TECHNISCHE
UNIVERSITÄT
WIEN
Vienna University of Technology

DISSERTATION

The Adjoint Gradient Method for Time-Optimal Control of Multibody Systems

ausgeführt zum Zwecke der Erlangung des akademischen Grades
eines Doktors der technischen Wissenschaften (Dr. techn.)
eingereicht an der Technischen Universität Wien,
Fakultät für Maschinenwesen und Betriebswissenschaften
von

Dipl.-Ing. Philipp Eichmeir
geboren am 01.12.1990
Mat. Nr.: 01057915

unter der Leitung von

Priv. Doz. Dipl.-Ing. Dr. Wolfgang Steiner
am Institut für Mechanik und Mechatronik, E325
Abteilung für Mechanik fester Körper

begutachtet von

Ao.Univ.Prof. Dipl.-Ing. Dr. Alois Steindl
Technische Universität Wien
Institut für Mechanik und Mechatronik
Getreidemarkt 9, 1060 Wien

Prof. Ahmed A. Shabana, PhD
University of Illinois at Chicago
Dep. of Mechanical & Industrial Engineering
842 W. Taylor Street, Chicago, IL 60607

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.ac.at

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Eidesstattliche Erklärung

Dipl.-Ing. Philipp Eichmeir
Edisonstraße 30/A8
4600 Wels

Ich erkläre an Eides statt, dass die vorliegende Arbeit nach den anerkannten Grundsätzen für wissenschaftliche Abhandlungen von mir selbstständig erstellt wurde. Alle verwendeten Hilfsmittel, insbesondere die zugrunde gelegte Literatur, sind in dieser Arbeit genannt und aufgelistet. Die aus den Quellen wörtlich entnommenen Stellen, sind als solche kenntlich gemacht.
Das Thema dieser Arbeit wurde von mir bisher weder im In- noch Ausland einer Beurteilerin/einem Beurteiler zur Begutachtung in irgendeiner Form als Prüfungsarbeit vorgelegt. Diese Arbeit stimmt mit der von den Begutachterinnen/Begutachtern beurteilten Arbeit überein.

Wien, am 25.03.2022

Philipp Eichmeir

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Danksagung

An dieser Stelle möchte ich die Gelegenheit nutzen, um mich bei jenen zu bedanken, die mich während meines Dissertationsstudiums begleitet und mich bei meiner Ausbildung unterstützt haben.

Mein besonderer Dank gilt meinem Doktorvater, Herrn Prof. Wolfgang Steiner, der bereits früh während meines Diplomstudiums mit seinen Lehrveranstaltungen mein Interesse an der technischen Mechanik und der Forschung weckte. *Lieber Wolfgang, ich danke dir für die Unterstützung bei meiner Forschungsarbeit und für die Betreuung meiner Dissertation; ich habe in den vergangenen Jahren unvorstellbar viel von dir gelernt. In besonders schöner Erinnerung bleiben mir die gemeinsamen Wanderungen und das gemeinsame Musizieren mit dir am Klavier.*

Herzlichen Dank an Herrn Prof. Alois Steindl von der Technischen Universität Wien, der die Betreuung meiner Arbeit als Erstgutachter übernommen hat und mich während meines Doktoratsstudiums stets mit seinem umfassenden Wissen unterstützt hat. In seinen Lehrveranstaltungen über die höhere Mechanik wurde mein Verständnis vertieft und meine Leidenschaft für die Forschung weiter gestärkt. *Lieber Herr Prof. Steindl, ich danke Ihnen für die Unterstützung bei meiner Arbeit und für die schöne Zeit in Wien. Sie haben die Gabe, die Freude, die Sie selbst an der Mathematik und Mechanik empfinden, weiterzugeben.*

Ein weiterer Dank geht an meinen Zweitgutachter, Herrn Prof. Ahmed A. Shabana, den ich leider noch nicht persönlich kennen lernen durfte. Aufgrund seiner bahnbrechenden Arbeiten im Bereich der Mehrkörperdynamik, begleiten mich seine wissenschaftlichen Abhandlungen bereits seit meinem Diplomstudium. *Dear Prof. Shabana, thank you for reviewing my thesis and for the highly valuable comments.*

Des Weiteren möchte ich mich bei meiner Kollegin Frau Prof. Karin Nachbagauer und meinem ehemaligen Kollegen Thomas Lauß für die schöne Zusammenarbeit und die Freundschaft bedanken. *Liebe Karin und lieber Thomas, danke für die Unterstützung, die fachlichen Diskussionen und vor allem für die unvergessliche Zeit in der Arbeit, auf Tagungsbesuchen und bei gemeinsamen Unternehmungen in der Freizeit.*

Nicht zuletzt danke ich meiner Familie, besonders meiner Mutter Gerlinde und meinem verstorbenen Vater Wilhelm für die Liebe und den Rückhalt bei meiner Ausbildung. *Danke, dass ihr immer an mich geglaubt, und mich jahrelang bei meiner Ausbildung unterstützt habt.*

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Kurzfassung

Die Theorie der optimalen Steuerung wird in vielen Bereichen der Naturwissenschaften eingesetzt, um zeitliche Abläufe von Systemen zu optimieren und Trajektorien zu planen. Zur Auffindung von optimalen Steuerungen wird eine Zielfunktion, unter Berücksichtigung von Nebenbedingungen, minimiert oder maximiert. Die Zielfunktion beschreibt dabei z. B.: die Abweichung zu einer Solltrajektorie, den Energieverbrauch eines Systems oder die benötigte Zeit eines Ablaufs.

Im Rahmen dieser Arbeit wird letztere Zielfunktion betrachtet, um die zeitoptimale Trajektorie für dynamische Systeme zu berechnen. Zeitoptimale Problemstellungen treten z. B. in der Fahrzeugmechanik, bei der Bestimmung der minimalen Rundenzeit einer Rennstrecke auf, oder in der Robotik, wenn die Bahnkurve eines Roboters so gestaltet werden soll, dass die Zeit für ein Punkt-zu-Punkt Manöver minimiert wird.

Bisher wurden solche Aufgabenstellungen weitgehend als Zwei-Punkt Randwertprobleme untersucht, die schwer zu lösen sind, und eine Startlösung nahe der optimalen Lösung erfordern. Das Ziel dieser Arbeit besteht nun darin, eine iterative, gradientenbasierte Lösungsstrategie zu entwickeln, die auf komplexe Mehrkörpersysteme angewendet werden kann. Die Beschreibung der Systeme erfolgt in einem ersten Schritt mit gewönlichen Differentialgleichungen und wird dann auf differential-algebraische Gleichungen erweitert. Für die Berechnung des Gradienten, d. h. die Variation des Steuersignals, die die größte lokale Reduktion der Endzeit verursacht, wird die sogenannte adjungierte Methode herangezogen.

Die adjungierte Methode bietet dabei eine effiziente Möglichkeit, um den Gradienten eines Kostenfunktionals hinsichtlich des Steuersignals zu berechnen. Die Grundidee dafür, ist die Einführung zusätzlicher adjungierter Variablen, die durch eine Reihe von adjungierten Differentialgleichungen bestimmt werden, aus denen der Gradient berechnet werden kann.

Dabei ist es Ziel dieser Arbeit, die klassische adjungierte Gradientenberechnung für zeitoptimale Problemstellungen in der Mehrkörperdynamik zu formulieren, und zu zeigen, dass damit zeitoptimale Problemstellungen in der Dynamik effizient berechnet werden können.

Die in dieser Arbeit vorgestellten Methoden bieten einen robusten Alternativweg zur Lösung des zugrundeliegenden Randwertproblems, und ermöglichen es, ausgehend von einer Initialsteuerung, iterativ die zeitoptimale Lösung von Systemen zu berechnen.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Abstract

The theory of optimal control is used in many areas of natural science to optimize time sequences and to plan trajectories. Thereby the control of a system is searched, which minimizes or maximizes a desired objective function, under consideration of constraints. The objective function describes, for example, the deviation from a target trajectory, the energy consumption of a system or the operation period of a process.

In the context of this work, the latter objective function is considered in order to compute the time-optimal trajectory for dynamic systems. Time-optimal control problems arise, for example, in vehicle mechanics, when determining the minimum lap time on a race track, or in robotics, when the trajectory of a robot has to be designed in order to minimize the time required for a point-to-point maneuver.

So far, such tasks have been widely studied as two-point boundary value problems, which are difficult to solve, and require an initial guess close to the optimal solution. However, the goal of this work is to develop an iterative gradient-based solution strategy that can be applied to complex multibody systems. The gradient, i. e., the variation of a control signal that results in the largest local change of the final time, is computed using the so-called adjoint method.

The adjoint method provides an efficient way to compute the gradient of a cost functional with respect to the control signal. The key idea is to introduce adjoint variables that are governed by a set of adjoint differential equations from which the gradient can then be computed in a straight forward way.

The goal of this work is to formulate the classical adjoint gradient approach for time-optimal control problems in multibody dynamics and to show how this method can be efficiently used to solve time-optimal control problems arising from modern engineering science.

The methods in this work provide a robust way to solve the underlying boundary value problem and enable the iterative computation of time-optimal solutions in multibody system dynamics.

TU WIEN Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Preface

The present work was written during my employment as research assistant at the University of Applied Sciences Upper Austria in cooperation with the Institute of Mechanics and Mechatronics (E325) at the Vienna University of Technology in the period 2017–2021.

My scientific career was initiated with the support program of a Josef Ressel Center of the Christian Doppler Research Society. During the two-year period of the Josef Ressel Center the methodical approaches for the solution of optimal lap times of racing cars were developed. By receiving a dissertation grant from the Austrian Research Promotion Agency (FFG), the work was completed in another two years during a $50\ \%$ employment. During this time a scientifically very promising way was found to extend the gradient based optimization methods to problems with given end conditions for state variables of the system.

As described in the abstract, the thesis deals with the solution of time-optimal control problems in multibody dynamics. However, the work is not meant to be a complete instruction guide for solving time-optimal control problems, but rather focuses on the adjoint method for minimal end time problems. Note that in this work the designations *adjoint method* and *adjoint gradient method* can be used synonymously.

The thesis is divided into eight chapters and organized as follows:

Chapter 1: In the first chapter, the reader is guided into the topic, state of the art methods are presented, a literature review is given, and the scientific contribution of the dissertation is emphasized.

Chapter 2: In the second chapter, a deeper understanding of the calculus of variations is given in order to comprehend the newly developed theory and methodological approaches in the still following chapters in this book. A special emphasis is placed on free endpoint problems, as these types of problems also occur when looking for time-optimal controls.

Chapter 3: The third chapter of this thesis gives an overview of standard optimization techniques and concentrates on customized methods to solve time-optimal control problems. To cite an example, a new constrained optimization strategy is presented, which gives a first insight into the solution strategy pursued to solve time-optimal control problems in multibody dynamics.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

In the following, Chapters 4 through 7 are devoted to optimal control theory. In detail, Chapter 4 deals with established methods in optimal control theory while Chapters 5, 6 and 7 are concerned with the formulation of the adjoint method. Several parts of these chapters have already been presented at conference visits, invited lectures and are published in peer reviewed journals.

Chapter 4:	An introduction to optimal control theory is given and standard solution methods based on the two-point boundary value problem are presented in this chapter. The necessary conditions for an optimal control solution are derived following Pontryagin's variational principles. Moreover, different boundary terms are considered for the optimal control task.
Chapter 5:	This chapter discusses the adjoint method for solving a special class of time-optimal control problems in which the final state of a system lies on a hypersurface. These problems occur, for example, in vehicle dynamics, where the final state is defined by crossing the finish line of a race track. Finally, the proposed method is demonstrated on examples from satellite and vehicle dynamics.
Chapter 6:	In this chapter, we introduce a more general approach for the determination of time-optimal controls subject to end conditions. In order to show the efficiency of this method, the algorithm is tested with several examples at the end of the chapter.
Chapter 7:	This chapter deals with the extension of the adjoint method discussed in the previous chapter to time-optimal control problems of multibody systems described by a set of differential-algebraic equations of index three. The presented approach has been published by the author only in excerpts in the open literature so far.
Chapter 8:	At the end of the thesis a short summary is given, highlighting the advantages and disadvantages of the newly developed methods and reflecting the author's experience regarding convergence behavior and robustness compared to previous methods in this field.

Note that in the present work the following convention is used to describe a general size x: Scalar sizes are written with x. Physical vectors are written with $\vec{x}$ while mathematical vectors are described by $\boldsymbol{x}$ and matrices are represented by $\mathbf{X}$. Partial derivatives of a function f are denoted by f_x and total time derivatives are written as $\dot{x}$. Note, due to notation simplicity, the dependencies on time are omitted in many derivations. A minimizing solution is given by x while a solution x^* denotes a modification of a minimizing solution.

– Philipp Eichmeir, *Wels, March 25, 2022*

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Contents

1 Introduction **1**
1.1 State of the Art . . . 1
1.2 Scientific Contribution . . . 3

2 The Calculus of Variations **5**
2.1 Historical Notes . . . 5
2.2 The Variation Principle for Free Final Points . . . 6
2.2.1 Boundary Conditions . . . 9
2.2.2 The δ-Operation for a Transformed Time Coordinate . . . 10
2.3 The Brachistochrone Problem . . . 11

3 Numerical Optimization **14**
3.1 Optimization without Constraints . . . 14
3.1.1 Search Directions . . . 15
3.1.2 Step Length Determination . . . 17
3.2 Optimization with Constraints . . . 18
3.2.1 Projected Gradient Method . . . 18
3.2.2 A Modified Gradient Method . . . 20

4 Time-Optimal Control Theory **25**
4.1 Historical Notes . . . 25
4.2 Problem Formulation . . . 26
4.3 The Variational Approach to Time-Optimal Control . . . 27
4.4 Boundary Conditions . . . 28
4.4.1 Final State Variables Lying on a Surface . . . 29
4.4.2 Some Final State Variables Specified . . . 29
4.5 Pontryagin's Minimum Principle . . . 30
4.6 Homotopy and Continuation Methods . . . 32
4.7 Problems . . . 35
4.7.1 The Lunar Ascent Problem . . . 35
4.7.2 Inverse Double Pendulum . . . 39

5 The Adjoint Method for Final States Lying on a Surface 43
5.1 Problem Formulation . . . 43
5.2 Transformation to a Hybrid Domain . . . 44
5.3 Transformation to Space Domain . . . 47
5.4 Examples . . . 49
  5.4.1 Orbital Transfer Problem . . . 49
  5.4.2 Minimum Lap Time Problem . . . 52

6 The Adjoint Method for a Set of Specified Final States 62
6.1 Problem Formulation and Solution Approach . . . 62
6.2 Bang-Bang Principle . . . 68
6.3 The Algorithm . . . 70
6.4 Examples . . . 71
  6.4.1 Trajectory Planning for a Moon Landing . . . 71
  6.4.2 Planar Robot Arm . . . 79
  6.4.3 Inverse Double Pendulum . . . 84

7 The Adjoint Method in Multibody Dynamics 87
7.1 Problem Formulation . . . 87
7.2 Optimal Control of Multibody Systems . . . 89
  7.2.1 Consistent Boundary Conditions . . . 92
  7.2.2 Bang-Bang Principle . . . 93
  7.2.3 An Introductory Example . . . 95
7.3 A Backward Differentiation Scheme . . . 99
  7.3.1 The Solution of the Adjoint Differential Equations . . . 100
  7.3.2 The Solution of the Adjoint-Influence Differential Equations . . . 102
  7.3.3 The Algorithm . . . 102
7.4 Examples . . . 103
  7.4.1 Planar Robot Arm . . . 104
  7.4.2 Industrial Robot . . . 108

8 Conclusion 114
8.1 Summary . . . 114
8.2 Outlook . . . 115

A Single Track Vehicle Model 117

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Chapter 1

Introduction

In the last few years, engineers and scientists have been increasingly faced with the challenge of creating more complex simulation models to solve formidable tasks in mechanical engineering. For this reason, there is a growing demand in both research and industry to develop efficient and reliable algorithms to cope with these tasks and to design optimal processes.

A powerful tool is offered by modern theory of optimal control, which is concerned with finding controls of systems in such a way that a desired target function is minimized. It enables scientists and industrial users to plan the trajectory of dynamic systems and to operate systems at their load limits.

If we consider a car, for example, we might be interested in the optimal control of the gas pedal, brakes, and steering, while the target function could display fuel consumption or measure the lap time on a race track. In this work we will focus on the latter class of problems and ask for controlling dynamic systems so that the time required is minimal. Time-optimal control problems are of great interest for a wide spectrum of applications in the field of mechanics not only originating from engineering but also from cybernetics, biomechanical or even medical investigations. A special focus lays on the development and the extension of modern numerical methods for solving time-optimal control problems which arise in the context of multibody dynamics.

The goal is to develop a robust algorithm for solving time-optimal control problems in multibody dynamics in a reasonable amount of time. The novel methods in this work should support scientific colleagues and researchers from industry in developing proposals for design improvements of mechanical systems in aerospace applications, vehicle dynamics and robotics.

1.1 State of the Art

The history of optimal control theory goes back to the first half of the 20th century and its research has made great progress with the Apollo moon program in the sixties of the past century [35, 37, 16]. Among the references mentioned, the works by Kelley, Bryson and Ho are today fundamental for modern optimal control.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Despite the long history of optimal control, the topic is still interesting for research. Even today, the computation of trajectories plays an important role in many application areas. Due to the significant increase of computing power in the last decades, the problems of optimal control could gain in complexity and difficulty. Nevertheless, among all optimal control tasks, the computation of time-optimal solutions remains a challenge.
While the problem of finding optimal trajectories can be formulated and solved in a variety of ways, a basic distinction can be made between (a) *direct* and (b) *indirect* methods discussed in fundamental books on optimal control as e. g. [39, 7, 16].

(a) Direct methods are also addressed as nonlinear programming and transform the dynamic problem to a static problem by a parameterization of the control variables, see [9] for example. Numerous methods follow a direct approach, e. g. interior point algorithms, active set, sequential quadratic programming, pseudospectral method, to name but a few. Various authors [58, 19] pursue a nonlinear programming strategy. The latter mentioned work by Casanova deals with the minimum lap time problem, which is still a challenging research issue in optimal control theory and is therefore addressed in this book as well. More recently, Miller [46] deals with time-optimal trajectory planning for guiding the Lunar Excursion Module from the Apollo program to and from the Moon's surface. There, the optimal control problem is solved by using a nonlinear programming strategy, too. Moreover, an efficient alternative to solving constrained optimization problems could be the interior point method which shows robustness with respect to initial values, see e. g. [73]. Here, the infinite-dimensional optimal control problem is reduced to a finite-dimensional static optimization problem, which can be solved with classical methods, like Newton methods. Therefore, the differential equation describing the dynamics of the system has to be discretized in time by finite differences yielding algebraic equations. This time discretization can grow tremendously in size in case of a long simulation time or a dynamic response as e. g. in bang-bang-control systems.

(b) Indirect methods rely on solving the optimality conditions stated by the Soviet mathematician Lev Semenovich Pontryagin, which leads to a two-point boundary value problem that can usually be solved by shooting methods or full collocation techniques. Various authors [45, 20, 6, 43, 30] discuss the solution methods of the boundary value problem, which are already well established throughout a broad spectrum of engineering sciences. In the article by Graichen [30], a homotopy method is presented, which is well suited to solve the boundary value problem particularly for singular control tasks. This homotopy method is used to obtain some reference solutions in order to compare the methods in this book, but more on that later.
For more details to indirect and direct methods, an overview of both is given in [11, 21, 22].

An alternative to the aforementioned methods is provided by gradient methods, which are regarded as particularly robust with respect to initial controls. The key idea of gradient methods is an iterative strategy to find a control history that leads to a minimum of the cost functional. Numerous strategies are available to compute the control for which a functional is minimal, as e. g., the method of the steepest descent, the conjugate gradient method, the gradient projection method, the Gauss-Newton method, or quasi-Newton methods like the Broyden-Fletcher-Goldfarb-Shanno

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

(BFGS) algorithm when estimating the Hessian, see e. g. [56] for an overview and comparison of performance behavior. In any cases, the gradient of the cost functional with respect to the control history has to be computed. Unfortunately, the traditional numerical gradient computation is very time consuming and especially in multibody simulations the determination of a gradient is often the bottleneck for computational efficiency. To determine the partial derivatives with respect to each time-discrete control parameter by means of a numerical differentiation, the entire dynamic system must be solved once for a variation of every discretization parameter. In order to circumvent the tedious gradient computation by numerical differentiation, the so-called adjoint method serves as the most efficient strategy in this case. Various authors have already utilized the adjoint method in the field of multibody systems, as e. g., [8, 34], but nevertheless, there are still numerous research questions.

The approach in multibody system dynamics is well suited for solving dynamical problems with large rotations and deformations. Compared with other numerical engineering methods like the Finite Element Method, which is utilized even in smaller enterprises, multibody simulations have still a high potential for growing research and industrial recognition. Various authors have already discussed the complexity and computational strategies of multibody systems [57, 60, 61, 63, 72]. The motion of multibody systems is frequently formulated by a set of differential-algebraic equations expressed in redundant generalized coordinates. Due to algebraic constraints, the equations of motion are extended by constraint forces using Lagrange multipliers, see [64] for details. Moreover, a vector of time dependent control signals is actuating the system, which we aim to identify in this work in such a way as to minimize a cost functional measuring the time period.

1.2 Scientific Contribution

The main novelty of the dissertation lies in the extension of the classical adjoint method, e. g. presented in [66] to compute the gradient of a cost functional for time-optimal control problems efficiently. The adjoint method is based on decoupling boundary conditions of the optimality conditions and computes the gradient of a cost functional by a sequential integration of the equations of motion and one or two systems of co-state (adjoint) equations.

The basic idea of the adjoint method goes back to the gradient technique pioneered by Bryson and Ho [16]. Using the gradient information in nonlinear optimization routines, optimal control problems in multibody systems can be solved iteratively as described in [47, 50, 66]. The mentioned articles discuss the application of the adjoint method for parameter identification and for solving inverse dynamical problems in multibody dynamics.

Several facets of the adjoint method regarding time-optimal control problems are considered in this thesis and the most promising formulation is enhanced to deal with constrained multibody simulations described by differential-algebraic equations of index three. Therefore, an efficient computation of the gradient using the adjoint equations and the adjoint-influence equations embedded in the framework of multibody system dynamics is necessary. Moreover, an architecture for time integration solvers for the adjoint equations as well as the adjoint-influence equations is developed. Because of the highly nonlinear multibody problems, special focus is given to time-

and memory-efficiency. On the one hand, this work promotes basic research in the field of optimal control theory, and on the other hand, it enables the computation of time-optimal controls in multibody system dynamics. While writing this thesis, the use of the adjoint method for solving time-optimal control problems has already been published by the author in renowned journals [21, 22, 23], the published papers consider only examples described by ordinary differential equations.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Chapter 2

The Calculus of Variations

Nature and mankind have always striven to design optimal shapes and processes. Many of them are intuitive or occur in a natural way so that a maximum or minimum is achieved. For example, if we consider a heavy chain suspended between two points, then the chain takes the shape for which the potential energy is minimized. This natural response and behavior were already observed by Euler while working on the calculus of variations. He also described this phenomenon in a famous quotation:

> *"For since the fabric of the universe is most perfect, and is the work of a most wise Creator, nothing at all takes place in the universe in which some relation of maximum and minimum does not appear."* [52]
>
> – Leonhard Euler, *Lausanne and Geneva, 1744*

The calculus of variations is a powerful tool to find functions for which a target functional is maximized or minimized. In this chapter we shall discuss variational techniques to address issues which are fundamental for solving time-optimal control problems.

2.1 Historical Notes

The history of the calculus of variations traces back to the 17th century [13, 28], and has its origin 1662 with the French mathematician Pierre de Fermat, when he discovered that light moves through stratified media in minimal time. However, almost 25 years earlier, Galileo Galilei had already formulated the problems of calculating the shape of a heavy chain and the Brachistochrone curve, but his assumed solutions were not correct. Later, Johann Bernoulli studied Galileo's two problems and used Fermat's approaches to present an approximate solution to the Brachistochrone problem. Therefore, he divided the area between the given end points into thin horizontal layers and made the assumption that the falling particle moves uniformly in each layer and that the trajectory curve at each interface changes as if a light beam is refracted, thus minimizing the total

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

time of descent. In 1685, Isaac Newton first formulated the basic principles of the calculus of variations and published a method for finding the shape of a projectile, described by a body of revolution, in such a way that the air resistance is minimized. The treatise in his *Philosophiae Naturalis Principia Mathematica* has a significant impact on the further progress of the calculus of variations, as it is the first problem in this field which was both correctly formulated and solved. In the late 17th century, Johann Bernoulli asked the mathematical community to solve Galileo's Brachistochrone problem. The problem was then solved by himself, his brother Jacob, by Leibniz and by Newton. Due to the controversies between Newton and Bernoulli at that time, Newton published his solution of the Brachistochrone problem anonymously. However, Johann Bernoulli revealed Newton's solution by noticing that one can recognize a lion from its touch.

After the solution of the Brachistochrone problem, the Bernoulli brothers posed many other problems, thus founding a new field of mathematics. Inspired by Johann Bernoulli, the Swiss mathematician Leonard Euler published in 1744 the textbook *"The Method of Finding Curves that Show Some Property of Maximum or Minimum"* that entered the history of the calculus of variations. Joseph-Louis Lagrange invented the method of variations, which Euler generously praised and prompted him to give the calculus of variations its name. Another groundbreaking work of Lagrange is the method of multipliers, which measures the sensitivity of the functional with respect to states. Euler also adopted this idea from Lagrange and wrote the necessary first order conditions for a stationary solution which we know today as the Euler-Lagrange equations.

2.2 The Variation Principle for Free Final Points

Many problems in geometry deal with the task of finding a plane curve between two points for which a functional is maximized/minimized. One of these problems is the mentioned Brachistochrone curve that is a path between two points on which a particle mass travels in minimal time under the influence of gravity. In this chapter we are interested in solving such problems, but before we describe the solution of the Brachistochrone problem, we first summarize some basic principles of the calculus of variations. Various textbooks deal with the calculus of variations, e. g., see [71, 40], where the book by Lanczos deserves special mention. Later in this work, we are dealing with free endpoint problems, thus for the variational task, we also consider the initial point fixed and the endpoint free.

In general, a cost functional can be expressed in the form

$$J(y(t)) = \int_{t_0}^{t_f} F(y(t), \dot{y}(t), t)\,\mathrm{d}t, \tag{2.1}$$

and we are looking for the function $y(t)$ which minimizes J. The initial point of the function $y(t)$ is prescribed by t_0 and $y(t_0) = y_0$, while we consider the final time t_f and the final function value $y(t_f)$ free. For example, if we are looking for the shortest path between two points, the integrand in Eq. (2.1) is given by the arc length $F := \sqrt{1 + (\dot{y}(t))^2}$. In order to find a function $y(t)$ which minimizes J, we investigate a change of $y(t)$. Hence, the modified function $y^*(t)$ reads

$$y^*(t) = y(t) + \delta y(t) = y(t) + \varepsilon\,\eta(t), \tag{2.2}$$

in which $y(t)$ is the function which minimizes J, $\eta(t)$ is an arbitrary test function and ε is a small disturbance parameter. Note that the test function $\eta(t)$ must also fulfill the initial condition, hence $\eta(0) = 0$. Analogously to Eq. (2.2), the modified final time t_f^* is given by

$$t_f^* = t_f + \delta t_f = t_f + \varepsilon\,\xi, \tag{2.3}$$

where t_f is the free final time, $\xi \in \mathbb{R}$ is an arbitrary test number and ε is again the small disturbance parameter. Inserting the modified function $y^*(t)$ and the modified final time t_f^* into Eq. (2.1) we obtain

$$J(y^*(t)) = J(y(t) + \varepsilon\,\eta(t)) = \int_{t_0}^{t_f+\varepsilon\,\xi} F(y(t) + \varepsilon\,\eta(t), \dot{y}(t) + \varepsilon\,\dot{\eta}(t), t)\,\mathrm{d}t. \tag{2.4}$$

In order to apply a solution strategy to our problem, we investigate the change of J caused by the variation of the function $y^*(t)$, yielding

$$\begin{aligned} J(y^*(t)) - J(y(t)) &= J(y(t) + \varepsilon\,\eta(t)) - J(y(t)) \\ &\approx J(y(t)) + \varepsilon \frac{\mathrm{d}}{\mathrm{d}\varepsilon}\Big|_{\varepsilon=0} J\big(y(t) + \varepsilon\,\eta(t)\big) - J(y(t)) = \delta J, \end{aligned} \tag{2.5}$$

in which we utilized a linear Taylor expansion of $J(y^*)$ with respect to the disturbance parameter ε. Hence, we obtain a 1st order necessary condition for a maximizing/minimizing solution $y(t)$ when the variation of the functional

$$\delta J = \varepsilon \frac{\mathrm{d}}{\mathrm{d}\varepsilon}\Big|_{\varepsilon=0} J(y(t) + \varepsilon\,\eta(t)), \tag{2.6}$$

vanishes. Applying Eq. (2.6) to Eq. (2.4) yields

$$\delta J = \varepsilon \frac{\mathrm{d}}{\mathrm{d}\varepsilon}\Big|_{\varepsilon=0} \int_{t_0}^{t_f+\varepsilon\,\xi} F(y(t) + \varepsilon\,\eta(t), \dot{y}(t) + \varepsilon\,\dot{\eta}(t), t)\,\mathrm{d}t, \tag{2.7}$$

or, respectively, after dividing the integral into two parts

$$\begin{aligned} \delta J = \varepsilon \frac{\mathrm{d}}{\mathrm{d}\varepsilon}\Big|_{\varepsilon=0} &\int_{t_0}^{t_f} F(y(t) + \varepsilon\,\eta(t), \dot{y}(t) + \varepsilon\,\dot{\eta}(t), t)\,\mathrm{d}t \\ + \varepsilon \frac{\mathrm{d}}{\mathrm{d}\varepsilon}\Big|_{\varepsilon=0} &\int_{t_f}^{t_f+\varepsilon\,\xi} F(y(t) + \varepsilon\,\eta(t), \dot{y}(t) + \varepsilon\,\dot{\eta}(t), t)\,\mathrm{d}t. \end{aligned} \tag{2.8}$$

Since the integration limits of the first integral do not depend on ε, we can simply shift the derivative with respect to ε under the integral resulting in

$$\delta J = \varepsilon \int_{t_0}^{t_f} \left[\frac{\partial F}{\partial y}\eta + \frac{\partial F}{\partial \dot{y}}\dot{\eta}\right] \mathrm{d}t + \varepsilon \frac{\mathrm{d}}{\mathrm{d}\varepsilon}\Big|_{\varepsilon=0} \int_{t_f}^{t_f+\varepsilon\,\xi} F(y(t) + \varepsilon\,\eta(t), \dot{y}(t) + \varepsilon\,\dot{\eta}(t), t)\,\mathrm{d}t. \tag{2.9}$$

The second integral of Eq. (2.9) deserves a closer look and can be rewritten by using the Leibniz integration rule. In integral calculus, the Leibniz rule is a way to draw a derivative into the integral, while the integration limits depend on the variable to be differentiated.

TU Bibliothek WIEN Your knowledge hub
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

An integration by the Leibniz rule results in

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\varepsilon}\int_{t_f}^{t_f+\varepsilon\xi} & F(y(t)+\varepsilon\,\eta(t),\dot{y}(t)+\varepsilon\,\dot{\eta}(t),t)\,\mathrm{d}t \\ &= \int_{t_f}^{t_f+\varepsilon\xi}\frac{\mathrm{d}}{\mathrm{d}\varepsilon}F(y(t)+\varepsilon\,\eta(t),\dot{y}(t)+\varepsilon\,\dot{\eta}(t),t)\,\mathrm{d}t + F\Big|_{t_f+\varepsilon\xi}\frac{\mathrm{d}}{\mathrm{d}\varepsilon}\Big(t_f+\varepsilon\xi\Big). \end{aligned} \tag{2.10}$$

Substituting the second integral in Eq. (2.9) by Eq. (2.10), we obtain

$$\delta J = \varepsilon\int_{t_0}^{t_f}\left[\frac{\partial F}{\partial y}\eta+\frac{\partial F}{\partial \dot{y}}\dot{\eta}\right]\mathrm{d}t + \varepsilon\,F(y(t_f),\dot{y}(t_f),t_f)\,\xi, \tag{2.11}$$

in which we evaluated the derivative with respect to ε at $\varepsilon = 0$, causing the integral term in Eq. (2.10) to disappear. After applying an integration by parts of the term including $\dot{\eta}(t)$ in Eq. (2.11), we obtain

$$\delta J = \int_{t_0}^{t_f}\left[\frac{\partial F}{\partial y}\eta - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial F}{\partial \dot{y}}\right)\eta\right]\varepsilon\mathrm{d}t + \varepsilon\frac{\partial F}{\partial \dot{y}}\bigg|_{t_0}^{t_f} + F(y(t_f),\dot{y}(t_f),t_f)\,\varepsilon\xi. \tag{2.12}$$

Note that the boundary term at $t = t_0$ vanishes since the initial condition is not varied. If we now replace ε, η and ξ by the original definitions $\delta y = \varepsilon\eta$ and $\delta t_f = \varepsilon\xi$, introduced in Eq. (2.2) and Eq. (2.3), we end up with

$$\delta J = \int_{t_0}^{t_f}\left[\frac{\partial F}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial F}{\partial \dot{y}}\right)\right]\delta y\,\mathrm{d}t + \frac{\partial F}{\partial \dot{y}}\bigg|_{t_f}\delta y(t_f) + F(y(t_f),\dot{y}(t_f),t_f)\,\delta t_f. \tag{2.13}$$

Notice that δJ depends linearly on the variations $\delta y(t)$, $\delta y(t_f)$ and δt_f. If we are looking for a minimizing/maximizing function $y(t)$, δJ must be zero for any variation $\delta y(t)$, $\delta y(t_f)$ and δt_f. Hence, from Eq. (2.13) we finally obtain the Euler-Lagrange equation

$$\frac{\partial F}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial F}{\partial \dot{y}}\right) = 0, \tag{2.14}$$

with associated boundary terms:

$$\frac{\partial F}{\partial \dot{y}}\bigg|_{t_f} = 0, \qquad F(y(t_f),\dot{y}(t_f),t_f) = 0. \tag{2.15}$$

The conditions mentioned are necessary conditions for a minimizing/maximizing solution. Using the second variation it can be shown that

$$\delta^2 J = \frac{\varepsilon^2}{2}\frac{\mathrm{d}^2}{\mathrm{d}\varepsilon^2}\bigg|_{\varepsilon=0} J\big(y(t)+\varepsilon\,\eta(t)\big) \geq 0, \tag{2.16}$$

which is a necessary 2nd order condition for a local minimum. The derivation is analogous to Eqs. (2.6–2.13), but leads to lengthy expressions, which are not shown here. An elaborate derivation is given in [71, Sec. 10.3].

2.2.1 Boundary Conditions

In this section, we want to examine the boundary terms of δJ in Eq. (2.13) or, respectively, in Eq. (2.15). In general, the variation of the final time and of the final states are not independent [39, Sec. 4.3]. The relation between δt_f, $\delta y(t_f)$ and δy_f is illustrated in Fig. 2.1. It should be noted here that the total variation δy_f takes into account the additional influence of the time variation δt_f on the change of the final function value. The solid line shows the function $y(t)$ ending at t_f, while the dashed line shows a modified function $y^*(t)$ terminating at $t_f + \delta t_f$. Up to first order, the variation of the final values is given by

$$\begin{aligned}\delta y_f &= y^*(t_f + \delta t_f) - y(t_f) \\ &= y(t_f + \delta t_f) + \delta y(t_f + \delta t_f) - y(t_f) \\ &= \dot{y}(t_f)\delta t_f + \delta y(t_f).\end{aligned} \tag{2.17}$$

Depending on the problem, different cases can now occur for the boundary terms. In the following, we will take a closer look at two of these cases, as they will also play an important role later on.

Case I: Final Time and Final State Unrelated

If the final time t_f and the final function value $y(t_f)$ are free and independent, the variations in Eq. (2.13) are arbitrary. After rearranging Eq. (2.17) and substituting $\delta y(t_f)$ in Eq. (2.13) results in

$$\delta J = \int_{t_0}^{t_f} \left[\frac{\partial F}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial F}{\partial \dot{y}}\right)\right] \delta y \,\mathrm{d}t + \left.\frac{\partial F}{\partial \dot{y}}\right|_{t_f} \delta y_f + \left.\left(F - \frac{\partial F}{\partial \dot{y}}\,\dot{y}\right)\right|_{t_f} \delta t_f. \tag{2.18}$$

If we now claim both boundary terms in Eq. (2.18) to disappear, the coefficient of δy_f is zero, thus we obtain $F(y(t_f), \dot{y}(t_f), t_f) = 0$ for a free end time, as in Eq. (2.15).

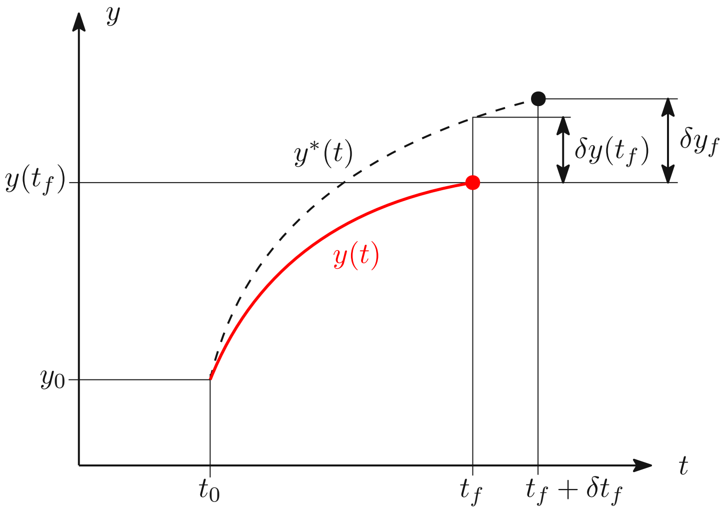

Figure 2.1: Final time variation and the relationship between δt_f, $\delta y(t_f)$ and δy_f

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Case II: Final Time and Final State Related

If the final time t_f and the final function value $y(t_f)$ are free but related, the variations in Eq. (2.13) are not arbitrary. A special type of endpoint constraint occurs when the variation of the endpoint is located on a curve. For example, the final time t_f can be implicitly defined by a scalar equation of the form

$$\Phi(y(t_f)) = y_f, \tag{2.19}$$

in which $y_f \in \mathbb{R}$ is a given number. The operation period with modified function terminates when the condition

$$\Phi(y(t_f) + \delta y_f) = \Phi(y(t_f)) + \Phi_y(y(t_f))\delta y_f = y_f \tag{2.20}$$

is met. Since $\Phi(y(t_f)) = y_f$ and after inserting δy_f from Eq. (2.17), we obtain

$$\Phi_y(y(t_f))\delta y_f = \Phi_y(y(t_f))\Big(\dot{y}(t_f)\delta t_f + \delta y(t_f)\Big) = 0. \tag{2.21}$$

Solving for δt_f we end up with the relation

$$\delta t_f = -\frac{\Phi_y(y(t_f))}{\Phi_y(y(t_f))\dot{y}(t_f)}\delta y(t_f). \tag{2.22}$$

From Eq. (2.22) we observe that the scalar product $\Phi_y\dot{y}$ and the time derivative $\dot{y}(t_f)$ must be nonzero. Hence, the curve defined by Eq. (2.19) must be intersected in cross direction. Substituting δt_f in Eq. (2.13) and collecting all variations results in

$$\delta J = \int_{t_0}^{t_f} \left[\frac{\partial F}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial F}{\partial \dot{y}}\right)\right] \delta y \,\mathrm{d}t + \left(\frac{\partial F}{\partial \dot{y}} - F\,\frac{\Phi_y}{\Phi_y \dot{y}}\right)\Bigg|_{t_f} \delta y(t_f). \tag{2.23}$$

Finally, we have combined both boundary terms from Eq. (2.13) to one single boundary term in Eq. (2.23). As argued before, for an admissible extremal solution $y(t)$ the variation of the functional δJ must disappear, thus the brackets must be zero.

2.2.2 The δ-Operation for a Transformed Time Coordinate

Due to the complexity of many variational problems, an analytical solution is not always possible and in such cases an extremal solution can only be determined numerically. For problems with free end time, an additional difficulty arises because many algorithms require a fixed interval to be applicable. To overcome this problem, on the one hand, a dimensionless unit interval can be introduced. On the other hand, the time coordinate can be transformed to a *"spatial"* variable $s(t)$ with fixed endpoints, when this variable is strictly monotone increasing with respect to the original time coordinate. If this applies, we can carry out a complete elimination of the time coordinate (confer [21] for example), and utilize a formulation in space domain s instead.
However, Fig. 2.2 illustrates that the change of the integration variable has also an impact on the δ-operation describing the function variation.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

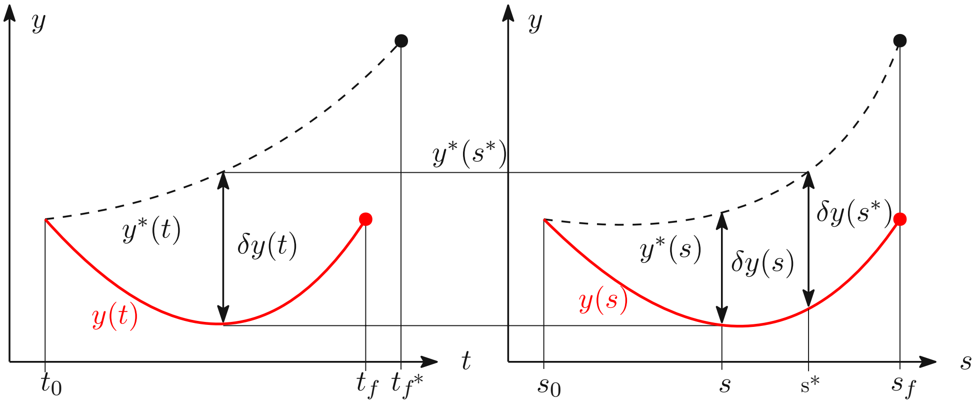

Figure 2.2: The relationship between the variations $\delta y(t)$ and $\delta y(s)$

Let $y^*(t) = y(t) + \delta y(t)$ be again a perturbed function in terms of the original time. Setting $s^*(t) = s(t) + \delta s(t)$, the function variation in terms of the time t reads

$$\begin{aligned}\delta y(t) &= y^*(s^*(t)) - y(s(t)) \\ &= y(s^*(t)) + \delta y(s^*(t)) - y(s(t)) \\ &= y(s(t)) + \frac{\mathrm{d}y}{\mathrm{d}s}\delta s(t) + \delta y(s(t)) - y(s(t)) \\ &= \frac{\mathrm{d}y}{\mathrm{d}s}\delta s(t) + \delta y(s(t)),\end{aligned} \tag{2.24}$$

in which we expanded $y(s^*(t)) = y(s(t) + \delta s(t))$ and $\delta y(s(t) + \delta s(t))$ about s in a linear Taylor series. We will come back to this relation in Chapter 5.

2.3 The Brachistochrone Problem

In order to demonstrate the application of some variational techniques, we discuss the Brachistochrone problem in Fig. 2.3. As history has shown, the solution of this problem plays an essential role in the progress of the calculus of variations. Hence, this example is often discussed in connection with the calculus of variations, and the solution is presented in many standard books.

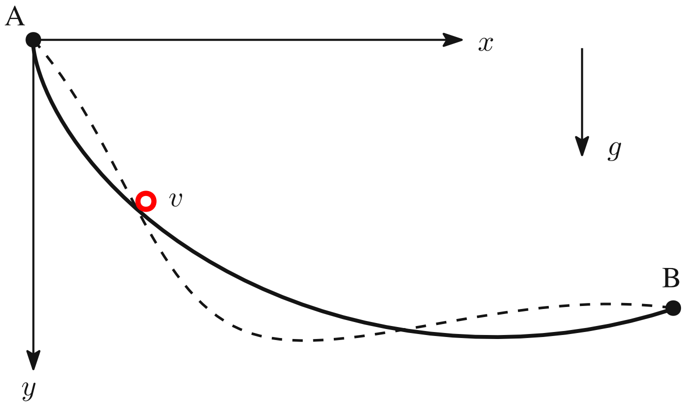

Figure 2.3: Brachistochrone Problem

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

The Brachistochrone, as e. g. discussed in [16, 27, 39], is a frictionless wire between an initial point A and an end point B, on which a point mass slides in shortest time to the end point under the influence of gravity. The goal is to find a function y which minimizes the functional

$$J = \int_{t_0}^{t_f} 1 \mathrm{d}t = \int_0^L \frac{\mathrm{d}l}{v}, \tag{2.25}$$

where J simply measures the length of the time interval $t_f - t_0$. Here, we transformed the integral from time t into the arc length coordinate l of the curve by using the relation $\mathrm{d}t = \mathrm{d}l/v$, where v is the velocity along the path and L is the total length of the curve. Without friction, the system is conservative, hence we get

$$\frac{1}{2}mv^2 = mgy, \quad \text{or,} \quad v = \sqrt{2gy}, \tag{2.26}$$

where y denotes the ordinate position, m the mass of the particle and g the gravitational acceleration. For a frictionless wire we observe that the velocity v is independent of the mass m of the particle. Since we do not know the arc length L of the searched function y a priori, we make another change to the integration variable by:

$$\mathrm{d}l = \sqrt{1 + y'^2(x)}\, \mathrm{d}x. \tag{2.27}$$

After inserting Eq. (2.27) and Eq. (2.26) into Eq. (2.25), we obtain the functional to be minimized for the solution of the Brachistochrone problem.

Now a function $y(x)$ is sought, which satisfies the boundary conditions prescribed by point A and B and minimizes the functional

$$J = \int_{x_0}^{x_f} \sqrt{\frac{1 + y'^2(x)}{2gy(x)}}\, \mathrm{d}x. \tag{2.28}$$

Minimizing Eq. (2.28) gives the curve $y(x)$ of fastest descent under the influence of gravity. We consider the initial point in the origin fixed by $A = (0, 0)$ and the end point fixed by $B = (5, 2)$. In order to apply a solution strategy to our problem, we use Eq. (2.13) and simply replace the time dependence of y by the coordinate x, where $\delta y(t_f) \to \delta y(x_f)$ and $\delta t_f \to \delta x_f$. Both variations vanish due to the boundary conditions $y(x_f) = y_f$ and x_f prescribed. Hence, for a minimizing solution we have to solve

$$\frac{\partial F}{\partial y} - \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{\partial F}{\partial \dot{y}}\right) = 0. \tag{2.29}$$

Since the integrand in Eq. (2.28) does not depend explicitly on x, we can further simplify the differential equation in Eq. (2.29), see also [71, Sec. 2.3.2]. Therefore, we consider the total derivative of $F(y(x), y'(x))$ with respect to x yielding

$$\frac{\mathrm{d}F}{\mathrm{d}x} = \frac{\partial F}{\partial y} y'(x) + \frac{\partial F}{\partial y'} y''(x). \tag{2.30}$$

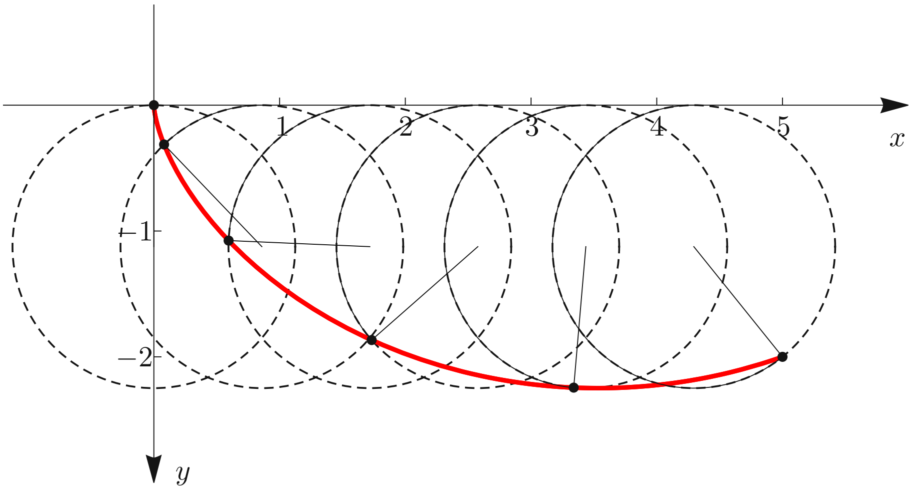

Figure 2.4: Solution of the Brachistochrone problem and visualization of the cycloid generated by a rolling wheel

Multiplying the Euler-Lagrange equation in Eq. (2.29) by $y'(x)$ and using the identity in Eq. (2.30), we obtain

$$0 = \frac{\partial F}{\partial y} y'(x) - \frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{\partial F}{\partial y'}\right) y'(x) = \frac{\mathrm{d}}{\mathrm{d}x}\left(F - \frac{\partial F}{\partial y'} y'(x)\right), \tag{2.31}$$

or after an integration over x:

$$F - \frac{\partial F}{\partial y'} y'(x) = C, \tag{2.32}$$

in which C is an integration constant.

After applying Eq. (2.32) to the Brachistochrone problem in Eq. (2.28), we get

$$\frac{1}{\sqrt{2gy\left(1 + y'^2(x)\right)}} = C, \tag{2.33}$$

or, equivalently,

$$y\left(1 + y'^2(x)\right) = D, \tag{2.34}$$

which is a separable differential equation for $y(x)$. Here, $D = 1/\left(2gC^2\right)$ is a new positive constant to be determined. The solution of Eq. (2.34) is given by a parametric curve of a cycloid:

$$x(\varphi) = \frac{1}{2}D\left(\varphi - \sin(\varphi)\right), \quad y(\varphi) = \frac{1}{2}D\left(1 - \cos(\varphi)\right). \tag{2.35}$$

If we now insert the final point $B = (x(\varphi_f), y(\varphi_f)) = (5, 2)$ into Eq. (2.35) and solve for the constant D and the final value φ_f, the solution is given by $D = 2.2487$ and $\varphi(t) \in [0, 3.820]$. Finally, the solution of the Brachistochrone curve is shown in Fig. 2.4 as a cycloid of a rolling wheel.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Chapter 3

Numerical Optimization

The interest in obtaining optimal solutions has inspired scientists and engineers for centuries. However, even nowadays, armed with computer aided algorithms, the computation of optimal solutions is a challenging task. One way to find a solution is to formulate the problem as an optimization task. In numerical implementation, optimization problems are solved by iterative algorithms, which are proven robust in many fields of natural science.

This chapter is not intended to be a comprehensive presentation of optimization procedures, but rather it focuses on numerical optimization techniques that will later be used for the iterative computation of time-optimal controls. Numerical optimization problems (confer [48, 44]) can be basically divided into static and dynamic optimization problems. In a static optimization problem, the optimization variables are elements of an n-dimensional Euclidean space, while in a dynamic optimization problem, the optimization variables are elements of an infinite dimensional space, such as a time function. If the latter case involves differential equations, it is an optimal-control optimization problem. In the following, we consider optimization problems in the Euclidean space, which we later extend to dynamic optimization problems.

3.1 Optimization without Constraints

Let us begin our discussion of optimization methods by considering the static unconstrained optimization problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}), \tag{3.1}$$

in which $\boldsymbol{x} \in \mathbb{R}^n$ denotes a vector of optimization variables and f the objective function. The goal now is to find the variables $\boldsymbol{x}^*$ for which the function f assumes a minimum. In many cases an analytical solution for $\boldsymbol{x}^*$ is not possible and iterative methods like the gradient or Newton method have to be utilized to find a minimum of f. Starting from a point $\boldsymbol{x}_k$, one tries to find a better point $\boldsymbol{x}_{k+1}$ such that $f(\boldsymbol{x}_{k+1}) < f(\boldsymbol{x}_k)$. The process terminates when a sufficient criterion for a minimum is met. In order to obtain a better iteration point, approaching the minimum of f, we choose a descent direction $\boldsymbol{p}_k \in \mathbb{R}^n$ at point $\boldsymbol{x}_k$ for which the objective function is locally

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

decreasing, which means that the directional derivative $\mathrm{d}/\mathrm{d}\kappa$ evaluated at $\kappa = 0$ is negative, i. e.

$$\frac{\mathrm{d}}{\mathrm{d}\kappa}\bigg|_{\kappa=0} f(\boldsymbol{x}_k + \kappa \boldsymbol{p}_k) = \nabla f^{\mathsf{T}}(\boldsymbol{x}_k)\boldsymbol{p}_k < 0. \tag{3.2}$$

Now the objective function f is traced along the straight line $\boldsymbol{x}(\kappa) = \boldsymbol{x}_k + \kappa \boldsymbol{p}_k$ and we look for a suitable parameter $\kappa > 0$ so that $f(\boldsymbol{x}_k + \kappa \boldsymbol{p}_k) < f(\boldsymbol{x}_k)$ applies. An update formula is then given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \kappa\, \boldsymbol{p}_k. \tag{3.3}$$

In summary, an optimization algorithm involves two main steps: the computation of a search direction $\boldsymbol{p}_k$ (line search method) and the determination of a suitable update step size κ. In the following we will take a closer look at both tasks.

3.1.1 Search Directions

There are numerous methods for determining a search direction $\boldsymbol{p}_k$, which differ in terms of computational effort and convergence speed. In general, a search direction is given by

$$\boldsymbol{p}_k = -\mathbf{H}_k \nabla f(\boldsymbol{x}_k), \tag{3.4}$$

in which $\nabla f(\boldsymbol{x}_k) \in \mathbb{R}^n$ is the gradient of f and $\mathbf{H}_k \in \mathbb{R}^{n\times n}$ is a positive definite matrix, for which Eq. (3.2) is satisfied automatically. Now the question arises how to choose the matrix $\mathbf{H}_k$:

1. The simplest method to determine a search direction, is the gradient or steepest descent method, which uses the search direction along the negative gradient. If we choose the direction of the steepest descent, then $\mathbf{H}_k := \mathbf{I}$ in Eq. (3.4), where $\mathbf{I}$ is an $n \times n$ identity matrix. Hence, the search direction is then given by $\boldsymbol{p}_k = -\nabla f(\boldsymbol{x}_k)$. Note, when a quadratic problem is well conditioned, the negative gradient points in the direction of the minimum $(\boldsymbol{x}^*, f(\boldsymbol{x}^*))$, see Fig. 3.1. In this case, the gradient method converges quickly to the optimal solution. Unfortunately, perfect conditioning requires an eigenvalue calculation, which can become an unpleasant task.

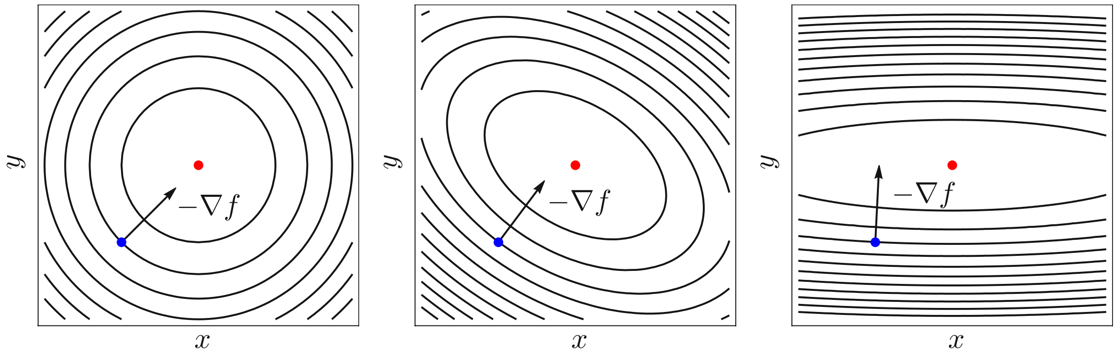

Figure 3.1: Comparison of a good-, moderately good- and ill-conditioned problem

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

2. A more sophisticated search direction is given by the Newton search direction. The idea of Newton's method is to approximate the cost function locally by a quadratic function and minimize this function instead. The Newton direction is given by

$$\boldsymbol{p}_k = -\left(\nabla^2 f(\boldsymbol{x}_k)\right)^{-1} \nabla f(\boldsymbol{x}_k), \tag{3.5}$$

in which we introduced the matrix $\mathbf{H}_k := \left(\nabla^2 f\left(\boldsymbol{x}_k\right)\right)^{-1}$ and assume that $\nabla^2 f(\boldsymbol{x}_k)$ is positive definite. Figure 3.2 shows a comparison between the gradient- and the Newton-direction, where the gradient direction is depicted in blue and the Newton direction is plotted in red.

