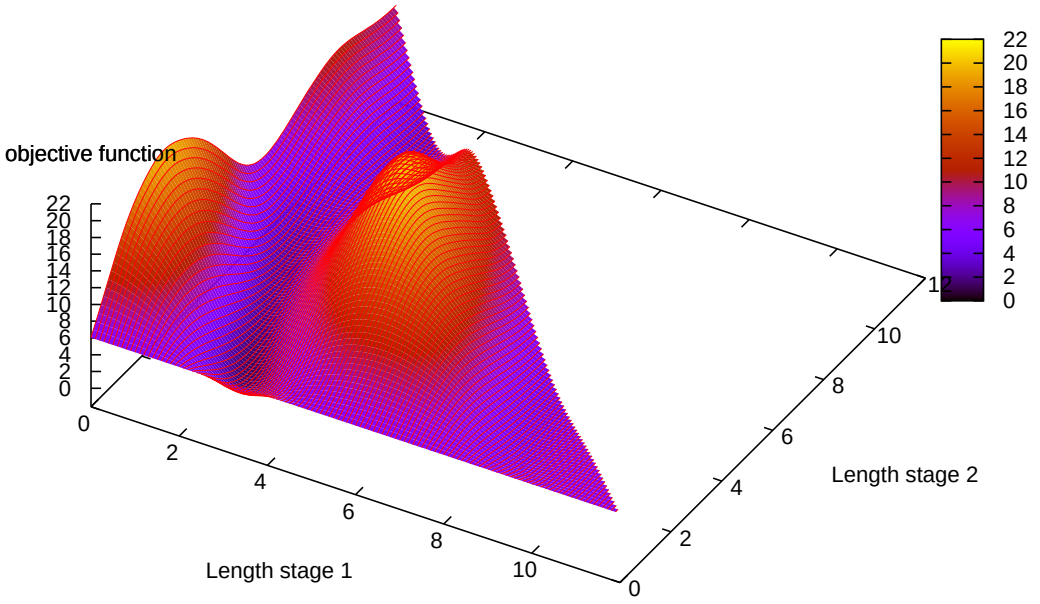

Figure B.2: Objective function value of the fishing problem in switching time formulation, dependent on $\tilde{t}_1$ and $\tilde{t}_2$, the begin respectively end of the second stage with $w(t) = w^2 = 1$.

B.5 Convex behavior of the relaxed problem

In this section we will investigate convexity of a relaxed problem formulation of (1.19). Let us consider the optimization problem resulting from the discretization of the control space to piecewise constant controls by a direct method of optimization. The feasible set is the hypercube in $\mathbb{R}^{n_\tau \times n_w}$ and thus convex. Concerning the objective value, we will show again some results obtained by simulation that give an insight into the issue.
We consider problem (1.19) with

$$n_\tau = 3, \Psi = \Psi_\tau = \{0.0, 5.5, 8.0\}, [t_0, t_f] = [0, 12] \tag{B.8}$$

with a partition of $[t_0, t_f]$ with respect to jumps in the control as in figures B.3 and B.4. Note that the independent optimization variables are now the constant values of $w(\cdot)$ on $[t_0, \tau_2]$, $[\tau_2, \tau_3]$ and $[\tau_3, t_f]$ and not the time points τ_i as before. The objective functions plotted in figure B.5 are convex for all four values of $w(\cdot)$ on the first stage. A comparison of figure B.5 with figure B.2 shows how disadvantageous the switching time approach is with respect to convexity.

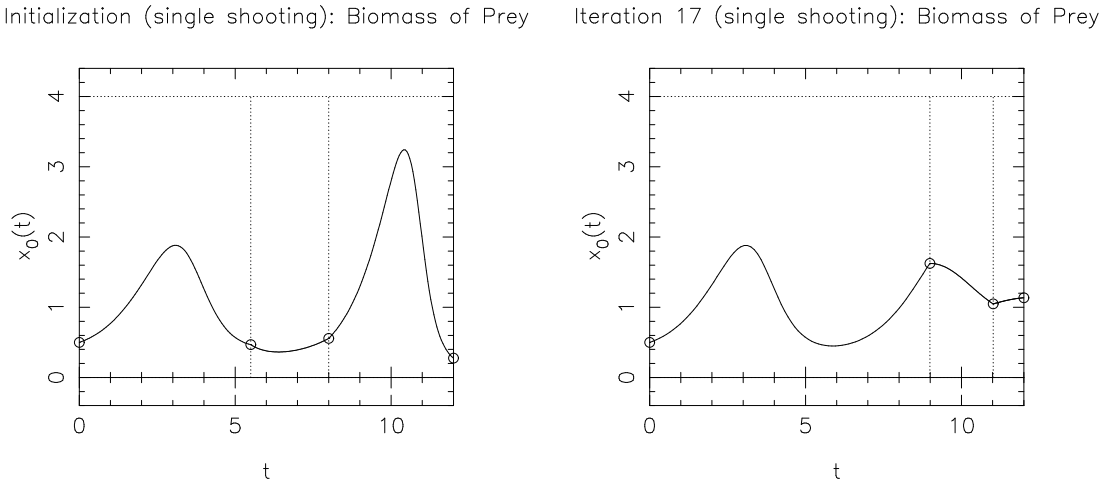

Figure B.3: The two dashed lines in each plot show the length of the three stages, h_0, h_1, and h_2. The first differential state, the biomass of the prey $x_0(t)$, is obtained by integration. The left plot shows the initialization of $\boldsymbol{h} = (5.5, 2.5, 4)^T$ with corresponding state trajectory. The right plot shows the variables when convergence has been achieved to an accuracy of 10^{-6} in the point $\hat{\boldsymbol{h}}$.

Although the objective function is not necessarily convex for all Ψ_τ in the complete feasible region, the hypercube $[0,1]^{n_\tau \times n_w}$, it is convex in a large vicinity of the solution. We consider another discretization, the one used in section 6.5. We proceed as follows. The binary control function $w(\cdot)$ of the relaxed problem (1.19) is discretized by 60 constant control approximations on an equidistant grid, compare section 2.2.3. This problem is solved to optimality. Figure B.6 shows the objective function in the vicinity of the optimal solution: for four selected stages the constant control on this stage is changing its value while the other 59 values are fixed to the optimal value[1]. Again, the trajectories indicate the convexity of the objective function at least in the vicinity of the optimum.
In a third scenario we discretize the control on an equidistant grid with 24 intervals and initialize the values such that the control function is identical to the one used as initialization in the switching time optimization in the preceding section, i.e., with $\boldsymbol{h} = (5.5, 2.5, 4)^T$. This is achieved by setting

$$q_i \;=\; \begin{cases} 1 & i = 11, 12, 13, 14, 15 \\ 0 & \text{else} \end{cases} .$$

The optimization does not encounter any problems related to local minima as in the switching time optimization approach, but does end in the global relaxed minimum simular to those shown in figure 6.7.

[1] we do not show all sixty graphs, but they are all convex

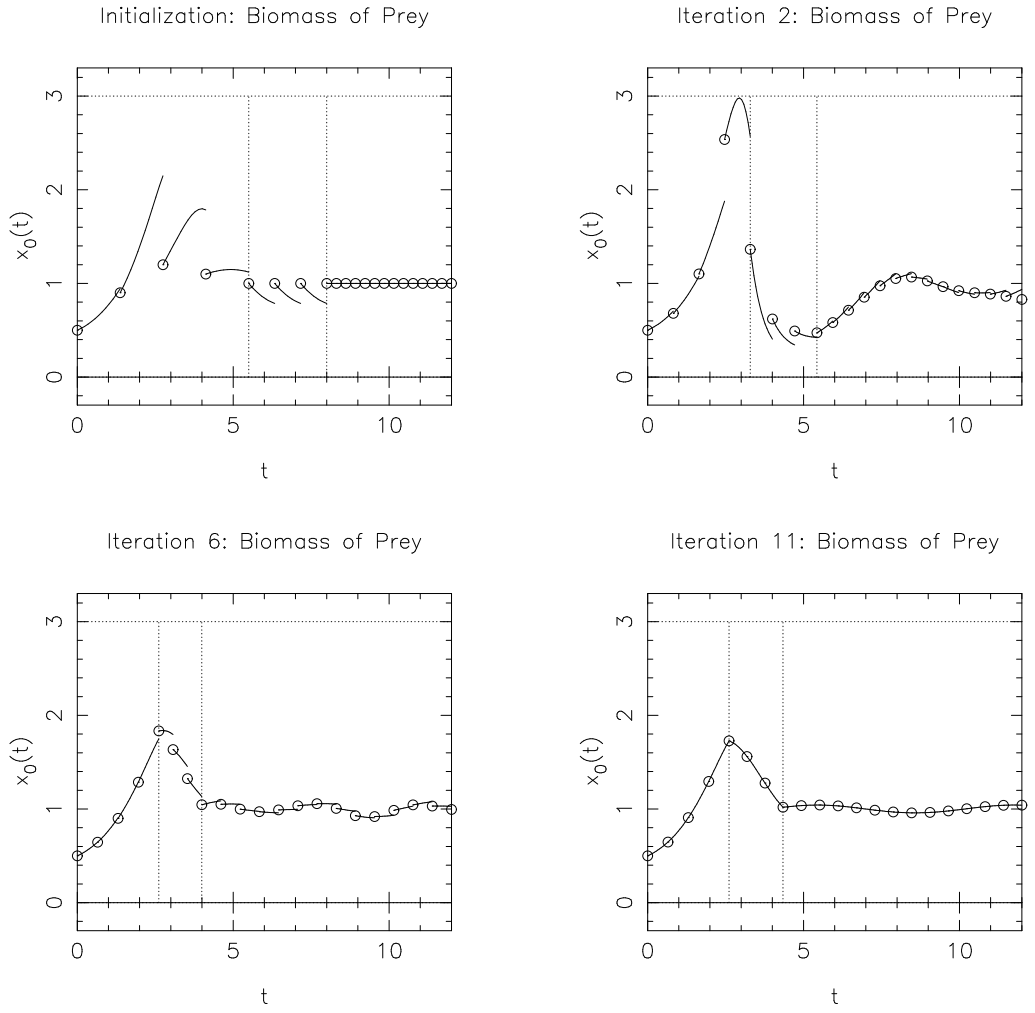

Figure B.4: SQP iterations of the fishing problem optimization in switching time formulation with three stages. The two dashed lines in each plot show the length of the three stages, h_0, h_1, and h_2. The circles indicate the values of the multiple shooting nodes $s_i^{x_0}$ for the first differential state, the biomass of the prey. The values $x_0(t)$ obtained by piecewise integration are also plotted. The top left plot shows the initialization of $\boldsymbol{h} = (5.5, 2.5, 4)^T$, identical to the single shooting approach, and $\boldsymbol{s^{x_0}}$. $\boldsymbol{s^{x_0}}$ has been entered manually such as to resemble the expected behavior of the biomass. The top right plot shows the optimization variables after two iterations, on the bottom left one after six. The bottom right plot shows the variables when convergence has been achieved to an accuracy of 10^{-6} in the point $\boldsymbol{h}^*$.

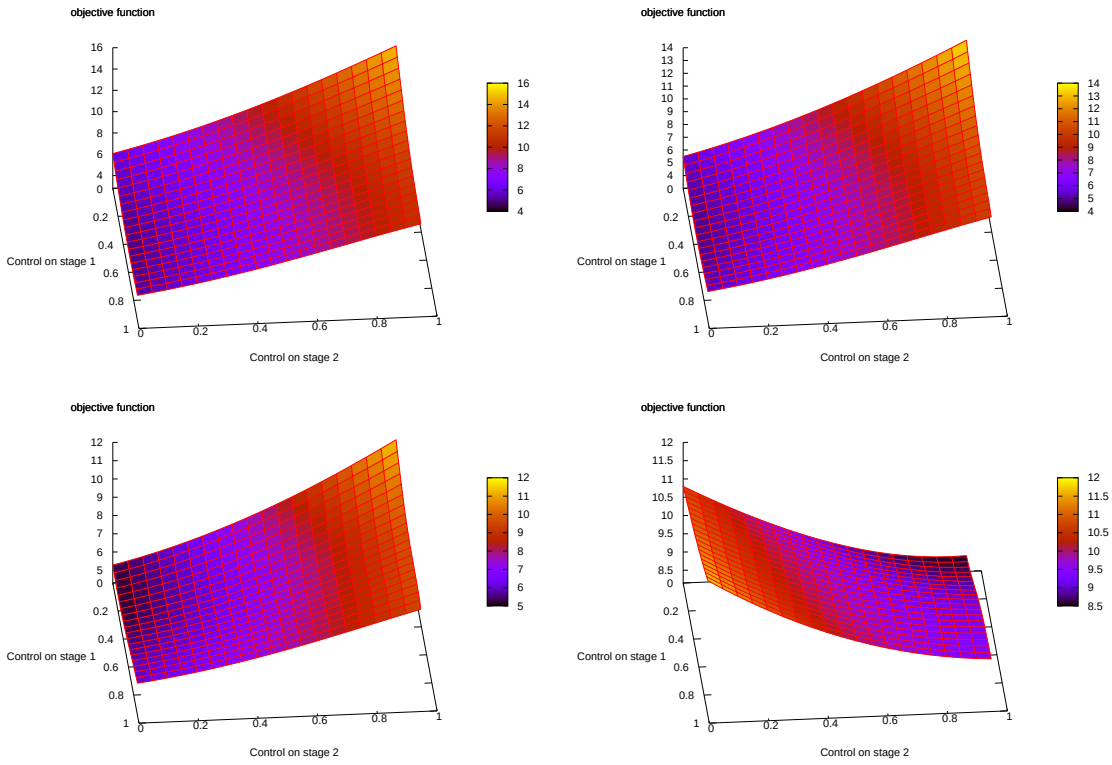

Figure B.5: The objective function value dependent on the constant value of $w(\cdot)$ on $[\tau_2, \tau_3]$ and $[\tau_3, t_f]$. $w(\cdot)$ on $[t_0, \tau_2]$ is fixed to 0, the optimal value 0.212344, 0.5 and 1 from top left to bottom right.

