D I S S E R T A T I O N

submitted

to the

Combined Faculty for the Natural Sciences and Mathematics

of

Heidelberg University, Germany

for the degree of

Doctor of Natural Sciences

put forward by

Diplom-Mathematiker Diplom-Physiker

Felix Johannes Maximilian Lenders

born in Ulm

Date of the oral examination

February 07, 2018

Numerical Methods for Mixed-Integer Optimal Control with Combinatorial Constraints

Referees

Professor Dr. Christian Kirches

Professor Dr. Dr. h.c. mult. Hans Georg Bock

Zusammenfassung

Diese Dissertation befasst sich mit numerischen Methoden für gemischt-ganzzahlige Optimalsteuerungsprobleme mit kombinatorischen Nebenbedingungen. Es wird ein Approximationssatz bewiesen, der ein gemischt-ganzzahliges Optimalsteuerungsproblem mit kombinatorischen Nebenbedingungen in Beziehung zu einen kontinuierlichen Optimalsteuerungsproblem mit sogenannten verschwindenden Nebenbedingungen setzt und das Fundament für numerische Rechnungen bildet. Ein Rundungsalgorithmus, der auf dieser Korrespondenz aufbaut und die verschwindenden Nebenbedingungen beachtet, wird entwickelt.

Direkte Diskretisierungen von Optimalsteuerungsproblemen mit verschwindenden Nebenbedingungen sind Beispiele von Mathematischen Programmen mit Komplementaritätsnebenbedingungen. Diese bilden eine anspruchsvolle Klasse von Problemen aufgrund ihrer inhärenten Nicht-Konvexität und fehlenden Regularität. Ein Active-Set Algorithmus für Mathematische Programme mit Komplementaritätsnebenbedingungen wird entwickelt und es wird bewiesen, dass dieser Algorithmus global konvergent zu Bouligand-stationären Punkten ist, sofern gewisse technische Voraussetzungen erfüllt sind.

Zur effizienten Berechnung newtonartiger Schritte bei Optimalsteuerungsproblemen wird die verallgemeinerte Lanczos-Methode für Trust-Region Probleme in Hilberträumen entwickelt. Um Echtzeitanforderungen in Online Optimalsteuerungskontexten gewährleisten zu können wird ein Gauß-Newton Vorkonditionierer für die iterative Lösung des Trust-Region Problems erarbeitet.

Die vorgestellten Methoden werden implementiert und ihre Anwendbarkeit und Effektivität wird an Hand von Benchmark-Problemen unter Beweis gestellt.

Abstract

This thesis is concerned with numerical methods for Mixed-Integer Optimal Control Problems with Combinatorial Constraints. We establish an approximation theorem relating a Mixed-Integer Optimal Control Problem with Combinatorial Constraints to a continuous relaxed convexified Optimal Control Problems with Vanishing Constraints that provides the basis for numerical computations. We develop a a Vanishing-Constraint respecting rounding algorithm to exploit this correspondence computationally.

Direct Discretization of the Optimal Control Problem with Vanishing Constraints yield a subclass of Mathematical Programs with Equilibrium Constraints. Mathematical Programs with Equilibrium Constraint constitute a class of challenging problems due to their inherent non-convexity and non-smoothness. We develop an active-set algorithm for Mathematical Programs with Equilibrium Constraints and prove global convergence to Bouligand stationary points of this algorithm under suitable technical conditions.

For efficient computation of Newton-type steps of Optimal Control Problems, we establish the Generalized Lanczos Method for trust region problems in a Hilbert space context. To ensure real-time feasibility in Online Optimal Control Applications with tracking-type Lagrangian objective, we develop a Gauß-Newton preconditioner for the iterative solution method of the trust region problem.

We implement the proposed methods and demonstrate their applicability and efficacy on several benchmark problems.

Acknowledgments

I would like to express gratitude to my advisors Professor Dr. Christian Kirches and Professor Dr. Dr. h.c. mult. Hans Georg Bock. Their constant support and trust in my capabilities and the encouraging and open-minded spirit maintained in their research groups made the work on this thesis a pleasure.

Furthermore, I would like to thank Dr. Johannes Schlöder and Dr. Andreas Potschka for their support, advice and the many interesting discussions in the past years.

It was a pleasure to work in the stimulating academic environment established at the Interdisciplinary Center for Scientific Computing and the Faculty of Mathematics and Computer Science at Heidelberg University. In particular, I would like to thank the members of the Simulation and Optimization, the Model-Based Optimizing Control and Experimental Design Group for contributing to this unique atmosphere. For the many cups of coffee we shared together and the interesting discussions, I would like to thank Anja Bettendorf, Lilli Bergner, Dr. Holger Diedam, Dr. Kathrin Hatz, María Elena Suárez Garcés, Jürgen Gutekunst, Dr. Christian Hoffmann, Dr. Dennis Janka, Johannes Herold, Dr. Robert Kircheis, Dr. Tom Kraus, Dr. Huu Chuong La, Conrad Leidereiter, Dr. Simon Lenz, Enrique Guerrero Merino, Andreas Meyer, Nadia Said, Dr. Andreas Schmidt, Dr. Andreas Sommer and Leonard Wirsching.

I am grateful to Professor Dr. Sebastian Sager for his support and for inviting me to the retreat of his Mathematical Algorithmic Optimization group at Otto von Guericke University Magdeburg.

I would like to thank Matthias Kümmerer, Conrad Leidereiter, Paul Manns, Marta Sauter and Dr. Christina Schindler for commenting on parts of the manuscript of this thesis.

For helping me through the jungle of administration, I would like to thank Anastasia Valter, Silke Thiel, Abir Al-Laham, Margret Rothfuß, Ria Hillenbrand-Lynott, Anja Vogel and Dorothea Heukäufer and Thomas Klöpfer, Dr. Hermann Lauer and Martin Neisen for maintaining the computing facilities.

I gratefully acknowledge funding by the Heidelberg Graduate School of Mathematical and Computational Methods for the Sciences and the German National Academic Foundation and support within the project GOSSIP by the German Federal Ministry of Education and Research, grant n^{o} 05M2013-GOSSIP.

Contents

1. Introduction

Ordinary differential equations constitute a fundamental tool for quantitatively describing dynamic processes. By introducing an explicit external control variable, this naturally extends to the description of systems that can be influenced.

Finding a suitable control to achieve a certain goal and in particular to minimize a selected cost function is an important problem. Many questions in engineering, natural sciences, economics and even humanities can be cast into this framework.

In this thesis, we focus on numerical methods for the challenging class of *Mixed-Integer Optimal Control Problems*. In Mixed-Integer Optimal Control Problems, some of the control influences are constrained to map into a *finite set*. Mixed-Integer Optimal Control Problems generalize *Optimal Control Problems* and *Integer Programming Problems*. The latter are static optimization problems with integrality requirements on the variables. Analyzing Mixed-Integer Optimal Control Problems is challenging due to their combinatorial, nonlinear and dynamic nature. An intuitive understanding of such systems is hard to obtain and thus applications of such systems have high potential for optimization. Mixed-Integer Optimal Control Problems have gained increasing attention in the past fifteen years with the emergence of practical solution methods. Recent approaches are based on convexification, relaxation and suitable rounding to compute suboptimal solutions with arbitrary small optimality loss and have been applied successfully to real-time control.

The techniques are limited to the situation in which no constraints, with the exception of the integrality constraint, are imposed on integer control variables. Generalizing this setting, we focus on the class of *Mixed-Integer Optimal Control Problems with Combinatorial Constraints*, where by combinatorial constraints we understand mixed state-control constraints that depend on integer control variables.

Direct discretizations of relaxed convexifications of Mixed-Integer Optimal Control Problems with Combinatorial Constraints are optimization problems with a special non-smooth structure. They exhibit *Vanishing Constraint* behavior, which reflects the combinatorial origin of the problem. *Mathematical Programming Problems with Vanishing Constraints* have non-convex feasible set and violate standard regularity assumptions. Therefore tailored optimization algorithms addressing the lack of smoothness of the problem must be used. We develop such an algorithm based on the SLEQP method as part of the thesis.

1.1. Contributions and results of this thesis

In this thesis, we develop an efficient numerical method for Mixed-Integer Optimal Control Problems with Combinatorial Constraints. We obtain novel results and advances over established techniques in various parts of Optimal Control and Nonlinear Programming. They are described in the following.

Convexification and Relaxation for Mixed-Integer Optimal Control Problems with Combinatorial Constraints

Sager [Sag06] proposes and analyzes a convexification and relaxation approach towards Mixed-Integer Optimal Control Problems without Combinatorial Constraints. Kirches and Jung [Kir10; Jun13] extend Sager's partial outer convexification technique towards Mixed-Integer Optimal Control Problems with Combinatorial Constraints, but do not analyze approximation properties nor guarantee feasibility. We establish a novel analysis of partial outer convexification and relaxation of Mixed-Integer Optimal Control Problems with Combinatorial Constraints. As main contribution, we prove Theorem 6.7 that asserts that every feasible point of the relaxed convexified problem can be approximated arbitrarily well by an integer feasible point. A corollary to this is that a suboptimal solution to a Mixed-Integer Optimal Control Problem with *arbitrarily small* feasibility and optimality loss can be obtained by solving the relaxed convexified problem, which is a *continuous* optimal control problem with function space vanishing constraint.

Rounding Scheme

Applying the rounding schemes of Sager [Sag06] or Jung [Jun13] for reconstruction of integer feasible points can lead to severe violation of Combinatorial Constraints. We introduce a novel class of rounding schemes, *Vanishing Constraint convergent rounding schemes* and prove that these are applicable for integer reconstruction in the sense of the Approximation Theorem and respect Combinatorial Constraints. We propose a novel rounding scheme VC-SOS-SUR which strictly respects Combinatorial Constraints and guarantees ε-feasibility. VC-SOS-SUR exhibits linear computational complexity in the temporal discretization grid and thus provides an efficient mean for practical computations.

Sequential LPEC method for Equilibrium Constrained Problems with Global Convergence to B-stationary Points

Discretizations of relaxed convexified Mixed-Integer Optimal Control Problems are Mathematical Programs with Vanishing Constraints that require tailored optimization algorithms to address the non-smooth and non-convex structure of the problems. We propose a novel class of algorithms to solve composite non-smooth optimization problems that comprise Mathematical Programs with Vanishing Constraints and Mathematical Programs with Equilibrium Constraints. The algorithm class builds upon the SLEQP method for Nonlinear Programming Problems without Equilibrium Constraints. With Theorem 7.8, we establish a global convergence result for our method and show that convergence to Bouligand stationary points is ensured. This distinguishes our contribution from the methods considered in the literature, which may only converge to points satisfying weaker stationary conditions that do not preclude the existence of first-order descent directions.

Practical Sequential LPEC method with EQP Acceleration

The algorithmic framework we present for theoretical analysis described a generic class of algorithms and we obtain a convergence result that covers a broad range of possible realizations. We specify a particular realization extending the SLEQP implementation of Waltz and Nocedal [Byr+03] with an additional equality constrained quadratic programming phase to obtain a Newton-type step that promotes fast local convergence. A trust region globalization is used to allow using exact Hessians and indefinite Hessian approximations. An iterative method built upon Krylov subspace techniques is used to solve the trust region subproblem addressing the situation in which evaluations of the Hessian matrix are expensive, but matrix vector products with the Hessian can be computed with reasonable effort as it is typically the case for discretizations of optimal control problems.

Trust Region Problems in Hilbert Space

Trust region problems constitute an important subproblem in optimization and are also an important building block in our Sequential LPEC EQP method. We show existence of solutions to Hilbert space trust region problems under suitable compactness assumptions on the negative part of the defining quadratic form and generalize Gould's Generalized Lanczos Method for trust region problems [Gou+99] to a Hilbert space setting. The Hilbert space setting allows direct application of the method to problems formulated in function space and covers in particular applications within PDE

constrained optimal control. We develop a novel heuristic to address ill-conditioning and establish hot-starting results upon trust-region radius change.

Gauß-Newton Preconditioner for Model Predictive Control

We propose a novel preconditioner for applications in Nonlinear Model Predictive Control with least-squares tracking objective based on the Gauß-Newton Approximation to the Hessian. This constitutes an important ingredient in a real-time feasible algorithm based on iterative methods for online optimal control and can be incorporated into our SLPECEQP method that allows for preconditioning. We are able to significantly decrease the number of matrix vector evaluations with the Hessian that constitute the dominant computational expense.

Implementations

We have implemented all developed algorithms in the software packages `trlib`, `SLPECEQP` and `OptimIND`. The algorithm for trust region problems is implemented in the C11 software package `trlib` and features a vector free reverse communication interface that only makes use of the Hilbert space structure of the problem without any assumption on a possible discretization. Our implementation `trlib` is now included as core optimization solver in the scientific computation environment `SciPy`.

With `SLPECEQP` we have developed a hybrid Python, C and Fortran implementation of the sequential LPEC EQP method. With `OptimIND` we have established a multiple shooting discretization of optimal control problems in Python and C++, relying on Internal Numerical Differentiation and Automatic Differentiation for consistent and efficient derivative generation.

We compare `trlib` and `SLPECEQP` with state-of-the-art solvers for trust region problems and for nonlinear programming respectively using the benchmark collections `CUTEr` and `CUTEst` and find performance competitive to state-of-the-art solvers for the respective problem classes.

Case Studies

We analyze an example of Cesari to study the proposition of the Approximation Theorem for Mixed-Integer Optimal Control and the VC-SOS-SUR rounding scheme by means of an example.

We further demonstrate the efficacy of the methods and implementations for an optimal control problem describing the re-entry of an Apollo type vehicle into earth's atmosphere, for variants of a Mixed-Integer Optimal Control Problem by Egerstedt with and without Combinatorial Constraints and for applications in online optimal

control studying a nonlinear batch reactor and a continuously stirred tank reactor. As comparison for the online optimal control applications, we compute reference solutions to the offline problems using Pontryagin's Maximum Principle.

Publications

During the work on this thesis, we contributed the following publications:

[Kir+15] C. Kirches, M. Jung, F. Lenders, and S. Sager. "Approximation properties of complementarity problems from mixed-integer optimal control." In: *Mixed-integer Nonlinear Optimization: A Hatchery for Modern Mathematics*. Ed. by L. Liberti, S. Sager, and A. Wiegele. Vol. 12. Oberwolfach Reports 4. 2015, pp. 2736–2737. URL: https://www.mfo.de/document/1543/OWR_2015_46.pdf.

[KL16] C. Kirches and F. Lenders. "Approximation Properties and Tight Bounds for Constrained Mixed-Integer Optimal Control." In: *Optimization Online* (Apr. 2016). (submitted to Mathematical Programming). URL: http://www.optimization-online.org/DB_HTML/2016/04/5404.html.

[LKB17] F. Lenders, C. Kirches, and H. G. Bock. "pySLEQP: A Sequential Linear Quadratic Programming Method Implemented in Python." In: *Modeling, Simulation and Optimization of Complex Processes*. Ed. by H. G. Bock, H. X. Phu, R. Rannacher, and J. P. Schlöder. Springer Verlag, 2017, pp. 103–113. DOI: 10.1007/978-3-319-67168-0_9.

[LKP16] F. Lenders, C. Kirches, and A. Potschka. "trlib: A vector-free implementation of the GLTR method for iterative solution of the trust region problem." In: *Optimization Online* (Nov. 2016). (submitted to Optimization Methods and Software). URL: http://www.optimization-online.org/DB_HTML/2016/11/5724.html.

Parts of Chapter 6 are based on [KL16] and [Kir+15], parts of Chapter 15 are based on [Kir+15]. Parts of Chapter 7 and Chapter 12 are based on [LKB17]. Chapter 8 and Chapter 11 are based on [LKP16].

1.2. Thesis outline

This thesis is divided into three parts. The first part introduces the necessary background and surveys the literature, the second part develops our theoretical and algorithmic contributions and the final third part presents the implementations and numerical results.

Part I starts with Chapter 2 that introduces the notational conventions used in the thesis and recalls the definitions of the necessary function spaces of Lebesgue, Bochner and Sobolev type. Chapter 3 gives an overview on numerical methods for Optimal Control Problems and Mixed-Integer Optimal Control Problems and in particular describes the Direct Multiple Shooting Discretization of Optimal Control Problems. It is followed by Chapter 4 that provides material from Nonlinear Programming required for analyzing Mathematical Programs with Vanishing and Equilibrium Constraints and our sequential LPEC EQP algorithm. Necessary conditions, Constraint Qualifications and the relation to multiplier uniqueness and penalization are discussed. In Chapter 5 we introduce the classes of Mathematical Programs with Vanishing and Equilibrium Constraints, review stationarity concepts and give a literature overview on solution methods and applications.

Part II begins with the analysis of Mixed-Integer Optimal Control Problems with Combinatorial Constraints in Chapter 6. We introduce the notion of partial outer convexification and relaxation and establish a novel approximation result between a relaxed convexified problem and a Mixed-Integer Optimal Control Problem. Vanishing Constraint-convergent rounding schemes as effective reconstruction algorithms are proposed. We close the chapter with a consideration of the results by means of an example of Cesari.

In the following Chapter 7, we develop a sequential LPEC algorithm for a composite non-smooth optimization problem with equilibrium constraints that covers the case of Mathematical Programs with Equilibrium Constraints. We show that stationary points of the composite non-smooth optimization problem are exactly Bouligand stationary points for Mathematical Programs with Equilibrium Constraints. We establish global convergence of the algorithm and give a practical variant of the algorithm that promotes fast local convergence with a Newton-type EQP step.

Chapter 8 considers trust region problems in Hilbert space. We show existence of solutions under suitable compactness assumptions and necessary conditions are derived. We develop a generalization of Gould's Generalized Lanczos Method to Hilbert spaces. We close the chapter by presenting a heuristic addressing ill-conditioned problems.

The final Chapter 9 of the second part proposes a Gauß-Newton preconditioner for Nonlinear Model Predictive Control to approach real-time feasibility by quickly computing solutions to the trust region subproblem.

In *Part III*, we first review in Chapter 10 performance profiles as a method to analyze the performance of different solvers on a set of benchmark problems and introduce the `CUTEr` and `CUTEst` benchmark collections for nonlinear programming. In Chapter 11, we introduce our vector free implementation `trlib` of the Generalized Lanczos Method, assess its performance on the `CUTEst` benchmark collection by comparing

it with state-of-the-art iterative solvers and solve a PDE constrained trust region problem to demonstrate the applicability to Hilbert space problems. We continue in Chapter 12 by introducing our SLPECEQP implementation and by analyzing the performance of our implementation on the CUTEr benchmark collection, comparing it with state-of-the-art active set solvers for nonlinear programming. Chapter 13 introduces our implementation OptimIND of the direct multiple shooting discretization for optimal control problems building on Internal Numerical Differentiation and Automatic Differentiation. The last three chapters of the part are case studies for Optimal Control, Mixed-Integer Optimal Control and Online Optimal Control. Re-entry of an Apollo type space shuttle is considered in Chapter 14 and the optimal control problem used as an example to compare OptimIND with SLPECEQP with MUSCOD-II. In Chapter 15, we consider an example of Egerstedt as a case study for Mixed-Integer Optimal Control Problems and consider variants of the problem with and without Combinatorial Constraints. We analyze the behavior of our convexification, relaxation and rounding approach with OptimIND and SLPECEQP with the smoothing approach of Hoheisel together with the interior point algorithm IpOpt and with applying the Mixed-Integer Nonlinear Programming Solver Bonmin. Chapter 16 considers the applicability of a real-time optimal control scheme based on the SLPECEQP algorithm on a nonlinear batch reactor and a continuously stirred tank reactor. Reference solutions to the offline problems are computed by solving the necessary conditions of Pontryagin's Maximum Principle and the effectiveness of the Gauß-Newton preconditioner is studied.

We conclude the thesis with an outlook in Chapter 17.

Computational Environment

The computational results presented in Chapter 12 have been obtained on an Ubuntu Linux 14.04 system powered by an Intel Core i7-920 CPU with 24 GB of main memory. Results presented in Chapters 11, 14, 15 and 16 have been obtained on a Ubuntu Linux 16.04 system powered by an Intel Core i7-6800K CPU with 32 GB of main memory.

Part I.

Background

2. Notational Conventions and Function Spaces

In this chapter, we declare the notational conventions and introduce the function spaces that are used in this thesis.

2.1. Notation

Sets, Operations involving Natural Numbers and Logical Propositions

We denote by $\mathbb{N} = \{0, 1, \ldots\}$ the natural numbers and by $\mathbb{R}$ the real numbers. For a natural number n we denote by $[n]$ the set $[n] := \{0, \ldots, n-1\}$.

The Iverson bracket $[\cdot]$ is used for propositions as generalization of the Kronecker δ and is defined for a logical proposition P by

$$[P] := \begin{cases} 1, & P \text{ true}, \\ 0, & P \text{ false}. \end{cases}$$

Vectors in $\mathbb{R}^n$, Matrices and Indexing

We use lowercase latin letters to denote vectors $a, b, x, y, z \in \mathbb{R}^n$. In chapters 8 and 11 we used boldface letters for coordinate vectors $\boldsymbol{x} \in \mathbb{R}^n$ representing a discretization of a vector x in a function space. The i-th component of the vector x resp. $\boldsymbol{x}$ is denoted by x_i resp. $\boldsymbol{x}_i$. For an index set $\mathcal{I} \subseteq [n]$, we define $e_{\mathcal{I}} \in \mathbb{R}^n$ resp. $\boldsymbol{e}_{\mathcal{I}}$ by $e_{\mathcal{I}} := ([i \in \mathcal{I}])_{i \in [n]}$ and for $i \in [n]$ we define $e_i := e_{\{i\}}$, resp. $\boldsymbol{e}_i$ to be the i-th unit vector in the standard basis of $\mathbb{R}^n$. In particular, $x = \sum_{i \in [n]} x_i e_i$ and $e_{[n]}$ the vector having all entries 1.

Matrices are denoted by uppercase latin letters $A, B, C \in \mathbb{R}^{m \times n}$. a_{ij} denotes the entry in the i-th row, j-th column of A. A^T denotes the transpose of A. The identity is denoted by I.

We use subscripts to index sequences of objects, $(x^n)_{n \in \mathbb{N}}$.

Vector Spaces, Subsets of Vector Spaces

If $(X, \|\cdot\|)$ is a normed vector space, we denote by X' its *topological* dual space consisting of *continuous* linear functionals $\varphi : X \to \mathbb{R}$ and equip it with the operator norm $\|\varphi\| := \sup_{\|x\|\leq 1} |\varphi(x)|$. If $(X, \|\cdot\|_X)$ and $(Y, \|\cdot\|_Y)$ are normed vector spaces, we denote by $(\mathcal{L}(X,Y), \|\cdot\|_{\mathcal{L}(X,Y)})$ the space of *continuous* linear mappings $T : X \to Y$ with the norm $\|T\|_{\mathcal{L}(X,Y)} := \sup_{\|x\|_X \leq 1} \|T(x)\|_Y$.

For a subset $S \subseteq X$ of a topological vector space X, we denote the interior of S by $\operatorname{int} S$, the closure of S by $\overline{S}$, the convex hull of S by $\operatorname{co} S$ and the closed convex hull by $\overline{\operatorname{co}}\, S := \overline{\operatorname{co} S}$.

For a measurable set $\Omega \subseteq \mathbb{R}^n$, a vector space X, a subset $M \subseteq X$ and $A(\Omega, X)$ a subspace of the space of mappings $\Omega \to Y$ we define by slight abuse of notation the set $A(\Omega, M)$ as the set of functions $f \in A(\Omega, X)$ such that $f(x) \in M$ for almost all $x \in \Omega$.

Derivatives

For a function $f : D \subseteq X \to Y$ between to normed spaces X and Y, that is Fréchet-differentiable in $\xi \in D$, we denote its derivative in ξ by $\frac{\mathrm{d}f}{\mathrm{d}x}(\xi) \in \mathcal{L}(X,Y)$ or $\frac{\mathrm{d}f}{\mathrm{d}x}$ if the base point ξ is clear. If $X = \mathbb{R}$ we may write $\dot{f}$ instead of $\frac{\mathrm{d}f}{\mathrm{d}x}$. If X and Y are Hilbert spaces, we denote the adjoint of $\frac{\mathrm{d}f}{\mathrm{d}x}$ by $\nabla f \in \mathcal{L}(Y', X')$. In the case $X = \mathbb{R}^n, Y = \mathbb{R}^m$, we use $\frac{\mathrm{d}f}{\mathrm{d}x}$ as well to denote the matrix $\frac{\mathrm{d}f}{\mathrm{d}x} \in \mathbb{R}^{m\times n}$ representing $\frac{\mathrm{d}f}{\mathrm{d}x}$ with respect to the standard coordinate basis and similar $\nabla f \in \mathbb{R}^{n\times m}$ to denote the matrix representing ∇f. In particular, $\nabla f = (\frac{\mathrm{d}f}{\mathrm{d}x})^T$.