3. Unfortunately, in many cases, the computation of the Hessian matrix is cumbersome and time-consuming. An alternative method, which is less expensive and also exhibits a good convergence rate, is the quasi-Newton method. The basic idea of the quasi-Newton method is to approximate the Hessian matrix by the gradients of two successive iterations. Therefore, we apply a linear Taylor expansion of the gradient at $\boldsymbol{x}_{k+1}$ yielding

$$\begin{aligned}
\nabla f(\boldsymbol{x}_{k+1}) &= \nabla f(\boldsymbol{x}_k) + \nabla^2 f(\boldsymbol{x}_{k+1})\,(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) \\
\nabla^2 f(\boldsymbol{x}_{k+1})\,(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k) &= \nabla f(\boldsymbol{x}_{k+1}) - \nabla f(\boldsymbol{x}_k) \\
\mathbf{H}_{k+1}\boldsymbol{s}_k &= \boldsymbol{y}_k,
\end{aligned} \tag{3.6}$$

in which we introduced the abbreviations $\mathbf{H}_{k+1} = \nabla^2 f(\boldsymbol{x}_{k+1})$ for the Hessian, $\boldsymbol{s}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$ for the step length and $\boldsymbol{y}_k = \nabla f(\boldsymbol{x}_{k+1}) - \nabla f(\boldsymbol{x}_k)$ for the difference of two successive gradients. However, this information is not sufficient to determine the search direction, i. e. $\mathbf{H}_{k+1}$, since this is an under-determined problem. Due to the fact that the symmetric matrix $\mathbf{H}_{k+1}$ is composed of $\frac{1}{2}n(n+1)$ elements, but Eq. $(3.6)_3$ provides only n equations, additional constraints have to be typically imposed on $\mathbf{H}_k$, e. g., minimum difference between $\mathbf{H}_k$ and $\mathbf{H}_{k+1}$.

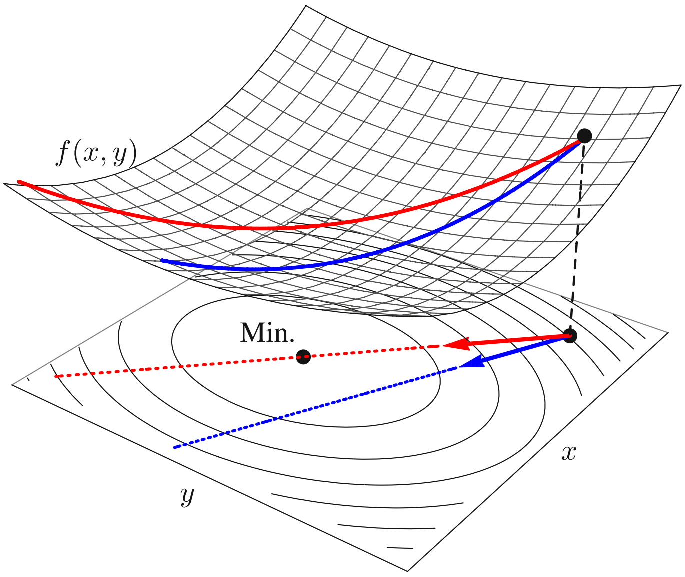

Figure 3.2: Comparison of the gradient- and Newton-direction for a quadratic function

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

A common method for approximating the Hessian matrix is the BFGS-method, independently developed by Broyden, Fletcher, Goldfarb and Shanno in 1970. It defines $\mathbf{H}_{k+1}$ by the optimization problem:

$$\mathbf{H}_{k+1} = \underset{\mathbf{H}}{\operatorname{argmin}} \|\mathbf{H} - \mathbf{H}_k\|, \tag{3.7}$$

where $\|\cdot\|$ represents the Frobenius norm, subject to the conditions

$$\mathbf{H} = \mathbf{H}^\mathsf{T}, \quad \boldsymbol{s}_k = \mathbf{H}\,\boldsymbol{y}_k. \tag{3.8}$$

The analytical solution of Eq. (3.7) and Eq. (3.8) reads

$$\mathbf{H}_{k+1} = \left(\mathbf{I} - \rho_k \boldsymbol{s}_k \boldsymbol{y}_k^\mathsf{T}\right) \mathbf{H}_k \left(\mathbf{I} - \rho_k \boldsymbol{s}_k \boldsymbol{y}_k^\mathsf{T}\right) + \rho_k \boldsymbol{s}_k \boldsymbol{s}_k^\mathsf{T}, \tag{3.9}$$

with $\rho_k := 1/(\boldsymbol{y}_k^\mathsf{T} \boldsymbol{s}_k)$. The quasi-Newton method can be initiated, e. g., by computing the exact Hessian, if this is possible, by approximating the Hessian using numerical differentiation, or by using the identity matrix. The latter mentioned approach corresponds to a gradient update step. A comprehensive derivation of the BFGS-method can be found in many standard optimization textbooks, e. g., in [48, Sec. 8.1].

3.1.2 Step Length Determination

Step length algorithms are iterative. In the literature, two strategies are distinguished, the *imprecise line search* and *exact line search*. We will consider the latter case, which is default in MATLAB's optimization toolbox and is frequently used in many other software packages. Once we have found a suitable search direction $\boldsymbol{p}_k$, the question is how far we should walk along this direction so that the objective function

$$\phi(\kappa) := f(\boldsymbol{x}_k + \kappa \boldsymbol{p}_k), \tag{3.10}$$

decreases. Classical step length algorithms are initiated by estimating an interval $[\kappa_a, \kappa_b]$ in which a minimum is located. The interval can be found, for example, by starting with a sufficiently small κ_a and by successively increasing κ_b until the function value $f(\kappa_b)$ starts to increase. Then, the interval is reduced step by step. The choice of a step size always represents a compromise between accuracy and computational effort. Thus, it is often more convenient to make as few function calls as possible and to avoid additional gradient computations along the search direction. A computationally efficient and easy to implement method for determining a suitable step size is to put a quadratic interpolation polynomial through a V-shaped assembly of three points $(\kappa_1, \phi(\kappa_1))$, $(\kappa_2, \phi(\kappa_2))$ and $(\kappa_3, \phi(\kappa_3))$. A suitable interpolation polynomial is readily given by

$$\phi(\kappa) \approx p(\kappa) = a\kappa^2 + b\kappa + c. \tag{3.11}$$

where a, b and c denote the polynomial coefficients. The determination of a step length using a polynomial is demonstrated on the *Styblinski-Tang function* [68] and is shown in Fig. 3.3.

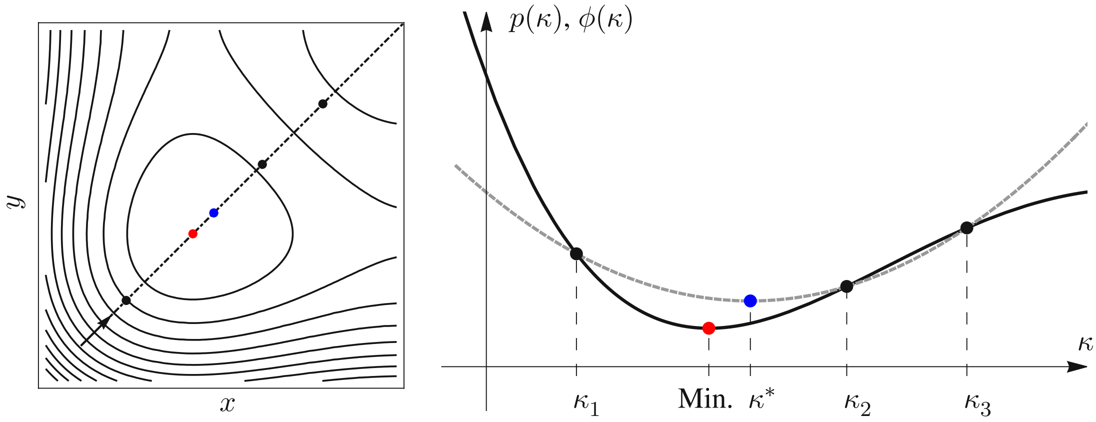

Figure 3.3: Step size determination demonstrated on the *Styblinski-Tang function*

Hence, the minimum of the quadratic polynomial in Eq. (3.11) can be simply determined by

$$\kappa^* = \frac{1}{2} \frac{\left(\kappa_3^2 - \kappa_2^2\right)\phi_1 + \left(\kappa_1^2 - \kappa_3^2\right)\phi_2 + \left(\kappa_2^2 - \kappa_1^2\right)\phi_3}{\left(\kappa_3 - \kappa_2\right)\phi_1 + \left(\kappa_1 - \kappa_3\right)\phi_2 + \left(\kappa_2 - \kappa_1\right)\phi_3} \tag{3.12}$$

in which we used the abbreviations $\phi_1 = \phi(\kappa_1)$, $\phi_2 = \phi(\kappa_2)$ and $\phi_3 = \phi(\kappa_3)$. The process can be repeated by selecting three closer V-shaped points from $(\kappa_1, \phi(\kappa_1))$, $(\kappa_2, \phi(\kappa_2))$, $(\kappa_3, \phi(\kappa_3))$ and $(\kappa^*, \phi(\kappa^*))$. There are many other methods for determining the step size, such as using higher order interpolation polynomials or using an interval nesting method by applying the golden ratio, see [48, Sec. 3.4] for more details.

3.2 Optimization with Constraints

In many optimization problems, it is necessary to satisfy constraint conditions, e. g. if we think back to the variation task with given end conditions. We consider the static optimization problem

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} f(\boldsymbol{x}), \tag{3.13}$$

subjected to the scalar auxiliary condition

$$g(\boldsymbol{x}) = 0. \tag{3.14}$$

On the one hand, the function $f(\boldsymbol{x})$ should be minimized and on the other hand, the variables $\boldsymbol{x}$ must fulfill the auxiliary condition $g(\boldsymbol{x}) = 0$. In this section, we present two efficient optimization algorithms to solve such problems. Generalizations for more than one auxiliary condition can be found easily.

3.2.1 Projected Gradient Method

The key idea of the projected gradient method (according to [58], or in detail [44, Sec. 12.4]) is to compute a search direction along the constraint manifold in Eq. (3.14). For this purpose, the

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

negative gradient of the objective function in Eq. (3.13) is projected onto the tangent space of the constraint manifold and this new direction is then used to update the optimization variables $\boldsymbol{x}_k$. Therefore, the negative gradient is divided into a tangential $\boldsymbol{t}(\boldsymbol{x}_k) \in \mathbb{R}^n$ and a normal part $\eta \nabla g(\boldsymbol{x}_k)$ with respect to the constraint manifold, which is illustrated in Fig. 3.4. Hence, we can write the negative gradient as a linear combination with

$$-\nabla f(\boldsymbol{x}_k) := \boldsymbol{t}(\boldsymbol{x}_k) + \eta \nabla g(\boldsymbol{x}_k), \tag{3.15}$$

where η is a proper scaling factor defining the step length perpendicular to the constraint manifold. If we multiply Eq. (3.15) from the left side with $\nabla g^{\mathsf{T}}(\boldsymbol{x}_k)$ we obtain

$$-\nabla g^{\mathsf{T}}(\boldsymbol{x}_k)\nabla f(\boldsymbol{x}_k) = \eta \nabla g^{\mathsf{T}}(\boldsymbol{x}_k)\nabla g(\boldsymbol{x}_k), \tag{3.16}$$

in which we used $\nabla g^{\mathsf{T}}(\boldsymbol{x}_k) \cdot \boldsymbol{t}(\boldsymbol{x}_k) = 0$. Solving Eq. (3.16) for the step length η, we get

$$\eta = -\frac{\nabla g^{\mathsf{T}}(\boldsymbol{x}_k)\nabla f(\boldsymbol{x}_k)}{\nabla g^{\mathsf{T}}(\boldsymbol{x}_k)\nabla g(\boldsymbol{x}_k)}. \tag{3.17}$$

If we insert η from Eq. (3.17) into Eq. (3.15) and rearrange to $\boldsymbol{t}(\boldsymbol{x}_k)$, we obtain

$$\boldsymbol{t}(\boldsymbol{x}_k) = -\left(\mathbf{I} - \frac{\nabla g(\boldsymbol{x}_k)\nabla g^{\mathsf{T}}(\boldsymbol{x}_k)}{\nabla g^{\mathsf{T}}(\boldsymbol{x}_k)\nabla g(\boldsymbol{x}_k)}\right)\nabla f(\boldsymbol{x}_k), \tag{3.18}$$

where $\mathbf{I}$ is an $n \times n$ identity matrix. Introducing the projection matrix

$$\mathbf{T}(\boldsymbol{x}_k) = \mathbf{I} - \frac{\nabla g(\boldsymbol{x}_k)\nabla g^{\mathsf{T}}(\boldsymbol{x}_k)}{\nabla g^{\mathsf{T}}(\boldsymbol{x}_k)\nabla g(\boldsymbol{x}_k)}, \tag{3.19}$$

an update tangential to the constraint manifold is given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \kappa\, \mathbf{T}(\boldsymbol{x}_k)\nabla f(\boldsymbol{x}_k), \tag{3.20}$$

where $\boldsymbol{p}_k := -\mathbf{T}(\boldsymbol{x}_k)\nabla f(\boldsymbol{x}_k)$ is here a tangential search direction regarding the constraint condition. In the case of a linear constraint equation $g(\boldsymbol{x}_k)$, the update step by the projected gradient in

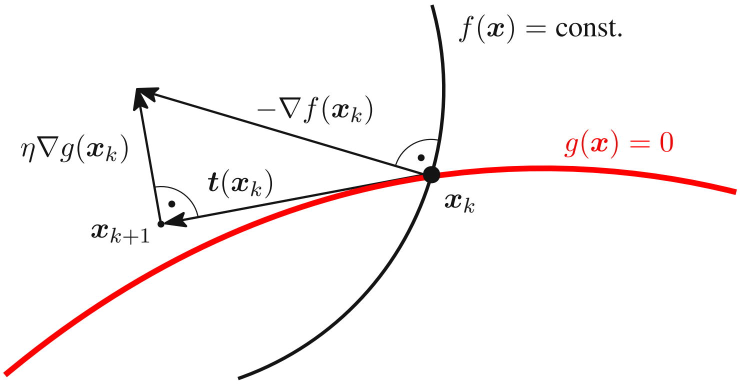
$f(\boldsymbol{x}) = \text{const.}$
$-\nabla f(\boldsymbol{x}_k)$
$\eta\nabla g(\boldsymbol{x}_k)$
$g(\boldsymbol{x}) = 0$
$\boldsymbol{t}(\boldsymbol{x}_k)$
$\boldsymbol{x}_k$
$\boldsymbol{x}_{k+1}$

Figure 3.4: Update step of the projected gradient method (confer [44, Sec. 12.4])

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

Eq. (3.20) does not lead to a deviation from the constraint manifold. However, in many problems there are nonlinear constraints and the projected search direction leads to a violation of the constraint conditions even when choosing small step sizes κ. Hence, after an update step tangential to the constraint manifold a correction step must be applied. The corrector step is a search direction which is orthogonal to the projected search direction to get back to the constraint manifold. An iteration rule is given by

$$\boldsymbol{x}^*_{k+1}(\varepsilon) = \boldsymbol{x}_{k+1} - \varepsilon \nabla g(\boldsymbol{x}_k), \tag{3.21}$$

where $\varepsilon > 0$ is the update step size of the corrector step. If we claim that $g(\boldsymbol{x}^*_{k+1}) = 0$, the update step in Eq. (3.20) is small and we can linearize for ε, resulting in

$$\begin{aligned} 0 = g(\boldsymbol{x}^*_{k+1}(\varepsilon)) &\approx g(\boldsymbol{x}^*_{k+1}(0)) + \left[\frac{\partial g}{\partial \boldsymbol{x}^*_{k+1}} \frac{\mathrm{d}\boldsymbol{x}^*_{k+1}}{\mathrm{d}\varepsilon}\right]\bigg|_{\varepsilon=0} \varepsilon \\ &\approx g(\boldsymbol{x}_{k+1}) - \frac{\partial g}{\partial \boldsymbol{x}_{k+1}} \nabla g(\boldsymbol{x}_k)\varepsilon. \end{aligned} \tag{3.22}$$

If $\boldsymbol{x}_{k+1} \approx \boldsymbol{x}_k$, then we can replace $\partial g / \partial \boldsymbol{x}_{k+1}$ by $\nabla g^\mathsf{T}(\boldsymbol{x}_k)$ in Eq. (3.22) and after solving for ε, we obtain

$$\varepsilon = \frac{g(\boldsymbol{x}_k)}{\nabla g^\mathsf{T}(\boldsymbol{x}_k) \nabla g(\boldsymbol{x}_k)}. \tag{3.23}$$

Substituting ε in Eq. (3.21) we get

$$\boldsymbol{x}^*_{k+1} = \boldsymbol{x}_{k+1} - \frac{g(\boldsymbol{x}_{k+1})}{\nabla g^\mathsf{T}(\boldsymbol{x}_k) \nabla g(\boldsymbol{x}_k)} \nabla g(\boldsymbol{x}_k). \tag{3.24}$$

After applying Eq. (3.24) to remove the residual in the constraint, we set $\boldsymbol{x}_{k+1} := \boldsymbol{x}^*_{k+1}$. Basically, each iteration of the projected gradient method involves two steps:

1. Projection of the negative gradient by Eq. (3.20) and selection of an (optimal) step size κ by using a line search algorithm from Sec. 3.1.2.
2. Elimination of the residual in Eq. (3.14) by utilizing Eq. (3.24). Note that this step may need to be repeated iteratively until the residual in the constraint is sufficiently small.

If the residual in the constraint equation can not be eliminated in the second step, then the update step size κ must be reduced in the first step.

3.2.2 A Modified Gradient Method

In this section we now introduce a handsome modification of the projected gradient approach, as presented in [22], for solving constrained optimization problems, where the initial point $\boldsymbol{x}_0$ does not have to satisfy the constraint condition in Eq. (3.14). Finding a vector that satisfies the constraint at the beginning is often not an easy task. Hence, we assume that we do not know a state vector $\boldsymbol{x}_0 \in \mathbb{R}^n$ which satisfies this condition a priori. However, the variations of the cost

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

function f in Eq. (3.13) and the auxiliary condition g in Eq. (3.14) are given by

$$\begin{aligned}\delta f &= \nabla f^{\mathsf{T}} \delta \boldsymbol{x} \\ \delta g &= \nabla g^{\mathsf{T}} \delta \boldsymbol{x}.\end{aligned} \tag{3.25}$$

As a descent direction, approaching the constrained minimum of f, we now introduce a linear combination of the two gradients ∇f and ∇g:

$$\delta \boldsymbol{x} := -\kappa \left(\nabla f + \nu \nabla g\right). \tag{3.26}$$

For the determination of the multiplier ν, we claim that the descent direction should always point to the hypersurface $g(\boldsymbol{x}) = 0$ by setting

$$\delta g := -\varepsilon\, g(\boldsymbol{x}), \quad \text{where} \quad \varepsilon > 0. \tag{3.27}$$

In this case, δg, i. e. the change of g, will be positive if $g < 0$ and negative if $g > 0$. Inserting Eq. (3.26) into Eq. (3.25)$_2$ and equating the resulting term with Eq. (3.27) yields

$$-\,\kappa \nabla g^{\mathsf{T}} \left(\nabla f + \nu \nabla g\right) = -\varepsilon\, g(\boldsymbol{x}). \tag{3.28}$$

From this, ν can be computed as

$$\nu = \frac{1}{\nabla g^{\mathsf{T}} \nabla g} \left(\frac{\varepsilon}{\kappa} g(\boldsymbol{x}) - \nabla g^{\mathsf{T}} \nabla f\right). \tag{3.29}$$

Hence, the combined search direction (cf. Eq. (3.26)) is given by

$$\delta \boldsymbol{x} = -\kappa \left(\nabla f + \frac{\varepsilon g(\boldsymbol{x}) - \kappa \nabla g^{\mathsf{T}} \nabla f}{\kappa \nabla g^{\mathsf{T}} \nabla g} \nabla g\right) = -\kappa \underbrace{\left(\mathbf{I} - \frac{\nabla g \nabla g^{\mathsf{T}}}{\nabla g^{\mathsf{T}} \nabla g}\right)}_{=:\mathbf{T}(\boldsymbol{x})} \nabla f - \varepsilon \frac{g}{\nabla g^{\mathsf{T}} \nabla g} \nabla g. \tag{3.30}$$

Here, $\mathbf{I}$ is an $n \times n$ identity matrix and $\mathbf{T}(\boldsymbol{x})$ corresponds to the projection matrix of the projected gradient method introduced in Eq. (3.19) in Sec. 3.2.1. However, note that the projected gradient method is restricted to problems which assume the constraint g to be fulfilled at the beginning. In summary, the projected gradient algorithm involves two steps. Recall, in a first step, the negative gradient is projected, which causes, in general, to leave the manifold of the constraint. In a second step, an iteration rule similar to the second summand on the right side of Eq. (3.30) is applied to remove the residual in the constraint equation, confer Eq. (3.24).

A comparison shows that the introduced method corresponds to an incomplete iteration of the projected gradient method, since both steps are carried out simultaneously, and therefore the constraint is not forced to be fulfilled in every iteration step, but the error is reduced continually. Hence, this iteration enables to start the optimization at a point which is not lying on the constraint manifold at the beginning.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Convergence of the Algorithm

1. We first show that if $\boldsymbol{x}^*$ is an optimizing point, the update step from Eq. (3.30) is zero. In the constrained optimum, the following conditions must be satisfied

$$g(\boldsymbol{x}^*) = 0 \quad \text{and} \quad \nabla f(\boldsymbol{x}^*) = \lambda \nabla g(\boldsymbol{x}^*), \tag{3.31}$$

where λ is a Lagrange multiplier. The latter condition expresses that the two gradients are parallel. Evaluating Eq. (3.30) with the optimizing point $\boldsymbol{x}^*$ yields

$$\begin{aligned} \delta\boldsymbol{x} &= -\kappa \left(\mathbf{I} - \frac{\nabla g(\boldsymbol{x}^*) \nabla g^{\mathsf{T}}(\boldsymbol{x}^*)}{\nabla g^{\mathsf{T}}(\boldsymbol{x}^*) \nabla g(\boldsymbol{x}^*)} \right) \nabla f(\boldsymbol{x}^*) - \varepsilon \frac{g(\boldsymbol{x}^*)}{\nabla g^{\mathsf{T}}(\boldsymbol{x}^*) \nabla g(\boldsymbol{x}^*)} \nabla g(\boldsymbol{x}^*) \\ &= -\kappa \left(\mathbf{I} - \frac{\nabla g(\boldsymbol{x}^*) \nabla g^{\mathsf{T}}(\boldsymbol{x}^*)}{\nabla g^{\mathsf{T}}(\boldsymbol{x}^*) \nabla g(\boldsymbol{x}^*)} \right) \lambda \nabla g(\boldsymbol{x}^*) = \mathbf{0}. \end{aligned} \tag{3.32}$$

2. The choice of a suitable update parameter ε depends on the problem, i. e. how much the initial state violates the constraint condition $g(\boldsymbol{x}^*) = 0$.
However, we can make a statement about the value range of ε. For that purpose, we consider an update step from $\boldsymbol{x}_k$ to $\boldsymbol{x}_{k+1}$. Using the update formula in Eq. (3.30) the relation between the point $\boldsymbol{x}_k$ and $\boldsymbol{x}_{k+1}$ is given by

$$\boldsymbol{x}_{k+1} - \boldsymbol{x}_k = -\kappa\, \mathbf{T}(\boldsymbol{x}_k) \nabla f(\boldsymbol{x}_k) - \varepsilon\, \frac{g(\boldsymbol{x}_k)}{\nabla g^{\mathsf{T}}(\boldsymbol{x}_k) \nabla g(\boldsymbol{x}_k)} \nabla g(\boldsymbol{x}_k). \tag{3.33}$$

A linear Taylor expansion of the constraint condition $g(\boldsymbol{x}_k)$ in Eq. (3.14) about the point $\boldsymbol{x}_k$ yields

$$\begin{aligned} g(\boldsymbol{x}_{k+1}) &\approx g(\boldsymbol{x}_k) + \nabla g^{\mathsf{T}}(\boldsymbol{x}_k) \big(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k \big) \\ &\approx g(\boldsymbol{x}_k) - \kappa\, \nabla g^{\mathsf{T}}(\boldsymbol{x}_k) \mathbf{T}(\boldsymbol{x}_k) \nabla f(\boldsymbol{x}_k) - \varepsilon\, g(\boldsymbol{x}_k), \end{aligned} \tag{3.34}$$

in which we inserted Eq. (3.33) for the increment $\boldsymbol{x}_{k+1} - \boldsymbol{x}_k$. Using the relation $\nabla g^{\mathsf{T}}(\boldsymbol{x}_k) \cdot \mathbf{T}(\boldsymbol{x}_k) = \mathbf{0}^{\mathsf{T}}$, Eq. (3.34) simplifies to

$$g(\boldsymbol{x}_{k+1}) \approx g(\boldsymbol{x}_k) - \varepsilon\, g(\boldsymbol{x}_k). \tag{3.35}$$

In each step k we claim the error of the final condition to decrease:

$$\big| g(\boldsymbol{x}_{k+1}) \big| < \big| g(\boldsymbol{x}_k) \big|, \tag{3.36}$$

or, respectively,

$$\big| g(\boldsymbol{x}_k)\, (1 - \varepsilon) \big| < \big| g(\boldsymbol{x}_k) \big|. \tag{3.37}$$

Hence,

$$\big| 1 - \varepsilon \big| < 1, \quad \text{for} \quad g(\boldsymbol{x}_k) \neq 0, \tag{3.38}$$

a suitable update parameter ε is therefore given by $0 < \varepsilon < 2$. Notice that Eq. (3.34) is only valid

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

if $\boldsymbol{x}_{k+1} - \boldsymbol{x}_k$ is a sufficiently small update step. According to Eq. (3.33), this can be assumed if, on the one hand, κ is small and, on the other hand, ε or $g(\boldsymbol{x}_k)$ are small. Hence, in this case, the sequence $g(\boldsymbol{x}_k)$ should converge to zero.

3. Using the abbreviations $\nabla g_k = \nabla g(x_k)$ and $\nabla f_k = \nabla f(x_k)$, the change of the objective function after one (sufficiently small) update step from Eq. (3.33) is given by

$$\begin{aligned}
f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}_k) = \\
&= \nabla f^{\mathsf{T}}(\boldsymbol{x}_k)\big(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\big) \\
&= -\kappa \nabla f_k^{\mathsf{T}} \left(\mathbf{I} - \frac{\nabla g_k \nabla g_k^{\mathsf{T}}}{\nabla g_k^{\mathsf{T}} \nabla g_k}\right) \nabla f_k - \varepsilon g(\boldsymbol{x}_k) \frac{\nabla f_k^{\mathsf{T}} \nabla g_k}{\nabla g_k^{\mathsf{T}} \nabla g_k} \\
&= -\kappa \left(\|\nabla f_k\|^2 - \frac{(\nabla f_k^{\mathsf{T}} \nabla g_k)^2}{\|\nabla g_k\|^2}\right) - \varepsilon g(\boldsymbol{x}_k) \frac{\nabla f_k^{\mathsf{T}} \nabla g_k}{\|\nabla g_k\|^2} \\
&= -\kappa \|\nabla f_k\|^2 \left(1 - \frac{(\nabla f_k^{\mathsf{T}} \nabla g_k)^2}{\|\nabla f_k\|^2 \|\nabla g_k\|^2}\right) - \varepsilon g(\boldsymbol{x}_k) \frac{\nabla f_k^{\mathsf{T}} \nabla g_k}{\|\nabla f_k\| \|\nabla g_k\|} \frac{\|\nabla f_k\|}{\|\nabla g_k\|}.
\end{aligned} \tag{3.39}$$

Since

$$\cos(\alpha_k) := \frac{\nabla f_k^{\mathsf{T}} \nabla g_k}{\|\nabla f_k\| \|\nabla g_k\|}, \tag{3.40}$$

defining the angle between the gradients of f and g and by using $1 - \cos^2(\alpha_k) = \sin^2(\alpha_k)$, the change of the objective function can be written in the form

$$f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}_k) = \underbrace{-\kappa \|\nabla f_k\|^2 \sin^2(\alpha_k)}_{\leq 0} - \varepsilon g(\boldsymbol{x}_k) \cos(\alpha_k) \frac{\|\nabla f_k\|}{\|\nabla g_k\|}. \tag{3.41}$$

Hence, as $g(\boldsymbol{x}_k)$ will tend to zero as shown above, $f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}_k) \leq 0$ for large k. So, if $f(\boldsymbol{x})$ is bounded in the neighborhood of the constraint equation $g(\boldsymbol{x}) = 0$, the algorithm should converge to the constrained minimum of f, provided that the step size κ is sufficiently small. Moreover, in the vicinity of the minimizing point, the angle α_k approaches zero and the constraint equation $g(\boldsymbol{x}) = 0$ is satisfied more and more. Hence, $f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}_k)$ will tend to zero for $k \to \infty$.

A Simple Example

As a simple example, we consider the cost function $f(x, y) = 2x^2 + y^2$ and the auxiliary condition $g(x, y) = x - y + 2 = 0$. We start the optimization procedure at the point $x = -3$ and $y = 3$. Choosing $\varepsilon = 0.3$ and $\kappa = 0.2$, an update of x and y, which reduces the cost function and approaches the constraint condition, can be computed from Eq. (3.30), yielding

$$\begin{aligned}
\delta x &= \frac{1}{2}\varepsilon y - 4\kappa x - \frac{1}{2}\varepsilon x + 2\kappa x - \kappa y - \varepsilon \\
\delta y &= \frac{1}{2}\varepsilon x - 2\kappa y - \frac{1}{2}\varepsilon y - 2\kappa x + \kappa y + \varepsilon.
\end{aligned} \tag{3.42}$$

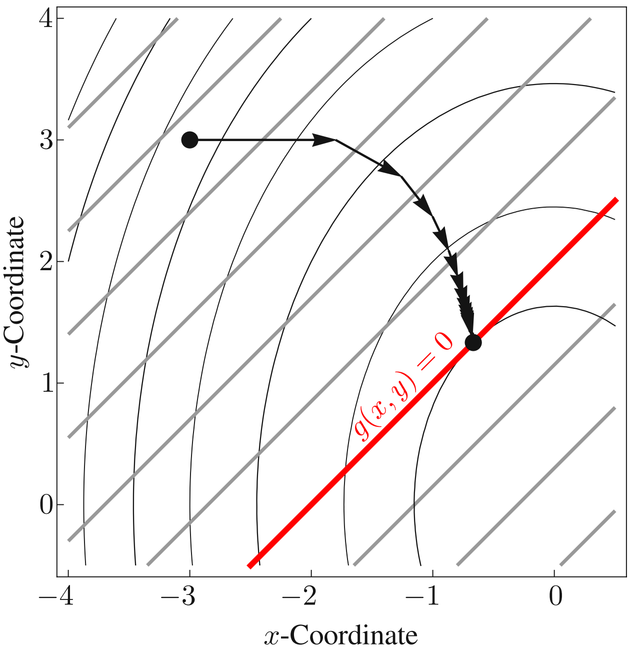

Figure 3.5: Visualization of the optimization procedure with one auxiliary condition

After several updates, the method converges to the constrained optimum as depicted in Fig. 3.5. The contour lines of $g(x, y)$ are shown in gray, while the contour lines of the cost function $f(x, y)$ are depicted in black. Moreover, the contour line $g(x, y) = 0$ is shown in red. At the optimal point, given by $x = -2/3$ and $y = 4/3$, this line is tangent to a contour line of the cost function $f(x, y)$. This simple static optimization problem with an initial state, which is not fulfilling the constraint, shows that the proposed method converges to the constrained optimum.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

Chapter 4

Time-Optimal Control Theory

Among all optimal control problems, the computation of time-optimal trajectories is one of the toughest challenges. In the present chapter we consider a dynamic system described by a minimal set of independent coordinates and given initial values. The focus is on the optimal control of the system such that the time for a prescribed process or maneuver becomes a minimum. In order to solve such problems, two methods are basically pursued in optimal control theory. In the first, the problem can be solved by the minimum principle according to Pontryagin [54], and in the second, the dynamic programming strategy developed by Bellman can be utilized [4]. In this chapter, we follow the solution strategy based on Pontryagin's minimum principle using the variational approach [54, 16], resulting in a two-point boundary value problem. The objective of this chapter is to provide a detailed insight into the basic theory, including discussion on the adjoint equations and the derivation of the necessary conditions for an optimal solution.

4.1 Historical Notes

Optimal control theory is rooted in the calculus of variations and evolved from many disciplines into the theory we know today. There are many articles on the history of optimal control [13, 70, 59]. In the summary of Bryson [13], the beginnings can be traced back to classical control, random processes, linear/nonlinear programming, dynamic programming and the maximum principle.

In 1919, the rocket scientist Robert H. Goddard posed one of the first optimal control problems in aerospace [26], which was solved analytically as a variation problem three decades later. With the introduction of the computer in the 50s, the theory of optimal control made tremendous progress, since computing power was now available for the elaborate computations for the first time. Before that, rather simple and mainly analytical problems could be solved, so that the theory of optimal control was rarely used by scientists and engineers.

In the 1950s, the U.S. mathematician Richard Ernest Bellman and his colleagues developed the dynamic programming algorithm. This method was based on an extension of the Hamilton-Jacobi theory introduced over hundred years earlier by William Rowan Hamilton and Karl Gustav Jacob

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Jacobi. However, Bellman and his collaborators underestimated the expensive computations of the dynamic programming strategy and so the algorithms exceeded the memory capacity of the computers for systems with only a few degrees of freedom.

In 1956, the well-known maximum principle [54] was formulated as a hypothesis by Lev Semenovich Pontryagin and proved in the subsequent years by him and his collaborators in the former Soviet Union. The maximum principle was formulated for constrained controls and is an extension of the necessary condition of Karl Wilhelm Theodor Weierstrass in the 19th century. Pontryagin's maximum principle is based on the Hamiltonian function and states that for an optimal control, the Hamiltonian function must assume a maximum value at every time point.[1] Countless scientific articles followed, and George Leitmann, Richard Bellman and other pioneers, published the first real book on optimal control theory [42] in 1962. Only a couple of years later, the first textbook on optimal control was published by Athans and Falb [2].

In the following years, optimal control theory made further progress with the increasing computing power and the initiation of the Mercury space program in 1958. The first numerical computations of optimal trajectories came from Bryson and Ross [17], Breakwell [12], Okhotsimskii and Eneev [51]. The numerical solutions were obtained by shooting methods, which showed good results for conservative systems in spaceflight, but were not suitable for non-conservative systems owing to numerical instabilities and sensitivities with respect to initial values. To overcome the stability problem, gradient methods were introduced for the initialization of the shooting method, which were developed by Kelley [36], Bryson and Denham [14].

Another remedy of the stability problem was created by the development of the multiple shooting method, in which the computation interval is divided into subsegments. Algorithms of the multiple shooting method were developed in Fortran by Bulirsch [13, 18] and by Oberle [49] at the University of Munich. These algorithms are still state of the art and widely recognized today to solve challenging boundary value problems.

4.2 Problem Formulation

We consider a dynamic system described by a minimal set of coordinates, see also [21]; hence the initial value problem considered, can be stated in the form

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \tag{4.1}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$ denotes the vector of state variables and $\boldsymbol{u}(t) \in \mathbb{R}^m$ the vector of control inputs. In mechanics, the control inputs can either arise in the form of control forces/torques or in the form of kinematic variables. The goal is to find controls which minimize the operation period after which a state with some predefined properties is reached without violating inequality constraints for the state and control variables.

---

[1] Note, originally, the maximum principle was formulated for maximization problems, however, it also applies to minimization problems and is therefore often denoted as minimum principle in literature.

For that purpose a cost functional of the form

$$J = \int_{t_0}^{t_f} \left[1 + \Pi(\boldsymbol{x}(t))\right] \,\mathrm{d}t, \tag{4.2}$$

is introduced, where $\Pi(\boldsymbol{x}(t))$ is a proper penalty function which is zero if the states satisfy the constraint conditions, and increases rapidly if they are not satisfied. Moreover, Π should be continuously differentiable at least once. Unless violating constraints, the cost functional J is simply the length of the time interval $t_f - t_0$. Notice that the controls $\boldsymbol{u}(t) = (u_i(t)) \in \mathbb{R}^m$ can be constrained to satisfy the condition given by $u_{i,\min} \le u_i(t) \le u_{i,\max}$. The time-optimal control problem is then formulated as the problem of finding controls $\boldsymbol{u}(t)$ in the time interval $t \in [t_0, t_f]$ which minimize the functional J.

4.3 The Variational Approach to Time-Optimal Control

Before we account for control limitations stated above, we consider in a first step the solution approach for unbounded controls. Following [39], we first augment the cost functional in Eq. (4.2) by the state equations in Eq. (4.1) yielding

$$\bar{J} = \int_{t_0}^{t_f} \left[1 + \Pi(\boldsymbol{x}) + \boldsymbol{p}^\mathsf{T} \left(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) - \dot{\boldsymbol{x}}\right)\right] \,\mathrm{d}t, \tag{4.3}$$

where $\boldsymbol{p}(t) \in \mathbb{R}^n$ denotes the vector of adjoint variables.[2] If the state equations are satisfied, the term multiplied with $\boldsymbol{p}(t)$ is zero and $\bar{J}$ is identical with J for any choice of $\boldsymbol{p}(t)$. Let us now consider an infinitesimal variation of the controls $\delta\boldsymbol{u}(t)$. This results in an infinitesimal change of the states $\delta\boldsymbol{x}(t)$ and, hence, of the final time δt_f. The first order variation of $\bar{J}$ is given by

$$\begin{aligned}\delta\bar{J} = &\int_{t_0}^{t_f} \left[\Pi_{\boldsymbol{x}}\delta\boldsymbol{x} + \boldsymbol{p}^\mathsf{T} \left(\boldsymbol{f}_{\boldsymbol{x}}\delta\boldsymbol{x} + \boldsymbol{f}_{\boldsymbol{u}}\delta\boldsymbol{u} - \delta\dot{\boldsymbol{x}}\right)\right] \mathrm{d}t \\ &+ \left[1 + \Pi(\boldsymbol{x}(t_f)) + \boldsymbol{p}^\mathsf{T}(t_f) \left(\boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f)) - \dot{\boldsymbol{x}}(t_f)\right)\right] \delta t_f,\end{aligned} \tag{4.4}$$

where $\boldsymbol{f}_{\boldsymbol{x}}$ and $\boldsymbol{f}_{\boldsymbol{u}}$ denote the partial derivatives of $\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t))$ with respect to $\boldsymbol{x}(t)$ and $\boldsymbol{u}(t)$, analogously, the same abbreviation applies to $\Pi_{\boldsymbol{x}}$. Note that a variation of the final time also contributes to a change of the integral but since the equations of motion are fulfilled at $t = t_f$, the variation of the cost functional reduces to

$$\delta\bar{J} = \int_{t_0}^{t_f} \left[\Pi_{\boldsymbol{x}}\delta\boldsymbol{x} + \boldsymbol{p}^\mathsf{T} \left(\boldsymbol{f}_{\boldsymbol{x}}\delta\boldsymbol{x} + \boldsymbol{f}_{\boldsymbol{u}}\delta\boldsymbol{u} - \delta\dot{\boldsymbol{x}}\right)\right] \mathrm{d}t + \left[1 + \Pi(\boldsymbol{x}(t_f))\right] \delta t_f. \tag{4.5}$$

Performing an integration by parts of the last term of the integral $\boldsymbol{p}^\mathsf{T}(t)\delta\dot{\boldsymbol{x}}(t)$ yields

$$\int_{t_0}^{t_f} \boldsymbol{p}^\mathsf{T} \delta\dot{\boldsymbol{x}} \,\mathrm{d}t = -\int_{t_0}^{t_f} \dot{\boldsymbol{p}}^\mathsf{T} \delta\boldsymbol{x} \,\mathrm{d}t + \boldsymbol{p}^\mathsf{T}(t_f)\delta\boldsymbol{x}(t_f), \tag{4.6}$$

---

[2] The adjoint variables are also commonly referred to as co-state variables in mathematics or shadow prices for optimal control tasks in economics.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

where $\delta \boldsymbol{x}(t_0) = \mathbf{0}$ as the initial conditions are fixed. Thus, after substituting into Eq. (4.5), we obtain

$$\begin{aligned} \delta \bar{J} = \int_{t_0}^{t_f} & \left[\boldsymbol{p}^{\mathsf{T}} \boldsymbol{f}_{\boldsymbol{u}} \delta \boldsymbol{u} + \left(\Pi_{\boldsymbol{x}} + \boldsymbol{p}^{\mathsf{T}} \boldsymbol{f}_{\boldsymbol{x}} + \dot{\boldsymbol{p}}^{\mathsf{T}} \right) \delta \boldsymbol{x} \right] \mathrm{d}t \\ & - \boldsymbol{p}^{\mathsf{T}}(t_f) \delta \boldsymbol{x}(t_f) + \left[1 + \Pi(\boldsymbol{x}(t_f)) \right] \delta t_f . \end{aligned} \tag{4.7}$$

If the variation $\delta \bar{J}$ in Eq. (4.7) is assumed to vanish, one can write the first order necessary conditions for an optimal solution:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \quad \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \\ \dot{\boldsymbol{p}}(t) &= -\Pi_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(t)) - \boldsymbol{f}_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \boldsymbol{p}(t) \\ \mathbf{0} &= \quad \boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \boldsymbol{p}(t), \end{aligned} \tag{4.8}$$

for $t \in [t_0, t_f]$. Note that the adjoint variables can be interpreted as a measure of sensitivity, i. e., how much the cost functional would change due to a small state perturbation. The left boundary conditions are defined by the initial conditions of the state equations

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0 . \tag{4.9}$$

For the right boundary conditions, we can first conclude from Eq. (4.7) that

$$0 = -\boldsymbol{p}^{\mathsf{T}}(t_f) \delta \boldsymbol{x}(t_f) + \left[1 + \Pi(\boldsymbol{x}(t_f)) \right] \delta t_f . \tag{4.10}$$

Based on Eq. (4.10), boundary conditions can be deduced, which form a two-point boundary value problem with Eq. (4.8) and Eq. (4.9). This boundary value problem can then be solved, for example, using indirect methods, i. e., shooting methods or with discretization techniques in which both states and controls are discretized. The solution of boundary value problems is discussed by various authors [20, 6, 53, 43]. The resulting boundary conditions depend on the relations of δt_f and $\delta \boldsymbol{x}(t_f)$ which are discussed below.

4.4 Boundary Conditions

Now, special attention is given to the boundary terms of $\delta \bar{J}$ summarized in Eq. (4.10). Notice, for free final time problems the variation δt_f is non-zero, and, since $\Pi \geq 0$, the term $1 + \Pi(\boldsymbol{x}(t_f))$ is non-zero as well.

Depending on the problem, different final conditions result, and appropriate replacements in Eq. (4.10) for $\delta \boldsymbol{x}(t_f)$ and δt_f need to be made. In this section, we consider two cases for the boundary terms, however there are various scenarios that may occur for these boundary terms, see [39, Sec. 5.1] for an overview.

4.4.1 Final State Variables Lying on a Surface

In the first case, the final time $t_f > t_0$ of a maneuver or an operation period may be implicitly defined by a scalar terminal condition for the state variables of the form

$$\Phi(\boldsymbol{x}(t_f)) = s_f, \tag{4.11}$$

where $\Phi(\boldsymbol{x}(t_f))$ is a function $\mathbb{R}^n \to \mathbb{R}$ and s_f a given number. For example, if we consider a car driving one lap on a race track, the operation period terminates, when the center of mass of the vehicle crosses the finish line. Since the final states must satisfy Eq. (4.11), the variation of the final time and of the final states are not independent [40, 39]. To rewrite the boundary terms, we proceed in exactly the same manner as in Sec. 2.2.1. The relation between δt_f and $\delta\boldsymbol{x}(t_f)$ is given in Eq. (2.17). Analogously, the variation of the final states where the operation period ends is given by

$$\delta\boldsymbol{x}_f = \dot{\boldsymbol{x}}(t_f)\delta t_f + \delta\boldsymbol{x}(t_f). \tag{4.12}$$

The operation period with modified controls ends when the condition $\Phi(\boldsymbol{x}(t_f)+\delta\boldsymbol{x}_f) = \Phi(\boldsymbol{x}(t_f))+\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))\delta\boldsymbol{x}_f = s_f$ is satisfied. After inserting $\delta\boldsymbol{x}_f$ from Eq. (4.12), we get

$$\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))\delta\boldsymbol{x}_f = \Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))\left(\dot{\boldsymbol{x}}(t_f)\delta t_f + \delta\boldsymbol{x}(t_f)\right) = 0. \tag{4.13}$$

If we solve for δt_f, we end up with the relation

$$\delta t_f = -\frac{\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))}{\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))\dot{\boldsymbol{x}}(t_f)}\delta\boldsymbol{x}(t_f), \tag{4.14}$$

where $\dot{\boldsymbol{x}}(t_f)$ can be evaluated by the right-hand side $\boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))$ of the differential equations. Substituting δt_f into Eq. (4.10) and collecting all variations results in

$$0 = \left[\boldsymbol{p}^{\mathsf{T}}(t_f) + \frac{1+\Pi(\boldsymbol{x}(t_f))}{\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))\boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))}\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))\right]\delta\boldsymbol{x}(t_f). \tag{4.15}$$

Hence, the right boundary conditions read

$$\boldsymbol{p}(t_f) = -\frac{1+\Pi(\boldsymbol{x}(t_f))}{\boldsymbol{f}^{\mathsf{T}}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))\Phi_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(t_f))}\Phi_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(t_f)). \tag{4.16}$$

4.4.2 Some Final State Variables Specified

Now we are interested in problems where some of the state variables at the final time are prescribed. Such boundary conditions arise in robotics, for example, when the control of a robot has to be designed for a rest-to-rest maneuver. If the prescribed states are ordered forward in the vector $\boldsymbol{x}(t)$, we can assign $i = 1, \ldots, r$ for the prescribed states, and $i = r+1, \ldots, n$ for the remaining states.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Hence, the boundary conditions for the states are defined by

$$x_i(t_f) = x_{f,i} \quad \text{with} \quad i = 1, \dots, r. \tag{4.17}$$

First, we rewrite the right boundary term in Eq. (4.10) by the relation in Eq. (4.12), i. e.

$$\delta \boldsymbol{x}(t_f) = \delta \boldsymbol{x}_f - \boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))\delta t_f, \tag{4.18}$$

in which we inserted again for $\dot{\boldsymbol{x}}(t_f) = \boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))$ yielding

$$0 = -\boldsymbol{p}^\mathsf{T}(t_f)\delta \boldsymbol{x}_f + \left[1 + \Pi(\boldsymbol{x}(t_f)) + \boldsymbol{p}^\mathsf{T} \boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))\right] \delta t_f. \tag{4.19}$$

Now the variation of the final states can be divided into the variations of the prescribed states and the variations of the states which are considered free. Hence, Eq. (4.19) becomes

$$\begin{aligned} 0 = &- \sum_{i=1}^{r} p_i(t_f)\delta x_{f,i} - \sum_{i=r+1}^{n} p_i(t_f)\delta x_{f,i} \\ &+ \left[1 + \Pi(\boldsymbol{x}(t_f)) + \boldsymbol{p}^\mathsf{T}(t_f)\boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))\right] \delta t_f. \end{aligned} \tag{4.20}$$

For the states $i = 1, \dots, r$, which are prescribed at the endpoint, the corresponding variation $\delta x_{f,i}$ vanishes, while for all free final states $i = r + 1, \dots, n$ the variation $\delta x_{f,i}$ is arbitrary and the associated adjoint variable $p_i(t_f)$ is zero. Hence, the boundary conditions can be summarized by

$$\begin{aligned} x_i(t_f) &:= x_{f,i} \quad && i = 1, \dots, r \\ p_i(t_f) &:= 0 && i = r + 1, \dots, n \end{aligned} \tag{4.21}$$

Since the final time is considered free, the remaining variation δt_f is arbitrary, so the bracketed expression in Eq. (4.20) must vanish:

$$0 = 1 + \Pi(\boldsymbol{x}(t_f)) + \boldsymbol{p}^\mathsf{T} \boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f)). \tag{4.22}$$

This equation corresponds to the Hamiltonian function and states that the Hamiltonian function must be zero for time-optimal control problems, but more on the role of the Hamiltonian function in the next section.

4.5 Pontryagin's Minimum Principle

In the derivation of the necessary conditions for an optimal solution, no restrictions have been placed on the control so far. In reality, however, there are usually constraints on control variables, e. g. if we consider the drive torque of a car as the control input. When the control variable is constrained, the third optimality condition in Eq. $(4.8)_3$ loses its validity upon closer inspection. Equation $(4.8)_3$ can be interpreted as gradient of the cost functional in Eq. (4.2) with respect to

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

the control signals and thus, as in classical minimization problems in mathematics, gradient zero can no longer be achieved due to the constraint. Hence, we require a new condition in place of the zero gradient statement in Eq. (4.8)$_3$ which is valid for constrained controls. In order to find optimal solutions for constrained controls, the Russian mathematician L. S. Pontryagin formulated the minimum principle in 1956. In brief, the minimum principle, see also [39, Sec. 5.3], can be summarized as follows. A constrained control is optimal if the following condition holds:

$$\bar{J}(\boldsymbol{u}^*(t)) - \bar{J}(\boldsymbol{u}(t)) = \delta\bar{J}(\boldsymbol{u}(t), \boldsymbol{u}^*(t)) \geq 0, \qquad \forall \text{ admissible } \boldsymbol{u}^*(t). \tag{4.23}$$

In order to write the following formulas in a more compact form, we now introduce the Hamiltonian function with

$$\mathcal{H}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}(t)) := 1 + \Pi(\boldsymbol{x}(t)) + \boldsymbol{p}^{\mathsf{T}}(t)\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)). \tag{4.24}$$

Once the state equations and adjoint equations are solved subject to control limitations, the variations $\delta\boldsymbol{x}(t)$, $\delta\boldsymbol{x}(t_f)$ and δt_f in Eq. (4.7) vanish. Then, we can use Eq. (4.24) to rewrite Eq. (4.7), and thus we obtain

$$\delta\bar{J}(\boldsymbol{u}(t), \delta\boldsymbol{u}(t)) = \int_{t_0}^{t_f} \mathcal{H}_{\boldsymbol{u}}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}(t))\, \delta\boldsymbol{u}(t)\, \mathrm{d}t. \tag{4.25}$$

Now recall, if m denotes the number of control inputs, then $\mathcal{H}_{\boldsymbol{u}}$ can also be written as difference quotient by

$$\mathcal{H}_{u_i}(\boldsymbol{x}(t), u_i(t), \boldsymbol{p}(t)) \;=\; \frac{\mathcal{H}(\boldsymbol{x}(t), u_i(t) + \delta u_i(t), \boldsymbol{p}(t)) - \mathcal{H}(\boldsymbol{x}(t), u_i(t), \boldsymbol{p}(t))}{\delta u_i(t)}, \tag{4.26}$$

for $i = 1, \ldots, m$, or, after rearranging,

$$\mathcal{H}_{\boldsymbol{u}}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}(t))\; \delta\boldsymbol{u}(t) = \mathcal{H}(\boldsymbol{x}(t), \boldsymbol{u}(t) + \delta\mathbf{u}(t), \mathbf{p}(t)) - \mathcal{H}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}(t)). \tag{4.27}$$

Substituting Eq. (4.27) into the varied functional in Eq. (4.25) yields

$$\delta\bar{J}(\boldsymbol{u}(t), \delta\boldsymbol{u}(t)) = \int_{t_0}^{t_f} \Big[\mathcal{H}(\boldsymbol{x}(t), \boldsymbol{u}(t) + \delta\boldsymbol{u}(t), \boldsymbol{p}(t)) - \mathcal{H}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}(t))\Big]\, \mathrm{d}t. \tag{4.28}$$

Since $\delta\bar{J} \geq 0$ for all admissible $\delta\boldsymbol{u}(t)$, as stated in Eq. (4.23), we deduce from Eq. (4.28) that the condition

$$\mathcal{H}(\boldsymbol{x}(t), \boldsymbol{u}(t) + \delta\boldsymbol{u}(t), \boldsymbol{p}(t)) \geq \mathcal{H}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}(t)), \qquad \forall \text{ admissible } \delta\boldsymbol{u}(t), \tag{4.29}$$

must be satisfied for optimal constrained controls. This condition is called Pontryagin's minimum principle, and states that an optimal control must minimize the Hamiltonian. The minimum principle is a necessary condition for an optimal solution and replaces Eq. (4.8)$_3$ for constrained controls. If the control value is within the admissible range $u_{i,\min} < u_i(t) < u_{i,\max}$, the control can be computed according to Eq. (4.8)$_3$.

However, if the control value according to Eq. (4.8)$_3$ falls outside the admissible value range, the maximum or minimum value of the control applies. Note that even for unconstrained control problems, where $\boldsymbol{u}(t)$ cannot be uniquely expressed explicitly by $\boldsymbol{x}(t)$ and $\boldsymbol{p}(t)$ using Eq. (4.8)$_3$, the solution that minimizes the Hamiltonian must be used.