B.6 Formulation as a time–optimal control problem

The singular arc in the optimal trajectory is caused by the objective function. Instead of penalizing deviations from the steady state over a given time horizon, one could as well want to minimize the time to bring the system into this steady state $\boldsymbol{x_T}$. We reformulate problem (6.12) in this sense and obtain

$$\min_{\boldsymbol{x},w,T} \quad T \tag{B.9a}$$

subject to the ODE

$$\begin{aligned} \dot{x}_0(t) &= x_0(t) - x_0(t)x_1(t) - c_0 x_0(t)\, w(t), && \text{(B.9b)} \\ \dot{x}_1(t) &= -x_1(t) + x_0(t)x_1(t) - c_1 x_1(t)\, w(t), && \text{(B.9c)} \end{aligned}$$

initial values

$$\boldsymbol{x}(t_0) = \boldsymbol{x_0} = (0.5, 0.7)^T, \tag{B.9d}$$

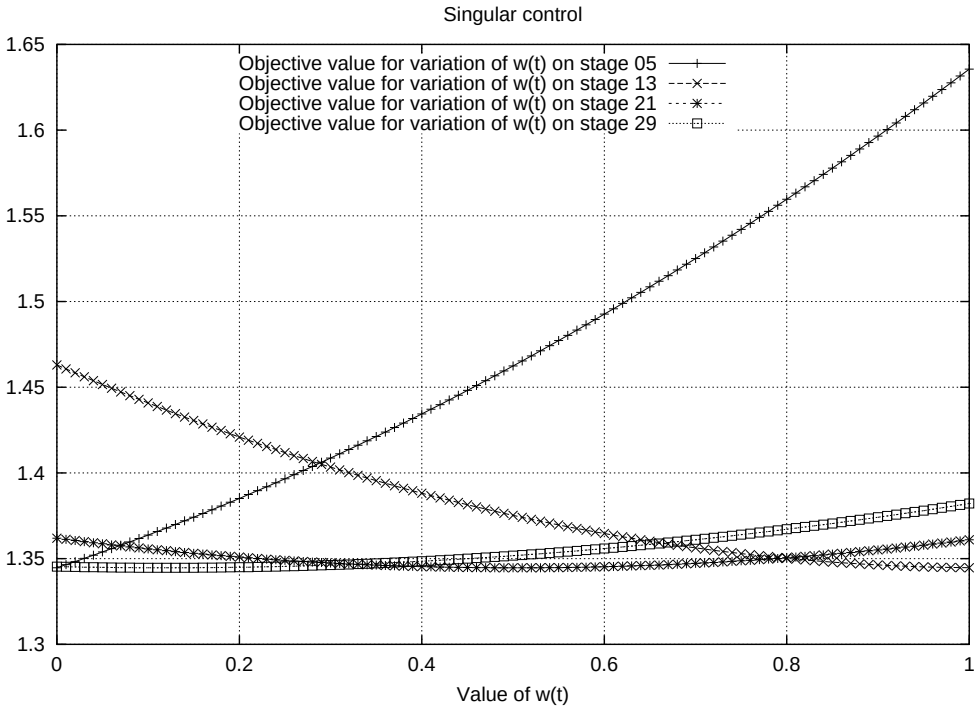

Figure B.6: The objective function in the vicinity of the optimal solution of the relaxed discretized problem (1.19): for four selected stages the control on this stage is changing its constant value while the other 59 values are fixed.

the terminal condition

$$\boldsymbol{x}(T) \quad = \quad \boldsymbol{x_T} = (1,1)^T, \tag{B.9e}$$

and the integer constraints

$$w(\cdot) \quad \in \quad \Omega(\Psi_{\text{free}}). \tag{B.9f}$$

Figure B.7 shows the optimal trajectory with one switching point at $\tilde{t}_1 = 2.68714$ that brings the system in $T^* = 4.31521$ to the desired state $\boldsymbol{x_T}$. This trajectory extended with $w(t) = 0, \boldsymbol{x}(t) = \boldsymbol{x_T}$ on $[T^*, t_f]$ is a suboptimal solution of problem (6.12) with objective value $\Phi = 1.39895$, compare the results of section 6.4.

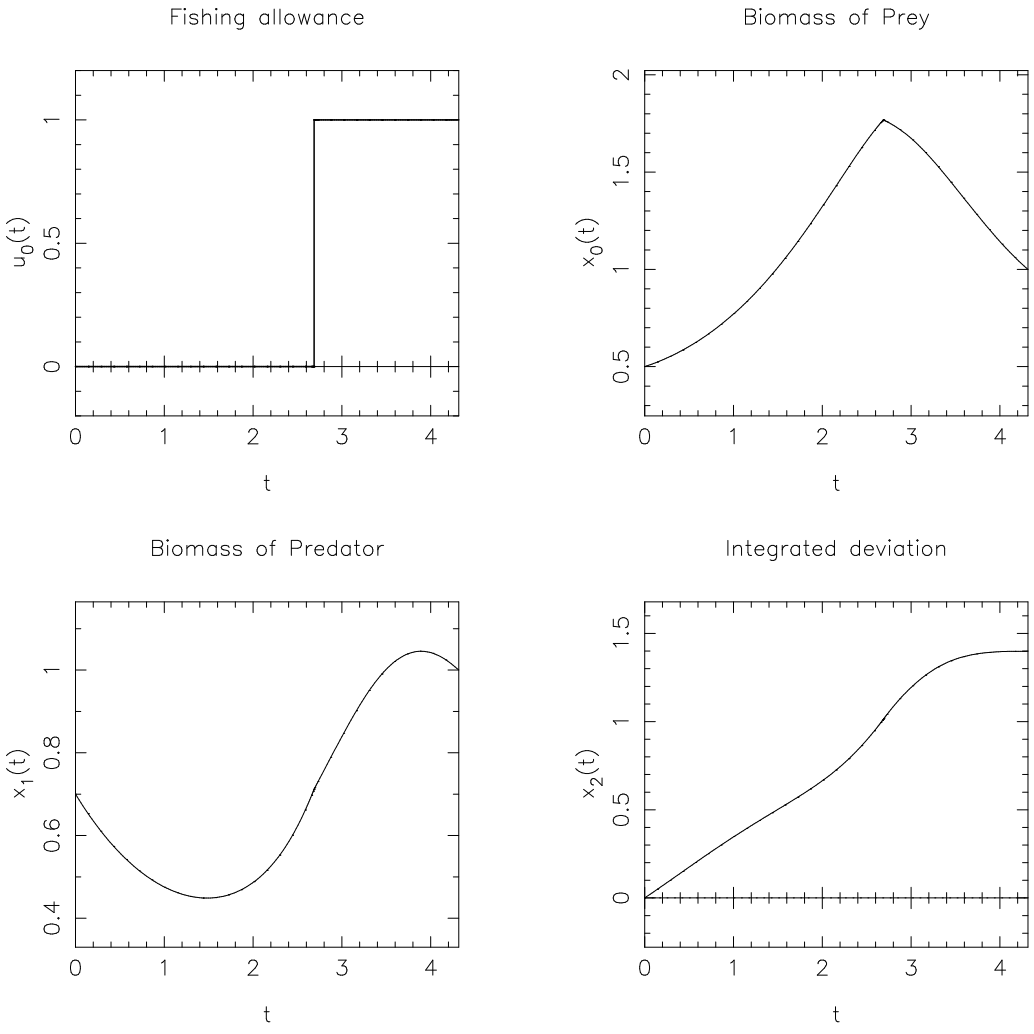

Figure B.7: Optimal trajectory for the time–optimal fishing problem.

Appendix C

Parameters of the subway optimization problem

$T^{\max}$	=	65	Maximal driving time, [sec]
S	=	2112	Driving distance, [ft]
S_4	=	700 or 1200	Distance for point constraint, [ft]
S_4	=	1200	Distance for path constraint start, [ft]
W	=	$78000 \in [W_{\text{empty}}, W_{\text{full}}] = [72000, 110000]$	Weight of the train, [lbs]
W_{eff}	=	$W + \frac{1}{10} W_{\text{empty}}$	Effective weight of the train, [lbs]
γ	=	3600/5280	Scaling factor for units, $[\frac{sec}{h}/\frac{ft}{mile}]$
a	=	100	Front surface of the train, $[ft^2]$
n_{wag}	=	10	Number of wagons
b	=	0.045	
c	=	$0.24 + \frac{0.034(n_{\text{wag}}-1)}{100 n_{\text{wag}}}$	
C	=	0.367	Constant braking when coasting
g	=	32.2	Gravity, $[ft/sec^2]$
e	=	$1.0 \in [0.7, 1]$	Percentage of working machines
v_1	=	$0.979474 \in [0.71, 1.03]$	Velocity limits, $[mph]$
v_2	=	$6.73211 \in [6.05, 6.86]$	
v_3	=	$14.2658 \in [13.07, 14.49]$	
v_4	=	22.0	Velocity limit point constraint, $[mph]$
v_5	=	24.0	Velocity limit path constraint, $[mph]$
a_1	=	$6017.611205 \in [5998.6162, 6118.9179]$	Accelerations, $[lbs]$
a_2	=	$12348.34865 \in [11440.7968, 17188.6252]$	
a_3	=	$11124.63729 \in [10280.0514, 15629.0954]$	
$u_{\max}$	=	4.4	Maximal deceleration, $[ft/sec^2]$
p_1	=	$106.1951102 \in [105.880645, 107.872258]$	Energy consumption
p_2	=	$180.9758408 \in [168.931957, 245.209888]$	
p_3	=	$354.136479 \in [334.626716, 458.188550]$	

If intervals are given, the first value corresponds to an empty train, the second one to the full one. For a value W in between we interpolate linearly.

The coefficients $b_i(w(t))$ and $c_i(w(t))$ are given by

$$\begin{aligned}
b_0(1) &= -0.1983670410E02,\\
b_1(1) &= \ \ 0.1952738055E03,\\
b_2(1) &= \ \ 0.2061789974E04,\\
b_3(1) &= -0.7684409308E03,\\
b_4(1) &= \ \ 0.2677869201E03,\\
b_5(1) &= -0.3159629687E02,\\
b_0(2) &= -0.1577169936E03,\\
b_1(2) &= \ \ 0.3389010339E04,\\
b_2(2) &= \ \ 0.6202054610E04,\\
b_3(2) &= -0.4608734450E04,\\
b_4(2) &= \ \ 0.2207757061E04,\\
b_5(2) &= -0.3673344160E03,
\end{aligned}$$

$$\begin{aligned}
c_0(1) &= \ \ 0.3629738340E02,\\
c_1(1) &= -0.2115281047E03,\\
c_2(1) &= \ \ 0.7488955419E03,\\
c_3(1) &= -0.9511076467E03,\\
c_4(1) &= \ \ 0.5710015123E03,\\
c_5(1) &= -0.1221306465E03,\\
c_0(2) &= \ \ 0.4120568887E02,\\
c_1(2) &= \ \ 0.3408049202E03,\\
c_2(2) &= -0.1436283271E03,\\
c_3(2) &= \ \ 0.8108316584E02,\\
c_4(2) &= -0.5689703073E01,\\
c_5(2) &= -0.2191905731E01.
\end{aligned}$$

Appendix D

Parameters of the calcium problem

The parameters $\boldsymbol{p}$ are given by

$$\begin{aligned} k_1 &= 0.09, & K_{11} &= 2.67, \\ k_2 &= 2.30066, & k_{12} &= 0.7, \\ k_3 &= 0.64, & k_{13} &= 13.58, \\ K_4 &= 0.19, & k_{14} &= 153.0, \\ k_5 &= 4.88, & K_{15} &= 0.16, \\ K_6 &= 1.18, & k_{16} &= 4.85, \\ k_7 &= 2.08, & K_{17} &= 0.05, \\ k_8 &= 32.24, & p_1 &= 100, \\ K_9 &= 29.09, & p_2 &= 5, \\ k_{10} &= 5.0, & T &= 22. \end{aligned}$$

The initial value $\boldsymbol{x_0}$ is

$$\begin{aligned} x_0(0) &= 0.03966, \\ x_1(0) &= 1.09799, \\ x_2(0) &= 0.00142, \\ x_3(0) &= 1.65431. \end{aligned}$$

The reference state $\boldsymbol{x^s}$, i.e., the concentrations corresponding to the unstable steady state surrounded by the limit cycle, have been determined by using the XPPAUT software, Ermentrout (2002), by path–following of a Hopf–bifurcation through variation of the parameter k_2 as

$$\begin{aligned} x_0^s &= 6.78677, \\ x_1^s &= 22.65836, \\ x_2^s &= 0.38431, \\ x_3^s &= 0.28977. \end{aligned}$$

Appendix E

Details of the waste cut problem

The material properties for the batch distillation correspond to the components 1, 3, and 7 from Domenech & Enjalbert (1981); Farhat *et al.* (1990)) and are given by

k	Boiling point(°C)	Antoine coefficients A_k	B_k	C_k
1	184.4	7.63846	1976.3	231.0
2	245.0	7.96718	2502.2	247.0
3	272.5	8.65385	3149.1	273.0

The parameters of the objective function value are set as follows. The slop cut disposal costs are

$$s_1 \;=\; s_2 = 0.$$

The product prices are

$$c^1_{\text{price}} \;=\; c^2_{\text{price}} = c^3_{\text{price}} = 4.5.$$

The energy consumption costs are

$$c_{\text{energy}} \;=\; 1.$$

If the feedstock purchase costs were taken into consideration, the product prices should be even higher to retain marginal profitability of the process.
The feed is assumed to contain all three components in equal amounts, i.e., $X_{k,0}|_{\tilde{t}_0} = 0.33$ for $k = 1, 2$. The starting mass of feed is set to $M|_{\tilde{t}_0} = 1$.