For a function $f : D \subseteq \mathbb{R}^n \to \mathbb{R}^m$ that admits partial derivatives in $\xi \in X$ we denote by $\frac{\partial f}{\partial x}(\xi) \in \mathbb{R}^{m\times n}$ the matrix consisting of all partial derivatives, $\frac{\partial f}{\partial x} = (\partial_j f_i)_{i\in[m], j\in[n]}$. If f happens to be differentiable as well in $\xi \in X$, $\frac{\mathrm{d}f}{\mathrm{d}x}(\xi) = \frac{\partial f}{\partial x}(\xi)$. If f is k-times partial differentiable in $\xi \in X$ and $\alpha \in \mathbb{N}^n$ is a multi-index with $|\alpha| := \sum_{i\in[n]} \alpha_i = k$, we denote by $\partial^\alpha f := \frac{\partial^k f}{\partial x_0^{\alpha_0} \cdots \partial x_{n-1}^{\alpha_{n-1}}}$.

2.2. Function spaces

We introduce the notion of Lebesgue, Bochner, and Sobolev spaces and of absolute continuity. We identify the Sobolev space $W^{1,p}(I, \mathbb{R}^n)$ with spaces of absolutely continuous functions.

2.2.1. Lebesgue, Bochner, and Sobolev spaces

We recall the definition Lebesgue, Bochner and Sobolev spaces and refer to the monographs by Adams and Fournier [AF03], Clarke [Cla13], Yosida [Yos78] and Wloka [Wlo71].

2.1 Definition (Lebesgue spaces $L^p(\Omega)$).
For a non-empty set $\Omega \subseteq \mathbb{R}^n$ and $1 \leq p \leq \infty$ we denote by $L^p(\Omega)$ the *Lebesgue space* of all equivalence classes of measurable functions $f : \Omega \to \mathbb{R}$ such that $|f|^p$ is integrable and equip it with the norm

$$\|f\|_{L^p} := \begin{cases} \sqrt[p]{\int_\Omega |f(x)|^p \,\mathrm{d}x}, & 1 \leq p < \infty, \\ \operatorname{ess\,sup}_{x\in\Omega} |f(x)|, & p = \infty. \end{cases}$$

△

2.2 Definition (Bochner spaces $L^p(\Omega, X)$).
If $(X, \|\cdot\|)$ is a separable Banach space and $p \geq 1$, the *Bochner space* $L^p(\Omega, X)$ consists of all equivalence classes of measurable functions $f : \Omega \to X$ such that $\|f\|^p$ is integrable and is equipped with the norm

$$\|f\|_{L^p} := \begin{cases} \sqrt[p]{\int_\Omega \|f(x)\|^p \,\mathrm{d}x}, & 1 \leq p < \infty, \\ \operatorname{ess\,sup}_{x\in\Omega} \|f(x)\|, & p = \infty. \end{cases}$$

△

The Lebesgue and Bochner spaces satisfy the following properties:

2.3 Proposition.
Let $\Omega \subseteq \mathbb{R}^n$ be a measurable set, X a Banach space, $1 \leq p \leq \infty$. Then:

(1) $(L^p(\Omega, X), \|\cdot\|_{L^p})$ is a Banach space ([AF03, 2.16]).

(2) If $(X, \langle\cdot,\cdot\rangle)$ is a Hilbert space, we declare an inner product on $L^2(\Omega, X)$ by

$$(f, g)_{L^2} := \int_\Omega \langle f(x), g(x)\rangle \,\mathrm{d}x.$$

This turns $(L^2(\Omega, X), (\cdot,\cdot)_{L^2})$ into a Hilbert space ([AF03, 2.18]).

(3) If Ω is of bounded measure, $L^q(\Omega, X) \subseteq L^p(\Omega, X)$ for all $p \leq q \leq \infty$ and the canonical injection is continuous: $\|f\|_p \leq (\operatorname{vol}\Omega)^{1/p-1/q}\|f\|_q$ for all $f \in L^q(\Omega, X)$ ([AF03, 2.14]).

(4) If $1 \leq p < \infty$ and $1/p + 1/q = 1$, the mapping

$$L^q(\Omega, X') \to L^p(\Omega, X)', \quad g \mapsto \int_\Omega \langle g(x), \cdot \rangle \, \mathrm{d}x$$

defines an isometric isomorphism between the dual of $L^p(\Omega, X)$ and $L^q(\Omega, X')$ ([AF03, 2.44, 2.45]). △

2.4 Definition (Sobolev spaces $W^{k,p}$).
Let $\Omega \subseteq \mathbb{R}^n$ be an open set, X a separable Banach space, $1 \leq p \leq \infty$, $k \geq 0$.

The Sobolev space $W^{k,p}(\Omega, X)$ is defined to be the space of all functions of $L^p(\Omega, X)$ that admit all weak derivatives of order at most k:

$$W^{k,p}(\Omega, X) := \{f \in L^p(\Omega, X) \mid \partial^\alpha f \in L^p(\Omega, X) \text{ for all } |\alpha| \leq k\}.$$

It is endowed with the Sobolev-norm

$$\|f\|_{W^{k,p}} := \begin{cases} \sqrt[p]{\sum_{|\alpha| \leq k} \|\partial^\alpha f\|_{L^p}}, & 1 \leq p < \infty \\ \max_{|\alpha| \leq k} \|\partial^\alpha f\|_{L^\infty}, & p = \infty. \end{cases}$$ △

It is also possible to define the Sobolev space $W^{k,p}(\Omega, X)$ as the completion of $\{f \in C^k(\Omega, X) \mid \|f\|_{W^{k,p}}\}$ with respect to the Sobolev norm $\|\cdot\|_{W^{k,p}}$. We prefer the definition given here as it provides a less abstract description. Meyers and Serrin [MS64] proved that these definitions coincide.

2.2.2. Absolutely continuous functions

We will be concerned in particular with the space $W^{1,p}(I, \mathbb{R}^n)$ for a non-empty open interval $I \subseteq \mathbb{R}$ and will characterize it in the following using absolutely continuous functions, see [Rud66, Ch. 7].

2.5 Definition (Absolute continuity).
Let $I \subseteq \mathbb{R}$ be a non-empty interval. A measurable function $f : I \to \mathbb{R}$ is *absolutely continuous* if there exists $d \in L^1(I)$ satisfying

$$|f(s) - f(r)| \leq \int_r^s d(\tau) \, \mathrm{d}\tau \quad \text{for all } r, s \in I,\ r \leq s.$$

A measurable function $f : I \to \mathbb{R}^n$ is *absolutely continuous* if every component $f_i : I \to \mathbb{R}$ is absolutely continuous for every $i \in [n]$. △

Absolutely continuous functions satisfy the Fundamental Theorem of Calculus:

2.6 Proposition (Fundamental Theorem of Calculus).
Let $I \subseteq \mathbb{R}$ be a non-empty interval and $f : I \to \mathbb{R}^n$ be absolutely continuous and $t_0 \in I$.

Then there exists $\dot{f} \in L^1(I, \mathbb{R}^n)$ such that the Fundamental Theorem of Calculus holds:

$$f(t) = f(t_0) + \int_{t_0}^{t} \dot{f}(\tau)\,\mathrm{d}\tau \quad \text{for all } t \in I.$$

△

We thus note that for an open non-empty interval $I \subseteq \mathbb{R}$ and a function $f \in W^{1,p}(I, \mathbb{R}^n)$ there exists an absolutely continuous representative $g : I \to \mathbb{R}^n$ with $g(t) = f(t)$ almost everywhere: If $t_0 \in \operatorname{int} I$ is fixed, since $\dot{f} \in L^p(I, \mathbb{R}^n)$, the difference $f(t) - \int_{t_0}^{t} \dot{f}(\tau)\,\mathrm{d}\tau$ equals a constant c almost everywhere as its derivative vanishes almost everywhere. Setting $g(t) := c + \int_{t_0}^{t} \dot{f}(\tau)\,\mathrm{d}\tau$ yields then a desired absolutely continuous representative.

We can identify the space $W^{1,p}(I, \mathbb{R}^n)$ for an open non-empty interval $I \subseteq \mathbb{R}$ with absolutely continuous functions $f : I \to \mathbb{R}^n$ such that $\dot{f} \in L^p(I, \mathbb{R}^n)$ and use this identification to generalize the definition of $W^{1,p}(I, \mathbb{R}^n)$ for arbitrary bounded non-empty intervals $I \subseteq \mathbb{R}$.

2.7 Definition (Absolutely Continuous functions $W^{1,p}(I, \mathbb{R}^n)$).
Let $I \subseteq \mathbb{R}$ be a non-empty bounded interval and $1 \leq p \leq \infty$.

The space $W^{1,p}(I, \mathbb{R}^n)$ consists of absolutely continuous functions $f : I \to \mathbb{R}^n$ such that $\dot{f} \in L^p(I, \mathbb{R}^n)$.

It is equipped with the Sobolev norm

$$\|f\|_{W^{1,p}} := \begin{cases} \sqrt[p]{\|f\|_{L^p}^p + \|\dot{f}\|_{L^p}^p}, & 1 \leq p < \infty \\ \max\{\|f\|_{L^\infty}, \|\dot{f}\|_{L^\infty}\}, & p = \infty. \end{cases}$$

△

We have just noted that this is no clash with the previous definition of $W^{1,p}(I, \mathbb{R}^n)$ for open non-empty intervals, as the spaces can be identified.

3. Numerical Methods for Optimal Control Problems

In this chapter, we survey solution methods for optimal control problems and mixed-integer optimal control problems.

In particular, we discuss the direct multiple shooting discretization which, applied to the relaxed partial outer convexification (RC) of a mixed-integer optimal control problem developed in Chapter 6, yields a *Mathematical Program with Vanishing Constraints*.

3.1. Approaches to solve Optimal Control Problems

We consider the optimal control problem

$$\begin{aligned} \min_{\substack{x \in W^{1,\infty}([0,1],\mathbb{R}^{n_x}),\\ u \in L^{\infty}([0,1],\mathbb{R}^{n_u})}} \quad & \phi(x(1)) \\ \text{s.t.} \quad & \dot{x}(t) = f(x(t),u(t)) \quad \text{a.e. } t \in [0,1], \\ & x(0) = x^0, \\ & 0 \leq d(x(t),u(t)) \quad \text{a.e. } t \in [0,1], \end{aligned} \tag{OCP}$$

where $D_x \subseteq \mathbb{R}^{n_x}$ and $D_u \subseteq \mathbb{R}^{n_u}$ are domains, $\phi : D_x \to \mathbb{R}$ is a continuously differentiable function and $f : D_x \times D_u \to \mathbb{R}^{n_x}$ and $d : D_x \times D_u \to \mathbb{R}^{n_d}$ are continuous functions.

This problem formulation differs from (MIOCP) considered in Chapter 6 in the omission of the discrete control function v and the omission of the combinatorial constraint $c(x(t),u(t),v(t)) \geq 0$. As noted for the mixed-integer case, it is well known that more general classes of problems can be reduced to this class of problems, namely problems with free endtime, with non-autonomous dynamics and problems with Bolza type objective function that are a sum of a Mayer type final time objective contribution and a Lagrange type objective contribution. Furthermore, problems with point constraints, multi-stage problems and problems with additional parameter dependence may be considered.

Solution approaches for (OCP) can be roughly classified into *direct methods*, *indirect methods* and *dynamic programming methods*. An exhaustive comparison of these

approaches is given for example by Binder et al. [Bin+01] and an illustrative side-by-side comparison of the methods for the example of a missile guidance problem by Subchan and Zbikowski [SZ09]. We thus give a short outline of the different approaches and focus in particular on the *direct multiple shooting* approach, which has been used as discretization technique for the optimal control examples considered in this thesis. The special case of linear optimal control problems is considered separately as their solutions satisfy a bang-bang principle that can be exploited.

3.1.1. Direct Methods

Direct methods discretize (OCP) before optimizing and yield a finite dimensional nonlinear program that can be solved using algorithms from finite-dimensional nonlinear programming. Notable direct methods are *direct single shooting*, *direct multiple shooting* and *direct collocation*.

Direct Single Shooting In direct single shooting, the control space $L^\infty([0,1], \mathbb{R}^{n_u})$ is replaced by a finite-dimensional subspace of $L^\infty([0,1], \mathbb{R}^{n_u})$ and the initial value problem constraint eliminated by integration. Satisfaction of the path constraint $d(x(t), u(t)) \geq 0$ is requested only at a finite grid in time, see for example [SS78; Kra85; Kra94] for details and practical implementation of this approach.

Single shooting has the advantage of being easy to implement, the resulting nonlinear program is low-dimensional. It suffers from the drawbacks that the resulting nonlinear program is highly nonlinear for nonlinear right-hand sides f, that the solution of the initial value problem may not exist outside of a possibly tiny vicinity of the solution and that a priori knowledge on the solution trajectory cannot be exploited in the solution process of the nonlinear program.

Direct Multiple Shooting Direct multiple shooting [Pli81; BP84; Lei95; Lei99] aims at circumventing the difficulties of single shooting. Again, the control space $L^\infty([0,1], \mathbb{R}^{n_u})$ is replaced by a finite-dimensional subspace. To eliminate the initial value problem constraint, a temporal *multiple shooting grid* is chosen and initial guesses on the *multiple shooting nodes* are introduced. The initial value problem is then integrated only over the *multiple shooting intervals* and continuity at the multiple shooting nodes of the resulting piecewise trajectory is enforced as additional constraint.

Under suitable conditions, multiple shooting reduces the nonlinearity of the resulting nonlinear program, as has been shown in the context of *Lifted Newton Methods* [AD10], enlarges the region in which the initial value problem constraint is defined and allows trivially the exploitation of a priori trajectory knowledge.

We will discuss the method in further detail in the next section and focus exclusively on direct multiple shooting in this thesis.

Direct Collocation Direct collocation [THE75; Bär83; Bie84; Sch90; Sch96; Str93; Str95] discretize states and controls on a temporal *collocation grid* and enforces the initial value problem constraint to hold on every *collocation interval* as requested by a chosen collocation scheme. This yields a possibly very large, but sparse nonlinear program.

A priori trajectory information can be exploited in direct collocation methods. Direct collocation has the drawback, that due to the chosen collocation grid, it is hard to integrate the initial value problem constraint with a given accuracy as either the step size has to be very small or, if adaptivity based on error control is used, the discretization grid has to be replaced by a suitable refinement during the course of optimization, which results in a different nonlinear program. It has furthermore been reported [KB06] that collocation shows oscillatory behavior on singular arcs and requires advanced regularization to circumvent this phenomenon.

3.1.2. Indirect Methods

Indirect methods formulate necessary optimality conditions of (OCP) in function space and then solve a discretization of these necessary conditions. Necessary optimality conditions are given by Pontryagin's *maximum principle* and are stated in the form of a multipoint boundary value problem. The maximum principle has been formulated and proven by Pontryagin, Boltyanski, Gamkrelidze and Miscenko, and their publication in the seminal book "The Mathematical Theory of Optimal Processes" [Pon+61] pointed the way for further developments and coined standard notation and terminology. Precursors to the maximum principle had already been formulated by Carathéodory and Hestenes, see Pesch and Bulirsch [Pes94] and Pesch and Plail [PP09] for historical notes. Using methods of non-smooth analysis, Clarke et al. considerably generalized the maximum principle, compare [Vin10; Cla13].

Using indirect methods it is possible to compute very accurate solutions to an optimal control problem. However, formulating and solving the arising boundary value problems can be very challenging. The boundary value problem is typically ill-conditioned and nontrivial analytical considerations are required to eliminate controls by adjoints that may introduce many different special cases to be considered. Furthermore, an analysis of the *switching structure* that defines activity of the constraints is required, which itself is highly problem- and data-dependent and requires insight into the problem to be determined. These difficulties render indirect methods impractical for many real-world applications.

Discretization of the boundary value problem leads to a root-finding problem that can be solved by a globalized Newton method, where different discretization methods for the boundary value problem lead to different indirect methods. All these methods share the property that the basin of attraction of Newton's method is typically very small due to the ill-conditioning of the boundary value problem and advanced globalization methods such as the *restrictive monotony test* [BKS00] or *backward step control* [Pot16] that are realized in the framework of *affine invariant Newton methods* [Deu74; Deu06] are required.

Popular indirect methods are *indirect single shooting*, *indirect multiple shooting* [Fox60; Kel68; Osb69; Bul71; Boc77; Boc78b; Boc78a; Boc81b; Obe86] and *indirect collocation* [Vai65; RS72; DW75; Bär83; ACR79; AMR88].

3.1.3. Dynamic Programming

Dynamic Programming is based on Bellman's principle of optimality [Bel57], that asserts that "an optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision". This principle leads to a partial differential equation, the Hamilton-Jacobi-Carathéodory-Bellman equation

$$\dot{V}(t,x) + \min_u \langle \nabla V(t,x), f(x,u) \rangle = 0, \quad V(1,x) = \phi(x),$$

for the cost-to-go function $V(t,x) := \min_{\tilde{x},\tilde{u}\text{ feasible}, x(t)=\tilde{x}} \phi(x(1))$, see e.g. [Pes94; Ber05].

Using the Hamilton-Jacobi-Carathéodory-Bellmann equation has the advantage that the obtained solution is a global optimum, as solution of the equation involves tabulation of the complete state space. In some cases it is possible to solve this equation analytically. For most instances, this is not possible and attempting to solve this equation numerically is usually impossible due to the *curse of dimensionality* unless the problem size is tiny.

Analysis of the Hamilton-Jacobi-Carathéodory-Bellmann equation is involved as, in general, it does not have a classical smooth solution and a suitable notion of generalized solutions is necessary.

3.1.4. Linear Optimal Control Problems

A linear optimal control problem is a problem such that the dynamics $f(x,u) = Ax+Bu$ are described by a linear function. Furthermore it is required that no path constraints of the form $d(x(t),u(t)) \geq 0$ are present and that only box constraints are imposed on the controls. For such problems, a bang-bang principle can be shown that asserts that

the reachable set of all admissible controls is identical to the reachable set of bang-bang controls, which take values only at the boundary of the box, see for example [HL69]. This can be exploited in solution methods as consideration can be restricted to bang-bang controls.

3.2. Direct Multiple Shooting Discretization

We now discuss a direct multiple shooting discretization of (OCP). It was introduced for optimal control problems in the diploma thesis of Plitt [Pli81] supervised by Bock and published by Bock and Plitt [BP84]. The method has been extended to systems constrained by differential-algebraic equations by Leineweber [Lei99; Lei+03b; Lei+03a], by Schlöder [Sch88] and Schäfer [Sch05] to efficiently exploit the structure of large-scale systems with a small number of degrees of freedom and by Potschka [Pot11] to partial differential equations. Gallitzendörfer and Bock [GB94; Gal97] analyze how the intrinsic parallel structure of the discretization can be exploited for efficient parallel implementations. The software packages `MUSCOD` [BP84], `MUSCOD-II` [DLS01; Die+16], `OMUSES/HQP` [FMT02], `muse` [Jan10; Jan15], `MuShROOM` [Kir+10a], `MUSCOP` [Pot11] and `ACADO` [HFD11] provide implementations of the multiple shooting discretization. We provide an implementation `OptimIND` of the multiple shooting discretization with interfaces to the `python` programming language.

For the discretization, a *multiple shooting grid* $0 = \tau_0 < \ldots < \tau_{N-1} = 1$ is used that partitions the time horizon $[0, 1]$. On every shooting interval $[\tau_i, \tau_{i+1}]$, a finite-dimensional subspace $V_i \subseteq L^\infty([\tau_i, \tau_{i+1}], \mathbb{R}^{n_u})$ is chosen with a fixed basis $\{\xi_{ij}, j \in [\dim V_i]\}$. Denote by $\xi_i : \mathbb{R}^{\dim V_i} \to V_i, q \mapsto \sum_{j\in[\dim V_i]} q_j \xi_{ij}$ the coordinate isomorphism.

Associated to every shooting node τ_i is now a guess $s^i \in \mathbb{R}^{n_x}$ of $x(\tau_i)$ and q^i parameterizing $u|[\tau_i, \tau_{i+1}]$. By $x^{i+1}(s^i, q^i)$ we denote the solution of the initial value problem $\dot{x} = f(t, x(t), \phi_i(q^i)(t)), x(\tau_i) = s^i$ evaluated at $t = \tau_{i+1}$. The multiple shooting discretization eliminates the initial value problem by integration on the shooting intervals and enforces continuity on the shooting nodes. Satisfaction of the path constraint is enforced on the shooting nodes only. This results in the following nonlinear program:

$$
\begin{aligned}
\min_{s,q} \quad & \phi(s^{N-1}) \\
\text{s.t.} \quad & 0 = x^{i+1}(s^i, q^i) - s^{i+1}, && i \in [N-1], \\
& 0 = \xi_{N-2}(q^{N-2})(\tau_{N-1}) - \xi_{N-1}(q^{N-1})(\tau_{N-1}), \\
& 0 \le d(s^i, \xi_i(q^i)(\tau_i)), && i \in [N].
\end{aligned}
$$

The nonlinear program is separable in the sense that the coupling between (s^i, q^i) and

(s^j, q^j) for all $i \neq j$ is linear. The Jacobian matrices thus have block structure, which can be exploited by using tailored linear algebra algorithms in the subproblems of nonlinear programming methods, see Bock and Plitt [Pli81; BP84; Boc87], Steinbach [Ste94; Ste95; Ste96], Leineweber [Lei95; Lei99; Lei+03a], Schäfer [Sch05] and Kirches [Kir+11; Kir10].

Enforcing the path constraint only on a discrete time grid may render the solution of the discretized problem infeasible to the solution (OCP). In many real-world applications it turns out the solution of the discretized problem satisfies the path constraint along the complete time horizon. If this is not the case, an adaptive refinement strategy of the shooting grid can be employed.

As an alternative a semi-infinite programming approach has been developed by Potschka [Pot06; PBS09] that tracks the constraint violations in the interior of shooting intervals.

3.3. Solving Mixed-Integer Optimal Control Problems

We now review approaches to compute solutions to mixed-integer optimal control problems that will be analyzed in further detail in Chapter 6.

Convexification, Relaxation and Rounding In Chapter 6 we outline an approach using partial outer convexification and reconstruction via a Vanishing Constraint convergent rounding scheme. This yields a suboptimal solution to a mixed-integer optimal control problem with arbitrary small optimality and feasibility loss. The computational effort involves solving the relaxed partial outer convexification and subsequent reconstruction using the rounding scheme. The relaxed partial outer convexification is a continuous optimal control problem with vanishing constraints, that in principle can be dealt with any of the previously mentioned methods for continuous optimal control problems. The non-convex vanishing constraint poses additional challenges and care must be exercised to properly treat this constraint. Reconstruction using the rounding scheme (VC-SOS-SUR) is computationally cheap, as the computational effort is linear in the size of the temporal grid. Kirches [Kir10; Kir+13a] has demonstrated that this approach can be sufficiently fast to be real-time feasible.

Direct Discretization Direct discretization of (MIOCP) similar as described for continuous optimal control problems yields a mixed-integer nonlinear program. Mixed-integer nonlinear programs are NP hard [GJ79]. Solution algorithms for mixed-integer nonlinear programs constitute a very active area of contemporary research,

see Belotti et al. [Bel+13] for a recent survey. A naïve approach solving a mixed-integer nonlinear program is full enumeration of the integer search space, which has a complexity exponential in the number of integer variables and is thus computationally prohibitive. Among the most successful approaches are methods based on branch-and-bound, branch-and-cut, outer approximation or Benders decomposition. Gerdts [Ger05] studies the application of branch-and-bound to a mixed-integer optimal control problem of an automotive test-drive with gear shifts. Mayne and Raković [MR03] use outer approximation for the optimal control of constrained piecewise affine-discrete time systems.