4.6 Homotopy and Continuation Methods

Solving a two-point boundary value problem is not a trivial task, due to the challenge of finding a good initial estimate for all unknown variables $\boldsymbol{x}(t)$, $\boldsymbol{p}(t)$, $\boldsymbol{u}(t)$ and t_f. In particular, finding a suitable initial estimate for the adjoint variables $\boldsymbol{p}(t)$, which causes the boundary value solver to converge can become a cumbersome procedure. The application of homotopy, or more sophisticated continuation methods, can help to overcome this problem, see [1, Sec. 8.3]. Homotopy methods introduce a set of additional parameters λ_i to the system and construct a problem for which a good estimate of the solution is known. A continuation method is an extension to the homotopy approach, where the introduced homotopy parameters are regarded as system variables. In the article by Graichen and Petit [30], a promising homotopy approach is presented based on an auxiliary optimal control problem for which the adjoint variables are simply zero for the entire time interval. Starting from any initial trajectory of the system $\bar{\boldsymbol{x}}(t)$, $\bar{\boldsymbol{u}}(t)$, where $t \in [t_0, \bar{t}_f]$, the modified problem can be smoothly transformed into the real problem. Following [30], the modified cost functional is of the form

$$\bar{J} = (1-\lambda_1)\, S(t_f) + \int_{t_0}^{t_f} \Big[\lambda_1\big(1+\Pi(\boldsymbol{x}(t))\big) + (1-\lambda_1)\, h(\boldsymbol{u}(t))\Big]\, \mathrm{d}t, \tag{4.30}$$

which is equal to Eq. (4.2) for $\lambda_1 = 1$. Herein, $\lambda_1 \in [0,1]$ is a homotopy parameter to transform the problem, $S(t_f)$ is a scrap function and $h(\boldsymbol{u}(t))$ is an additional cost term. The scrap function is defined by

$$S(t_f) := \frac{1}{2}\left(t_f - \bar{t}_f\right)^2, \tag{4.31}$$

where $\bar{t}_f$ denotes the final time of the initial trajectory. Furthermore, the additional cost term is given by

$$h(\boldsymbol{u}(t)) := \frac{1}{2}\sum_{i=1}^{m}\Big(u_i(t) - \bar{u}_i(t)\Big)^2, \tag{4.32}$$

in which $\bar{u}_i$ is the initial control. In analogy to the derivation in Sec. 4.3 of the adjoint system in Eq. (4.8)$_2$, we obtain

$$\dot{\boldsymbol{p}}(t) = -\lambda_1 \Pi_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(t)) - \boldsymbol{f}_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(t), \boldsymbol{u}(t))\boldsymbol{p}(t), \tag{4.33}$$

for the modified cost functional in Eq. (4.30). Furthermore, Eq. (4.8)$_3$ is substituted by

$$\boldsymbol{0} = (1-\lambda_1)\big(\boldsymbol{u}(t) - \bar{\boldsymbol{u}}(t)\big) + \boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}}(\boldsymbol{x}(t), \boldsymbol{u}(t))\boldsymbol{p}(t), \tag{4.34}$$

in which the term multiplied with $(1-\lambda_1)$ acts as a regularization term. For ill-posed, or even singular control tasks, where the control can not be determined from Eq. (4.8)$_3$, a regularization

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

term is often introduced to the cost functional. We will return to the role of singular and bang-bang controls later. Finally, the resulting boundary term associated to the homotopy approach is given by

$$\begin{aligned} 0 = \lambda_1\big[1 + \Pi(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))\big] &+ \boldsymbol{p}^\mathsf{T}(t_f)\boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f)) \\ &+ (1-\lambda_1)\big[h(\boldsymbol{u}(t_f)) + S_{t_f}(t_f)\big]. \end{aligned} \tag{4.35}$$

Again, the prescribed states are ordered forward and numbered by $i = 1, \dots, r$. For all free final states $i = r+1, \dots, n$ the corresponding adjoint variables $p_i(t_f)$ are zero. In order to satisfy the right boundary conditions, a second homotopy parameter $\lambda_2 \in [0, 1]$ is introduced to modify the boundary conditions in Eq. (4.21). Hence,

$$\begin{aligned} x_i(t_f) &:= \lambda_2\, x_{f,i} + (1-\lambda_2)\bar{x}_{f,i} \qquad && i = 1, \dots, r \\ p_i(t_f) &:= 0 && i = r+1, \dots, n, \end{aligned} \tag{4.36}$$

where $\bar{x}_{f,i}$ refers to the final state of the initial trajectory. For $\lambda_1 = 0$ and $\lambda_2 = 0$, the solution of the optimal control problem is a priori given by $\boldsymbol{u}(t) = \bar{\boldsymbol{u}}(t)$, $t_f = \bar{t}_f$, $\boldsymbol{x}(t) = \bar{\boldsymbol{x}}(t)$ and $\boldsymbol{p}(t) = \boldsymbol{0}$, since Eq. (4.33–4.36) and the state equations are satisfied. Subsequently, the problem can be smoothly transformed into the original problem by solving a series of boundary value problems. As already briefly mentioned, the presented method is particularly interesting for singular optimal control tasks where the control occurs linearly in the Hamiltonian. If the control appears linearly, then Eq. $(4.8)_3$ provides no information about the selection of $\boldsymbol{u}(t)$. The additional cost term in Eq. (4.32) circumvents this problem. In contrast to classical regularization terms as e. g. εu^2, ε small, Eq. (4.34) is a clever choice. Since, on the one hand, this approach (widely) avoids difficulties with singular controls, and on the other hand, this formulation serves as a homotopy approach to support the boundary value solver in convergence issues. Hence, we can compute the control variables $\boldsymbol{u}(t)$ from Eq. (4.34) by

$$\boldsymbol{u}(t) = \bar{\boldsymbol{u}}(t) - \frac{1}{1-\lambda_1}\boldsymbol{f}_{\boldsymbol{u}}^\mathsf{T}(\boldsymbol{x}(t), \boldsymbol{u}(t))\boldsymbol{p}(t). \tag{4.37}$$

However, when the quadratic part of the cost function $h(\boldsymbol{u}(t))$ becomes very small, i. e. when $\lambda_1 \to 1$, numerical difficulties may arise. For more information, an elaborate derivation of the auxiliary optimal control problem can be found in the article [30].

The Homotopy Algorithm

Before solving a numerical example, we summarize the steps to compute the optimal control using the homotopy method. Note that the final time is free, but the two-point boundary value solver needs a fixed grid over time. To provide a fixed time mesh, we introduce a normalized time $\tau = t/t_f$ ranging from 0 to 1. With the state vector $\boldsymbol{z} := (\boldsymbol{x}^\mathsf{T}(\tau), \boldsymbol{p}^\mathsf{T}(\tau), t_f)^\mathsf{T} \in \mathbb{R}^{2n+1}$, we obtain

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

the right-hand side of the boundary value problem by

$$\boldsymbol{z}'(\tau) = (z_1', z_2', \ldots, z_{2n+1}')^{\mathsf{T}} = t_f \begin{pmatrix} \boldsymbol{f}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau)) \\ -\lambda_1 \Pi_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(\tau)) - \boldsymbol{f}_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau))\boldsymbol{p}(\tau) \\ 0 \end{pmatrix}, \tag{4.38}$$

where the control $\boldsymbol{u}(\tau)$ can be computed by Eq. (4.37) if we replace the time dependencies with the normalized time τ. However, one must pay attention since $\bar{\boldsymbol{u}}(t)$ is only defined on $[t_0, \bar{t}_f]$, and therefore we replace $\bar{\boldsymbol{u}}(t) \to \bar{\boldsymbol{u}}(\tau \bar{t}_f)$. The last entry of the vector in Eq. (4.38) is the differential equation for the time transformation. Finally, the boundary conditions can be summarized by

$$\begin{aligned} z_j(0) &= x_{0,j} & j &= 1, \ldots, n \\ z_j(1) &= \lambda_2\, x_{f,j} + (1 - \lambda_2)\bar{x}_{f,j} & j &= 1, \ldots, r \\ z_j(1) &= 0 & j &= r+1, \ldots, n \\ 0 &= \lambda_1 \big[1 + \Pi(\boldsymbol{x}(1))\big] + \boldsymbol{p}^{\mathsf{T}}(1)\boldsymbol{f}(\boldsymbol{x}(1), \boldsymbol{u}(1)) \\ &\quad + (1 - \lambda_1)\big[h(\boldsymbol{u}(1)) + S(1)\big]. \end{aligned} \tag{4.39}$$

In order to initiate the algorithm, we define a mesh grid on which the solution should be computed. Using the presented homotopy method, the procedure for solving two-point boundary value problems for time-optimal trajectories incorporates the following steps:

1. Guess a final time $\bar{t}_f$, a control history $\bar{\boldsymbol{u}}(t)$ and set $\lambda_1 = 0$ and $\lambda_2 = 0$. Note, a good estimation saves computation time and avoids transformation difficulties during the solution procedure.
2. Compute the state trajectories $\bar{\boldsymbol{x}}(t)$ by a forward solution of the ODE-state equations in Eq. $(4.8)_1$ in the interval $[t_0, \bar{t}_f]$ starting with $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$. Store the solution in the memory.
3. Initialize $z_1(\tau), \ldots, z_n(\tau)$ by interpolating the state values $\bar{\boldsymbol{x}}(t)$ at the grid points from the memory. Set the values of the adjoint variables $z_{n+1}(\tau), \ldots, z_{2n}(\tau)$ to zero and the final time $z_{2n+1}(\tau)$ to $\bar{t}_f$.
4. Solve the boundary value problem consisting of Eq. (4.38) and Eq. (4.39) by using the initial guess for $\boldsymbol{z}(\tau)$ from the previous iteration. If the solver fails, reduce the value of the homotopy parameters, otherwise save the solution $\boldsymbol{z}(\tau)$ and increase one or both homotopy parameters λ_1 and/or λ_2 for the next iteration.
5. Interpolate the new solution at the grid points as a new proposal for the next initial estimate of the boundary value solution.
6. Repeat step 4 and 5 until $\lambda_2 = 1$ and λ_1 is equal or sufficiently close to one.

Concluding, if the initial solution is far away from the optimal solution, the necessary sequence of boundary value problems can grow rapidly using the homotopy method. Furthermore, the solution path from the modified problem to the original problem may include turning points, as e. g.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

in [48, Sec. 11.3], so that the boundary solver is unable to converge. In this case, the continuation method can circumvent this problem for one varying homotopy parameter. One possibility is, for example, to increase λ_2 rapidly to 1, so that the boundary conditions are fulfilled, and afterwards to increase the value of λ_1 using a continuation method by adding $\lambda_1'(\tau) = 0$ to Eq. (4.38). In this case, an additional boundary condition of the form

$$\lambda_1 - \lambda_{1,\text{old}} = s \tag{4.40}$$

is numerically favorable. Herein, s is the step size which can be determined via the convergence behavior of the boundary value solver routine. A simple suggestion would be to reduce the step size s if the boundary value solver is struggling to obtain a solution, otherwise s can be increased. To cite a further example, we can also use a boundary condition of the form

$$\|\boldsymbol{z}(1) - \boldsymbol{z}_{\text{old}}(1)\|^2 = \text{s}^2, \tag{4.41}$$

Alternatively, one can also use the solution vector at $\tau = 0$ instead.

4.7 Problems

In order to discuss the solution of the optimality conditions with boundary value solvers and for comparison with the optimization methods developed below, we consider a classical example from satellite dynamics and an academic example which is widely studied in control theory. The first example deals with planning the time-optimal trajectory of a spacecraft from the lunar surface into a circular orbit, which is also studied in [39]. In a second example we look for the excitation force of a cart double pendulum system to transfer the double pendulum from the lower to the upper rest position in minimal time. Both problems are implemented in MATLAB and solved by the `bvp4c` routine, which solves the two-point boundary value problem using a collocation method, see the article by Kierzenka and Shampine [38] for solver details.

4.7.1 The Lunar Ascent Problem

Almost 60 years ago, time-optimal control and minimal fuel problems have been focused in the Apollo program. Not surprisingly, this topic continues to excite, and trajectory planning in space flight remains the subject of many articles and textbooks. The time-optimal control problem for a space vehicle in the lunar gravitational field shown in Fig. 4.1 is as well addressed in the book by Kirk [39, Sec. 5.1], in which a space vehicle is launched from the surface of the Moon into a circular orbit. We are interested in controlling the spacecraft minimizing the cost functional

$$J = \int_{t_0}^{t_f} 1 \mathrm{d}t, \tag{4.42}$$

measuring only the time period, since no penalty terms are introduced.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

We consider only planar motions of the spacecraft described in polar coordinates $r(t)$ and $\varphi(t)$. Then the system equations of the space vehicle can be readily formulated by

$$\begin{aligned}
\dot{r}(t) &= v_r(t) \\
\dot{\varphi}(t) &= \frac{v_t(t)}{r(t)} \\
\dot{v}_r(t) &= \frac{v_t^2(t)}{r(t)} - \frac{\mu}{r^2(t)} + \frac{F}{m(t)}\sin(u(t)) \\
\dot{v}_t(t) &= -\frac{v_r(t)v_t(t)}{r(t)} + \frac{F}{m(t)}\cos(u(t)) \\
\dot{m}(t) &= -\frac{F}{gI_{sp}},
\end{aligned} \tag{4.43}$$

where $m(t)$ denotes the mass of the vehicle, g Earth's gravitational acceleration at sea level, $\mu = R^2 g_M$ Moon's gravitational constant, g_M Moon's gravitational acceleration, and R the radius of the Moon. The vehicle can be maneuvered by a constant rocket thrust F in a variable direction described by the angle $u(t)$ which is considered as our control variable.
Restricting to small thrust angles the state equations become linear in $u(t)$ and therefore would lead to a classical bang-bang control. Nevertheless, notice that these differential equations are nonlinear in the states and control variables. Hence, no constraints/bounds on the thrust angle are introduced. Moreover, I_{sp} denotes the specific impulse of the rocket propellant. If we introduce

$$\boldsymbol{x}(t) = \left(r, \varphi, v_r, v_t, m\right)^{\mathsf{T}}, \tag{4.44}$$

as the vector of state variables, the system equations in Eq. (4.43) have the form of Eq. $(4.8)_1$.

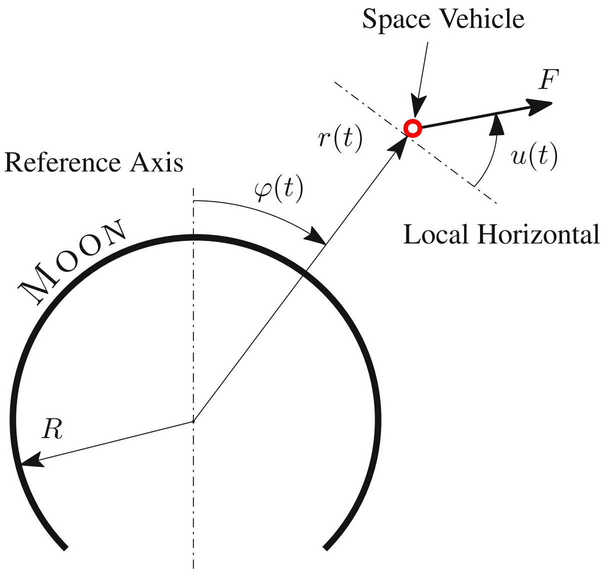

Figure 4.1: The Lunar Excursion Module in the gravity field of the Moon

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

The adjoint equations are derived by Eq. $(4.8)_2$ yielding

$$\begin{aligned}
\dot{p}_1(t) &= \frac{p_2(t)v_t(t)}{r^2(t)} + p_3(t)\left(\frac{v_t^2(t)}{r^2(t)} - \frac{2\mu}{r^3(t)}\right) - \frac{p_4(t)v_r(t)v_t(t)}{r^2(t)} \\
\dot{p}_2(t) &= 0 \\
\dot{p}_3(t) &= -p_1(t) + \frac{p_4(t)v_t(t)}{r(t)} \\
\dot{p}_4(t) &= -\frac{p_2(t)}{r(t)} - \frac{2p_3(t)v_t(t)}{r(t)} + \frac{p_4(t)v_r(t)}{r(t)} \\
\dot{p}_5(t) &= \frac{p_3(t)F}{m^2(t)}\sin\left(u(t)\right) + \frac{p_4(t)F}{m^2(t)}\cos\left(u(t)\right),
\end{aligned} \tag{4.45}$$

where we have introduced a set of adjoint variables

$$\boldsymbol{p}(t) = \left(p_1, p_2, p_3, p_4, p_5\right)^{\mathsf{T}}. \tag{4.46}$$

From Eq. $(4.8)_3$ we obtain a condition for the optimal control $u(t)$ given by

$$0 = \frac{F}{m(t)}\Big(p_3(t)\cos\left(u(t)\right) - p_4(t)\sin\left(u(t)\right)\Big), \tag{4.47}$$

from which we can express the control input in terms of the adjoint variables $p_3(t)$ and $p_4(t)$ with

$$u(t) = \arctan\left(\frac{p_3(t)}{p_4(t)}\right), \tag{4.48}$$

or, respectively,

$$\sin(u(t)) = \pm\frac{p_3(t)}{\sqrt{p_3^2(t) + p_4^2(t)}} \qquad \cos(u(t)) = \pm\frac{p_4(t)}{\sqrt{p_3^2(t) + p_4^2(t)}} \tag{4.49}$$

to circumvent troubles with the domain of the arctan function. As already predicted, care must also be taken with unconstrained controls that can not be expressed unambiguously from Eq. $(4.8)_3$. According to Pontryagin's minimum principle introduced in Sec. 4.5, the negative sign must be taken in Eq. (4.49) to obtain a minimum of the Hamiltonian. The space vehicle with the initial mass m_0 is launched at the surface of the Moon with the polar coordinates $(R, 0)$, where R is the radius of the Moon. Thus, the vehicle starts from rest, the initial velocities v_r and v_t are zero at $t = t_0 = 0$. The boundary conditions read

$$r(0) = R \quad \varphi(0) = 0 \quad v_r(0) = 0 \quad v_t(0) = 0 \quad m(0) = m_0, \tag{4.50}$$

where $m_0 = 4774\,\mathrm{kg}$ is the initial mass of the ascent stage including a full propellant tank. The final radius of the vehicle position is given by the radius of the Moon R and the altitude h of the orbit. We require that the final radial velocity v_r is zero and the centrifugal force is balanced by the gravitational force, so we obtain also a condition for the tangential velocity v_t. Moreover, the final angle $\varphi(t_f)$ and the final mass $m(t_f)$ are arbitrary, therefore the associated adjoint variables $p_2(t_f)$ and $p_5(t_f)$ must be zero.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

The right boundary conditions for the state/adjoint variables are of the form as in Eq. (4.21) and are prescribed by

$$r(t_f) = R + h \quad v_r(t_f) = 0 \quad v_t(t_f) = \sqrt{\frac{\mu}{R+h}} \quad p_2(t_f) = 0 \quad p_5(t_f) = 0, \tag{4.51}$$

forming a set of five boundary conditions at $t = t_f$. Since the final time is free, we consider an additional boundary condition according to Eq. (4.22) given by:

$$0 = 1 - \frac{F}{m(t_f)}\sqrt{p_3^2(t_f) + p_4^2(t_f)}, \tag{4.52}$$

in which the control is eliminated by Eq. (4.49) and the final values $r(t_f)$, $v_r(t_f)$, $v_t(t_f)$, $p_2(t_f)$ and $p_5(t_f)$ are inserted. The parameters for the numerical computations are taken from [46] and can be summarized by: $R = 1738\,\mathrm{km}$, $h = 111\,\mathrm{km}$, $g = 9.81 \times 10^{-3}\,\mathrm{km/s^2}$, $g_M = 1.622 \times 10^{-3}\,\mathrm{km/s^2}$, $\mu = 4899.48\,\mathrm{km^3/s^2}$, $I_{sp} = 311\,\mathrm{s}$ and $F = 16\,\mathrm{kN}$. Recall, before we can apply the boundary value solver directly, the problem must be transformed to a dimensionless time scale $\tau \in [0, 1]$. To facilitate convergence, one homotopy parameter is introduced to rescale the orbit altitude h (defined in the boundary conditions) from small to the original size. The initial guess for the boundary value routine for the state and adjoint variables have been adapted to satisfy the boundary conditions. Then, the states and adjoint variables between the initial and final values have been assumed linear. Of course, neither the state equations nor the adjoint equations are satisfied by this assumption, but the boundary conditions are fulfilled, and in this case it is sufficient that the solver converges. Finally, the solution is obtained by the boundary value solver routine `bvp4c` using 500 mesh points. The control angle of the thrust nozzle is depicted in Fig. 4.2. Initially, the control input is about $67.5\,\mathrm{deg}$ to accelerate towards the orbit. After $329\,\mathrm{s}$ the thrust

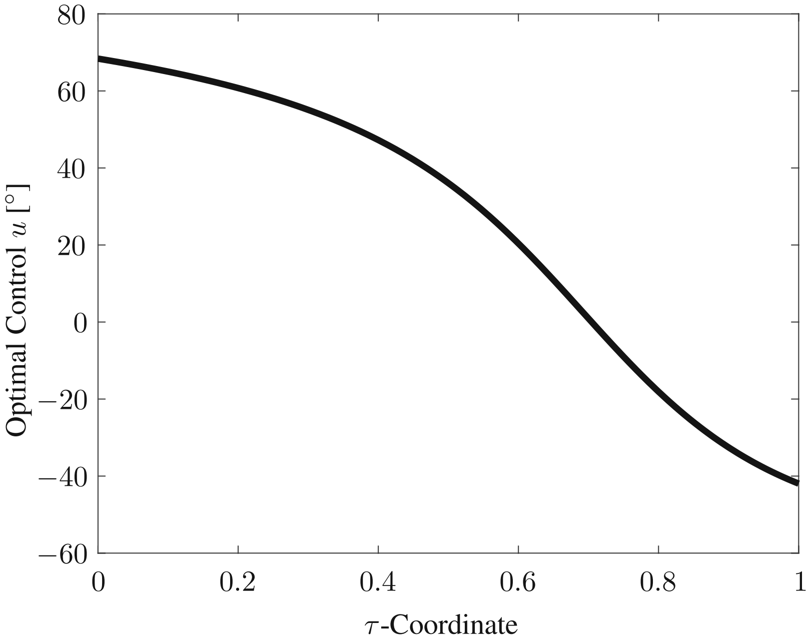

Figure 4.2: Optimal control input for the thruster for lunar ascent

nozzle angle becomes negative to reduce the radial velocity and reach a balanced state in the orbit. The optimal trajectory for the orbit injection is depicted in Fig. 4.3. The arrows show the orientation of the thrust force acting on the space vehicle.

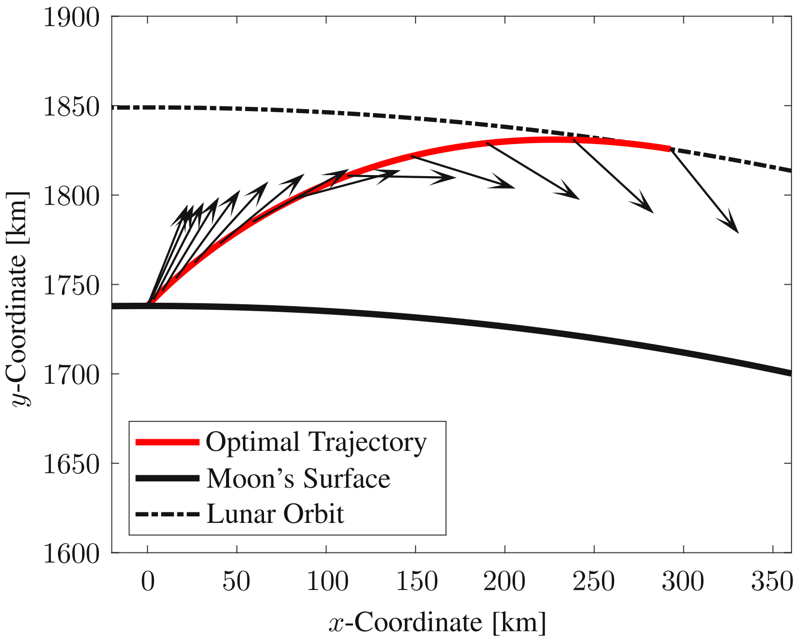

Figure 4.3: Optimal trajectory for lunar ascent

4.7.2 Inverse Double Pendulum

As a second example, we study an inverse double pendulum consisting of a cart and two rods connected by hinge joints. The cart double pendulum is a popular model for demonstrating, how to handle and control chaotic processes, see for instance [22]. The system under consideration is depicted in Fig. 4.4 and we are looking for the excitation force $u(t)$ of the cart which is required to swing up the double pendulum into the upper rest position in minimal time. Hence, the cost functional to be minimized is given by

$$J = \int_{t_0}^{t_f} [1 + \Pi(x(t))] \, \mathrm{d}t. \tag{4.53}$$

While the control force can directly restricted by $-u^* \leq u(t) \leq u^*$ with $u^* = 15$, the sliding distance is limited by $-x^* \leq x(t) \leq x^*$ using a penalty approach. The penalty function is readily given by

$$\Pi(x(t)) := \begin{cases} 0 & \text{for} \quad |x(t)| < x^* \\ \dfrac{\mu}{2} \left(|x(t)| - x^*\right)^2 & \text{otherwise,} \end{cases} \tag{4.54}$$

in which $\mu = 12$ is a weighting constant to tune the function and to ensure that $x(t)$ complies with the limit $x^* = 1$. The system has three degrees of freedom, given by the sliding distance $x(t)$ of the cart and the pendulum angles $\varphi_1(t)$ and $\varphi_2(t)$.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

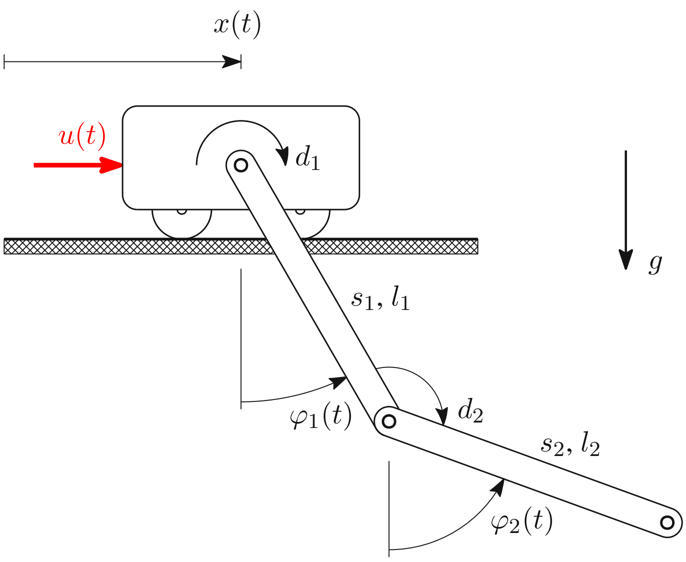

Figure 4.4: The cart double pendulum system

The pendulum lengths are given by $l_1 = l_2 = 0.5\,\mathrm{m}$ and their center of gravity positions read $s_1 = s_2 = 0.25\,\mathrm{m}$. The mass of the cart is $m_c = 1.5\,\mathrm{kg}$, and the inertia parameters of the rods are $m_1 = m_2 = 1\,\mathrm{kg}$ and $J_1 = J_2 = 0.0208\,\mathrm{kg\,m^2}$. In the hinge joints a linear damping force acts with the damping constants $d_1 = 0.01\,\mathrm{N\,m\,s}$ and $d_2 = 0.002\,\mathrm{N\,m\,s}$. The equations of motion of the system read

$$\begin{aligned}
&(m_c + m_1 + m_2)\,\dot{v}(t) + (m_1 s_1 + m_2 l_1)\cos(\varphi_1(t))\dot{\omega}_1(t) + m_2 s_2 \cos(\varphi_2(t))\dot{\omega}_2(t) \\
&\qquad = u(t) + (m_1 s_1 + m_2 l_1)\sin(\varphi_1(t))\omega_1^2(t) + m_2 s_2 \sin(\varphi_2(t))\omega_2^2(t) \\
&(m_1 s_1 + m_2 l_1)\cos(\varphi_1(t))\dot{v}(t) + \left(J_1 + m_1 s_1^2 + m_2 l_1^2\right)\dot{\omega}_1(t) \\
&\qquad + m_2 l_1 s_2 \cos(\varphi_1(t) - \varphi_2(t))\dot{\omega}_2(t) \\
&\qquad = -d_1\omega_1(t) + d_2\left(\omega_2(t) - \omega_1(t)\right) - g\left(m_1 s_1 + m_2 l_1\right)\sin(\varphi_1(t)) \\
&\qquad - m_2 l_1 s_2 \sin(\varphi_1(t) - \varphi_2(t))\omega_2^2(t) \\
&m_2 s_2 \cos(\varphi_2(t))\dot{v}(t) + m_2 l_1 s_2 \cos(\varphi_1(t) - \varphi_2(t))\dot{\omega}_1(t) + \left(J_2 + m_2 s_2^2\right)\dot{\omega}_2(t) \\
&= -d_2\left(\omega_2(t) - \omega_1(t)\right) - m_2 g s_2 \sin(\varphi_2(t)) + m_2 l_1 s_2 \sin(\varphi_1(t) - \varphi_2(t))\omega_1^2(t),
\end{aligned} \tag{4.55}$$

where $\omega_1(t) = \dot{\varphi}_1(t)$, $\omega_2(t) = \dot{\varphi}_2(t)$ and $v(t) = \dot{x}(t)$. These equations can be readily written in first order form by introducing the vector of state variables

$$\boldsymbol{x}(t) = (x, \varphi_1, \varphi_2, v, \omega_1, \omega_2)^{\mathsf{T}}\,. \tag{4.56}$$

A closer look at the equations of motion reveals that the control enters linearly and consequently one expects either a singular or a bang-bang control if we minimize the final time. Besides, we do not get any information about the choice of the control force from Eq. $(4.8)_3$. Therefore, we resort to the homotopy approach presented in Sec. 4.6, where the control can then be expressed by Eq. (4.37) instead. Note that the derivation of the adjoint differential equations in Eq. (4.33) leads

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

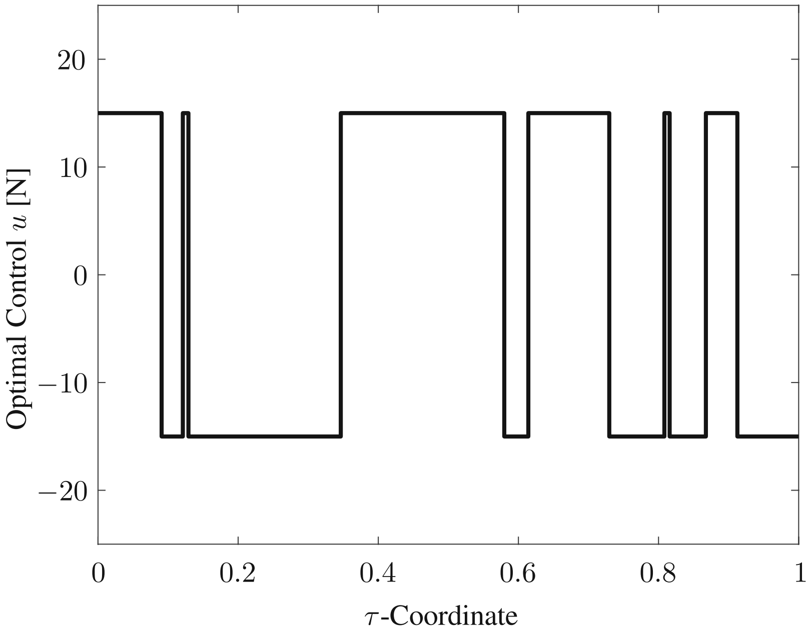

Figure 4.5: Optimal excitation force for the upswing maneuver

to unwieldy expressions that would fill entire pages, hence the adjoint equations are not stated explicitly. Initially, the system is at rest with the coordinates

$$\varphi_1(0) = 0 \quad \varphi_2(0) = 0 \quad x(0) = 0 \quad \omega_1(0) = 0 \quad \omega_2(0) = 0 \quad v(0) = 0. \tag{4.57}$$

The final configuration of the system at $t = t_f$ is prescribed by

$$\varphi_1(t_f) = \pm\,(2a+1)\,\pi \quad \varphi_2(t_f) = \pm\,(2b+1)\,\pi \quad \omega_1(t_f) = 0 \quad \omega_2(t_f) = 0 \quad v(t_f) = 0 \tag{4.58}$$

with $a, b \in \mathbb{N}$. Notice that the final value for the cart position $x(t_f)$ is free. However, before we can solve now the proposed two-point boundary value problem we have to utilize a transformation to the τ-domain. Finally, the resulting control signal is depicted in Fig. 4.5, which shows a pure bang-bang control. We count eleven switching points, where the control signal alternates between the control constraints $+u^*$ and $-u^*$. The minimum final time t_f is given by $3.0332\,\mathrm{s}$ and the series of the time-optimal swing up maneuver is illustrated in Fig. 4.6.

Since the solution of the boundary value problem is a challenging task and we do not want to give the reader the impression that the solution was obtained by pressing one button, we would like to conclude this example by spending a few remarks on handling such problems. As already mentioned in Sec. 4.6, one problem in solving boundary value problems is finding suitable initial estimates for all state and adjoint variables that are sufficiently close to the optimal solution, so that the boundary value solver converges. Hence, to overcome this problem, the `bvp4c` routine was modified by the author to utilize a continuation method and make the solver more robust with respect to the initial guess of the solution. Therefore, the original cost functional in Eq. (4.53) was replaced by Eq. (4.30) and the final conditions in Eq. (4.58) were replaced by Eq. (4.39). Then the resulting auxiliary optimal control problem summarized in Eq. (4.30) and Eq. (4.37) was

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

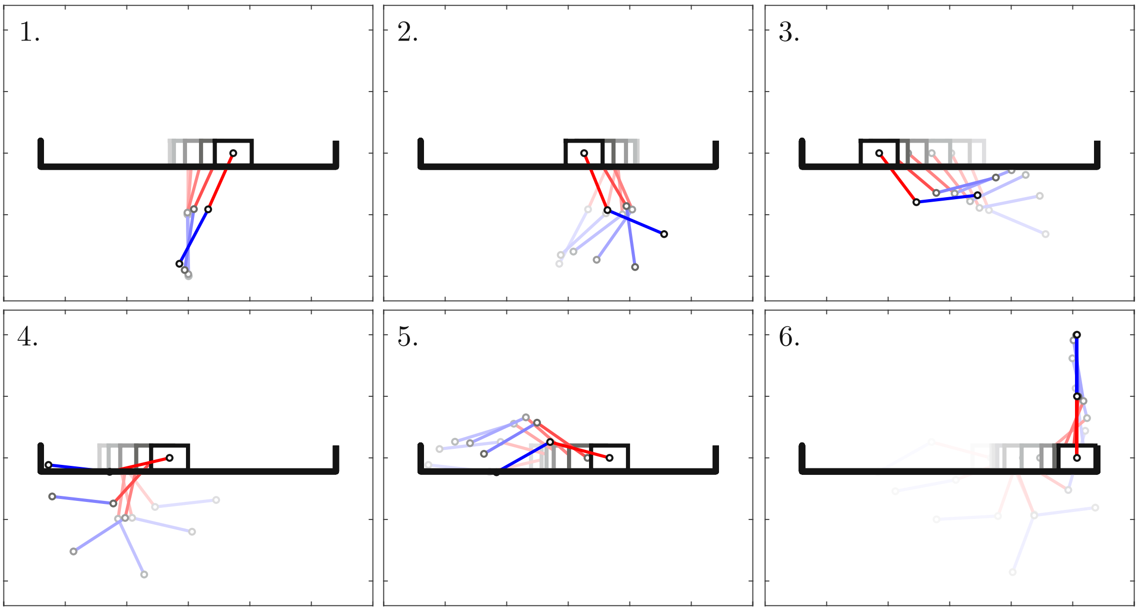

Figure 4.6: Series of the upswing maneuver of the cart double pendulum

solved instead and transformed to the original problem by solving a sequence of boundary value problems. A step size control for the continuation method, denoting the change from one boundary value problem to the next, can be utilized via the convergence behavior of the Newton method, which is used within the `bvp4c` routine. A simple suggestion for this comes from Seydel in [62], who uses the number of iteration steps of the Newton method as a measure for the sensitivity of the problem, and allows to traverse sensitive areas of the solution curve. Based on the residual control of `bvp4c`, the adaptive mesh refinement yields the solution on a grid of 2482 points, while the maximum residual is smaller than 9.99×10^{-10}. Note, the fraction of the regularization term that includes the initial control in the cost functional becomes negligibly small.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Chapter 5

The Adjoint Method for Final States Lying on a Surface

In this chapter, we discuss a special class of time-optimal control problems for dynamic systems, where the final state of a system lies on a hypersurface. As explained in the previous chapter, the optimal control can be found by solving a two-point boundary value problem. However, the treatment of the boundary value problem requires an appropriate estimation for all state and adjoint variables, otherwise the boundary value solver will not converge. An alternative to solving the underlying boundary value problem is given by gradient methods, independently developed by Kelley [37] and by Bryson [15] already in the sixties of the past century. Unfortunately, the numerical computation of a cost functional gradient with respect to controls is time-consuming and is often the bottleneck for efficient computing power. The novel method in this chapter uses the adjoint method to compute the gradient of a cost functional, which exploits the decoupling of boundary conditions by sequentially integrating the equations of motion and the adjoint equations, by one forward and one backward integration scheme. With the gradient information, the control input can be improved iteratively converging to an optimal control. Even by choosing a suboptimal initial guess for the control inputs, in general, the proposed method converges to the (local) optimal solution. Several parts of this chapter have already been published by the author in the Journal of Computational and Nonlinear Dynamics [21].

5.1 Problem Formulation

Before describing the adjoint gradient method for a special class of boundary condition, the time-optimal control problem is summarized here. We consider a first-order system described by the state equations

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0. \tag{5.1}$$

We pursue the same goal as in the preceding chapter and look for controls that minimize the cost functional

$$J = \int_{t_0}^{t_f} \left[1 + \Pi(\boldsymbol{x}(t), \boldsymbol{u}(t))\right] \, \mathrm{d}t. \tag{5.2}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

In contrast to the boundary value problem, a control constraint is often considered by a penalty term in the cost functional when using a gradient based method. Thus, we introduced a penalty term $\Pi(\boldsymbol{x}(t), \boldsymbol{u}(t))$ that also includes a control dependence to account for the limits of $\boldsymbol{u}(t)$ and, hence, the cost functional is open to include any inequality constraint. In order to cite two possible application scenarios, the penalty term can be used to limit the drive force/torque of a system, to account for natural bounds due to engine performance, or to consider obstacles in automotive and robotics. In this chapter we are interested in problems where the final state $\boldsymbol{x}(t_f)$ of Eq. (5.1) terminates on a hypersurface. In time domain, this end point constraint may be expressed by a "spatial" variable

$$s(t) := \Phi(\boldsymbol{x}(t)), \tag{5.3}$$

which ranges between the fixed boundaries $s_0 = \Phi(\boldsymbol{x}(t_0))$ and $s_f = \Phi(\boldsymbol{x}(t_f))$. If we apply an iterative solution strategy to find $\delta J = 0$ in Eq. (5.2), a problem arises because a modified control causes a change of the time interval $[t_0, t_f]$ to $[t_0, t_f^*]$.

If $t_f^* < t_f$, which should be the case for a control variation, the improved control $\boldsymbol{u}^*(t)$ can be truncated at $t = t_f^*$ although the information about $\boldsymbol{u}^*(t)$ is available also for $t_f^* < t < t_f$. However, if the control update was selected too large, the terminal condition in Eq. (5.3) may be eventually met at a later time instance $t_f^* > t_f$. In this case, we have no information about $\boldsymbol{u}^*(t)$ for $t_f < t < t_f^*$. Hence, for time-optimal control problems, it is convenient to transform the time coordinate into a coordinate within a fixed interval. For this purpose, it is natural to express the time by the variable $s(t)$ defined in Eq. (5.3). Depending on the problem, i. e. which boundary conditions are imposed, different forms of the adjoint method can be pursued. Based on Eq. (5.3), two solution strategies concerning the adjoint gradient method are pursued in this chapter:

(a) A hybrid formulation, where the state and adjoint equations are solved in the original time domain and only the controls and the gradient are defined in the fixed interval, and

(b) the complete transformation of the time coordinate to a fixed interval.

Both approaches originate from [21], are robust with respect to poor initial controls and yield a shorter final time and, hence, an improved control after every iteration.

5.2 Transformation to a Hybrid Domain

First, we discuss a hybrid formulation of the adjoint method. In order to apply a solution strategy to our time-optimal control problem, we extend Eq. (5.2) by a zero term containing the state equations multiplied with a set of adjoint variables. For arbitrary adjoint variables $\boldsymbol{p}(t)$, the augmented cost functional from Eq. (5.2) reads

$$\bar{J} = \int_{t_0}^{t_f} \Big[1 + \Pi(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{p}^\mathsf{T}\left(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) - \dot{\boldsymbol{x}}\right)\Big]\, \mathrm{d}t. \tag{5.4}$$

Now we try to compute the gradient of the cost functional by applying the δ-operation. In our initial exposure to optimal control theory, we have found the variation of the augmented cost

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

functional in Eqs. (4.1–4.7). However, since our penalty function may now depend also on the controls, we observe an additional control variation of the peanlty function, i. e., $\Pi_{\boldsymbol{u}}\delta\boldsymbol{u}(t)$. Hence, Eq. (5.4) is given by

$$\begin{aligned}\delta\bar{J} = \int_{t_0}^{t_f} &\left[\left(\Pi_{\boldsymbol{u}} + \boldsymbol{p}^{\mathsf{T}}\boldsymbol{f}_{\boldsymbol{u}}\right)\delta\boldsymbol{u} + \left(\Pi_{\boldsymbol{x}} + \boldsymbol{p}^{\mathsf{T}}\boldsymbol{f}_{\boldsymbol{x}} + \dot{\boldsymbol{p}}^{\mathsf{T}}\right)\delta\boldsymbol{x}\right]\,\mathrm{d}t \\ &- \boldsymbol{p}^{\mathsf{T}}(t_f)\delta\boldsymbol{x}(t_f) + \Big[1 + \Pi(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))\Big]\delta t_f.\end{aligned} \tag{5.5}$$

Since the end time and thus the boundary term in Eq. (5.5) is implicitly defined by Eq. (5.3), it is difficult to realize a complete transformation of the time coordinate. If we are using a multibody simulation software, this would require a massive intervention into the software architecture. Hence, it would be preferable to solve the equations of motion in the time domain and to transform the gradient formula only. Before we consider a proper transformation, we rewrite the boundary term in Eq. (5.5) by using Eq. (4.14) for δt_f, leading to

$$\begin{aligned}\delta\bar{J} = \int_{t_0}^{t_f} &\left[\left(\Pi_{\boldsymbol{u}} + \boldsymbol{p}^{\mathsf{T}}\boldsymbol{f}_{\boldsymbol{u}}\right)\delta\boldsymbol{u} + \left(\Pi_{\boldsymbol{x}} + \boldsymbol{p}^{\mathsf{T}}\boldsymbol{f}_{\boldsymbol{x}} + \dot{\boldsymbol{p}}^{\mathsf{T}}\right)\delta\boldsymbol{x}\right]\,\mathrm{d}t \\ &- \left[\boldsymbol{p}^{\mathsf{T}}(t_f) + \frac{1 + \Pi(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))}{\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))\boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))}\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))\right]\delta\boldsymbol{x}(t_f),\end{aligned} \tag{5.6}$$

in which we inserted $\boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))$ for $\dot{\boldsymbol{x}}(t_f)$ again. At a first glance, it seems that the integration variable t can be directly substituted by $s = \Phi(\boldsymbol{x}(t))$. However, Fig. 2.2 illustrates that the change of the integration variable has also an impact on the δ-operation describing the control variation. Let $\boldsymbol{u}^*(t) = \boldsymbol{u}(t) + \delta\boldsymbol{u}(t)$ be a perturbed control in terms of the original time, then a control variation $\delta\boldsymbol{u}(t)$ causes a state variation $\delta\boldsymbol{x}(t)$. Thus, we will also observe a variation of the state dependent coordinate $s(t) = \Phi(\boldsymbol{x}(t))$, which is up to first order given by

$$\delta s(t) = \Phi_{\boldsymbol{x}}(\boldsymbol{x}(t))\,\delta\boldsymbol{x}(t). \tag{5.7}$$

Analogous to a varied function in Eq. (2.24), we deduce for a perturbed control $\delta\boldsymbol{u}(t) = \delta\boldsymbol{u}(s(t)) + (\mathrm{d}\boldsymbol{u}/\mathrm{d}s)\delta s(t)$. Hence, after inserting Eq. (5.7) for $\delta s(t)$, the control variation in terms of the time t reads

$$\delta\boldsymbol{u}(t) = \delta\boldsymbol{u}(s(t)) + \frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}s}\delta s(t) = \delta\boldsymbol{u}(s(t)) + \frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}s}\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t))\,\delta\boldsymbol{x}(t). \tag{5.8}$$

Since we are looking for $\delta\boldsymbol{u}(s)$ which causes the largest local decrease of the cost functional J, the variation $\delta\boldsymbol{u}(t)$ in Eq. (5.6) must be expressed by this relation, yielding

$$\begin{aligned}\delta\bar{J} = &\int_{t_0}^{t_f}\left(\Pi_{\boldsymbol{u}} + \boldsymbol{p}^{\mathsf{T}}\boldsymbol{f}_{\boldsymbol{u}}\right)\delta\boldsymbol{u}(s(t))\,\mathrm{d}t \\ &+ \int_{t_0}^{t_f}\left[\Pi_{\boldsymbol{x}} + \Pi_{\boldsymbol{u}}\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}s}\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t)) + \boldsymbol{p}^{\mathsf{T}}\left(\boldsymbol{f}_{\boldsymbol{x}} + \boldsymbol{f}_{\boldsymbol{u}}\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}s}\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t))\right) + \dot{\boldsymbol{p}}^{\mathsf{T}}\right]\delta\boldsymbol{x}(t)\,\mathrm{d}t \\ &- \left[\boldsymbol{p}^{\mathsf{T}}(t_f) + \frac{1 + \Pi(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))}{\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))\boldsymbol{f}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))}\Phi_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(t_f))\right]\delta\boldsymbol{x}(t_f).\end{aligned} \tag{5.9}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

So far, we have not put any restrictions on the adjoint variables. Now we shall find it convenient to choose $\boldsymbol{p}(t)$ such that all terms multiplied with $\delta\boldsymbol{x}$ vanish. Thus, the adjoint variables are defined by the linear final value problem

$$\begin{aligned}\dot{\boldsymbol{p}}(t) &= -\Pi_{\boldsymbol{x}}^{\mathsf{T}} - \left(\Pi_{\boldsymbol{u}}\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}s}\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t))\right)^{\mathsf{T}} - \left(\boldsymbol{f}_{\boldsymbol{x}} + \boldsymbol{f}_{\boldsymbol{u}}\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}s}\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t))\right)^{\mathsf{T}}\boldsymbol{p}(t)\\ \boldsymbol{p}(t_f) &= -\frac{1+\Pi(\boldsymbol{x}(t_f),\boldsymbol{u}(t_f))}{\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t_f))\boldsymbol{f}(\boldsymbol{x}(t_f),\boldsymbol{u}(t_f))}\Phi_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(t_f)),\end{aligned} \tag{5.10}$$

which can be solved backward in time after the system equations in Eq. (5.1) have been solved for $\boldsymbol{x}(t)$ yielding the time variant Jacobi matrices $\boldsymbol{f}_{\boldsymbol{x}}(\boldsymbol{x}(t),\boldsymbol{u}(t))$, $\Phi_{\boldsymbol{x}}(\boldsymbol{x}(t))$ and $\Pi_{\boldsymbol{x}}(\boldsymbol{x}(t),\boldsymbol{u}(t))$. Notice that we have added the term $\boldsymbol{f}_{\boldsymbol{u}}(\mathrm{d}\boldsymbol{u}/\mathrm{d}s)\Phi_{\boldsymbol{x}}$ to the adjoint system. If $\boldsymbol{p}(t)$ satisfies Eq. (5.10), the first variation of the cost functional in Eq. (5.9) reduces to

$$\delta\bar{J} = \int_{t_0}^{t_f}\left(\Pi_{\boldsymbol{u}} + \boldsymbol{p}^{\mathsf{T}}\boldsymbol{f}_{\boldsymbol{u}}\right)\delta\boldsymbol{u}(s(t))\,\mathrm{d}t, \tag{5.11}$$

in which we can now substitute the integration variable t by s. With

$$\mathrm{d}t = \frac{1}{\dot{s}}\mathrm{d}s = \frac{1}{\Phi_{\boldsymbol{x}}\dot{\boldsymbol{x}}}\mathrm{d}s = \frac{1}{\Phi_{\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})}\mathrm{d}s = g(\boldsymbol{x},\boldsymbol{u})\mathrm{d}s, \tag{5.12}$$

where $g(\boldsymbol{x},\boldsymbol{u}) = 1/(\Phi_{\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}))$, we obtain

$$\delta\bar{J} = \int_{s_0}^{s_f}\left(\Pi_{\boldsymbol{u}} + \boldsymbol{p}^{\mathsf{T}}\boldsymbol{f}_{\boldsymbol{u}}\right)\delta\boldsymbol{u}(s)g(\boldsymbol{x},\boldsymbol{u})\mathrm{d}s. \tag{5.13}$$

Finally, the largest local decrease of the cost functional in Eq. (5.11) results from the control variation

$$\delta\boldsymbol{u}(s) = -\kappa\, g(\boldsymbol{x},\boldsymbol{u})\left(\Pi_{\boldsymbol{u}}^{\mathsf{T}} + \boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}}\boldsymbol{p}\right), \tag{5.14}$$

where κ is a sufficiently small positive number. With this result, we are ready to apply a gradient based optimization procedure for finding a minimum of J, which can be summarized as follows:

1. Choose an initial guess for the control history $\boldsymbol{u}(s)$ which causes the states to satisfy the terminal condition in Eq. (4.11).
2. Solve Eq. (4.1) for the state variables $\boldsymbol{x}(t)$ until Eq. (4.11) is satisfied. During the numerical integration of the state equations, the argument s in $\boldsymbol{u}(s)$ must be replaced by the original time t with Eq. (5.3).
3. Solve Eq. (5.10) starting at $t = t_f$ for the adjoint variables $\boldsymbol{p}(t)$ by a backward integration, e. g., by using a multistep integration scheme, as described in [32, Sec. 3.1].
4. Select a value for κ and compute the control variation $\delta\boldsymbol{u}(s)$ for every s from Eq. (5.14) by involving the inverse function of $s(t)$. Alternatively, a quasi-Newton method can be applied to the update formula in Eq. (5.14), e. g., the BFGS-method.

5. Update the control history by setting $\boldsymbol{u}(s) \to \boldsymbol{u}(s) + \delta\boldsymbol{u}(s)$. With the updated control $\boldsymbol{u}(s)$, the cost functional should be reduced, if κ was selected sufficiently small.

6. Vary κ, resolve the state equations and evaluate the cost functional until an optimal value for κ is found, which minimizes the cost functional along the search direction. For that purpose, a line search algorithm as described in Sec. 3.1.2 can be used.

7. Replace the control history $\boldsymbol{u}(s)$ by the updated control and repeat steps 2 through 6. Terminate the optimization procedure when Eq. (5.13) becomes sufficiently small (zero gradient).

In summary, an advantage of the presented method is that the equations of motion and also the adjoint system can be solved in the time domain and only the control variables and the gradient formula are transformed into the state space. Unfortunately, this requires the derivative of the controls with respect to s which can become infinite for bang-bang controls and could cause an undesirable excitation term in the adjoint equations in Eq. $(5.10)_1$. Note that s is often used as an arc length parameter, but here no specific parametrization is chosen yet. In the next paragraph we show that this problem can be overcome by an elimination of the time coordinate also in the state equations in Eq. (4.1). However, the price to pay is that a standard multibody simulation software is probably no longer applicable.

5.3 Transformation to Space Domain

The function $s(t)$ defined by Eq. (5.3) can also be used to eliminate the time coordinate from the state equations in Eq. (4.1). Using Eq. (5.12), we obtain

$$\dot{\boldsymbol{x}} = \frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}s}\frac{\mathrm{d}s}{\mathrm{d}t} = \boldsymbol{x}'(s)\Phi_{\boldsymbol{x}}\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) \qquad \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \tag{5.15}$$

or, respectively,

$$\boldsymbol{x}'(s) \ = g(\boldsymbol{x}(s),\boldsymbol{u}(s))\boldsymbol{f}(\boldsymbol{x}(s),\boldsymbol{u}(s)) := \tilde{\boldsymbol{f}}(\boldsymbol{x}(s),\boldsymbol{u}(s)), \qquad \boldsymbol{x}(s_0) = \boldsymbol{x}_0, \tag{5.16}$$

in which we used again the abbreviation $g(\boldsymbol{x},\boldsymbol{u}) = 1/(\Phi_{\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}))$. Any derivative with respect to s is denoted by $(\cdot)'$, here. For measuring the original time, we make use of Eq. (5.3) once more and include the additional differential equation

$$t'(s) = \frac{\mathrm{d}t}{\mathrm{d}s} = g(\boldsymbol{x}(s),\boldsymbol{u}(s)), \qquad t(s_0) = t_0, \tag{5.17}$$

which is used to determine the final time $t_f = t(s_f)$. Applying a transformation to space, the cost functional in Eq. (5.2) can be readily rewritten to

$$J = t_f - t_0 + \int_{s_0}^{s_f} \Pi(\boldsymbol{x}(s),\boldsymbol{u}(s))\, g(\boldsymbol{x}(s),\boldsymbol{u}(s))\mathrm{d}s. \tag{5.18}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

In order to derive now a gradient formula of Eq. (5.18) with respect to the controls $\boldsymbol{u}(s)$ we may again extend J by the state equations in Eq. (5.16) and by Eq. (5.17) yielding

$$\bar{J} = t_f - t_0 + \int_{s_0}^{s_f} \Big[\Pi(\boldsymbol{x},\boldsymbol{u})\, g(\boldsymbol{x},\boldsymbol{u}) + \boldsymbol{p}^{\mathsf{T}}\Big(\tilde{\boldsymbol{f}}(\boldsymbol{x},\boldsymbol{u}) - \boldsymbol{x}'\Big) + \xi\Big(g(\boldsymbol{x},\boldsymbol{u}) - t'\Big)\Big]\,\mathrm{d}s, \tag{5.19}$$

in which $\boldsymbol{p}(s) \in \mathbb{R}^n$ denotes a vector of adjoint variables and $\xi(s) \in \mathbb{R}$ is a single adjoint variable associated to the differential equation defining the time coordinate. The linear change of the extended cost functional is given by

$$\begin{aligned} \delta\bar{J} = \delta t_f + \int_{s_0}^{s_f} \Big[& g\Pi_{\boldsymbol{x}}\delta\boldsymbol{x} + \Pi g_{\boldsymbol{x}}\delta\boldsymbol{x} + g\Pi_{\boldsymbol{u}}\delta\boldsymbol{u} + \Pi g_{\boldsymbol{u}}\delta\boldsymbol{u} \\ &+ \boldsymbol{p}^{\mathsf{T}}\left(\tilde{\boldsymbol{f}}_{\boldsymbol{x}}\delta\boldsymbol{x} + \tilde{\boldsymbol{f}}_{\boldsymbol{u}}\delta\boldsymbol{u} - \delta\boldsymbol{x}'\right) + \xi\Big(g_{\boldsymbol{x}}\delta\boldsymbol{x} + g_{\boldsymbol{u}}\delta\boldsymbol{u} - \delta t'\Big)\Big]\mathrm{d}s, \end{aligned} \tag{5.20}$$

where the initial time is fixed, hence $\delta t_0 = 0$. The same applies after an integration by parts of the terms including $\delta\boldsymbol{x}'$ and $\delta t'$ yielding

$$\begin{aligned} \int_{s_0}^{s_f} \boldsymbol{p}^{\mathsf{T}}\delta\boldsymbol{x}'\,\mathrm{d}s &= -\int_{s_0}^{s_f} \boldsymbol{p}'^{\mathsf{T}}\delta\boldsymbol{x}\,\mathrm{d}s + \boldsymbol{p}^{\mathsf{T}}(s_f)\delta\boldsymbol{x}(s_f) \\ \int_{s_0}^{s_f} \xi\delta t'\,\mathrm{d}s &= -\int_{s_0}^{s_f} \xi'\delta t\,\mathrm{d}s + \xi(s_f)\delta t_f, \end{aligned} \tag{5.21}$$

i. e., the variations $\delta\boldsymbol{x}(s_f)$ and δt_0 are zero. Inserting Eq. (5.21) into Eq. (5.20) and collecting all terms including $\delta\boldsymbol{x}(s)$, $\delta\boldsymbol{u}(s)$ and $\delta t(s)$, we obtain

$$\begin{aligned} \delta\bar{J} = \Big(1 - \xi(s_f)\Big)\delta t_f + \int_{s_0}^{s_f} \Big[&\Big(g\Pi_{\boldsymbol{u}} + \Pi g_{\boldsymbol{u}} + \boldsymbol{p}^{\mathsf{T}}\tilde{\boldsymbol{f}}_{\boldsymbol{u}} + \xi g_{\boldsymbol{u}}\Big)\,\delta\boldsymbol{u} \\ &+ \Big(g\Pi_{\boldsymbol{x}} + \Pi g_{\boldsymbol{x}} + \boldsymbol{p}^{\mathsf{T}}\tilde{\boldsymbol{f}}_{\boldsymbol{x}} + \boldsymbol{p}'^{\mathsf{T}} + \xi g_{\boldsymbol{x}}\Big)\,\delta\boldsymbol{x} + \xi'\delta t\Big]\mathrm{d}s \\ &- \boldsymbol{p}^{\mathsf{T}}(s_f)\delta\boldsymbol{x}(s_f). \end{aligned} \tag{5.22}$$

The adjoint variables $\xi(s)$ and $\boldsymbol{p}(s)$ can now be again defined such that all terms with $\delta\boldsymbol{x}(s)$ and $\delta t(s)$ disappear. This is the case if $\xi'(s) = 0$ and $\xi(s_f) = 1$, i. e., $\xi(s) = 1$ for $s \in [s_0, s_f]$, and if

$$\begin{aligned} \boldsymbol{p}'(s) &= -(1 + \Pi)g_{\boldsymbol{x}}^{\mathsf{T}} - g\,\Pi_{\boldsymbol{x}}^{\mathsf{T}} - \tilde{\boldsymbol{f}}_{\boldsymbol{x}}^{\mathsf{T}}\boldsymbol{p} \\ \boldsymbol{p}(s_f) &= \boldsymbol{0}. \end{aligned} \tag{5.23}$$

This final value problem may be solved for $\boldsymbol{p}(s)$. Finally, the first variation of the cost functional reduces to

$$\delta\bar{J} = \int_{s_0}^{s_f} \Big[(1 + \Pi)g_{\boldsymbol{u}} + g\,\Pi_{\boldsymbol{u}} + \boldsymbol{p}^{\mathsf{T}}\tilde{\boldsymbol{f}}_{\boldsymbol{u}} + g_{\boldsymbol{u}}\Big]\,\delta\boldsymbol{u}(s)\,\mathrm{d}s, \tag{5.24}$$

showing the direct influence of $\delta\boldsymbol{u}(s)$ on $\bar{J}$. Finally, an update of the control signals is computed from Eq. (5.24), by choosing

$$\delta\boldsymbol{u}(s) := -\kappa\left[(1 + \Pi)g_{\boldsymbol{u}}^{\mathsf{T}} + g\,\Pi_{\boldsymbol{u}}^{\mathsf{T}} + \tilde{\boldsymbol{f}}_{\boldsymbol{u}}^{\mathsf{T}}\boldsymbol{p} + g_{\boldsymbol{u}}^{\mathsf{T}}\right], \tag{5.25}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

in which κ is again a small positive number determining the size of the control update. Finally, we summarize the main steps of the algorithm for the s-domain:

1. Select a control history $\boldsymbol{u}(s)$ in the s-interval $[s_0, s_f]$.
2. Solve the state equations in Eq. (5.16) for $\boldsymbol{x}(s)$ and the differential equation in Eq. (5.17) for $t(s)$, simultaneously.
3. Solve the adjoint differential equations in Eq. (5.23) for the adjoint variables $\boldsymbol{p}(s)$ backward in s.
4. Compute the control update $\delta\boldsymbol{u}(s)$ for every s by using Eq. (5.25). Note, instead of using the update $\delta\boldsymbol{u}(s)$ directly, Eq. (5.25) can be used to apply a quasi-Newton method.
5. Improve the control history by setting $\boldsymbol{u}(s) \to \boldsymbol{u}(s) + \delta\boldsymbol{u}(s)$. If κ was selected sufficiently small, the updated control $\boldsymbol{u}(s)$ reduces the cost functional in Eq. (5.18), in which the final time is given by $t_f = t(s_f)$.
6. Find a suitable factor κ, e. g., by using a line search algorithm.
7. Repeat steps 2 through 6 and stop the algorithm when $\delta\bar{J}$ in Eq. (5.24) is sufficiently small.