Figures E.1, E.3 and E.5 show the temperature and mole fraction profiles for the optimal solution of the batch recycling scenario, figures E.2, E.4 and E.6 the corresponding state trajectories of the solution of the mixed–integer optimal control problem.

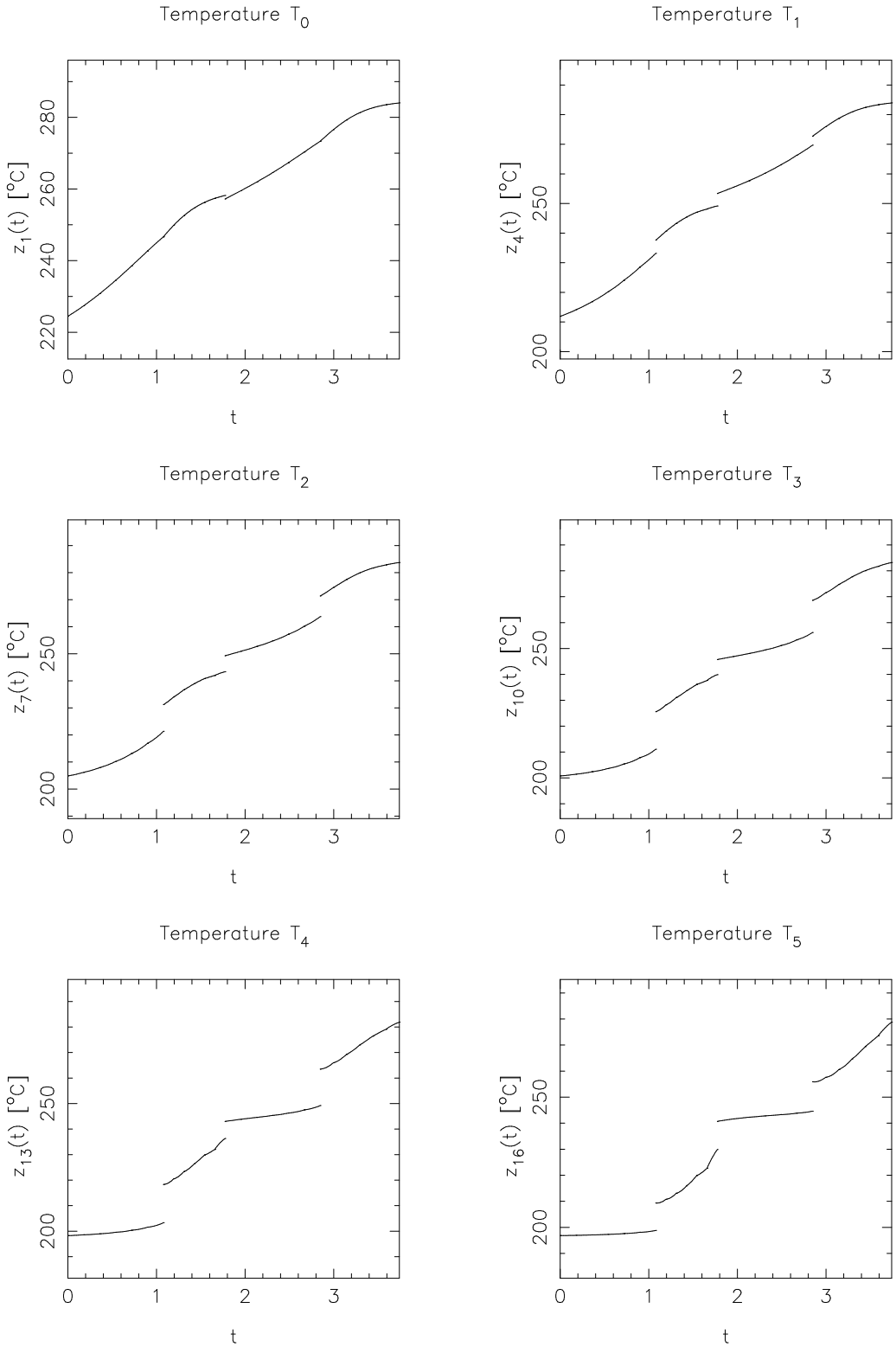

Figure E.1: Temperature profile for scenario B with batch recycling.

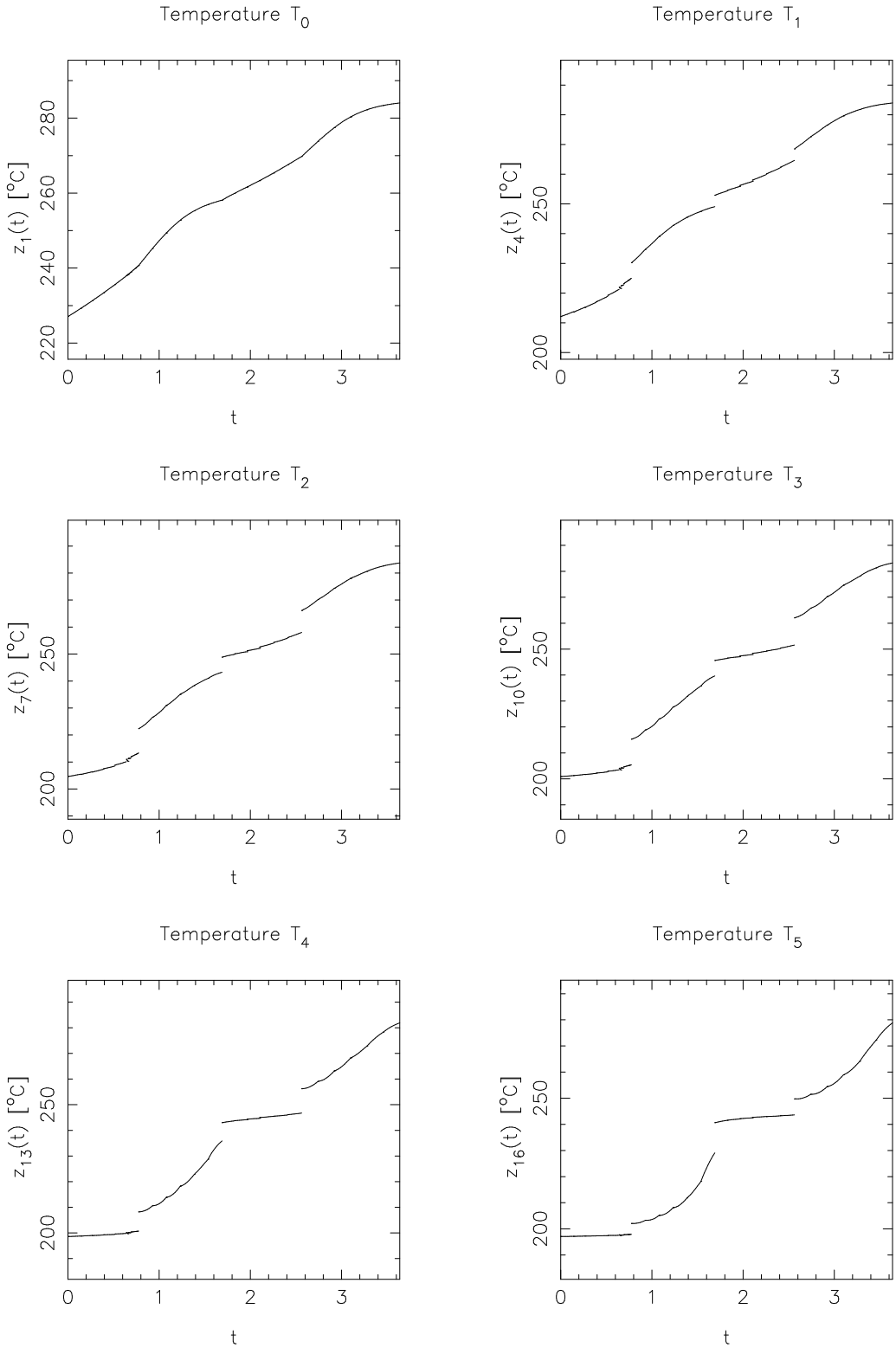

Figure E.2: Temperature profile for scenario C with flexible recycling.

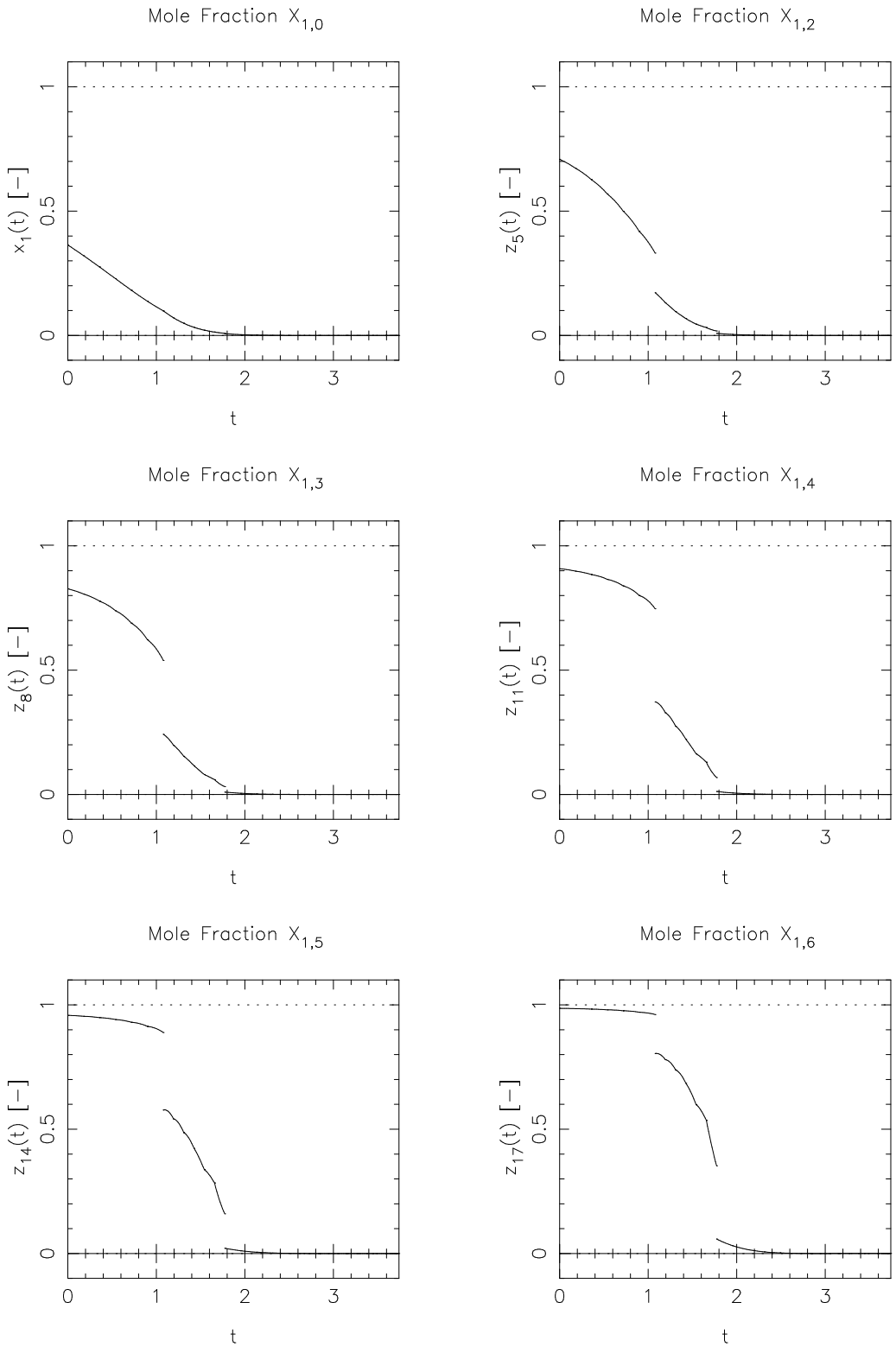

Figure E.3: Mole fraction profile of component 1 for scenario B with batch recycling.