Indirect Method: Competing Hamiltonians Applying indirect methods to Mixed-Integer Optimal Control Problems is possible, as the maximum principle holds also for admissible control sets that are disjoint. Bock and Longman [BL82] developed the *Competing Hamiltonians* approach for the computation of energy-optimal braking of the New York subway. This method also suffers from the mentioned drawbacks of the indirect approach and, in addition, requires computation and comparison of the values of the Hamilton function for every possible mode of operation.

Dynamic Programming Dynamic Programming can be directly applied to Mixed-Integer Optimal Control Problems as integer controls can be naturally treated without control space discretization, the approach is appealing as it yields a global solution. For all but very tiny examples it is however impractical as the *curse of dimensionality* inhibits the necessary space tabulation. Buchner [Buc10] and Hellström et al. [Hel+09] have applied dynamic programming to mixed-integer control of trucks.

Linear Case If (MIOCP) or the partial outer convexification of (MIOCP) is a linear optimal control problem, the bang-bang principle holds and thus optimal solutions to the mixed-integer problem can be found among optimal solutions of the relaxed problem.

3.4. Summary

In this chapter, we have reviewed different methods to solve optimal control problems and mixed-integer optimal control problem. For our purposes, the method of choice to discretize optimal control problems is *Bock's direct multiple shooting method*. We will approach mixed-integer optimal control problems by the partial outer convexification, relaxation and reconstruction approach laid out in detail in Chapter 6. Discretization of the relaxation of the partial outer convexification of a mixed-integer optimal control

problem leads to a *Mathematical Program with Vanishing Constraints*. The class of Mathematical Programs with Vanishing Constraints will be discussed in Chapter 5 and, in Chapter 7, a novel algorithm for solving Mathematical Programs with Vanishing Constraints will be presented.

4. Nonlinear Programming

The aim of this chapter is to provide necessary background from nonlinear programming to discuss and analyze Mathematical Programs with Vanishing Constraints, Mathematical Programs with Equilibrium Constraints and algorithms for their solution. We define a nonlinear programming problem in finite-dimensional real space and review necessary concepts from convex analysis to state necessary conditions for solutions of nonlinear programs. We refer to the text books by Clarke [Cla13], Rockafellar [Roc70] and Nocedal and Wright [NW06] as references.

4.1. Problem Definition

4.1 Definition (Nonlinear Programming Problem).
Let $\mathcal{I}, \mathcal{E}$ be disjoint finite index sets, $f : \mathbb{R}^n \to \mathbb{R}$ and $c : \mathbb{R}^n \to \mathbb{R}^{\mathcal{I}\cup\mathcal{E}}$ be continuously differentiable functions.

The problem

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & 0 = c_{\mathcal{E}}(x), \\ & 0 \le c_{\mathcal{I}}(x), \end{aligned} \tag{NLP}$$

is the *nonlinear program* defined by $f, c, \mathcal{I}$ and $\mathcal{E}$. △

It is possible to formulate problems with lower and upper bounds for the constraints in this form by introducing slack variables. For theoretical purposes this form is thus no loss of generality. In implementations we consider instead always a formulation with lower and upper bounds on the constraints, as this liberates from introducing slacks and enlarging problem dimensions.

4.2 Definition (Local Solution of Nonlinear Program).
The vector $x^* \in \mathbb{R}^n$ is a *local solution* of (NLP), if there is a neighborhood U of x^* such that for all $x \in U$ with $c_{\mathcal{E}}(x) = 0, c_{\mathcal{I}}(x) \ge 0$ the relation $f(x^*) \le f(x)$ holds.

It is a *strict local solution*, if furthermore $f(x^*) < f(x)$ provided that $x \ne x^*$. △

4.3 Definition (Feasible Set).
The *feasible set* of (NLP) is defined as the set

$$\Omega := \{x \in \mathbb{R}^n \mid c_{\mathcal{E}}(x) = 0, c_{\mathcal{I}}(x) \ge 0\}.$$ △

4.4 Definition (Active Set).
Let $x \in \Omega$ be feasible for (NLP). The index set $\mathcal{A}(x) := \{i \mid c_i(x) = 0\}$ is the *active set* at x. △

4.5 Definition (Lagrange function).
The *Lagrange function* associated to (NLP) is $L : \mathbb{R}^n \times \mathbb{R}^{\mathcal{E}} \times \mathbb{R}^{\mathcal{I}} \to \mathbb{R}$ defined by $L(x, \lambda, \mu) := f(x) - \langle \lambda, c_{\mathcal{E}}(x) \rangle - \langle \mu, c_{\mathcal{I}}(x) \rangle$. △

4.2. Selected Material from Convex Analysis

To formulate necessary conditions we need the concepts of *cones*, in particular the tangential and feasibility cone.

4.6 Definition (Cone, Polar Cone).
A set $C \subseteq \mathbb{R}^k$ is a *cone*, if $\mathbb{R}_{\geq 0} C \subseteq C$.
The *polar cone* of a cone C is defined as $C^\circ := \{y \in \mathbb{R}^k \mid \langle y, x \rangle \leq 0 \text{ for all } x \in C\}$. △

4.7 Proposition.
Let C, C_1, C_2 be cones.

(1) C° is closed convex.

(2) $C_1 \subseteq C_2 \Rightarrow C_2^\circ \subseteq C_1^\circ$

(3) $C^{\circ\circ} = \overline{\bigcap_{L \text{ convex}, C \subseteq L} L}$.

(4) If C_1, C_2 are convex and $C_1^\circ = C_2^\circ$, then $C_1 = C_2$. △

4.8 Definition (Tangential and Feasibility Cone).
Let $x \in \Omega$ be feasible for (NLP).

The *tangential cone* of Ω at x is defined by

$$T(\Omega, x) := \left\{ d \in \mathbb{R}^n \mid \text{there is } (x_n)_{n \in \mathbb{N}} \text{ in } \Omega \text{ with } x_n \to x, \tfrac{x_n - x}{\|x_n - x\|} \to \tfrac{d}{\|d\|} \right\}.$$

The *linearized feasibility cone* of Ω at x is defined by

$$F(\Omega, x) := \bigcap_{\substack{i \in \mathcal{I}, \\ c_i(x) = 0}} \{d \mid \langle \nabla c_i(x), d \rangle \geq 0\} \quad \cap \quad \bigcap_{i \in \mathcal{E}} \{d \mid \langle \nabla c_i(x), d \rangle = 0\}.$$

△

Both $T(\Omega, x)$ and $F(\Omega, x)$ are closed cones, $F(\Omega, x)$ is in addition also a convex cone. Note that $T(\Omega, x)$ is a geometric object as it depends on the function c only via the feasible region Ω, while $F(\Omega, x)$ depends directly on c. The inclusion $T(\Omega, x) \subseteq F(\Omega, x)$ holds.

4.9 Lemma (Farkas [Far02]).
Let $A \in \mathbb{R}^{n\times m}$ and $b \in \mathbb{R}^n$. Then exactly one of the following statements is true:

(1) There is $x \in \mathbb{R}^m$ such that $x \geq 0$ and $Ax = b$.

(2) There is $y \in \mathbb{R}^n$ such that $\langle A, y\rangle \geq 0$ and $\langle b, y\rangle < 0$. △

From a geometric viewpoint, the Farkas Lemma is essentially a separation property of finite-dimensional real space. In infinite-dimensional Banach or Hilbert spaces, separation theorems may be fundamentally different from the finite-dimensional case, see [Roc70]. This demonstrates that generalizations of necessary conditions to infinite-dimensional Banach or Hilbert spaces cannot be obtained by trivial generalization of the finite-dimensional case.

4.3. Necessary Conditions for Nonlinear Programs

We first state necessary conditions for the solution of the nonlinear program assuming the general Guignard Constraint Qualification and discuss simpler Constraint Qualifications that imply this condition afterwards.

4.10 Theorem (Karush-Kuhn-Tucker Conditions, [Kar39; KT51]).
Let $x^* \in \Omega$ be a local minimizer of f such that the regularity condition GCQ $T(\Omega, x^*)^\circ = F(\Omega, x^*)^\circ$ holds.

Then there exist Lagrange Multipliers $\lambda^* \in \mathbb{R}^{\mathcal{E}}, \mu^* \in \mathbb{R}^{\mathcal{I}}$ *satisfying:*

$$\begin{aligned} &\nabla_x L(x^*, \lambda^*, \mu^*) = 0 && \text{(Stationarity)},\\ &\mu^* \geq 0 && \text{(Dual Feasibility)},\\ &\langle \mu^*, c_{\mathcal{I}}(x^*)\rangle = 0 && \text{(Complementarity)}. \end{aligned} \tag{KKT}$$

Proof. Note that $\langle \nabla f(x^*), d\rangle \geq 0$ for all $d \in T(\Omega, x^*)$ is necessary for x^* to be a local minimizer. This means $-\nabla f(x^*) \in T(\Omega, x^*)^\circ$.

By means of Farkas' Lemma, satisfaction of the (KKT) is equivalent to $-\nabla f(x^*) \in F(\Omega, x^*)^\circ$: Setting $A := \begin{pmatrix} \nabla c_{\mathcal{E}}(x^*) & -\nabla c_{\mathcal{E}}(x^*)^T & \nabla c_{\mathcal{I}}(x^*)^T \end{pmatrix}$, it holds for all $d \in \mathbb{R}^n$ that $d \in F(\Omega, x^*)$ if and only if $A^T d \geq 0$. Thus $-\nabla f(x^*) \in F(\Omega, x^*)^\circ$ if and only if

$\langle \nabla f(x^*), d\rangle \geq 0$ for all $d \in \mathbb{R}^n$ with $A^T d \geq 0$. Alternative 2 of Farkas' Lemma cannot be satisfied and hence there must be $\xi = \begin{pmatrix} \lambda^+ \\ \lambda^- \\ \mu \end{pmatrix} \in \mathbb{R}^{\mathcal{E}\dot\cup\mathcal{E}\cup\mathcal{I}}$ with $\xi \geq 0$ and $A\xi = -\nabla f$.

By GCQ $T(\Omega, x^*)^\circ = F(\Omega, x^*)^\circ$ which proves the Theorem. □

The constraint qualification GCQ cannot be dropped, as in general, only $T(\Omega, x^*) \subseteq F(\Omega, x^*)$ and thus $F(\Omega, x^*)^\circ \subseteq T(\Omega, x^*)^\circ$, but there are examples for which the polar cones $F(\Omega, x^*)^\circ$ and $T(\Omega, x^*)^\circ$ are not identical.

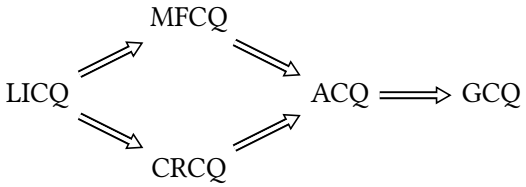
4.4. Constraint Qualifications

The condition GCQ is unsuited to be checked in algorithms and hard to handle in computations. This motivates the consideration of further, stronger regularity conditions. For a review of constraint qualifications, proofs of their relationships as well as examples and counterexamples see Peterson [Pet73].

4.11 Definition (Constraint Qualifications).
Let $x \in \Omega$ be feasible.

- **LICQ:** Linear Independence CQ, [Hes66, p. 29] holds at x if $(\nabla c_i)_{i\in\mathcal{A}(x)}$ is linear independent.
- **MFCQ:** Mangasarian-Fromovitz CQ, [MF67] holds at x if $(\nabla c_i(x))_{i\in\mathcal{E}}$ is linear independent and there is d with $\langle \nabla c_{\mathcal{E}}(x), d\rangle = 0$, and $\langle \nabla c_i(x), d\rangle > 0$ for all $i \in \mathcal{I}$ with $c_i(x) = 0$.
- **SMFCQ:** Strict Mangasarian-Fromovitz CQ at (x, λ, μ), where (x, λ, μ) satisfy (KKT), [MF67] holds at x if $(\nabla c_i(x))_{i\in\mathcal{E}}$ is linear independent and there is d with $\langle \nabla c_{\mathcal{E}}(x), d\rangle = 0$ and $\langle \nabla c_i(x), d\rangle = 0$ for all $i \in \mathcal{I}$ with $c_i(x) = 0, \mu_i > 0$ and $\langle \nabla c_i(x), d\rangle > 0$ for all $i \in \mathcal{I}$ with $c_i(x) = 0, \mu_i = 0$.
- **CRCQ:** Constant Rank CQ, [Jan84] holds at x if for every $\mathcal{J} \subseteq \mathcal{A}(x)$ the rank of $\nabla c_{\mathcal{J}}(y)$ is constant in a neighborhood of x.
- **ACQ:** Abadie CQ, [Aba67] holds at x if $T(\Omega, x) = F(\Omega, x)$.
- **GCQ:** Guignard CQ [Gui69] holds at x if $T(\Omega, x)^\circ = F(\Omega, x)^\circ$ △

4.12 Lemma (Relationship between Constraint Qualifications).
Let $x \in \Omega$ be feasible. Then the following implications hold:



△

4.5. Uniqueness of Multipliers

For a fixed local minimizer $x^* \in \Omega$, multipliers λ^*, μ^* satisfying (KKT) need not necessarily be unique. The following Theorem relates the set of multipliers to constraint qualifications and states that multipliers are unique provided that LICQ holds.

4.13 Theorem (Constraint Qualifications and Uniqueness of Multipliers).
Let $x^* \in \Omega$ be a local minimizer of f. Let $\Lambda := \{(\lambda, \mu) \mid (x^*, \lambda, \mu \text{ satisfy (KKT)}\}$. Then

(1) Λ is closed and convex.

(2) If GCQ holds, then $\Lambda \neq \emptyset$.

(3) MFCQ holds if and only if Λ is a compact set.

(4) If LICQ holds, then $\Lambda = \{(\lambda^*, \mu^*)\}$ is a singleton.

(5) If $\Lambda \neq \emptyset$, then Λ is a singleton if and only if there is (λ^*, μ^*) such that SMFCQ holds at (x^*, λ^*, μ^*). △

In addition to the already cited references, see [Wac13] for a discussion and proofs.

4.6. Penalization

For theoretical and practical purposes it is important to note that there is an equivalence between (NLP) and a class of unconstrained non-smooth optimization problems, denoted by *penalization*. This is often exploited as merit function mechanism in globalization strategies of algorithms that attempt to solve (NLP) and we will make use of it in the construction of a SLPECEQP method. The penalization result here has been established by Han and Mangasarian [HM79, Section 4].

4.14 Theorem (Penalization).
Let $\|\cdot\|_{\mathcal{E}}$ be a norm on $\mathbb{R}^{\mathcal{E}}$ and $\|\cdot\|_I$ a norm on $\mathbb{R}^{\mathcal{I}}$, and $\|\cdot\|'_{\mathcal{E}}$, $\|\cdot\|'_I$ their respective dual norms.

Let $\phi(x,\gamma) := f(x) + \gamma\|c_{\mathcal{E}}(x)\|_{\mathcal{E}} + \gamma \min\{0, \|c_I(x)\}\|_I$.

If x^* is a local minimizer of (NLP) such that MFCQ holds with (x^*, λ^*, μ^*) satisfying (KKT), then x^* is a local minimizer of $\phi(x,\gamma)$ for every $\gamma > \max\{\|\lambda^*\|'_{\mathcal{E}}, \|\mu^*\|'_I\}$.

If there is $\overline{\gamma}$ such that $\lim_{t\downarrow 0} \frac{1}{t}(\phi(x^* + td, \gamma) - \phi(x^*, \gamma)) = 0$ for all $d \in \mathbb{R}^n$ and $\gamma \geq \overline{\gamma}$, then there are λ^*, μ^* such that (x^*, λ^*, μ^*) satisfy (KKT). △

4.7. Summary

In this chapter, we have collected the necessary material from convex analysis to analyze *Mathematical Programs with Vanishing and Equilibrium Constraints* in Chapter 5 and the novel algorithm to solve these programs in Chapter 7. We have discussed necessary conditions of nonlinear programs and constraint qualifications that imply necessary conditions. The relation between uniqueness of multipliers and constraint qualifications has been considered and penalty reformulations of nonlinear programs have been given.

5. Mathematical Programs with Vanishing and Equilibrium Constraints

In this chapter, we introduce two challenging classes of nonlinear problems, *Mathematical Programs with Vanishing Constraints* (MPVC) and *Mathematical Programs with Equilibrium Constraints* (MPEC). We review their properties, lack of constraint qualifications, stationarity concepts and existing solution approaches. We note that MPVC can be considered as a subclass of MPEC. MPVC play an important role in mixed-integer optimal control problems, as already seen in Chapter 6 discretizations of such problems lead to MPVC. In the subsequent chapter, we propose a Sequential Linear Equilibrium Constraint Equality Constraint Quadratic Programming Method (SLPECEQP) for MPEC.

MPVC and MPEC have attracted a lot of theoretical and algorithmic research interest in the past twenty years due to the challenges posed by this problem class and the wide applicability to real-world problems. The results presented in this section are a summary of the analyses of Scheel and Scholtes [SS00; Sch02; Sch04], Outrata [Out99; Out00], Izmailov and Solodov [IS09], Achtziger and Kanzow [AK08], Flegel and Kanzow [FK03; Fle05], Hoheisel and Kanzow [HK07; HK08; Hoh09; HK09b; HKS13], Kanzow and Schwartz [KS13], Luo, Pang, Ralph [LPR96], Pang and Fukushima [PF99] and Gfrerer [Gfr14].

Furthermore, MPVC and MPEC in infinite dimensional spaces are analyzed by Hintermüller et al. [HK09a; HS11] and Wachsmuth [Wac15; Wac16]. Considering bilevel optimization and bilevel optimal control problems leads in suitable formulation to MPEC. MPEC results in bilevel form have been obtained by Chen and Florian [CKA95], Ye [Ye95; Ye00; Ye05], Dempe [Dem02; DZ13] and Mehlitz [Meh17].

5.1. Mathematical Programs with Vanishing Constraints

Mathematical Programs with Vanishing Constraints (MPVC) constitute a challenging class of nonlinear programs and in our context naturally arise from discretizations of mixed-integer optimal control problems. Their feasible set is non-convex and

violates constraint qualifications, which renders standard algorithms for nonlinear program unsuitable as they rely, for instance, on satisfaction of LICQ. Mathematical Programs with Vanishing Constraints can be seen by introduction of slacks to be a subclass of *Mathematical Programs with Equilibrium Constraints* (MPEC). We will note below that the slack must not bee unique and thus Mathematical Programs with Vanishing Constraints are slightly less degenerate than Mathematical Programs with Equilibrium Constraints.

5.1 Definition (MPVC).
Let $\mathcal{I}, \mathcal{E}$ be disjoint finite index sets, $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \to \mathbb{R}$, $c : \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \to \mathbb{R}^{\mathcal{I} \cup \mathcal{E}}$ and $g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \to \mathbb{R}^{n_s}$ be continuously differentiable functions.

$$\begin{aligned} \min_{x,s} \quad & f(x,s) \\ \text{s.t.} \quad & 0 = c_{\mathcal{E}}(x,s), \\ & 0 \le c_{\mathcal{I}}(x,s), \\ & 0 \le s * g(x,s), \\ & 0 \le s, \end{aligned} \tag{MPVC}$$

is called a *Mathematical Program with Vanishing Constraints*. Here $*$ denotes the component-wise product of vectors. △

If $s_i = 0$, the constraint $s_i g_i(x) \ge 0$ is satisfied regardless of $g_i(x,s)$, the constraint $g_i(x,s)$ "vanishes". This gives the equivalent logical reformulation of (MPVC):

$$\begin{aligned} \min_{x,s} \quad & f(x,s) \\ \text{s.t.} \quad & 0 = c_{\mathcal{E}}(x,s), \\ & 0 \le c_{\mathcal{I}}(x,s), \\ & 0 \le s, \\ & 0 < s_i \quad \Rightarrow \quad 0 \le g_i(x,s). \end{aligned}$$

This logical reformulation is a special case of so called *Constraint Programming Problems* which are studied in the context of Mixed-Integer Linear Programming Problems by Achterberg [Ach07]. Variants with two-sided general constraints of the form $0 \le g_1(x) * g_2(x)$ can be reduced to (MPVC) by introduction of slack variables.

Part of the challenges given by MPVC is that they do not satisfy regularity assumptions that nonlinear programs are usually expected to satisfy:

5.2 Proposition (Violation of standard constraint qualifications for MPVC).
Let x, s be feasible for (MPVC).

If $\{i \mid s_i = 0\} \neq \emptyset$, then LICQ is violated in x, s.

If $\{i \mid s_i = 0 \text{ and } g_i(x,s) \ge 0\} \neq \emptyset$, then MFCQ is violated in x, s. △

For a proof see [AK08]. Considering Theorem 4.13, violation of LICQ may result in non-unique multipliers. Violation of MFCQ implies non-uniqueness of multipliers and unboundedness of the set of multipliers.

5.2. Mathematical Programs with Equilibrium Constraints

Mathematical Programs with Equilibrium Constraints constitute a more general class with even less regularity than MPVC. They are defined as follows:

5.3 Definition (MPEC).
Let $\mathcal{I}, \mathcal{E}$ be disjoint finite index sets, $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s} \to \mathbb{R}$ and $c : \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s} \to \mathbb{R}^{\mathcal{I} \cup \mathcal{E}}$ be continuously differentiable functions.

$$\begin{aligned} \min_{x,s,t} \quad & f(x,s,t) \\ \text{s.t.} \quad & 0 = c_{\mathcal{E}}(x,s,t), \\ & 0 \le c_{\mathcal{I}}(x,s,t), \\ & 0 \le s \perp t \ge 0, \end{aligned} \tag{MPEC}$$

is called a *Mathematical Program with Equilibrium Constraints*, sometimes also called *Mathematical Program with Complementarity Constraints*. Here $0 \le s \perp t \ge 0$ denotes $0 \le s, 0 \le t, \langle s, t \rangle = 0$. △

Again, with two-sided general constraints of the form $0 \le h_1(x) \perp h_2(x) \ge 0$ can be reduced to the presented form by introduction of slack variables.

As for MPVC, MPEC lack satisfaction of standard constraint qualifications:

5.4 Proposition (Violation of standard constraint qualifications for MPEC).
Let x, s, t be feasible for (MPEC). Then MFCQ is violated for the nonlinear programming formulation of (MPEC):

$$\begin{aligned} \min_{x,s,t} \quad & f(x,s,t) \\ \text{s.t.} \quad & 0 = c_{\mathcal{E}}(x,s,t), \\ & 0 \le c_{\mathcal{I}}(x,s,t), \\ & 0 \le s, \\ & 0 \le t, \\ & 0 = \langle s, t \rangle. \end{aligned}$$

△

For a proof see [CKA95; SS00]. Violation of MFCQ implies non-uniqueness of multipliers and unboundedness of the multiplier set, see Theorem 4.13.

5.3. Relation between MPVC and MPEC

We will now show that MPVC can be considered as subclass of MPEC by introduction of slacks:

5.5 Lemma (MPVCs are MPECs).
Every MPVC can be written as an MPEC by the introduction of slack variables:
Then the problem (MPVC) has a solution if and only if the problem

$$\begin{aligned} \min_{x,s,t} \quad & f(x,s) \\ \text{s.t.} \quad & 0 = c_{\mathcal{E}}(x,s), \\ & 0 \le c_{\mathcal{I}}(x,s), \\ & 0 \le g(x,s) + t, \\ & 0 \le s \perp t \ge 0, \end{aligned}$$

has a solution. △

Note however, that the slack t is not uniquely defined and thus MPVC are truly a different class of problems than MPEC and must be treated separately from MPEC for certain questions. Nevertheless, for our purposes we will treat (MPVC) as (MPEC) by introduction of slacks and thus focus from now on (MPEC).

5.4. Related Nonlinear Programs to MPEC

In this section, we introduce several nonlinear programs that are related to (MPEC) as a tool for the analysis of (MPEC) in a decomposition approach.

5.6 Definition (Set of Degenerate Indices).
Let x, s, t be feasible for (MPEC). Then the *set of degenerate indices* is defined by $\mathcal{D}(x,s,t) := \{i \mid s_i = t_i = 0\}$. △

5.7 Definition (MPEC-Lagrangian).
The MPEC Lagrangian

$$L_\perp : \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s} \times \mathbb{R}^{\mathcal{E}} \times \mathbb{R}^{\mathcal{I}} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s} \to \mathbb{R}$$

is given by

$$L_\perp(x,s,t,\lambda,\mu,\nu,\sigma) := f(x,s,t) - \langle \lambda, c_{\mathcal{E}}(x,s,t)\rangle - \langle \mu, c_{\mathcal{I}}(x,s,t)\rangle - \langle \nu, s\rangle - \langle \sigma, t\rangle.$$

△

The MPEC Lagrangian differs from the standard Lagrangian L of a nonlinear program as defined in 4.5 with nonlinear formulation of the equilibrium constraint $0 \leq s, 0 \leq t, 0 = \langle s, t \rangle$ in omitting the multiplier pairing for $0 = \langle s, t \rangle$.