5.4 Examples

Two numerical examples from [21] are used, after revision, to apply the proposed methods for computing the time-optimal control for dynamic systems. For the first example, the adjoint method in a hybrid domain is used for trajectory planning of a space flight; while in the second example, the adjoint method in the s-domain is utilized for computing time-optimal vehicle maneuvers. Because of the consistent transformation, both presented methods allow the application of preferable quasi-Newton methods using the BFGS-update for the Hessian.

5.4.1 Orbital Transfer Problem

As a first example, we consider the time-optimal flight to the sphere of influence. The sphere of influence is a domain around a celestial body that indicates where the gravity of the body has a dominating effect on the dynamics of other space objects [67, Sec. 8.2]. We are interested in the optimal control of a space vehicle to reach the sphere of influence in minimal time. At $t = t_0$, the spacecraft is in a circular orbit of the Moon with known parameters. The maneuver under consideration is finished if $r(t)$ is equal to the radius r_{final} of the sphere of influence of the Moon. Hence, using Eq. (5.3), our terminal condition reads

$$r(t) = \Phi(\boldsymbol{x}(t)) \quad \text{and} \quad r(t_f) = r_{\text{final}}, \tag{5.26}$$

which defines t_f implicitly. We want to compute $u(r)$ such that the flight time to the sphere of influence is minimal.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Hence, neither involving a penalty function nor a control constraint, the cost functional is simply given by

$$J = \int_{t_0}^{t_f} 1 \, \mathrm{d}t, \tag{5.27}$$

i. e., the length of the time interval $t_f - t_0$. For finding the optimal control, we use the hybrid method described in Sec. 5.2 and determine the control u as a function of r, which corresponds to the coordinate s. Hence, if we substitute the time dependence in Eq. (4.43), i. e., $u(t) \to u(r)$, we can refer to the state equations in Eq. (4.43). The associated adjoint equations can be directly derived from Eq. (5.10) and read

$$\begin{aligned}
\dot{p}_1(t) &= \frac{p_2(t)v_t(t)}{r^2(t)} + p_3(t)\left[\frac{v_t^2(t)}{r^2(t)} - \frac{2\mu}{r^3(t)} - \frac{\mathrm{d}u(r)}{\mathrm{d}r}\frac{F}{m(t)}\cos(u(r))\right] \\
&\quad - p_4(t)\left[\frac{v_r(t)v_t(t)}{r^2(t)} - \frac{\mathrm{d}u(r)}{\mathrm{d}r}\frac{F}{m(t)}\sin(u(r))\right] \\
\dot{p}_2(t) &= 0 \\
\dot{p}_3(t) &= -p_1(t) + \frac{p_4(t)v_t(t)}{r(t)} \\
\dot{p}_4(t) &= -\frac{p_2(t)}{r(t)} - \frac{2p_3(t)v_t(t)}{r(t)} + \frac{p_4(t)v_r(t)}{r(t)} \\
\dot{p}_5(t) &= \frac{p_3(t)F}{m^2(t)}\sin(u(r)) + \frac{p_4(t)F}{m^2(t)}\cos(u(r)),
\end{aligned} \tag{5.28}$$

with the initial values $p_1(t_f) = -1/v_r(t_f)$ and $p_2(t_f) = p_3(t_f) = p_4(t_f) = p_5(t_f) = 0$. The formula for the control update, given by Eq. (5.14), yields

$$\delta u(r) = -\kappa \frac{F}{m(r)v_r(r)}\Big[p_3(r)\cos(u(r)) - p_4(r)\sin(u(r))\Big]. \tag{5.29}$$

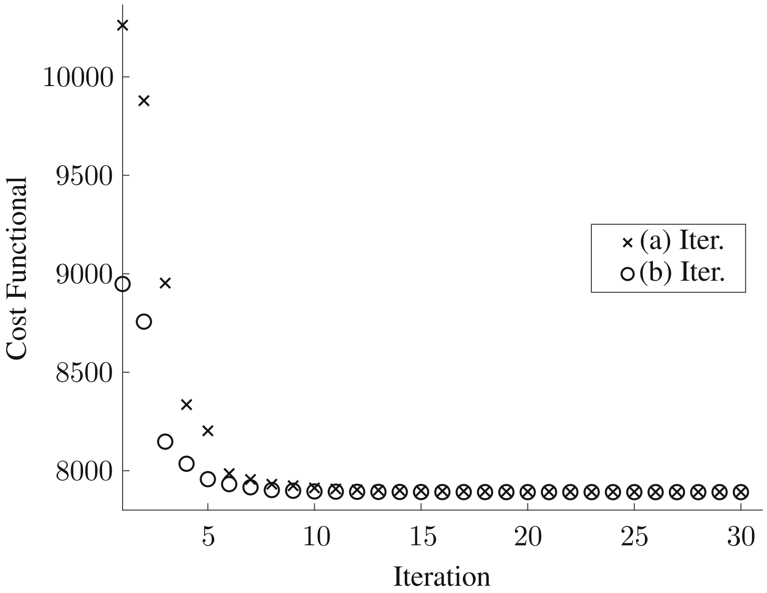

Figure 5.1: Convergence of the orbit transfer

For the numerical example, the following parameters are used: The initial mass of the vehicle $m_0 = 10.47\,\mathrm{t}$, the specific impulse $2.4 \times 10^3\,\mathrm{s}$, the thrust force $F = 16\,\mathrm{kN}$, the radius of the Moon $R = 1738\,\mathrm{km}$, Earth's gravitational acceleration $g = 9.8106 \times 10^{-3}\,\mathrm{km/s^2}$, Moon's gravitational acceleration $g_M = 1.622 \times 10^{-3}\,\mathrm{km/s^2}$, Moon's gravitational constant $\mu = R^2 g_M$ and the radius of the sphere of influence $r_{\text{final}} = 6.6 \times 10^4\,\mathrm{km}$. Initially, the space vehicle is in a circular Moon orbit with altitude $h = 111\,\mathrm{km}$. Hence, the initial conditions at $t = t_0 = 0$ are given by:

$$r(0) = R + h \quad \varphi(0) = 0 \quad v_r(0) = 0 \quad v_t(0) = \sqrt{\frac{\mu}{R+h}} \quad m(0) = m_0. \tag{5.30}$$

We consider two different initial controls, which are both far away from the optimal solution: (a) $u(r) = 0$ and (b) $u(r) = r\,\pi/(2r_{\text{final}})$. In the first case, the thrust always acts in tangential direction. In the latter case, the thrust acts tangentially at the beginning and radially at the end, i. e., the sphere of influence is crossed in normal direction. The controls $u(r)$ are discretized by 500 data points in time. Notice that a traditional gradient computation by numerical differentiation would require 500 additional forward solutions of the state equations, whereas with the proposed method the gradient is obtained already from one solution of the adjoint system equations. Figure 5.1 depicts the decrease of the final time with an increasing number of quasi-Newton iterations for both initial controls. The final time 1.0262×10^4 s for the initial control (a) and 8.9482×10^3 s for the initial control (b) are reduced to $t_f = 7.8913 \times 10^3$ s after the optimization process. The time-optimal control angles $u(r)$ computed from the initial controls (a) and (b) are shown in Fig. 5.2.

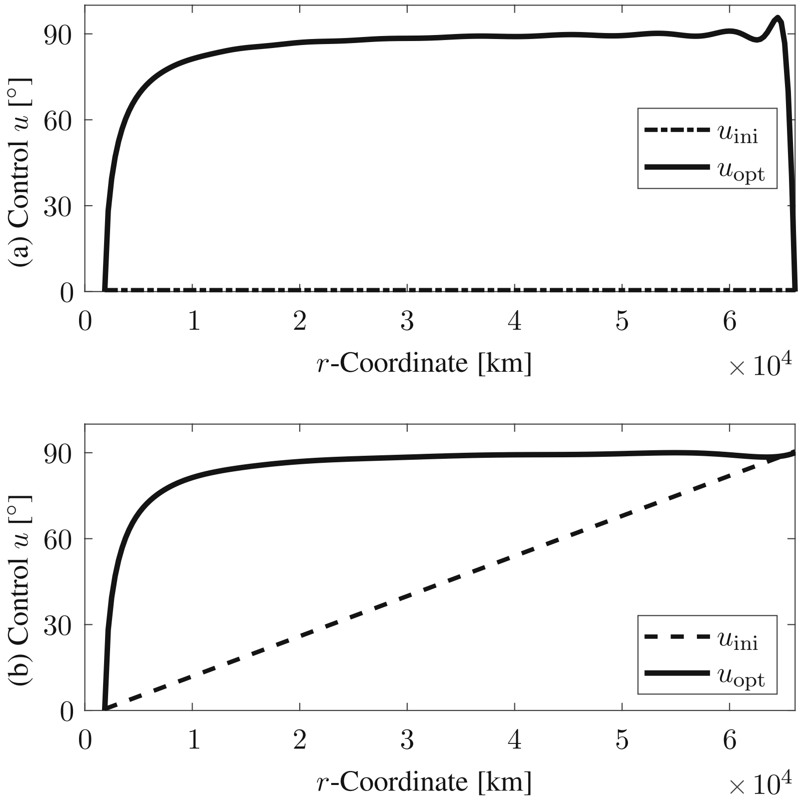

Figure 5.2: Control angle of the thrust nozzle

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

The flight trajectories of the space vehicle are plotted in Fig. 5.3. The optimized trajectories are almost identical for both initial controls showing that the spacecraft moves nearly on a straight flight path. Notice that for the initial control (a) some small oscillations remain in the control $u(r)$ at the end of the maneuver. The reason for this numerical artifact is that the final value $u(r_{\text{final}})$ is never updated by the gradient formula in Eq. (5.29) due to the zero values for p_3 and p_4 at $t = t_f$ and, hence, the control keeps its original value $u(r_{\text{final}}) = 0$. Clearly, the final control value has no influence on the flight time. For the initial control (a), the algorithm converges to a step function at $r = r_{\text{final}}$ which requires a higher number of iterations. However, as it can be seen from the convergence plot in Fig. 5.1, the cost functional no longer decreases already after 20 iterations. This can be explained by the fact that the control history close to $t = t_f$ has only small influence on the cost functional. Hence, the oscillations in $u(r)$ do not change the flight time and the flight path very much. However, Fig. 5.1 shows that choosing the initial control (b), for which the final value at $t = t_f$ coincides with the expected value $\pi/2$, increases the convergence rate and the oscillations at the end of the optimal control history in Fig. 5.2 vanish.

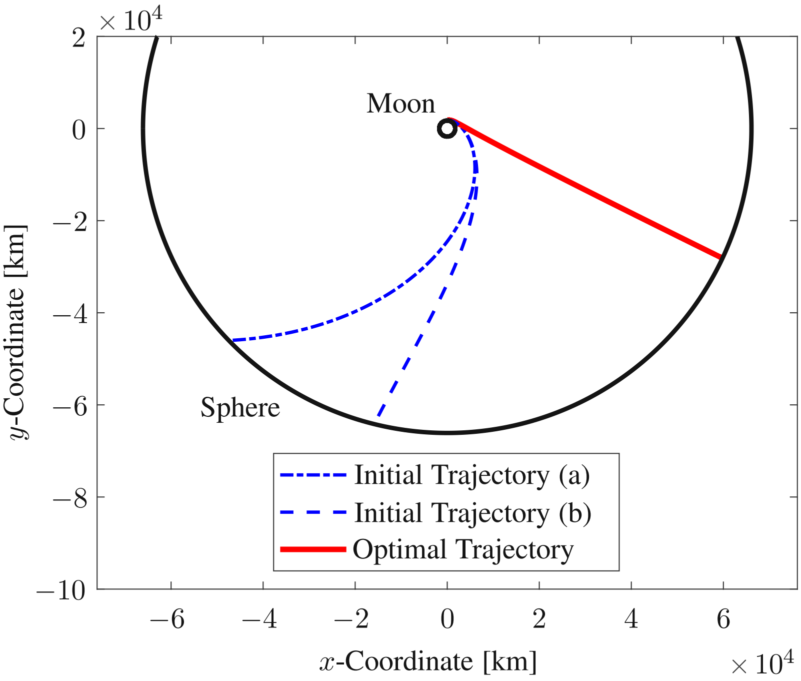

Figure 5.3: Time-optimal trajectory of the space vehicle

5.4.2 Minimum Lap Time Problem

As a second example, we consider a single track model of a car for which accelerating, braking and steering should be determined such that the time for a vehicle maneuver becomes minimal. A single track vehicle model is frequently studied in literature, as shown, e. g., in [65] for oversteer vehicles, to investigate the handling and stability characteristics of cars. However, from the point of view of optimal control theory, a classical question is how to minimize the end time, as also discussed in [53].

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

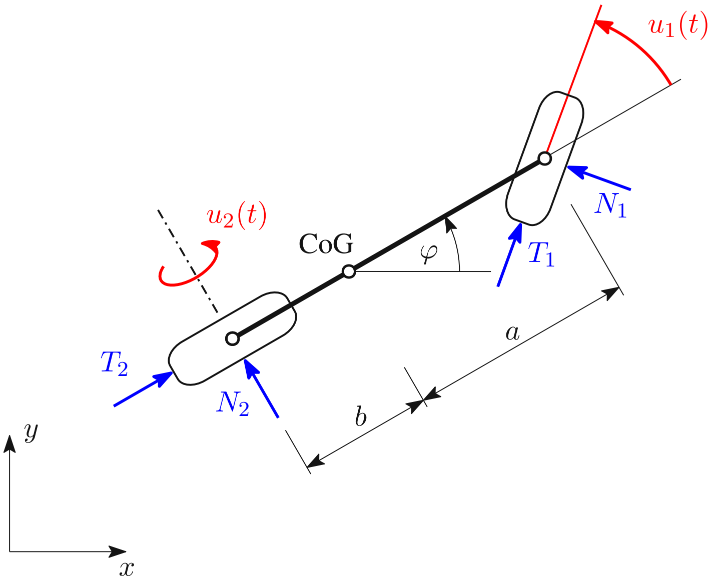

Figure 5.4: Single track vehicle model

We use the single track model in Fig. 5.4 and a Pacejka tire model for the wheel contact forces T_1, N_1, T_2 and N_2. The vehicle position is described by the coordinates (x, y) of the center of mass and by the orientation angle φ. The equations of motion are given by Eq. (A.2) in Appendix A. To formulate the terminal condition and the penalty function, forcing the car to stay within the boundaries of the road, we introduce the curvilinear coordinates (s, r), which describe the vehicle position by the distance s along the road center line and the deviation r measured normal to the center line. An elaborate derivation of the curvilinear coordinates is given in the Appendix A. With the curvilinear coordinate $s(t)$ and Eq. (5.3), the terminal condition, expressing that the vehicle crosses the finish line, is readily formulated as

$$s(t) = \Phi(\boldsymbol{x}(t)) \quad \text{and} \quad s(t_f) = s_{\text{final}}, \tag{5.31}$$

where s_{final} is the arc length coordinate of the finish line. Moreover, the penalty function for the road boundaries can be described by the lateral coordinate $r(t)$ as follows:

$$\Pi_1(r(t)) := \begin{cases} 0 & \text{for} \quad |r(t)| < \bar{r} \\ \frac{1}{2}\left(|r(t)| - \bar{r}\right)^2 & \text{for} \quad \bar{r} < |r(t)| < r^* \\ \frac{1}{2}\left(r^* - \bar{r}\right)^2 + \left(r^* - \bar{r}\right)\left(|r(t)| - r^*\right) & \text{otherwise.} \end{cases} \tag{5.32}$$

Here, $\bar{r} = w - h$, where w is the half road width and h is a small thickness parameter to activate the penalty function before the track constraint is violated. The penalty function is zero if $|r(t)| < \bar{r}$ and increases rapidly if $|r(t)| > \bar{r}$. Note that $\Pi_1(r(t))$ is quadratic in $r(t)$ if $\bar{r} < |r(t)| < r^*$, whereas it increases linearly if $|r(t)| > r^*$. Hence, $\Pi_1(r(t))$ is a once continuously differentiable function which is necessary to avoid jumps in the adjoint equations. A similar penalty function is

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

also used to constrain the driving torque $u_2(t)$:

$$\Pi_2(u_2(t)) := \begin{cases} 0 & \text{for} \quad |u_2(t)| < \bar{u}_2 \\ \frac{1}{2}\left(|u_2(t)| - \bar{u}_2\right)^2 & \text{for} \quad \bar{u}_2 < |u_2(t)| < u_2^* \\ \frac{1}{2}\left(u_2^* - \bar{u}_2\right)^2 + \left(u_2^* - \bar{u}_2\right)\left(|u_2(t)| - u_2^*\right) & \text{otherwise.} \end{cases} \tag{5.33}$$

Here, $\bar{u}_2(t)$ is the maximum drive torque provided by the powertrain and u_2^* separates the domains of quadratic and linear increase. Both penalty functions are weighted and combined to one resulting penalty function $\Pi(r(t), u_2(t)) = \alpha_1 \Pi_1(r(t)) + \alpha_2 \Pi_2(u_2(t))$. The cost functional to be minimized can then be defined by

$$\begin{aligned} J &= \int_{t_0}^{t_f} \left(1 + \Pi(r(t), u_2(t))\right) \mathrm{d}t \\ &= t_f - t_0 + \int_{s_0}^{s_f} \Pi(r(s), u_2(s))\, g(r(s), \varphi(s), v(s), w(s)) \mathrm{d}s, \end{aligned} \tag{5.34}$$

where $t_0 = 0$ and t_f is the minimal time to drive the vehicle through the road track. Here, g is the function introduced in Eq. (5.16), which is derived in Appendix A. For the solution of the problem posed above, we use the adjoint method in s-domain as described in Sec. 5.3. Notice that the boundaries for s are fixed, since we know $s = s_0$ for the initial state and $s = s_{\text{final}}$ for the terminal state when the finish line is crossed. The transformed first order state equations in the s-domain read

$$\begin{aligned} r'(s) &= \frac{g}{r(s)\, k(s) - 1} \Big[\Big(v(s) \cos(\varphi(s)) - w(s) \sin(\varphi(s)) \Big) \Big(x_c''(s) r(s) - y_c'(s) \Big) \\ &\qquad + \Big(v(s) \sin(\varphi(s)) + w(s) \cos(\varphi(s)) \Big) \Big(x_c'(s) + y_c''(s) r(s) \Big) \Big] \\ \varphi'(s) &= g\, \Omega(s) \\ v'(s) &= \frac{g}{m} \Big(m w(s) \Omega(s) + T_1 \cos(u_1(s)) + T_2 - N_1 \sin(u_1(s)) \Big) \\ w'(s) &= \frac{g}{m} \Big(-m v(s) \Omega(s) + N_1 \cos(u_1(s)) + N_2 + T_1 \sin(u_1(s)) \Big) \\ \Omega'(s) &= \frac{g}{J} \Big(N_1 a \cos(u_1(s)) - N_2 b + T_1 a \sin(u_1(s)) \Big) \\ \omega_1'(s) &= -\frac{R\, g}{J_1} T_1 \\ \omega_2'(s) &= \frac{g}{J_2} \Big(u_2(s) - T_2 R \Big), \end{aligned} \tag{5.35}$$

which are also derived in Appendix A. Moreover, we can add the differential equation $t'(s) = g$ to Eq. (5.35) from which we can determine $t_f = t(s_f)$. Here, $v(s)$ and $w(s)$ denote the components of the absolute vehicle velocity vector with respect to the chassis fixed reference frame. The yaw rate is given by $\Omega(s) = \dot{\varphi}(s)$ and $\omega_1(s)$ and $\omega_2(s)$ denote the angular velocities of the wheels. The

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

inertia parameters of the vehicle are given by m, J, J_1 and J_2. The wheelbase is given by $a+b$ and both wheels have the radius R. The control inputs of the model are the steering angle $u_1(s)$ of the front wheel and the driving/braking torque $u_2(s)$ of the rear wheel.

Note that the tire forces T_1, N_1, T_2 and N_2 have a decisive impact on the handling of the vehicle. However, with respect to the computational effort, we use a simplified Pacejka tire model proposed in [53], where the camber angle of the tire is neglected.

With the lateral and longitudinal slips

$$\begin{aligned}
\alpha_1 &= u_1(s) - \arctan\left(\frac{w(s)+\Omega(s)a}{v(s)}\right)\\
\kappa_1 &= \frac{\omega_1(s)R - v(s)\cos(u_1(s)) - (w(s)+\Omega(s)a)\sin(u_1(s))}{v(s)\cos(u_1(s)) + (w(s)+\Omega(s)a)\sin(u_1(s))}\\
\alpha_2 &= -\arctan\left(\frac{w(s)+\Omega(s)b}{v(s)}\right)\\
\kappa_2 &= \frac{\omega_2(s)R - v(s)}{v(s)},
\end{aligned} \tag{5.36}$$

for the front and the rear axis, the tire forces are given by

$$\begin{aligned}
T_1 &= \frac{G\,b}{a+b} f_t(\kappa_1)\, g_t(\kappa_1,\alpha_1) \qquad & N_1 &= \frac{G\,b}{a+b} f_n(\alpha_1)\, g_n(\alpha_1,\kappa_1)\\
T_2 &= \frac{G\,a}{a+b} f_t(\kappa_2)\, g_t(\kappa_2,\alpha_2) \qquad & N_2 &= \frac{G\,a}{a+b} f_n(\alpha_2)\, g_n(\alpha_2,\kappa_2),
\end{aligned} \tag{5.37}$$

in which G denotes the gravitational force on the vehicle. The Pacejka tire formulas[1] read

$$\begin{aligned}
f_i(x) &:= \mu_i \sin\Big(c_i \arctan\big(b_i x - a_i\left[b_i x - \arctan(b_i x)\right]\big)\Big)\\
g_i(x,y) &:= \cos\Big(k_i \arctan\big(d_i y \cos\left(\arctan\left(e_i x\right)\right)\big)\Big).
\end{aligned} \tag{5.38}$$

The numerical values for the Pacejka tire parameters related to the design properties are referenced from [53]. In summary, the parameters in longitudinal direction are given by: $a_t = 0.362$, $b_t = 11.1$, $c_t = 1.69$, $d_t = 12.4$, $e_t = -10.8$, $\mu_t = 1.20$ and $k_t = 1.09$. Moreover, the parameter set in lateral direction reads: $a_n = -1.11$, $b_n = 9.30$, $c_n = 1.19$, $d_n = 6.46$, $e_n = 4.20$, $\mu_n = 0.961$ and $k_n = 1.08$.

The adjoint equations associated to Eq. (5.35) which define the adjoint variables

$$\boldsymbol{p}(s) = \left(p_1, p_2, p_3, p_4, p_5, p_6, p_7\right)^{\mathsf{T}}, \tag{5.39}$$

to the state variables r, φ, v, w, Ω, ω_1, ω_2 can be derived from Eq. $(5.23)_1$. Due to rather lengthy expressions, we refrain from writing down the equations here. The final values for $\boldsymbol{p}(s)$ at $s = s_f$ are defined by Eq. $(5.23)_2$, i. e., by $\boldsymbol{p}(s_f) = \boldsymbol{0}$. Finally, an update formula for the controls is given

---

[1] The Pacejka formulas are often referred to as *"magic formulas"* in literature, since the high number of coefficients included have no physical meaning but rather describe a mathematical function that approximates the tire force law.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

by Eq. (5.25) and yields for the steering angle

$$\begin{aligned}\delta u_1(s) = -\,\kappa\Big[&-\frac{p_6\, g\, R}{J_1}\frac{\partial T_1}{\partial u_1} \\ &+ \frac{p_5\, g\, a}{J}\Big(T_1\cos(u_1) + \frac{\partial T_1}{\partial u_1}\sin(u_1) - N_1\sin(u_1) + \frac{\partial N_1}{\partial u_1}\cos(u_1)\Big) \\ &+ \frac{p_4\, g}{m}\Big(T_1\cos(u_1) + \frac{\partial T_1}{\partial u_1}\sin(u_1) - N_1\sin(u_1) + \frac{\partial N_1}{\partial u_1}\cos(u_1)\Big) \\ &+ \frac{p_3\, g}{m}\Big(-T_1\sin(u_1) + \frac{\partial T_1}{\partial u_1}\cos(u_1) - N_1\cos(u_1) - \frac{\partial N_1}{\partial u_1}\sin(u_1)\Big)\Big],\end{aligned} \tag{5.40}$$

and for the driving torque

$$\delta u_2(s) = -\kappa\left[\frac{\partial \Pi}{\partial u_2}g + \frac{p_7\, g}{J_2}\right]. \tag{5.41}$$

For the numerical computations, we consider a road center line which is composed of straight lines and clothoids. The vehicle starts at the center line of the track with the initial conditions

$$r(0) = 0 \quad v(0) = 12\,\frac{\mathrm{m}}{\mathrm{s}} \quad w(0) = 0 \quad \Omega(0) = 0 \quad \omega_1(0) = \frac{v(0)}{R} \quad \omega_2(0) = \frac{v(0)}{R}. \tag{5.42}$$

The vehicle parameters for the numerical computations can be summarized by $m = 1000\,\mathrm{kg}$, $J = 1750\,\mathrm{kg\,m^2}$, $J_w = 25\,\mathrm{kg\,m^2}$, $R = 0.3\,\mathrm{m}$, $a = 1.40\,\mathrm{m}$, $b = 1.35\,\mathrm{m}$ and $G = 9810\,\mathrm{N}$. Moreover, the penalty parameters are given by $w = 2\,\mathrm{m}$, $\alpha_1 = 2$, $\bar{r} = 1.7$, $r^* = 3$, $\alpha_2 = 6.19 \times 10^{-7}$ and $\bar{u}_2 = 1798$. In order to start the optimization procedure we have to find a reasonable initial guess for the control of the vehicle. Therefore, a simple driver model is applied, where the vehicle is forced to trace the road center line at a certain speed. Various authors have already discussed the problem of path planning strategies [19, 24], and the application of velocity profiles [19]. Since the focus is on the computation of time-optimal controls, we refrain from examining the theory of initial controls. It should be noted, however, that the proposed methods are robust with respect to the initial controls and are therefore applicable to less sophisticated initial controls.

(a) Single 90 Degree Curve

The goal of the first example is to identify the control of a vehicle driving through a single 90 degree curve. The s-domain of the curve is described using a parameter representation of straight lines and clothoids for the transition between two sections. A clothoid is a flat curve defined by the condition that the curvature decreases linearly with arc length. Hence, clothoids are used in road construction to achieve a smooth transition between roadway sections with different curvatures

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek, Your knowledge hub

The parameter representation of the lane centerline describing a 90 degree curve is given by

$$\vec{x}_c(s) = \begin{cases} \big(s,\, 0\big) & s \in [s_0, s_1[\\ \left(x_1 + \displaystyle\int_0^{s-s_1} \cos\big(A\,\sigma^2\big)\mathrm{d}\sigma,\ \displaystyle\int_0^{s-s_1} \sin\big(A\,\sigma^2\big)\mathrm{d}\sigma\right) & s \in [s_1, s_2[\\ \left(x_2 - \displaystyle\int_0^{s_3-s} \sin\big(A\,\sigma^2\big)\mathrm{d}\sigma,\ y_2 - \displaystyle\int_0^{s_3-s} \cos\big(A\,\sigma^2\big)\mathrm{d}\sigma\right) & s \in [s_2, s_3[\\ \big(x_3,\, y_3 + s - s_3\big) & s \in [s_3, s_f] \end{cases} \tag{5.43}$$

where $s_0 = 0$, $s_1 = 29$, $s_2 = 34$, $s_3 = 39$ and $s_f = 68$. The offset parameters are given by: $x_1 = 29$, $x_2 = 35$, $y_2 = 6$, $x_3 = 35$, $y_3 = 33$ and the clothoid parameters read $A = \pi/(4\,L^2)$ and $L = 5$.

Finally, the results of the time-optimal vehicle maneuver are summarized in Fig. 5.5, Fig. 5.6 and Fig. 5.7. In Fig. 5.5, the convergence of the cost functional is shown. After approximately 988 quasi-Newton iterations the total time can be reduced from $7.3742\,\mathrm{s}$ to $4.9504\,\mathrm{s}$. The optimization was stopped due to a sufficient small first-order optimality measure.

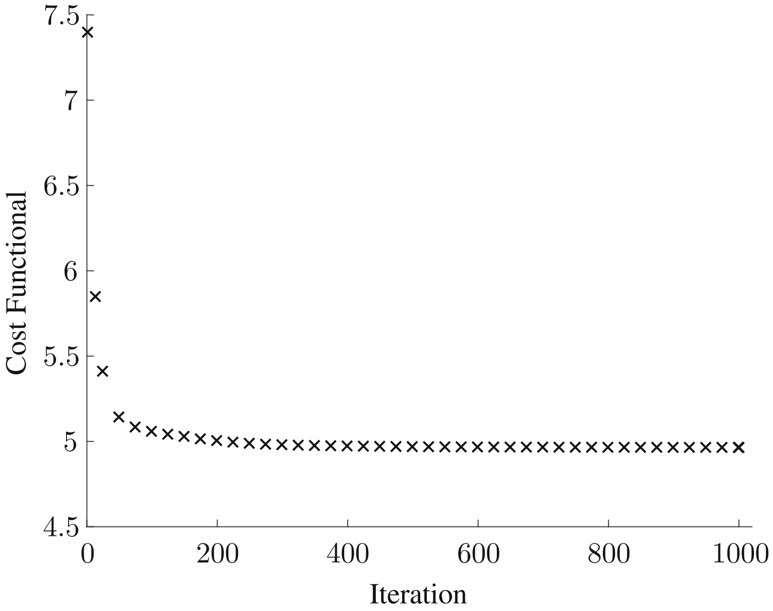

Figure 5.5: Convergence of the cost functional for driving through a 90 degree curve

The initial and the optimized trajectories are depicted in Fig. 5.6. The optimal control signals, i. e., the steering angle $u_{1,\mathrm{opt}}$ and the drive torque $u_{2,\mathrm{opt}}$ are compared to the initial controls $u_{1,\mathrm{ini}}$ and $u_{2,\mathrm{ini}}$ in Fig. 5.7. According to Pontryagin's minimum principle [54], see also [39, Sec. 5.4], we expect the following cases for the optimal drive torque, which appears linear in the Hamiltonian:

$$u_{2,\text{opt}}(s) := \begin{cases} u_{2,\max}, & p_7(s) < 0 \\ u_{2,\min}, & p_7(s) > 0 \\ \text{singular}, & p_7(s) = 0, \end{cases}$$

where $u_{2,\max} \simeq \bar{u}_2$ and $u_{2,\min} \simeq -\bar{u}_2$. If the adjoint variable $p_7(s)$ passes through zero we identify a switching point of the control $u_2(s)$. However, particular attention shall be paid to finite time intervals in which the adjoint variable $p_7(s)$ remains zero. Such intervals are called singular intervals. In that case, the necessary condition, which minimizes the Hamiltonian, provides no information about the optimal control. Here, the maximum transmitted force of the tire is the physical bound for the optimal control which is observed by the method itself. To circumvent the singular interval problem, usually the following traditional strategies are pursed: On the one hand, a singular control can be computed from differentiation of the switching condition $p_7 = 0$ and involving the state and costate equations, see [39, Sec. 5.6].

On the other hand, as a numerical approach, a regularization term can be added to the cost functional as described in Sec. 4.6. However, both strategies require either massive manipulations of the state and costate equations or an unphysical modification of the cost functional, whereas our method provides a natural approach to handle the singular case.

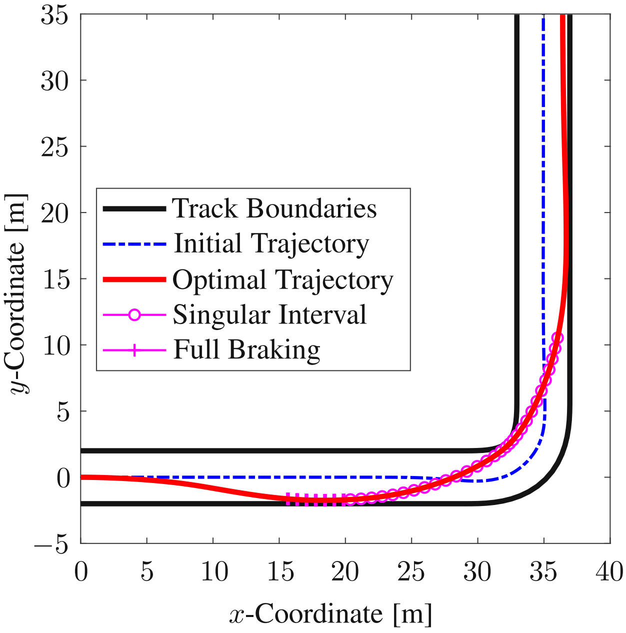

Figure 5.6: Comparison of the initial and the optimal trajectory for driving through a 90 degree curve

With the presented approach, the control for a singular interval can be detected without any further modification of the optimization process. Figure 5.7 shows that the numerical result is in agreement with the theoretical principle.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

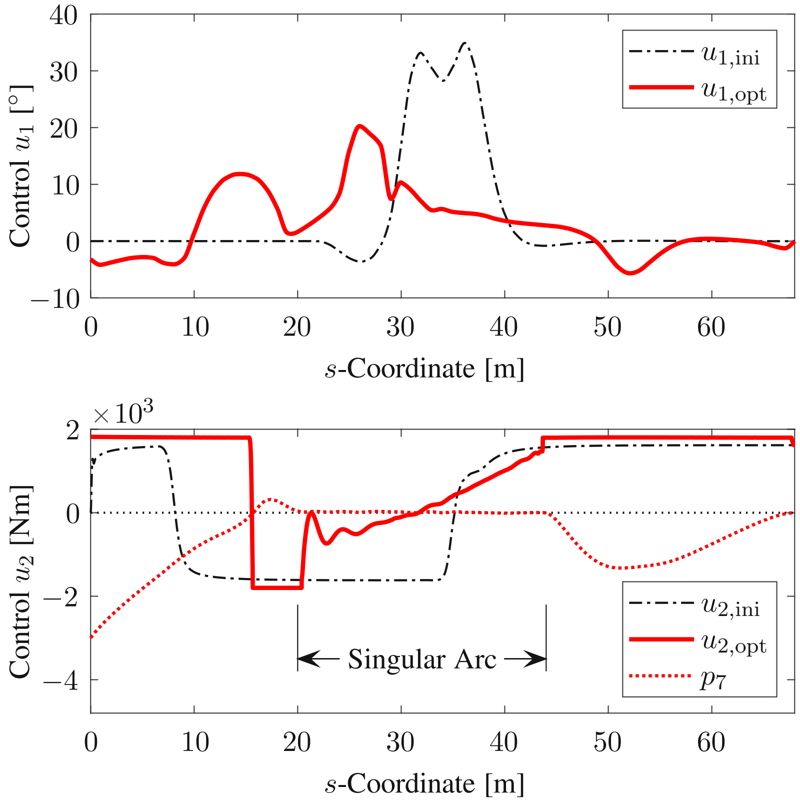

Figure 5.7: Comparison of the initial and the optimal control and verification of Pontryagin's minimum principle for driving through a 90 degree curve

(b) Hairpin Turn

Finally, we are looking for the time-optimal control of the single track vehicle driving through a hairpin turn. The description of the hairpin turn is given by the parameter representation:

$$\vec{x}_c(s) = \begin{cases} \left(s,\, 0\right) & s \in [s_0, s_1[\\ \left(x_1 + \displaystyle\int_0^{s-s_1} \cos\left(A\,\sigma^2\right)\mathrm{d}\sigma,\, \displaystyle\int_0^{s-s_1} \sin\left(A\,\sigma^2\right)\mathrm{d}\sigma\right) & s \in [s_1, s_2[\\ \left(x_3 - \displaystyle\int_0^{s_3-s} \sin\left(A\,\sigma^2\right)\mathrm{d}\sigma,\, y_3 - \displaystyle\int_0^{s_3-s} \cos\left(A\,\sigma^2\right)\mathrm{d}\sigma\right) & s \in [s_2, s_3[\\ \left(x_3 - \displaystyle\int_0^{s-s_3} \sin\left(A\,\sigma^2\right)\mathrm{d}\sigma,\, y_3 + \displaystyle\int_0^{s-s_3} \cos\left(A\,\sigma^2\right)\mathrm{d}\sigma\right) & s \in [s_3, s_4[\\ \left(x_4 + \displaystyle\int_0^{s_5-s} \cos\left(A\,\sigma^2\right)\mathrm{d}\sigma,\, y_4 - \displaystyle\int_0^{s_5-s} \sin\left(A\,\sigma^2\right)\mathrm{d}\sigma\right) & s \in [s_4, s_5[\\ \left(s_f - s,\, y_4\right) & s \in [s_5, s_f] \end{cases} \tag{5.44}$$

in which $s_0 = 0$, $s_1 = 25$, $s_2 = 31.3$, $s_3 = 37.6$, $s_4 = 43.9$, $s_5 = 50.2$ and $s_f = 75$. The coordinates are given by: $x_1 = 25$, $x_3 = 32.5$, $y_3 = 7.5$, $x_4 = 25$, and $y_4 = 15$.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

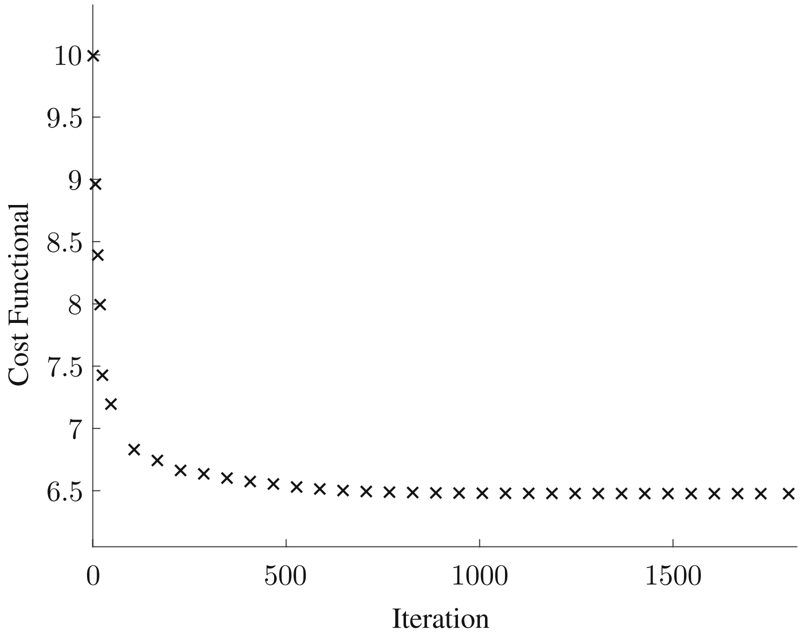

Figure 5.8: Convergence of the hairpin maneuver

Further, the clothoids for the road construction is defined by $A = \pi/(4\,L^2)$ and $L = 6.3$. The reduction of the cost functional over iterations is shown Fig. 5.8. After approximately 1804 quasi-Newton iterations we stop the optimization procedure. Note that for industrial applications in most cases a much lower number of iterations is sufficient, however, to demonstrate Pontryagin's theory on singular intervals a high accuracy is necessary. Finally, the total time can be reduced from $9.9182\,\mathrm{s}$ to $6.4712\,\mathrm{s}$.

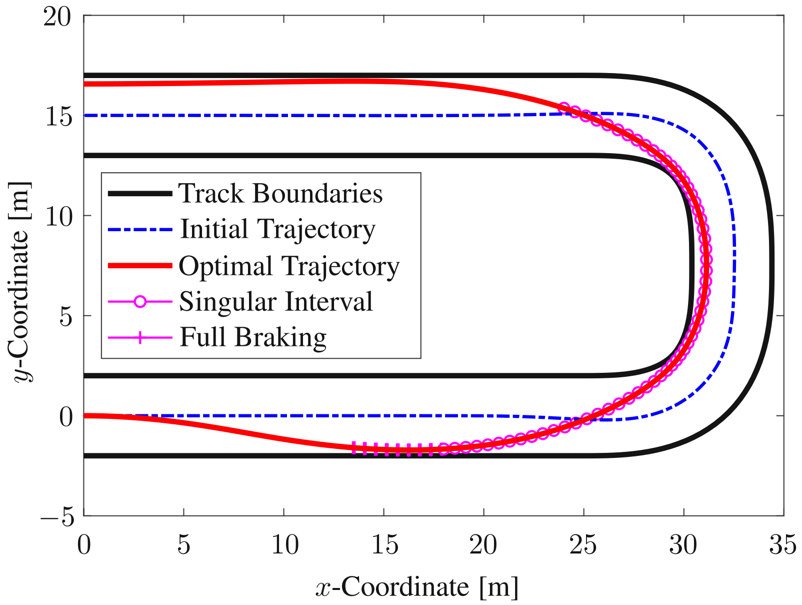

Figure 5.9: Comparison of the initial and the optimal trajectory for the hairpin maneuver

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

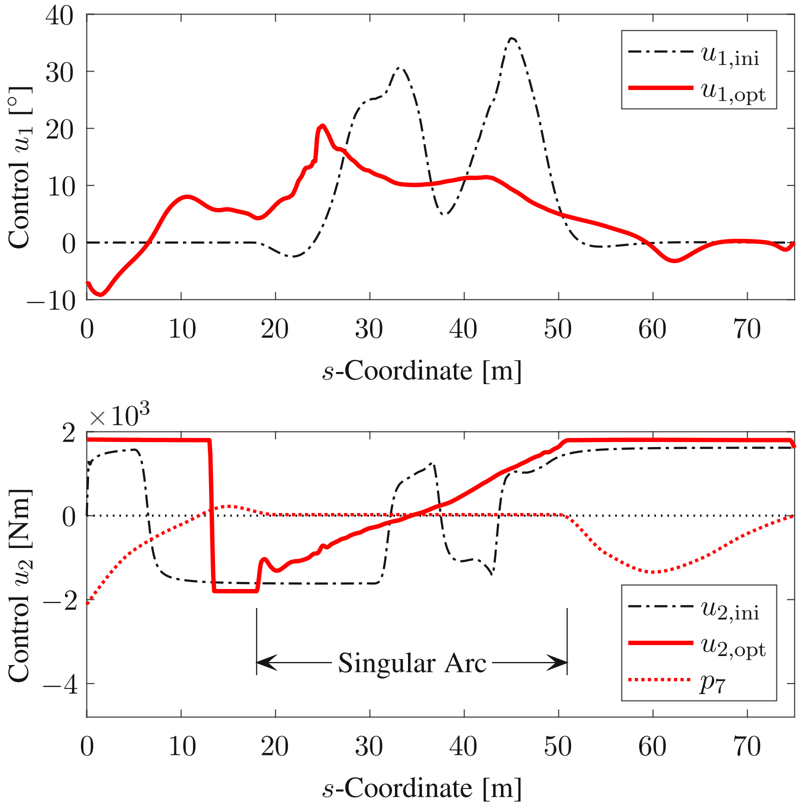

Figure 5.10: Comparison of the initial and the optimal control and verification of Pontryagin's minimum principle for the hairpin maneuver

In Fig. 5.9 we have once more established the result of the preprocessed driver model and the final solution of the time-optimal control problem. The control signals are shown in Fig. 5.10. In the latter diagram, we encounter the theory of Pontryagin, too, where the control behavior of the driving torque prevails in a similar manner as discussed before.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Chapter 6

The Adjoint Method for a Set of Specified Final States

It was already demonstrated that the adjoint method is a robust algorithm for determining the gradient of a cost functional. However, so far, we have only discussed a special class of time-optimal control problems, in which the final state of a system is given by a scalar condition. In this chapter, we are now concerned with solving problems with a given set of end conditions as described in [22]. In particular, we are interested in problems where a control has to be found such that a cost functional of the form as in Eq. (5.2) is minimized and the state variables at the final time t_f satisfy a set of (auxiliary) conditions.

In order to incorporate final constraints in the adjoint approach, additional penalty terms must be introduced in the cost functional. However, this may distort the optimal control due to weighting factors required for these terms and raise serious concerns about the magnitude of the weighting factors. Here, the modified gradient approach from Sec. 3.2.2 is the cornerstone upon which the method in this section is built. Following the fundamental ideas in applied optimal control presented in the book by Bryson and Ho [16], the adjoint method can be extended to account for end conditions by introducing additional adjoint-influence differential equations beside the classical adjoint system to associate the control variations with the final conditions, referring to [16, Sec. 7.4]. The key idea is to introduce a clever descent direction, as presented in Sec. 3.2.2, which reduces the error in the final conditions and approaches the constrained optimum of the cost functional within each iteration simultaneously. In many cases, time-optimal controls lead to so-called *bang-bang* controls. To improve the convergence rate of the proposed method in this case, we pay also attention to the efficient determination of the switching points with the adjoint method. The present chapter includes a revision of two published articles [22, 23].

6.1 Problem Formulation and Solution Approach

Instead of a scalar bound that terminates the integration scheme of the state equations, we now consider a vectorial condition that imposes the final state. We require that the system satisfies a

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

set of end conditions given by

$$\boldsymbol{\phi}(\boldsymbol{x}(t_f), t_f) = \mathbf{0}, \quad \boldsymbol{\phi} : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^r, \tag{6.1}$$

In the following, we first compute the gradient of the cost functional in Eq. (5.2) and then the gradient of the final conditions in Eq. (6.1), both with respect to the control signal. Afterwards, we consider how the gradients can be combined in a reasonable way.
As the first step, to compute the gradient of the cost functional, we expand J in Eq. (5.2) with the state equations, yielding

$$\bar{J} = \int_{t_0}^{t_f} \Big[1 + \Pi(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{p}^\mathsf{T} \left(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) - \dot{\boldsymbol{x}} \right) \Big] \, \mathrm{d}t. \tag{6.2}$$

Equipped with the knowledge from Eqs. (4.1–4.7), or respectively, Eq. (5.5), the resulting variation of the extended cost functional in Eq. (6.2) is given by

$$\begin{aligned} \delta\bar{J} = &\int_{t_0}^{t_f} \Big[\Big(\Pi_{\boldsymbol{u}} + \boldsymbol{p}^\mathsf{T} \boldsymbol{f}_{\boldsymbol{u}} \Big) \delta\boldsymbol{u} + \Big(\Pi_{\boldsymbol{x}} + \boldsymbol{p}^\mathsf{T} \boldsymbol{f}_{\boldsymbol{x}} + \dot{\boldsymbol{p}}^\mathsf{T} \Big) \delta\boldsymbol{x} \Big] \, \mathrm{d}t \\ &- \boldsymbol{p}^\mathsf{T}(t_f)\delta\boldsymbol{x}(t_f) + \big(1 + \Pi_f\big)\delta t_f, \end{aligned} \tag{6.3}$$

up to first order terms, where $\Pi_{\boldsymbol{x}}$, $\Pi_{\boldsymbol{u}}$, $\boldsymbol{f}_{\boldsymbol{x}}$ and $\boldsymbol{f}_{\boldsymbol{u}}$ denote the partial derivatives of Π and $\boldsymbol{f}$ with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$ and $\Pi_f = \Pi(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f))$. Since we have not made any restrictions on the vector of adjoint variables $\boldsymbol{p}(t)$ yet, we now define them by the linear time-variant final value problem

$$\dot{\boldsymbol{p}} = -\Pi_{\boldsymbol{x}}^\mathsf{T} - \boldsymbol{f}_{\boldsymbol{x}}^\mathsf{T} \boldsymbol{p} \quad \text{with} \quad \boldsymbol{p}(t_f) = \mathbf{0}, \tag{6.4}$$

which can be solved backward in time. In this case, the first variation of the cost functional in Eq. (6.3) reduces to

$$\delta\bar{J} = \int_{t_0}^{t_f} \Big(\Pi_{\boldsymbol{u}} + \boldsymbol{p}^\mathsf{T} \boldsymbol{f}_{\boldsymbol{u}} \Big) \delta\boldsymbol{u} \, \mathrm{d}t + \big(1 + \Pi_f\big)\delta t_f, \tag{6.5}$$

showing the direct influence of $\delta\boldsymbol{u}(t)$ and δt_f on $\delta\bar{J}$. In a similar way, the auxiliary conditions in Eq. (6.1) are extended by the state equations

$$\bar{\boldsymbol{\phi}} = \int_{t_0}^{t_f} \mathbf{P}^\mathsf{T} \big(\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) - \dot{\boldsymbol{x}} \big) \mathrm{d}t + \boldsymbol{\phi}(\boldsymbol{x}(t_f), t_f), \tag{6.6}$$

where we have introduced one more set of adjoint variables arranged in the matrix $\mathbf{P}(t) \in \mathbb{R}^{n \times r}$. Again, the integral term on the right side is zero for any matrix $\mathbf{P}(t)$, if the state equations are satisfied. Next, we compute the variation of $\bar{\boldsymbol{\phi}}$, yielding

$$\delta\bar{\boldsymbol{\phi}} = \int_{t_0}^{t_f} \mathbf{P}^\mathsf{T} \big(\boldsymbol{f}_{\boldsymbol{u}} \delta\boldsymbol{u} + \boldsymbol{f}_{\boldsymbol{x}} \delta\boldsymbol{x} - \delta\dot{\boldsymbol{x}} \big) \mathrm{d}t + \boldsymbol{\phi}_{\boldsymbol{x}}(\boldsymbol{x}(t_f), t_f) \, \delta\boldsymbol{x}_f + \boldsymbol{\phi}_t(\boldsymbol{x}(t_f), t_f) \, \delta t_f, \tag{6.7}$$

in which $\boldsymbol{\phi}_{\boldsymbol{x}}$ and $\boldsymbol{\phi}_t$ denote the partial derivatives of the function $\boldsymbol{\phi}(\boldsymbol{x}, t)$ with respect to $\boldsymbol{x}$ and t. Now recall that the variation $\delta\boldsymbol{x}(t_f)$ describes the variation of $\boldsymbol{x}$ at fixed end time t_f, whereas $\delta\boldsymbol{x}_f$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

is the variation of $\boldsymbol{x}$ including also the end time variation. According to Eq. (4.12), the relation between both is given by $\delta\boldsymbol{x}_f = \dot{\boldsymbol{x}}(t_f)\,\delta t_f + \delta\boldsymbol{x}(t_f)$. If we insert this into Eq. (6.7) and perform an integration by parts of $\mathbf{P}^\mathsf{T}\delta\dot{\boldsymbol{x}}$, we obtain

$$\begin{aligned} \delta\bar{\boldsymbol{\phi}} = &\int_{t_0}^{t_f} \left[\mathbf{P}^\mathsf{T}\boldsymbol{f}_{\boldsymbol{u}}\delta\boldsymbol{u} + \left(\mathbf{P}^\mathsf{T}\boldsymbol{f}_{\boldsymbol{x}} + \dot{\mathbf{P}}^\mathsf{T}\right)\delta\boldsymbol{x}\right]\mathrm{d}t \\ &+ \left(\boldsymbol{\phi}_{\boldsymbol{x}} - \mathbf{P}^\mathsf{T}\right)\Big|_{t_f}\delta\boldsymbol{x}(t_f) + \left(\boldsymbol{\phi}_{\boldsymbol{x}}\dot{\boldsymbol{x}} + \boldsymbol{\phi}_t\right)\Big|_{t_f}\delta t_f. \end{aligned} \tag{6.8}$$

Now we define $\mathbf{P}(t)$ by the matrix differential equations

$$\dot{\mathbf{P}} = -\boldsymbol{f}_{\boldsymbol{x}}^\mathsf{T}\mathbf{P} \quad \text{where} \quad \mathbf{P}(t_f) = \boldsymbol{\phi}_{\boldsymbol{x}}^\mathsf{T}(\boldsymbol{x}(t_f), t_f), \tag{6.9}$$

which can be integrated backward in time. For that purpose, we have to solve one set of n ordinary differential equations for each component of the final condition $\boldsymbol{\phi}(\boldsymbol{x}(t_f), t_f) = \mathbf{0}$. With $\mathbf{P}(t)$ from Eq. (6.9), and with

$$\dot{\boldsymbol{\phi}}_f := \left(\boldsymbol{\phi}_{\boldsymbol{x}}\dot{\boldsymbol{x}} + \boldsymbol{\phi}_t\right)\Big|_{t_f} = \left(\boldsymbol{\phi}_{\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\phi}_t\right)\Big|_{t_f}, \tag{6.10}$$

as the total time derivative of $\boldsymbol{\phi}$ at $t = t_f$, the variation in Eq. (6.8) reduces to

$$\delta\bar{\boldsymbol{\phi}} = \int_{t_0}^{t_f} \mathbf{P}^\mathsf{T}\boldsymbol{f}_{\boldsymbol{u}}\delta\boldsymbol{u}\,\mathrm{d}t + \dot{\boldsymbol{\phi}}_f\,\delta t_f, \tag{6.11}$$

showing again the direct influence of $\delta\boldsymbol{u}(t)$ and δt_f on $\delta\bar{\boldsymbol{\phi}}$. Like Eq. (3.25) for a constrained static optimization problem, Eq. (6.5) and Eq. (6.11) describe the variations of the cost functional and the auxiliary condition in the case of an optimal control problem with varying end time and constraints on the final state. Similarly, a descent direction of the optimal control problem is also obtained by choosing

$$\begin{aligned} \delta t_f &= -\kappa\alpha\left(1 + \Pi_f + \boldsymbol{\nu}^\mathsf{T}\dot{\boldsymbol{\phi}}_f\right) \\ \delta\boldsymbol{u}(t) &= -\kappa\left(\Pi_{\boldsymbol{u}}^\mathsf{T} + \boldsymbol{f}_{\boldsymbol{u}}^\mathsf{T}\boldsymbol{p} + \boldsymbol{f}_{\boldsymbol{u}}^\mathsf{T}\mathbf{P}\,\boldsymbol{\nu}\right), \end{aligned} \tag{6.12}$$

where the number α serves for conditioning the problem and κ can be interpreted as step size for a time and a control update which have to be selected appropriately. Moreover, $\boldsymbol{\nu} \in \mathbb{R}^r$ is a vector of multipliers including weighted associations between $\delta\bar{J}$ and $\delta\bar{\boldsymbol{\phi}}$, which can be determined as follows: Since we claim that the variations $\delta\boldsymbol{u}(t)$ and δt_f should result in a better approximation of the condition $\boldsymbol{\phi}(\boldsymbol{x}(t_f), t_f) = \mathbf{0}$, we set

$$\delta\bar{\boldsymbol{\phi}} := -\varepsilon\,\boldsymbol{\phi}(\boldsymbol{x}(t_f), t_f), \quad \text{where} \quad 0 < \varepsilon < 2. \tag{6.13}$$

In order to compute $\boldsymbol{\nu}$, we equate Eq. (6.11) with Eq. (6.13) and insert Eq. (6.12) for $\delta\boldsymbol{u}$ and δt_f.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Introducing the abbreviations

$$\begin{aligned} \mathbf{A} &:= \int_{t_0}^{t_f} \mathbf{P}^{\mathsf{T}} \boldsymbol{f}_{\boldsymbol{u}} \boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}} \mathbf{P} \,\mathrm{d}t + \alpha\, \dot{\boldsymbol{\phi}}_f \dot{\boldsymbol{\phi}}_f^{\mathsf{T}} && \in \mathbb{R}^{r \times r} \\ \boldsymbol{b} &:= \int_{t_0}^{t_f} \mathbf{P}^{\mathsf{T}} \boldsymbol{f}_{\boldsymbol{u}} \left(\Pi_{\boldsymbol{u}}^{\mathsf{T}} + \boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}} \boldsymbol{p} \right) \mathrm{d}t + \alpha\, \dot{\boldsymbol{\phi}}_f \left(1 + \Pi_f \right) && \in \mathbb{R}^{r} \end{aligned} \tag{6.14}$$

we finally obtain

$$\boldsymbol{\nu} = \frac{\varepsilon}{\kappa} \mathbf{A}^{-1} \boldsymbol{\phi}(\boldsymbol{x}(t_f), t_f) - \mathbf{A}^{-1} \boldsymbol{b}, \tag{6.15}$$

which is used in the update formulas in Eq. (6.12) to reduce the deviation from the end condition. It has to be mentioned that the presented derivations follow the basic ideas in [16, Sec. 7.4]. At a first glance, it may seem that this version of the adjoint method in this chapter is the answer to all of our problems, since it can also be used to deal with scalar terminal conditions as discussed in Chapter 5. However, a decisive disadvantage is that the underlying cost functional can not longer be used to determine the step size update and thus the convergence rate suffers compared to the customized methods in the previous chapter.