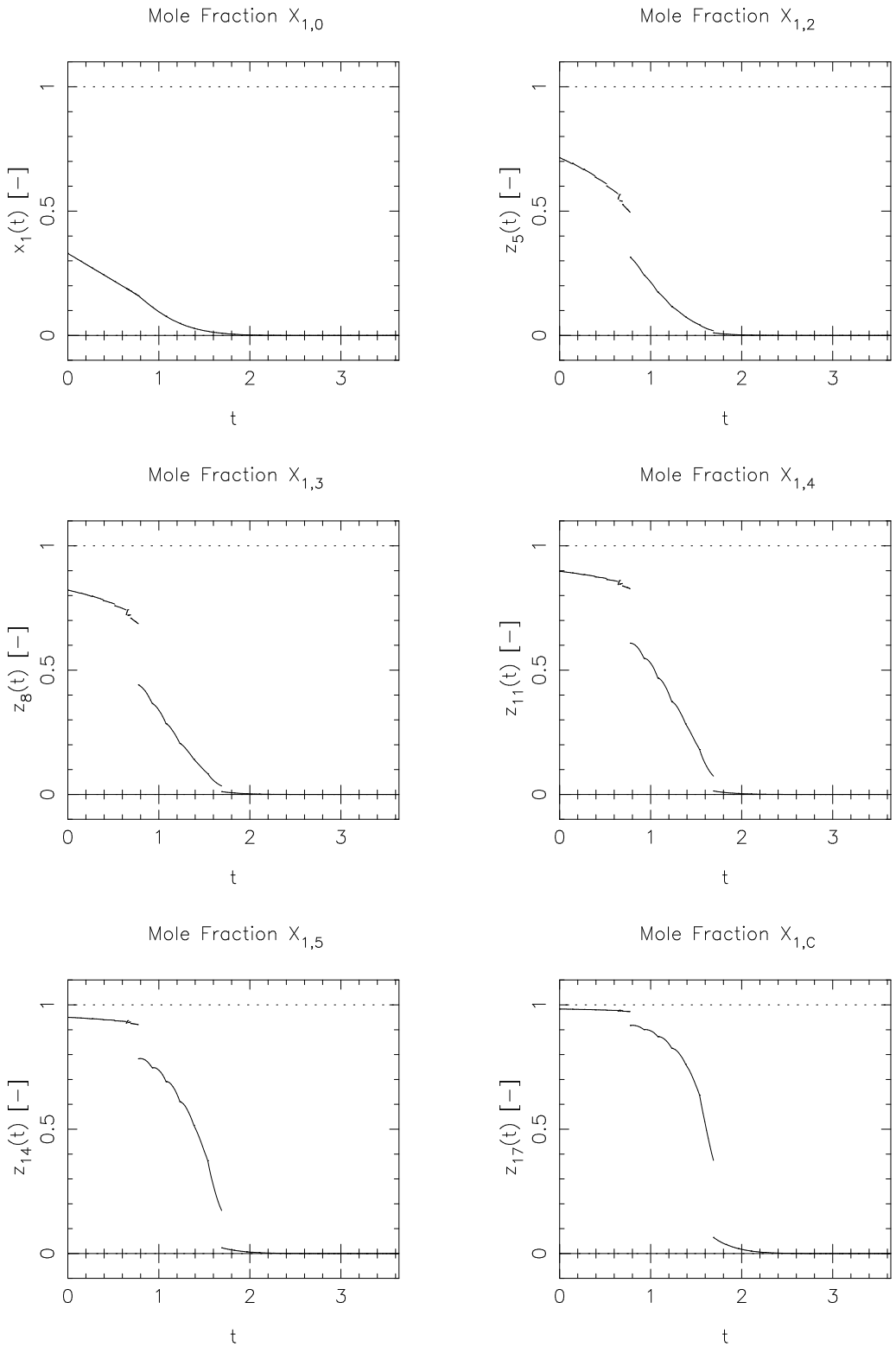

Figure E.4: Mole fraction profile of component 1 for scenario C with flexible recycling.

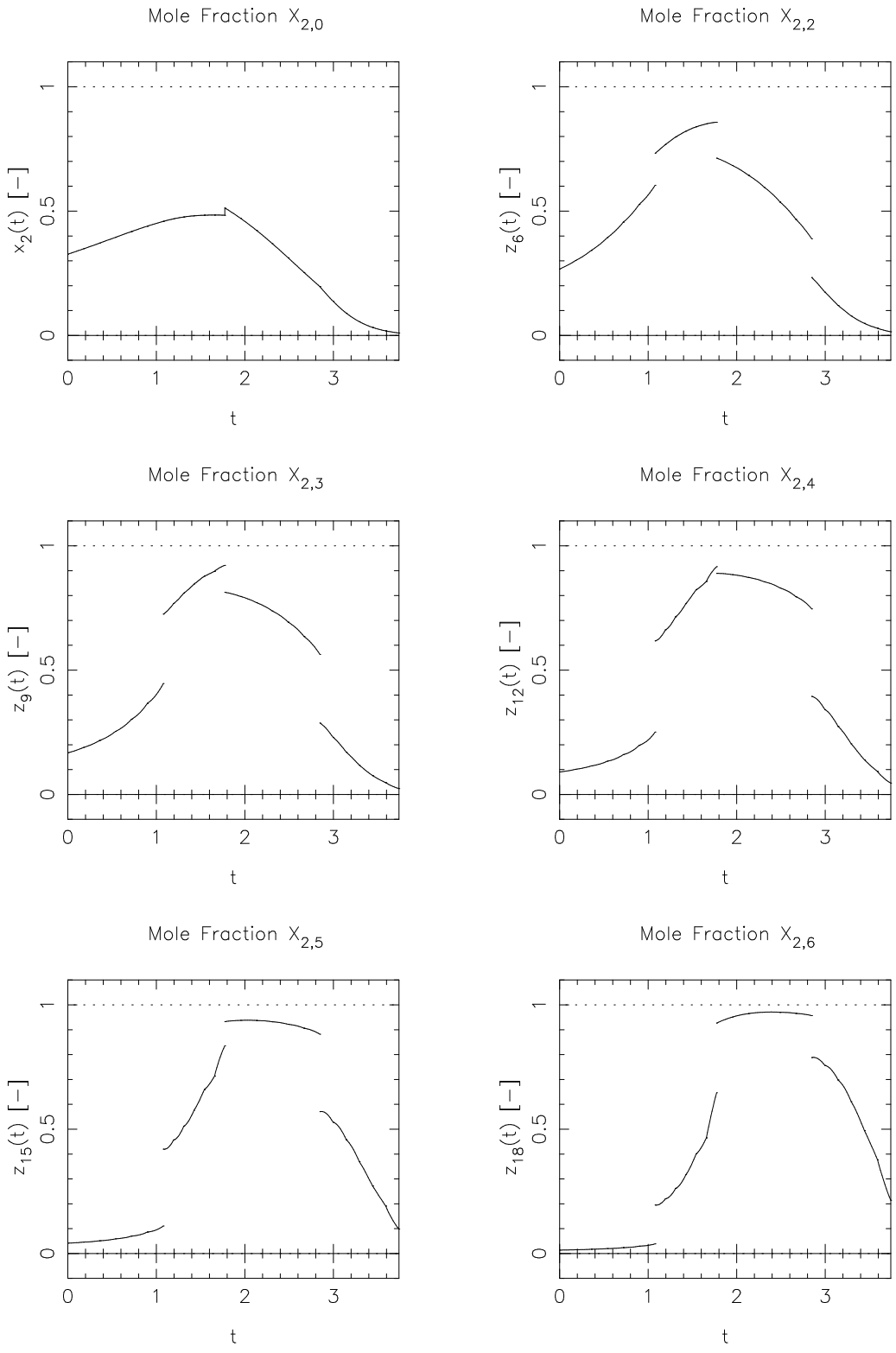

Figure E.5: Mole fraction profile of component 2 for scenario B with batch recycling.

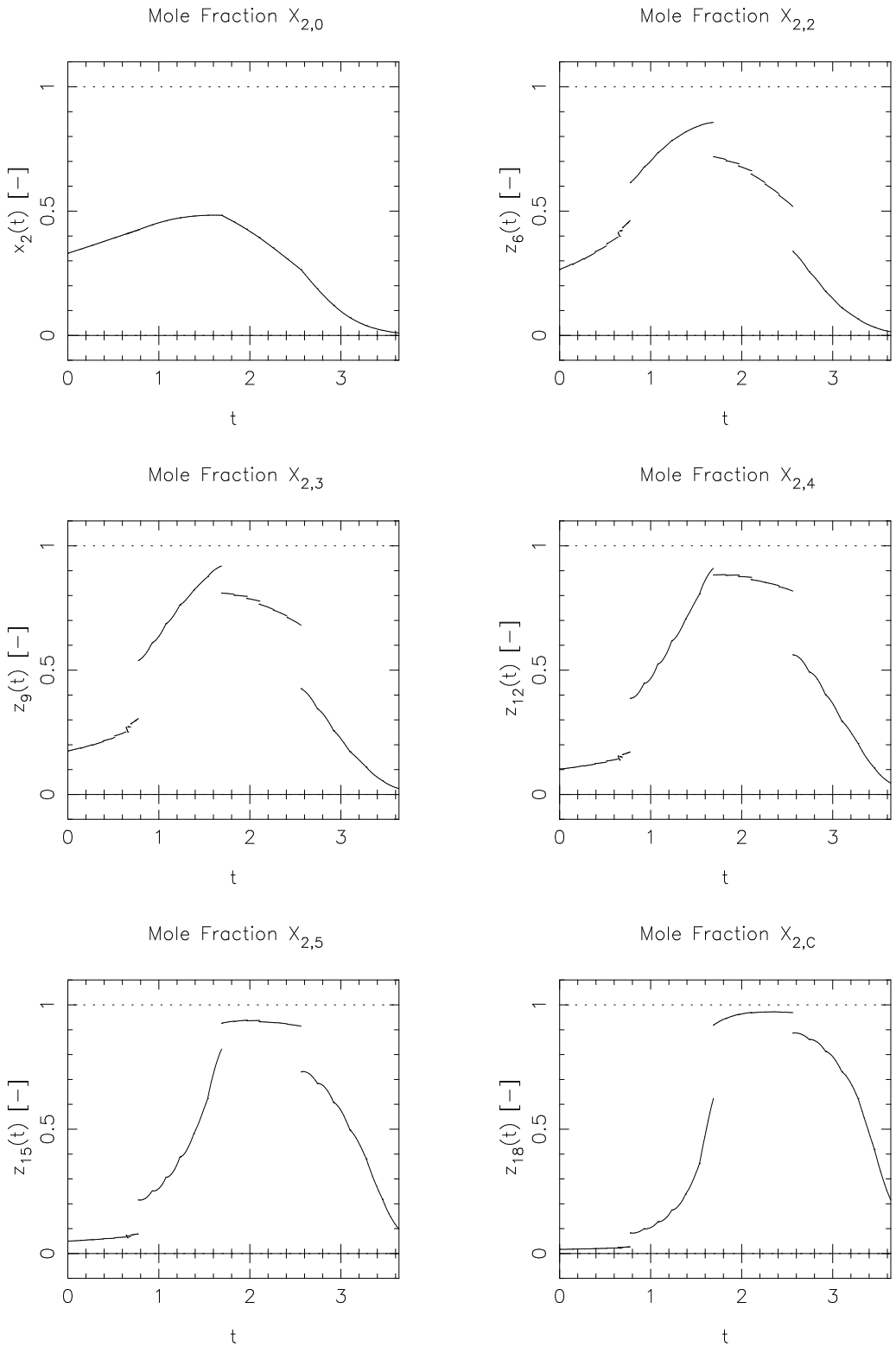

Figure E.6: Mole fraction profile of component 2 for scenario C with flexible recycling.

List of Figures

Notation

Throughout the thesis bold letters are used for vectors and matrices.

Mathematical symbols and abbreviations

Here $\boldsymbol{A} \in \mathbb{R}^{n \times m}$ denotes an arbitrary matrix, $\boldsymbol{x} \in \mathbb{R}^n$ a vector with subvectors $\boldsymbol{x_1}, \boldsymbol{x_2}$ and $\boldsymbol{f}(\boldsymbol{x}, \dots)$ a function depending on $\boldsymbol{x}$. The non–bold $x \in \mathbb{R}$ is a real value and $i \in \mathbb{N}$ an integer.

$\boldsymbol{A}_{i\cdot}$	ith row of matrix $\boldsymbol{A}$, row vector
$\boldsymbol{A}_{\cdot i}$	ith column of matrix $\boldsymbol{A}$, vector
$\boldsymbol{A}^T$	Transpose of matrix $\boldsymbol{A}$
$\boldsymbol{A}^{-1}$	Inverse of matrix $\boldsymbol{A}$
$\boldsymbol{A}^{-T}$	Transpose of the inverse of matrix $\boldsymbol{A}$
f_i	i-th entry of vector $\boldsymbol{f}$
$\boldsymbol{f_x}$	Partial derivative of $\boldsymbol{f}$ with respect to $\boldsymbol{x}$, $\boldsymbol{f_x} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}$
id	Identity map
$\mathbb{N}$	Set of natural numbers
$\mathbb{N}_0$	Set of natural numbers, including zero
$\mathbb{R}$	Set of real numbers
x_i	i-th entry of vector $\boldsymbol{x}$
$\boldsymbol{x_0}$	Start value $\boldsymbol{x_0} \in \mathbb{R}^{n_x}$ of initial value problem
$\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2})$	Meant as one vector where $\boldsymbol{x_1}$ fills the first n_{x_1} entries and $\boldsymbol{x_2}$ the entries n_{x_1+1} to $n_{x_1+x_2}$
$\lceil x \rceil$	x is rounded up
$\lfloor x \rfloor$	x is rounded down
$\lvert\boldsymbol{x}\rvert$	Euclidean norm of $\boldsymbol{x}$, $\lvert\boldsymbol{x}\rvert = \lvert\lvert\boldsymbol{x}\rvert\rvert_2$
$\boldsymbol{x}(\cdot)$	Function $\boldsymbol{x} : [t_0, t_f] \mapsto \mathbb{R}^{n_x}$
$\dot{\boldsymbol{x}}(\cdot)$	Time derivative of $\boldsymbol{x}(\cdot)$, $\dot{\boldsymbol{x}} = \frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} : [t_0, t_f] \mapsto \mathbb{R}^{n_x}$
$\mathbb{Z}$	Set of integer numbers
Δ	Small increment or difference
Δ^j	Operator, representing the time differentiation of a function along the trajectories of the dynamic system, see section 2.1.2
∇	Gradient
$\cap$	Intersection (of two sets)
$\cup$	Union (of two sets)