5.8 Definition (TNLP, RNLP, branch NLP$_{(\mathcal{S}_0,\mathcal{S}_+)}$).
Let $\bar{x}, \bar{s}, \bar{t}$ be feasible for (MPEC). The *tightened NLP* at $\bar{x}, \bar{s}, \bar{t}$ is defined as

$$\begin{array}{lll} \min\limits_{x,s,t} & f(x,s,t) & \\ \text{s.t.} & 0 = c_{\mathcal{E}}(x,s,t), & \\ & 0 \leq c_{\mathcal{I}}(x,s,t), & \\ & 0 = s_i, & \text{if } \bar{s}_i = 0, \\ & 0 \leq s_i, & \text{if } \bar{s}_i > 0, \\ & 0 = t_i, & \text{if } \bar{t}_i = 0, \\ & 0 \leq t_i, & \text{if } \bar{t}_i > 0, \end{array} \tag{TNLP}$$

the *relaxed NLP* at $\bar{x}, \bar{s}$ is defined as

$$\begin{array}{lll} \min\limits_{x,s,t} & f(x,s,t) & \\ \text{s.t.} & 0 = c_{\mathcal{E}}(x,s,t), & \\ & 0 \leq c_{\mathcal{I}}(x,s,t), & \\ & 0 = s_i, & \text{if } \bar{t}_i > 0, \\ & 0 \leq s_i, & \text{if } \bar{t}_i = 0, \\ & 0 = t_i, & \text{if } \bar{s}_i > 0, \\ & 0 \leq t_i, & \text{if } \bar{s}_i = 0. \end{array} \tag{RNLP}$$

Let $(\mathcal{S}_0, \mathcal{S}_+)$ be a partition of $\mathcal{D}(\bar{x}, \bar{s}, \bar{t})$, $\mathcal{D}(\bar{x}, \bar{s}, \bar{t}) = \mathcal{S}_0 \dot{\cup} \mathcal{S}_+$. Then the branch NLP$_{(\mathcal{S}_0, \mathcal{S}_+)}$ is defined as

$$\begin{array}{lll} \min\limits_{x,s,t} & f(x,s,t) & \\ \text{s.t.} & 0 = c_{\mathcal{E}}(x,s,t), & \\ & 0 \leq c_{\mathcal{I}}(x,s,t), & \\ & 0 = s_i, & \text{if } i \in \mathcal{S}_0 \text{ or } \bar{t}_i > 0, \\ & 0 \leq s_i, & \text{if } i \in \mathcal{S}_+, \\ & 0 = t_i, & \text{if } i \in \mathcal{S}_+ \text{ or } \bar{s}_i > 0, \\ & 0 \leq t_i, & \text{if } i \in \mathcal{S}_0. \end{array} \tag{NLP$_{(\mathcal{S}_0,\mathcal{S}_+)}$}$$

△

Using the branch programs, the MPEC can be decomposed by taking partitions of the set of degenerate indices. As the number of partitions of $\mathcal{D}(\bar{x}, \bar{s}, \bar{t})$ is $2^{|\mathcal{D}(\bar{x},\bar{s},\bar{t})|}$, a decomposition approach requires consideration of $2^{|\mathcal{D}(\bar{x},\bar{s},\bar{t})|}$ branch programs which demonstrates the combinatorial nature of this class of problems. The branch programs allow a description of the feasibility cones of the MPEC:

5.9 Lemma ([SS00]).
For the feasibility cones $F^{\mathrm{TNLP}}(\Omega_{\mathrm{TNLP}}, \bar{x}, \bar{s}, \bar{t})$, $F^{\mathrm{MPEC}}(\Omega_{\mathrm{MPEC}}, \bar{x}, \bar{s}, \bar{t})$, $F^{\mathrm{RNLP}}(\Omega_{\mathrm{RNLP}}, \bar{x}, \bar{s}, \bar{t})$ at $\bar{x}, \bar{s}, \bar{t}$ the following inclusions hold:

$$F^{\mathrm{TNLP}} = \bigcap_{\mathcal{S}_0 \dot\cup \mathcal{S}_+ = \mathcal{D}(\bar{x}, \bar{s}, \bar{t})} F^{\mathrm{NLP}_{(\mathcal{S}_0, \mathcal{S}_+)}} \subseteq F^{\mathrm{MPEC}} \subseteq \bigcup_{\mathcal{S}_0 \dot\cup \mathcal{S}_+ = \mathcal{D}(\bar{x}, \bar{s}, \bar{t})} F^{\mathrm{NLP}_{(\mathcal{S}_0, \mathcal{S}_+)}} = F^{\mathrm{RNLP}}.$$

△

5.10 Lemma ([SS00]).
Let x, s, t be feasible for (MPEC).

(1) The point x, s, t is a local minimizer of (MPEC) if and only if it is a local minimizer of $(\mathrm{NLP}_{(\mathcal{S}_0, \mathcal{S}_+)})$ for every partition $\mathcal{D}(x, s, t) = \mathcal{S}_0 \dot\cup \mathcal{S}_+$.

(2) If x, s, t is a local minimizer of (RNLP), it is a local minimizer of (MPEC).

(3) If x, s, t is a local minimizer of (MPEC), it is a local minimizer of (TNLP).

(4) If strict complementarity holds at x, s, t, i.e. $\mathcal{D}(x, s, t) = \emptyset$, then $F^{\mathrm{TNLP}} = F^{\mathrm{MPEC}} = F^{\mathrm{RNLP}}$ and x, s, t is a local minimizer of (MPEC), if and only if it a local minimizer of (TNLP) and if and only if it is a local minimizer of (RNLP). △

5.11 Definition (CQ for MPEC).
We say that (MPEC) satisfies a constraint qualification in $\bar{x}, \bar{s}, \bar{t}$, if this is true for (TNLP) at $\bar{x}, \bar{s}, \bar{t}$ of MPEC. △

If (MPEC) satisfies MPEC-LICQ or MPEC-MFCQ in $\bar{x}, \bar{s}, \bar{t}$, then (TNLP), (RNLP) and $(\mathrm{NLP}_{(\mathcal{S}_0, \mathcal{S}_+)})$ satisfy LICQ respective MFCQ.

5.5. Stationarity for MPEC

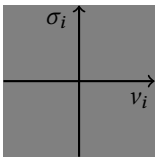
5.12 Definition (Stationarity for MPEC).
Let x, s, t be feasible for (MPEC).

- B-stationarity [Luo+96]: The point x, s, t satisfies ***B**ouligand stationarity* if $d = 0$ is a local minimizer of the Linear Program with Equilibrium Constraints obtained by linearizing (MPEC) at x, s, t:

$$\begin{aligned} \min_d \quad & \langle \nabla f(x,s,t), d \rangle \\ \text{s.t.} \quad & 0 = c_{\mathcal{E}}(x,s,t) + \langle \nabla c_{\mathcal{E}}(x,s,t), d \rangle, \\ & 0 \le c_{\mathcal{I}}(x,s,t) + \langle \nabla c_{\mathcal{I}}(x,s,t), d \rangle, \\ & 0 \le s + d_s \perp t + d_t \ge 0. \end{aligned}$$

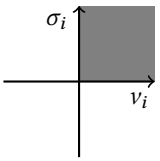
- weak stationarity [SS00]: The point x, s, t satisfies *weak stationarity* if the corresponding (TNLP) at x, s, t admits satisfaction of Karush-Kuhn-Tucker conditions, i.e. there are ν, σ such that

$$\begin{aligned} \nabla_{x,s,t} L_\perp(x, s, t, \lambda, \mu, \nu, \sigma) &= 0, \\ 0 \le \mu \perp c_{\bar{I}}(x, s, t) &\ge 0, \\ s &\ge 0, \\ t &\ge 0, \\ \nu * s &= 0, \\ \sigma * t &= 0. \end{aligned}$$



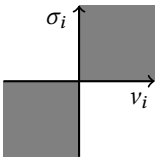
- strong stationarity [LPR96]: The point x, s, t satisfies *strong stationarity* if it is weakly stationary and

$$\nu_i \ge 0 \text{ and } \sigma_i \ge 0 \quad \text{if } i \in \mathcal{D}(x, s, t).$$



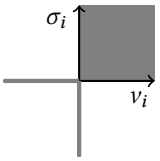
- C-stationarity [SS00]: The point x, s, t satisfies ***C**larke stationarity* if it is weakly stationary and

$$\nu_i \sigma_i \ge 0 \quad \text{if } i \in \mathcal{D}(x, s, t).$$



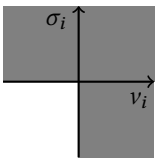
- M-stationarity [YY97; Out99; Out00; Ye00]: The point x, s, t satisfies ***M**ordukhovich stationarity* if it is weakly stationary and

$$(\nu_i > 0 \text{ and } \sigma_i > 0) \text{ or } \nu_i \sigma_i = 0 \quad \text{if } i \in \mathcal{D}(x, s, t).$$



- A-stationarity [FK03]: The point x, s, t satisfies ***A**badie/**A**lternative stationarity* if it is weakly stationary and

$$\nu_i \ge 0 \text{ or } \sigma_i \ge 0 \quad \text{if } i \in \mathcal{D}(x, s, t).$$

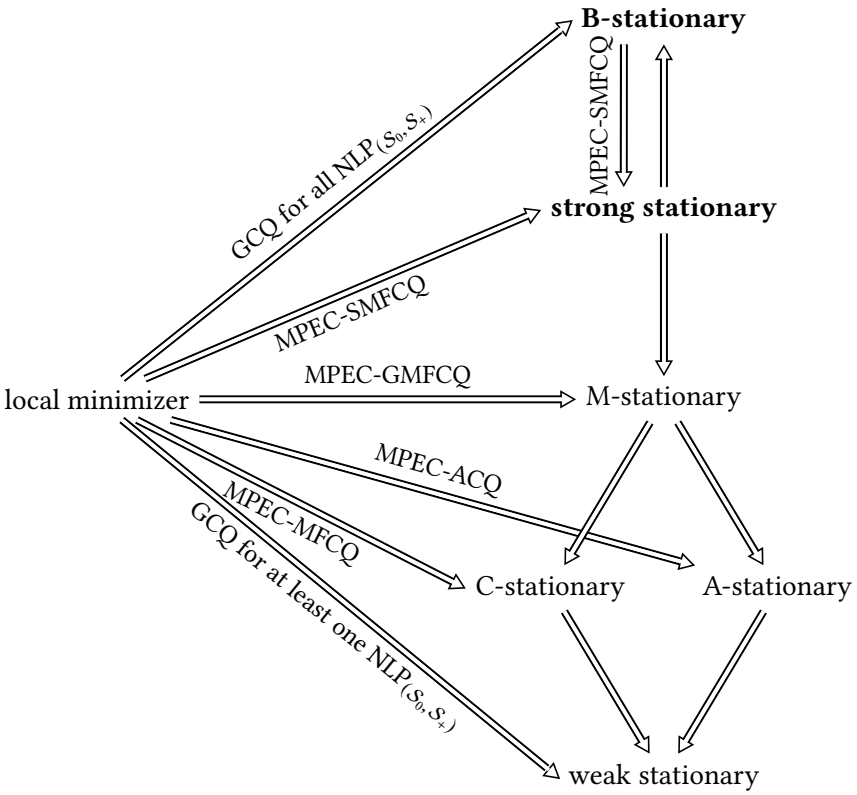


△

With the exception of A-stationarity, the names for these stationarity concepts have been coined by Scheel and Scholtes [SS00].

The following Theorem summarizes the relationship between the stationarity concept and shows that Bouligand stationarity and strong stationarity are equivalent if MPEC-SMFCQ holds, and that all other stationarity concepts are weaker.

5.13 Theorem (Relations between stationarity concepts).
Let x, s, t be feasible for (MPEC). Then the following implications hold:



△

For a definition of MPEC-GMFCQ we refer to [Ye05] and for [SS00; LPR96; Out99; FK03; Ye05] proofs of the statement of the Theorem.

We consider Bouligand stationarity as the stationarity concept of interest, as it prevents the existence of first-order descent directions by definition. Clarke stationarity, Mordukhovich stationarity and weak stationarity suffer from this artifact. They can be prone to *spurious* points: These are points that satisfy the respective necessary condition, but descent directions do exist and thus they cannot be local minimizers.

Scheel and Scholtes [SS00] give the following counterexample for the existence of descent directions in points satisfying Clarke and weak stationarity:

5.14 Example (Descent directions in C-stationary points).
Consider the following problem:

$$\begin{aligned} \min_{s,t} \quad & (s-1)^2 + (t-1)^2 \\ \text{s.t.} \quad & 0 \le s \perp t \ge 0. \end{aligned}$$

Then $(s, t, v, \sigma) = (0, 0, -2, -2)$ is Clarke stationary and thus weak stationary, but increasing either s or t reduces the objective. The unit vectors are descent directions.△

As counterexample for Mordukhovich stationarity, Leyffer and Munson [LM07] give the following problem:

5.15 Example (Descent directions in M-stationary points).
Consider the following problem:

$$\begin{aligned} \min_{s,t} \quad & (t-1)^2 + s^2(s+1) \\ \text{s.t.} \quad & 0 \le s \perp t \ge 0. \end{aligned}$$

Then $(s, t, v, \sigma) = (0, 0, -2, 0)$ is Mordukhovich, Clarke and weak stationary, but increasing t decreases the objective function. △

5.6. Numerical Methods for MPEC

Several approaches have been considered in the literature to solve Mathematical Programs with Equilibrium Constraints. These can be classified into nonlinear equation approaches, smoothing approaches and structural approaches.

5.6.1. Nonlinear Equation and Smoothing, Regularization and Penalization Approach

In nonlinear equation, smoothing, regularization and penalization approaches, the complementarity constraint is expressed as a nonlinear equation. These approaches have been successful in a range of applications spanning from chemical engineering [RDB04; BRB08], switched systems [BB09], engineering and economics [FP97], market and risk analysis [Su05], automotive engineering [Kir+10a] to locomotion and cerebral palsy gait modeling [Hat14].

In general for these approaches only convergence to points satisfying weak, M-, C- or A-stationarity can be shown, whereas B-stationarity is the desired stationarity

criterion. Convergence to spurious stationary points have been observed in real-world applications for example by Chen et al. [Che+06] in an application involving the Pennsylvania-Jersey-Maryland electricity market.

Nonlinear Equation Approach In the nonlinear equation approach, the equilibrium constraint $0 \leq s \perp h(x,s) \geq 0$ is formulated as nonlinear equation, for example as $0 \leq s, \sigma - h(x,s) = 0, 0 \leq \sigma, \langle s, \sigma \rangle \leq 0$, [Ley06] and the resulting nonlinear program treated with an algorithm for solving nonlinear programs. A major obstacle of the nonlinear equation approach is that the resulting nonlinear program does not satisfy constraint qualifications as LICQ or MFCQ. Convergence proofs of standard nonlinear programming algorithms rely on satisfaction of these constraint qualifications, so convergence is not necessarily guaranteed. Numerically, the lack of satisfaction of these constraint qualifications can be seen by severe ill-conditioning of subproblems and infeasible subproblems arbitrarily close to a solution, as consistence of linearizations is only guaranteed under MFCQ. Despite these difficulties, Fletcher and Leyffer [FL04] have shown that the nonlinear equation approach can give good results on a wide range of practical problems if active set methods are used, and that smoothing or penalty approaches are favorable if interior point methods are used.

Bard [Bar88] used a branch-and-bound approach to solve complementarity problems arising from a reformulation of a bilevel optimization problem. Ralph [Ral94] describes a stabilized Newton type method for solving the non-smooth equations using a path-generation technique in which a piecewise-linear path from one iterate to the next Newton point is traced. It has been extended by Dirkse and Ferris [DF95] with step-size selection mechanism and implemented in the PATH solver. Further extension and analysis in the framework of semi-smooth Newton methods are given by Munson et al. [Mun+01]. Outrata et al. [OKZ98] have suggested a semi-smooth Newton method using an implicit formulation of the equilibrium constraint. Luo et al. [LPR98] consider a piecewise SQP approach and show convergence of the method if multipliers are unique and Zhang and Liu [ZL01] propose an extreme point piecewise SQP algorithm. Izmailov et al. [IPS12] present a lifting approach to obtain a semi-smooth system that can be solved using a semi-smooth Newton method.

Fletcher et al. [Fle+06] have shown the following that local SQP methods in conjunction with the nonlinear equation approach can converge to a strongly stationary point if started in a neighborhood of a strong stationary point and MPEC-LICQ and further nontrivial technical assumptions are satisfied. The analysis does not extend to globalized SQP methods.

Leyffer [LLN06] analyzes formulations using nonlinear complementarity functions as defined below to formulate the equilibrium constraint and applying nonlinear

programming techniques to these *without smoothing*, gives a convergence proof to strong stationary points provided that similar, strong assumptions as given by Fletcher et al. [Fle+06] for SQP methods are satisfied.

Smoothing and Regularization Approach The smoothing and regularization approach builds upon the nonlinear equation approach by formulation the equilibrium constraint $0 \leq s \perp t \geq 0$ as $\phi(s,t) = 0$ where ϕ is a nonlinear complementarity problem (NCP) function that satisfies the axiom $\phi(a,b) = 0$ if and only if $0 \leq a \perp b \geq 0$. ϕ is then identified as the limit $\tau \to 0$ of a family of smooth functions ϕ^τ and the parametric nonlinear program with constraint $\phi^\tau(s,t) = 0$ is considered such that for $\tau \neq 0$ the nonlinear program is regular in the sense that constraint qualifications are satisfied. Then a sequence of nonlinear programs with $\tau \to 0, \tau \neq 0$ is solved or during the course of solution of one nonlinear program the parameter τ is driven to zero.

Facchinei et al. [FJQ99] have shown that this approach using a suitable NCP function guarantees convergence to C-stationary points. This analysis is complemented by Scholtes [Sch01], Ralph and Wright [RW04] giving conditions that guarantee convergence to M-, C- or B-stationary points. The requirements for B-stationarity, namely satisfaction of certain second order necessary conditions and satisfaction of so called *upper level strict complementarity* are rather restrictive and not generic. Fukushima and Pang [FP99] give an analysis that establishes convergence to B-stationary points if certain second order necessary conditions and a certain non-degeneracy assumption is satisfied.

Fukushima and Tseng [FT99] suggest an SQP method using a so called ε-feasible set and show convergence to B-stationary points if the ε-feasible sets show uniform satisfaction of LICQ.

Raghunathan and Biegler [RB03] proposed a smoothing approach in conjunction with an interior point algorithm where τ is reduced along with the barrier parameter. A similar approach has been suggested by Liu and Sun [LS04] with a different relaxation scheme that ensures convergence to weak stationary points and gives a monitoring condition for convergence to strong stationary points. Lin and Fukushima [LF03; LF05] give an expansive simplex relaxation approach and show convergence to C-stationary points. De Miguel et al. [DeM+05] considered such an approach with a two-sided relaxation on complementarity and non-negativity, solving a sequence of nonlinear programs with an interior point algorithm. Kadrani et al. [KDB09], Steffensen and Ulbrich [SU10] and Kanzow and Schwartz [KS13] suggest approaches that ensure convergence to M-stationarity points. Hatz [Hat+13] developed a lifting approach for bilevel optimization problems that ensures MPEC-LICQ of the lifted problem.

Stein [Ste12] introduces a lifting approach that gives a smooth but degenerate lifted problem which can be solved upon regularization and discusses the relationship between stationarity concepts of the equilibrium constraint problem and the lifted problem.

Hoheisel et al. [HKS13] review these different regularization concepts, improve on some of the convergence results and provide a comparison of the numerical behavior of these schemes.

Penalization Approach The penalization approach also builds on the nonlinear equation approach and considers a modified problem where the complementarity constraint is added as soft constraint in a penalty formulation to the objective function.

A penalty method with an interior point type regularization has been suggested by Luo, Pang and Ralph [LPR96, Ch. 6.1] and a convergence result has been obtained under strong assumptions. Leyffer [Ley05] has analyzed the convergence properties of this algorithm and provided a simple example in which the algorithm fails to converge to a stationary point.

Leyffer et al. [LLN06] give an interior point penalty algorithm and establish convergence to C-stationary points of this algorithm and a monitoring condition for strong stationary points that is satisfied if the penalty parameter sequence is bounded.

Fukushima et al. [FLP98] suggest using a SQP method with a penalization of a NCP formulation of the complementarity constraint and give a global convergence proof that requires a non-degeneracy assumption of the problem. Hu and Ralph [HR04] extend the analysis of smoothing approaches by Scholtes, Ralph and Wright to penalty approaches and provide conditions for convergence to B-stationary points and find similar conditions to those of the smoothing case. Jiang and Ralph [JR00] suggest a smoothing method with a SQP algorithm using an either explicit or implicit formulation and provide global convergence result under a non-degeneracy assumption. Stöhr and Scholtes [SS99; Stö00] establish the existence of an exact penalty function and show global convergence of a trust region SQP method.

Benson [Ben+06] used a penalty approach with a ℓ^∞ penalty function together with an interior point method. Anitescu [Ani05a; Ani05b; ATW07] considers a particular variant of the penalization approach to deal with the potential infeasibility of subproblems. Using a globalized SQP algorithm with an elastic mode penalty formulation of infeasible subproblems, convergence to C-stationary points can be shown.

5.6.2. Structural Treatment Approach

By structural treatment approach we understand approaches that keep complementarities without explicitly reformulating them as a more general object in which the complementarity is not longer present.

Scholtes [Sch04] proposes a SQPEC method to solve MPEC where the complementarity constraint is a structural constraint in the quadratic subproblem. However, this does not preclude convergence to spurious points as Leyffer and Munson [LM07] point out. In the example 5.15 that still exhibits descent directions in the M-stationary point $(s, t, \nu, \sigma) = (0, 0, -2, 0)$, the SQPEC method generates iterates that converge quadratically to $(s, t) = (0, 0)$.

Leyffer and Munson [LM07] propose a Filter SLPEC method and claim convergence to B-stationary points. Their preprint however gives only an outline of such a method and does not answer several questions, most importantly infeasibility handling.

Giallombardo and Ralph [GR08] consider a piecewise decomposition trust search algorithm for MPEC and establish multiplier convergence results and convergence to B-stationary points if upper level strict complementarity holds. Kirches [Kir+10a; Kir+13b] presents a SQPVC method for MPVC with a non-convex parametric algorithm for the QPVC subproblem that ensures convergence to strong stationary points of the subproblems if MPEC-LICQ is satisfied. Benko and Gfrerer [BG16] propose an SQPEC algorithm, provide an active set method to compute M-stationary points of the QPEC subproblem and show convergence to M-stationary points.

For practical implementations, it is important that structural information about complementarities is available. Modeling languages such as AMPL [FGK90; FGK02] and GAMS [GAM] have been augmented to provide formulation of equilibrium constraints as structural constraints and may also in addition provide methods to reformulate them as nonlinear programs [FFG99; FDM05].

In Chapter 7 we develop a SLPECEQP method that ensures global convergence to B-stationary points under certain assumptions.

5.7. Summary

We have introduced the challenging classes of *Mathematical Programs with Vanishing Constraints* and *Mathematical Programs with Equilibrium Constraints* and have reviewed their properties. We have noted that Mathematical Programs with Vanishing Constraints constitute a subclass of Mathematical Programs with Equilibrium Constraints and focused on the latter. Standard constraint qualifications fail for both classes and necessitate the introduction of tailored constraint qualifications and stationary concepts. This is done via a decomposition approach. Several stationarity

concepts have been proposed in the literature, of which we single out *B-stationary* and *strong stationarity* as the most interesting concepts as they prevent by definition first order descent directions. The next weaker concept of M-stationary does not prevent this, as has been shown for an example.

We have reviewed approaches to solve Mathematical Programs with Equilibrium Constraints and their convergence properties. Very few of the algorithms proposed so far guarantee convergence to B-stationary points under mild assumptions. In Chapter 7 we develop a sequential LPEC algorithm and prove that it ensures global convergence to B-stationary points under suitable assumptions.