An Introductory Example

As an introductory example we consider the Brachistochrone problem, discussed in Sec. 2.3, formulated as an optimal control task [22]. The cost functional to be minimized is given by

$$J = \int_{t_0}^{t_f} 1 \,\mathrm{d}t, \tag{6.16}$$

which is simply the length of the time interval. The absolute motion of the particle mass is described by the coordinates $x(t)$, $y(t)$ and the velocity $v(t)$, which is measured tangential to the wire. If we introduce the state vector

$$\boldsymbol{x}(t) = \left(x, y, v\right)^{\mathsf{T}}, \tag{6.17}$$

the state equations are given by

$$\begin{aligned} \dot{x}(t) &= v(t) \sin(u(t)) \\ \dot{y}(t) &= v(t) \cos(u(t)) \\ \dot{v}(t) &= g \cos(u(t)), \end{aligned} \tag{6.18}$$

where g is the gravitational acceleration and $u(t)$ is the control input defining the local slope angle of the wire. The initial values of the problem at $t_0 = 0$ read

$$x(0) = 0 \quad y(0) = 0 \quad v(0) = 0, \tag{6.19}$$

and the final conditions at the final time t_f are given by

$$x(t_f) = x_f \quad y(t_f) = y_f, \tag{6.20}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

where $v(t_f) =$ free. Thus, the set of auxiliary conditions can be collected in a vector:

$$\boldsymbol{\phi}(x,y) := \begin{pmatrix} x(t) - x_f \\ y(t) - y_f \end{pmatrix}. \tag{6.21}$$

Since we have no state constraints in the form of a penalty function, the adjoint variables $\boldsymbol{p}(t)$ in Eq. (6.4) do not enter the gradient formula and only the adjoint-influence system has to be solved. The adjoint-influence differential equations from Eq. (6.9) are then given by

$$\begin{aligned} &\dot{\mathbf{P}}(t) = \\ &-\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ P_{11}(t)\sin(u(t)) + P_{21}(t)\cos(u(t)) & P_{12}(t)\sin(u(t)) + P_{22}(t)\cos(u(t)) \end{pmatrix}, \end{aligned} \tag{6.22}$$

which may be solved backward in time by starting at $t = t_f$. Moreover, the final conditions are simply given by

$$\mathbf{P}(t_f) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}. \tag{6.23}$$

Following the algorithm, the components of the matrix $\mathbf{A} = (A_{ij}) \in \mathbb{R}^{r\times r}$ can be determined by Eq. $(6.14)_1$, yielding

$$\begin{aligned} A_{11} &= \int_{t_0}^{t_f} \Big[P_{11} v\cos(u) - \Big(P_{21} v + P_{31} g\Big)\sin(u)\Big]^2 \mathrm{d}\tau + \alpha\, v^2(t_f)\sin^2(u(t_f)) \\ A_{12} &= \int_{t_0}^{t_f} \Big[P_{11} v\cos(u) + \Big(P_{21} v - P_{31} g\Big)\sin(u)\Big] \\ &\qquad \Big[P_{11} v\cos(u) + \Big(P_{21} v - P_{31} g\Big)\sin(u)\Big]\mathrm{d}\tau + \alpha\, v^2(t_f)\sin(u(t_f))\cos(u(t_f)) \\ A_{22} &= \int_{t_0}^{t_f} \Big[P_{12} v\cos(u) - \Big(P_{22} v + P_{32} g\Big)\sin(u)\Big]^2 \mathrm{d}\tau + \alpha\, v^2(t_f)\cos^2(u(t_f)), \end{aligned} \tag{6.24}$$

where $A_{12} = A_{21}$. Since the adjoint variables $\boldsymbol{p}(t)$ remain zero for $t \in [t_0, t_f]$, we obtain $\boldsymbol{b} = \alpha(v(t_f)\sin(u(t_f)), v(t_f)\cos(u(t_f)))^{\mathsf{T}}$ by Eq. $(6.14)_2$.

In a next step we relate the gradients of the final conditions, the final time and the controls. Therefore, we compute the multiplier $\boldsymbol{\nu} = (\nu_1, \nu_2)^{\mathsf{T}}$ from Eq. (6.15) as

$$\begin{aligned} \nu_1 &= \frac{\varepsilon}{\kappa}\left(\frac{A_{22}\,(x(t_f) - x_f) - A_{12}\,(y(t_f) - y_f)}{A_{11}A_{22} - A_{21}A_{12}}\right) - \frac{A_{22}b_1 - A_{12}b_2}{A_{11}A_{22} - A_{21}A_{12}} \\ \nu_2 &= \frac{\varepsilon}{\kappa}\left(\frac{A_{11}\,(y(t_f) - y_f) - A_{21}\,(x(t_f) - x_f)}{A_{11}A_{22} - A_{21}A_{12}}\right) - \frac{A_{11}b_2 - A_{21}b_1}{A_{11}A_{22} - A_{21}A_{12}}. \end{aligned} \tag{6.25}$$

Finally, the updates of the final time t_f and the control history $u(t)$ are given by

$$\begin{aligned}\delta t_f &= -\kappa\alpha\Big[1+\nu_1 v(t_f)\sin(u(t_f))+\nu_2 v(t_f)\cos(u(t_f))\Big]\\ \delta u(t) &= -\kappa\Big[v(t)\cos(u(t))\Big(P_{11}(t)\nu_1+P_{12}(t)\nu_2\Big)\\ &\qquad -v(t)\sin(u(t))\Big(P_{21}(t)\nu_1+P_{22}(t)\nu_2\Big)\\ &\qquad -g\sin(u(t))\Big(P_{31}(t)\nu_1+P_{32}(t)\nu_2\Big)\Big].\end{aligned} \tag{6.26}$$

We prescribe the final conditions with $x_f = 5$ and $y_f = -2$. As initial guess we choose the final time $t_f = 2$ and we simply select the trivial control history $u(t) = 0$ with $t \in [t_0, t_f]$, which corresponds to free fall of a particle mass. Note that there are no special requirements on the selection of an initial control. The variation of the final time is scaled with $\alpha = 0.01$ and the update parameter and the step size are set to $\varepsilon = 0.5$ and $\kappa = 1$.

The convergence plot is depicted in Fig. 6.1 and shows that after 30 iterations the final time is given by $1.2931\,\mathrm{s}$ and the end point error is close to zero. On closer inspection, it is noticeable that the end time first decreases and then slightly increases again. This can be explained by the fact that the initial sharp decrease in the end time does not permit the end conditions to be completely fulfilled and thus the end time increases again. In order to show that the computed solution represents the optimal path, we compare the result with the analytical solution of the Brachistochrone problem in Sec. 2.3. The trajectory in Fig. 6.2 is in perfect agreement with the analytical solution of the Brachistochrone problem.

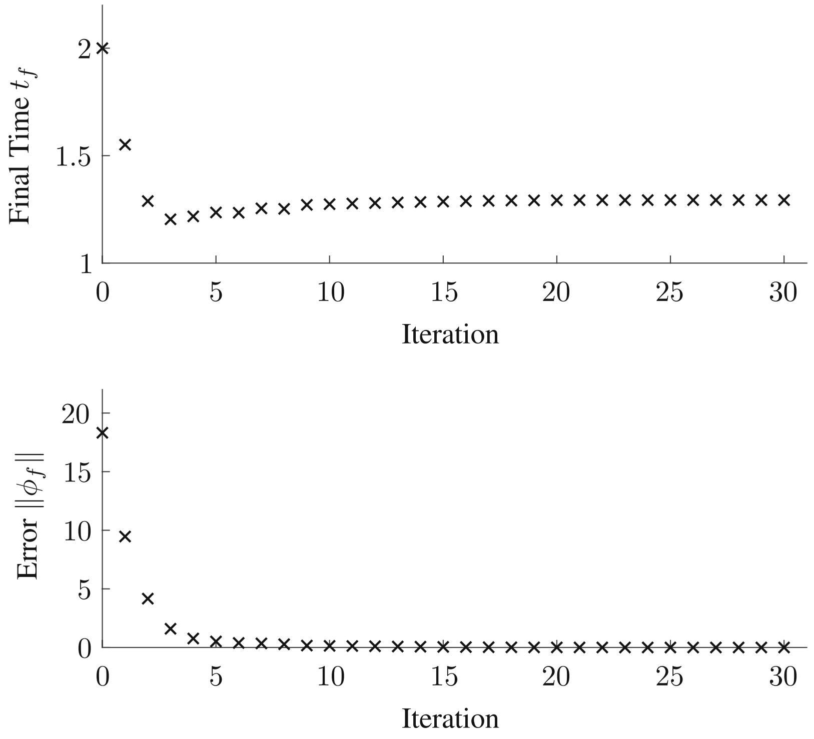

Figure 6.1: Convergence for the Brachistochrone optimal control problem

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

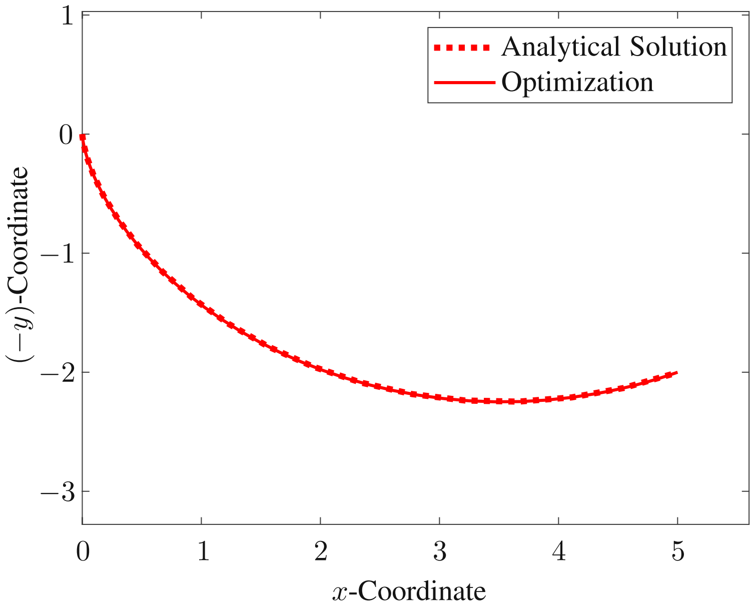

Figure 6.2: Trajectory of the Brachistochrone optimal control problem

6.2 Bang-Bang Principle

In the introductory example above, the control occurs nonlinearly in Eq. (6.18). However, if the control appears linearly in the state equations and therefore also linear in the Hamiltonian, two cases can be considered: either the control is singular or of *bang-bang* type, see [54]. Here, we want to pay attention to the non-singular case, considering bounded (linear) inputs where only the switching points are unknown. In order to make statements about the control, we introduce the Hamiltonian:

$$\mathcal{H}(\boldsymbol{x}(t),\boldsymbol{u}(t),\boldsymbol{p}(t),\mathbf{P}(t)) := 1 + \Pi(\boldsymbol{x}(t),\boldsymbol{u}(t)) + \big(\boldsymbol{p}(t)+\mathbf{P}(t)\boldsymbol{\nu}\big)^{\mathsf{T}}\boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{u}(t)), \tag{6.27}$$

including both sets of adjoint variables from Eq. (6.4) and Eq. (6.9). On the one hand, the presented adjoint method is an efficient way to compute the direction of the steepest descent of a cost functional. On the other hand, the adjoint variables can be investigated to evaluate the optimality conditions regarding the Hamiltonian function as described in [16, Sec. 7.4]. Now we focus on the latter mentioned role of the adjoint variables in order to introduce a switching function for the efficient computation of bang-bang controls. Following the minimum principle, see also [39, Sec. 5.4], and introducing the switching function

$$h_i(t) := \boldsymbol{f}_{u_i}^{\mathsf{T}}\big(\boldsymbol{p}(t)+\mathbf{P}(t)\boldsymbol{\nu}\big), \tag{6.28}$$

the bang-bang control law reads

$$u_{i,\text{opt}}(t) := \begin{cases} u_{i,\max}, & h_i(t) < 0 \\ u_{i,\min}, & h_i(t) > 0 \\ \text{singular}, & h_i(t) = 0. \end{cases} \tag{6.29}$$

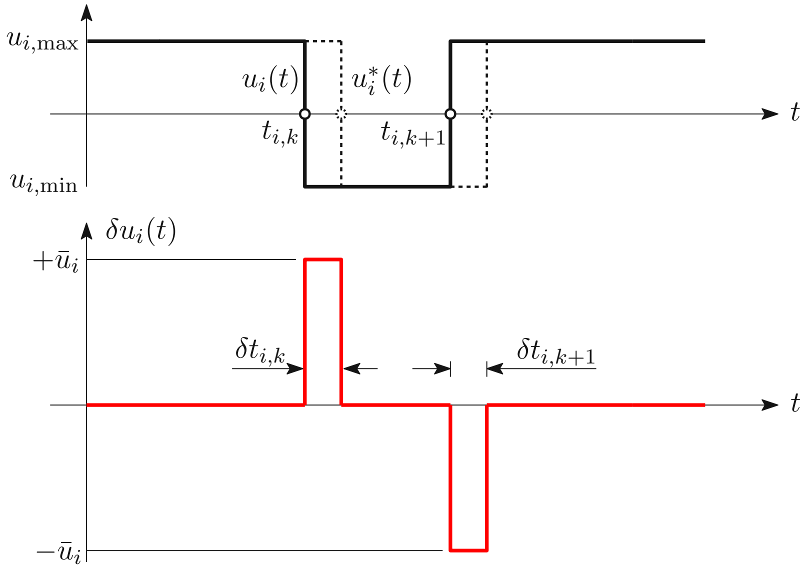

Figure 6.3: Influence of the variation of switching times on bang-bang controls

If the function $h_i(t)$ passes through zero, the control signal u_i changes from one extreme (admissible) value to the other. A variation $\delta u_i(t)$ of the bang-bang control $u_i(t)$, results from a variation of the switching times $t = t_{i,1} \ldots t_{i,N_i}$, where N_i is the total number of switching points associated to u_i. The maximum number of switching times N_i can be determined if we consider a linear, stationary system. If the eigenvalues of the $n \times n$ coefficient matrix of the states are all real and non-positive then each control switches at most $n-1$ times, see [39, Sec. 5.4]. In order to determine the exact number N_i for nonlinear systems, a solution of the whole control history is required first. Afterwards the solution can be recomputed by introducing switching times to achieve a higher accuracy. Since either $u_i = u_{i,\min}$ or $u_i = u_{i,\max}$, a disturbed control $u_i^*(t)$ will look like the dashed line in the upper diagram of Fig. 6.3. The difference between $u_i^*(t)$ and the original signal $u_i(t)$ is shown in the lower diagram of Fig. 6.3 and is defined by the shift $\delta t_{i,k}$ of the switching points as follows:

$$\delta u_i(t) = \begin{cases} \pm \bar{u}_i & \text{for } t \in [t_{i,k};\, t_{i,k} + \delta t_{i,k}] \\ 0 & \text{otherwise,} \end{cases} \tag{6.30}$$

where $\bar{u}_i = u_{i,\max} - u_{i,\min}$. The negative sign of $\bar{u}_i$ has to be taken if the control switches from $u_{i,\min}$ to $u_{i,\max}$ and the positive sign applies for a switch from $u_{i,\max}$ to $u_{i,\min}$ at $t_{i,k}$. The bang-bang controls are now numbered with $i = 1, \ldots l$, and the remaining controls with $i = l+1, \ldots m$. Let $\delta t_{i,k}$ be an infinitesimal small variation. Then, inserting Eq. (6.30) into Eq. (6.5) and into Eq. (6.11) yields

$$\begin{aligned} \delta \bar{J} &= \sum_{i=1}^{l} \sum_{k=1}^{N_i} \pm \bar{u}_i \left(\boldsymbol{p}^\mathsf{T} \boldsymbol{f}_{u_i} \right) \Big|_{t_{i,k}} \delta t_{i,k} + \sum_{i=l+1}^{m} \int_{t_0}^{t_f} \left(\Pi_{u_i} + \boldsymbol{p}^\mathsf{T} \boldsymbol{f}_{u_i} \right) \delta u_i \,\mathrm{d}t + (1 + \Pi_f)\, \delta t_f \\ \delta \bar{\boldsymbol{\phi}} &= \sum_{i=1}^{l} \sum_{k=1}^{N_i} \pm \bar{u}_i \left(\mathbf{P}^\mathsf{T} \boldsymbol{f}_{u_i} \right) \Big|_{t_{i,k}} \delta t_{i,k} + \sum_{i=l+1}^{m} \int_{t_0}^{t_f} \mathbf{P}^\mathsf{T} \boldsymbol{f}_{u_i} \delta u_i \,\mathrm{d}t + \dot{\boldsymbol{\phi}}_f \, \delta t_f \,. \end{aligned} \tag{6.31}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Note that we assumed that the penalty function Π depends only on the variables $u_{l+1}, \dots u_m$. Hence, the derivative Π_{u_i} was set to zero for the bang-bang controls. The descent direction for the switching points is now chosen analogously to Eq. (6.12). Hence, we set

$$\delta t_{i,k} = \mp\kappa\,\beta\bar{u}_i\Big(\boldsymbol{f}_{u_i}^{\mathsf{T}}\boldsymbol{p} + \boldsymbol{f}_{u_i}^{\mathsf{T}}\mathbf{P}\,\boldsymbol{\nu}\Big)\Big|_{t=t_{i,k}}, \tag{6.32}$$

in which κ is again the gradient step size for $\delta t_{i,k}$ and β is a conditioning parameter, analogously to α. For the computation of the multiplier $\boldsymbol{\nu}$, we insert Eq. (6.32) for the bang-bang controls and Eq. (6.12) for the final time and the remaining controls into Eq. $(6.31)_2$ and set $\delta\bar{\boldsymbol{\phi}} = -\varepsilon\boldsymbol{\phi}$. Using the modified abbreviations

$$\mathbf{A} := \beta\sum_{i=1}^{l}\sum_{k=1}^{N_i}\bar{u}_i^2\,\mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{u_i}\boldsymbol{f}_{u_i}^{\mathsf{T}}\mathbf{P}\Big|_{t_{i,k}} + \sum_{i=l+1}^{m}\int_{t_0}^{t_f}\mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{u_i}\boldsymbol{f}_{u_i}^{\mathsf{T}}\mathbf{P}\,\mathrm{d}t + \alpha\,\dot{\boldsymbol{\phi}}_f\dot{\boldsymbol{\phi}}_f^{\mathsf{T}}$$

$$\boldsymbol{b} := \beta\sum_{i=1}^{l}\sum_{k=1}^{N_i}\bar{u}_i^2\mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{u_i}\boldsymbol{f}_{u_i}^{\mathsf{T}}\boldsymbol{p}\Big|_{t_{i,k}} + \sum_{i=l+1}^{m}\int_{t_0}^{t_f}\mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{u_i}\left(\Pi_{u_i}^{\mathsf{T}} + \boldsymbol{f}_{u_i}^{\mathsf{T}}\boldsymbol{p}\right)\mathrm{d}t + \alpha\,\dot{\boldsymbol{\phi}}_f\,(1+\Pi_f)\,,$$

the multipliers $\boldsymbol{\nu}$ can be again obtained from Eq. (6.15) in case bang-bang controls appear.

6.3 The Algorithm

Finally, we summarize the steps of the proposed adjoint gradient approach to compute time-optimal controls. Since the final time t_f is free, a transformation into a unit interval is useful. Therefore, a new time coordinate $\tau \in [0,1]$ is introduced, with $t = t_f\,\tau$. Derivatives with respect to τ are denoted as $(\cdot)'$ and given by $(\cdot)' = t_f\,\mathrm{d}(\cdot)/\mathrm{d}t$. The procedure to solve time-optimal control problems with specified final values by the method of descent can be summarized as follows:

1. Select an initial control history $\boldsymbol{u}(\tau)$ or switching points $t_{i,k}$ for bang-bang controls and define a final time t_f.

2. Solve the state equations
$$\boldsymbol{x}'(\tau) = t_f\;\boldsymbol{f}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau)), \tag{6.33}$$
with the initial condition $\boldsymbol{x}(0) = \boldsymbol{x}_0$ in the time interval $\tau \in [0,1]$.

3. Solve the adjoint equations
$$\boldsymbol{p}'(\tau) = -t_f\left(\Pi_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau)) + \boldsymbol{f}_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau))\boldsymbol{p}(\tau)\right), \tag{6.34}$$
with the final condition $\boldsymbol{p}(1) = \mathbf{0}$ starting at $\tau = 1$ to $\tau = 0$.

4. Solve the adjoint-influence matrix differential equations
$$\mathbf{P}'(\tau) = -t_f\boldsymbol{f}_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau))\mathbf{P}(\tau) \quad \text{with} \quad \mathbf{P}(1) = \boldsymbol{\phi}_{\boldsymbol{x}}^{\mathsf{T}}(\boldsymbol{x}(t_f), t_f), \tag{6.35}$$
from $\tau = 1$ to $\tau = 0$.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

5. Compute the multiplier $\boldsymbol{\nu}$ from Eq. (6.15).
6. Update the controls and the final time by adding $\delta\boldsymbol{u}(\tau)$ and δt_f to the previous estimates of $\boldsymbol{u}(\tau)$ and t_f, where

$$\begin{aligned} \delta t_f &= -\kappa\,\alpha\left(1 + \Pi_f + \boldsymbol{\nu}^{\mathsf{T}}\dot{\boldsymbol{\phi}}_f\right) \\ \delta\boldsymbol{u}(\tau) &= -\kappa\left(\Pi_{\boldsymbol{u}}^{\mathsf{T}} + \boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}}\boldsymbol{p} + \boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}}\mathbf{P}\boldsymbol{\nu}\right), \end{aligned} \tag{6.36}$$

   or, respectively, for bang-bang controls the update for $t_{i,k}$ is given by

$$\delta t_{i,k} = \mp\kappa\,\beta\bar{u}_i\left(\boldsymbol{f}_{u_i}^{\mathsf{T}}\boldsymbol{p} + \boldsymbol{f}_{u_i}^{\mathsf{T}}\mathbf{P}\,\boldsymbol{\nu}\right)\Big|_{\tau=t_{i,k}/t_f}. \tag{6.37}$$

   The parameters α and β serve for the problem conditioning if the value range of the two updates differs significantly. The value of the parameter ε controls the decrease of the auxiliary condition and must be selected once such that the error of the auxiliary condition decreases over iterations. The number κ denotes the step size of the updates and has to be chosen appropriately, i. e., sufficiently small. A strategy for finding a suitable update step size κ can be achieved for example by using a heuristic approach or by introducing a merit function, see [10, Sec. 9.5.2] and [44, Sec. 15.5] for more information.
7. Repeat steps 2. through 6. until the errors of the auxiliary condition and the according gradients are sufficiently small. If the process fails to converge, the step size κ must be reduced.

Note that steps 3 and 4 are independent since there is no dependence between the differential equations, and therefore can be executed in parallel to save computation time.

6.4 Examples

In order to show the application of the adjoint gradient technique for a set of final states specified, various examples are presented here. In a first example, we consider a Moon-landing maneuver as in the Apollo program, discussing several flight scenarios of the Lunar Excursion Module to and from the Moon's surface in minimum time. In a second example, we determine the time-optimal control of a robot model for a rest-to-rest maneuver. Finally, a cart double pendulum system is investigated to perform an upswing maneuver. Note, in the first example, the control appears nonlinearly in the Hamiltonian, while in the other examples, the control is linear in the Hamiltonian, and, hence, here we are dealing with bang-bang controls.

6.4.1 Trajectory Planning for a Moon Landing

In order to apply the proposed theory, we consider a Moon-landing, as e. g., presented by Miller [46]. In detail, we discuss the ascent, descent and abort maneuvers of the Apollo Lunar Excursion Module to and from the Moon's surface in minimum time. The goal is to find the control of the thrust nozzle of the space vehicle to minimize the final time. Note, in the Apollo program, trajectory planning was not primarily based on time optimality, instead, the flight plan was developed regarding safety of the astronauts and mission success. A detailed description of the Apollo flight plan is published by the National Aeronautics and Space Administration in [5].

Nevertheless, trajectory planning is an important issue in aerospace systems [42], and especially time optimization is becoming increasingly important. Hence, the following numerical examples consider the time-optimal Moon-landing of the Lunar Excursion Module in the gravity field of the Moon, see Fig. 4.1. The challenge now is to find the control $u(t)$ such that the cost functional

$$J = \int_{t_0}^{t_f} 1 \, \mathrm{d}t, \tag{6.38}$$

is minimized. If we assume the time-optimal control in an orbital plane, we may again consider only planar motions. The first order state equations of the space vehicle, analogous to [46], are given by Eq. (4.43) and the associated state vector reads

$$\boldsymbol{x}(t) = \big(r, v_r, v_t, \varphi, m\big)^\mathsf{T}. \tag{6.39}$$

From Eq. (4.43) we can derive the adjoint-influence equations defining $\mathbf{P} = (P_{ij}) \in \mathbb{R}^{5\times q}$. As part of the following Apollo missions, we prescribe initial states and claim the space vehicle to satisfy a set of conditions at the final time. Depending on the number of prescribed final conditions q, we obtain the adjoint system

$$\begin{aligned}
\dot{P}_{1j}(t) &= \frac{P_{2j}(t)v_t(t)}{r^2(t)} + P_{3j}(t)\left(\frac{v_t^2(t)}{r^2(t)} - \frac{2\mu}{r^3(t)}\right) - \frac{P_{4j}(t)v_r(t)v_t(t)}{r^2(t)} \\
\dot{P}_{2j}(t) &= 0 \\
\dot{P}_{3j}(t) &= -P_{1j}(t) + \frac{P_{4j}(t)v_t(t)}{r(t)} \\
\dot{P}_{4j}(t) &= -\frac{P_{2j}(t)}{r(t)} - \frac{2P_{3j}(t)v_t(t)}{r(t)} + \frac{P_{4j}(t)v_r(t)}{r(t)} \\
\dot{P}_{5j}(t) &= \frac{P_{3j}(t)F}{m^2(t)}\sin(u(t)) + \frac{P_{4j}(t)F}{m^2(t)}\cos(u(t)),
\end{aligned} \tag{6.40}$$

where $j = 1, \ldots, q$. If we compare Eq. (6.40) with Eq. (4.43), we recognize that the latter mentioned equations occur column-wise in $\mathbf{P}(t)$. Note that Eq. (4.43) and Eq. (6.40) can be readily transformed by Eq. (6.33) and Eq. (6.35), respectively, if we introduce the τ-domain with $\tau \in [0, 1]$. For the following numerical computations, the same parameter set is used as in Sec. 4.7 for the lunar ascent problem.

Below, we now discuss the ascent, descent and abort maneuvers of the Lunar Excursion Module with the proposed theory. Therefore, the state equations from Eq. (4.43) and the adjoint-influence equations from Eq. (6.40) are the basis for all three missions. Depending on the mission, steps 5 and 6 of the algorithm in Sec. 6.3 lead to lengthy expressions and are therefore not written down in detail below.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Mission a.: The Lunar Ascent Problem

As a first mission, we consider the example from Sec. 4.7.1, where a spacecraft is supposed to reach a stable state in the lunar parking orbit in minimum time. Recall, the space vehicle with the mass of the ascent stage m_0 is launched at the surface of the Moon with the polar coordinates $r = R$ and $\varphi = 0$. Thus, the space vehicle starts from rest, the initial velocities v_r and v_t are zero at t_0. In summary we observe five initial conditions given by Eq. (4.50), in which m_0 is again the mass of the ascent stage of the space vehicle. Again, we are looking for the optimal input angle $u(t)$ of the thrust nozzle to reach a stable state in the orbit in minimum time. The final values of the Lunar Excursion Module are given by Eq. (4.51) and are collected in the final constraint vector in Eq. (6.1) yielding

$$\boldsymbol{\phi}(r, v_r, v_t) = \begin{pmatrix} r(t) - (R+h) \\ v_r(t) \\ v_t(t) - \sqrt{\frac{\mu}{R+h}} \end{pmatrix} . \tag{6.41}$$

We guess a final time $t_f = 440\,\mathrm{s}$ and use the initial control $u(t) = 0$ for $t \in [0, 440]$ which is far away from the optimal control. In order to start the algorithm, the optimization parameters $\varepsilon = 0.3$, $\alpha = 0.5$ and $\kappa = 1$ are chosen. The convergence of the final time and the final conditions error for the lunar ascent problem are shown in Fig. 6.4. After 50 iterations, we already observe good results despite a poor initial guess of the control variable. The optimal final time is given by $462.6575\,\mathrm{s}$. In Fig. 6.5 the optimized control is compared to the solution of the boundary value problem from Sec. 4.7.1, showing a good agreement. It has to be emphasized that u_{opt} denotes the solution obtained by optimization and u_{bvp} is the reference solution from the `bvp4c` routine.

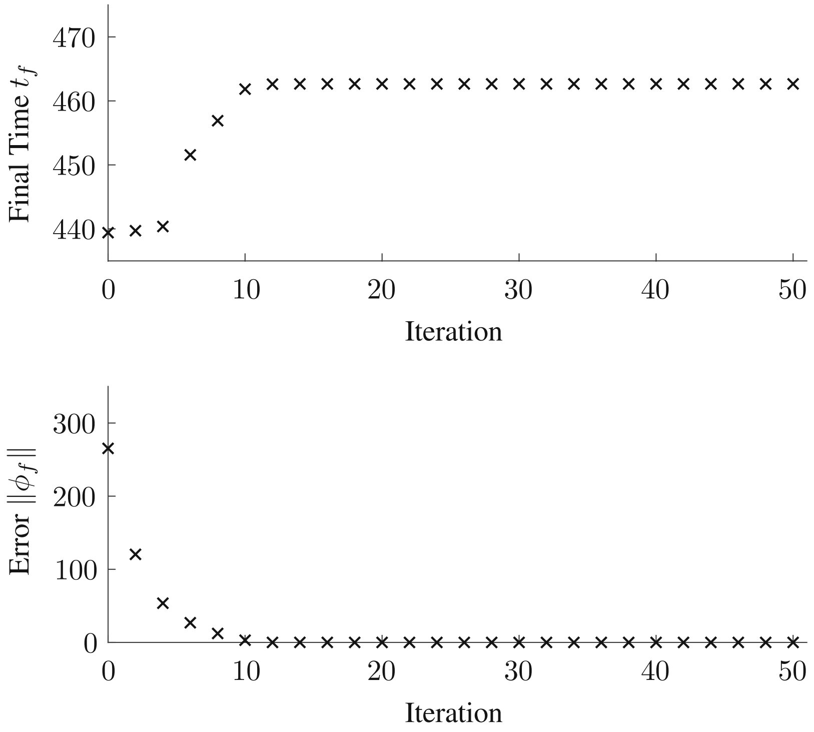

Figure 6.4: Convergence of the time-optimal ascent maneuver

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

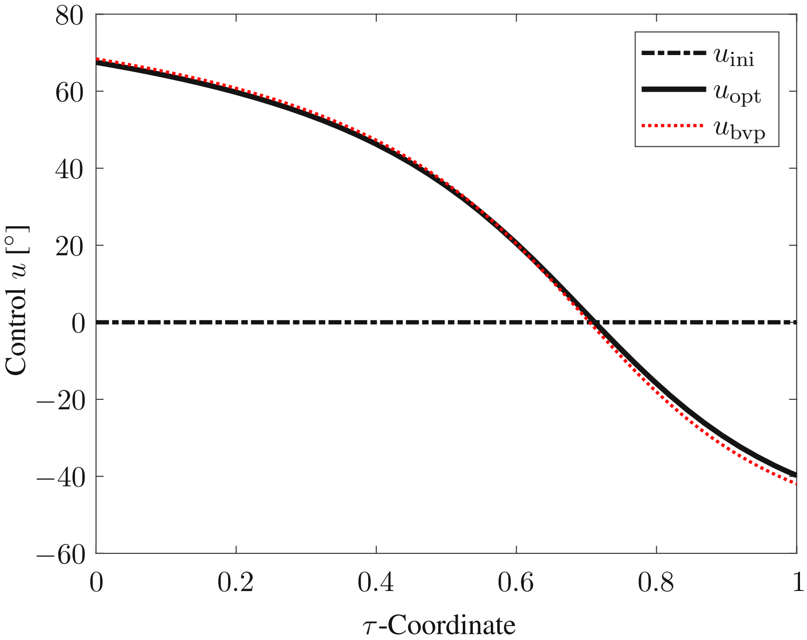

Figure 6.5: Comparison of the optimization solution with the boundary value solution

Mission b.: The Lunar Descent Problem

As a second mission we consider a flight from the lunar parking orbit to the surface of the Moon. The descent maneuver is initiated at the position defined by $r = R + h$ and $\varphi = 0$, where R is again the radius of the Moon and h the altitude of the orbit. We prescribe the initial radial velocity $v_r(0) = 0$ and claim that the centrifugal force is balanced by the gravitational force at $t_0 = 0$, so we receive also a condition for the tangential velocity $v_t(0) = \sqrt{\mu/(R+h)}$. During the descent maneuver the Lunar Excursion Module consists of the ascent and the descent stages comprising the total mass $M_0 = 15239\,\text{kg}$. Finally, the initial conditions read

$$r(0) = R + h \quad \varphi(0) = 0 \quad v_r(0) = 0 \quad v_t(0) = \sqrt{\frac{\mu}{R+h}} \quad m(0) = M_0, \tag{6.42}$$

and the final conditions at $t = t_f$ are given by

$$r(t_f) = R \quad v_r(t_f) = 0 \quad v_t(t_f) = 0. \tag{6.43}$$

Hence, the final constraints in Eq. (6.1) can be written as

$$\boldsymbol{\phi}(r, v_r, v_t) = \begin{pmatrix} r(t) - R \\ v_r(t) \\ v_t(t) \end{pmatrix}. \tag{6.44}$$

As initial guess for the final time we use $t_f = 480\,\text{s}$ and for the control history we simply choose $u(t) = \pi$ for $t \in [0, 440]$. Since we have already found a suitable parameter setup for the descent maneuver, it is an easy task to find a setup for further missions. For the present optimization

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

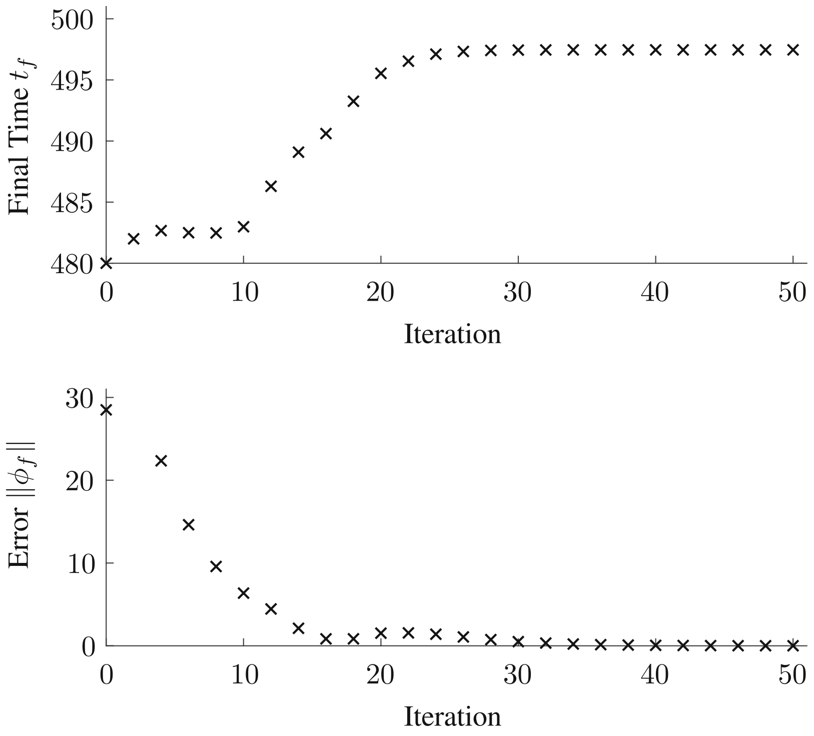

Figure 6.6: Convergence of the time-optimal descent maneuver

problem, we can use the same values as before: $\varepsilon = 0.3$, $\alpha = 0.5$ and $\kappa = 1$.

The convergence of the optimization procedure is depicted in Fig. 6.6. After 50 iterations we stop the optimization procedure as the values of the endpoint error, the control gradient and the end time gradient are sufficiently small. The optimal final time is $497.4586\,\mathrm{s}$ and the control signal

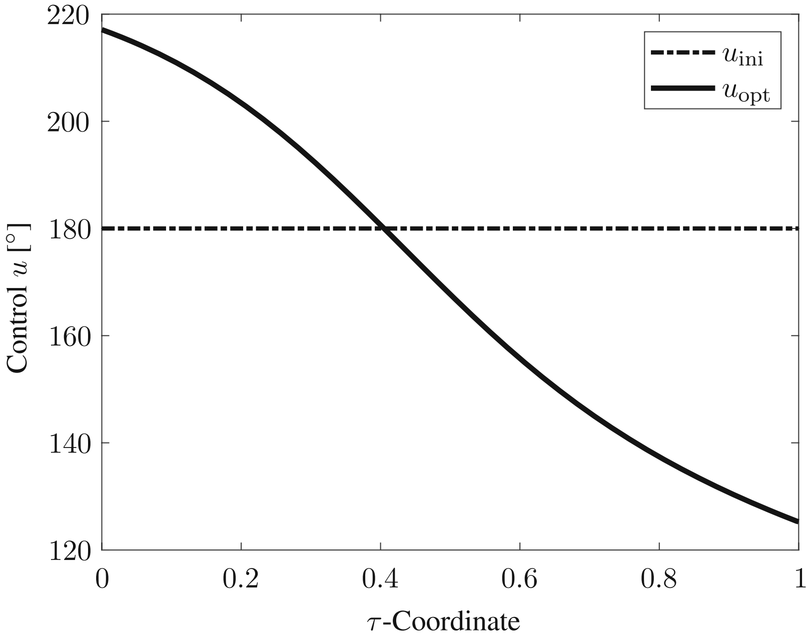

Figure 6.7: Control of the time-optimal descent maneuver

of the time-optimal descent flight is shown in Fig. 6.7. Figure 6.8 shows the flight path and the arrows indicate the force vector of the thrust nozzle acting on the space vehicle. The results shown in Fig. 6.7 and in Fig. 6.8 are in perfect agreement with the solution presented in [46], in which the results are obtained by solving the underlying boundary value problem. Note that we retain the results of this problem in the computer memory, as we will need one part of the solution for the next mission to compute the optimal trajectory for a *pull-up* maneuver.

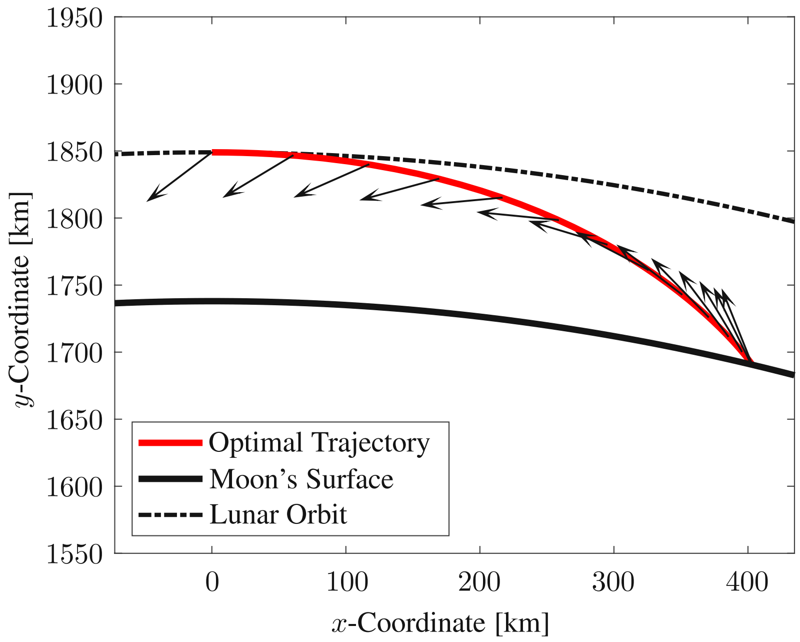

Figure 6.8: Trajectory of the time-optimal descent maneuver

Mission c.: The Lunar Abort Maneuver

In the case of technical problems during the landing phase, the Apollo program has provided various emergency maneuvers. An overview of the Apollo mission and the different abort scenarios is given in [46]. If an abort of the landing procedure is initiated in the early phase (altitude greater than $80\,\mathrm{km}$) of the descent flight, then the space vehicle is supposed to re-rendezvous with the Command and Service Module in the lunar orbit. Hence, the problem of the abort maneuver can be divided into two steps. In a first step, we apply the time-optimal descent maneuver from the previous problem until $t^* = 150\,\mathrm{s}$ is reached, which marks the point in time when the descent flight is aborted. Then, in a second step, we initiate the pull-up maneuver with $t_0 = t^*$ to re-rendezvous and to dock with the Command and Service Module again. The initial conditions are given by the final values of the descent maneuver and the initial mass is prescribed by the mass of the ascent stage m_0, since the descent stage is jettisoned.
The initial values are then given by

$$r(t_0) = r(t^*) \quad \varphi(t_0) = \varphi(t^*) \quad v_r(t_0) = v_r(t^*) \quad v_t(t_0) = v_t(t^*) \quad m(t_0) = m_0. \tag{6.45}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

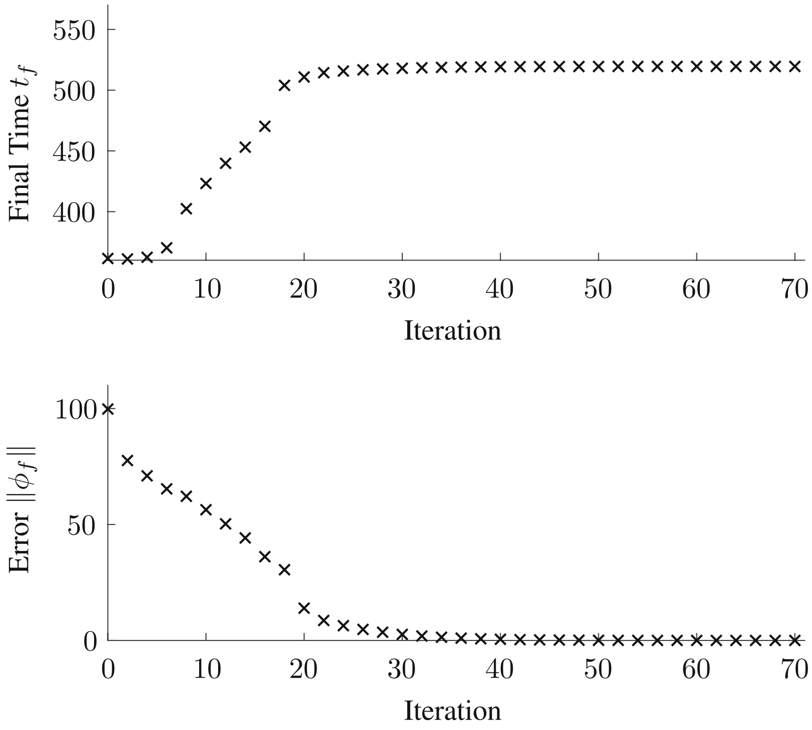

Figure 6.9: Convergence of the time-optimal abort maneuver

The numerical values for the initial states are defined by: $r(t^*) = 1828.9\,\mathrm{km}$, $\varphi(t^*) = 0.1167\,\mathrm{rad}$, $v_r(t^*) = -0.2599\,\mathrm{km/s}$, $v_t(t^*) = 1.2172\,\mathrm{km/s}$ and $m(t^*) = m_0 = 4774\,\mathrm{kg}$, which are taken from the previous example. Again, we demand the final radial velocity to be zero and the final tangential velocity to be equal to the orbital velocity. In order to achieve a rendezvous with the Command and Service Module, we define the final position of the Lunar Excursion Module by

$$r(t_f) = R + h \quad \varphi(t_f) = \Omega\, t_f \quad v_r(t_f) = 0 \quad v_t(t_f) = \sqrt{\frac{\mu}{R+h}}. \tag{6.46}$$

Herein, Ω denotes the angular velocity of the Command and Service Module and is given by $\Omega = \sqrt{\mu/\left(R+h\right)^3}$ and t_f is also the time that has elapsed since the descent maneuver was initiated. Again, the constraints are collected in the vector in Eq. (6.1) yielding

$$\boldsymbol{\phi}(r, \varphi, v_r, v_t) = \begin{pmatrix} r(t) - (R+h) \\ \varphi(t) - \Omega\, t \\ v_r(t) \\ v_t(t) - \sqrt{\frac{\mu}{R+h}} \end{pmatrix}. \tag{6.47}$$

For the initial control we simply choose $u(t) = 0$ for $t \in [150, 350]$. It has to be mentioned again that finding a suitable optimization setting is a simple task. For the proposed mission, the optimization setup is given by: $\varepsilon = 0.1$, $\alpha = 0.5$ and $\kappa = 1$. After 70 control updates the optimality measures are sufficiently small and we already achieve reasonable results.
In Fig. 6.9 the convergence history of the optimization procedure for the pull-up maneuver is

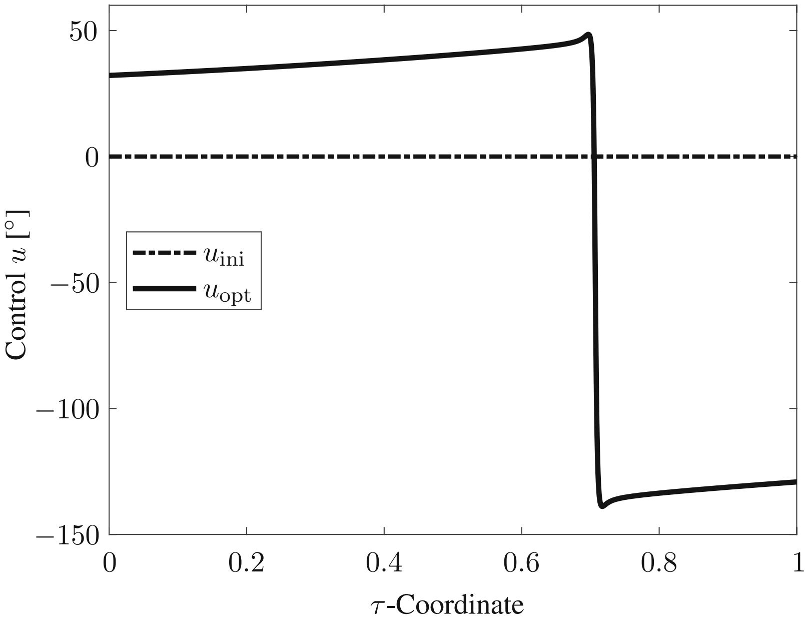

Figure 6.10: Control of the time-optimal abort maneuver

shown. We end up with an optimal final time of $519.4992\,\text{s}$ including the initial time $t^* = 150$. Finally, the identified control angle is plotted in Fig. 6.10, where the unit interval $[0, 1]$ corresponds to the time span $[t^*, t_f]$ of the pull-up scenario. The resulting trajectory is presented in Fig. 6.11, including both, the time-optimal descent flight and the pull-up maneuver initiated at time $t = t^* = 150\,\text{s}$, which perfectly matches the results presented in [46]. Note that we have joined the trajectory of the time-optimal descent flight (until $t = t^*$) with the solution of the time-optimal pull-up maneuver.

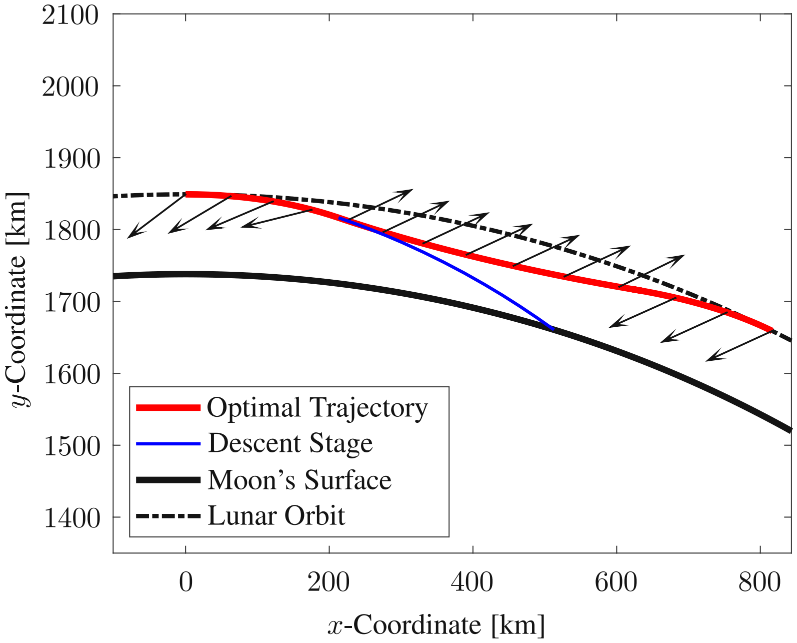

Figure 6.11: Trajectory of the time-optimal abort maneuver

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

As before, the arrows show the force acting on the space vehicle. The trajectory of the jettisoned descent stage is depicted by the thin blue line and terminates at the surface of the Moon.

6.4.2 Planar Robot Arm

In this example we want to show that the proposed method is also able to deal efficiently with bang-bang controls, see [22]. Figure 6.12 depicts a model of a SCARA robot consisting of two revolute joints and two robot arms with the lengths l_1 and l_2. The system has two degrees of freedom, where the angles $\varphi_1(t)$ and $\varphi_2(t)$ form a set of minimal independent generalized coordinates. The two robot arms have the masses m_1 and m_2 and the moments of inertia J_1 and J_2 around their centers of gravity.
The aim is to control the moments $u_1(t)$ and $u_2(t)$ in the joints so that the tool center point (TCP) reaches a defined position in minimum time. Hence, the cost functional to be minimized reads

$$J = \int_{t_0}^{t_f} 1 \mathrm{d}t = t_f - t_0. \tag{6.48}$$

The equations of motion are given by

$$\begin{aligned}
&\left(J_1 + m_1 s_1^2 + m_2 l_1^2 + m_3 l_1^2\right)\dot{\omega}_1(t) + l_1\left(m_2 s_2 + m_3 l_2\right)\cos(\varphi_1(t) - \varphi_2(t))\dot{\omega}_2(t) \\
&\qquad = u_1(t) - u_2(t) - l_1\left(s_2 m_2 + l_2 m_3\right)\sin(\varphi_1(t) - \varphi_2(t))\omega_2^2(t) \\
&l_1\left(m_2 s_2 + m_3 l_2\right)\cos(\varphi_1(t) - \varphi_2(t))\dot{\omega}_1(t) + \left(J_2 + m_2 s_2^2 + m_3 l_2^2\right)\dot{\omega}_2(t) \\
&\qquad = u_2(t) + l_1\left(s_2 m_2 + l_2 m_3\right)\sin(\varphi_1(t) - \varphi_2(t))\omega_1^2(t),
\end{aligned} \tag{6.49}$$

where $\omega_1(t) = \dot{\varphi}_1(t)$, $\omega_2(t) = \dot{\varphi}_2(t)$. The state equations can be readily written in the form of Eq. (6.33) by introducing the vector of state variables

$$\mathbf{x}(t) = (\varphi_1, \varphi_2, \omega_1, \omega_2)^{\mathsf{T}}. \tag{6.50}$$

and by applying a transformation to the τ-space. Moreover, the initial conditions of the robot read:

$$\varphi_1(0) = -\pi/4 \quad \varphi_2(0) = 0 \quad \omega_1(0) = 0 \quad \omega_2(0) = 0. \tag{6.51}$$

We claim the robot to take a rest position with zero velocities

$$\omega_1(t_f) = 0 \quad \omega_2(t_f) = 0, \tag{6.52}$$

and the coordinates

$$x_f = l_1\cos(\varphi_1(t_f)) + l_2\cos(\varphi_2(t_f)) \quad y_f = l_1\sin(\varphi_1(t_f)) + l_2\sin(\varphi_2(t_f)), \tag{6.53}$$

in minimum time t_f.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

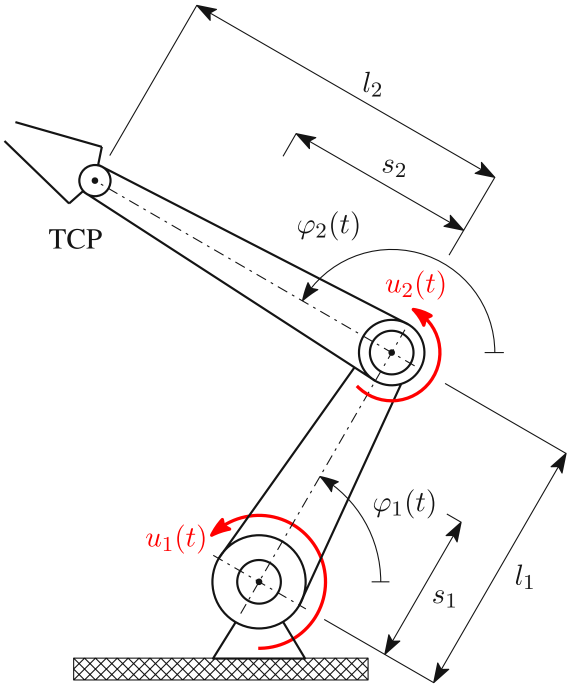

Figure 6.12: Planar two-axis robot

Hence, the function $\boldsymbol{\phi}(\boldsymbol{x}, t)$ introduced in Eq. (6.1) reads

$$\boldsymbol{\phi}(\varphi_1, \varphi_2, \omega_1, \omega_2) := \begin{pmatrix} l_1\cos(\varphi_1(t)) + l_2\cos(\varphi_2(t)) - x_f \\ l_1\sin(\varphi_1(t)) + l_2\sin(\varphi_2(t)) - y_f \\ \omega_1(t) \\ \omega_2(t) \end{pmatrix}. \tag{6.54}$$

Note that φ_1 and φ_2 at $t = t_f$ could also be computed by solving the first two equations $\phi_1 = 0$ and $\phi_2 = 0$. In this case the function $\boldsymbol{\phi}$ can be described in the simpler form $\boldsymbol{\phi}(\varphi_1, \varphi_2, \omega_1, \omega_2) := (\varphi_1 - \varphi_{1,f}, \varphi_2 - \varphi_{2,f}, \omega_1, \omega_2)^{\mathsf{T}}$, however, we use the original form in Eq. (6.54) instead. Since we do not need a penalty function Π for state constraints in our example, we get the solution $\boldsymbol{p}(\tau) = \boldsymbol{0}$ in Eq. (6.34). It should be mentioned that the adjoint-influence differential equations in Eq. (6.35) contain rather lengthy expressions and are therefore not reproduced here. To deal with the unavoidable limitations of the drive torques $u_1(t)$ and $u_2(t)$, we compare two strategies below:

1. A penalty approach; where any violation of the limitations produces additional costs in the cost functional. In this case, no restriction of the control histories need to be introduced for the optimization process.
2. Since the controls $u_1(t)$ and $u_2(t)$ appear linearly in the equations of motion and also in the Hamiltonian in Eq. (6.27), the time-optimal control should be a bang-bang control. In this case, only the switching points have to be determined.