Roman symbols

$\boldsymbol{a}$	Time dependent vector
a_{ij}	Coefficients
$\boldsymbol{A^1}, \boldsymbol{A^2}, \boldsymbol{A^3}$	Time–dependent matrices
$\tilde{\boldsymbol{b}}$	Vector of generic binary variables
$\boldsymbol{B}$	Left hand side matrix of dynamic system, page 11
B^k	Index set $\left\{i \mid y_i^k = 1\right\}$
$\boldsymbol{c}$	Path and control constraints
$\boldsymbol{C^0}$	Part of constraint function that is independent of $\boldsymbol{u}$
$\boldsymbol{C^U}$	Part of constraint function that is multiplied by $\boldsymbol{u}$
E	Mayer term $E(\boldsymbol{x}(t_f), \boldsymbol{z}(t_f), \boldsymbol{p})$ of the objective functional
E_i	Subset of time horizon, $E_i \subseteq [t_0, t_f]$
$E_{2,i}, E_{3,i}$	Subsets of E_2 resp. E_3, yielding a partition
$\boldsymbol{f}$	Right hand side function of the differential equations, maps state variables, control functions, parameters and time to $\mathbb{R}^{n_x}$
$\tilde{\boldsymbol{f}}$	Abbreviation for right hand side function with some arguments fixed
$\boldsymbol{f^i}$	Short for $\boldsymbol{f}(\boldsymbol{x}^*, \boldsymbol{w^i}, \boldsymbol{u}^*, \boldsymbol{p}^*)$
$\boldsymbol{f^{\mathrm{impl}}}$	Fully implicit form of the right hand side function of the differential equations
F	Generic function, in particular objective function
F^{FB}	Fischer–Burmeister function
$\boldsymbol{F^0}$	Part of right hand side function that is independent of $\boldsymbol{u}$
$\boldsymbol{F^U}$	Part of right hand side function that is multiplied by $\boldsymbol{u}$
$\boldsymbol{g}$	Right hand side function of the algebraic equations, maps state variables, control functions, parameters and time to $\mathbb{R}^{n_z}$
$\boldsymbol{G}$	Equality constraint function for finite–dimensional optimization problem
$\tilde{\boldsymbol{G}}$	Equality constraints in collocation
h	Stage length $t_f - t_0$
$\boldsymbol{h}$	Vector of stage lengths $\tilde{t}_{i+1} - \tilde{t}_i$ in multistage formulation
$\boldsymbol{H}$	Inequality constraint function for finite–dimensional optimization problem
$\mathbf{H^k}$	Approximation of or exact Hessian
i	Index, usually entry of vector or matrix or an interval
$\boldsymbol{I}$	Unity matrix, dimension given by context
$j, j(t)$	Index
J	Optimal cost–to–go function
k, k_i	As subscript: index of model stage
L	Integrand in the Lagrange term of the objective functional
N^k	Index set $\left\{i \mid y_i^k = 0\right\}$
n	See *dimensions*
$\boldsymbol{p}$	Continuous parameters
$P^k = P(\boldsymbol{\beta^k})$	Optimal control problem with additional penalty terms

$\boldsymbol{q}$	Vector of control parameters
r	Integer denoting rank or degree of singularity
$\boldsymbol{r}^{\mathbf{eq}}$	Interior point equality constraints
$\boldsymbol{r}^{\mathbf{ieq}}$	Interior point inequality constraints
$\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}$	Node value for the differential variable $\boldsymbol{x}$ on interval i
$\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}$	Node value for the algebraic variable $\boldsymbol{z}$ on interval i
$\boldsymbol{s}^{\boldsymbol{y}}, \boldsymbol{s}$	Vector of all node values
$\boldsymbol{S}$	Switching conditions
t	Time
t_0, t_f	Start and end time of optimal control problem
$t_{\text{entry}}, t_{\text{exit}}$	Start and end time of an arc
T	(Free) end time
$\bar{t}$	Specific time, $\bar{t} \in [t_0, t_f]$
$\tilde{t}$	Specific time, $\tilde{t} \in [t_0, t_f]$
$\tilde{\boldsymbol{t}}$	Vector of stage transition times, see page 17.
t_i	Begin of i-th multiple shooting interval, $1 \leq i \leq n_{\text{ms}}$.
$\boldsymbol{tr}_{\boldsymbol{k}}$	Stage transition function
$\boldsymbol{u}$	Continuous control functions
$\boldsymbol{u}^{\min}, \boldsymbol{u}^{\max}$	Lower and upper bounds on $\boldsymbol{u}$
$\hat{\boldsymbol{u}}$	Piecewise approximation of the controls
$\boldsymbol{v}$	Binary parameters
$\hat{v}, \tilde{v}$	Generic integer variables
$\boldsymbol{w}$	Binary control functions
$\tilde{\boldsymbol{w}}$	Control functions in convexified formulation
$\bar{\boldsymbol{w}}$	Binary control functions in convexified formulation
$\boldsymbol{w}^{\boldsymbol{i}}$	i–th vertex of the cube $[0,1]^{n_w}$
$W^{\boldsymbol{x}}_{\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}}$	Wronskian, defined as $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}}(t; \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}, \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}, \boldsymbol{q}, \boldsymbol{p})$
$W^{\boldsymbol{x}}_{\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}}$	Wronskian, defined as $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}}(t; \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}, \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}, \boldsymbol{q}, \boldsymbol{p})$
$W^{\boldsymbol{x}}_{\boldsymbol{q}}$	Wronskian, defined as $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{q}}(t; \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}, \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}, \boldsymbol{q}, \boldsymbol{p})$
$W^{\boldsymbol{x}}_{\boldsymbol{p}}$	Wronskian, defined as $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{p}}(t; \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}, \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}, \boldsymbol{q}, \boldsymbol{p})$
$W^{\boldsymbol{x}}$	$W^{\boldsymbol{x}} = (W^{\boldsymbol{x}}_{\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{x}}}, W^{\boldsymbol{x}}_{\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{z}}}, W^{\boldsymbol{x}}_{\boldsymbol{q}}, W^{\boldsymbol{x}}_{\boldsymbol{p}})^T$
$W^{\boldsymbol{z}}_{\bullet}$	Wronskian, defined as above for $\boldsymbol{z}(t)$
$\boldsymbol{x}$	Differential state variables $\boldsymbol{x} : [t_0, t_f] \mapsto \mathbb{R}^{n_x}$. In chapter 3: continuous variables
$\bar{\boldsymbol{x}}$	Differential state variables corresponding to control $\bar{\boldsymbol{w}}(\cdot)$
$\boldsymbol{X}$	Covex, compact set
$\boldsymbol{y}$	State variables $\boldsymbol{y} : [t_0, t_f] \mapsto \mathbb{R}^{n_y}$ consisting of a differential and an algebraic part, $\boldsymbol{y} = (\boldsymbol{x}, \boldsymbol{z})$. In chapter 3: continuous variables
$\boldsymbol{Y}$	Polyhedral set of integer points, e.g., $\boldsymbol{Y} = \{0,1\}^{n_y}$
$\boldsymbol{z}$	Algebraic state variables $\boldsymbol{z} : [t_0, t_f] \mapsto \mathbb{R}^{n_z}$
$\mathbf{0}$	Null matrix, dimension given by context

Greek symbols

α	Step length. In chapter 3: objective function value
α_i	Real coefficients
$\boldsymbol{\beta}$	Penalty parameter vector
β_{init}	Penalty initialization
β_{inc}	Penalty increment
$\delta, \varepsilon, \boldsymbol{\varepsilon_c}, \boldsymbol{\varepsilon_r}$	Tolerances
Φ	Objective functional, sum of Bolza and Lagrange term
Φ^{BN}	Objective function value of solution to problem (BN), original
Φ^{BL}	Objective function value of solution to problem (BL), convexified
Φ^{RN}	Objective function value of solution to problem (RN), relaxed original
Φ^{RL}	Objective function value of solution to problem (RL), relaxed convexified
$\boldsymbol{\varphi}$	Control parameter vector
$\boldsymbol{\zeta}, \boldsymbol{\zeta_2}, \boldsymbol{\zeta_3}$	Control functions
$\boldsymbol{\lambda}, \tilde{\boldsymbol{\lambda}}$	Lagrange multipliers of optimal control or time–independent optimization problem
$\boldsymbol{\mu}, \tilde{\boldsymbol{\mu}}$	Lagrange multipliers of inequality constraints
$\boldsymbol{\nu}$	Lagrange multipliers of end–point constraints
ψ	End–point Lagrangian
$\Omega(\Psi)$	Set of binary functions fullfilling conditions on the switching times
$\bar{\Omega}(\Psi)$	Set of relaxed binary functions fullfilling conditions on the switching times
Ψ	Set of time points when a discontinuity in the binary control function vector $\boldsymbol{w}(t)$ may occur, either $\Psi = \Psi_\tau$ or $\Psi = \Psi_{\text{free}}$
Ψ_τ	Finite set of possible switching times, $\Psi_\tau = \{\tau_1, \tau_2, \ldots, \tau_{n_\tau}\}$
Ψ_{free}	Whole time interval under consideration, $\Psi_{\text{free}} = [t_0, t_f]$
Ψ_{MIN}	Minimum distance between two switching times
$\boldsymbol{\sigma}$	Switching function
$\boldsymbol{\sigma^i}$	i–th vertex of the cube $[0,1]^{n_{\bar{w}}}$
τ	Time
$\boldsymbol{\tau}$	Vector of possible switching times τ_i, $1 \leq i \leq n_\tau$
$\boldsymbol{\xi}$	(Continuous) Optimization variable of the (MI)NLP
$\boldsymbol{\omega}$	Integer optimization variable of the MINLP

Gothic and other symbols

$\mathcal{A}(\boldsymbol{\xi})$	Active set at point $\boldsymbol{\xi}$
$\mathfrak{C}(T)$	Controllable set at time T
$\mathfrak{C}_{BB}(T)$	Controllable set with bang–bang functions at time T
$\mathfrak{C}$	Controllable set
$\mathfrak{C}_{BB}$	Controllable set with bang–bang functions at
$\mathcal{G}$	Control discretization grid
$\mathcal{H}$	Hamiltonian
$\mathcal{K}$	Subset $\mathcal{K} \subseteq \mathcal{X}$
$\mathcal{L}$	Lagrangian
$\mathcal{R}$	Admissible region
$\partial\mathcal{R}$	Boundary of the admissible region
int $(\mathcal{R})$	Interior of the admissible region
$\mathcal{S}$	Short form to describe a trajectory. See page 95.
$\mathcal{T}$	Trajectory $(\boldsymbol{x}(\cdot), \boldsymbol{z}(\cdot), \boldsymbol{u}(\cdot), \boldsymbol{w}(\cdot), \boldsymbol{p}, \boldsymbol{v})$
$\mathcal{U}_m$	Set $\{\boldsymbol{u} : [t_0, t_f] \mapsto \mathbb{R}^{n_u}, \boldsymbol{u}(t)$ measurable$\}$
$\mathcal{U}_{BB}$	Set of measurable bang–bang functions
$\mathcal{X}$	Real linear space if not specified differently
$*$	Variable belonging to an optimal solution

Dimensions

n_c	Number of path constraints $c_i(\cdot)$
n_{control}	Number of control discretization intervals
n_{col}	Number of discretization points in collocation
n_{ext}	Number of grid refinement steps (for extrapolation)
n_G	Number of equality constraints
n_H	Number of inequality constraints
n_p	Number of continuous parameters
n_u	Number of continuous control functions
n_v	Number of binary parameters
n_w	Number of binary control functions
$n_{\tilde{w}}$	Number of binary control functions in the convexified problem
n_x	Number of differential variables
n_y	Number of state variables, $n_y = n_x + n_z$
n_z	Number of algebraic variables
n_{ms}	Number of multiple shooting intervals
n_{mos}	Number of model stages
$n_{r^{\text{eq}}}$	Number of interior point equalities $r_i^{\text{eq}}(\cdot)$
$n_{r^{\text{ieq}}}$	Number of interior point inequalities $r_i^{\text{ieq}}(\cdot)$
n_ξ	Number of variables after parameterization and discretization of the optimal control problem to a NLP
n_τ	Number of possible switching times

Applications

Fishing problem

c_0, c_1	Parameters for the quota of caught fish
w	Fishing control
x_0	Biomass of prey species
x_1	Biomass of predator species

Rocket car

w	Acceleration / Deceleration
x_0	Covered distance
x_1	Velocity

F–8 aircraft

w	Tail deflection angle
x_0	Angle of attack
x_1	Pitch angle
x_2	Pitch rate

Subway

w	Operation mode: 1 series, 2 parallel, 3 coasting, 4 braking
x_0	Position of the train
x_1	Velocity of the train
u	Braking deceleration, fixed to $u_{\max}$

All parameters of the model are given and explained in appendix C.