Part II.

Contributions

6. Mixed-Integer Optimal Control Problems

In this chapter, we introduce the challenging class of Mixed-Integer Optimal Control Problems (MIOCP). Associated to a Mixed-Integer Optimal Control Problem, we formulate a relaxed convexified problem that has vanishing constraint structure. As main result, we generalize a result of Sager [Sag06] and prove that there is a correspondence between feasible points of the relaxed convexified problem and ε-feasible points of the Mixed-Integer Optimal Control Problem. We present SOS-1 respecting vanishing constraint sum-up rounding (VC-SOS-SUR) as an algorithm to computationally exploit this correspondence.

We conclude the chapter by an counterexample of Cesari [Ces83] that illustrates the results in the case of an ill-posed problem.

Parts of the results of this chapter are published in [Kir+15; KL16].

6.1. Problem Formulation

In the following we will introduce the class of Mixed-Integer Optimal Control Problems. This class of problems covers many real-world problems. Variants and examples of this class of problems have been already considered by Bock and Longman [BL80; BL82; BL85], Kaya and Noakes [KN03], Gerdts [Ger05], Sager [Sag06; SRB09; SBD12], Kirches [Kir10] and Ringkamp et al. [ROL17]. Challenges are presented in particular by the constraints $v(t) \in V$ for a discrete set V and the combinatorial constraint $c(x(t), u(t), v(t)) \geq 0$ in which the discrete control v enters. Direct discretizations of such problems constitute mixed-integer nonlinear programs, a problem class that is NP hard [GJ79]. In this chapter, we show how suboptimal solutions that approximate solutions to this problem with arbitrary small optimality and feasibility loss can be computed with a complexity equal to that of solving a continuous optimal control problem.

6.1 Definition (Mixed-Integer Optimal Control Problem).
Let $V := \{v_1, \ldots, v_{|V|}\} \subseteq \mathbb{R}^{n_v}$ be a finite, discrete set of choices. Let $D_x \subseteq \mathbb{R}^{n_x}$, $D_u \subseteq \mathbb{R}^{n_u}$ be domains. Let $\phi : D_x \to \mathbb{R}$ be a continuously differentiable function and

$f : D_x \times D_u \times V \to \mathbb{R}^{n_x}$, $d : D_x \times D_u \to \mathbb{R}^{n_d}$ and $c : D_x \times D_u \times V \to \mathbb{R}^{n_v}$ be continuous functions.

Then the following problem denotes a *mixed-integer optimal control problem*:

$$\begin{aligned} \min_{\substack{x \in W^{1,\infty}([0,1],\mathbb{R}^{n_x}),\\ u \in L^\infty([0,1],\mathbb{R}^{n_u}),\\ v \in L^\infty([0,1],\mathbb{R}^{n_v})}} \quad & \phi(x(1)) \\ \text{s.t.} \quad & \dot{x}(t) = f(x(t),u(t),v(t)) && \text{a.e. } t \in [0,1], \\ & x(0) = x^0, \\ & v(t) \in V && \text{a.e. } t \in [0,1], \\ & 0 \le d(x(t),u(t)) && \text{a.e. } t \in [0,1], \\ & 0 \le c(x(t),u(t),v(t)) && \text{a.e. } t \in [0,1]. \end{aligned} \tag{MIOCP}$$

△

It is well known that more general classes of problems can be reduced to this class of problems, namely problems with free endtime, with non-autonomous dynamics and problems with Bolza type objective function that are a sum of a Mayer type final time objective contribution and a Lagrange type objective contribution. Furthermore, problems with point constraints, multi-stage problems and problems with additional parameter dependence may be considered. These additional ingredients do not contribute to the arguments of this chapter and we restrict our consideration to the defined problem class for notational simplicity.

An *admissible* or *feasible* point of (MIOCP) is a point that satisfies all the constraints:

6.2 Definition (Admissible Point).
Functions $x \in W^{1,\infty}([0,1],\mathbb{R}^{n_x})$, $u \in L^\infty([0,1],\mathbb{R}^{n_u})$ and $v \in L^\infty([0,1],\mathbb{R}^{n_v})$ are *admissible*, if

(1) $x(t) \in D_x$ for all $t \in [0,1]$ and $u(t) \in D_u$ and $v(t) \in V$ for almost all $t \in [0,1]$,

(2) $\dot{x}(t) = f(x(t),u(t),v(t))$ for almost all $t \in [0,1]$,

(3) $x(0) = x^0$,

(4) $0 \le d(x(t),u(t))$ and $0 \le c(x(t),u(t),v(t))$ for almost all $t \in [0,1]$. △

6.3 Definition (Minimum and Local Minimum).
An admissible tuple (x^*,u^*,v^*) is a *minimizer* if $\phi(x^*(1)) \le \phi(x(1))$ for all admissible points (x,u,v).

It is a *strong local minimizer*, if there is $\varepsilon > 0$ such that $\phi(x^*(1)) \le \phi(x(1))$ for all admissible points (x,u,v) that satisfy $\|x - x^*\|_{L^\infty} \le \varepsilon$.

It is a *weak local minimizer*, if there is $\varepsilon > 0$ such that $\phi(x^*(1)) \le \phi(x(1))$ for all admissible points (x,u,v) that satisfy $\|x - x^*\|_{W^{1,\infty}} \le \varepsilon$ and $\|u - u^*\|_{L^\infty} \le \varepsilon$ and $\|v - v^*\|_{L^\infty} \le \varepsilon$. △

It is evident, that every weak local minimizer is also a strong local minimizer. Strong local minimizers are local minimizers if the state space $W^{1,\infty}$ is endowed with the topology induced by the L^∞-norm and the control spaces are equipped with the trivial topology. The notion of strong and weak local minimizer plays an important role in the calculus of variations and the analysis of the maximum principle and is mentioned for these reasons. In this thesis, we focus on strong local minimizers and will for brevity use local minimizer as synonym for strong local minimizer.

We assume the following regularity of the data:

6.4 Assumption.

L. The mappings $f(\cdot, u, v)$ and $c(\cdot, u, v)$ are uniformly Lipschitz continuous for every $u \in D_u$ and $v \in V$, i.e., there exists $L_f > 0$ and $L_c > 0$ such that for all $x, \hat{x} \in D_x$, $u \in D_u$ and $v \in V$:

$$\begin{aligned} \|f(x,u,v) - f(\hat{x},u,v)\| &\leq L_f \|x - \hat{x}\|, \\ \|c(x,u,v) - c(\hat{x},u,v)\| &\leq L_c \|x - \hat{x}\|. \end{aligned}$$ △

We will need the Lipschitz continuity assumption on f for the proof of Theorems 6.7 and 6.12, while the Lipschitz continuity assumption on c is only required to provide the existence of an ε-feasible grid for the reconstruction via a rounding scheme in Section 6.4.

6.2. Convexification and Relaxation

We introduce now a *convexified* problem associated to (MIOCP) that provides a reformulation of (MIOCP) suitable for subsequent relaxation, such that solutions of this relaxed problem can be approximated arbitrary well by binary feasible solutions. This convexification and relaxation approach is similar to the notion of *generalized curves* introduced by L.C. Young [You37] to study existence questions in the calculus of variations and to the notion of *relaxed controls* used by Cesari and Berkovitz to obtain existence results for optimal control problems, see [Ces83, Ch. 18], [Ber74] and [BM12, Ch. 3]. It is a generalization of the results of Sager [Sag06; SBD12] who studied the case of (MIOCP) without combinatorial constraints of the type $c(x(t), u(t), v(t)) \geq 0$ that depend on discrete controls. Sager used this convexification approach to proof a conjecture of Veliov [Vel03; Vel05] concerning the Hausdorff-distance of reachable sets.

6.5 Definition (Convexification and relaxation).
The *partial outer convexification* of (MIOCP) is given by

$$\begin{aligned}
\min_{\substack{x\in W^{1,\infty}([0,1],\mathbb{R}^{n_x}),\\ u\in L^\infty([0,1],\mathbb{R}^{n_u}),\\ \omega\in L^\infty([0,1],\mathbb{R}^{|V|})}} \quad & \phi(x(1)) \\
\text{s.t.} \quad & \dot{x}(t) = \textstyle\sum_{i\in[|V|]} \omega_i(t) f(x(t),u(t),v_i) && \text{a.e. } t\in[0,1], \\
& x(0) = x^0, && \\
& \omega(t)\in\{0,1\}^{|V|} && \text{a.e. } t\in[0,1], \\
& 1 = \textstyle\sum_{i\in[|V|]} \omega_i(t) && \text{a.e. } t\in[0,1], \\
& 0 \le d(x(t),u(t)) && \text{a.e. } t\in[0,1], \\
& 0 \le \omega_i(t) c(x(t),u(t),v_i),\ i\in[|V|] && \text{a.e. } t\in[0,1].
\end{aligned} \tag{BC}$$

The *relaxed partial outer convexification* of (MIOCP) is given by

$$\begin{aligned}
\min_{\substack{x\in W^{1,\infty}([0,1],\mathbb{R}^{n_x}),\\ u\in L^\infty([0,1],\mathbb{R}^{n_u}),\\ \alpha\in L^\infty([0,1],\mathbb{R}^{|V|})}} \quad & \phi(x(1)) \\
\text{s.t.} \quad & \dot{x}(t) = \textstyle\sum_{i\in[|V|]} \alpha_i(t) f(x(t),u(t),v_i) && \text{a.e. } t\in[0,1], \\
& x(0) = x^0, && \\
& \alpha(t)\in[0,1]^{|V|} && \text{a.e. } t\in[0,1], \\
& 1 = \textstyle\sum_{i\in[|V|]} \alpha_i(t) && \text{a.e. } t\in[0,1], \\
& 0 \le d(x(t),u(t)) && \text{a.e. } t\in[0,1], \\
& 0 \le \alpha_i(t) c(x(t),u(t),v_i),\ i\in[|V|] && \text{a.e. } t\in[0,1].
\end{aligned} \tag{RC}$$

△

The binary convexified problem (BC) is equivalent to (MIOCP) in the following sense:

6.6 Proposition.
Problem (MIOCP) *has a solution if and only if* (BC) *has a solution.*

If $(x_B^*, u_B^*, \omega_B^*)$ *is a solution of* (BC)*, then* (x^*, u^*, v^*) *is a solution of* (MIOCP)*, where* $x^* = x_B^*$, $u^* = u_B^*$ *and* v^* *is defined by*

$$v^*(t) := \sum_{i\in[|V|]} \omega_i(t) v_i.$$

Proof. The mapping

$$L^\infty([0,1],\{0,1\}^{|V|}) \to L^\infty([0,1],V), \omega \mapsto v(t) := \sum_{i\in[|V|]} \omega_i(t) v_i$$

defines a bijection between the subset $\{\omega \in L^\infty([0,1],\{0,1\}^{|V|} \mid \sum_{i\in[|V|]} \omega_i(t) = 1\}$ of $L^\infty([0,1],\{0,1\}^{|V|})$ and $L^\infty([0,1],V)$ and preserves the objective function value. □

The *relaxed partial outer convexification* arises by relaxing the integrality constraint $\omega_i \in \{0,1\}$ to $\alpha_i \in [0,1]$. The constraints $\omega_i(t)c(x(t),u(t),v_i) \geq 0$ in (BC) and $\alpha_i(t)c(x(t),u(t),v_i) \geq 0$ in (RC) are *vanishing constraints*: If $\omega_i(t) = 0$ or $\alpha_i(t) = 0$, they are satisfied regardless of c, whereas the constraint $c(x(t),u(t),v_i) \geq 0$ has to be taken into account if $\omega_i(t) > 0$ or $\alpha_i(t) > 0$.

In contrast to *outer convexification* are *inner convexification* approaches, where instead of the linear convex combination $\dot{x}(t) = \sum_i \omega_i(t) f(x(t),u(t),v_i)$ the nonlinear expression $\dot{x}(t) = f(x(t),u(t),\sum_i \omega_i(t)v_i)$ is used. Jung et al. [JKS13; Jun13] compared these approaches and found that the outer convexification approach is superior as it yields tighter relaxations and does not require $f(x(t),u(t),\cdot)$ to be defined on the convex hull of V.

6.3. Relation between Mixed-Integer and Relaxed problem

We now get to the main result of the chapter, which states that for a feasible point of (RC) there is an essentially feasible point of (BC) with essentially the same objective function value:

6.7 Theorem (Zero Integrality Gap in Function Space).
Let $(\overline{x},u,\overline{\alpha})$ be feasible for (RC) and suppose that $t \mapsto f(\overline{x}(t),u(t),v_i), i \in [|V|]$ are $W^{1,\infty}$ functions. Let $\varepsilon > 0$.

Then there are functions $x^\varepsilon \in W^{1,\infty}([0,1],\mathbb{R}^{n_x})$ and $\omega^\varepsilon \in L^\infty([0,1],\{0,1\}^{|V|})$ such that

$$|\phi(x^\varepsilon(1)) - \phi(\overline{x})| < \varepsilon$$

and

$$\begin{aligned}
\dot{x}^\varepsilon(t) &= \textstyle\sum_{i\in[|V|]} \omega_i^\varepsilon(t) f(x^\varepsilon(t),u(t),v_i) && \text{a.e. } t \in [0,1],\\
x^\varepsilon(0) &= x^0,\\
1 &= \textstyle\sum_{i\in[|V|]} \omega_i^\varepsilon(t) && \text{a.e. } t \in [0,1],\\
-\varepsilon &\leq d(x^\varepsilon(t),u(t)) && \text{a.e. } t \in [0,1],\\
-\varepsilon &\leq \omega_i^\varepsilon(t) c(x^\varepsilon(t),u(t),v_i), i \in [|V|] && \text{a.e. } t \in [0,1].
\end{aligned}$$

△

This Theorem shows that every feasible point of the relaxed problem can be approximated arbitrarily well by a binary feasible point. The Theorem and its proof are not constructive and the binary feasible point depends on the prescribed accuracy $\varepsilon > 0$. We will later on show how to construct such a point.

We defer the proof to the end of the section, as some preparatory results are required. Grönwall's Lemma, the Banach-Alaoglu and the Krein-Milman Theorem will be utilized, which we restate for convenience:

6.8 Lemma (Grönwall, [Grö19], [Cla13, Thm 6.41]).
Let $x \in W^{1,\infty}([0,1],\mathbb{R}^n)$ such that there are $\gamma, \beta \in L^1([0,1])$ with $\gamma(t) \geq 0$, a.e. $t \in [0,1]$ and

$$\|\dot{x}(t)\| \leq \gamma(t)\|x(t)\| + \beta(t) \quad \text{a.e } t \in [0,1].$$

Then the following estimate holds for all $t \in [0,1]$:

$$\|x(t) - x(0)\| \leq \int_0^t (\gamma(s)\|x(0)\| + \beta(s)) e^{\int_s^t \gamma(\tau)\,\mathrm{d}\tau}\,\mathrm{d}s.$$

△

6.9 Theorem (Banach-Alaoglu, [Ban32; Ala40; Bou38], [Cla13, Cor. 3.15]).
Let X be a normed space. If $K \subseteq X'$ is bounded and closed with respect to the weak * topology, then K is compact with respect to the weak * topology. △

For a definition of the weak * topology, we refer to [Cla13, Ch. 3.3].

6.10 Definition (Extreme Point).
Let X be a normed space and $K \subseteq X$ a non-empty subset. A point $x \in K$ is an *extreme point of* K, if x cannot be written as proper convex combination of elements in K, i.e., $\tau y + (1-\tau)y' \neq x$ for all $y, y' \in K$ and $\tau \in (0,1)$. △

6.11 Theorem (Krein-Milman, [KM40], [Cla13, Thm 8.56]).
Let X be a normed space, $K \subseteq X$ a non-empty compact convex subset of X. Let E be the set of extreme points of K.

Then $K = \overline{\mathrm{co}}\, E$. △

Using Grönwall's Lemma, the influence of control perturbations on the relaxed control α can be bounded.

6.12 Theorem (Influence of Control Perturbation).
Let $x, y \in W^{1,\infty}([0,1],\mathbb{R}^{n_x})$, $\alpha, \beta \in L^\infty([0,1],[0,1]^{|V|})$ and $u \in L^\infty([0,1],\mathbb{R}^{n_u})$ such that

$$\begin{aligned}
\dot{x}(t) &= \textstyle\sum_{i\in[|V|]} \alpha_i(t) f(x(t),u(t),v_i) && \text{a.e. } t \in [0,1],\\
\dot{y}(t) &= \textstyle\sum_{i\in[|V|]} \beta_i(t) f(y(t),u(t),v_i) && \text{a.e. } t \in [0,1],\\
x(0) &= x^0,\\
y(0) &= x^0 && .
\end{aligned}$$

(1) If there exists $\delta_f \in L^1([0,1],\mathbb{R})$ with

$$\left\| \int_0^t \sum_{i\in[|V|]} (\alpha_i(\tau) - \beta_i(\tau)) f(x(\tau),u(\tau),v_i)\,\mathrm{d}\tau \right\| \leq \delta_f(t) \quad \text{a.e. } t \in [0,1], \tag{6.1}$$

then $\|x(t) - y(t)\| \leq \delta_f(t) e^{L_f t}$ for all $t \in [0,1]$.

(2) *Assume that* $t \mapsto f(x(t), u(t), v_i), i \in [|V|]$ *are* $W^{1,\infty}$ *functions. Define for* $j = 0, 1$*:*

$$M_j := \operatorname*{ess\,sup}_{t\in[0,1]} \left\| \frac{\mathrm{d}^j}{\mathrm{d}t^j} \sum_{i\in[|V|]} f(x(t), u(t), v_i) \right\|.$$

If there is $\delta > 0$ *such that*

$$\left\| \int_0^t (\alpha(\tau) - \beta(\tau)) \,\mathrm{d}\tau \right\| \leq \delta \quad \text{a.e. } t \in [0,1], \tag{6.2}$$

then $\|x(t) - y(t)\| \leq \delta(M_0 + tM_1)e^{L_f t}$ a.e. $t \in [0,1]$.

Proof. To ease notation, let $f_i(t,x) := f(x, u(t), v_i)$.

(1) By Lipschitz-continuity assumption L and $\sum \beta_i = 1$ the following estimate follows:

$$\begin{aligned} \left\| \sum_{i\in[|V|]} \beta_i(\tau)(f_i(\tau, x(\tau)) - f_i(\tau, y(\tau))) \right\| &\leq \sum_{i\in[|V|]} |\beta_i(\tau)| \, \|f_i(\tau, x(\tau)) - f_i(\tau, y(\tau))\| \\ &\leq \sum_{i\in[|V|]} L_f \|x(\tau) - y(\tau)\| \end{aligned}$$

It follows for $t \in [0,1]$:

$$\begin{aligned} \|x(t) - y(t)\| &= \left\| \int_0^t \sum_{i\in[|V|]} (\alpha_i(\tau) f_i(\tau, x(\tau)) - \beta_i(\tau) f_i(\tau, y(\tau))) \,\mathrm{d}\tau \right\| \\ &= \left\| \int_0^t \sum_{i\in[|V|]} (\alpha_i(\tau) f_i(\tau, x(\tau)) - \beta_i(\tau) f_i(\tau, x(\tau)) \right. \\ &\qquad \left. \beta_i(\tau) f_i(\tau, x(\tau)) - \beta_i(\tau) f_i(\tau, y(\tau))) \,\mathrm{d}\tau \right\| \\ &\leq \delta_f(t) + L_f \left\| \int_0^t x(\tau) - y(\tau) \,\mathrm{d}\tau \right\|. \end{aligned}$$

The result follows by applying Grönwall's Lemma 6.8 to $\|x(t) - y(t)\|$.

(2) Set $\varphi := \alpha - \beta$, let $t \in [0,1]$. By partial integration we find

$$\begin{aligned}
&\left\| \int_0^t \sum_{i\in[|V|]} \varphi_i(\tau) f_i(\tau, x(\tau))\,\mathrm{d}\tau \right\| \\
&\quad \le \left\| \sum_{i\in[|V|]} \Big[f_i(t,x(t)) \int_0^t \varphi_i(\tau)\,\mathrm{d}\tau - \int_0^t \tfrac{\mathrm{d}}{\mathrm{d}\tau} f_i(\tau,x(\tau)) \int_0^\tau \varphi_i(s)\,\mathrm{d}s \Big]\,\mathrm{d}\tau \right\| \\
&\quad \le (M_0 + tM_1)\delta.
\end{aligned}$$

Using (1) with $\delta_f(t) := \delta(M_0 + tM_1)$ concludes the proof. □

6.13 Definition.
Let $(\overline{x}, u, \overline{\alpha})$ be feasible for (RC). Then define the sets Γ and Γ_N for $N \in \mathbb{N}$ by

$$\Gamma := \left\{ \alpha \in L^\infty([0,1], \mathbb{R}^{|V|}) \,\middle|\, \begin{smallmatrix} 1=\sum_{i\in[|V|]} \alpha_i(t) \text{ and} \\ 0\le \alpha_i(t) c_i(\overline{x}(t), u(t), v_i),\ i\in[|V|] \end{smallmatrix} \text{ a.e. } t \in [0,1] \right\},$$

$$\Gamma_N := \left\{ \alpha \in \Gamma \,\middle|\, 0 = \int_{k/N}^{(k+1)/N} \sum_{i\in[|V|]} (\alpha_i(\tau) - \overline{\alpha}_i(\tau)) f(\overline{x}(\tau), u(\tau), v_i)\,\mathrm{d}\tau,\ k \in [N] \right\}.$$

△

6.14 Lemma (weak * compactness of Γ, Γ_N).
*The sets Γ and Γ_N for all $N \in \mathbb{N}$ are $L^1([0,1], \mathbb{R}^{|V|})$-weakly * compact, i.e. compact in the weak * topology on $L^\infty([0,1], \mathbb{R}^{|V|})$.*

Proof. We consider $L^1([0,1], (\mathbb{R}^{|V|})')'$ equipped with the weak * topology and the isometric isomorphism

$$L^\infty([0,1], \mathbb{R}^{|V|}) \to L^1([0,1], (\mathbb{R}^{|V|})')', \alpha \mapsto \left(\beta \mapsto \int_0^1 \langle \alpha(\tau), \beta(\tau) \rangle\,\mathrm{d}\tau \right)$$

to define the weak * topology on $L^\infty([0,1], \mathbb{R}^{|V|})$. Then Γ and Γ_N are closed in the weak * topology. As $\Gamma_N \subseteq \Gamma \subseteq \{\alpha \in L^\infty([0,1], \mathbb{R}^{|V|}) \mid \|\alpha\|_\infty \le 1\}$, Γ is mapped by this isometric isomorphism to a subset of the unit ball in $L^1([0,1], (\mathbb{R}^{|V|})')'$. The latter is compact in the weak * topology by the Banach-Alaoglu Theorem 6.9. Weak * compactness of Γ and Γ_N follows now from the fact that Γ and Γ_N are closed subsets.□

6.15 Lemma (Extremal points are binary feasible).
Let $\alpha \in \Gamma_N$ be an extremal point for $N \in \mathbb{N}$. Then $\alpha(t) \in \{0,1\}^{|V|}$ for a.e. $t \in [0,1]$.