In the following, both approaches are discussed in detail.

(a) The Penalty Approach

The first solution strategy is to put bounds on $u_1(t)$ and $u_2(t)$ by using a penalty approach. Hence, we extend the cost functional in Eq. (6.48) in the following form

$$J = \int_{t_0}^{t_f} \left[1 + \Pi(u_1(t), u_2(t))\right] \mathrm{d}t, \tag{6.55}$$

in which $\Pi(u_1(t), u_2(t)) := \mu_1 \Pi_1(u_1(t)) + \mu_2 \Pi_2(u_2(t))$ is used to force the controls to satisfy the inequalities $-u_1^* \leq u_1(t) \leq u_1^*$ and $-u_2^* \leq u_2(t) \leq u_2^*$. A simple formulation of the penalty function reads

$$\Pi_i(u_i(t)) := \begin{cases} 0 & \text{for} \quad |u_i(t)| < u_i^* \\ \dfrac{1}{2}\left(|u_i(t)| - u_i^*\right)^2 & \text{otherwise.} \end{cases} \tag{6.56}$$

The updates of the controls and the final time are then given by Eq. (6.36). For the numerical computation we use the parameter set $l_1 = l_2 = 1$, $m_1 = m_3 = 1$, $m_2 = 0.5$, $J_1 = 0.0833$, $J_2 = 0.0417$, $u_1^* = 4$, $u_2^* = 2$, the weights $\mu_1 = \mu_2 = 10$ for the penalty functions, $\alpha = 0.5$ for the problem scaling, $\varepsilon = 0.1$ to approach the final condition and $\kappa = 0.2$ for the update step size. The optimization is initiated with the control signals $u_1(t) = u_2(t) = 0$ for the entire interval $[t_0, t_f]$, where $t_0 = 0$ and $t_f = 3\,\mathrm{s}$. The resulting drive signals are depicted in Fig. 6.14 where the normalized time coordinate $\tau = t/t_f$ is plotted on the abscissa. One can observe, that u_1 and u_2 converge to bang-bang controls where the jumps appear exactly at the times where the switching functions $h_1(\tau)$ and $h_2(\tau)$, introduced in Eq. (6.28), cross the zero axis. Nevertheless, we notice

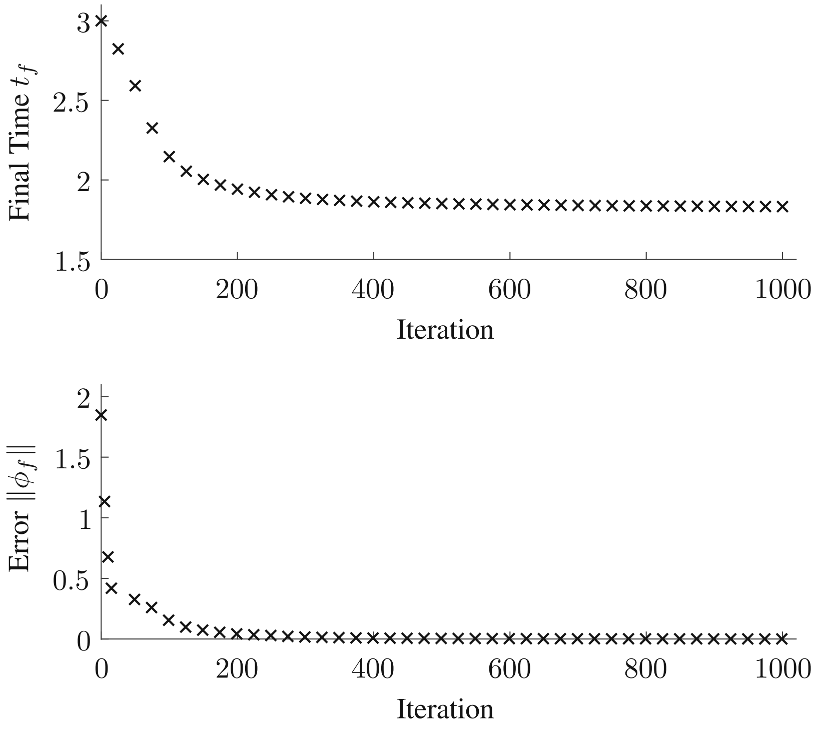

Figure 6.13: Convergence of conventional control optimization using a penalty approach

TU Bibliothek Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar. The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
WIEN Your knowledge hub

a rather slow convergence of this approach as expected for bang-bang controls. We stopped the optimization procedure after 950 iterations. The convergence of the final time and the constraint error is shown in Fig. 6.13, while the minimal final time is given by $1.8334\,\mathrm{s}$ and the constraint error is sufficient small.

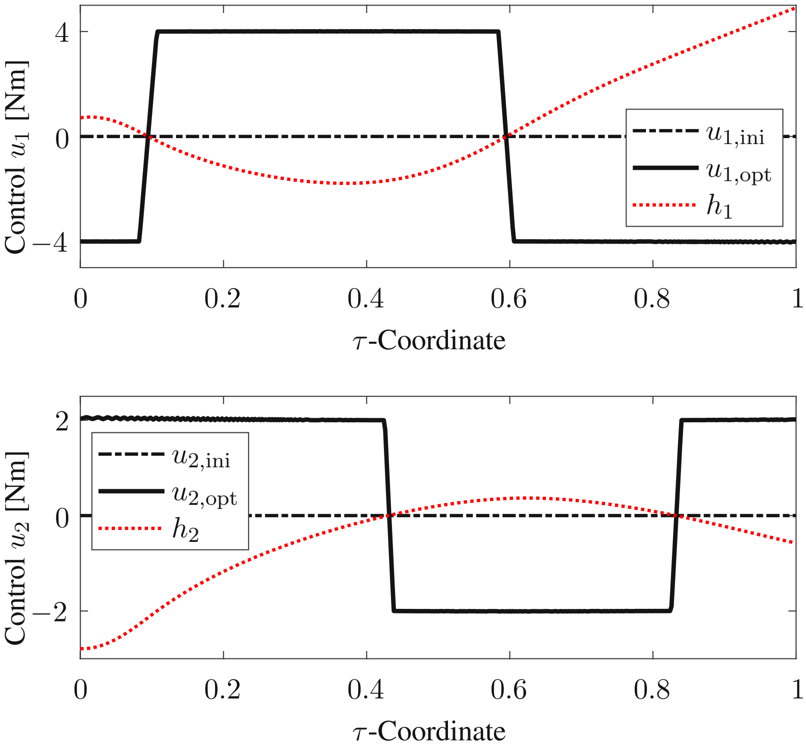

Figure 6.14: Control histories for the rest-to-rest maneuver

(b) The Bang-Bang Approach

An alternative, and more efficient solution strategy is provided by optimizing directly the switching points rather than using a penalty approach. Therefore, we introduce four switching points $t_{1,1}$, $t_{1,2}$ and $t_{2,1}$, $t_{2,2}$ instead of computing $u_1(t)$ and $u_2(t)$ for the whole interval $t \in [t_0, t_f]$:

$$u_1(t) := \begin{cases} -u_1^* & \text{for} \quad t < t_{1,1} \\ +u_1^* & \text{for} \quad t_{1,1} \le t \le t_{1,2} \\ -u_1^* & \text{for} \quad t_{1,2} < t \end{cases}, \quad u_2(t) := \begin{cases} +u_2^* & \text{for} \quad t < t_{2,1} \\ -u_2^* & \text{for} \quad t_{2,1} \le t \le t_{2,2} \\ +u_2^* & \text{for} \quad t_{2,2} < t. \end{cases} \tag{6.57}$$

Close attention shall be paid to maintain a certain distance between the initial points and to limit the step size of the update, otherwise there are no special requirements for the selection of the initial switch points. As initial estimates we impose an end time $t_f = 2\,\mathrm{s}$ and select the switching points: $t_{1,1} = 0.4$, $t_{1,2} = 1.4$, $t_{2,1} = 0.2$ and $t_{2,2} = 0.8$ (in terms of the physical time coordinate). If we set the condition numbers to $\alpha = 0.1$ and $\beta = 0.1$ and use the parameter $\varepsilon = 0.05$ and the constant step sizes $\kappa = 1$, the final time is updated by Eq. $(6.36)_1$ and the update for the switching

TU Bibliothek. Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar. The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek. WIEN Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

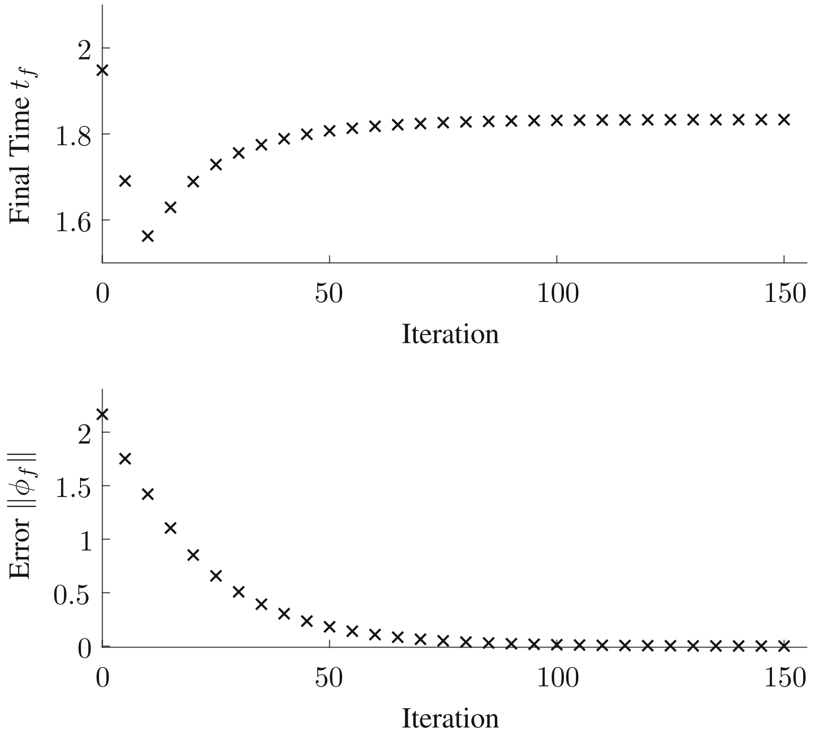

Figure 6.15: Convergence of the switching point optimization

points is obtained by Eq. (6.37). Here, 150 iterations are already sufficient to determine the time-optimal control. The resulting convergence and the solution are depicted in Fig. 6.15, Fig. 6.16 and

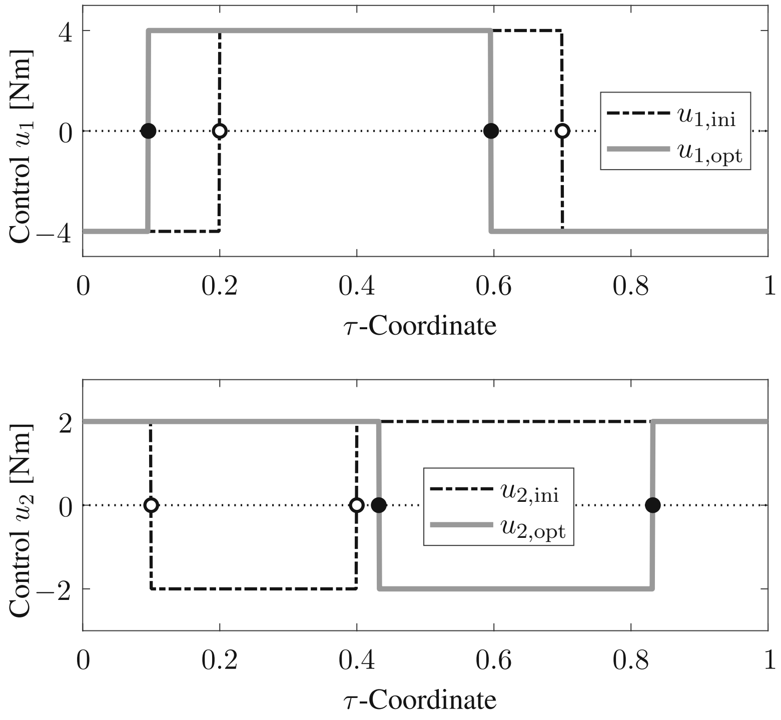

Figure 6.16: Bang-bang controls for the rest-to-rest maneuver

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Fig. 6.17. Note that the convergence of the bang-bang approach depicted in Fig. 6.15 is much faster than for the penalty approach. The resulting time-optimal controls are depicted in Fig. 6.16 and the trajectory is shown in Fig. 6.17. The optimal switching times are obtained as $t_{1,1} = 0.1758$, $t_{1,2} = 1.0918$, $t_{2,1} = 0.7915$, and $t_{2,2} = 1.5241$. In Fig. 6.16 the determined switching times fit well with the control history in Fig. 6.14 and corresponds perfectly with the zero crossings of the switching function $h(\tau)$. The time-optimal rest-to-rest maneuver is performed in $1.8320\,\mathrm{s}$ which coincides well with the optimal final time ($1.8334\,\mathrm{s}$) of the penalty approach.

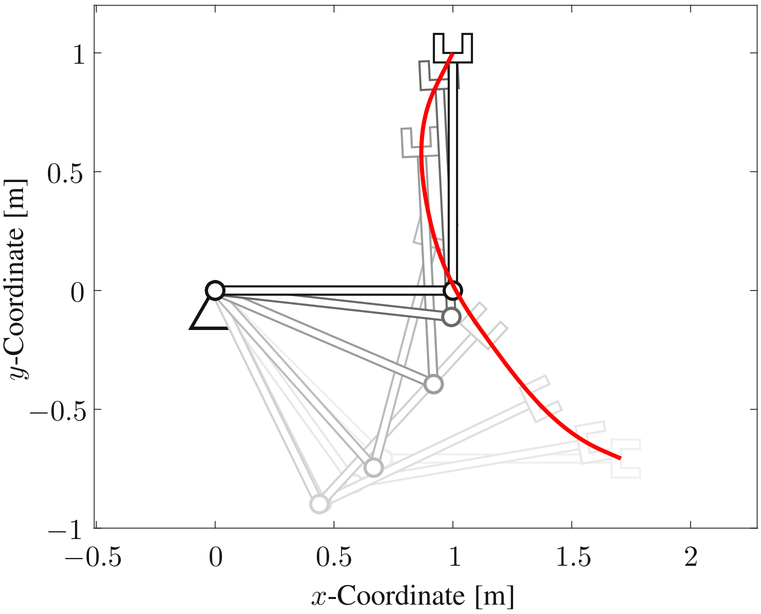

Figure 6.17: Time-optimal path for a rest-to-rest maneuver

6.4.3 Inverse Double Pendulum

As third example we consider the inverse double pendulum depicted in Fig. 4.4 and studied in Sec. 4.7.2 to show once again the application of the algorithm to a more challenging problem. The solution via the adjoint method does not require much effort, since many equations of the boundary value task are already implemented and much of them can be adopted. In summary, we can use the cost functional in Eq (4.53), the penalty function in Eq. (4.54), the state equations in Eq. (4.55) and the associated adjoint equations, which are not displayed here, but are already implemented. Moreover, the boundary conditions in Eq. (4.57) and in Eq. (4.58) can also be copied and therefore the function $\boldsymbol{\phi}(\boldsymbol{x}, t)$ defining the final constraints according to Eq. (6.1), is readily given by

$$\boldsymbol{\phi}(\omega_1, \omega_2, v, \varphi_1, \varphi_2) := \begin{pmatrix} \omega_1 \\ \omega_2 \\ v \\ \underset{2\pi}{\text{modulo}}\, |\varphi_1| - \pi \\ \underset{2\pi}{\text{modulo}}\, |\varphi_2| - \pi \end{pmatrix} . \tag{6.58}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

Again, we use the direct optimization of the switching points presented in Section 6.2. For that purpose, we introduce eleven switching points $t_1 < t_2 < \cdots < t_{11}$ and the control function

$$u(t) := \begin{cases} +u^* & \text{for} \quad t < t_1 \\ -u^* & \text{for} \quad t_1 \le t < t_2 \\ +u^* & \text{for} \quad t_2 \le t < t_3 \\ & \vdots \\ -u^* & \text{for} \quad t_{11} \le t, \end{cases} \tag{6.59}$$

in which $u^* = 15$. Initially, the final time is estimated as $t_f = 3$ and for the switching points we choose $t_1 = 0.15$, $t_2 = 0.30$, $t_3 = 0.45$, $t_4 = 1.20$, $t_5 = 1.80$, $t_6 = 1.95$, $t_7 = 2.25$, $t_8 = 2.40$, $t_9 = 2.55$, $t_{10} = 2.70$ and $t_{11} = 2.85$. With $\alpha = 0.2$ and $\beta = 0.005$, the update of the final time is determined by Eq. $(6.36)_1$ and the update of the switching points is received by Eq. (6.37) using the update parameter $\varepsilon = 0.05$ and the constant step size $\kappa = 1$. The optimization procedure is stopped after 700 iterations and the convergence is depicted in Fig 6.18. The optimized switching points were obtained as $t_1 = 0.2758$, $t_2 = 0.3669$, $t_3 = 0.3909$, $t_4 = 1.0520$, $t_5 = 1.7594$, $t_6 = 1.8632$, $t_7 = 2.2141$, $t_8 = 2.4527$, $t_9 = 2.4753$, $t_{10} = 2.6320$ and $t_{11} = 2.7684$. Figure 6.19 shows the identified force acting on the cart in comparison with the initial control and a solution which was obtained from solving the underlying boundary value problem. The solution obtained with our method is in perfect agreement with the solution of the boundary value solver.

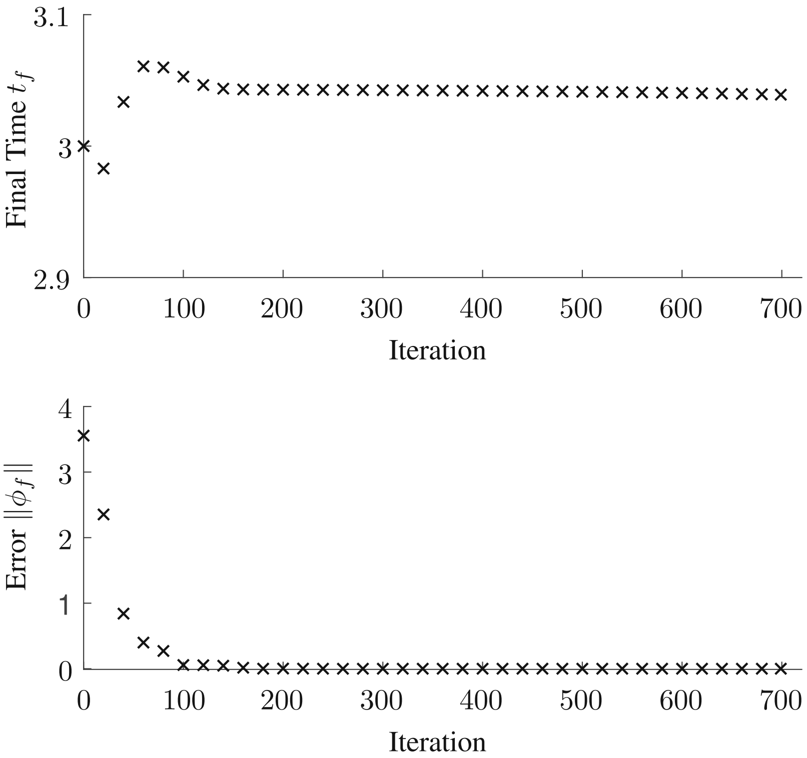

Figure 6.18: Convergence of the switching point optimization for the upswing

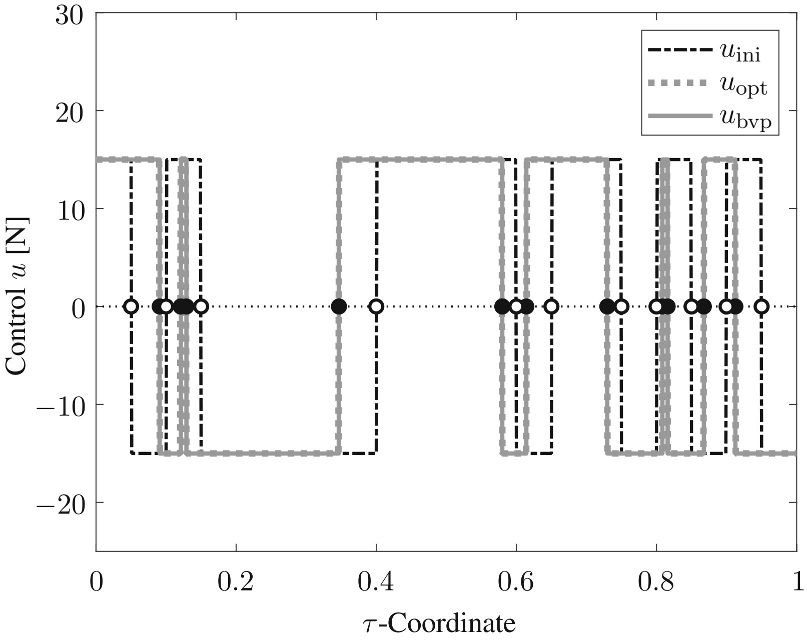

Figure 6.19: Bang-bang controls for the upswing

The solution of the optimization procedure end up with a final time of $3.0339\,\mathrm{s}$, while the final time of the boundary value solution is given by $3.0332\,\mathrm{s}$.

However, no secret is to be made of this; the troubles encountered in Chapter 4 should draw the reader's attention to the novel approach in this chapter. If we think back to the lengthy solution process of the two-point boundary value problem in Sec. 4.7.2, it becomes clear once again that the proposed theory offers a good alternative for solving time-optimal control problems.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Chapter 7

The Adjoint Method in Multibody Dynamics

The adjoint gradient method has already been utilized to find time-optimal solutions for academic examples and for a variety of aerospace and robotics problems, and holds great promise for solving even more complicated problems. In this chapter, the adjoint method, introduced in Chapter 6, is extended to solve time-optimal control problems in multibody dynamics.

If independent coordinates are used, the equations of motion, in many cases, can be too complex to formulate. Hence, in multibody dynamics the equations of motion are described by a set of differential-algebraic equations (DAE) in terms of redundant generalized coordinates. Due to algebraic constraints, the equations of motion are extended by constraint forces using Lagrange multipliers, see [63], leading to a differential-algebraic boundary value problem in optimal control theory. Once the solution of the equations of motion, the adjoint equations and the adjoint-influence equations is obtained, the gradient of the cost functional can be computed efficiently, which improves the final time and approaches the final conditions. Finally, the optimal solution is found by applying an iterative optimization scheme.

Within this chapter, special care is given to find consistent boundary conditions for the adjoint and the adjoint-influence equations. Moreover, an architecture for time integration solvers for the adjoint equations as well as the adjoint-influence equations is developed herein.

7.1 Problem Formulation

The key idea of the adjoint gradient method for solving optimal control problems, regarding final constraints, has already been shown in Sec. 6.1 for nonlinear systems described by a minimal set of independent coordinates. In particular focusing on multibody systems, an extension of the theory is proposed here.

Therefore, the dynamics of constrained multibody systems is described by a set of differential-algebraic equations of the form

$$\begin{aligned} \mathbf{M}(\boldsymbol{q}(t))\ddot{\boldsymbol{q}}(t) + \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}(\boldsymbol{q}(t))\boldsymbol{\lambda}(t) &= \boldsymbol{f}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \boldsymbol{u}(t)) \\ \boldsymbol{C}(\boldsymbol{q}(t)) &= \mathbf{0}, \end{aligned} \tag{7.1}$$

where $\boldsymbol{q}(t) \in \mathbb{R}^n$ denotes the vector of generalized coordinates and $\boldsymbol{u}(t) \in \mathbb{R}^m$ the vector of control inputs. Moreover, $\mathbf{M}(\boldsymbol{q}(t))$ is the mass matrix of the system and $\boldsymbol{f}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \boldsymbol{u}(t))$ is called the vector of applied and gyroscopic forces. The constraint equations are collected in $\boldsymbol{C}(\boldsymbol{q}(t))$, the matrix $\boldsymbol{C}_{\boldsymbol{q}}(\boldsymbol{q}(t))$ denotes the associated constraint Jacobian and $\boldsymbol{\lambda}(t) \in \mathbb{R}^s$ is a vector of Lagrange multipliers.

The system equations in Eq. (7.1) are called descriptor system including second order differential equations and algebraic constraints for the generalized coordinates $\boldsymbol{q}(t)$ and the Lagrange multipliers $\boldsymbol{\lambda}(t)$. In order to solve the descriptor system one can rewrite the system to pure second order differential equations by forming the second time derivative of the algebraic constraint in Eq. $(7.1)_2$. However, the resulting equations may cause the drift phenomenon when using a conventional integration scheme. To circumvent the drift problem, numerous methods have already been developed, such as the stabilization approach suggested by Baumgarte [3] in the year 1972. Meanwhile, there are modern integrators that can directly address the differential-algebraic equations [31]. Nevertheless, before we solve the descriptor system in Eq. (7.1), in many cases it is numerically advantageous to reduce the index of the system from three to two. Therefore, the *Gear-Gupta-Leimkuhler* (GGL) formulation in [25] can be used, which is a widely recognized approach in the literature. If we introduce the velocity vector $\boldsymbol{v}(t) := \dot{\boldsymbol{q}}(t)$ we can rewrite the equations of motion by adding the constraint equations at velocity level and, hence, a further set of Lagrange multipliers $\boldsymbol{\xi}(t)$ yielding

$$\begin{aligned} \dot{\boldsymbol{q}}(t) &= \boldsymbol{v}(t) - \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}(\boldsymbol{q}(t))\boldsymbol{\xi}(t) \\ \mathbf{M}(\boldsymbol{q}(t))\dot{\boldsymbol{v}}(t) + \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}(\boldsymbol{q}(t))\boldsymbol{\lambda}(t) &= \boldsymbol{f}(\boldsymbol{q}(t), \boldsymbol{v}(t), \boldsymbol{u}(t)) \\ \boldsymbol{C}(\boldsymbol{q}(t)) &= \mathbf{0} \\ \boldsymbol{C}_{\boldsymbol{q}}(\boldsymbol{q}(t))\boldsymbol{v}(t) &= \mathbf{0}. \end{aligned} \tag{7.2}$$

For given initial conditions $\boldsymbol{q}(t_0) = \boldsymbol{q}_0$ and $\boldsymbol{v}(t_0) = \boldsymbol{v}_0$, the equations of motion in Eq. (7.2) can be solved numerically for $\boldsymbol{q}(t)$, $\boldsymbol{v}(t)$ and $\boldsymbol{\lambda}(t)$ by applying a DAE-solver for index two equations. Now we are looking for control inputs $\boldsymbol{u}(t)$, which minimize the cost functional

$$J = \int_{t_0}^{t_f} \big[1 + \Pi(\boldsymbol{q}(t), \boldsymbol{v}(t), \boldsymbol{u}(t))\big]\,\mathrm{d}t. \tag{7.3}$$

Here, $\Pi(\boldsymbol{q}(t), \boldsymbol{v}(t), \boldsymbol{u}(t))$ denotes again a penalty function in order to introduce constraints on $\boldsymbol{q}(t)$, $\boldsymbol{v}(t)$ and $\boldsymbol{u}(t)$. The final time t_f is considered free, and we claim the system to satisfy a set of final conditions of the form

$$\boldsymbol{\phi}(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f) = \mathbf{0}, \quad \boldsymbol{\phi} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^r. \tag{7.4}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

However, some of the final boundary conditions may be regarded free. From Pontryagin's minimum principle, the optimal control can be obtained by solving a two-point boundary value problem, which can be derived from Eq. (7.2), Eq. (7.3) and Eq. (7.4) in case of multibody systems. As we already know, solving such two-point boundary value problems is already a challenge for ordinary differential equations.

7.2 Optimal Control of Multibody Systems

The key idea of the adjoint gradient method for solving optimal control problems, regarding final constraints, has been shown already in Chapter 6 for nonlinear systems. In particular focusing on multibody systems, an extension of the theory is proposed here: First, we derive the gradient of the cost functional in Eq. (7.3) with respect to the controls. Second, we proceed for the final constraints in Eq. (7.4) analogously. Finally, we combine both gradients by a linear combination to achieve an improvement with respect to the function J while approaching the final constraints $\boldsymbol{\phi}(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f)$ within one update.
In order to compute the gradient of J with respect to the controls $\boldsymbol{u}(t)$, we augment the cost functional by the equations of motion in Eq. (7.2), yielding

$$\begin{aligned}\bar{J} = \int_{t_0}^{t_f} \Big[1 + \Pi + \boldsymbol{w}^\mathsf{T} \left(\boldsymbol{v} - \dot{\boldsymbol{q}} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T} \boldsymbol{\xi} \right) & \\ + \boldsymbol{p}^\mathsf{T} \left(\boldsymbol{f} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T} \boldsymbol{\lambda} - \mathbf{M} \dot{\boldsymbol{v}} \right) + \boldsymbol{\mu}^\mathsf{T} \boldsymbol{C} + \boldsymbol{\sigma}^\mathsf{T} \boldsymbol{C}_{\boldsymbol{q}} \boldsymbol{v} \Big] \mathrm{d}t, & \end{aligned} \tag{7.5}$$

in which $\boldsymbol{w}(t)$, $\boldsymbol{p}(t) \in \mathbb{R}^n$ and $\boldsymbol{\mu}(t)$, $\boldsymbol{\sigma}(t) \in \mathbb{R}^s$ denote adjoint variables. Although the extended cost functional in Eq. (7.5) looks impressive, its variation is not difficult, only cumbersome. The first variation of $\bar{J}$ results from a control variation $\delta \boldsymbol{u}(t)$ and can be written as

$$\begin{aligned}\delta \bar{J} = \int_{t_0}^{t_f} \Big[& \Pi_{\boldsymbol{q}} \delta \boldsymbol{q} + \Pi_{\boldsymbol{v}} \delta \boldsymbol{v} + \Pi_{\boldsymbol{u}} \delta \boldsymbol{u} + \boldsymbol{w}^\mathsf{T} \left(\delta \boldsymbol{v} - \delta \dot{\boldsymbol{q}} - \left(\boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T} \boldsymbol{\xi} \right)_{\boldsymbol{q}} \delta \boldsymbol{q} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T} \delta \boldsymbol{\xi} \right) \\ & + \boldsymbol{p}^\mathsf{T} \left(\boldsymbol{f}_{\boldsymbol{q}} \delta \boldsymbol{q} + \boldsymbol{f}_{\boldsymbol{v}} \delta \boldsymbol{v} + \boldsymbol{f}_{\boldsymbol{u}} \delta \boldsymbol{u} - \left(\boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T} \boldsymbol{\lambda} \right)_{\boldsymbol{q}} \delta \boldsymbol{q} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T} \delta \boldsymbol{\lambda} - (\mathbf{M} \dot{\boldsymbol{v}})_{\boldsymbol{q}} \delta \boldsymbol{q} - \mathbf{M} \delta \dot{\boldsymbol{v}} \right) \\ & + \boldsymbol{\mu}^\mathsf{T} \boldsymbol{C}_{\boldsymbol{q}} \delta \boldsymbol{q} + \boldsymbol{\sigma}^\mathsf{T} \left((\boldsymbol{C}_{\boldsymbol{q}} \boldsymbol{v})_{\boldsymbol{q}} \delta \boldsymbol{q} + \boldsymbol{C}_{\boldsymbol{q}} \delta \boldsymbol{v} \right) \Big] \mathrm{d}t + (1 + \Pi_f) \delta t_f, \end{aligned} \tag{7.6}$$

where we used the abbreviation for the penalty term $\Pi_f := \Pi(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), \boldsymbol{u}(t_f))$. Now we perform an integration by parts of terms including $\delta \dot{\boldsymbol{q}}$ and $\delta \dot{\boldsymbol{v}}$ yielding

$$\begin{aligned} \int_{t_0}^{t_f} \boldsymbol{w}^\mathsf{T} \delta \dot{\boldsymbol{q}} \, \mathrm{d}t &= - \int_{t_0}^{t_f} \dot{\boldsymbol{w}}^\mathsf{T} \delta \boldsymbol{q} \, \mathrm{d}t + \boldsymbol{w}^\mathsf{T}(t_f) \delta \boldsymbol{q}(t_f) \\ \int_{t_0}^{t_f} \boldsymbol{p}^\mathsf{T} \mathbf{M} \delta \dot{\boldsymbol{v}} \, \mathrm{d}t &= - \int_{t_0}^{t_f} \frac{\mathrm{d}}{\mathrm{d}t} \left(\boldsymbol{p}^\mathsf{T} \mathbf{M} \right) \delta \boldsymbol{v} \, \mathrm{d}t + \boldsymbol{p}^\mathsf{T}(t_f) \mathbf{M}(\boldsymbol{q}(t_f)) \delta \boldsymbol{v}(t_f), \end{aligned} \tag{7.7}$$

where the initial values $\boldsymbol{q}(t_0)$ and $\boldsymbol{v}(t_0)$ are fixed, and hence, the variation $\delta \boldsymbol{v}(t_0)$ and $\delta \boldsymbol{q}(t_0)$ are zero.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

After collecting all variations $\delta\boldsymbol{q}$, $\delta\boldsymbol{v}$, $\delta\boldsymbol{\lambda}$ and $\delta\boldsymbol{u}$ we obtain:

$$\begin{aligned}\delta\bar{J} = \int_{t_0}^{t_f} \Big\{ \Big[&\Pi_{\boldsymbol{q}} + \dot{\boldsymbol{w}}^\mathsf{T} - \boldsymbol{w}^\mathsf{T}\left(\boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\xi}\right)_{\boldsymbol{q}} + \boldsymbol{p}^\mathsf{T}\left(\boldsymbol{f}_{\boldsymbol{q}} - \left(\boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\lambda}\right)_{\boldsymbol{q}} - (\mathbf{M}\dot{\boldsymbol{v}})_{\boldsymbol{q}}\right) + \boldsymbol{\mu}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}} \\ &+ \boldsymbol{\sigma}^\mathsf{T}\left(\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v}\right)_{\boldsymbol{q}}\Big]\delta\boldsymbol{q} + \Big[\Pi_{\boldsymbol{v}} + \boldsymbol{w}^\mathsf{T} + \boldsymbol{p}^\mathsf{T}\boldsymbol{f}_{\boldsymbol{v}} + \frac{\mathrm{d}}{\mathrm{d}t}\left(\boldsymbol{p}^\mathsf{T}\mathbf{M}\right) + \boldsymbol{\sigma}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}\Big]\delta\boldsymbol{v} \\ &- \boldsymbol{p}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\delta\boldsymbol{\lambda} - \boldsymbol{w}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\delta\boldsymbol{\xi} + \Big[\Pi_{\boldsymbol{u}} + \boldsymbol{p}^\mathsf{T}\boldsymbol{f}_{\boldsymbol{u}}\Big]\delta\boldsymbol{u}\Big\}\mathrm{d}t - \boldsymbol{w}^\mathsf{T}(t_f)\delta\boldsymbol{q}(t_f) \\ &- \boldsymbol{p}^\mathsf{T}(t_f)\mathbf{M}(\boldsymbol{q}(t_f))\delta\boldsymbol{v}(t_f) + \left(1 + \Pi_f\right)\delta t_f.\end{aligned} \tag{7.8}$$

In order to circumvent the computation of the variations $\delta\boldsymbol{q}(t)$, $\delta\boldsymbol{v}(t)$, $\delta\boldsymbol{\lambda}(t)$ and $\delta\boldsymbol{\xi}(t)$, we choose $\boldsymbol{w}(t)$, $\boldsymbol{p}(t)$, $\boldsymbol{\mu}(t)$ and $\boldsymbol{\sigma}(t)$ as follows

$$\begin{aligned}\dot{\boldsymbol{w}} &= -\Pi_{\boldsymbol{q}}^\mathsf{T} + \left(\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{\xi}\right)_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{w} - \mathbf{G}\boldsymbol{p} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\mu} - \left(\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v}\right)_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\sigma} \\ \frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{M}\boldsymbol{p}) &= -\Pi_{\boldsymbol{v}}^\mathsf{T} - \boldsymbol{w} - \boldsymbol{f}_{\boldsymbol{v}}^\mathsf{T}\boldsymbol{p} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\sigma} \\ \boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{p} &= \mathbf{0} \\ \boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{w} &= \mathbf{0},\end{aligned} \tag{7.9}$$

in which we introduced the abbreviation $\mathbf{G} = \boldsymbol{f}_{\boldsymbol{q}}^\mathsf{T} - \left(\boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\lambda}\right)_{\boldsymbol{q}}^\mathsf{T} - (\mathbf{M}\dot{\boldsymbol{v}})_{\boldsymbol{q}}^\mathsf{T}$. Moreover, if we choose the trivial final values $\boldsymbol{w}(t_f) = \mathbf{0}$ and $\boldsymbol{p}(t_f) = \mathbf{0}$, the boundary conditions are compatible with the constraint equations in Eq. $(7.9)_3$ and in Eq. $(7.9)_4$. Due to the choice of the Lagrange multipliers in Eq. (7.9) the variation of the cost functional in Eq. (7.8) simplifies to

$$\delta\bar{J} = \int_{t_0}^{t_f} (\Pi_{\boldsymbol{u}} + \boldsymbol{p}^\mathsf{T}\boldsymbol{f}_{\boldsymbol{u}})\delta\boldsymbol{u}\,\mathrm{d}t + \left(1 + \Pi_f\right)\delta t_f, \tag{7.10}$$

which directly relates the changes $\delta\boldsymbol{u}(t)$ and δt_f with $\delta\bar{J}$. Since the equations of motion shall satisfy the final conditions $\boldsymbol{\phi}(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f)$ in Eq. (7.4), we examine how a change in the control affects the final conditions. Hence, we enhance the final conditions in Eq. (7.4) also by the equations of motion in Eq. (7.2) yielding

$$\begin{aligned}\bar{\boldsymbol{\phi}} = \int_{t_0}^{t_f} \Big[&\mathbf{W}^\mathsf{T}(\boldsymbol{v} - \dot{\boldsymbol{q}} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\xi}) + \mathbf{P}^\mathsf{T}\left(\boldsymbol{f} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\lambda} - \mathbf{M}\dot{\boldsymbol{v}}\right) \\ &+ \boldsymbol{\mathcal{M}}^\mathsf{T}\boldsymbol{C} + \boldsymbol{\mathcal{S}}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v}\Big]\mathrm{d}t + \boldsymbol{\phi}(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f),\end{aligned} \tag{7.11}$$

where we introduced some further sets of adjoint variables arranged in the matrices $\mathbf{W}(t), \mathbf{P}(t) \in \mathbb{R}^{n\times r}$ and $\boldsymbol{\mathcal{M}}(t)$, $\boldsymbol{\mathcal{S}}(t) \in \mathbb{R}^{s\times r}$. The variation of the extended end condition vector in Eq. (7.11) due to an infinitesimal change in the controls read

$$\begin{aligned}\delta\bar{\boldsymbol{\phi}} = \int_{t_0}^{t_f} \Big\{&\mathbf{W}^\mathsf{T}\left(\delta\boldsymbol{v} - \delta\dot{\boldsymbol{q}} - \left(\boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\xi}\right)_{\boldsymbol{q}}\delta\boldsymbol{q} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\delta\boldsymbol{\xi}\right) + \mathbf{P}^\mathsf{T}\Big(\boldsymbol{f}_{\boldsymbol{q}}\delta\boldsymbol{q} + \boldsymbol{f}_{\boldsymbol{v}}\delta\boldsymbol{v} + \boldsymbol{f}_{\boldsymbol{u}}\delta\boldsymbol{u} \\ &- \left(\boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\lambda}\right)_{\boldsymbol{q}}\delta\boldsymbol{q} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\delta\boldsymbol{\lambda} - (\mathbf{M}\dot{\boldsymbol{v}})_{\boldsymbol{q}}\delta\boldsymbol{q} - \mathbf{M}\delta\dot{\boldsymbol{v}}\Big) + \boldsymbol{\mathcal{M}}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}\delta\boldsymbol{q} \\ &+ \boldsymbol{\mathcal{S}}^\mathsf{T}\left(\left(\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v}\right)_{\boldsymbol{q}}\delta\boldsymbol{q} + \boldsymbol{C}_{\boldsymbol{q}}\delta\boldsymbol{v}\right)\Big\}\,\mathrm{d}t\ + \boldsymbol{\phi}_{\boldsymbol{q}}(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f)\,\delta\boldsymbol{q}_f \\ &+ \boldsymbol{\phi}_{\boldsymbol{v}}(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f)\,\delta\boldsymbol{v}_f + \boldsymbol{\phi}_t(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f)\,\delta t_f,\end{aligned} \tag{7.12}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

where $\boldsymbol{\phi}_{\boldsymbol{q}}$, $\boldsymbol{\phi}_{\boldsymbol{v}}$ and $\boldsymbol{\phi}_t$ are partial derivatives of the final conditions $\boldsymbol{\phi}(\boldsymbol{q}, \boldsymbol{v}, t)$ with respect to $\boldsymbol{q}$, $\boldsymbol{v}$ and t. Again we may carry out an integration by parts of the terms containing $\delta \boldsymbol{q}$ and $\delta \boldsymbol{v}$, yielding

$$
\begin{aligned}
\delta \bar{\phi} = \int_{t_0}^{t_f} \Big\{ \Big[& \dot{\mathbf{W}}^{\mathsf{T}} - \mathbf{W}^{\mathsf{T}} \left(\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \boldsymbol{\xi} \right)_{\boldsymbol{q}} + \mathbf{P}^{\mathsf{T}} \left(\boldsymbol{f}_{\boldsymbol{q}} - \left(\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \boldsymbol{\lambda} \right)_{\boldsymbol{q}} - (\mathbf{M} \dot{\boldsymbol{v}})_{\boldsymbol{q}} \right) + \boldsymbol{\mathcal{M}}^{\mathsf{T}} \boldsymbol{C}_{\boldsymbol{q}} \\
& + \boldsymbol{\mathcal{S}}^{\mathsf{T}} \left(\boldsymbol{C}_{\boldsymbol{q}} \boldsymbol{v} \right)_{\boldsymbol{q}} \Big] \delta \boldsymbol{q} + \Big[\mathbf{W}^{\mathsf{T}} + \mathbf{P}^{\mathsf{T}} \boldsymbol{f}_{\boldsymbol{v}} + \frac{\mathrm{d}}{\mathrm{d}t} \left(\mathbf{P}^{\mathsf{T}} \mathbf{M} \right) + \boldsymbol{\mathcal{S}}^{\mathsf{T}} \boldsymbol{C}_{\boldsymbol{q}} \Big] \delta \boldsymbol{v} - \mathbf{P}^{\mathsf{T}} \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \delta \boldsymbol{\lambda} \\
& - \mathbf{W}^{\mathsf{T}} \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \delta \boldsymbol{\xi} + \mathbf{P}^{\mathsf{T}} \boldsymbol{f}_{\boldsymbol{u}} \delta \boldsymbol{u} \Big\} \mathrm{d}t - \mathbf{W}^{\mathsf{T}}(t_f) \delta \boldsymbol{q}(t_f) - \mathbf{P}^{\mathsf{T}}(t_f) \mathbf{M}(\boldsymbol{q}(t_f)) \delta \boldsymbol{v}(t_f) \\
& + \boldsymbol{\phi}_{\boldsymbol{q}}(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f) \delta \boldsymbol{q}_f + \boldsymbol{\phi}_{\boldsymbol{v}}(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f) \delta \boldsymbol{v}_f + \boldsymbol{\phi}_t(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f) \delta t_f .
\end{aligned}
\tag{7.13}
$$

Recall that $\delta \boldsymbol{q}_f \neq \delta \boldsymbol{q}(t_f)$, thus we have to make appropriate substitutions in order to combine the boundary terms. With reference to Eq. (4.12), the relation is given by $\delta \boldsymbol{q}_f = \dot{\boldsymbol{q}}(t_f) \delta t_f + \delta \boldsymbol{q}(t_f)$ and analogously $\delta \boldsymbol{v}_f = \dot{\boldsymbol{v}}(t_f) \delta t_f + \delta \boldsymbol{v}(t_f)$, hence

$$
\begin{aligned}
\delta \bar{\phi} = \int_{t_0}^{t_f} \Big\{ \Big[& \dot{\mathbf{W}}^{\mathsf{T}} - \mathbf{W}^{\mathsf{T}} \left(\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \boldsymbol{\xi} \right)_{\boldsymbol{q}} + \mathbf{P}^{\mathsf{T}} \left(\boldsymbol{f}_{\boldsymbol{q}} - \left(\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \boldsymbol{\lambda} \right)_{\boldsymbol{q}} - (\mathbf{M} \dot{\boldsymbol{v}})_{\boldsymbol{q}} \right) + \boldsymbol{\mathcal{M}}^{\mathsf{T}} \boldsymbol{C}_{\boldsymbol{q}} \\
& + \boldsymbol{\mathcal{S}}^{\mathsf{T}} \left(\boldsymbol{C}_{\boldsymbol{q}} \boldsymbol{v} \right)_{\boldsymbol{q}} \Big] \delta \boldsymbol{q} + \Big[\mathbf{W}^{\mathsf{T}} + \mathbf{P}^{\mathsf{T}} \boldsymbol{f}_{\boldsymbol{v}} + \frac{\mathrm{d}}{\mathrm{d}t} \left(\mathbf{P}^{\mathsf{T}} \mathbf{M} \right) + \boldsymbol{\mathcal{S}}^{\mathsf{T}} \boldsymbol{C}_{\boldsymbol{q}} \Big] \delta \boldsymbol{v} - \mathbf{P}^{\mathsf{T}} \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \delta \boldsymbol{\lambda} \\
& - \mathbf{W}^{\mathsf{T}} \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \delta \boldsymbol{\xi} + \mathbf{P}^{\mathsf{T}} \boldsymbol{f}_{\boldsymbol{u}} \delta \boldsymbol{u} \Big\} \mathrm{d}t + \left(\boldsymbol{\phi}_{\boldsymbol{q}} - \mathbf{W}^{\mathsf{T}} \right) \Big|_{t_f} \delta \boldsymbol{q}(t_f) + \left(\boldsymbol{\phi}_{\boldsymbol{v}} - \mathbf{P}^{\mathsf{T}} \mathbf{M} \right) \Big|_{t_f} \delta \boldsymbol{v}(t_f) \\
& + \left(\boldsymbol{\phi}_{\boldsymbol{q}} \dot{\boldsymbol{q}} + \boldsymbol{\phi}_{\boldsymbol{v}} \dot{\boldsymbol{v}} + \boldsymbol{\phi}_t \right) \Big|_{t_f} \delta t_f .
\end{aligned}
\tag{7.14}
$$

Since we have not imposed any restrictions on the adjoint matrices yet, we now define $\mathbf{W}(t)$, $\mathbf{P}(t)$, $\boldsymbol{\mathcal{M}}(t)$ and $\boldsymbol{\mathcal{S}}(t)$ by a set of matrix differential-algebraic equations in the following way

$$
\begin{aligned}
\dot{\mathbf{W}} &= \left(\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \boldsymbol{\xi} \right)_{\boldsymbol{q}}^{\mathsf{T}} \mathbf{W} - \mathbf{G} \mathbf{P} - \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \boldsymbol{\mathcal{M}} - \left(\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \boldsymbol{v} \right)_{\boldsymbol{q}}^{\mathsf{T}} \boldsymbol{\mathcal{S}} \\
\frac{\mathrm{d}}{\mathrm{d}t} (\mathbf{M} \mathbf{P}) &= -\mathbf{W} - \boldsymbol{f}_{\boldsymbol{v}}^{\mathsf{T}} \mathbf{P} - \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \boldsymbol{\mathcal{S}} \\
\boldsymbol{C}_{\boldsymbol{q}} \mathbf{P} &= \mathbf{0} \\
\boldsymbol{C}_{\boldsymbol{q}} \mathbf{W} &= \mathbf{0},
\end{aligned}
\tag{7.15}
$$

with the final values

$$
\left(\boldsymbol{\phi}_{\boldsymbol{q}}^{\mathsf{T}} - \mathbf{W} \right) \Big|_{t_f} = \mathbf{0} \quad \text{and} \quad \left(\boldsymbol{\phi}_{\boldsymbol{v}}^{\mathsf{T}} - \mathbf{M} \mathbf{P} \right) \Big|_{t_f} = \mathbf{0}. \tag{7.16}
$$

At a first glance, we can solve Eq. (7.16) for $\mathbf{W}(t_f)$ and $\mathbf{P}(t_f)$ to get appropriate final values. Unfortunately, a closer look at Eq. (7.16) shows that a suitable choice of the boundary conditions is not quite simple, because the choice must also fulfill the boundary conditions of the constraints in Eq. (7.15)$_3$ and Eq. (7.15)$_4$. Later we will devote attention to finding suitable boundary conditions. Nevertheless, if we assume that we have already found suitable final conditions in Eq. (7.16) to

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

solve the system in Eq. (7.15), the variation of the end conditions in Eq. (7.14) reduces to

$$\delta\bar{\boldsymbol{\phi}} = \int_{t_0}^{t_f} \mathbf{P}^{\mathsf{T}} \boldsymbol{f}_{\boldsymbol{u}} \delta\boldsymbol{u}\,\mathrm{d}t + \dot{\boldsymbol{\phi}}_f \delta t_f, \tag{7.17}$$

in which we used the abbreviation $\dot{\boldsymbol{\phi}}_f := \big(\boldsymbol{\phi}_{\boldsymbol{q}}\dot{\boldsymbol{q}} + \boldsymbol{\phi}_{\boldsymbol{v}}\dot{\boldsymbol{v}} + \boldsymbol{\phi}_t\big)\big|_{t_f}$ as the total time derivative of $\boldsymbol{\phi}$ at $t = t_f$. Now Eq. (7.17) shows the direct impact of $\delta\boldsymbol{u}(t)$ and δt_f on $\delta\bar{\boldsymbol{\phi}}$. Combining the variation of Eq. (7.10) and the variation of the final conditions in Eq. (7.17), a descent direction can be identified as

$$\begin{aligned} \delta t_f &= -\kappa\alpha\Big(1 + \Pi_f + \boldsymbol{\nu}^{\mathsf{T}}\dot{\boldsymbol{\phi}}_f\Big) \\ \delta\boldsymbol{u}(t) &= -\kappa\Big(\Pi_{\boldsymbol{u}}^{\mathsf{T}} + \boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}}\boldsymbol{p} + \boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}}\mathbf{P}\,\boldsymbol{\nu}\Big). \end{aligned} \tag{7.18}$$

Herein, analogously to Eq. (6.12), $\boldsymbol{\nu} \in \mathbb{R}^r$ is a vector of multipliers for the clever combination of the gradients of the cost functional and the final conditions. Through this combination we want to achieve an improvement of the end time on the one hand and approach the end conditions on the other hand. Hence, if we claim the variations $\delta\boldsymbol{u}(t)$ and δt_f resulting in a better approximation of the conditions in Eq. (7.4) we demand

$$\delta\bar{\boldsymbol{\phi}} := -\varepsilon\,\boldsymbol{\phi}(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f), \tag{7.19}$$

where $\varepsilon > 0$ determines the approach to the final conditions. In order to compute the multiplier $\boldsymbol{\nu}$, we equate Eq. (7.17) with Eq. (7.19) and insert Eq. (7.18) for $\delta\boldsymbol{u}(t)$ and δt_f. Introducing the abbreviations

$$\begin{aligned} \mathbf{A} &:= \int_{t_0}^{t_f} \mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{\boldsymbol{u}}\boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}}\mathbf{P}\,\mathrm{d}t + \alpha\dot{\boldsymbol{\phi}}_f\dot{\boldsymbol{\phi}}_f^{\mathsf{T}} && \in \mathbb{R}^{r\times r} \\ \boldsymbol{b} &:= \int_{t_0}^{t_f} \mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{\boldsymbol{u}}\left(\Pi_{\boldsymbol{u}}^{\mathsf{T}} + \boldsymbol{f}_{\boldsymbol{u}}^{\mathsf{T}}\boldsymbol{p}\right)\mathrm{d}t + \alpha\left(1 + \Pi_f\right)\dot{\boldsymbol{\phi}}_f && \in \mathbb{R}^r \quad , \end{aligned} \tag{7.20}$$

the multiplier $\boldsymbol{\nu}$ for the linear combination of the gradients in Eq. (7.18) is obtained as

$$\boldsymbol{\nu} = \frac{\varepsilon}{\kappa}\mathbf{A}^{-1}\boldsymbol{\phi}(\boldsymbol{q}(t_f), \boldsymbol{v}(t_f), t_f) - \mathbf{A}^{-1}\boldsymbol{b}. \tag{7.21}$$

Finally, Eq. (7.18) can be used to update the control inputs and the final time to accomplish an improvement of the final time and to approach the final conditions.