Calcium oscillation

p_1	Costs of inhibitor u_1
p_2	Costs of inhibitor u_2
u_1	Uncompetitive inhibitor of the PMCA ion pump
u_2	Inhibitor of PLC activation by the G–protein
w_1, w_2	0-1 decision, when to inhibit with u_1 resp. u_2
x_0	Concentration of activated G-protein
x_1	Concentration of active phospholipase C
x_2	Concentration of intracellular calcium
x_3	Concentration of intra–ER calcium
k_i, K_i	Reaction coefficients

Batch distillation with waste cut recycling

A_k, B_k, C_k	Antoine coefficients given in the appendix
$\boldsymbol{c}$	Parameters of the objective function
D	Distillate flow
L	Liquid flow
L_ℓ	Liquid flow entering tray ℓ
M	Molar reboiler content, $M = x_0$
N	Number of trays, here $N = 5$
n_{comp}	Number of components, here $n_{\text{comp}} = 3$
P_1, P_2, P_3	Product outputs of the three components or short for production cuts 1 and 2
P	Overall profit
p_2	Recycling ratio (in transition stages) of first slop cut
p_3	Recycling ratio (in transition stages) of second slop cut
p_4, p_5, p_6, p_7	Concentration of component 1 and 2 in slop cut reservoirs S1 resp. S2
pr_0, pr_1	Local parameters needed for the calculation of purities
R	Reflux ratio
R_1	Recycling ratio (in transition stages) of first slop cut
R_2	Recycling ratio (in transition stages) of second slop cut
s_i	Parameters of the objective function
S_1, S_2	Slop cut stages 1 and 2, also slop cut reservoirs (tanks)
S_1', S_1'	Amounts of slop cut material that are not recycled
$\tilde{t}_i$	Time when a model stage change occurs
T	Terminal time of the process
T_ℓ	Temperature on tray ℓ
u_0	Reflux ratio
u_i	$i = 1 \ldots N+1$: Flux from reservoir S1 to tray $i-1$
u_i	$i = N+2 \ldots 2N+2$: Flux from reservoir S2 to tray $i-(N+2)$
$\hat{u}_1$	Flux from reservoir S1
$\hat{u}_2$	Flux from reservoir S2
V	Vapor flow
w_i	$i = 1 \ldots N+1$: Is flux from reservoir S1 directed to tray $i-1$?
w_i	$i = N+2 \ldots 2N+2$: Is flux from S2 directed to tray $i-(N+2)$?
x_0	Molar reboiler content
x_1	Mole fraction $X_{1,0}$
x_2	Mole fraction $X_{2,0}$
x_3	Content of slop cut reservoir S1
x_4	Content of slop cut reservoir S2
$X_{k,i}$	Concentration of component $k \in \{1,2,3\}$ on tray $i \in \{0,1,\ldots,6\}$ with tray $0 \approx$ reboiler and tray $6 \approx$ condenser
$X_{P_1}, X_{P_2}, X_{P_3}$	Prespecified purity requirements
ρ	Total pressure
$\rho_k^s(T)$	Partial pressures of the undiluted components

Abbreviations

BVP	Boundary value problem
DAE	Differential–algebraic equation
ECP	Extended cutting planes
END	External numerical differentiation
GBD	Generalized Benders decomposition
HJB	Hamilton–Jacobi–Bellman
IND	Internal numerical differentiation
IVP	Initial value problem
KKT	Karush–Kuhn–Tucker
LICQ	Linear independence constraint qualification
LP	Linear program
MIDO	Mixed–integer dynamic optimization
MILP	Mixed–integer linear program
MINLP	Mixed–integer nonlinear program
MIOC	Mixed–integer optimal control
MIOCP	Mixed–integer optimal control problem
MIQP	Mixed–integer quadratic program
MLDO	Mixed–logic dynamic optimization
MPEC	Mathematical program with equlibrium constraints
MS MINTOC	Multiple shooting based mixed–integer optimal control algorithm
MSMIOCP	Multistage mixed–integer optimal control problem
MSOCP	Multistage optimal control problem
NLP	Nonlinear program
OA	Outer approximation
OCP	Continuous optimal control problem without integer constraints
ODE	Ordinary differential equation
PDAE	Partial differential–algebraic equation
QP	Quadratic program
SOS	Special ordered set property
SQP	Sequential quadratic programming
s.t.	subject to

Danksagung

Was du mit Geld nicht bezahlen kannst, bezahle wenigstens mit Dank.

Deutsches Sprichwort

Dankbarkeit, man spürt sie ja so selten bei den Menschen, und gerade die Dankbarsten finden nicht den Ausdruck dafür, sie schweigen verwirrt, sie schämen sich und tun manchmal stockig, um ihr Gefühl zu verbergen.

Stefan Zweig (1881-1942)

Der Fahrgast, als der Taxichauffeur das hohe Trinkgeld wortlos wegsteckt: "Sagt man eigentlich in Berlin nicht 'Danke'?" - "Det ist untaschiedlich", erwidert der Taxifahrer, "manche saren et, manche saren et nich."

unbekannt

Dankbarkeit ist im menschlichen Leben selten gesät und in der Politik schon gar nicht. Ich habe eigentlich in meiner eigenen Partei den größten Ärger mit denen gehabt, denen ich am meisten geholfen habe.

Helmut Kohl, 1972

Ich danke allen, die mich bei der Erstellung dieser Arbeit in irgendeiner Art und Weise unterstützt haben, insbesondere aber Jan Albersmeyer, John Barleycorn, Cordula Becker, Georg Bock, Ulli Brandt-NocheinName, Achim Brasch, Achim Dahlbokum, Jens Derbinski, der DFG, Moritz Diehl, Rolf und Sabine Gertjejanßen, Benni Gill, Iwo Hahn, Mark Hörter, Renate Kohlhaus, Stefan Körkel, Olaf Kroon, Peter Kühl, Julian Langer, Dirk Lebiedz, Helmut Maurer, Lars Migge, Katja Mombaur, Herbert Niemeier, Marcus Oswald, Andreas Potschka, Cesar de Prada, Jens Prinke, Ulrich Quednau, Gerhard Reinelt, Gisela Renken, Peter Riede, Margret Rothfuß, Daniel Sager, Walter Sager, Andreas Schäfer, Johannes Schlöder, Hanna Seitz.

Sie wissen sicherlich wofür ich ihnen dankbar bin, ganz bestimmt aber weiß ich es. Und ich werde versuchen, dieses nicht mit dem "*größten Ärger*" zurück zu zahlen...

Herausheben möchte ich aus den genannten Personen Georg Bock, der mich überhaupt erst überzeugt hat zu promovieren und ohne dessen Ideen diese Arbeit nicht zustande gekommen wäre. Ich erwähne dies ausdrücklich, denn

Dem, der einen in der Nacht führt, muss man am Tage Dank sagen.

Afrikanisches Sprichwort

Bibliography

Akrotirianakis, I., Maros, I., & Rustem, B. 2001. An outer approximation based branch and cut algorithm for convex 0-1 MINLP problems. *Optimization methods and software*, **16**, 21–47. Optimization Methods and Software.

Alamir, M., & Attia, S. A. 2004. On solving optimal control problems for switched hybrid nonlinear systems by strong variations algorithms. *In: 6th IFAC symposium, NOLCOS, Stuttgart, Germany, 2004.*

Allgor, R.J. 1997. *Modeling and computational issues in the development of batch processes.* Ph.D. thesis, M.I.T.

Antsaklis, P., & Koutsoukos, X. 1998. *On hybrid control of complex systems: A survey.* In 3rd International Conference ADMP'98, Automation of Mixed Processes: Dynamic Hybrid Systems, pages 1–8, Reims, France, March 1998.

Aristotle. 350 B.C.. *Physics.* Internet classics archive, http://classics.mit.edu/Aristotle/physics.html.

Ascher, U.M., & Petzold, L.R. 1998. *Computer methods for ordinary differential equations and differential–algebraic equations.* Philadelphia: SIAM.

Attia, S.A., Alamir, M., & Canudas de Wit, C. 2005. Sub optimal control of switched nonlinear systems under location and switching constraints. *In: IFAC World congress.*

Aumann, R.J. 1965. Integrals of set–valued functions. *Journal of mathematical analysis and applications*, **12**, 1–12.

Balas, E., & Martin, R. 1980. Pivot and complement: a heuristic for 0-1 programming. *Management Science*, **26**, 86–96.

Balas, E., & Perregaard, M. 1999. *Lift and project for mixed 0-1 programming: Recent progress.* Tech. rept. MSRR No. 627, Graduate School of Industrial Administration, Carnegie Mellon University.

Balas, E., Ceria, S., & Cornuejols, G. 1993. A lift-and-project cutting plane algorithm for mixed 0-1 programs. *Mathematical Programming*, **58**, 295–324.

Balas, E., Ceria, S., Dawande, M., Margot, F., & Pataki, G. 2001. OCTANE: A new heuristic for pure 0–1 programs. *Operations Research*, **49(2)**, 207–225.

Bär, V. 1984. *Ein Kollokationsverfahren zur numerischen Lösung allgemeiner Mehrpunktrandwertaufgaben mit Schalt– und Sprungbedingungen mit Anwendungen in der optimalen Steuerung und der Parameteridentifizierung.* M.Sci. thesis, Universität Bonn.

Barton, P.I., & Lee, C.K. 2002. Modeling, simulation, sensitivity analysis and optimization of hybrid systems. *ACM transactions on modeling and computer simulation*, **12**(4), 256–289.

Barton, P.I., & Lee, C.K. 2004. Design of process operations using hybrid dynamic optimization. *Computers and chemical engineering*, **28**(6–7), 955–969.

Bauer, I. 1999. *Numerische Verfahren zur Lösung von Anfangswertaufgaben und zur Generierung von ersten und zweiten Ableitungen mit Anwendungen bei Optimierungsaufgaben in Chemie und Verfahrenstechnik.* Ph.D. thesis, Universität Heidelberg.

Bellman, R.E. 1957. *Dynamic programming.* Princeton: University Press.

Berridge, M. 1997. The AM and FM of calcium signaling. *Nature*, **386**, 759.

Biegler, L.T. 1984. Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. *Computers and chemical engineering*, **8**, 243–248.

Binder, T., Blank, L., Bock, H.G., Bulirsch, R., Dahmen, W., Diehl, M., Kronseder, T., Marquardt, W., Schlöder, J.P., & Stryk, O.v. 2001. Introduction to model based optimization of chemical processes on moving horizons. *Pages 295–340 of:* Grötschel, M., Krumke, S.O., & Rambau, J. (eds), *Online Optimization of Large Scale systems: State of the Art.* Springer.

Bisswanger, H. 2002. *Enzyme kinetics.* Weinheim: Wiley.

Bixby, R.E., Fenelon, M., Gu, Z., Rothberg, E., & Wunderling, R. 2004. *The sharpest cut: The impact of Manfred Padberg and his work.* SIAM. Chap. 18: Mixed-Integer Programming: A Progress Report.

Bock, H.G. 1978a. Numerical solution of nonlinear multipoint boundary value problems with applications to optimal control. *Zeitschrift für Angewandte Mathematik und Mechanik*, **58**, 407.

Bock, H.G. 1978b. *Numerische Berechnung zustandsbeschränkter optimaler Steuerungen mit der Mehrzielmethode.* Heidelberg: Carl-Cranz-Gesellschaft.

Bock, H.G. 1981. Numerical treatment of inverse problems in chemical reaction kinetics. *Pages 102–125 of:* Ebert, K.H., Deuflhard, P., & Jäger, W. (eds), *Modelling of chemical reaction systems*. Springer Series in Chemical Physics, vol. 18. Heidelberg: Springer.

Bock, H.G. 1983. Recent advances in parameter identification techniques for ODE. *In:* Deuflhard, P., & Hairer, E. (eds), *Numerical treatment of inverse problems in differential and integral equations*. Boston: Birkhäuser.

Bock, H.G. 1987. *Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*. Bonner Mathematische Schriften, vol. 183. Bonn: Universität Bonn.

Bock, H.G., & Longman, R.W. 1982. Computation of optimal controls on disjoint control sets for minimum energy subway operation. *In: Proceedings of the American Astronomical Society. Symposium on engineering science and mechanics*.

Bock, H.G., & Plitt, K.J. 1984. A multiple shooting algorithm for direct solution of optimal control problems. *Pages 243–247 of: Proceedings 9th IFAC world congress Budapest*. Pergamon Press.

Bock, H.G., Eich, E., & Schlöder, J.P. 1988. Numerical solution of constrained least squares boundary value problems in differential-algebraic equations. *In:* Strehmel, K. (ed), *Numerical treatment of differential equations*. Leipzig: Teubner.

Boggs, P.T., & Tolle, J.W. 1995. Sequential quadratic programming. *Acta numerica*, 1–51.

Boggs, P.T., Tolle, J.W., & Wang, P. 1982. On the local convergence of quasi–Newton methods for constrained optimization. *SIAM journal on control and optimization*, **20**(2), 161–171.

Bonami, P., Biegler, L.T., Conn, A.R., Cornuéjols, G., Grossmann, I.E., Laird, C.D., Lee, J., Lodi, A., Margot, F., Sawaya, N., & Wächter, A. 2005. *An algorithmic framework for convex mixed integer nonlinear programs*. Tech. rept. IBM T. J. Watson Research Center.

Borchers, B., & Mitchell, J.E. 1994. An improved Branch and Bound algorithm for Mixed Integer Nonlinear Programming. *Computers and operations research*, **21**(4), 359–367.

Brandt-Pollmann, U. 2004. *Numerical solution of optimal control problems with implicitly defined discontinuities with applications in engineering*. Ph.D. thesis, IWR, Universität Heidelberg.

Brandt-Pollmann, U., Lebiedz, D., Diehl, M., Sager, S., & Schlöder, J.P. 2005. Real-time nonlinear feedback control of pattern formation in (bio)chemical reaction–diffusion processes: A model study. *Chaos*, **15**, 033901. selected for online-publication in Virtual Journal of Biological Physics Research, July 15, 2005.

Brenan, K.E., Campbell, S.L., & Petzold, L.R. 1996. *Numerical solution of initial-value problems in differential-algebraic equations*. Philadelphia: SIAM. Classics in Applied Mathematics 14.

Bryson, A.E., & Ho, Y.-C. 1975. *Applied Optimal Control*. New York: Wiley.