Proof. Assume to the converse, that this is not the case. Then there is a constant $0 < \delta < \frac{1}{2}$ and disjoint indices $i_1, i_2 \in [|V|]$ such that $\{t \in [0,1] \,|\, \delta < \alpha_i(t) < 1-\delta \text{ for } i = i_1, i_2\}$ has positive measure. In particular, there is $0 \le j \le N$ such that $T_j := \{t \in (j/N, (j+1)/N) \,|\, \delta < \alpha_i(t) < 1-\delta \text{ for } i = i_1, i_2\}$ has positive measure. Let $T_j = \dot{\bigcup}_{\ell \in [n_x+1]} T_\ell$ be a finite partition of T_j into $n_x + 1$ disjoint subsets of positive measure.
Define for $\ell \in [|V|]$ the function $\beta^\ell \in L^\infty([0,1], \mathbb{R}^{|V|})$ by

$$\beta_k^\ell(t) := \begin{cases} 0, & t \notin T_\ell \text{ or } k \neq i_1, i_2, \\ \delta, & t \in T_\ell \text{ and } k = i_1, \\ -\delta, & t \in T_\ell \text{ and } k = i_2. \end{cases} \tag{6.3}$$

If $\gamma \in [-1,1]^{n_x+1}$, let $\beta(\gamma) \in L^\infty([0,1], \mathbb{R}^{|V|})$ be defined by the linear combination $\beta(\gamma) := \sum_{\ell \in [n_x+1]} \gamma_\ell \beta^\ell$.
Then

- $\alpha(t) + \beta(\gamma)(t) \in [0,1]^{|V|}$ for a.e. $t \in [0,1]$ by definition of β^ℓ;
- $\sum_{i \in [|V|]} \beta_i(t) = 0$ by construction, thus $\sum_{i \in [|V|]} (\alpha_i(t) + \beta(\gamma)_i(t)) = 1$ for a.e. $t \in [0,1]$;
- If $\alpha_i(t) + \beta(\gamma)_i(t) > 0$ the implication $\alpha_i(t) > 0$ holds as $\beta_i^\ell(t) = 0$ if $\alpha_i(t) = 0$ by construction;

which proves $\alpha + \beta(\gamma) \in \Gamma$. Furthermore, $\alpha - \beta(\gamma) = \alpha + \beta(-\gamma) \in \Gamma$.
We show that there is $\gamma \in [-1,1]^{n_x+1}, \gamma \neq 0$ with $\alpha + \beta(\gamma) \in \Gamma_N$, which then shows $\alpha = \frac{1}{2}(\alpha - \beta(\gamma)) + \frac{1}{2}(\alpha + \beta(\gamma))$ that α is a proper convex combination of points in Γ_N in contradiction to α being an extreme point.
To find such a γ, we note $\alpha + \beta(\gamma) \in \Gamma_N$ if and only if

$$\begin{aligned} 0 &= \int_{j/N}^{(j+1)/N} \sum_{i \in [|V|]} \beta(\gamma)_i(\tau) f(x(\tau), u(\tau), v_i) \, \mathrm{d}\tau \\ &= \sum_{\ell \in [n_x+1]} \left(\int_{j/N}^{(j+1)/N} \sum_{i \in [|V|]} \beta_i^\ell(\tau) f(x(\tau), u(\tau), v_i) \, \mathrm{d}\tau \right) \gamma_\ell . \end{aligned}$$

This condition is equivalent for γ to satisfy a consistent underdetermined linear system, which has nontrivial solution $0 \neq \tilde{\gamma}$. Rescaling yields a desired $\gamma := \frac{1}{\|\tilde{\gamma}\|_\infty} \tilde{\gamma}$.□

We are now empowered with the tools to prove Theorem 6.7:

PROOF. Γ_N is convex, non-empty since $\overline{\alpha} \in \Gamma_N$ and compact in the weak * topology on $L^\infty([0,1], \mathbb{R}^{|V|})$ by Lemma 6.14.

Application of the Krein-Milman Theorem 6.11 yields an extremal point ω^N which then is binary feasible by 6.15.

Define $x^N \in W^{1,\infty}([0,1], \mathbb{R}^{n_x})$ as solution of the initial value problem

$$\begin{cases} \dot{x}^N(t) = \sum_{i \in [|V|]} \alpha_i(t) f(x^N(t), u(t), v_i) & \text{a.e. } t \in [0,1] \\ x(0) = x^0. \end{cases}$$

Let $M_j := \operatorname{ess\,sup}_{t \in [0,1]} \| \frac{\mathrm{d}^j}{\mathrm{d}t^j} \sum_{i \in [|V|]} f(\overline{x}(t), u(t), v_i) \|$ for $j = 0, 1$. By definition of Γ_N, the estimate

$$\begin{aligned} &\left\| \int_0^t \sum_{i \in [|V|]} (\omega_i(\tau) - \overline{\alpha}_i(\tau)) f(\overline{x}(\tau), u(\tau), v_i) \, \mathrm{d}\tau \right\| \\ &\quad = \left\| \int_{\lfloor t/N \rfloor}^t \sum_{i \in [|V|]} (\omega_i(\tau) - \overline{\alpha}_i(\tau)) f(\overline{x}(\tau), u(\tau), v_i) \, \mathrm{d}\tau \right\| \leq \tfrac{M_0}{N} \end{aligned}$$

holds. Thus by 6.12, (2) $\|x^N(t) - \overline{x}(t)\| \leq \frac{M_0}{N}(M_0 + tM_1)e^{L_f t}$.

By continuity of ϕ, d and c the result follows with $\omega^\varepsilon = \omega^N$ and $x^\varepsilon = x^N$, provided N is large enough. □

6.4. Rounding Scheme

Theorem 6.7 asserts that for a feasible point of the relaxed problem there is a point of the binary problem that is close in optimality and feasibility. It is, however, not constructive and thus the question how to practically reconstruct a binary solution from a relaxed solution arises. Theorem 6.12 gives a hint: If α is the relaxed control of a feasible point of the relaxed problem and ω is chosen such that $\left\| \int_0^t (\alpha(\tau) - \omega(\tau)) \, \mathrm{d}\tau \right\|$ is small *for all* $t \in [0,1]$, then the trajectories associated with α and ω are also close. The difficulty remains in staying feasible with respect to the combinatorial constraints $0 \leq \omega_i(t) c(x(t), u(t), v_i)$ for all i. Sager [Sag06; SBD12] has developed a reconstruction algorithm, Sum-Up Rounding in the absence of combinatorial constraints, which has linear complexity in the size of the temporal grid. Jung [Jun13] established an algorithm, Next-Forced Rounding, that provides improved approximation properties but is anticipative and has quadratic complexity in the size of the temporal grid.

We present a novel rounding scheme addressing the case of combinatorial constraints that extends Sum-Up Rounding for problems without combinatorial constraint.

First we introduce the notion of ε-feasible grids that allows us to stay feasible even after rounding:

6.16 Definition (ε-feasibility).
Let (x, u, α) be feasible for (RC), $\varepsilon > 0$ an acceptable constraint violation.

A temporal grid $0 = t_0 < \ldots < t_N = 1$ is *ε-feasible*, if for every $\xi \in L^\infty([0,1], \mathbb{R}^{n_x})$ with $\|\xi(t) - x(t)\| \leq \varepsilon$, a.e. $t \in [0,1]$ the following implication holds:

If $\int_{t_j}^{t_{j+1}} \alpha_i(t)\,\mathrm{d}t > 0$, also $c(\xi(t), u(t), v_i) \geq -\varepsilon$, a.e. $t \in [t_j, t_{j+1}]$. △

6.17 Lemma.
Let (x, u, α) be feasible for (RC) *and $\varepsilon > 0$ an acceptable constraint violation.*

Then there exists an ε-feasible grid.

Proof. Follows from Lipschitz continuity assumption on c. □

Next we introduce the notion of a Vanishing Constraint convergent algorithm:

6.18 Definition.
An algorithm that is defined for inputs

- a function $\alpha \in L^\infty([0,1], \mathbb{R}^{|V|})$ such that $\sum_{j \in [|V|]} \alpha_j(t) = 1$, a.e. $t \in [0,1]$,
- a temporal grid $0 = t_0 < \ldots < t_N = 1$

and outputs a function $\omega \in L^\infty([0,1], \mathbb{R}^{|V|})$ such that $\sum_{j \in [|V|]} \omega_j(t) = 1$, a.e. $t \in [0,1]$ is called *Vanishing Constraint convergent* if there exists a constant $C > 0$ such that

$$\int_{t_i}^{t_{i+1}} \alpha_j(t)\,\mathrm{d}t = 0 \quad \Rightarrow \quad \omega_j(t) = 0 \text{ a.e. } t \in (t_i, t_{i+1}), \tag{6.4}$$

$$\sup_{t \in [0,1]} \left\| \int_0^t (\alpha(\tau) - \omega(\tau))\,\mathrm{d}\tau \right\|_\infty \leq C\overline{\Delta}, \quad \overline{\Delta} := \max(t_{i+1} - t_i). \tag{6.5}$$

△

The first requirement is needed to ensure the combinatorial constraint. Without this feasibility requirement it cannot be guaranteed that $c(x(t), u(t), \sum_{j \in [|V|]} \omega_j(t) v_j) \geq 0$ for the case $\alpha(t) = 0$, as only $\alpha(t) c_i(x(t), u(t), v_i) \geq 0$ holds in the solution of (RC). The second requirement ensures applicability of Theorem 6.12.

6.19 Proposition.
Let (x, u, α) be feasible for (RC) *and $\varepsilon > 0$ an acceptable constraint violation.*

Assume that a Vanishing Constraint convergent algorithm exists.

Then there exists an ε-feasible temporal grid $0 = t_0 < \ldots < t_N = 1$ such that application of the algorithm with input α and $t_0, \ldots, t_N$ yields a point $(x^\varepsilon, u, \omega^\varepsilon)$ that satisfies the conclusion of Theorem 6.7.

Proof. By Lemma 6.17 there exists an ε-feasible grid. Using a refinement of this grid that ensures that $\sup_{t\in[0,1]} \left\| \int_0^t (\alpha(\tau) - \omega(\tau))\,\mathrm{d}\tau \right\|$ is sufficiently small, the existence of $(x^\varepsilon, u, \omega^\varepsilon)$ that satisfies the conclusion of Theorem 6.7 can be concluded as in the proof of Theorem 6.7 with the exception of the feasibility of $\omega_i^\varepsilon(t)c(x^\varepsilon(t), u(t), v_i) \geq -\varepsilon$. This is satisfied by ε-feasibility of the grid and the property (6.4) on the algorithm. □

The Sum-Up Rounding Scheme of Sager is not Vanishing Constraint convergent as the following example shows:

6.20 Example.
Consider $\alpha \in L^\infty([0,1], \mathbb{R}^2)$ defined by

$$\alpha(t) = \begin{pmatrix} \frac{3}{5} \\ \frac{2}{5} \end{pmatrix} [t \leq \tfrac{2}{3}] + \begin{pmatrix} 1 \\ 0 \end{pmatrix} [t > \tfrac{2}{3}].$$

Application of the Sum-Up Rounding Scheme of Sager applied on the grid $0, \frac{2}{3}, 1$ yields

$$\omega^{\mathrm{SUR}} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} [t \leq \tfrac{2}{3}] + \begin{pmatrix} 0 \\ 1 \end{pmatrix} [t > \tfrac{2}{3}].$$

△

This motivates our definition of the Vanishing Constraint SOS-Sum-Up Rounding scheme:

6.21 Definition.
Let $0 = t_0 < \ldots < t_N = 1$ be a temporal grid.

The *Vanishing Constraint SOS-Sum-Up Rounding Scheme* is defined recursively by $\omega^{\mathrm{VC}}|(t_i, t_{i+1}) := (\omega_j^i)_{j\in[|V|]}$ with

$$\omega_j^i := \left[j = \arg \max_{\substack{k\in[|V|], \\ \int_{t_i}^{t_{i+1}} \alpha_k(t)\,\mathrm{d}t > 0}} \int_0^{t_{i+1}} \alpha_k(t)\,\mathrm{d}t - \int_0^{t_i} \omega_k^{\mathrm{VC}}(t)\,\mathrm{d}t \right]. \qquad \text{(VC-SOS-SUR)}$$

If the maximum in (VC-SOS-SUR) is attained for several indices k, exactly one has to be chosen by arg max. △

The rounding scheme (VC-SOS-SUR) differs from the Sum-Up Rounding scheme of Sager by the addition of the feasibility requirement $\int_{t_i}^{t_{i+1}} \alpha_k(t)\,\mathrm{d}t > 0$ in the selection of the index. For the Example 6.20, (VC-SOS-SUR) yields

$$\omega^{\mathrm{VC}} \equiv \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

which satisfies the feasibility requirement.

The Vanishing Constraint SOS-Sum-Up Rounding Scheme satisfies the feasibility property while maintaining the favorable properties of the Sum-Up Rounding Scheme, namely preservation of the Special Ordered Set (SOS) property and being computationally cheap:

6.22 Proposition.
Let (x, u, α) be feasible for (RC), $0 = t_0 < \ldots < t_N = 1$ *be a ε-feasible grid for an acceptable constraint violation $\varepsilon > 0$.*

Let ω^{VC} be defined by (VC-SOS-SUR). *Then*

(1) $\int_{t_i}^{t_{i+1}} \alpha_j(t)\,\mathrm{d}t = 0 \quad \Rightarrow \quad \omega_j^{\mathrm{VC}}(t) = 0$ a.e. $t \in (t_i, t_{i+1})$,

(2) the Special Ordered Set Property $\sum_{j\in[|V|]} \omega_j^{\mathrm{VC}} = 1$, a.e. $t \in [0,1]$ *holds and*

(3) the computational complexity to evaluate ω^{VC} is $O(N)$.

Proof. Satisfaction of the implication $\int_{t_i}^{t_{i+1}} \alpha_j(t)\,\mathrm{d}t = 0 \quad \Rightarrow \quad \omega_j^{\mathrm{VC}}(t) = 0$ a.e. $t \in (t_i, t_{i+1})$ and of the Special Ordered Set Property follow immediately by definition.

The recursive definition of ω^{VC} provides an algorithm to evaluate ω^{VC} in N steps which implies the last assertion. □

We conjecture that (VC-SOS-SUR) is a Vanishing Constraint convergent algorithm and thus Prop. 6.19 holds with (VC-SOS-SUR). However, a proof for the existence of $C > 0$ such that $\sup_{t\in[0,1]} \left\| \int_0^t (\alpha(\tau) - \omega^{\mathrm{VC}}(\tau))\,\mathrm{d}\tau \right\|_\infty \leq C\overline{\Delta}$ has not been found so far as analysis of (VC-SOS-SUR) is challenging due to the requirement $\int_{t_i}^{t_{i+1}} \alpha_j(t)\,\mathrm{d}t > 0$ for rounding up. Numerical investigations provide overwhelming evidence for the following conjecture:

6.23 Conjecture.
The Vanishing Constraint SOS-Sum-Up Rounding Scheme (VC-SOS-SUR) satisfies

$$\sup_{t\in[0,1]} \left\| \int_0^t (\alpha(\tau) - \omega^{\mathrm{VC}}(\tau))\,\mathrm{d}\tau \right\|_\infty \leq \tfrac{1}{2}(|V| - 1)\overline{\Delta}.$$

In particular, VC-SOS-SUR is a Vanishing Constraint convergent algorithm. △

We assume the truth of this conjecture and use (VC-SOS-SUR) as Vanishing Constraint convergent algorithm to obtain a reconstruction algorithm as stated in Prop. 6.19.

6.5. Cesari's Example: An Ill-Posed Problem

It is important to notice the direction of the result given by the Approximation Theorem 6.7:

Any feasible point of the relaxed problem can be approximated arbitrary well by a binary one. Cesari [Ces83] provided an example where one binary feasible point and infinitely many relaxed feasible points exist, with different objective function values:

6.24 Example (Cesari, [Ces83, Ch. 18.7]).
Let parameters $0 < c \leq \frac{1}{8}$ and $0 < \sigma < 1$ be given, and consider the following problem,

$$\begin{aligned} \min_{x,u,v} \quad & \int_0^1 1 - 2|v(t) - \tfrac{1}{2}| \, \mathrm{d}t \\ \text{s.t.} \quad & \dot{x}(t) = (\tfrac{v(t)-x_1(t)}{\sigma+t}, u(t), (x_1(t) - x_2(t))^2)^T && \text{a.e. } t \in [0,1], \\ & x(0) = (\tfrac{1}{2}, \tfrac{1}{2}, 0)^T, \\ & x_3(1) = 0, \\ & u(t) \in [-c, c] && \text{a.e. } t \in [0,1], \\ & v(t) \in \{0, \tfrac{1}{2}, 1\} && \text{a.e. } t \in [0,1], \end{aligned} \tag{6.6}$$

wherein the function $x \in W^{1,\infty}([0,1], \mathbb{R}^3)$ is assumed to be absolutely continuous and $u, v \in L^\infty([0,1], \mathbb{R})$ measurable. △

It can be seen that, due to the terminal constraint and the growth condition imposed by $|u(t)| \leq c$, this MIOCP has only one feasible point:

6.25 Lemma.
The only feasible point of (6.6) *is given by* $(x(t), u(t), v(t)) = ((\frac{1}{2}, \frac{1}{2}, 0)^T, 0, \frac{1}{2})$ *with objective function value* 1.

Proof. ([Ces83]) Let $(x_1(t), x_2(t), x_3(t), u(t), v(t))$ be feasible for (6.6), then $x_3(1) = x_3(0) = 0$ and $\dot{x}_3(t) \geq 0$ implies $x_3(t) \equiv 0$ a.e. $t \in [0,1]$. Thus $x_1(t) \equiv x_2(t)$ a.e. $t \in [0,1]$, therefore $|\dot{x}_1(t)| = |v(t)| \leq c$ and thus x is Lipschitz of rank $c \leq \frac{1}{8}$. Hence $|x_1(t) - \frac{1}{2}| = |x_1(t) - x_1(0)| \leq ct \leq \frac{1}{8}$ which implies $\frac{3}{8} \leq x_1(t) \leq \frac{5}{8}$ a.e. $t \in [0,1]$. Since $|u(t) - x_1(t)| = |\dot{x}_1(t)(t+\sigma)| \leq 2c = \frac{1}{4}$ it follows that $v(t) \equiv \frac{1}{2}$ a.e. $t \in [0,1]$. We find $\frac{1}{2} = v(t) = x_1(t) + (t+\sigma)\dot{x}_1(t) = \frac{\mathrm{d}}{\mathrm{d}t}((t+\sigma)x_1(t))$, integrating and using $x_1(0) = \frac{1}{2}$ yields $x_1(t) \equiv \frac{1}{2}$ a.e. $t \in [0,1]$. Thus $x_2(t) \equiv \frac{1}{2}$ a.e. $t \in [0,1]$ which implies $\dot{x}_2(t) \equiv 0$ and $u(t) \equiv 0$ a.e. $t \in [0,1]$. It is easy to confirm that $(x_1(t), x_2(t), x_3(t), u(t), v(t)) = (\frac{1}{2}, \frac{1}{2}, 0, 0, \frac{1}{2})$ indeed is feasible for (6.6). □

Applying partial outer convexification with respect to the three choices for $v(t)$, the counterpart problem BC-VC for (6.6) reads

$$\begin{array}{lll} \min\limits_{x,u,\omega} & \displaystyle\int_0^1 \omega_2(t)\ \mathrm{d}t & \\ \text{s.t.} & \dot{x}(t) = \sum_{i=1}^3 \omega_i(t) f(t, x(t), u(t), e_i) & \text{a.e. } t \in [0,1], \\ & x(0) = (\tfrac{1}{2}, \tfrac{1}{2}, 0)^T, & \\ & x_3(1) = 0, & \\ & u(t) \in [-c, c] & \text{a.e. } t \in [0,1], \\ & \omega_i(t) \in \{0,1\} & \text{a.e. } t \in [0,1], \\ & 1 = \omega_1(t) + \omega_2(t) + \omega_3(t) & \text{a.e. } t \in [0,1]. \end{array} \tag{6.7}$$

Again, this problem has only one feasible point with objective function value 1. The situation changes upon relaxation:

6.26 Lemma.
The relaxation of (6.7) *obtained by substituting* $\alpha_i(t) \in [0,1]$ *for* $\omega(t) \in \{0,1\}$*, has optimal objective function value* 0.

Proof. One immediately confirms that $(x(t), u(t), \alpha(t)) \equiv ((\frac{1}{2}, \frac{1}{2}, 0)^T, 0, (\frac{1}{2}, 0, \frac{1}{2})^T)$ is feasible with objective function value 0. Since $\alpha_2(t) \geq 0$ for all $t \in [0,1]$ and for every feasible point, this point is also optimal. □

We see that there is a gap between the optimal objective function values of (6.7) and its relaxation. If, however, one allows arbitrarily small violations of the end point constraint $x_3(1) = 0$, this gap can be made to zero as was done in and claimed by the Approximation Theorem 6.7. We show this in detail for (VC-SOS-SUR):

6.27 Proposition.
If (VC-SOS-SUR) *is applied to* $(x(t), u(t), \alpha(t)) \equiv ((\frac{1}{2}, \frac{1}{2}, 0)^T, 0, (\frac{1}{2}, 0, \frac{1}{2})^T)$ *on an equidistant grid of size* $2N$*, the violation of the constraint* $x_3(t) = 0$ *is given by*

$$|x_3^{\mathrm{VC}}(1)| \leq \tfrac{1}{16N^2\sigma(\sigma+1)}.$$

Proof. We consider a solution of the IVP in (6.7) obtained by rounding of $\alpha(t)$: We replace $\alpha(t)$ by $\omega^A(t)$ given by $\omega^A(t) := (1 - [t \in A], 0, [t \in A])^T$, where $A \subseteq [0,1]$ is measurable and specify for A generated by (VC-SOS-SUR) later. The objective function stays 0 for this control.

Since $\frac{\mathrm{d}}{\mathrm{d}t}((t+\sigma)x_1^A(t)) = (t+\sigma)\dot{x}_1^A(t) + x_1^A(t) = \frac{1}{2}\omega_2(t) + \omega_3(t) = [t \in A]$, integrating and using $x_1^A(0) = \frac{1}{2}$ yields $x_1^A(t) = \frac{1}{t+\sigma}\left(\frac{\sigma}{2} + \int_0^t [\tau \in A]\,\mathrm{d}\tau\right)$. Furthermore we find by

$v(t) \equiv 0$ that $x_2^A(t) \equiv \frac{1}{2}$a.e. $t \in [0,1]$. Thus

$$\dot{x}_3^A(t) = \left[\tfrac{1}{t+\sigma}\left(\tfrac{\sigma}{2} + \int_0^t [\tau \in A]\,\mathrm{d}\tau\right) - \tfrac{1}{2}\right]^2 = \left(\tfrac{1}{t+\sigma}\int_0^t ([\tau \in A] - \tfrac{1}{2})\,\mathrm{d}\tau\right)^2$$

and therefore

$$x_3^A(1) = \int_0^1 \left(\tfrac{1}{t+\sigma}\int_0^t ([\tau \in A] - \tfrac{1}{2})\,\mathrm{d}\tau\right)^2 \mathrm{d}t.$$

Now, applying (VC-SOS-SUR) on an equidistant grid of size $2N$ yields

$$A = A_N := \bigcup_{i=0}^{N-1} ((2i+1)\Delta, 2i+2\Delta), \qquad \text{with } \Delta := \tfrac{1}{2N}.$$

Denoting $t_N(t) := \frac{1}{N}\lfloor Nt \rfloor$ the largest multiple of $\frac{1}{N}$ that is $\le t$, we find:

$$\int_0^t ([\tau \in A_N] - \tfrac{1}{2})\,\mathrm{d}\tau = \int_{t_N(t)}^t ([\tau \in A_N] - \tfrac{1}{2})\,\mathrm{d}\tau = \begin{cases} -\frac{1}{2}(t - t_N(t)), & t \le t_N(t) + \Delta \\ \frac{1}{2}(t - t_N(t)) - \Delta, & t \ge t_N(t) + \Delta, \end{cases}$$

Thus $\left|\int_0^t ([\tau \in A_N](\tau) - \frac{1}{2})\,\mathrm{d}\tau\right| \le \frac{1}{4N}$ and therefore

$$x_3^{A_N}(1) \le \frac{1}{16N^2}\int_0^1 \frac{\mathrm{d}t}{(t+\sigma)^2} = \frac{1}{16N^2}\left(\frac{1}{\sigma} - \frac{1}{\sigma+1}\right) = \frac{1}{16N^2\sigma(\sigma+1)}.$$ □

Cesari's example is in some way pathological and ill-posed, as its solution depends in a discontinuous way on the terminal constraint $x_3(1) = \lambda$ with qualitatively different solutions for $\lambda = 0$ and $\lambda \neq 0$. Using [Ces83, Thm 18.6.i], it is possible to find a priori conditions to be imposed on (MIOCP) that ensure amenability to approximation without feasibility loss in constraints. An example of such a condition is the requirement that $c(\cdot,\cdot,v_i) = c_i$ does not depend on x and u, which is a rather restrictive condition so that we refrain from formulating a result. Palladino and Vinter [PV14] study a related question concerning relaxation gaps for certain optimal control problems. They consider problems where the differential equation constraint is expressed as differential inclusion $\dot{x}(t) \in F(t, x(t))$ for a multifunction F. The relaxation in this case is given by a convexification of the velocity sets $\dot{x}(t) \in \overline{\mathrm{co}}\, F(t, x(t))$. The relaxation gap is then defined as the difference between the infima of the two problems. This question is important from a theoretical point of view, since the latter formulation admits minimizers if certain technical assumptions are satisfied. They find a relationship between the occurrence of a relaxation gap and optimal solutions being non-trivial Fritz-John points with zero Fritz-John multiplier of the cost function. Cesari's example fits these considerations: Problem (6.6) has only a single feasible point and the solution does not depend on the objective function.