7.2.1 Consistent Boundary Conditions

Now we will pay closer attention to the boundary conditions in Eq. (7.16) of the adjoint-influence differential-algebraic equations. Note, one has to take care that the choice $\mathbf{W}(t_f) := \mathbf{0}$ and $\mathbf{P}(t_f) := \mathbf{0}$ satisfies the constraint equations in Eq. $(7.15)_3$ and Eq. $(7.15)_4$ at $t = t_f$, however, this is not in agreement with Eq. (7.16). In order to overcome this issue, we follow the idea proposed by Gear, Gupta and Leimkuhler in [25]. Hence, we consider the constraints in Eq. $(7.2)_3$

and in Eq. (7.2)$_4$ at the final time $t = t_f$. If we carry out a variation of both equations with respect to $\boldsymbol{q}$ and $\boldsymbol{v}$ and multiply them with two arbitrary matrices $\mathbf{L} \in \mathbb{R}^{n\times s}$ and $\mathbf{U} \in \mathbb{R}^{n\times s}$, we obtain

$$\left(\mathbf{L}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}\delta\boldsymbol{q}\right)\Big|_{t_f} = \mathbf{0}, \quad \left(\mathbf{U}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}\delta\boldsymbol{v}\right)\Big|_{t_f} = \mathbf{0}, \tag{7.22}$$

or, respectively,

$$\left(\mathbf{L}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}\delta\boldsymbol{q} + \mathbf{U}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}\delta\boldsymbol{v}\right)\Big|_{t_f} = \mathbf{0}. \tag{7.23}$$

This zero term can now be subtracted from Eq. (7.14), leading to the extended boundary conditions:

$$\begin{aligned}\left(\boldsymbol{\phi}_{\boldsymbol{q}}^\mathsf{T} - \mathbf{W} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\mathbf{L}\right)\Big|_{t_f} &= \mathbf{0}\\ \left(\boldsymbol{\phi}_{\boldsymbol{v}}^\mathsf{T} - \mathbf{M}\mathbf{P} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\mathbf{U}\right)\Big|_{t_f} &= \mathbf{0}.\end{aligned} \tag{7.24}$$

These boundary conditions replace Eq. (7.16), and can now be solved for the adjoint matrices without violating the constraint conditions in Eq. (7.15)$_3$ and Eq. (7.15)$_4$. One may solve Eq. (7.24)$_1$ and Eq. (7.15)$_4$, given by the system

$$\begin{pmatrix}\mathbf{I} & \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\\ \boldsymbol{C}_{\boldsymbol{q}} & \mathbf{0}\end{pmatrix}\Bigg|_{t_f}\begin{pmatrix}\mathbf{W}(t_f)\\ \mathbf{L}\end{pmatrix} = \begin{pmatrix}\boldsymbol{\phi}_{\boldsymbol{q}}^\mathsf{T}\\ \mathbf{0}\end{pmatrix}\Bigg|_{t_f}, \tag{7.25}$$

for $\mathbf{W}(t_f)$ and $\mathbf{L}$ first, and then solve Eq. (7.24)$_2$ and Eq. (7.15)$_3$, reading

$$\begin{pmatrix}\mathbf{M} & \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\\ \boldsymbol{C}_{\boldsymbol{q}} & \mathbf{0}\end{pmatrix}\Bigg|_{t_f}\begin{pmatrix}\mathbf{P}(t_f)\\ \mathbf{U}\end{pmatrix} = \begin{pmatrix}\boldsymbol{\phi}_{\boldsymbol{v}}^\mathsf{T}\\ \mathbf{0}\end{pmatrix}\Bigg|_{t_f}, \tag{7.26}$$

for $\mathbf{P}(t_f)$ and $\mathbf{U}$. Note that $\mathbf{L}$ and $\mathbf{U}$ only serve to compute consistent boundary conditions and are not further needed.

7.2.2 Bang-Bang Principle

Since control forces commonly appear linear in the equations of motion of multibody systems, it is often the case that a control acts as a bang-bang control. We will now turn our attention to the efficient calculation of bang-bang controls.

Recall the discussion in Sec. 6.2, the basic idea of this approach is the introduction of switching times instead of computing the whole control history for bang-bang controls. Therefore, we introduce the Hamiltonian

$$\begin{aligned}\mathcal{H} := 1 + \Pi(\boldsymbol{q},\boldsymbol{v},\boldsymbol{u}) &- \left(\boldsymbol{w} + \mathbf{W}\boldsymbol{\nu}\right)^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\xi}\\ &+ \left(\boldsymbol{p} + \mathbf{P}\boldsymbol{\nu}\right)^\mathsf{T}\left(\boldsymbol{f} - \boldsymbol{C}_{\boldsymbol{q}}^\mathsf{T}\boldsymbol{\lambda} - \mathbf{M}\dot{\boldsymbol{v}}\right) + \boldsymbol{\mu}^\mathsf{T}\boldsymbol{C} + \boldsymbol{\sigma}^\mathsf{T}\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v},\end{aligned} \tag{7.27}$$

which shows an easy access to the optimality criterion for constrained multibody systems. Analogously to Eq. (6.29), when the control $u_i(t)$ occurs linearly in the Hamiltonian, we consider the

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

following cases for an optimal control

$$u_{i,\mathrm{opt}}(t) := \begin{cases} u_{i,\max}, & h_i(t) < 0 \\ u_{i,\min}, & h_i(t) > 0 \\ \text{singular}, & h_i(t) = 0. \end{cases} \tag{7.28}$$

Here, $h_i(t)$ denotes again the switching function and can be obtained by differentiating the Hamiltonian in Eq. (7.27) with respect to the control u_i yielding

$$h_i(t) := \Pi_{u_i} + \boldsymbol{f}_{u_i}^{\mathsf{T}}(\boldsymbol{q}(t), \boldsymbol{v}(t))\Big(\boldsymbol{p}(t) + \mathbf{P}(t)\boldsymbol{\nu}\Big). \tag{7.29}$$

In the non-singular case, i. e., $h_i(t) < 0$ or $h_i(t) > 0$, the variation of the control u_i is given by the variation of the switching times $t_{i,k}$, hence

$$\delta u_i(t) = \begin{cases} \pm\bar{u}_i & \text{for } t \in [t_{i,k};\, t_{i,k} + \delta t_{i,k}] \\ \quad 0 & \text{otherwise}, \end{cases} \tag{7.30}$$

where we used the abbreviation $\bar{u}_i = u_{i,\max} - u_{i,\min}$. Switching from $u_{i,\min}$ to $u_{i,\max}$ results in a negative variation while switching from $u_{i,\max}$ to $u_{i,\min}$ results in a positive variation of $\delta u_i(t)$. If there are pure bang-bang controls among the controls, the variations in Eq. (7.10) and Eq. (7.17) can be rewritten as

$$\begin{aligned} \delta\bar{J} &= \sum_{i=1}^{l}\sum_{k=1}^{N_i} \pm\bar{u}_i \Big(\boldsymbol{p}^{\mathsf{T}}\boldsymbol{f}_{u_i}\Big)\Big|_{t_{i,k}} \delta t_{i,k} + \sum_{i=l+1}^{m}\int_{t_0}^{t_f} \Big(\Pi_{u_i} + \boldsymbol{p}^{\mathsf{T}}\boldsymbol{f}_{u_i}\Big)\,\delta u_i \,\mathrm{d}t + \big(1 + \Pi_f\big)\delta t_f \\ \delta\bar{\boldsymbol{\phi}} &= \sum_{i=1}^{l}\sum_{k=1}^{N_i} \pm\bar{u}_i \Big(\mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{u_i}\Big)\Big|_{t_{i,k}} \delta t_{i,k} + \sum_{i=l+1}^{m}\int_{t_0}^{t_f} \mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{u_i}\delta u_i \,\mathrm{d}t + \dot{\boldsymbol{\phi}}_f\,\delta t_f, \end{aligned} \tag{7.31}$$

in which we assigned $i = 1, \dots, l$ for the bang-bang controls, and $i = l+1, \dots, m$ for the non-bang-bang controls. Note that the penalty function Π depends only on the variables $u_{l+1}, \dots, u_m$, hence, $\Pi_{u_i} = 0$ applies to bang-bang controls. Analogously to Eq. (7.18), we finally identify a descent direction from Eq. (7.31) as

$$\delta t_{i,k} = \mp\kappa\beta\bar{u}_i\Big(\boldsymbol{f}_{u_i}^{\mathsf{T}}\boldsymbol{p} + \boldsymbol{f}_{u_i}^{\mathsf{T}}\mathbf{P}\,\boldsymbol{\nu}\Big)\Big|_{t_{i,k}}. \tag{7.32}$$

Herein, the multiplier $\boldsymbol{\nu}$ can be computed according to Eq. (7.21), where the matrix $\mathbf{A}$ and the vector $\boldsymbol{b}$ can be computed by

$$\begin{aligned} \mathbf{A} :=& \beta\sum_{i=1}^{l}\sum_{k=1}^{N_i} \bar{u}_i^2\,\mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{u_i}\boldsymbol{f}_{u_i}^{\mathsf{T}}\mathbf{P}\Big|_{t_{i,k}} + \sum_{i=l+1}^{m}\int_{t_0}^{t_f} \mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{u_i}\boldsymbol{f}_{u_i}^{\mathsf{T}}\mathbf{P}\,\mathrm{d}t + \alpha\dot{\boldsymbol{\phi}}_f\dot{\boldsymbol{\phi}}_f^{\mathsf{T}} \\ \boldsymbol{b} :=& \beta\sum_{i=1}^{l}\sum_{k=1}^{N_i} \bar{u}_i^2\mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{u_i}\boldsymbol{f}_{u_i}^{\mathsf{T}}\boldsymbol{p}\Big|_{t_{i,k}} + \sum_{i=l+1}^{m}\int_{t_0}^{t_f} \mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{u_i}\Big(\Pi_{u_i}^{\mathsf{T}} + \boldsymbol{f}_{u_i}^{\mathsf{T}}\boldsymbol{p}\Big)\,\mathrm{d}t + \alpha\left(1 + \Pi_f\right)\dot{\boldsymbol{\phi}}_f. \end{aligned} \tag{7.33}$$

In case of bang-bang controls, we use the modified abbreviations in Eq. (7.33) instead of Eq. (7.20).

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

7.2.3 An Introductory Example

In order to illustrate the derivation of the adjoint equations and the gradient method on switching point optimization presented above, we consider a simple example where a car is approximated by a unit point mass that can be accelerated and decelerated along a straight line. A corresponding example is often discussed in literature of optimal control theory (confer [39, Sec. 5.4]), since the optimal solution can be computed analytically. However, we consider a redundant formulation of the problem in Fig. 7.1, where the total mass of the car ($m = 1$) is divided equally between the two axle masses $m_1 = m_2 = 1/2$ which are connected by a rigid rod. The position of the axles are the generalized coordinates given by

$$\boldsymbol{q}(t) = \big(q_1, q_2\big)^{\mathsf{T}}. \tag{7.34}$$

Of course, $q_1(t)$ and $q_2(t)$ are not independent since the rod constrains both mass points to move in the same way. Hence, we observe one constraint to link the particle masses according to the chassis frame. Referring to Fig. 7.1, the length of the rod is 1 and therefore we obtain the following constraint equation

$$C(\boldsymbol{q}(t)) = q_2(t) - q_1(t) - 1 = 0. \tag{7.35}$$

To circumvent numerical troubles in the simulation, e. g. with the HHT integration scheme, Eq. (7.2) is used to provide an index two formulation of the equations of motion, reading

$$\begin{aligned}
\dot{q}_1(t) &= v_1(t) + \xi(t)\\
\dot{q}_2(t) &= v_2(t) - \xi(t)\\
\dot{v}_1(t) &= 2\big(u(t) + \lambda(t)\big)\\
\dot{v}_2(t) &= -2\lambda(t)\\
0 &= q_2(t) - q_1(t) - 1\\
0 &= v_2(t) - v_1(t).
\end{aligned} \tag{7.36}$$

For a given control force $u(t)$, Eq. (7.36) can be solved for $q_1(t)$, $q_2(t)$, the corresponding velocities $v_1(t)$, $v_2(t)$ and the Lagrange multipliers $\lambda(t)$ and $\xi(t)$. The control goal is to reach a rest position $v_2(t_f) = 0$ at the origin with $q_2(t_f) = 0$ for given initial conditions in minimum time.

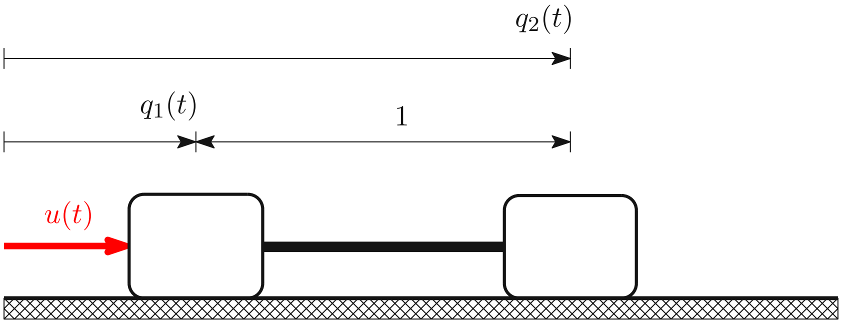

Figure 7.1: Two linked masses

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Hence, we are looking for the control force $u(t)$ which minimizes the cost functional

$$J = \int_{t_0}^{t_f} 1\,\mathrm{d}t = t_f - t_0, \tag{7.37}$$

where the constraint equations

$$\boldsymbol{\phi}(q_2(t), v_2(t)) := \begin{pmatrix} q_2(t) \\ v_2(t) \end{pmatrix}, \tag{7.38}$$

must become zero at the final time $t = t_f$. Since the control force appears linear in Eq. (7.36), we are dealing with a bang-bang control. Hence, the control is of the form $u(t) = \pm 1$ and can only occur either as full acceleration $\{+1\}$, as full deceleration $\{-1\}$, or as the switching sequences $\{+1, -1\}$ or $\{-1, +1\}$. To be more precise, if we consider the latter switching case with one switching point $t_{1,1}$, the control law is given by

$$u(t) := \begin{cases} -1 & \text{for} \quad t < t_{1,1} \\ +1 & \text{otherwise,} \end{cases} \tag{7.39}$$

where the switching point $t_{1,1}$ has to be determined. Before we can compute update formulas for $t_{1,1}$ and t_f, we have to compute the final time dependency of the function $\boldsymbol{\phi}$ from a forward solution of the equations of motion. As a first step, we compute the Jacobian matrices of $\boldsymbol{\phi}$ in Eq. (7.38) with respect to $\boldsymbol{q}(t)$ and $\boldsymbol{v}(t)$ yielding the constant matrices:

$$\boldsymbol{\phi}_{\boldsymbol{q}} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad \boldsymbol{\phi}_{\boldsymbol{v}} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}. \tag{7.40}$$

Note that the final conditions of the adjoint equations in Eq. (7.9) are simply given by $\boldsymbol{w}(t_f) = \mathbf{0}$ and $\boldsymbol{p}(t_f) = \mathbf{0}$. Since we utilize the switching point optimization, we do not require penalty terms, and consequently there are no excitation terms in Eq. (7.9). Hence, we obtain the trivial solution $\boldsymbol{w}(t) = \mathbf{0}$ and $\boldsymbol{p}(t) = \mathbf{0}$ for the entire interval $[t_0, t_f]$.

To derive the adjoint-influence equations in Eq. (7.15), we introduce the adjoint matrices $\mathbf{W} \in \mathbb{R}^{2\times 2}$, $\mathbf{P} \in \mathbb{R}^{2\times 2}$, $\boldsymbol{\mathcal{M}} \in \mathbb{R}^{1\times 2}$ and $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{1\times 2}$. Due to the simple structure of the equations of motion, many terms are zero and thus the adjoint-influence equations derived from Eq. (7.15)

$$\begin{aligned} \dot{\mathbf{W}}(t) &= -\begin{pmatrix} -1 \\ 1 \end{pmatrix} \boldsymbol{\mathcal{M}}(t) & \qquad \mathbf{0} &= \begin{pmatrix} -1 & 1 \end{pmatrix} \mathbf{W}(t) \\ \dot{\mathbf{P}}(t) &= -2\begin{pmatrix} -1 \\ 1 \end{pmatrix} \boldsymbol{\mathcal{S}}(t) - 2\mathbf{W}(t) & \qquad \mathbf{0} &= \begin{pmatrix} -1 & 1 \end{pmatrix} \mathbf{P}(t). \end{aligned} \tag{7.41}$$

are rather compact.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

The boundary conditions for the adjoint-influence equations provided by Eq. (7.25) can be computed by solving

$$\begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & 1 \\ -1 & 1 & 0 \end{pmatrix} \begin{pmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \\ L_{11} & L_{12} \end{pmatrix} \Bigg|_{t_f} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \end{pmatrix}, \tag{7.42}$$

which yields $W_{11}(t_f) = 1/2$, $W_{12}(t_f) = 0$, $W_{21}(t_f) = 1/2$ and $W_{22}(t_f) = 0$. Moreover, Eq. (7.26) reads

$$\begin{pmatrix} \frac{1}{2} & 0 & -1 \\ 0 & \frac{1}{2} & 1 \\ -1 & 1 & 0 \end{pmatrix} \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \\ U_{11} & U_{12} \end{pmatrix} \Bigg|_{t_f} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \tag{7.43}$$

and yields $P_{11}(t_f) = 0$, $P_{12}(t_f) = 1$, $P_{21}(t_f) = 0$ and $P_{22}(t_f) = 1$. Here, L_{11}, L_{12}, U_{11} and U_{12} are only introduced to solve the system of equations and are not required any further.

From the algebraic constraints in Eq. (7.41), we deduce $P_{11} = P_{21}$, $P_{12} = P_{22}$, $W_{11} = W_{21}$ and $W_{12} = W_{22}$, which are in perfect agreement with the boundary conditions for W_{ij} and P_{ij}. If we insert these relationships into the differential equations introduced in Eq. (7.41), we obtain $\boldsymbol{\mathcal{M}}(t) = \mathbf{0}$ and $\boldsymbol{\mathcal{S}}(t) = \mathbf{0}$. Hence, the differential equation in Eq. $(7.41)_1$ is given by $\dot{\mathbf{W}}(t) = \mathbf{0}$. From the boundary conditions, we then obtain

$$\mathbf{W}(t) = \begin{pmatrix} \frac{1}{2} & 0 \\ \frac{1}{2} & 0 \end{pmatrix} = \text{const.} \tag{7.44}$$

After inserting Eq. (7.44) into the differential equation for $\mathbf{P}(t)$ in Eq. $(7.41)_2$, we obtain

$$\dot{\mathbf{P}}(t) = \begin{pmatrix} -1 & 0 \\ -1 & 0 \end{pmatrix}. \tag{7.45}$$

With the boundary conditions for P_{ij} at $t = t_f$, we get

$$\mathbf{P}(t) = \begin{pmatrix} t_f - t & 1 \\ t_f - t & 1 \end{pmatrix}. \tag{7.46}$$

As a next step, we compute the multiplier $\boldsymbol{\nu}$ from Eq. (7.21) by using the abbreviations for $\mathbf{A}$ and $\boldsymbol{b}$ defined in Eq. (7.33) for bang-bang controls. The total time derivative of $\boldsymbol{\phi}$ at the final time is given by

$$\dot{\boldsymbol{\phi}}_f = \begin{pmatrix} \dot{q}_2 \\ \dot{v}_2 \end{pmatrix} \Bigg|_{t_f} = \begin{pmatrix} v_2 - \xi \\ -2\lambda \end{pmatrix} \Bigg|_{t_f}, \tag{7.47}$$

in which we inserted Eq. $(7.36)_2$ and Eq. $(7.36)_4$ for $\dot{q}_2$ and $\dot{v}_2$.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

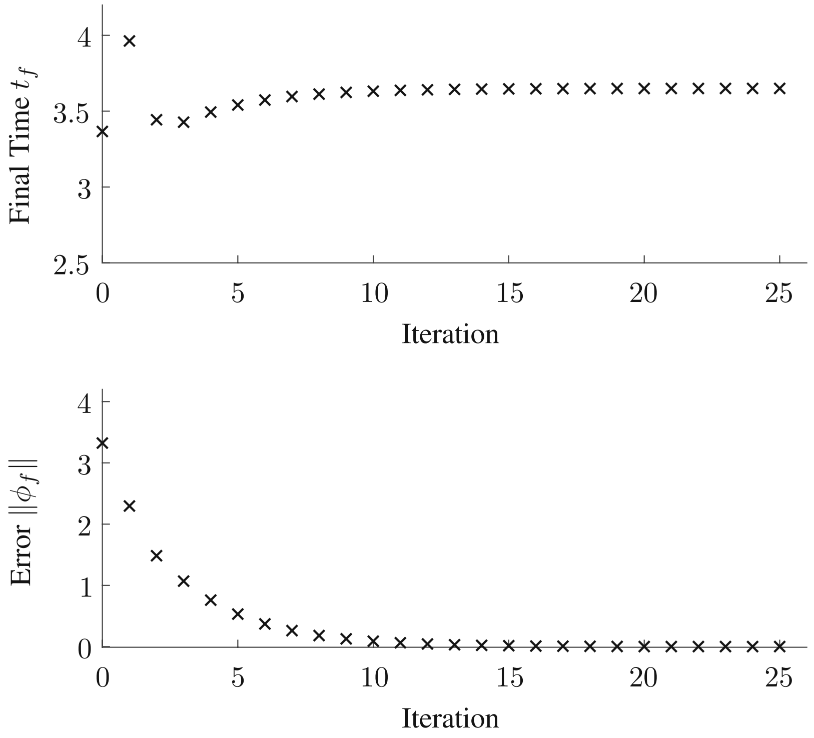

Figure 7.2: Convergence of the mass transfer problem

Now the $\mathbf{A}$-matrix from Eq. $(7.33)_1$ is given by

$$\mathbf{A} := 4\beta \begin{pmatrix} (t_f - t)^2 & t_f - t \\ t_f - t & 1 \end{pmatrix}\Bigg|_{t_{1,1}} + \alpha \begin{pmatrix} (v_2 - \xi)^2 & -2\lambda\,(v_2 - \xi) \\ -2\lambda\,(v_2 - \xi) & 4\lambda^2 \end{pmatrix}\Bigg|_{t_f} . \tag{7.48}$$

A glance at Eq. $(7.33)_2$ shows that $\boldsymbol{b} = \alpha\,(v_2(t_f) - \xi(t_f), -2\lambda(t_f))^{\mathsf{T}}$ and the multiplier $\boldsymbol{\nu} = (\nu_1, \nu_2)^{\mathsf{T}}$ is obtained by

$$\boldsymbol{\nu} = \frac{\varepsilon}{\kappa}\mathbf{A}^{-1} \begin{pmatrix} q_2(t_f) \\ v_2(t_f) \end{pmatrix} - \alpha\mathbf{A}^{-1} \begin{pmatrix} v_2(t_f) - \xi(t_f) \\ -2\lambda(t_f) \end{pmatrix} . \tag{7.49}$$

Since we observe a shift from -1 to $+1$ at $t_{1,1}$, we apply the positive sign in the update formula in Eq. (7.32). Hence, with $\bar{u} = 2$ and $\boldsymbol{f}_u = (1, 0)^{\mathsf{T}}$, the updates of the end time and of the switching point is finally given by

$$\begin{aligned} \delta t_f &= -\kappa\alpha\big[1 + \nu_1\,(v_2(t_f) - \xi(t_f)) - 2\nu_2\lambda(t_f)\big] \\ \delta t_{1,1} &= +2\kappa\beta\big[\,(t_f - t_{1,1})\,\nu_1 + \nu_2\big] . \end{aligned} \tag{7.50}$$

We choose the initial values $q_1(0) = -3$, $q_2(0) = -2$ and $v_1(0) = v_2(0) = 2.236$ and set the initial switching point to $t_{1,1} = 1.5$. We guess a final time $t_f = 3$ and choose the update parameters $\alpha = \beta = 1$, $\varepsilon = 0.3$ and $\kappa = 1$. The convergence analysis of the final time and of the end condition error is shown in Fig. 7.2. The trajectory to transfer the system to the origin in minimal time is depicted in Fig. 7.3.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

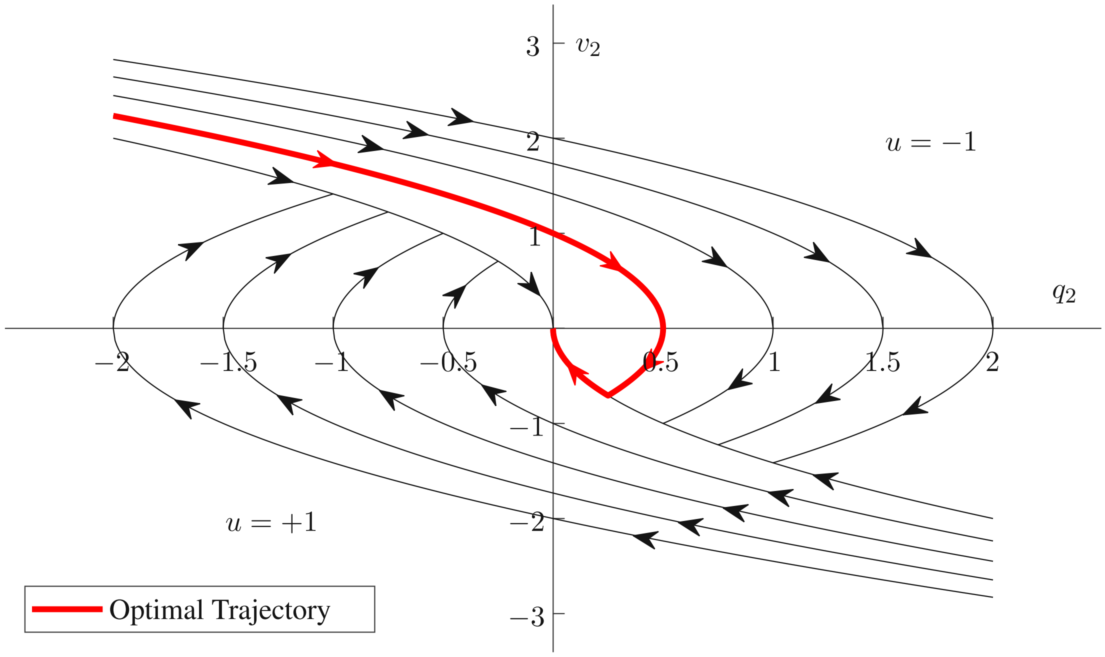

Figure 7.3: Switching arcs for two linked masses

However, in general, the adjoint equations in Eq. (7.9) and the adjoint-influence equations in Eq. (7.15) must be solved numerically by using an integration scheme. For this purpose we can use a multi-step procedure, as it is widely used in multibody dynamics for solving differential-algebraic equations. A particularly well suited method is the backward differentiation scheme, for which we derive the formulas in the next section.

7.3 A Backward Differentiation Scheme

Writing down the co-system differential-algebraic equations for a dynamic system in redundant coordinates is not a difficult exercise; obtaining solutions is another matter. Hence, this section is dedicated to the efficient solution of the adjoint differential system in Eq. (7.9) and the adjoint-influence differential system in Eq. (7.12). Therefore, we use a backward differentiation formula (BDF) which is known to be perfect for stiff problems, has superior stability properties and can be used to directly discretize differential-algebraic equations.

A discretization scheme for the adjoint equations, as demonstrated in [47], benefits from tailored solvers capable of performing an accurate and time-efficient solution. Note, if the forward solution is provided by the Hilbert-Hughes-Taylor (HHT) solver, then the hidden constraint $\boldsymbol{C}_{\boldsymbol{q}}(\boldsymbol{q}(t))\boldsymbol{\xi}(t) = \boldsymbol{0}$ is satisfied, i. e., $\boldsymbol{\xi}(t) = \boldsymbol{0}$. Since the numerical methods in this section already provide very long expressions, we will assume for simplicity that the forward solution is provided by the HHT solver and set $\boldsymbol{\xi}(t) = \boldsymbol{0}$. Since the final time t_f is free, we apply a transformation into a unit interval. Moreover, both sets of differential equations are solved backward in physical

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

time and so it is convenient to claim our new time coordinate τ to run from t_f to t_0. Hence, the relation between t and our new time coordinate τ is given by $t = t_f\,(1-\tau)$. If $z(t)$ is approached by $k+1$ points, the general form of the backward differentiation formula is given by

$$z'(\tau_n) \approx \sum_{i=0}^{k} \eta_i(\tau_n, \dots, \tau_{n-i}) z(\tau_{n-i}), \tag{7.51}$$

where the coefficients η_i depend on the sizes of the time step intervals. Note, if constant step sizes are used, then we get constant values for η_i depending on the integration order. For integration order $k = 1$, the coefficients for varying solver step sizes can be computed by

$$\eta_0 = \frac{1}{\tau_n - \tau_{n-1}}, \quad \eta_1 = -\frac{1}{\tau_n - \tau_{n-1}}. \tag{7.52}$$

If we insert Eq. (7.52) into Eq. (7.51), we obtain the implicit Euler method, whose order of accuracy is, of course, one. The first BDF integration step can be computed with the coefficients in Eq. (7.52). Due to the transformation to the τ-domain, the variables in the first step are given by $\tau_{n-1} = 0$ and $\tau_n = h$, where h is the step size of the first integration step. Afterwards the integration order can be raised and the subsequent step can be computed with the coefficients:

$$\begin{gathered}\eta_0 = \frac{2\tau_n - \tau_{n-1} - \tau_{n-2}}{(\tau_n - \tau_{n-1})\,(\tau_n - \tau_{n-2})}, \quad \eta_1 = \frac{\tau_n - \tau_{n-2}}{(\tau_{n-1} - \tau_{n-2})\,(\tau_{n-1} - \tau_n)}, \\ \eta_2 = \frac{\tau_n - \tau_{n-1}}{(\tau_{n-2} - \tau_n)\,(\tau_{n-2} - \tau_{n-1})}.\end{gathered} \tag{7.53}$$

The integration order $k = 1$ and $k = 2$ of the BDF method are A-stable, see [69, Sec. 12.12] for an elaborate derivation and a stability plot in the complex plane. However, in some cases it might be convenient to increase the integration order further, but care must be taken because even order $k = 3$ can lead to stability problems of the integration scheme. It has to be emphasized, order $k = 2$ is sufficient in most cases and delivers accurate results in our examples examined so far.

7.3.1 The Solution of the Adjoint Differential Equations

In this section, we discretize the differential equations and derive the resulting BDF rule by following Nachbagauer et al. [47]. Derivatives with respect to τ are denoted as primes and can be determined with $\mathrm{d}/\mathrm{d}t = \mathrm{d}/\mathrm{d}\tau \cdot \mathrm{d}\tau/\mathrm{d}t = -(1/t_f) \cdot \mathrm{d}/\mathrm{d}\tau$. Applying Eq. (7.51) to the adjoint system in Eq. (7.9), we obtain

$$\begin{aligned}\sum_{i=0}^{k} \eta_i \boldsymbol{w}(\tau_{n-i}) &= t_f \Big(\Pi_{\boldsymbol{q}}^{\mathsf{T}} + \mathbf{G}\boldsymbol{p} + \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\mu} + (\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v})_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\sigma}\Big)\Big|_{\tau_n} \\ \sum_{i=0}^{k} \eta_i \mathbf{M}(\boldsymbol{q}(\tau_{n-i}))\boldsymbol{p}(\tau_{n-i}) &= t_f \Big(\Pi_{\boldsymbol{v}}^{\mathsf{T}} + \boldsymbol{w} + \boldsymbol{f}_{\boldsymbol{v}}^{\mathsf{T}}\boldsymbol{p} + \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\sigma}\Big)\Big|_{\tau_n} \\ \boldsymbol{C}_{\boldsymbol{q}}(\boldsymbol{q}(\tau_n))\boldsymbol{p}(\tau_n) &= \boldsymbol{0} \\ \boldsymbol{C}_{\boldsymbol{q}}(\boldsymbol{q}(\tau_n))\boldsymbol{w}(\tau_n) &= \boldsymbol{0}.\end{aligned} \tag{7.54}$$

The values for $\boldsymbol{q}(t)$, $\boldsymbol{v}(t)$ and $\boldsymbol{\lambda}(t)$ are obtained by a forward simulation of the equations of motion in Eq. (7.1), and can be readily transformed to the dimensionless time coordinate τ by $\boldsymbol{q}(\tau) = \boldsymbol{q}(t_f(1-\tau))$, $\boldsymbol{v}(\tau) = \boldsymbol{v}(t_f(1-\tau))$ and $\dot{\boldsymbol{v}}(\tau) = \dot{\boldsymbol{v}}(t_f(1-\tau))$. As a first step, we compute $\boldsymbol{w}(\tau_n)$ from Eq. $(7.54)_1$, yielding

$$\boldsymbol{w}(\tau_n) = \frac{t_f}{\eta_0}\Big(\mathbf{G}\boldsymbol{p} + \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\mu} + (\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v})_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\sigma} + \Pi_{\boldsymbol{q}}^{\mathsf{T}}\Big)\Big|_{\tau_n} - \frac{1}{\eta_0}\sum_{i=1}^{k}\eta_i\boldsymbol{w}(\tau_{n-i}), \tag{7.55}$$

where we applied a simple index shift of the sum to solve the equation for $\boldsymbol{w}(\tau_n)$. After inserting $\boldsymbol{w}(\tau_n)$ from Eq. (7.55) into Eq. $(7.54)_2$, we get

$$\begin{aligned}&\left[\eta_0\mathbf{M}\boldsymbol{p} - \frac{t_f^2}{\eta_0}\left(\mathbf{G}\boldsymbol{p} + \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\mu} + (\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v})_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\sigma}\right) - t_f\boldsymbol{f}_{\boldsymbol{v}}^{\mathsf{T}}\boldsymbol{p} - t_f\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\sigma}\right]\Bigg|_{\tau_n}\\ &= t_f\Pi_{\boldsymbol{v}}^{\mathsf{T}}(\tau_n) + \frac{t_f^2}{\eta_0}\Pi_{\boldsymbol{q}}^{\mathsf{T}}(\tau_n) - \sum_{i=1}^{k}\left(\eta_i\mathbf{M}(\boldsymbol{q}(\tau_{n-i}))\boldsymbol{p}(\tau_{n-i}) + \frac{\eta_i}{\eta_0}t_f\boldsymbol{w}(\tau_{n-i})\right).\end{aligned} \tag{7.56}$$

Moreover, we also can insert $\boldsymbol{w}(\tau_n)$ from Eq. (7.55) into Eq. $(7.54)_4$, yielding

$$\begin{aligned}\left[\boldsymbol{C}_{\boldsymbol{q}}\mathbf{G}\boldsymbol{p} + \boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\mu} + \boldsymbol{C}_{\boldsymbol{q}}\left(\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v}\right)_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\sigma}\right]\Big|_{\tau_n} =& -\boldsymbol{C}_{\boldsymbol{q}}(\tau_n)\Pi_{\boldsymbol{q}}^{\mathsf{T}}(\tau_n)\\ &+\frac{1}{t_f}\boldsymbol{C}_{\boldsymbol{q}}(\tau_n)\sum_{i=1}^{k}\eta_i\boldsymbol{w}(\tau_{n-i}).\end{aligned} \tag{7.57}$$

A closer observation of Eq. (7.55) and of Eq. (7.57) shows that $\boldsymbol{q}(\tau_{n-i})$, $\boldsymbol{p}(\tau_{n-i})$ and $\boldsymbol{w}(\tau_{n-i})$ are known from the initial conditions or, respectively, from previous integration steps.
If we arrange Eq. (7.56), Eq. (7.57) and Eq. $(7.54)_3$ in matrix form, we receive:

$$\begin{aligned}&\begin{pmatrix}\eta_0\mathbf{M} - \frac{t_f^2}{\eta_0}\mathbf{G} - t_f\boldsymbol{f}_{\boldsymbol{v}}^{\mathsf{T}} & -\frac{t_f^2}{\eta_0}\left(\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v}\right)_{\boldsymbol{q}}^{\mathsf{T}} - t_f\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} & -\frac{t_f^2}{\eta_0}\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}\\ \boldsymbol{C}_{\boldsymbol{q}}\mathbf{G} & \boldsymbol{C}_{\boldsymbol{q}}\left(\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v}\right)_{\boldsymbol{q}}^{\mathsf{T}} & \boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}\\ \boldsymbol{C}_{\boldsymbol{q}} & \mathbf{0} & \mathbf{0}\end{pmatrix}\Bigg|_{\tau_n}\begin{pmatrix}\boldsymbol{p}(\tau_n)\\ \boldsymbol{\sigma}(\tau_n)\\ \boldsymbol{\mu}(\tau_n)\end{pmatrix}\\ &= \begin{pmatrix} t_f\Pi_{\boldsymbol{v}}^{\mathsf{T}}(\tau_n) + \frac{t_f^2}{\eta_0}\Pi_{\boldsymbol{q}}^{\mathsf{T}}(\tau_n) - \sum_{i=1}^{k}\left(\eta_i\mathbf{M}(\boldsymbol{q}(\tau_{n-i}))\boldsymbol{p}(\tau_{n-i}) + \frac{\eta_i}{\eta_0}t_f\boldsymbol{w}(\tau_{n-i})\right)\\ -\boldsymbol{C}_{\boldsymbol{q}}(\tau_n)\Pi_{\boldsymbol{q}}^{\mathsf{T}}(\tau_n) + \frac{1}{t_f}\boldsymbol{C}_{\boldsymbol{q}}\sum_{i=1}^{k}\eta_i\boldsymbol{w}(\tau_{n-i})\\ \mathbf{0}\end{pmatrix},\end{aligned} \tag{7.58}$$

which can be solved for $\boldsymbol{p}(\tau_n)$, $\boldsymbol{\sigma}(\tau_n)$ and $\boldsymbol{\mu}(\tau_n)$ after Eq. (7.55) is solved for $\boldsymbol{w}(\tau_n)$. In order to initiate the integration scheme, we set the integration order to $k = 1$ and insert the final values for $\boldsymbol{w}(1) = \boldsymbol{p}(1) = \mathbf{0}$ into Eq. (7.58). To achieve a higher accuracy of the integration scheme, the order k can be increased after Eq. (7.58) is solved for the first time.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

7.3.2 The Solution of the Adjoint-Influence Differential Equations

The differential equations for the adjoint-influence functions can be solved by Eq. (7.55) and Eq. (7.58). First, one has to compute consistent final conditions by solving Eq. (7.25) and Eq. (7.26) for $\mathbf{W}(t_f)$ and $\mathbf{P}(t_f)$. Then, $\mathbf{P}(t_f)$, $\boldsymbol{\mathcal{S}}(t_f)$ and $\boldsymbol{\mathcal{M}}(t_f)$ can be determined by solving

$$\left.\begin{pmatrix} \eta_0\mathbf{M} - \frac{t_f^2}{\eta_0}\mathbf{G} - t_f\boldsymbol{f}_{\boldsymbol{v}}^{\mathsf{T}} & -\frac{t_f^2}{\eta_0}\left(\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v}\right)_{\boldsymbol{q}}^{\mathsf{T}} - t_f\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} & -\frac{t_f^2}{\eta_0}\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \\ \boldsymbol{C}_{\boldsymbol{q}}\mathbf{G} & \boldsymbol{C}_{\boldsymbol{q}}\left(\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v}\right)_{\boldsymbol{q}}^{\mathsf{T}} & \boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}} \\ \boldsymbol{C}_{\boldsymbol{q}} & \mathbf{0} & \mathbf{0} \end{pmatrix}\right|_{\tau_n} \begin{pmatrix} \mathbf{P}(\tau_n) \\ \boldsymbol{\mathcal{S}}(\tau_n) \\ \boldsymbol{\mathcal{M}}(\tau_n) \end{pmatrix}$$
$$= \begin{pmatrix} -\sum_{i=1}^{k}\left(\eta_i\mathbf{M}(\boldsymbol{q}(\tau_{n-i}))\mathbf{P}(\tau_{n-i}) + \frac{\eta_i}{\eta_0}t_f\mathbf{W}(\tau_{n-i})\right) \\ \frac{1}{t_f}\boldsymbol{C}_{\boldsymbol{q}}\sum_{i=1}^{k}\eta_i\mathbf{W}(\tau_{n-i}) \\ \mathbf{0} \end{pmatrix}. \tag{7.59}$$

Afterwards, we can determine the next solver time step by first computing $\mathbf{W}(t_f)$ through

$$\mathbf{W}(\tau_n) = \frac{t_f}{\eta_0}\left.\left(\mathbf{G}\mathbf{P} + \boldsymbol{C}_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\mathcal{M}} + \left(\boldsymbol{C}_{\boldsymbol{q}}\boldsymbol{v}\right)_{\boldsymbol{q}}^{\mathsf{T}}\boldsymbol{\mathcal{S}}\right)\right|_{\tau_n} - \frac{1}{\eta_0}\sum_{i=1}^{k}\eta_i\mathbf{W}(\tau_{n-i}), \tag{7.60}$$

and then solving Eq. (7.59) again. Note, due to the independence of the adjoint and the adjoint-influence differential-algebraic equations, they can be solved simultaneously in order to save computation time.

7.3.3 The Algorithm

Now we summarize the descent algorithm, from which the update of time-optimal controls can be derived. The algorithm can be subdivided into eight steps.

1. Select an initial control history $\boldsymbol{u}(\tau)$ and/or switching points $t_{i,k}$ for bang-bang controls and guess a final time t_f.
2. Solve the equations of motion in Eq. (7.1) with the initial values $\boldsymbol{q}(0) = \boldsymbol{q}_0$ and $\dot{\boldsymbol{q}}(0) = \boldsymbol{v}_0$ in the time interval $t \in [t_0, t_f]$. The solution may be obtained, e. g. by applying the HHT integration scheme. Store $\boldsymbol{q}(t)$, $\boldsymbol{v}(t)$ and $\dot{\boldsymbol{v}}(t)$ along the forward simulation and compute them as functions of the dimensionless time coordinate τ by applying $\boldsymbol{q}(\tau) = \boldsymbol{q}(t_f(1-\tau))$, $\boldsymbol{v}(\tau) = \boldsymbol{v}(t_f(1-\tau))$ and $\dot{\boldsymbol{v}}(\tau) = \dot{\boldsymbol{v}}(t_f(1-\tau))$.
3. Solve the adjoint differential-algebraic equation system in Eq. (7.9) by computing $\boldsymbol{w}(\tau_n)$ from Eq. (7.55) first, and then solve the linear system of equations in Eq. (7.58) for $\boldsymbol{p}(\tau_n)$, $\boldsymbol{\sigma}(\tau_n)$ and $\boldsymbol{\mu}(\tau_n)$. Both steps are initiated with the trivial final conditions $\boldsymbol{w}(1) = \mathbf{0}$ and $\boldsymbol{p}(1) = \mathbf{0}$. Note, in the first step, the integration order k is initiated with one and may be increased afterwards. If there are no state constraints in form of penalty functions, one derives the trivial solution $\boldsymbol{w}(\tau) = \mathbf{0}$ and $\boldsymbol{p}(\tau) = \mathbf{0}$ for the whole τ-interval.

4. Calculate consistent final conditions by solving Eq. (7.25) and Eq. (7.26) for $\mathbf{W}(t_f)$ and $\mathbf{P}(t_f)$ which correspond to the final conditions in the τ-domain.

5. Using the final conditions from step 4, solve the adjoint-influence matrix differential equations in Eq. (7.15). Therefore, apply the BDF-scheme introduced in Eq. (7.60) and in Eq. (7.59). Note, the solution process is the same as described in step 3 for the adjoint system.

6. Compute the multiplier $\boldsymbol{\nu}$ from Eq. (7.21).

7. Update the final time by

$$\delta t_f = -\kappa\alpha\Big(1 + \Pi_f + \boldsymbol{\nu}^\mathsf{T}\boldsymbol{\phi}_f\Big), \tag{7.61}$$

and update the control by

$$\delta\boldsymbol{u}(t) = -\kappa\Big(\Pi_{\boldsymbol{u}}^\mathsf{T} + \boldsymbol{f}_{\boldsymbol{u}}^\mathsf{T}\boldsymbol{p} + \boldsymbol{f}_{\boldsymbol{u}}^\mathsf{T}\mathbf{P}\,\boldsymbol{\nu}\Big), \tag{7.62}$$

or, respectively, for bang-bang controls

$$\delta t_{i,k} = \mp\kappa\beta\bar{u}_i\Big(\boldsymbol{f}_{u_i}^\mathsf{T}\boldsymbol{p} + \boldsymbol{f}_{u_i}^\mathsf{T}\mathbf{P}\,\boldsymbol{\nu}\Big)\Big|_{t_{i,k}/t_f}, \tag{7.63}$$

by adding δt_f, $\delta\boldsymbol{u}(t)$ and $\delta t_{i,k}$ to the previous estimates of t_f, $\boldsymbol{u}(t)$ and $t_{i,k}$. Herein, α and β are again scaling factors for the conditioning of the optimization problem and ε has to be tuned to approach the final conditions. Finally, the variable κ determines the size of the update and has to be chosen appropriately.

8. Repeat steps 1 through 7 until the final error and the update gradients are sufficiently small. In this case we can assume that we have found a constrained (local) minimum of the optimal control problem.

7.4 Examples

In this section we will now threat two problems from robotics in order to apply the proposed theory. Modern robot systems are developed according to innovative technologies of lightweight construction. This reduces the moving mass of the system and makes the robots highly agile. Hence, the knowledge of the optimal control law of robots related to an optimization criterion, e. g., the end time, is of high interest. So far, in robotics, the computation of time-optimal manipulation is usually performed by parameterizing the trajectory, e. g. by spline functions, and optimizing the grid points instead. This parameterization reduces the dimension of the optimization task and thus allows to find a solution to the problem. In opposite, the presented method is not subject to any restrictions, can be applied to complete time histories, and is even open to arbitrary parameterizations of the control variables.

First, we consider the planar robot arm from Sec. 6.4.2, which is now described by redundant coordinates to verify the proposed theory and to show its correctness. Once again, the problem

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

is solved using, on the one hand, a penalty approach to limit the drive torques and, on the other hand, the bang-bang method to directly identify the optimal switching points as described in Section 6.4.2. Second, a rigid multibody model of a complex 3D-robot model will be investigated to minimize the time for a point-to-point operation in order to demonstrate the applicability for more complex industrial problem setups.

7.4.1 Planar Robot Arm

As a first example, the SCARA robot arm from Sec. 6.4.2 is investigated by using a redundant coordinate formulation. We consider the robot arm in Fig. 6.12 consisting of two bodies and search for controls so that the robot moves from point-to-point in minimal time. Therefore, we introduce the redundant coordinates

$$\boldsymbol{q}(t) = \left(x_1, y_1, \varphi_1, x_2, y_2, \varphi_2, x_3, y_3\right)^{\mathsf{T}}, \tag{7.64}$$

in which (x_1, y_1, φ_1) and (x_2, y_2, φ_2) denote the coordinates of the centers of gravity of the first and second robot arm. Moreover, (x_3, y_3) describe the position of the tool center point (TCP). Since the coordinates are not independent, we observe six algebraic constraint equations

$$\boldsymbol{C}(\boldsymbol{q}) = \begin{pmatrix} x_1(t) - s_1\cos(\varphi_1(t)) \\ y_1(t) - s_1\sin(\varphi_1(t)) \\ x_2(t) - l_1\cos(\varphi_1(t)) - s_2\cos(\varphi_2(t)) \\ y_2(t) - l_1\sin(\varphi_1(t)) - s_2\sin(\varphi_2(t)) \\ x_3(t) - l_1\cos(\varphi_1(t)) - l_2\cos(\varphi_2(t)) \\ y_3(t) - l_1\sin(\varphi_1(t)) - l_2\sin(\varphi_2(t)) \end{pmatrix}, \tag{7.65}$$

which describe the kinematic chain of the robot. Herein, l_1, and l_2 denote the lengths of the robot axes while the center of gravity positions are given by s_1 and s_2. Due to the redundant formulation, the mass matrix is constant and readily given by

$$\mathbf{M} = \operatorname{diag}\left(m_1, m_1, J_1, m_2, m_2, J_2, m_3, m_3\right). \tag{7.66}$$

The SCARA robot is again actuated by the control torques $u_1(t)$ and $u_2(t)$, hence the generalized forces are given by

$$\boldsymbol{f}(u_1(t), u_2(t)) = \begin{pmatrix} 0 \\ 0 \\ u_1(t) - u_2(t) \\ 0 \\ 0 \\ u_2(t) \\ 0 \\ 0 \end{pmatrix}, \tag{7.67}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

which can be derived from the principle of virtual work. Finally, the cost functional to be minimized is given by

$$J = \int_{t_0}^{t_f} 1 \mathrm{d}t. \tag{7.68}$$

Since we use a redundant formulation of the robot, we can directly prescribe the position and velocity of the TCP. Hence, the final conditions read

$$\boldsymbol{\phi}(x_3, y_3, v_{x_3}, v_{y_3}) := \begin{pmatrix} x_3(t) - x_f \\ y_3(t) - y_f \\ v_{x_3}(t) - v_{x_f} \\ v_{y_3}(t) - v_{y_f} \end{pmatrix}, \tag{7.69}$$

in which the end position is the same as in the robot example in Sec. 6.4.2 and is given by $x_f = 1$ and $y_f = 1$. A glance at the function $\boldsymbol{\phi}(x_3, y_3, v_{x_3}, v_{y_3})$ shows once again the elegance of the proposed method using a redundant formulation. If a minimal set of only two generalized coordinates is used to describe the robot equations, the desired final conditions can not be addressed directly and the function $\boldsymbol{\phi}$ becomes more complicated.

(a) The Penalty Approach

Since the controls $u_1(t)$ and $u_2(t)$ appear linear in the Hamiltonian, we first use a penalty approach to put bounds on the driving torques. Hence, we add the function $\Pi(u_1(t), u_2(t))$ to the cost functional in Eq. (7.68) yielding

$$J = \int_{t_0}^{t_f} \left[1 + \Pi(u_1(t), u_2(t))\right] \mathrm{d}t, \tag{7.70}$$

where $\Pi(u_1, u_2) := \mu_1 \Pi_1(u_1(t)) + \mu_2 \Pi_2(u_2(t))$ is again defined by

$$\Pi_i(u_i(t)) := \begin{cases} 0 & \text{for} \quad |u_i(t)| < u_i^* \\ \frac{1}{2}\left(|u_i(t)| - u_i^*\right)^2 & \text{otherwise.} \end{cases} \tag{7.71}$$

The computations are carried out with the same model parameters and the same weighting parameters μ_1 and μ_2 for the penalty functions as presented in Sec. 6.4.2. For the optimization procedure, we use the parameter setup given by $\alpha = 0.5$, $\varepsilon = 0.1$ and $\kappa = 0.2$. Finally, the results are depicted in Fig. 7.4 and Fig. 7.5. The final time t_f is reduced from 3 s to 1.8335 s, while the error in the prescribed final conditions approaches zero, as shown in Fig. 7.4. The control histories and the switching functions (from Eq. (7.29)) are plotted in Fig. 7.5 which are in agreement with the results from Sec. 6.4.2.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

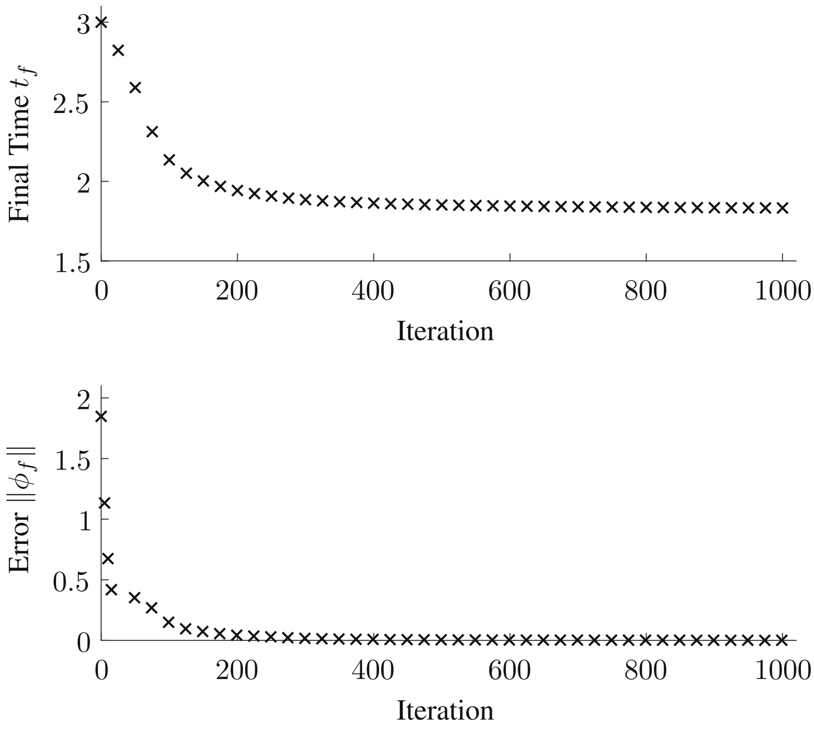

Figure 7.4: Convergence of conventional control optimization using a penalty approach

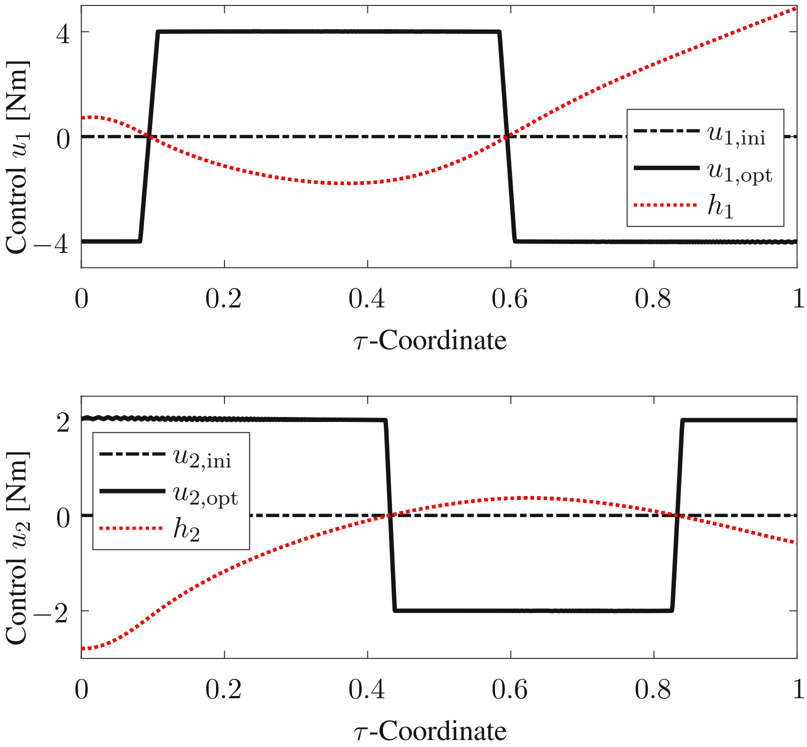

Figure 7.5: Controls for the rest-to-rest maneuver

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

(b) The Bang-Bang Approach

Armed with the knowledge that the controls converge to pure bang-bang controls, we can again determine the switching times instead of calculating the complete control history. Recall, one benefit of the switching point optimization is that only the end time is included in the cost functional in Eq. (7.68) yielding high accurate results, since no penalty functions need to be weighted to put bounds on controls. Hence, in case of bang-bang controls, the driving torques can be defined by

$$u_1(t) := \begin{cases} -u_1^* & \text{for} \quad t < t_{1,1} \\ +u_1^* & \text{for} \quad t_{1,1} \leq t \leq t_{1,2} \\ -u_1^* & \text{for} \quad t_{1,2} < t \end{cases}, \quad u_2(t) := \begin{cases} +u_2^* & \text{for} \quad t < t_{2,1} \\ -u_2^* & \text{for} \quad t_{2,1} \leq t \leq t_{2,2} \\ +u_2^* & \text{for} \quad t_{2,2} < t. \end{cases} \tag{7.72}$$

which are equal to the definition in Eq. (6.57). The initial controls are defined by the switching points: $t_{1,1} = 0.4$, $t_{1,2} = 1.4$, $t_{2,1} = 0.2$ and $t_{2,2} = 0.8$ referring to the SCARA robot described by a minimal set of coordinates in Sec. 6.4.2. For the switching point optimization, we use $\alpha = 1$, $\varepsilon = 0.05$ and $\kappa = 0.1$ yielding the results in Fig. 7.6 and Fig. 7.7. The optimal switching points are given by $t_{1,1} = 0.1757$, $t_{1,2} = 1.0917$, $t_{2,1} = 0.7913$, and $t_{2,2} = 1.5240$ and the final time can be reduced from 2 s to 1.8320 s. As expected, Fig. 7.6 shows a faster convergence than the penalty approach (in Fig. 7.4). The resulting controls are shown in Fig. 7.7. Herein, the solid line shows the time-optimal solution of the system described in redundant coordinates, while the dashed line is related to the solution of the system in minimal coordinates.