Bulirsch, R. 1971. *Die Mehrzielmethode zur numerischen Lösung von nichtlinearen Randwertproblemen und Aufgaben der optimalen Steuerung*. Tech. rept. Carl-Cranz-Gesellschaft, Oberpfaffenhofen.

Bulirsch, R., Nerz, E., Pesch, H. J., & von Stryk, O. 1991. *Combining direct and indirect methods in nonlinear optimal control: Range maximization of a hang glider*. Tech. rept. 313. Technische Universit Mnchen.

Burgschweiger, J., Gnädig, B., & Steinbach, M.C. 2004. *Optimization models for operative planning in drinking water networks*. Tech. rept. ZR-04-48. ZIB.

Buss, M., Glocker, M., Hardt, M., Stryk, O. v., Bulirsch, R., & Schmidt, G. 2002. *Nonlinear hybrid dynamical systems: Modelling, optimal control, and applications*. Vol. 279. Berlin, Heidelberg: Springer-Verlag.

Bussieck, M.R. 2005. *MINLP World home page*. http://www.gamsworld.org/minlp/.

Bussieck, M.R., & Prüssner, A. 2003. Mixed–integer nonlinear programming. *SIAG/OPT Newsletter: Views and news*, **14**(1), 1.

Caillau, J.-B., Gergaud, J., Haberkorn, T., Martinon, P., & Noailles, J. 2002. Numerical optimal control and orbital transfers. *Pages 39–49 of: Proceedings of the Workshop Optimal Control, Sonderforschungsbreich 255: Transatmosphärische Flugsysteme, Heronymus Munchen, ISBN 3-8979-316-X*.

Christensen, F.M., & Jorgensen, S.B. 1987. Optimal control of binary batch distillation with recycled waste cut. *Chemical engineering journal*, **34**, 57–64.

Chvatal, V. 1973. Edmonds polytopes and weakly Hamiltonian graphs. *Mathematical programming*, **5**, 29–40.

Dakin, R.J. 1965. A tree-search algorithm for mixed integer programming problems. *The computer journal*, **8**, 250–255.

Diehl, M. 2001. *Real-time optimization for large scale nonlinear processes*. Ph.D. thesis, Universität Heidelberg. http://www.ub.uni-heidelberg.de/archiv/1659/.

Diehl, M., Leineweber, D.B., & Schäfer, A.A.S. 2001. *MUSCOD-II Users' Manual*. IWR-Preprint 2001-25. Universität Heidelberg.

Diehl, M., Leineweber, D.B., Schäfer, A.A.S., Bock, H.G., & Schlöder, J.P. 2002. Optimization of multiple-fraction batch distillation with recycled waste cuts. *AIChE journal*, **48**(12), 2869–2874.

Diwekar, U.M. 1995. *Batch distillation: simulation, optimal design, and control*. Washington: Taylor & Francis.

Diwekar, U.M., Malik, R.K., & Madhavan, K.P. 1987. Optimal reflux rate policy determination for multicomponent batch distillation columns. *Computers and chemical engineering*, **11**, 629.

Domenech, S., & Enjalbert, M. 1981. Program for simulation batch rectification as unit operation. *Computers and chemical engineering*, **5**(181).

Dür, M., & Stix, V. 2005. Probabilistic subproblem selection in branch-and-bound algorithms. *Journal of computational and applied mathematics*, **182**(1), 67–80.

Duran, M., & Grossmann, I.E. 1986. An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical programming*, **36**(3), 307–339.

Ermentrout, B. 2002. *Simulating, analyzing, and animating dynamical systems: A guide to XPPAUT for researchers and students*. Philadelphia: SIAM.

Esposito, W.R., & Floudas, C.A. 2000. Deterministic global optimization in optimal control problems. *Journal of global optimization*, **17**, 97–126.

Falk, J.E., & Soland, R.M. 1969. An algorithm for separable nonconvex programming problems. *Management science*, **15**, 550–569.

Farhat, S., Czernicki, M., Pibouleau, L., & Domenech, S. 1990. Optimization of multiple-fraction batch distillation by nonlinear programming. *AIChE journal*, **36**, 1349–1360.

Fletcher, R. 1987. *Practical methods of optimization*. 2 edn. Chichester: Wiley.

Fletcher, R., & Leyffer, S. 1994. Solving mixed integer nonlinear programs by outer approximation. *Mathematical programming*, **66**, 327–349.

Floudas, C.A., Akrotirianakis, I.G., Caratzoulas, S., Meyer, C.A., & Kallrath, J. 2005. Global optimization in the 21st century: Advances and challenges. *Computers and chemical engineering*, **29**(6), 1185–1202.

Franke, R., Meyer, M., & Terwiesch, P. 2002. Optimal control of the driving of trains. *Automatisierungstechnik*, **50**(12), 606–614.

Fuller, A.T. 1963. Study of an optimum nonlinear control system. *Journal of electronics and control*, **15**, 63–71.

Gallitzendörfer, J.V., & Bock, H.G. 1994. Parallel algorithms for optimization boundary value problems in DAE. *In:* Langendörfer, H. (ed), *Praxisorientierte parallelverarbeitung*. Hanser, München.

Garey, M.R., & Johnson, D.S. 1979. *Computers and intractability: A guide to the theory of NP-Completeness*. New York: W.H. Freeman.

Garrard, W.L., & Jordan, J.M. 1977. Design of nonlinear automatic control systems. *Automatica*, **13**, 497–505.

Gear, C.W., & Vu, T. 1983. Smooth numerical solutions of ordinary differential equations. *In:* Deuflhard, P., & Hairer, E. (eds), *Numerical treatment of inverse problems in differential and integral equations*. Boston: Birkhäuser.

Geoffrion, A.M. 1972. Generalized Benders decomposition. *Journal of optimization theory and applications*, **10**, 237–260.

Goldbeter, A. 1996. *Biochemical oscillations and cellular rhythms: The molecular bases of periodic and chaotic behaviour*. Cambridge University Press.

Gomory, R. 1958. Outline of an algorithm for integer solutions to linar programs. *Bulletin of the american mathematical society*, **64**, 275–278.

Grossmann, I.E. 2002. Review of nonlinear mixed-integer and disjunctive programming techniques. *Optimization and engineering*, **3**, 227–252.

Grossmann, I.E., & Kravanja, Z. 1997. Mixed-integer nonlinear programming: A survey of algorithms and applications. *In:* Biegler et al. (ed), *Large-scale optimization with applications. Part II: Optimal design and control*. The IMA Volumes in Mathematics and its Applications, vol. 93. Springer Verlag.

Grossmann, I.E., Aguirre, P.A., & Barttfeld, M. 2005. Optimal synthesis of complex distillation columns using rigorous models. *Computers and chemical engineering*, **29**, 1203–1215.

Grötschel, M., Lovász, L., & Schrijver, A. 1988. *Geometric algorithms and combinatorial optimization*. Algorithms and Combinatorics, vol. 2. Springer.

Gupta, O.K., & Ravindran, A. 1985. Branch and Bound experiments in convex nonlinear integer programming. *Management science*, **31**, 1533–1546.

Hall, L.A., Schulz, A.S., Shmoys, D.B., & Wein, J. 1997. Scheduling to minimize average completion time: Off-line and on-line approximation algorithms. *Mathematics of operations research*, **22**, 513–544.

Hartl, R.F., Sethi, S.P., & Vickson, R.G. 1995. A survey of the maximum principles for optimal control problems with state constraints. *SIAM review*, **37**, 181–218.

Hermes, H., & Lasalle, J.P. 1969. *Functional analysis and time optimal control*. Mathematics in science and engineering, vol. 56. New York and London: Academic Press.

Hoai, T.V., Reinelt, G., & Bock, H.G. 2005. Boxsteps methods for crew pairing problems. *Optimization and engineering*, (to appear).

Huber, J. 1998. *Large–scale SQP–methods and cross–over–techniques for Quadratic Programming*. M.Sci. thesis, Universität Heidelberg.

Iyengar, G. 2001. *Quadratic cuts for mixed 0-1 quadratic programs*. Tech. rept. IEOR Dept., Columbia University.

Johansson, K.H., Barabanov, A.E., & Aström, K.J. 2002. Limit cycles with chattering in relay feedback systems. *IEEE transactions on automatic control*, **47**(9), 1414–1423.

Johansson, K.H., Lygeros, J., & Sastry, S. 2004. *Encyclopedia of life support systems (EOLSS)*. UNESCO. Chap. Modelling of Hybrid Systems, page 6.43.28.1.

Johnson, E.L., Nemhauser, G.L., & Savelsbergh, M.W.P. 2000. Progress in linear programming-based algorithms for integer programming: An exposition. *Informs journal on computing*, **12**(1), 2–23.

Jünger, M., & Reinelt, G. 2004. Combinatorial optimization and integer programming. *Pages 321–327 of:* Derigs, U. (ed), *Encyclopedia of life support systems EOLSS, 6.5: Optimization and operations research*. EOLSS Publishers.

Kallrath, J. 2002. *Gemischt–ganzzahlige Optimierung: Modellierung in der Praxis*. Wiesbaden: Vieweg.

Karush, W. 1939. *Minima of functions of several variables with inequalities as side conditions*. M.Sci. thesis, Department of Mathematics, University of Chicago.

Kaya, C.Y., & Noakes, J.L. 1996. Computations and time-optimal controls. *Optimal control applications and methods*, **17**, 171–185.

Kaya, C.Y., & Noakes, J.L. 2003. A computational method for time-optimal control. *Journal of optimization theory and applications*, **117**, 69–92.

Kelley, J.E. 1960. The cutting-plane method for solving convex programs. *Journal of SIAM*, **8**, 703–712.

Kiss, I.Z., & Hudson, J.L. 2003. Chemical complexity: Spontaneous and engineered structures. *AIChE journal*, **49**(9), 2234 – 2241.

Körkel, S. 1995. *Der Lift & Project-Schnittebenenalgorithmus für gemischt-ganzzahlige 0/1-Optimierungsaufgaben*. M.Sci. thesis, University of Heidelberg.

Körkel, S. 2002. *Numerische Methoden für optimale Versuchsplanungsprobleme bei nichtlinearen DAE-Modellen*. Ph.D. thesis, Universität Heidelberg, Heidelberg.

Krämer-Eis, P. 1985. *Ein Mehrzielverfahren zur numerischen Berechnung optimaler Feedback–Steuerungen bei beschränkten nichtlinearen Steuerungsproblemen*. Bonner Mathematische Schriften, vol. 166. Bonn: Universität Bonn.

Kuhn, H.W., & Tucker, A.W. 1951. Nonlinear programming. *In:* Neyman, J. (ed), *Proceedings of the second Berkeley symposium on mathematical statistics and probability*. Berkeley: University of California Press.

Kummer, U., Olsen, L.F., Dixon, C.J., Green, A.K., Bornberg-Bauer, E., & Baier, G. 2000. Switching from simple to complex oscillations in calcium signaling. *Biophysical journal*, **79**(September), 1188–1195.

Land, A.H., & Doig, A.G. 1960. An automatic method of solving discrete programming problems. *Econometrica*, **28**, 497–520.

Lebiedz, D., & Brandt-Pollmann, U. 2003. Manipulation of Self-Aggregation Patterns and Waves in a Reaction-Diffusion System by Optimal Boundary Control Strategies. *Physical review letters*, **91**(20), 208301.

Lebiedz, D., Sager, S., Bock, H.G., & Lebiedz, P. 2005. Annihilation of limit cycle oscillations by identification of critical phase resetting stimuli via mixed-integer optimal control methods. *Physical review letters*, **95**, 108303.

Lebiedz, D., Sager, S., Shaik, O.S., & Slaby, O. 2006. Optimal control of self-organized dynamics in cellular signal transduction. *In: Proceedings of the 5th MATHMOD conference*. ARGESIM-Reports, ISBN 3-901608-25-7. (submitted).

Lee, C.K., Singer, A.B., & Barton, P.I. 2004. Global optimization of linear hybrid systems with explicit transitions. *Systems and control letters*, **51**(5), 363–375.

Lee, H.W.J., Teo, K.L., Jennings, L.S., & Rehbock, V. 1999. Control parametrization enhancing technique for optimal discrete-valued control problems. *Automatica*, **35**(8), 1401–1407.

Leineweber, D.B. 1999. *Efficient reduced SQP methods for the optimization of chemical processes described by large sparse DAE models*. Fortschritt-Berichte VDI Reihe 3, Verfahrenstechnik, vol. 613. Düsseldorf: VDI Verlag.

Leineweber, D.B., Bauer, I., Bock, H.G., & Schlöder, J.P. 2003. An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part I: Theoretical aspects. *Computers and chemical engineering*, **27**, 157–166.

Leyffer, S. 1993. *Deterministic methods for mixed-integer nonlinear programming.* Ph.D. thesis, University of Dundee.

Leyffer, S. 2001. Integrating SQP and branch-and-bound for mixed integer nonlinear programming. *Computational optimization and applications*, **18**(3), 295–309.

Leyffer, S. 2003 (June). *Complementarity constraints as nonlinear equations: Theory and numerical experiences.* Tech. rept. Argonne National Laboratory.

Linderoth, J.T., & Savelsbergh, M.W.P. 1999. A computational study of branch and bound search strategies for mixed integer programming. *INFORMS Journal on Computing*, **11**, 173–187.

Locatelli, A. 2001. *Optimal control – an introduction.* Basel Boston Berlin: Birkhäuser.

Logsdon, J.S., & Biegler, L.T. 1993. Accurate determination of optimal reflux policies for the maximum distillate problem in batch distillation. *I & EC Research*, **4**(32), 692.