6.6. Summary

In this chapter, we have introduced the class of mixed-integer optimal control problems (MIOCP) that comprises optimal control problem with combinatorial constraints. Direct discretizations constitute mixed-integer nonlinear programs, which are NP hard. We have formulated a relaxed partial outer convexification of (MIOCP), that is a continuous optimal control problem without combinatorial constraints but with vanishing constraints. As main result, a correspondence between solutions of the relaxed partial outer convexification and solutions of the original problem (MIOCP) has been formulated and proven. It has been shown how this correspondence can be practically used with a Vanishing Constraint convergent rounding algorithm. With (VC-SOS-SUR) we have formulated an algorithm that is linear in the size of the temporal grid and conjecture that it is Vanishing Constraint convergent.

7. Sequential LPEC EQP Method for Equilibrium Constrained Problems

In this chapter, we develop a Sequential Linear Equilibrium Constrained Equality Constrained Quadratic Programming Method (SLPECEQP) for MPEC.

This method extends the Sequential Linear Equality Constraint Quadratic Programming Method (SLEQP) of Nocedal and Waltz [Byr+03; Byr+05] for nonlinear programs to MPEC and is similar in spirit to the suggestion of a Filter-SLPEC method by Leyffer and Munson [LM07]. A precursor of the method for equality constrained nonlinear programming is given by the ETR algorithm of Lalee et al. [LNP98], the first proposition of such methods for nonlinear programming is by Fletcher and Sainz de la Maza [FM89] and Chin and Fletcher [CF03].

We start by describing a general algorithmic framework for composite non-smooth problems with equilibrium constraints that covers MPEC by penalization of the nonlinear constraints and treatment of the complementarities as structural constraints and show global convergence to B-stationary points under certain assumptions. In the second part of the chapter, we describe a practical implementation of an algorithm of this class with a Newton-type acceleration given by an equality constraint quadratic programming step.

Parts of the results of this chapter are published in [LKB17].

7.1. A Composite Non-Smooth Problem Formulation

We consider the following composite non-smooth problem:

7.1 Definition.
Let $\mathcal{E}, \mathcal{I}$ be finite, disjoint index sets, $F : \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s} \to \mathbb{R} \times \mathbb{R}^{\mathcal{E}} \times \mathbb{R}^{\mathcal{I}}$ be continuously differentiable and $\omega : \mathbb{R} \times \mathbb{R}^{\mathcal{E}} \times \mathbb{R}^{\mathcal{I}} \to \mathbb{R}$ be convex. Define $\phi : \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s} \to \mathbb{R}$ by composition, $\phi(x, s, t) := \omega(F(x, s, t))$. The composite non-smooth problem defined by F and ω is given by

$$\begin{aligned} \min_{x,s,t} \quad & \phi(x, s, t) \\ \text{s.t.} \quad & 0 \leq s \perp t \geq 0. \end{aligned} \tag{PenEC}$$

△

We consider a Sequential Linear Equilibrium Constraint Method for solving (PenEC).

Assuming that MPEC-MFCQ holds for (MPEC), boundedness of multipliers corresponding to $c_{\mathcal{E}}$ and $c_{\mathcal{I}}$ is ensured and solving (MPEC) is equivalent to solving exact penalty formulation (PenEC) with $F(x,s,t) := (f(x,s,t), c_{\mathcal{E}}(x,s,t), c_{\mathcal{I}}(x,s,t))$ and $\omega(f, c_{\mathcal{E}}, c_{\mathcal{I}}) := f + \gamma\|c_{\mathcal{E}}\|_1 + \gamma\|\min\{0, c_{\mathcal{I}}\}\|_1$ for sufficiently large $\gamma > 0$, compare Theorem 4.14.

The presentation, algorithmic development and convergence proof follows the nonlinear programming case without equilibrium constraints developed by Nocedal and Waltz [Byr+03; Byr+05], which is related and builds upon the algorithm and analysis developed by Fletcher and de la Maza [FM89], Chin and Fletcher [CF03] and the non-smooth trust-region convergence framework of Yuan [Yua85]. Most of the parts unrelated to the complementarity constraint and the linearized model hold without adaption as in [Byr+05] and are restated here with proof for completeness and convenience.

7.2. Stationarity of Composite Non-Smooth Problem

In this section, we define a stationarity concept for (PenEC) and show its equivalence to B-stationarity if applied to (MPEC). To this end, we introduce the definition of a linearized model ℓ and in addition for the algorithm require a quadratic model q that we introduce now as well.

7.2 Definition (Linearized model ℓ, quadratic model q of ϕ).
Let $z = (x,s,t) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s}$. Then the *linearized model* ℓ of ϕ at z is defined by

$$\ell : d \mapsto \omega(F(z) + \langle \nabla F(z), d\rangle)$$

Let furthermore $H \in \mathbb{R}^{(n_x+2n_s)\times(n_x+2n_s)}$ be symmetric. Then the *quadratic model* q of ϕ at z is defined by

$$q : d \mapsto \ell(d) + \tfrac{1}{2}\langle d, Hd\rangle.$$

We will write $\ell(z,d)$ for $\ell(d)$ and $q(z,d)$ for $q(d)$ if the linearization point z is not clear and if we are considering a sequence $(z^k)_k$ of points, we will write ℓ_k for $d \mapsto \ell(z^k, d)$ and q_k for $d \mapsto q(z^k, d)$ corresponding to a sequence $(H_k)_k$. △

With the linearized model we can introduce the notion of optimal linear decrease ψ, which ultimately allows us to define stationarity.

7.3 Definition (Optimal linear decrease ψ).
The *optimal linear decrease* $\psi : \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s} \times \mathbb{R}_{>0} \to \mathbb{R} \cup \{-\infty\}$ is defined by

$$\psi(z,\Delta) := \ell(z,0) - \min_{\substack{\|d\|_\infty \le \Delta,\\ 0 \le s+d_s \perp t+d_t \ge 0}} \ell(z,d),$$

provided that the minimum in the second term exists, and $\psi(z, \Delta) := -\infty$ otherwise. Again, we write ψ_k for $\Delta \mapsto \psi(z^k, \Delta)$ if we are considering a sequence $(z^k)_k$. △

7.4 Definition (Stationarity of ϕ).
A point $z^* = (x^*, s^*, t^*) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s}$ is defined to be *critical*, if $0 \le s^* \perp t^* \ge 0$ and $\psi(z^*, 1) = 0$. △

We will now show that this stationarity notion matches exactly B-stationarity for (MPEC).

7.5 Theorem.
Assume that (MPEC) *has a feasible point and satisfies MPEC-MFCQ. Let F and ω be defined by*

$$F(z) = (f(z), c_{\mathcal{E}}(z), c_{\mathcal{I}}(z)),$$
$$\omega(f, c_{\mathcal{E}}, c_{\mathcal{I}}) = f + \gamma\|c_{\mathcal{E}}\|_1 + \gamma\|\min\{0, c_{\mathcal{I}}\}\|_1$$

with $\gamma > 0$ sufficiently large.

Then z^* is critical for (PenEC) *if and only if z^* is B-stationary for* (MPEC).

Proof. The point z^* is critical by definition if and only if $d = 0$ solves

$$\min_{\substack{\|d\|_\infty \le 1,\\ 0 \le s + d_s \perp t + d_t \ge 0}} \ell(x, s, t, d). \tag{7.1}$$

By choice of F and ω, $\ell(z^*, d) = \Delta^f + \gamma\Delta^{\mathcal{E}} + \gamma\Delta^{\mathcal{I}}$ with

$$\begin{aligned}
\Delta^f &= \langle \nabla f(z^*), d\rangle,\\
\Delta^{\mathcal{E}} &= \|\langle \nabla c_{\mathcal{E}}(z^*), d\rangle\|_1,\\
\Delta^{\mathcal{I}} &= \|\min\{0, \langle \nabla c_{\mathcal{I}}(z^*), d\rangle\}\|_1.
\end{aligned}$$

By feasibility of (MPEC), satisfaction of MPEC-MFCQ and classical penalty arguments of Theorem 4.14, assuming that $\gamma > 0$ sufficiently large, solving (7.1) is equivalent to solving

$$\begin{aligned}
\min_d \quad & \langle \nabla f(z), d\rangle\\
\text{s.t.} \quad & 0 = c_{\mathcal{E}}(z) + \langle \nabla c_{\mathcal{E}}(z), d\rangle,\\
& 0 \le c_{\mathcal{I}}(z) + \langle \nabla c_{\mathcal{I}}(z), d\rangle,\\
& 0 \le s + d_s \perp t + d_t \ge 0,\\
& \|d\|_\infty \le 1.
\end{aligned}$$

The point z^* is thus critical if and only if $d = 0$ solves this LPEC, where the trust-region constraint is inactive and thus can be omitted. Thus, criticality of z^* is equivalent to B-stationarity. □

7.3. Algorithmic Framework

We consider a class of algorithms described by Algorithm 7.1. Let $\|\cdot\|$ be a norm on $\mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s}$, for example $\|\cdot\| = \|\cdot\|_2$. Since all norms on a finite-dimensional real vector space are equivalent, there is $C > 0$ with $\|\cdot\| \leq C \|\cdot\|_\infty$.

We note that the problem defining d_ℓ^k is always feasible and Algorithm 7.1 is thus well-defined.

7.4. Global Convergence Result

We show global convergence of Algorithm 7.1 and show that the sequence of points generated by Algorithm 7.1 contains either a critical point or has an accumulation point that is critical.

Throughout the section we assume that $(z^k)_k$ has been generated by Algorithm 7.1 and make the following general assumptions:

7.6 Assumption.

T. The trust-region radii for model ℓ are bounded, $\liminf \Delta_\ell^k > 0$.

C. $\{d \mid 0 \leq s^k + d_s \perp t^k + d_t \geq 0, \|d\|_\infty \leq \Delta_\ell^k\} \neq \emptyset$ for almost all k.

L. There exists a set $D \subset \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s}$ such that
   - $z^k \in \operatorname{int} D$ for all k and $z \in \operatorname{int} D$ for every accumulation point z of $(z^k)_k$,
   - D is bounded and convex,
   - F and ∇F are Lipschitz-continuous on D,
   - ω is Lipschitz-continuous on $F(D)$.

B. The sequence of Hessians H_k associated to q_k is uniformly bounded, there exists $C_H > 0$ such that $|\langle d, H_k d\rangle| \leq C_H \|d\|^2$ for all k and d. △

Assumptions L and B are standard assumptions for convergence of nonlinear programming algorithms while assumption C states that the complementarity constraint can be satisfied in a neighborhood of the iterates. In numerical examples we have observed convergence even if assumptions T and C are not satisfied. We need these assumption however to prove Lemmata 7.12 and 7.13.

Assumption C can be guaranteed by choosing an initial point z^0 that is feasible with respect to the complementarity, $0 \leq t^0 \perp s^0 \geq 0$ and by parametric computation of steps ii. and iii. along the projection of the complementarities.

By rescaling and skipping the first iterates we can strengthen these assumptions and assume from now on these stronger assumptions:

Algorithm 7.1: Sequential Linear Equilibrium Constraint Framework for solving (PenEC), with the notation $z^k = (x^k, s^k, t^k)$.

input :

- initial point x^0, s^0, t^0
- trust-region radius $\Delta_\ell^0 > 0$ for model ℓ
- master trust-region radius $\Delta^0 > 0$
- trust-region thresholds $0 < \rho^{\mathrm{u}} \leq \rho^{\mathrm{s}} < 1$
- master trust-region shrinking factors $0 < \kappa^{\mathrm{l}} \leq \kappa^{\mathrm{u}} < 1$
- linesearch constant $\eta > 0$ and reduction factor $\tau > 0$
- constant $\theta > 0$ coupling ℓ trust-region and master trust-region

for $k \geq 0$ **do**

i. Compute solution d_ℓ^k to $\min\limits_{\substack{\|d\|_\infty \leq \Delta_\ell^k, \\ \max\{0, s^k - \Delta_\ell^k\} \leq s^k + d_s \perp t^k + d_t \geq \max\{0, t^k - \Delta_\ell^k\}}} \ell_k(d)$.

ii. Compute Cauchy step $d_{\mathrm{C}}^k = \alpha^k d_\ell^k$ by computing maximal α^k among $(\tau^i \min\{1, \frac{\Delta^k}{\|d_\ell^k\|}\})_i$ such that $\phi(z^k) - q_k(\alpha^k d_\ell^k) \geq \eta(\phi(z^k) - \ell(\alpha^k d_\ell^k))$.

iii. Compute trial step d^k such that $\|d^k\| \leq \Delta^k$ and $q_k(d^k) \leq q_k(d_{\mathrm{C}}^k)$.

iv. Compute step performance ratio $\rho^k = \frac{\phi(z^k) - \phi(z^k + d^k)}{\phi(z^k) - q_k(d^k)}$.

v. Update master trust-region radius $\begin{cases} \Delta^{k+1} \geq \Delta^k, & \rho^k \geq \rho^{\mathrm{s}}, \\ \kappa^{\mathrm{l}} \|d^k\| \leq \Delta^{k+1} \leq \kappa^{\mathrm{u}} \Delta^k, & \rho^k < \rho^{\mathrm{s}}. \end{cases}$

vi. Update trust-region radius for model ℓ

$$\begin{cases} \Delta_\ell^{k+1} \geq \|d_{\mathrm{C}}^k\|_\infty, & \rho^k \geq \rho^{\mathrm{u}} \text{ and } \alpha^k = 1, \\ \Delta_\ell^k \geq \Delta_\ell^{k+1} \geq \|d_{\mathrm{C}}^k\|_\infty, & \rho^k \geq \rho^{\mathrm{u}} \text{ and } \alpha^k < 1, \\ \min\{\theta \|d^k\|_\infty, \Delta_\ell^k\} \leq \Delta_\ell^{k+1} \leq \Delta_\ell^k, & \rho^k < \rho^{\mathrm{u}}. \end{cases}$$

vii. Update iterate $z^{k+1} = \begin{cases} z^k + d^k, & \rho^k \geq \rho^{\mathrm{u}}, \\ z^k, & \rho^k < \rho^{\mathrm{u}}. \end{cases}$

end

7.7 Assumption.
T'. $\Delta_\ell^k \geq 1$ for all k.

C'. $\{d \mid 0 \leq s^k + d_s \perp t^k + d_t \geq 0, \|d\|_\infty \leq \Delta_\ell^k\} \neq \varnothing$ for all k. △

The central convergence result of this section is given by the following Theorem and states that Algorithm 7.1 converges in a critical point after a finite number of iterations or generates a sequence that contains a critical accumulation point.

7.8 Theorem (Convergence result).
Assume that $\liminf \phi(z^k) > -\infty$, then either there is $m \in \mathbb{N}$ with $\psi_m(1) = 0$ or $\liminf \psi_k(1) = 0$. △

In view of Theorem 7.5, if this Algorithm is applied to (MPEC), convergence to B-stationary points is guaranteed.

We defer the proof of Theorem 7.8 to the end of the section and start by formulating and proving several lemmata that sum up to the ultimate convergence result.

7.9 Definition.
For $z = (x, s, t) \in D$ and $\Delta > 0$, let $d(z, \Delta)$ be the solution of

$$\min_{\substack{\|d\|_\infty \leq \Delta, \\ \max\{0, s-\Delta\} \leq s+d_s \perp t+d_t \geq \max\{0, t-\Delta\}}} \ell(z, d).$$

△

7.10 Lemma.
Let $z \in D$, $\tau \in [0, 1]$ and $d, d' \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_s} \times \mathbb{R}^{n_s}$.
Then $\ell(z, 0) - \ell(z, \tau d) \geq \tau(\ell(z, 0) - \ell(z, d))$ and $\phi(z^k) - \ell_k(\tau d) \geq \tau(\phi(z^k) - \ell_k(d))$.

Proof. By convexity of ω, convexity of $\ell(z, \cdot)$ follows. Thus

$$\ell(z, \tau d) = \ell(z, \tau d + (1 - \tau)0) \leq \tau \ell(z, d) + (1 - \tau)\ell(z, 0),$$

which is equivalent to the assertion. □

7.11 Lemma (Lipschitz-continuity of ϕ, ℓ, approximation error of ℓ).
The function ϕ is Lipschitz-continuous and the mapping $d \mapsto \ell(z, d)$ is uniformly Lipschitz-continuous.
We denote by $L_\ell, L_{\nabla F}, L_\omega > 0$ Lipschitz constants such that

$$\begin{aligned} |\ell(z, d) - \ell(z, d')| &\leq L_\ell \|d - d'\|_\infty, \\ \|\nabla F(z) - \nabla F(z')\| &\leq L_{\nabla F} \|z - z'\|, \\ |\omega(\xi) - \omega(\xi')| &\leq L_\omega \|\xi - \xi'\|. \end{aligned}$$

Then the inequality $|\ell(z, d) - \phi(z + d)| \leq \frac{1}{2} L_\omega L_{\nabla F} \|d\|^2$ holds.

Proof. Lipschitz continuity of ϕ and $\ell(z,\cdot)$ follows from assumption L.

The following computation demonstrates the inequality:

$$\begin{aligned}
|\ell(z,d) - \phi(z+d)| &= |\omega(F(z) + \langle \nabla F(z), d\rangle) - \omega(F(z+d))| \\
&\le L_\omega \|F(z+d) - F(z) - \langle \nabla F(z), d\rangle\| \\
&\le L_\omega \left\| \int_0^1 \langle \nabla F(z+td) - \nabla F(z), d\rangle \,\mathrm{d}t \right\| \\
&\le L_\omega \|d\| \int_0^1 \|\nabla F(z+td) - \nabla F(z)\| \,\mathrm{d}t \\
&\le L_\omega L_{\nabla F} \|d\| \int_0^1 \|td\| \,\mathrm{d}t = \tfrac{1}{2} L_\omega L_{\nabla F} \|d\|^2.
\end{aligned}$$

□

7.12 Lemma (Comparison of achievable reduction to $\Delta = 1$ case).
For $\Delta \ge \Delta_\ell^k$ the inequality $\psi_k(\Delta) \ge \psi_k(1)$ holds.

Proof. If $\psi_k(1) = -\infty$, there is nothing to prove. We can thus assume $\psi_k(1) > -\infty$, which shows that $\{d \mid 0 \le s + d_s \perp t + d_t \ge 0, \|d\|_\infty \le 1\} \neq \emptyset$ and $d(z^k, 1)$ exists. By definition, $\psi_k(\Delta) = \ell_k(0) - \ell_k(d(z^k, \Delta))$. Since by assumption T' $\|d(z^k,1)\|_\infty \le 1 \le \Delta$ and by assumption C' $d(z^k,1)$ is feasible in the definition of the minimization problem of $\psi_k(\Delta)$ and it follows $\psi_k(\Delta) = \ell_k(0) - \ell_k(\Delta) \ge \ell_k(0) - \ell_k(1) = \psi_k(1)$. □

7.13 Lemma (Trust-region is active at noncritical points).
If $\psi_k(1) \neq 0$ and $\Delta \ge \Delta_\ell^k$, then $\|d(z^k,\Delta)\|_\infty \ge \min\{\Delta, \frac{1}{L_\ell}\psi_k(1)\}$.

Proof. Assume $\|d(z,\Delta)\|_\infty < \frac{1}{L_\ell}\psi_k(1)$. By Lipschitz-continuity of ℓ_k and optimality of $d(z,\Delta)$ it follows that $\psi_k(\Delta) = \ell_k(0) - \ell_k(d(z,\Delta)) = |\ell_k(0) - \ell_k(d(z,\Delta)) - \ell_k(0)| \le L_\ell \|d(z,\Delta)\|_\infty < \psi_k(1)$ which contradicts Lemma 7.12. □

7.14 Lemma (Lower bound on achievable reduction).
The predicted reduction satisfies

$$\phi(z^k) - q_k(d^k) \ge \phi(z^k) - q_k(d_{\mathrm{C}}^k) \ge \eta\alpha^k \psi_k(\Delta_\ell^k) \ge \eta\alpha^k \psi_k(1).$$

Proof. The first inequality $\phi(z^k) - q_k(d^k) \ge \phi(z^k) - q_k(d_{\mathrm{C}}^k)$ follows from the choice of $d_{\mathrm{C}}^k = \alpha^k d_\ell^k$ in iii. in Algorithm 7.1. The second inequality $\phi(z^k) - q_k(d_{\mathrm{C}}^k) \ge \eta\alpha^k \psi_k(\Delta_\ell^k)$ follows from ii. in Algorithm 7.1 and Lemma 7.10:

$$\phi(z^k) - q_k(\alpha^k d_\ell^k) \overset{\text{ii.}}{\ge} \eta(\phi(z^k) - \ell_k(\alpha^k d_\ell^k)) \overset{7.10}{\ge} \eta\alpha^k(\phi(z^k) - \ell_k(d_\ell^k)) = \eta\alpha^k \psi_k(\Delta_\ell^k).$$

The remaining inequality follows from Lemma 7.12. □

7.15 Lemma (Approximation error of quadratic model).
The estimation $|q_k(d^k) - \phi(z^k + d^k)| \leq \frac{1}{2}(L_\omega L_{\nabla F} + C_H)\|d^k\|^2$ *holds.*

Proof. By Lemma 7.11 and Assumption B, it follows

$$\begin{aligned}|q_k(d^k) - \phi(z^k + d^k)| &\leq |l_k(d^k) - \phi(z^k + d^k)| + |\tfrac{1}{2}\langle d^k, H_k d^k\rangle| \\ &\leq \tfrac{1}{2}(L_\omega L_{\nabla F} + C_H)\|d^k\|^2.\end{aligned}$$

□

7.16 Lemma (Bounds on Cauchy step size).
The Cauchy step satisfies $\alpha^k \Delta_\ell^k \geq \|d_{\mathrm{C}}^k\|_\infty \geq \min\{\Delta_\ell^k, \frac{1}{C}\Delta^k, \frac{1}{L_\ell}\psi_k(1), \frac{2(1-\eta)\tau}{\Delta_\ell^k C_H C^2}\psi_k(1)\}$.

Proof. By definition, in step ii. in Algorithm 7.1, $d_{\mathrm{C}}^k = \alpha^k d_\ell^k$. Since $\|d_\ell^k\|_\infty \leq \Delta_\ell^k$, the upper inequality follows immediately.

We now prove the lower inequality. In the case $\psi_k(1) = 0$, there is nothing to prove. We can thus assume $\psi_k(1) \neq 0$.

Suppose that $\alpha^k = \min\{1, \frac{\Delta^k}{\|d_\ell^k\|}\}$. By norm equivalence $\frac{\Delta^k}{\|d_\ell^k\|} \geq \frac{\Delta^k}{C\|d_\ell^k\|_\infty}$. Then

$$\|\alpha^k d_\ell^k\|_\infty = \min\{1, \tfrac{\Delta^k}{\|d_\ell^k\|}\}\|d_\ell^k\|_\infty \geq \min\{\|d_\ell^k\|_\infty, \tfrac{\Delta^k}{C}\} \overset{7.13}{\geq} \min\{\Delta_\ell^k, \tfrac{1}{L_\ell}\psi_k(1), \tfrac{\Delta^k}{C}\},$$

which proves the lower inequality in this case.