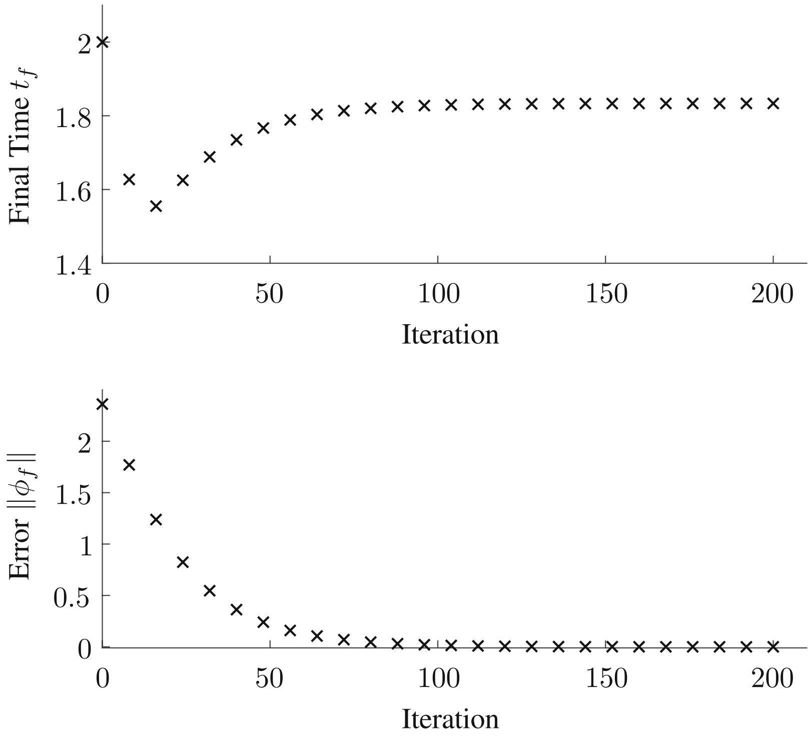

Figure 7.6: Convergence of the switching point optimization

TU Bibliothek Your knowledge hub WIEN
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

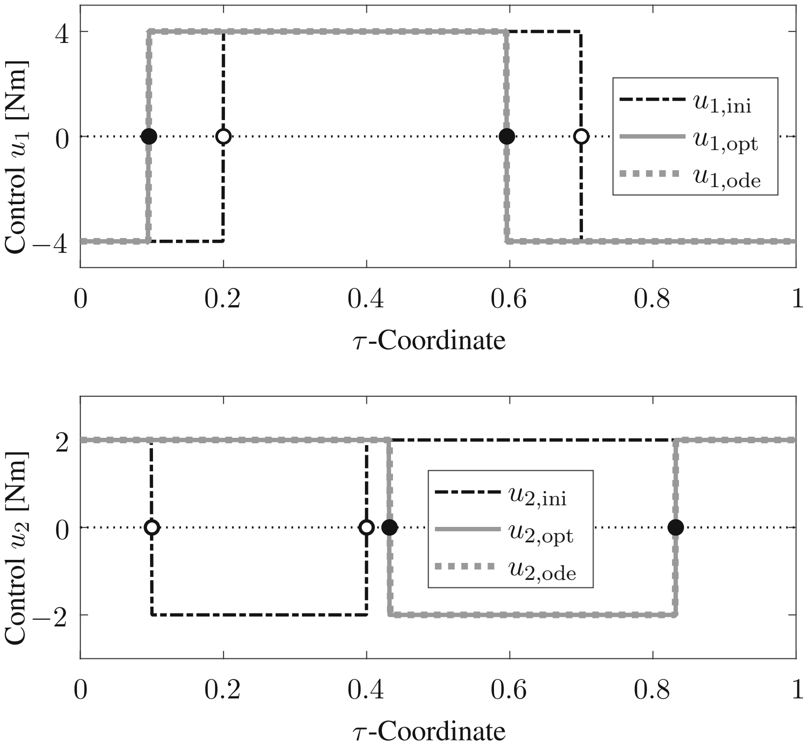

Figure 7.7: Bang-bang controls for the rest-to-rest maneuver

7.4.2 Industrial Robot

As a next example, we consider a rigid multibody model of a robot consisting of three robot arms $\mathcal{B}_1$, $\mathcal{B}_2$, $\mathcal{B}_3$ and a particle mass at the end of the kinematic chain. The three-dimensional model is depicted in Fig. 7.8 and has basically three independent degrees of freedom. However, we describe the system by the generalized dependent coordinates introduced by

$$\boldsymbol{q}(t) = \big(x_1, y_1, \varphi_1, x_2, y_2, z_2, \varphi_2, x_3, y_3, z_3, \varphi_3, x_4, y_4, z_4\big)^{\mathsf{T}}. \tag{7.73}$$

The center of gravity positions of the robot arms $\mathcal{B}_1$, $\mathcal{B}_2$ and $\mathcal{B}_3$ are described by (x_1, y_1, a_1), (x_2, y_2, z_2) and (x_3, y_3, z_3). The coordinates φ_1, φ_2 and φ_3 indicate the absolute angular position to the environment. Furthermore, the position of the end effector is represented by (x_4, y_4, y_4). The mass matrix of the robot is therefore given by

$$\mathbf{M} = \mathrm{diag}\big(m_1, m_1, J_1, m_2, m_2, m_2, J_2, m_3, m_3, m_3, J_3, m_4, m_4, m_4\big), \tag{7.74}$$

which has a diagonal structure and is constant due to the redundant formulation.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

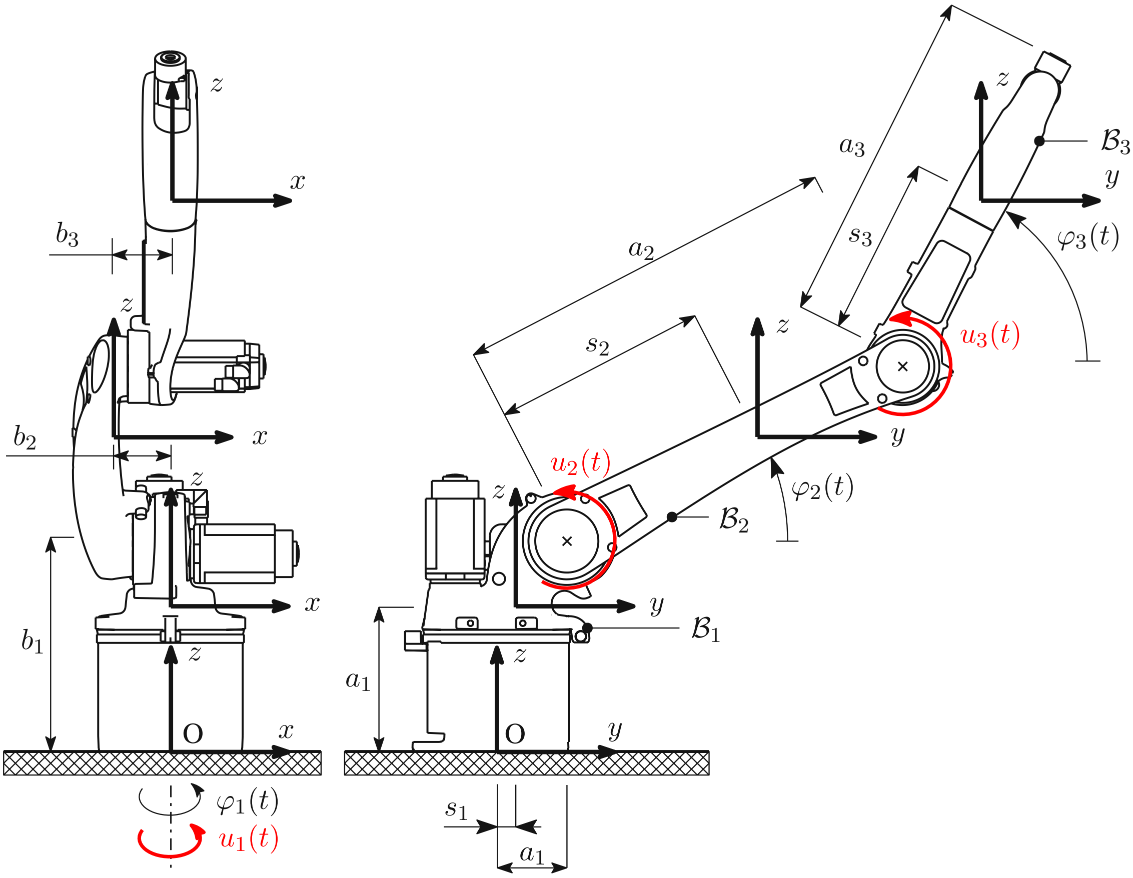

Figure 7.8: Industrial robot model

The coordinates in Eq. (7.73) are not independent, but constrained by the vector

$$\boldsymbol{C}(\boldsymbol{q}(t)) =$$

$$\begin{pmatrix}
x_1(t) + s_1 \sin(\varphi_1(t)) \\
y_1(t) - s_1 \cos(\varphi_1(t)) \\
x_2(t) - b_2 \cos(\varphi_1(t)) + \left[a_1 + s_2 \cos(\varphi_2(t))\right] \sin(\varphi_1(t)) \\
y_2(t) - b_2 \sin(\varphi_1(t)) - \left[a_1 + s_2 \cos(\varphi_2(t))\right] \cos(\varphi_1(t)) \\
z_2(t) - b_1 - s_2 \sin(\varphi_2(t)) \\
x_3(t) + b_{23} \cos(\varphi_1(t)) + \left[a_1 + a_2 \cos(\varphi_2(t)) + s_3 \cos\left(\varphi_2(t) + \varphi_3(t)\right)\right] \sin(\varphi_1(t)) \\
y_3(t) + b_{23} \sin(\varphi_1(t)) - \left[a_1 + a_2 \cos(\varphi_2(t)) + s_3 \cos\left(\varphi_2(t) + \varphi_3(t)\right)\right] \cos(\varphi_1(t)) \\
z_3(t) - b_1 - a_2 \sin(\varphi_2(t)) - s_3 \sin\left(\varphi_2(t) + \varphi_3(t)\right) \\
x_4(t) + b_{23} \cos(\varphi_1(t)) + \left[a_1 + a_2 \cos(\varphi_2(t)) + a_3 \cos\left(\varphi_2(t) + \varphi_3(t)\right)\right] \sin(\varphi_1(t)) \\
y_4(t) + b_{23} \sin(\varphi_1(t)) - \left[a_1 + a_2 \cos(\varphi_2(t)) + a_3 \cos\left(\varphi_2(t) + \varphi_3(t)\right)\right] \cos(\varphi_1(t)) \\
z_4(t) - b_1 - a_2 \sin(\varphi_2(t)) - a_3 \sin\left(\varphi_2(t) + \varphi_3(t)\right)
\end{pmatrix} \tag{7.75}$$

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

in which we introduced the abbreviation $b_{23} = (b_3 - b_2)$. The robot model is actuated by the driving torques $u_1(t)$, $u_2(t)$ and $u_3(t)$. Hence, the force vector including the control torques is given by

$$\boldsymbol{f}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \begin{pmatrix} 0 \\ 0 \\ u_1(t) - d_1\dot{\varphi}_1(t) \\ 0 \\ 0 \\ 0 \\ u_2(t) - u_3(t) - d_2\dot{\varphi}_2(t) + d_3\left(\dot{\varphi}_3(t) - \dot{\varphi}_2(t)\right) \\ 0 \\ 0 \\ 0 \\ u_3(t) - d_3\left(\dot{\varphi}_3(t) - \dot{\varphi}_2(t)\right) \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{7.76}$$

Initially, the robot is at rest, therefore the total time derivatives of the initial coordinates are given by $\dot{\boldsymbol{q}}_0 = \boldsymbol{0}$. When choosing the initial position, we have to be careful that, on the one hand, the initial position is feasible and, on the other hand, a singular robot constellation is avoided. Thus, we do not specify the TCP coordinates of the robot directly, but rather prescribe the initial orientation of the robot arms by $\varphi_{1,0} = \pi/6$, $\varphi_{2,0} = \pi/6$ and $\varphi_{3,0} = -\pi/4$, and compute the initial position with the kinematic relations:

$$\boldsymbol{q}_0 = \begin{pmatrix} -s_1\sin(\varphi_{1,0}) \\ s_1\cos(\varphi_{1,0}) \\ \varphi_{1,0} \\ b_2\cos(\varphi_{1,0}) - \left[a_1 + s_2\cos(\varphi_{2,0})\right]\sin(\varphi_{1,0}) \\ b_2\sin(\varphi_{1,0}) + \left[a_1 + s_2\cos(\varphi_{2,0})\right]\cos(\varphi_{1,0}) \\ b_1 + s_2\sin(\varphi_{2,0}) \\ \varphi_{2,0} \\ b_{23}\cos(\varphi_{1,0}) - \left[a_1 + a_2\cos(\varphi_{2,0}) + s_3\cos\left(\varphi_{2,0} + \varphi_{3,0}\right)\right]\sin(\varphi_{1,0}) \\ b_{23}\sin(\varphi_{1,0}) + \left[a_1 + a_2\cos(\varphi_{2,0}) + s_3\cos\left(\varphi_{2,0} + \varphi_{3,0}\right)\right]\cos(\varphi_{1,0}) \\ b_1 + a_2\sin(\varphi_{2,0}) + s_3\sin\left(\varphi_{2,0} + \varphi_{3,0}\right) \\ \varphi_{3,0} \\ b_{23}\cos(\varphi_{1,0}) - \left[a_1 + a_2\cos(\varphi_{2,0}) + a_3\cos\left(\varphi_{2,0} + \varphi_{3,0}\right)\right]\sin(\varphi_{1,0}) \\ b_{23}\sin(\varphi_{1,0}) + \left[a_1 + a_2\cos(\varphi_{2,0}) + a_3\cos\left(\varphi_{2,0} + \varphi_{3,0}\right)\right]\cos(\varphi_{1,0}) \\ b_1 + a_2\sin(\varphi_{2,0}) + a_3\sin\left(\varphi_{2,0} + \varphi_{3,0}\right) \end{pmatrix}. \tag{7.77}$$

TU Bibliothek, WIEN Your knowledge hub
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Bibliothek WIEN Your knowledge hub

In summary, we face an optimization problem in which we minimize

$$J = \int_{t_0}^{t_f} \left[1 + \Pi(u_1(t), u_2(t), u_3(t))\right] \mathrm{d}t, \tag{7.78}$$

Referring to the penalty approach of the planar robot model, we introduce a penalty term of the form

$$\Pi(u_1, u_2, u_3) := \mu_1 \Pi_1(u_1(t)) + \mu_2 \Pi_2(u_2(t)) + \mu_3 \Pi_3(u_3(t)), \tag{7.79}$$

where μ_1, μ_2 and μ_3 are weighting factors for the penalty functions to comply with the control restrictions, e. g. for $u_1^* = u_2^* = u_3^* = 20$ we set $\mu_1 = \mu_2 = \mu_3 = 1$. The penalty terms are again readily formulated by

$$\Pi_i(u_i(t)) := \begin{cases} 0 & \text{for} \quad |u_i(t)| < u_i^* \\ \dfrac{1}{2}\left(|u_i(t)| - u_i^*\right)^2 & \text{otherwise.} \end{cases} \tag{7.80}$$

The end position of the robot can be expressed by the final constraints given by

$$\boldsymbol{\phi}(x_4, y_4, z_4, v_{x_4}, v_{y_4}, v_{z_4}) = \begin{pmatrix} x_4(t) - x_f \\ y_4(t) - y_f \\ z_4(t) - z_f \\ v_{x_4}(t) \\ v_{y_4}(t) \\ v_{z_4}(t) \end{pmatrix}, \tag{7.81}$$

where the latter three conditions address the rest position of the robot and the end position is again defined via the orientation angles $\varphi_{1,f} = -\pi/4$, $\varphi_{2,f} = \pi/3$ and $\varphi_{3,f} = -\pi/4$ yielding

$$\begin{aligned} x_f &= b_{23}\cos(\varphi_{1,f}) - \Big[a_1 + a_2\cos(\varphi_{2,f}) + a_3\cos\left(\varphi_{2,f} + \varphi_{3,f}\right)\Big]\sin(\varphi_{1,f}) \\ y_f &= b_{23}\sin(\varphi_{1,f}) + \Big[a_1 + a_2\cos(\varphi_{2,f}) + a_3\cos\left(\varphi_{2,f} + \varphi_{3,f}\right)\Big]\cos(\varphi_{1,f}) \\ z_f &= b_1 + a_2\sin(\varphi_{2,f}) + a_3\sin\left(\varphi_{2,f} + \varphi_{3,f}\right). \end{aligned} \tag{7.82}$$

Note, the adjoint-influence equations, the $\mathbf{A}$-matrix, the $\boldsymbol{b}$-vector, and the update formulas are far too complicated to state here in comprehensible form. However, the derivation is readily accomplished by following the proposed algorithm in Sec. 7.3.3. For the numerical calculation we use the parameter set given by the damping coefficients $d_1 = 0.5\,\mathrm{Nms}$, $d_2 = 0.5\,\mathrm{Nms}$, $d_3 = 0.5\,\mathrm{Nms}$, the lengths $b_1 = 0.450\,\mathrm{m}$, $a_1 = 0.150\,\mathrm{m}$, $a_2 = 0.810\,\mathrm{m}$, $a_3 = 0.740\,\mathrm{m}$, $b_2 = 0.142\,\mathrm{m}$, $b_3 = 0.147\,\mathrm{m}$, the distances to the center of gravity $s_1 = 0.051\,\mathrm{m}$, $s_2 = 0.361\,\mathrm{m}$, $s_3 = 0.243\,\mathrm{m}$, the masses $m_1 = 13\,\mathrm{kg}$, $m_2 = 8\,\mathrm{kg}$, $m_3 = 10\,\mathrm{kg}$, $m_4 = 1\,\mathrm{kg}$ and the mass moments of inertia $J_1 = 0.196\,\mathrm{kg\,m^2}$, $J_2 = 0.468\,\mathrm{kg\,m^2}$, $J_3 = 0.536\,\mathrm{kg\,m^2}$. Finally, the results are obtained by the update parameter $\varepsilon = 0.08$ and the conditioning parameter $\alpha = 10$.

So far, we have used constant update values for κ in all our examples, on the one hand to emphasize the robustness of the algorithm, and on the other hand for convenience to avoid the implementation

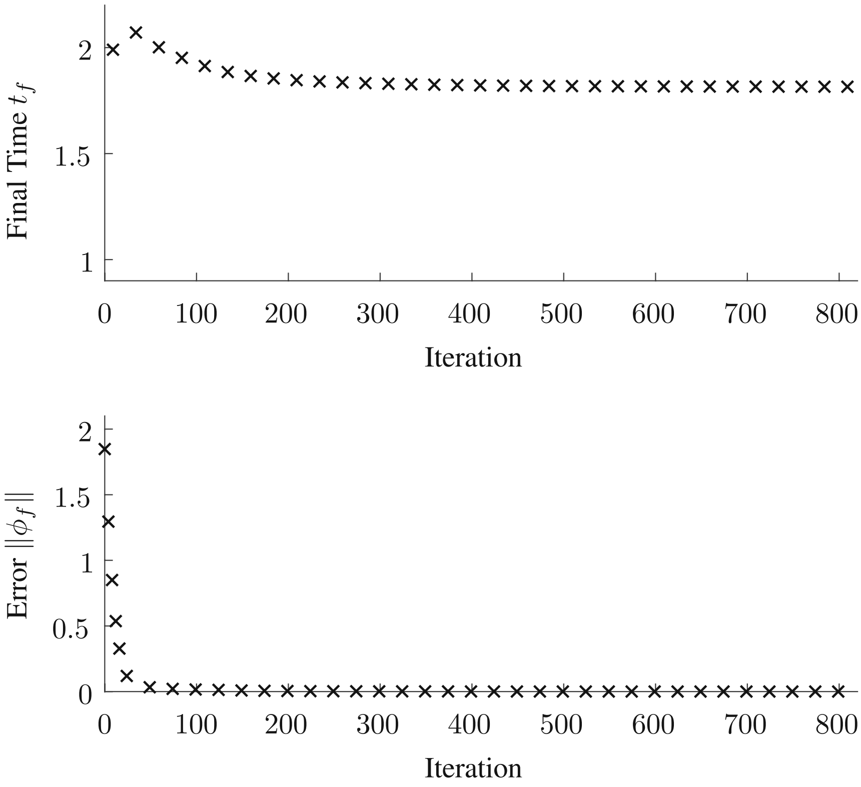

Figure 7.9: Convergence of the industrial robot model

of sophisticated step-size procedures. However, some of the results so far show slow convergence and initial start-up attempts of the present problem show that convergence suffers with increasing complexity. Hence, we now introduce a simple step size control using a merit function as a trade-off between convergence rate and implementation effort. Merit functions are employed in some cases for constrained optimization problems. A simple suggestion for a merit function with equality constraints, such as described in [44, Sec. 15.5], can also be used in the context of our problem and is formulated by

$$\mathcal{J} := \int_{t_0}^{t_f} \big[1 + \Pi(u_1, u_2, u_3)\big]\,\mathrm{d}t + \eta \sum_{i=1}^{6} \big|\phi_i(x_4, y_4, z_4, v_{x_4}, v_{y_4}, v_{z_4})\big|, \tag{7.83}$$

in which we introduced the weighting factor $\eta = 10$ controlling the violation of the final constraint conditions and ϕ_i denoting the i-th element of $\boldsymbol{\phi}(x_4, y_4, z_4, v_{x_4}, v_{y_4}, v_{z_4})$. The function $\mathcal{J}$ can then be used to apply classical update step size controls for κ in Eq. (7.61) and Eq. (7.62) as presented in Sec. 3.1.2. Finally, Fig. 7.9 shows the reduction of the cost functional over iterations. After approximately 500 iterations we stop the optimization, while the final time can be reduced from $2\,\mathrm{s}$ to $1.8335\,\mathrm{s}$. We observe that using a step size control based on a merit function exhibits a much better convergence rate than previous results using a constant update step size. Figure 7.10 visualizes the time-optimal trajectory while showing the initial and final robot configuration. In Fig. 7.11 the computed control signals and the corresponding switching functions from Eq. (7.29) are plotted, where the switching functions perfectly match the zero crossings of the controls.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

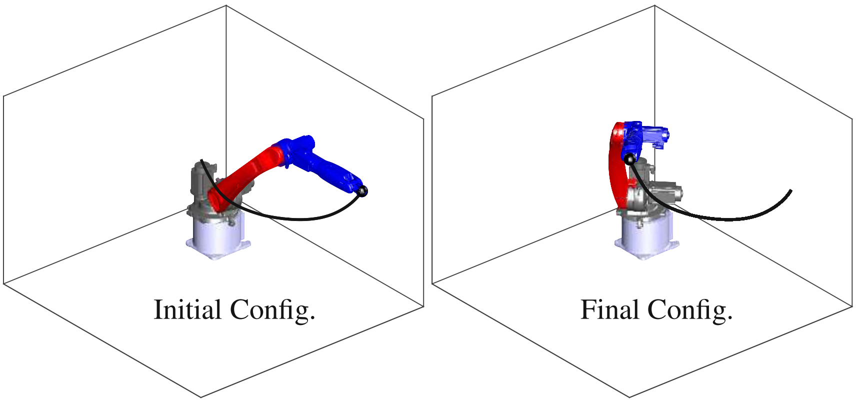

Figure 7.10: Configuration of the point-to-point operation of the industrial robot model

Note that all three controls converge to pure bang-bang controls with a total number of five switching points, while the control limits are well maintained by the penalty functions.

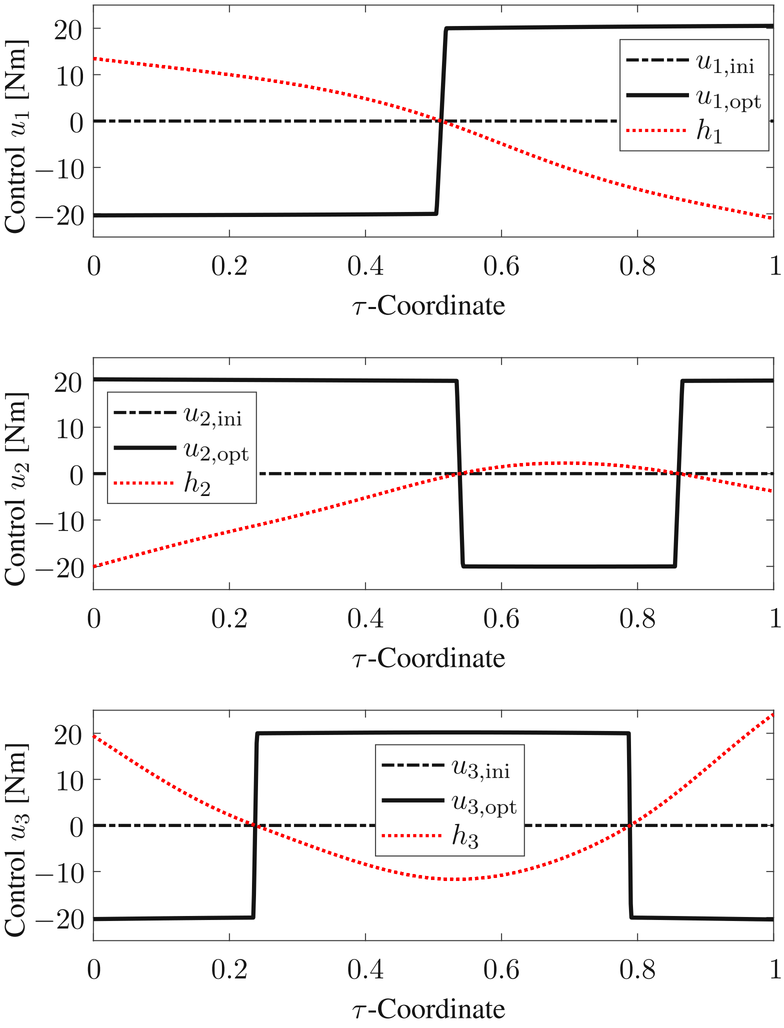

Figure 7.11: Controls of the industrial robot model

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Chapter 8

Conclusion

In this thesis, a gradient-based solution method is presented for determining the time-optimal control of dynamical systems subject to two classes of terminal constraints. Compared to conventional solution strategies related to the underlying boundary value problem, the proposed method converges even when choosing initial controls that are far away from the optimal solution. Many examples demonstrate the efficiency and accuracy of the proposed method and its applicability to time-optimal control problems from modern engineering.

8.1 Summary

While Chapter 2, 3 and 4 mainly discuss state of the art methods, novel methodological approaches have been elaborated in Chapter 5, 6 and 7.

In Chapter 5 and Chapter 6 we have introduced the adjoint gradient computation for systems described by a minimal set of independent coordinates, which presents a modern point of view on the solution of the two-point boundary value problem in optimal control. In Chapter 5, two approaches are presented, involving a hybrid formulation and a complete elimination of the time coordinate. The difference between both approaches can be summarized as a trade-off between robustness and implementation effort in standard software. However, both methods are open to use one of many standard numerical methods for minimizing a function, as all kind of Newton-like methods. In contrast, a more general approach is described in Chapter 6, which, however, is limited to gradient methods and thus has a lower convergence rate. Nevertheless, the nonlinear conjugate gradient method can be conceived, incorporating information from a prior update step to improve the algorithm convergence.

Finally, in Chapter 7 we presented a new way to solve time-optimal control problems in multibody dynamics. The novelty lies in the adaptation of the adjoint method to time-optimal control problems to differential-algebraic equations. With this method, one obtains an analytical formula for the gradient for which two additional systems of adjoint equations must be solved. For numerical evaluation, the gradient formula can be discretized arbitrarily fine without increasing the computational cost.

Further advantages of the adjoint approach in multibody dynamics can be summarized as follows:

- A desirable feature of the proposed method is that the initial controls do not have to satisfy the final constraints a priori.
- By using an efficient gradient-based approach, the solution of a two-point boundary value problem, which often requires the use of sophisticated homotopy strategies, can be avoided.
- After each successful control update, an improved control is obtained, which reduces the functional costs. The method can also be applied to more complex systems to find time-optimal controls. When using standard simulation software, e. g., multibody simulation programs, the gradient calculation could be automated if a module for the adjoint equations, as described in Section 7.3, is implemented.
- Singular time intervals in which the optimal control cannot be determined directly from Pontryagin's minimum principle can also be identified directly without further modification of the method.

8.2 Outlook

Although this thesis presents a self-contained theory, some questions remain open for further scientific research. In a future work considering even more complex problems, the convergence can be further accelerated by improving the step size of the time/control update in Eqs. (6.12) and (7.18). A suitable choice of step size values is suggested by Halkin [33] and Gottlieb [29] already in the sixties of the past century. In the latter mentioned work, the variation of the controls is transformed into a variation of the optimality condition. Instead of minimizing the cost functional, the Hamiltonian is considered. Unfortunately, both methods presented are restricted to fixed end-time problems, and moreover, the choice of the step size is very example-specific, so that a generally applicable algorithm is difficult to abstract.

To cite a further example for ongoing scientific research, a discrete scheme of the adjoint method in [41] have to be mentioned. The ideas presented, are easy to implement and can be applied to the theory in this thesis with reasonable effort.

In many examples in this work the whole time history is identified considering each time step as parameter, which is a scientifically handsome approach and show also the robustness of the method. However, the convergence rate suffers from a high number of optimization variables. One approach for parameterization is the presented switching time optimization, which reduces the dimension of the optimization problem to a small number. However, this method only works for pure bang-bang controls and it is necessary to know the number of switching cycles. Alternatively, a more general way to reduce the dimension space of the optimization task is to introduce spline functions and identify the grid points instead.

In conclusion, the presented method is well suited for the computation of time-optimal controls of

TU WIEN Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

multibody systems. This thesis presents different methods for this purpose, and describes advantages and disadvantages, as well as the implementation of the adjoint, adjoint-influence equations and the update formulas for the gradient-based optimization of multibody systems.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU Wien Bibliothek Your knowledge hub

Appendix A

Single Track Vehicle Model

In vehicle dynamics, an unambiguous assignment of the vehicle to the race track by means of Cartesian coordinates is not always possible, e. g., in the case of closely spaced road sections. Hence, in this case a transformation to curvilinear coordinates is advisable. Even in robotics a transformation to curvilinear coordinates can be advantageous if the final time is considered free and a penalty approach is used to consider obstacles. The relation between the curvilinear coordinates s and r and the Cartesian coordinates x and y is depicted in Fig. A.1 and given by

$$\begin{aligned} x(s,r) &= x_c(s) + y_c'(s)\, r \\ y(s,r) &= y_c(s) - x_c'(s)\, r, \end{aligned} \tag{A.1}$$

where $x_c(s)$ and $y_c(s)$ are the coordinates of the center line in arc length description and primes denote derivatives with respect to s. Using a Newton scheme, for example, s and r can be expressed by x and y from Eq. (A.1). In the case of Cartesian coordinates, the position of the vehicle in Fig. 5.4 can be described by the absolute coordinates x and y and the orientation angle φ. Since, for the road to wheel contact the tire forces T_1, N_1, T_2 and N_2 must be described in the body-fixed

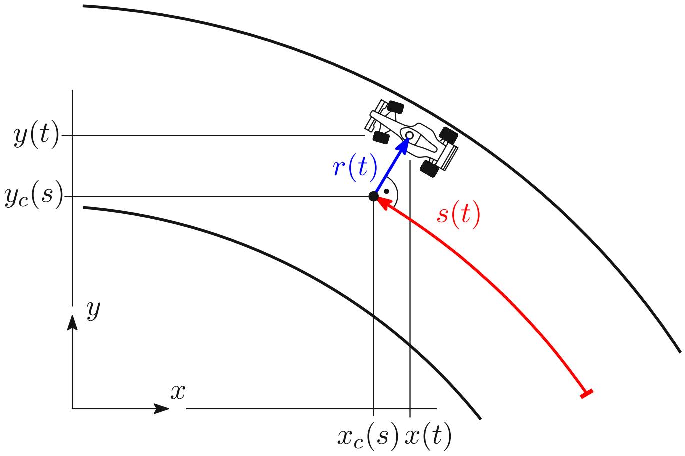

Figure A.1: Transformation from Cartesian- to curvilinear coordinates

frame of the chassis, it is convenient to formulate the equations of motion with respect to body-fixed coordinates, as proposed in [53, 55]. If $v(t)$ and $w(t)$ denote the components of the absolute velocity of the chassis with respect to the body fixed frame, the equations of motion read

$$\begin{aligned}
\dot{x}(t) &= v(t)\cos(\varphi(t)) - w(t)\sin(\varphi(t)) \\
\dot{y}(t) &= v(t)\sin(\varphi(t)) + w(t)\cos(\varphi(t)) \\
\dot{\varphi}(t) &= \Omega(t) \\
\dot{v}(t) &= \frac{1}{m}\Big(mw(t)\Omega(t) + T_1\cos(u_1(t)) + T_2 - N_1\sin(u_1(t))\Big) \\
\dot{w}(t) &= \frac{1}{m}\Big(-mv(t)\Omega(t) + N_1\cos(u_1(t)) + N_2 + T_1\sin(u_1(t))\Big) \\
\dot{\omega}_\varphi(t) &= \frac{1}{J}\Big(N_1 a\cos(u_1(t)) - N_2 b + T_1 a\sin(u_1(t))\Big) \\
\dot{\omega}_1(t) &= -\frac{1}{J_1}T_1\,R \\
\dot{\omega}_2(t) &= \frac{1}{J_2}\Big(u_2(t) - T_2\,R\Big),
\end{aligned} \tag{A.2}$$

where Ω is the angular velocity $\dot{\varphi}$ of the chassis about the vertical and ω_1 and ω_2 denote the angular velocities of the wheels. Moreover, m is the total mass of the vehicle and J denotes its moment of inertia. Both wheels have the radius R and the moments of inertia J_1 and J_2 about their rotation axis. The lengths a and b are the distances measured from the center of gravity to the rear axle and to the front axle. The steering angle is given by $u_1(t)$ and the driving/braking torque is denoted with $u_2(t)$. The arc length s is given by Eq. (A.1) from which it can be expressed by the Cartesian coordinates x, y. The transformation of the time derivative of the state vector is given by

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}s}\frac{\mathrm{d}s}{\mathrm{d}t} = \boldsymbol{x}'(s)\frac{1}{g}, \tag{A.3}$$

in which $g = 1/\dot{s}$. The velocity $\dot{s}$ is obtained from differentiating Eq. (A.1) with respect to time:

$$\begin{aligned}
\dot{x} &= x_c'(s)\;\dot{s} + y_c''(s)\;\dot{s}\;r + y_c'(s)\;\dot{r} \\
\dot{y} &= y_c'(s)\;\dot{s} - x_c''(s)\;\dot{s}\;r - x_c'(s)\;\dot{r}.
\end{aligned} \tag{A.4}$$

Solving for $\dot{s}$ and $\dot{r}$ and using ${x_c'}^2 + {y_c'}^2 = 1$ yields

$$\begin{aligned}
\dot{s} &= -\frac{1}{r\,k(s) - 1}\left(x_c'(s)\dot{x} + y_c'(s)\dot{y}\right) \\
\dot{r} &= -\frac{1}{r\,k(s) - 1}\Big[\left(r x_c''(s) - y_c'(s)\right)\dot{x} + \left(x_c'(s) + r y_c''(s)\right)\dot{y}\Big],
\end{aligned} \tag{A.5}$$

where $k(s) = x_c''(s)y_c'(s) - x_c'(s)y_c''(s)$ denotes the signed curvature of the mean trajectory. After inserting the first two state equations from Eq. (A.2) for $\dot{x}$ and $\dot{y}$ in $(\text{A.5})_1$, we obtain

$$\dot{s} = \frac{1}{g} = \frac{1}{1 - r\,k(s)}\left[x_c'(s)\left(v\cos(\varphi) - w\sin(\varphi)\right) + y_c'(s)\left(v\sin(\varphi) + w\cos(\varphi)\right)\right]. \tag{A.6}$$

TU WIEN Bibliothek Your knowledge hub
Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Now we replace the description of the vehicle position by (x, y) and use the curvilinear coordinates (s, r) instead. Since s is our independent variable now, we only need one state equation for r which is provided by Eq. (A.5)$_2$ after substituting again for $\dot{x}$ and $\dot{y}$ from the original state equations. Finally, after applying Eq. (A.3) to Eq. (A.2) and replacing the first two equations by Eq. (A.5)$_2$, we end up with the following transformed state equations:

$$
\begin{aligned}
r'(s) &= \frac{g}{r(s)\,k(s) - 1}\Big[\Big(v(s)\cos(\varphi(s)) - w(s)\sin(\varphi(s))\Big)\Big(x_c''(s)r(s) - y_c'(s)\Big) \\
&\qquad + \Big(v(s)\sin(\varphi(s)) + w(s)\cos(\varphi(s))\Big)\Big(x_c'(s) + y_c''(s)r(s)\Big)\Big] \\
\varphi'(s) &= g\,\Omega(s) \\
v'(s) &= \frac{g}{m}\Big(mw(s)\Omega(s) + T_1\cos(u_1(s)) + T_2 - N_1\sin(u_1(s))\Big) \\
w'(s) &= \frac{g}{m}\Big(-mv(s)\Omega(s) + N_1\cos(u_1(s)) + N_2 + T_1\sin(u_1(s))\Big) \\
\Omega'(s) &= \frac{g}{J}\Big(N_1 a\cos(u_1(s)) - N_2 b + T_1 a\sin(u_1(s))\Big) \\
\omega_1'(s) &= -\frac{R\,g}{J_1}T_1 \\
\omega_2'(s) &= \frac{g}{J_2}\Big(u_2(s) - T_2 R\Big),
\end{aligned}
\tag{A.7}
$$

where primes denote derivatives with respect to s.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Bibliography

[1] U. M. Ascher, R. M. M. Mattheij, and R. D. Russell. *Numerical solution of boundary value problems for ordinary differential equations*. SIAM, 1995.

[2] M. Athans and P. L. Falb. *Optimal Control – An Introduction to the Theory and Its Applications*. Dover Publications, Inc., Mineola, New York, 2007.

[3] J. Baumgarte. Stabilization of constraints and integrals of motion in dynamical systems. *Computer methods in applied mechanics and engineering*, 1(1):1–16, 1972.

[4] R. Bellman. *Dynamic Programming*. Dover Books on Computer Science. Dover Publications, 2013.

[5] F. V. Bennett. Apollo lunar ascent and descent trajectories. Technical report, National Aeronautics and Space Administration, Houston, Texas, March 1970. See also URL: `www.hq.nasa.gov/alsj/nasa58040.pdf`.

[6] E. Bertolazzi, F. Biral, and M. Da Lio. Symbolic-Numeric Indirect Method for Solving Optimal Control Problems for Large Multibody Systems. *Multibody System Dynamics*, 13(2):233–252, 2005.

[7] D. P. Bertsekas. *Dynamic Programming and Optimal Control: Volume I*. Athena scientific Belmont, MA, 1995.

[8] D. Bestle and P. Eberhard. Analyzing and Optimizing Multibody Systems. *Mechanics of structures and machines*, 20(1):67–92, 1992.

[9] J. T. Betts. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. SIAM, Philadelphia, PA, 2010.

[10] M. A. Bhatti. *Practical Optimization Methods: With* MATHEMATICA® *Applications*. Springer Science & Business Media, 2012.

[11] N. Dal Bianco, E. Bertolazzi, F. Biral, and M. Massaro. Comparison of direct and indirect methods for minimum lap time optimal control problems. *Vehicle System Dynamics*, 57(5):665–696, 2019.

[12] J. V. Breakwell. The Optimization of Trajectories. *Journal of the Society for Industrial and Applied Mathematics*, 7(2):215–247, 1959.

[13] A. E. Bryson. Optimal Control – 1950 to 1985. *IEEE Control Systems Magazine*, 16(3):26–33, 1996.

[14] A. E. Bryson and W. F. Denham. A Steepest Ascent Method for Solving Optimum Programming Problems. *ASME Jour. Appl. Mech*, 29, 1962.

[15] A. E. Bryson and W. F. Denham. Optimal Programming Problems with Inequality Constraint II: Solution by Steepest Ascent. *AIAA Journal*, 2(1):23–34, 1964.

[16] A. E. Bryson and Y. C. Ho. *Applied Optimal Control: Optimization, Estimation and Control*. Hemisphere, Washington, DC, 1975.

[17] A. E. Bryson and S. E. Ross. Optimum Rocket Trajectories with Aerodynamic Drag. *Journal of Jet Propulsion*, 28(7):465–469, 1958.

[18] R. Bulirsch. Die Mehrzielmethode zur numerischen Lösung von nichtlinearen Randwertproblemen und Aufgaben der optimalen Steuerung. *Report der Carl-Cranz-Gesellschaft*, 251, 1971.

[19] D. Casanova. *On Minimum Time Vehicle Manoeuvring: The Theoretical Optimal Lap*. PhD thesis, Cranfield University, Cranfield, UK, 2000.

[20] V. Cossalter, M. Da Lio, R. Lot, and L. Fabbri. A General Method for the Evaluation of Vehicle Manoeuvrability with Special Emphasis on Motorcycles. *Vehicle system dynamics*, 31(2):113–135, 1999.

[21] P. Eichmeir, T. Lauß, S. Oberpeilsteiner, K. Nachbagauer, and W. Steiner. The Adjoint Method for Time-Optimal Control Problems. *Journal of Computational and Nonlinear Dynamics*, 16(2), 2020.

[22] P. Eichmeir, K. Nachbagauer, T. Lauß, K. Sherif, and W. Steiner. Time-Optimal Control of Dynamic Systems Regarding Final Constraints. *Journal of Computational and Nonlinear Dynamics*, 16(3), 2021.

[23] P. Eichmeir, K. Nachbagauer, and W. Steiner. The Adjoint Gradient Method for Time-Optimal Control of a Moon Landing – Ascent, Descent, and Abort. *Journal of Computational and Nonlinear Dynamics*, Volume 2: 16th International Conference on Multibody Systems, Nonlinear Dynamics, and Control (MSNDC), 2020.

[24] R. Frezza, A. Beghi, and G. Notarstefano. Almost Kinematic Reducibility of a Car Model with Small Lateral Slip Angle for Control Design. In *Proceedings of the IEEE International Symposium on Industrial Electronics*, volume 1, pages 343–348, Dubrovnik, Croatia, 2005.

[25] C. W. Gear, G. K. Gupta, and B. Leimkuhler. Automatic integration of Euler-Lagrange equations with constraints. *Journal of Computational and Applied Mathematics*, 12:77–90, 1985.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

[26] R. H. Goddard. A method of reaching extreme altitudes (with 10 plates). *Smithsonian Miscellaneous Collections*, 1919.

[27] H. Goldstein. *Klassische Mechanik*. Aula Verlag, Wiesbaden, 1991.

[28] H. H. Goldstine. *A History of the Calculus of Variations from the 17th through the 19th Century*, volume 5. Springer Science & Business Media, 2012.

[29] R. G. Gottlieb. Rapid Convergence to Optimum Solutions Using a Min-H Strategy. *AIAA Journal*, 5(2):322–329, 1967.

[30] K. Graichen and N. Petit. Constructive Methods for Initialization and Handling Mixed State-Input Constraints in Optimal Control. *Journal of Guidance, Control, and Dynamics*, 31(5):1334–1343, 2008.

[31] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I, Nonstiff Problems*. Springer, 1993.

[32] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II*, volume 375. Springer Berlin Heidelberg, 1996.

[33] H. Halkin. Method of convex ascent. In *Computing Methods in Optimization Problems*, pages 211–239. Elsevier, 1964.

[34] E. J. Haug, R. A. Wehage, and N. K. Mani. Design Sensitivity Analysis of Large-Scale Constrained Dynamic Mechanical Systems. *Journal of Mechanisms, Transmissions, and Automation in Design*, 106(2):156–162, 1984.

[35] C. Johnson and J. Gibson. Singular Solutions in Problems of Optimal Control. *IEEE Transactions on Automatic Control*, 8(1):4–15, 1963.

[36] H. J. Kelley. Gradient Theory of Optimal Flight Paths. *Ars Journal*, 30(10):947–954, 1960.

[37] H. J. Kelley. Method of Gradients – Optimization Techniques with Applications to Aerospace Systems. In *Mathematics in Science and Engineering*, volume 5, pages 206–254, New York, 1962. Academic Press.

[38] J. Kierzenka and L. F. Shampine. A BVP solver based on residual control and the MATLAB PSE. *ACM Transactions on Mathematical Software*, 27(3):299–316, 2001.

[39] D. E. Kirk. *Optimal Control Theory: An Introduction*. Dover Publications, Mineola, New York, 2004.

[40] C. Lanczos. *The Variational Principles of Mechanics*. Dover, Mineola, New York, 1986.

[41] T. Lauß, S. Oberpeilsteiner, W. Steiner, and K. Nachbagauer. The Discrete Adjoint Gradient Computation for Optimization Problems in Multibody Dynamics. *Journal of Computational and Nonlinear Dynamics*, 12(3):031016, 2016.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

[42] G. Leitmann. *Optimization Techniques With Applications to Aerospace Systems*, volume 5. Academic Press, New York, 1962.

[43] R. Lot and F. Biral. A Curvilinear Abscissa Approach for the Lap Time Optimization of Racing Vehicles. *IFAC Proceedings Volumes*, 47(3):7559–7565, 2014.

[44] D. G. Luenberger and Y. Ye. *Linear and Nonlinear Programming*. Springer, 2016.

[45] J. Meditch. On the Problem of Optimal Thrust Programming For a Lunar Soft Landing. *IEEE Transactions on Automatic Control*, 9(4):477–484, 1964.

[46] D. Miller. Optimal Trajectory Planning for the Apollo Moon Landing: Descent, Ascent, and Aborts. Technical report, MIT, Cambridge, MA, 2014.

[47] K. Nachbagauer, S. Oberpeilsteiner, K. Sherif, and W. Steiner. The Use of the Adjoint Method for Solving Typical Optimization Problems in Multibody Dynamics. *Journal of Computational and Nonlinear Dynamics*, 10(6), 2015.

[48] J. Nocedal and S. Wright. *Numerical optimization*. Springer Science & Business Media, New York, 2006.

[49] H. J. Oberle. *BOUNDSCO: Hinweise zur Benutzung des Mehrziel-Verfahrens für die numerische Lösung von Randwertproblemen mit Schaltbedingungen*. Hamburger Beiträge zur Angewandten Mathematik 6, Universität Hamburg, 1987.

[50] S. Oberpeilsteiner, T. Lauss, W. Steiner, and K. Nachbagauer. A frequency domain approach for parameter identification in multibody dynamics. *Multibody system dynamics*, 43(2):175–191, 2018.

[51] D. E. Okhotsimskii and T. M. Eneev. Some Variational Problems Connected with the Launching of Artificial Satellites of the Earth. *J. Brit. Interplanet. Soc*, 16(5):261–262, 1958.

[52] W. Oldfather, C. A. Ellis, and D. J. M. Brown. Leonhard Euler's Elastic Curves. *Isis*, 20:72–160, 1933.

[53] B. Olofsson, K. Lundahl, K. Berntorp, and L. Nielsen. An Investigation of Optimal Vehicle Maneuvers for Different Road Conditions. *IFAC Proceedings Volumes*, 46(21):66–71, 2013.

[54] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mischchenko. The Mathematical Theory of Optimal Processes. *Wiley Interscience*, 1962.

[55] K. Popp and W. Schiehlen. *Ground Vehicle Dynamics*. Springer Science & Business Media, 2010.

[56] S. S. Rao. *Engineering Optimization*. John Wiley & Sons, New York, 2009.

[57] G. Rill and T. Schaeffer. *Grundlagen und Methodik der Mehrkörpersimulation*. Springer, 2014.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

[58] J. B. Rosen. The Gradient Projection Method for Nonlinear Programming. Part I. Linear Constraints. *Journal of the Society for Industrial and Applied Mathematics*, 8(1):181–217, 1960.

[59] R. W. H. Sargent. Optimal control. *Journal of Computational and Applied Mathematics*, 124(1-2):361–371, 2000.

[60] W. Schiehlen. Computational dynamics: theory and applications of multibody systems. *European Journal of Mechanics - A/Solids*, 25(4):566–594, 2006.

[61] R. Schwertassek and O. Wallrapp. *Dynamik flexibler Mehrkörpersysteme: Methoden der Mechanik zum rechnergestützten Entwurf und zur Analyse mechatronischer Systeme*. Springer, 2017.

[62] R. Seydel. A continuation algorithm with step control. In *Numerical methods for bifurcation problems*, pages 480–494. Springer, Basel, 1984.

[63] A. A. Shabana. *Dynamics of large scale flexible mechanical systems*. PhD thesis, University of Iowa, Iowa City, IA, 1982.

[64] A. A. Shabana. *Dynamics of Multibody Systems*. Cambridge University Press, 2013.

[65] A. Steindl, J. Edelmann, and M. Plöchl. Limit cycles at oversteer vehicle. *Nonlinear Dynamics*, 99(1):313–321, 2020.

[66] W. Steiner and S. Reichl. The Optimal Control Approach to Dynamical Inverse Problems. *Journal of Dynamic Systems, Measurement, and Control*, 134(2), 2012.

[67] W. Steiner and M. Schagerl. *Raumflugmechanik: Dynamik und Steuerung von Raumfahrzeugen*. Springer, Berlin, Heidelberg, 2004.

[68] M. A. Styblinski and T.-S. Tang. Experiments in nonconvex optimization: stochastic approximation with function smoothing and simulated annealing. *Neural Networks*, 3(4):467–483, 1990.

[69] E. Süli and D. F. Mayers. *An introduction to numerical analysis*. Cambridge University Press, 2003.

[70] H. J. Sussmann and J. C. Willems. 300 Years of Optimal Control: From The Brachystochrone to the Maximum Principle. *IEEE Control Systems Magazine*, 17(3):32–44, 1997.

[71] B. van Brunt. *The Calculus of Variations*. Springer, New York, 2004.

[72] T. M. Wasfy and A. K. Noor. Computational strategies for flexible multibody systems. *Appl. Mech. Rev.*, 56(6):553–613, 2003.

[73] S. J. Wright. Interior point methods for optimal control of discrete-time systems. *Journal of Optimization Theory and Applications*, 77(1):161–187, 1993.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub

Curriculum Vitæ

Personal Details

Name	Dipl.-Ing. Philipp Eichmeir
Day of birth	1. Dezember 1990
Place of birth	Wels, Upper Austria
Nationality	Austria
Private address	Edisonstraße 30/A8, 4600 Wels, Austria
Work address	University of Applied Sciences Upper Austria, Stelzhamerstraße 23, 4600 Wels, Austria
Phone	++43 (0) 50804 44488
Mail	philipp.eichmeir@fh-wels.at, eichmeir@gmx.at
Public links	orcid.org/0000-0001-5733-785X www.researchgate.net/profile/Philipp-Eichmeir

Education and Academic Graduation

since 01/2018	*Doctoral Programme in Technical Sciences (Dr. techn.)*, Vienna University of Technology, Institue of Mechanics and Mechatronics, Vienna, Austria.
09/2014 – 09/2016	*Master's Degree Programme (Dipl.-Ing.)*, Mechanical Engineering, University of Applied Sciences Upper Austria, Campus Wels
11/2011 – 09/2014	*Bachelor's Degree Programme (BSc.)*, Automation Engineering, University of Applied Sciences Upper Austria, Campus Wels
09/2005 – 06/2010	*Mechatronics*, Höhere technische Bundeslehranstalt, Wels
09/2001 – 07/2005	*Secondary School*, Schauergymnasium, Wels

Work Experience

since 11/2017	*Research Fellow*, University of Applied Sciences Upper Austria – Research and Development GmbH, Wels, Austria.
09/2017 – 10/2017	*Software Developer*, Engineering Center Steyr GmbH & Co KG, St. Valentin, Austria
11/2016 – 05/2017	*Engineer and Test Driver*, BMW Motoren GmbH, Steyr, Austria

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.

Academic Teaching

since 10/2020	*Calculus of Variations and Optimal Control*, University of Applied Sciences Upper Austria Campus Wels
since 10/2018	*Applied Mathematics*, University of Applied Sciences Upper Austria, Campus Wels
10/2018	*Technical Mechanics*, University of Applied Sciences Upper Austria Campus Wels

Awards

08/2020	*Receiving the Best Paper Award 2020 from the American Society of Mechanical Engineers for the paper titled "The Adjoint Gradient Method for Time-Optimal Control of a Moon Landing – Ascent, Descent, and Abort"*, ASME-IDETC, International Conference on Multibody Systems.
10/2015	*Honoring of the Bachelor thesis with the 3rd place in the competition Best Bachelor Thesis*, Conference of the Mechatronics Platform.

Peer Reviewed Publications in Journals

P. Eichmeir, T. Lauss, S. Oberpeilsteiner, K. Nachbagauer, W. Steiner, *The Adjoint Method for Time-Optimal Control Problems*, Journal for Computational and Nonlinear Dynamics, 16(2), 021003, doi.org/10.1115/1.4048808, 2021.

P. Eichmeir, T. Lauss, K. Nachbagauer, K. Sherif, W. Steiner, *Time-Optimal Control of Dynamic Systems Regarding Final Constraints*, Journal for Computational and Nonlinear Dynamics, 16(3), 031003, doi.org/10.1115/1.4049334, 2021.

P. Eichmeir, K. Nachbagauer, W. Steiner, *The Adjoint Gradient Method for Time-Optimal Control of a Moon Landing – Ascent, Descent, and Abort*, ASME International Conference on Multibody Systems, Nonlinear Dynamics, and Control, doi.org/10.1115/DETC2020-22034, 2020.

Conference Contributions and Presentations

P. Eichmeir, W. Steiner, *An Efficient Algorithm for Solving Time Optimal Control Problems in Multibody Dynamics*, International Conference on Multibody System Dynamics, Budapest, Hungary, 2021.

P. Eichmeir, K. Nachbagauer, W. Steiner, *A Gradient-Based Computation of Time Optimal Bang-Bang Controls*, International Conference on Multibody System Dynamics, Budapest, Hungary, 2021.

D. Lichtenecker, **P. Eichmeir**, K. Nachbagauer, *On the Role of Adjoint Gradients in Time-Optimal Control Problems under Final Constraints*, International Conference on Multibody System Dynamics, Budapest, Hungary, 2021.

P. Eichmeir, K. Nachbagauer, W. Steiner, *The Adjoint Gradient Method for Time-Optimal Control of a Moon Landing – Ascent, Descent, and Abort*, ASME International Conference on Multibody Systems, Nonlinear Dynamics, and Control, St. Louis, Missouri, USA, 2020.

P. Eichmeir, *Invited Lecture: Time Optimal Trajectory Planning Using the Adjoint Method*, Workshop on Analysis and Control of Multibody Systems, Hamburg University of Technology, Hamburg, Germany, 2020.

P. Eichmeir, W. Steiner, T. Lauss, S. Oberpeilsteiner, *Time Optimal Control of a Single Track Vehicle Using a Pacejka Tire Model*, ECCOMAS Thematic Conference on Multibody Dynamics, Duisburg, Germany, 2019.

P. Eichmeir, S. Oberpeilsteiner, T. Lauss, W. Steiner, *On time-optimal control in multibody dynamics*, International Congress on Industrial and Applied Mathematics, Valencia, Spain, 2019.

P. Eichmeir, S. Oberpeilsteiner, T. Lauss, W. Steiner, *Time-optimal control of a vehicle on a race track using a Pacejka tire model*, GAMM Annual Meeting of the International Association of Applied Mathematics and Mechanics, Vienna, Austria, 2019.

S. Oberpeilsteiner, T. Lauss, **P. Eichmeir**, W. Steiner, *FreeDyn – A Software for Advanced Multibody Simulation: Applications in the Field of Vehicle Dynamics*, ECCOMAS Thematic Conference on Multibody Dynamics, Duisburg, Germany, 2019.

S. Oberpeilsteiner, T. Lauss, **P. Eichmeir**, W. Steiner, *A new gradient based method for solving time optimal control problems in multibody dynamics*, International Conference on Multibody System Dynamics, Lisbon, Portugal, 2018.

Research Projects

Josef Ressel Centre for Advanced Multibody Dynamics: *Engaged in Research on the Development of Algorithms for Determining the Time-Optimal Control of a Racing Car.* Christian Doppler Gesellschaft, KTM, FH-Wels (2017–2019)

Industrial Research Project: *Engaged in Research on the Development of an Elastohydrodynamic Lubricating Film Model in Radial Plain Bearings with* Abaqus. BRP-Rotax, FH-Wels (2021)

Die approbierte gedruckte Originalversion dieser Dissertation ist an der TU Wien Bibliothek verfügbar.
The approved original version of this doctoral thesis is available in print at TU Wien Bibliothek.
TU WIEN Bibliothek Your knowledge hub