Logsdon, J.S., Diwekar, U.M., & Biegler, L.T. 1990. On the simultaneous optimal design and operation of batch distillation columns. *Transactions in chemical engineering, Part A*, **68**, 434.

Luo, Z., Pang, J., & Ralph, D. 1996. *Mathematical programs with equlibrium constraints.* Cambridge: Cambridge University Press.

Luyben, W.L. 1988. Multicomponent batch distillation. 1. Ternary systems with slop recycle. *Industrial and engineering chemistry research*, **27**, 642–647.

Luyben, W.L. 1990. Multicomponent batch distillation. 2. Comparison of alternative slop handling and operating strategies. *Industrial and engineering chemistry research*, **29**, 1915–1921.

Macki, J., & Strauss, A. 1995. *Introduction to optimal control theory.* Heidelberg: Springer.

Mak, K.T., & Morton, A.J. 1993. A modified Lin-Kernighan Traveling Salesman Heuristic. *Operations Research Letters*, **13**, 127–132.

Marquardt, W. 1995. Numerical methods for the simulation of differential-algebraic process models. *In:* Berber, R. (ed), *Methods of model based control.* NATO-ASI, vol. 293. E. Kluwer.

Maurer, H., & Osmolovskii, N. P. 2004. Second order sufficient conditions for time-optimal bang-bang control. *SIAM journal on control and optimization*, **42**, 2239–2263.

Maurer, H., Büskens, C., Kim, J.H.R., & Kaya, Y. 2005. Optimization methods for the verification of second-order sufficient conditions for bang-bang controls. *Optimal control methods and applications*, **26**, 129–156.

Mayur, D.N., May, R.A., & Jackson, R. 1970. The time-optimal problem in binary batch distillation with a recycled waste-cut. *Chemical engineering journal*, **1**, 15–21.

McCormick, G.P. 1976. Computability of global solutions to factorable nonconvex programs. part I. Convex underestimating problems. *Mathematical programming*, **10**, 147–175.

Morgan, A.J., & Jacob, R. 1998. Differential modulation of the phases of a Ca2+ spike by the store Ca2+-ATPase in human umbilical vein endothelial cells. *Journal of Physiology*, **513**, 83–101.

Mujtaba, I.M., & Macchietto, S. 1992. An optimal recycle policy for multicomponent batch distillation. *Computers and chemical engineering*, **16**, S273–S280.

Mujtaba, I.M., & Macchietto, S. 1996. Simultaneous optimization of design and operation of multicomponent batch distillation column - single and multiple separation duties. *Journal of process control*, **6**(1), 27–36.

Mujtaba, I.M., & Macchietto, S. 1998. Holdup issues in batch distillation - binary mixtures. *Chemical engineering science*, **53**(14), 2519–2530.

Murty, K.G. 1987. Some NP-complete problems in quadratic and nonlinear programming. *Mathematical programming*, **39**, 117–129.

Neumaier, A. 2004. *Complete search in continuous global optimization and constraint satisfaction*. Cambridge University Press. Pages 271–369.

Nocedal, J., & Wright, S.J. 1999. *Numerical optimization*. Heidelberg: Springer.

Nowak, I. 2005. *Relaxation and Decomposition Methods for Mixed Integer Nonlinear Programming*. Basel Boston Berlin: Birkhäuser.

Oldenburg, J. 2005. *Logic–based modeling and optimization of discrete–continuous dynamic systems*. Fortschritt-Berichte VDI Reihe 3, Verfahrenstechnik, vol. 830. Düsseldorf: VDI Verlag.

Oldenburg, J., Marquardt, W., Heinz, D., & Leineweber, D.B. 2003. Mixed logic dynamic optimization applied to batch distillation process design. *AIChE journal*, **49**(11), 2900–2917.

Ortega, J.M., & Rheinboldt, W.C. 1966. On discretization and differentiation of operators with application to Newton's method. *SIAM journal on numerical analysis*, **3**, 143–156.

Osborne, M.R. 1969. On shooting methods for boundary value problems. *Journal of mathematical analysis and applications*, **27**, 417–433.

Papamichail, I., & Adjiman, C.S. 2004. Global optimization of dynamic systems. *Computers and chemical engineering*, **28**, 403–415.

Pesch, H.J. 1994. A practical guide to the solution of real-life optimal control problems. *Control and cybernetics*, **23**, 7–60.

Pesch, H.J., & Bulirsch, R. 1994. The maximum principle, Bellman's equation and Caratheodory's work. *Journal of optimization theory and applications*, **80**(2), 203–229.

Petty, H. R. 2004. Dynamic chemical instabilities in living cells may provide a novel route in drug development. *Chembiochem*, **5**, 1359.

Plitt, K.J. 1981. *Ein superlinear konvergentes Mehrzielverfahren zur direkten Berechnung beschränkter optimaler Steuerungen*. M.Sci. thesis, Universität Bonn.

Pontryagin, L.S., Boltyanski, V.G., Gamkrelidze, R.V., & Miscenko, E.F. 1962. *The mathematical theory of optimal processes*. Chichester: Wiley.

Quesada, I., & Grossmann, I.E. 1992. An LP/NLP based branch and bound algorithm for convex MINLP optimization problems. *Computers and chemical engineering*, **16**, 937–947.

Raghunathan, A.U., & Biegler, L.T. 2003. Mathematical programs with equilibrium constraints (MPECs) in process engineering. *Computers and chemical engineering*, **27**, 1381–1392.

Rehbock, V., & Caccetta, L. 2002. Two defence applications involving discrete valued optimal control. *ANZIAM journal*, **44**(E), E33–E54.

Robbins, H.M. 1967. A generalized Legendre-Clebsch condition for the singular cases of optimal control. *IBM journal*, **11**(4), 361–372.

Sager, S. 2005. *Numerical methods for mixed–integer optimal control problems*. Tönning, Lübeck, Marburg: Der andere Verlag. ISBN 3-89959-416-9.

Sager, S., Bock, H.G., Diehl, M., Reinelt, G., & Schlöder, J.P. 2006. Numerical methods for optimal control with binary control functions applied to a Lotka-Volterra type fishing problem. *Pages 269–289 of:* Seeger, A. (ed), *Recent advances in optimization (proceedings of the 12th French-German-Spanish conference on optimization)*. Lectures Notes in Economics and Mathematical Systems, vol. 563. Heidelberg: Springer.

Santos, L.O., Oliveira, N.M.C de, & Biegler, L.T. 1995. Reliable and efficient optimization strategies for nonlinear model predictive control. *Pages 33–38 of: Proc. fourth IFAC symposium DYCORD+'95*. Oxford: Elsevier Science.

Schäfer, A.A.S. 2005. *Efficient reduced Newton-type methods for solution of large-scale structured optimization problems with application to biological and chemical processes*. Ph.D. thesis, Universität Heidelberg.

Schäfer, A.A.S., Brandt-Pollmann, U., Diehl, M., Bock, H.G., & Schlöder, J.P. 2003. Fast optimal control algorithms with application to chemical engineering. *Pages 300–307 of:* Ahr, D., Fahrion, R., Oswald, M., & Reinelt, G. (eds), *Operations research proceedings*. Heidelberg: Springer.

Schlöder, J.P. 1988. *Numerische Methoden zur Behandlung hochdimensionaler Aufgaben der Parameteridentifizierung*. Bonner Mathematische Schriften, vol. 187. Bonn: Universität Bonn.

Schulz, V.H. 1998. Solving discretized optimization problems by partially reduced SQP methods. *Computing and visualization in science*, **1**, 83–96.

Schulz, V.H., Bock, H.G., & Steinbach, M.C. 1998. Exploiting invariants in the numerical solution of multipoint boundary value problems for DAEs. *SIAM journal on scientific computing*, **19**, 440–467.

Schweiger, C., & Floudas, C. 1997. Interaction of design and control: Optimization with dynamic models. *Pages 388–435 of:* Hager, W.W., & Pardalos, P.M. (eds), *Optimal control: Theory, algorithms, and applications*. Kluwer Academic Publishers.

Seguchi, H., & Ohtsuka, T. 2003. Nonlinear receding horizon control of an underactuated hovercraft. *International journal of robust and nonlinear control*, **13**(3–4), 381–398.

Shaikh, M.S. 2004. *Optimal control of hybrid systems: Theory and algorithms*. Ph.D. thesis, Department of Electrical and Computer Engineering, McGill University, Montrl, Canada.

Shaikh, M.S., & Caines, P.E. 2006. On the hybrid optimal control problem: The hybrid minimum principle and hybrid dynamic programming. *IEEE transactions on automatic control*. (under review).

Skrifvars, H., Leyffer, S., & Westerlund, T. 1998. Comparison of certain MINLP algorithms when applied to a model structure determination and parameter estimation problem. *Computers and chemical engineering*, **22**, 1829–1835.

Srinivasan, B., Palanki, S., & Bonvin, D. 2003. Dynamic Optimization of Batch Processes: I. Characterization of the nominal solution. *Computers and chemical engineering*, **27**, 1–26.

Stein, O., Oldenburg, J., & Marquardt, W. 2004. Continuous reformulations of discrete-continuous optimization problems. *Computers and chemical engineering*, **28**(10), 3672–3684.

Stoer, J., & Bulirsch, R. 1992. *Introduction to numerical analysis*. Heidelberg: Springer.

Stryk, O. von, & Glocker, M. 2000. Decomposition of mixed-integer optimal control problems using branch and bound and sparse direct collocation. *Pages 99–104 of: Proc. adpm 2000 – the 4th international conference on automatisation of mixed processes: Hybrid dynamical systems*.

Stubbs, R.A., & Mehrotra, S. 1999. A branch-and-cut method for 0-1 mixed convex programming. *Mathematical programming*, **86**(3), 515–532.

Stubbs, R.A., & Mehrotra, S. 2002. Generating convex quadratic inequalities for mixed 0-1 programs. *Journal of global optimization*, **24**, 311–332.

Stursberg, O., Panek, S., Till, J., & Engell, S. 2002. Generation of optimal control policies for systems with switched hybrid dynamics. *Chap. Generation of Optimal Control Policies for Systems with Switched Hybrid Dynamics, pages 337–352 of:* Engell, S., Frehse, G., & Schnieder, E. (eds), *Modelling, analysis and design of hybrid systems*. Springer.

Sussmann, H.J. 1999. A maximum principle for hybrid optimal control problems. *In: Conference proceedings of the 38th IEEE conference on decision and control*.

Tawarmalani, M., & Sahinidis, N. 2002. *Convexification and global optimization in continuous and mixed-integer nonlinear programming: Theory, algorithms, software, and applications*. Kluwer Academic Publishers.

Terwen, S., Back, M., & Krebs, V. 2004. Predictive powertrain control for heavy duty trucks. *Pages 451–457 of: Proceedings of IFAC symposium in Advances in automotive control*.

Toumi, A., Engell, S., Diehl, M., Bock, H.G., & Schlöder, J.P. 2006. Efficient optimization of Simulated Moving Bed Processes. *Chemical engineering and processing*. (submitted as invited article).

Tsang, T.H., Himmelblau, D.M., & Edgar, T.F. 1975. Optimal control via collocation and non-linear programming. *International journal on control*, **21**, 763–768.

Turkay, M., & Grossmann, I.E. 1996. Logic-based MINLP algorithms for the optimal synthesis of process networks. *Computers and chemical engineering*, **20**, 959–978.

Unger, J., Kröner, A., & Marquardt, W. 1995. Structural analysis of differential-algebraic equation systems – theory and applications. *Computers and chemical engineering*, **19**, 867–882.

Vavasis, S.A. 1995. *Handbook of Global Optimization*. Kluwer Academic Press. Chap. Complexity issues in global optimization: A survey, pages 27–41.

Viswanathan, J., & Grossmann, I.E. 1990. A combined penalty function and outer-approximation method for MINLP optimization. *Computers and chemical engineering*, **14**, 769–782.

Vlastos, Gregory. 1967. *The encyclopedia of philosophy*. New York: MacMillan.

Volterra, V. 1926. Variazioni e fluttuazioni del numero d'individui in specie animali conviventi. *Mem. r. accad. naz. dei lincei.*, **VI-2**.

Walleczek, J. 2000. *Self-organized biological dynamics and nonlinear control*. Cambridge: Cambridge University Press.

Walter, W. 1993. *Gewöhnliche Differentialgleichungen*. Heidelberg: Springer. ISBN: 038756294X.

Weiser, M., & Deuflhard, P. 2001. *The central path towards the numerical solution of optimal control problems*. Tech. rept. 01-12. ZIB.

Westerlund, T., & Pettersson, F. 1995. A cutting plane method for solving convex MINLP problems. *Computers and chemical engineering*, **19**, S131–S136.

Wikipedia. 2005. *The free encyclopedia*. http://en.wikipedia.org/.

Wilson, R.B. 1963. *A simplicial algorithm for concave programming*. Ph.D. thesis, Harvard University.

Winfree, A. 2001. *The geometry of biological time*. New York: Springer.

Wolsey, L.A., & Nemhauser, G.L. 1999. *Integer and Combinatorial Optimization*. Chichester: Wiley.

Wunderling, R. 1996. *Paralleler und Objektorientierter Simplex-Algorithmus*. Ph.D. thesis, Konrad-Zuse-Zentrum Berlin.

Zelikin, M.I., & Borisov, V.F. 1994. *Theory of chattering control with applications to astronautics, robotics, economics and engineering*. Basel Boston Berlin: Birkhäuser.

Zhang, J., Johansson, K.H., Lygeros, J., & Sastry, S. 2001. Zeno hybrid systems. *International journal of robust and nonlinear control*, **11**, 435–451.