Now assume $\alpha^k < \min\{1, \frac{\Delta^k}{\|d_\ell^k\|}\}$ and we attempt to prove $\|d_{\mathrm{C}}^k\|_\infty \geq \frac{2(1-\eta)\tau}{\Delta_\ell^k C_H C^2}\psi_k(1)$ in that case. Then, the decrease condition $\phi(z^k) - q_k(\alpha d_\ell^k) \geq \eta(\phi(z^k) - \ell(\alpha d_\ell^k))$ in step ii. must have been violated for $\alpha^- := \frac{1}{\tau}\alpha^k$ as previous trial step length. In particular,

$$\phi(z^k) - \ell_k(\alpha^- d_\ell^k) - \tfrac{1}{2}(\alpha^-)^2\langle d_\ell^k, H_k d_\ell^k\rangle = \phi(z^k) - q_k(\alpha^- d_\ell^k) < \eta(\phi(z^k) - \ell(\alpha^- d_\ell^k)).$$

We thus find

$$\begin{aligned}\tfrac{1}{2}(\alpha^-)^2\langle d_\ell^k, H_k d_\ell^k\rangle > (1-\eta)(\phi(z^k) - \ell_k(\alpha^- d_\ell^k)) &\overset{7.10}{\geq} \alpha^-(1-\eta)(\phi(z^k) - \ell_k(d_\ell^k)) \\ = \alpha^-(1-\eta)\psi_k(\Delta_\ell^k) &\overset{7.12}{\geq} \alpha^-(1-\eta)\psi_k(1)\end{aligned}$$

and by norm equivalence and trust-region feasibility

$$\langle d_\ell^k, H_k d_\ell^k\rangle \leq C_H\|d_\ell^k\|^2 \leq C_H C^2\|d_\ell^k\|_\infty^2 \leq C_H C^2\|d_\ell^k\|_\infty \Delta_\ell^k.$$

Combining these two inequalities yields the desired result

$$\alpha^k\|d_\ell^k\|_\infty \geq \frac{\alpha^k}{C_H C^2 \Delta_\ell^k}\langle d_\ell^k, H_k d_\ell^k\rangle \geq \frac{\alpha^k}{C_H C^2 \Delta_\ell^k}\frac{2(1-\eta)}{\alpha^-}\psi_k(1) = \frac{2(1-\eta)\tau}{\Delta_\ell^k C_H C^2}\psi_k(1).$$

□

7.17 Lemma (Lower bound on master trust-region radius and Cauchy step). *Let* $\limsup \Delta_\ell^k < \infty$ *and there is* $\delta > 0$ *with* $\psi_k(1) \geq \delta > 0$ *for all* k.

Then there exists a lower bound $\underline{\Delta} > 0$ *for the master trust region radius with* $\Delta^k \geq \underline{\Delta}$ *and* $\alpha^k \Delta_\ell^k \geq \frac{1}{C}\underline{\Delta}$.

Proof. By assumption $\Delta_\ell^k \leq \overline{\Delta}_\ell$ with $1 \leq \overline{\Delta}_\ell < \infty$. Since $\psi_k(1) \geq \delta > 0$, it follows from Lemma 7.16 that $\|d_{\mathrm{C}}^k\|_\infty \geq \min\{\frac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}\}$ with $\Delta_{\mathrm{crit}} := \min\{\frac{1}{L_\ell}, \frac{2(1-\eta)\tau}{\overline{\Delta}_\ell C_H C^2}\}\delta$.

We will first show the recursive lower bound $\Delta_\ell^{k+1} \geq \min\{\frac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}, \Delta'_{\mathrm{crit}}\}$ for the ℓ trust region radius, where $\Delta'_{\mathrm{crit}} := \min\{\theta^2, (\kappa^{\mathrm{l}})^2\} \min\{(1-\rho^{\mathrm{u}}), (1-\rho^{\mathrm{s}})\} \frac{2\eta}{C^2 L_\omega L_{\nabla F} \overline{\Delta}_\ell} \delta$.

If the iteration is successful, $\rho^k \geq \rho^{\mathrm{u}}$, the update vi. in Algorithm 7.1 ensures $\Delta_\ell^{k+1} \geq \|d_{\mathrm{C}}^k\|_\infty \geq \min\{\frac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}\}$. On the other hand, if the iteration yields a discarded step, $\rho^k < \rho^{\mathrm{u}}$, we find

$$\phi(z^k) - q_k(d^k) \overset{7.14}{\geq} \eta\alpha^k \psi_k(1) \geq \eta \frac{\alpha^k \Delta_\ell^k}{\Delta_\ell^k} \delta \geq \eta \frac{\|d_{\mathrm{C}}^k\|_\infty}{\overline{\Delta}_\ell} \delta \geq \frac{\eta\delta}{\overline{\Delta}_\ell} \min\{\tfrac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}\}$$

and by Lemma 7.15 and step acceptance failure we thus find for $1 - \rho^k$:

$$1 - \rho^{\mathrm{u}} < 1 - \rho^k = \frac{\phi(z^k + d^k) - q_k(d^k)}{\phi(z^k) - q_k(d^k)} \leq \frac{\overline{\Delta}_\ell (L_\omega L_{\nabla F} + C_H) \|d^k\|^2}{2\eta\delta \min\{\frac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}\}}.$$

This inequality yields

$$\begin{aligned} \theta^2 \|d^k\|_\infty^2 \geq \tfrac{\theta^2}{C^2} \|d^k\|^2 &\geq \frac{2\eta(1-\rho^{\mathrm{u}})\delta}{\overline{\Delta}_\ell (L_\omega L_{\nabla F} + C_H)} \min\{\tfrac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}\} \\ &\geq \left(\min\{\tfrac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}, \Delta'_{\mathrm{crit}}\}\right)^2, \end{aligned}$$

where the last inequality follows from the fact that for both factors $\frac{2\eta(1-\rho^{\mathrm{u}})\delta}{\overline{\Delta}_\ell (L_\omega L_{\nabla F} + C_H)} \geq \min\{\frac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}, \Delta'_{\mathrm{crit}}\}$ and $\min\{\frac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}\} \geq \min\{\frac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}, \Delta'_{\mathrm{crit}}\}$. As step vi. ensures $\Delta_\ell^{k+1} \geq \min\{\theta\|d^k\|_\infty, \Delta_\ell^k\}$ the asserted recursive lower bound for Δ_ℓ^{k+1} follows.

Next, we show a recursive lower bound $\frac{1}{C}\Delta^{k+1} \geq \min\{\frac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}, \Delta'_{\mathrm{crit}}\}$ in a similar way for the master trust-region radius. Since $\Delta^{k+1} \geq \Delta^k$ for $\rho^k \geq \rho^{\mathrm{s}}$, there is nothing to prove in that case. Suppose therefore $\rho^k < \rho^{\mathrm{s}}$. Then $1 - \rho^{\mathrm{s}} \leq 1 - \rho^k$ and exactly the same reasoning as before with ρ^{s} instead of ρ^{u} and κ^{u} instead of θ by the update mechanism of step vi yields the desired result.

Combining the recursive lower bounds for the two trust-region radii yields

$$\min\{\Delta_\ell^{k+1}, \tfrac{1}{C}\Delta_\ell^{k+1}\} \geq \min\{\tfrac{1}{C}\Delta^k, \Delta_\ell^k, \Delta_{\mathrm{crit}}, \Delta'_{\mathrm{crit}}\},$$

and via induction finally

$$\min\{\Delta_\ell^{k+1}, \tfrac{1}{C}\Delta_\ell^{k+1}\} \geq \min\{\tfrac{1}{C}\Delta^0, \Delta_\ell^0, \Delta_{\text{crit}}, \Delta'_{\text{crit}}\} =: \underline{\Delta}_\ell.$$

Thus $\Delta^k \geq C\underline{\Delta}_\ell =: \underline{\Delta}$ and $\alpha^k \Delta_\ell^k \geq \|d_C^k\|_\ell \geq \underline{\Delta}_\ell = \frac{1}{C}\underline{\Delta}$. □

We are now in a position to prove the convergence result of Theorem 7.8:

Proof. We first consider the case that only a finite number of iterations yield a successful step, i.e. $\rho^k < \rho^{\text{u}}$ for almost all k. Then the sequence $(z^k)_k$ becomes stationary and in consequence also $(\psi_k(1))_k$. Since $\kappa^{\text{u}} < 1$ and $\Delta^{k+1} \leq \kappa^{\text{u}}\Delta^k$ for almost all k, it follows $\Delta^k \to 0$. By Lemma 7.17 it follows $\liminf \psi_k(1) = 0$ as otherwise $\Delta^k \to 0$ would contradict the assertion of the Lemma. Thus $\psi_k(1) = 0$ for almost all k.

Next, we consider the case that an infinite number of iterations yield a successful step, i.e. $\rho^k \geq \rho^{\text{u}}$ for infinitely many k and show that $\liminf \psi_k(1) > 0$ results in a contradiction. We consider the cases $\limsup \Delta_\ell^k < \infty$ and $\limsup \Delta_\ell^k = \infty$ separately.

Suppose $\delta := \liminf \psi_k(1) > 0$ and $\overline{\Delta}^\ell := \limsup \Delta_\ell^k < \infty$. We find by Lemma 7.17 that there is $\underline{\Delta}$ with $\Delta^k \geq \underline{\Delta}$ for all k. For every k with $\rho^k \geq \rho^{\text{u}}$ we thus find

$$\begin{aligned}\phi(z^k) - \phi(z^{k+1}) = \phi(z^k) - \phi(z^k + d^k) &\geq \rho^{\text{u}}(\phi(z^k) - q_k(d^k)) \overset{7.14}{\geq} \rho^{\text{u}}\eta\alpha^k\delta \\ &= \rho^{\text{u}}\eta\alpha^k \tfrac{\Delta_\ell^k}{\Delta_\ell^k}\delta \overset{7.17}{\geq} \tfrac{\rho^{\text{u}}\eta\underline{\Delta}\delta}{C\overline{\Delta}^\ell}\end{aligned}$$

and conclude

$$\begin{aligned}\phi(z^k) = \sum_{j<k}(\phi(z^{j+1}) - \phi(z^j)) &= \sum_{\substack{m<k \\ \rho^m \geq \rho^{\text{u}}}} (\phi(z^{m+1}) - \phi(z^m)) \\ &\leq -\tfrac{\rho^{\text{u}}\eta\underline{\Delta}\delta}{C\overline{\Delta}^\ell}|\{m \leq k \mid \rho^m \geq \rho^{\text{u}}\}| \to -\infty,\end{aligned}$$

which contradicts the assumption $\liminf \phi(z^k) > -\infty$.

The last case that remains to be considered is given by $\delta = \liminf \psi_k(1) > 0$ and $\limsup \Delta_\ell^k = \infty$. Δ_ℓ^k is eventually increased in step vi. of Algorithm 7.1 and an increase only occurs if $\alpha^k = 1$, there are infinitely many k such that $\Delta_\ell^k > 1, \alpha^k \geq 1$ and $\rho^k \geq \rho^{\text{u}}$. For every such k we similarly find

$$\phi(z^k) - \phi(z^{k+1}) = \phi(z^k) - \phi(z^k + d^k) \geq \rho^{\text{u}}(\phi(z^k) - q_k(d^k)) \overset{7.14}{\geq} \rho^{\text{u}}\eta\alpha^k\delta \geq \rho^{\text{u}}\eta\delta,$$

and exactly as in the previous case we conclude $\liminf \phi(z^k) = -\infty$ in contradiction to $\liminf \phi(z^k) > -\infty$. □

7.5. The SLPECEQP Algorithm

The framework 7.1 is rather generic and if viewed towards a practical implementation leaves a number of questions untouched and unanswered. In this section, we describe an active set algorithm that falls into the category of algorithms constituted by 7.1 and thus exhibits its global convergence properties. Fast local convergence is promoted with a Newton-type step. The algorithm extends the `SLIQUE/SLEQP` algorithm of Waltz and Nocedal [Byr+03; Byr+05] for nonlinear programming that is implemented in the commercial solver `Knitro` [BNW06].

Step ii. of 7.1 is computed by the solution of a *Linear Program with Equilibrium Constraints* and yields a step d_ℓ^k and a working set guess $\mathcal{W}^k$. The trial step d^k in step iii. of 7.1 is computed by solving an equality constrained quadratic program on the working set estimate $\mathcal{W}^k$. The `SLPECEQP` algorithm is outlined in algorithm 7.2 and attempts to solve (MPEC) by solving (PenEC) with

$$\begin{aligned} F(x,s,t) &:= (f(x,s,t), c_{\mathcal{E}}(x,s,t), c_{\mathcal{I}}(x,s,t)), \\ \omega(f, c_{\mathcal{E}}, c_{\mathcal{I}}) &:= f + \gamma\|c_{\mathcal{E}}\|_1 + \gamma\|\min\{0, c_{\mathcal{I}}\}\|_1. \end{aligned}$$

Algorithm 7.2: `SLPECEQP` algorithm for solving (MPEC), with the notation $z^k = (x^k, s^k, t^k)$. Roman enumeration corresponds to steps of Algorithm 7.1.

input :

- initial point x^0, s^0, t^0
- ℓ trust-region radius $\Delta_\ell^0 > 0$
- master trust-region radius $\Delta^0 > 0$

for $k \geq 0$ **do**

- **a. (i.)** Compute penalty choice γ^k, linear step d_ℓ^k, working set guess $\mathcal{W}^k$.
- **b.** Compute least-squares multiplier estimation $\lambda_{\text{LS}}^k, \mu_{\text{LS}}^k, \nu_{\text{LS}}^k, \sigma_{\text{LS}}^k$.
- **c. (ii.)** Compute Cauchy step $d_{\text{C}}^k = \alpha^k d_\ell^k$.
- **d. (iii.)** Compute trial step d^k as solution of an EQP on $\mathcal{W}^k$.
- **e. (iv.)** Compute step performance ratio $\rho^k = \frac{\phi(z^k) - \phi(z^k + d^k)}{\phi(z^k) - q_k(d^k)}$.
- **f.** Eventually compute second order correction, update trial step d^k.
- **g. (v.)** Update master trust-region radius Δ^{k+1}.
- **h. (vi.)** Update ℓ trust-region radius Δ_ℓ^{k+1}.
- **i. (vii.)** Update iterate z^{k+1}.

end

We describe and elaborate on the steps a.–i. of 7.2 in the following. To shorten notation, we set $f^k := f(z^k)$, $\nabla f^k := \nabla f(z^k)$, $c_{\mathcal{E}}^k := c_{\mathcal{E}}(z^k)$, $c_{\mathcal{I}}^k := c_{\mathcal{E}}(z^k)$, $\nabla c_{\mathcal{E}}^k := \nabla c_{\mathcal{E}}(z^k)$ and $\nabla c_{\mathcal{I}}^k := \nabla c_{\mathcal{E}}(z^k)$ and $c(z) := (c_{\mathcal{E}}(z), c_{\mathcal{I}}(z), s, t)$.

LPEC phase (Step a)

In this phase, the step d_ℓ^k, and the working set guess $\mathcal{W}^k$ are computed and it is ensured that the penalty parameter γ^k is of suitable size. Computing the step d_ℓ^k is done by solving a *Linear Program with Complementarity Constraints*. Júdice [Júd12] surveys methods for solving LPEC including also methods designed for MPEC. Methods targeted directly at LPEC are given by the Complementarity Active-Set algorithm of Júdice [Júd+07], the algorithm of Hu and Pang [Hu08; Hu+12] building on integer programming and by Yu and Pang [YMP] building on a branch-and-cut framework. Fang et al. [FLM12] develop a pivoting algorithm that builds on the Simplex Algorithm for Linear Programming. Kirches [Kir17] proposed an augmented Lagrangian Algorithm using Gradient Projection that we utilize in our implementation. In the absence of complementarity constraints, this is a linear problem for which highly efficient active set algorithms based on the Simplex method exist which provide $\mathcal{W}^k$ via Simplex basis information.

Byrd et al. [Byr+13] describe an approach building on parametric linear programming that samples several trust-region choices to promote quick active-set identification.

Step i. of 7.1 requires solving

$$\min_{\substack{\|d\|_\infty \le \Delta_\ell^k, \\ \max\{0, s^k - \Delta_\ell^k\} \le s^k + d_s \perp t^k + d_t \ge \max\{0, t^k - \Delta_\ell^k\}}} \ell_k(d), \tag{7.2}$$

where by choice of F and ω the linearized model ℓ_k is piecewise linear and given by the expression

$$\begin{aligned} \ell_k(d) &= f^k + \Delta_f + \gamma^k \Delta_{\mathcal{E}} + \gamma^k \Delta_{\mathcal{I}}, \\ \Delta_f &= \langle \nabla f^k, d \rangle, \\ \Delta_{\mathcal{E}} &= \| c_{\mathcal{E}}^k + \langle \nabla c_{\mathcal{E}}^k, d \rangle \|_1, \\ \Delta_{\mathcal{I}} &= \| \min\{0, c_{\mathcal{I}}^k + \langle \nabla c_{\mathcal{I}}^k, d \rangle\} \|_1. \end{aligned}$$

By introduction of slack variables u^+, u^- and v^-, (7.2) can be reformulated as Linear

Problem with Equilibrium Constraints:

$$\begin{aligned}\min_{d,u^+,u^-,v^-} \quad & \langle \nabla f^k, d\rangle + \gamma^k\langle 1, u^+ + u^-\rangle + \gamma^k\langle 1, v^-\rangle \\ \text{s.t.} \quad & c_{\mathcal{E}}^k + \langle \nabla c_{\mathcal{E}}^k, d\rangle = u^+ - u^-, \\ & c_{\mathcal{I}}^k + \langle \nabla c_{\mathcal{I}}^k, d\rangle \geq -v^-, \\ & \|d\|_\infty \leq \Delta_\ell^k, \\ & \max\{0, s^k - \Delta_\ell^k\} \leq s^k + d_s \perp t^k + d_t \geq \max\{0, t^k - \Delta_\ell^k\}, \\ & u^+, u^-, v^- \geq 0.\end{aligned}$$

The *working set* $\mathcal{W}^k$ is defined as a maximal linearly independent active subset of the active set $\mathcal{A}(z^k + d_\ell^k)$ at the solution of (7.2), omitting active trust-region constraints. Solving the Linear Problem with Equilibrium Constraints with an active-set method gives besides the step also a suitable $\mathcal{W}^k$.

The penalty parameter adaption is not based on classical approaches by comparison to Lagrange multiplier size, but on a steering-rules-based heuristic that ensures that if the linearized constraints are feasible, the penalty parameter is large enough that they are satisfied and otherwise provide a sufficient decrease in infeasibility. Denote $d_\ell(\gamma)$ the solution of (7.2) with γ instead of γ^k and $d_\ell(\infty)$ the solution of (7.2) with ∇f^k replaced by 0. Then the infeasibility of γ is defined by

$$\text{infeas}_k(\gamma) := \tfrac{1}{|\mathcal{E}\cup\mathcal{I}|}\left(\|c_{\mathcal{E}}^k + \langle \nabla c_{\mathcal{E}}^k, d_\ell(\gamma)\rangle\|_1 + \|\min\{0, c_{\mathcal{I}}^k + \langle \nabla c_{\mathcal{I}}^k, d_\ell(\gamma)\rangle\}\|_1\right).$$

Every evaluation of $\text{infeas}_k(\gamma)$ involves one solution of (7.2).

Using this notion and a constant $0 < \varepsilon < 1$, the sufficient decrease condition is given by

$$\text{infeas}_k(\gamma^{k-1}) - \text{infeas}_k(\gamma^k) \geq \varepsilon(\text{infeas}_k(\gamma^{k-1}) - \text{infeas}_k(\infty)) \tag{7.3}$$

and the employed heuristic can be made precise:

$$\gamma^k := \begin{cases} a^{i_k}\gamma^{k-1}, & \text{infeas}_k(\infty) < \texttt{tol}, \\ \gamma^{k-1}, & \text{infeas}_k(\gamma^{k-1}) - \text{infeas}_k(\infty) < \texttt{tol}, \\ & \text{infeas}_j(\infty) < \texttt{tol} \text{ and} \\ \|(\lambda_{\text{LS}}^k, \mu_{\text{LS}}^k)\|_\infty, & \|(\lambda_{\text{LS}}^j, \mu_{\text{LS}}^j)\|_\infty > 10^3(1 + \|(\lambda_{\text{LS}}^j, \mu_{\text{LS}}^j)\|_\infty), \\ & j = k-5, \ldots, k-1 \\ a^{j_k}\gamma^{k-1}, & \text{else,} \end{cases}$$

with $i_k := \min\{m \geq 0 \,|\, \text{infeas}_k(a^m\gamma^{k-1}) < \texttt{tol}\}$ and $j_k := \min\{m \geq 0 \,|\, (7.3) \text{ holds}\}$. Here $\texttt{tol} \geq 0$ denotes a comparison tolerance and $a > 1$ an increase factor. The first

case eventually increases the penalty parameter by a factor a until the linearized constraints are satisfied, provided that they are feasible. The second case keeps the penalty parameter if no reduction in infeasibility is possible. In the third case, the penalty parameter is reset to the size of Lagrange multipliers if, for five consecutive iterations, it is much larger than the Lagrange multipliers. In the final case, the penalty parameter is increased by a factor a until the sufficient decrease condition (7.3) is satisfied. An analysis of this penalty update scheme is given by Byrd et al. [BNW08].

In our implementation the choices $\gamma^0 = 10$, $\varepsilon = 10^{-1}$, $a = 10$ and $\texttt{tol} = 10^{-8}$ are made.

Least-Squares Multiplier Estimation (Step b)

Although the linear phase already provides a multiplier estimation, a least-squares multiplier estimation is computed that is optimal in the sense that is satisfies (KKT) as well as possible. The cost of doing this is one backsolve with a matrix that nevertheless has be to factorized in the solution of the Equality Constrained Quadratic Program (EQP), if the EQP is solved with preconditioner being identity.

The minimal residual problem is given by

$$\begin{aligned} \min_{\lambda,\mu,\nu,\sigma} \quad & \|\nabla_z L_\perp(z^k, \lambda, \mu, \nu, \sigma)\|^2 \\ \text{s.t.} \quad & \lambda_i = 0, i \notin \mathcal{W}^k, \\ & \mu_i = 0, i \notin \mathcal{W}^k, \\ & \nu_i = 0, i \notin \mathcal{W}^k, \\ & \sigma_i = 0, i \notin \mathcal{W}^k. \end{aligned}$$

Denote the solution of this problem by $\lambda, \mu, \nu, \sigma$. The least squares estimation is then defined by

$$\begin{aligned} \lambda_{\text{LS}} &:= \lambda, \\ \mu_{\text{LS}} &:= \max\{0, \mu\}, \\ \nu_{\text{LS}} &:= \nu, \\ \sigma_{\text{LS}} &:= \sigma. \end{aligned}$$

Denoting by A_k the matrix composed from the gradients corresponding to indices in $\mathcal{W}^k$, the nonzero components $\Lambda_{\mathcal{W}^k}$ of the tuple $\lambda, \mu, \nu, \sigma$ are obtained as

$$\Lambda_{\mathcal{W}^k} = \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} I & A_k^T \\ A_k & 0 \end{pmatrix}^{-1} \begin{pmatrix} -\nabla f^k \\ 0 \end{pmatrix}.$$

Quadratic Model and Cauchy Step (Step c)

For the quadratic model, we choose the exact Hessian of the Lagrangian $H_k := \nabla^2_{zz} L_\perp(z^k, \lambda_{\mathrm{LS}}, \mu_{\mathrm{LS}}, \nu_{\mathrm{LS}}, \sigma_{\mathrm{LS}})$ with the least-squares multiplier estimation. This choice has the advantage that it defines a quadratic model of the original problem (MPEC) rather than that of the penalized problem (PenEC). The Cauchy step $d^k_{\mathrm{C}} = \alpha^k d^k_\ell$ that is crucial for global convergence is now computed by backtracking linesearch until the condition $\phi(z^k) - q_k(\alpha^k d^k_\ell) \geq \eta(\phi(z^k) - \ell(\alpha^k d^k_\ell))$ is met with the choice $\eta = 10^{-1}$ in our implementation.

Trial Step by solving an EQP (Step d)

We compute a Newton-like trial step by solving an equality constrained quadratic program (EQP) in a trust-region. Instead of H_k used for the quadratic model in the globalization, we incorporate information on violated constraints not covered by the working set in the quadratic model for the EQP step. To this end we set $H^{\mathrm{EQP}}_k := \nabla^2_{zz} L_\perp(z^k, \lambda^k_{\mathrm{EQP}}, \mu^k_{\mathrm{EQP}}, \nu^k_{\mathrm{LS}}, \sigma^k_{\mathrm{LS}})$ with

$$(\lambda_{\mathrm{EQP}})^k_i := \begin{cases} (\lambda^k_{\mathrm{LS}})_i, & i \in \mathcal{W}^k \text{ or } c^k_i = 0, \\ -\operatorname{sgn}(c_i + \langle \nabla c_i, \alpha_m d_m \rangle)\, \gamma^k, & i \notin \mathcal{W}^k \text{ and } c^k_i \neq 0, \end{cases}$$

$$(\mu_{\mathrm{EQP}})^k_i := \begin{cases} (\mu^k_{\mathrm{LS}})_i, & i \in \mathcal{W}^k \text{ or } c^k_i \geq 0, \\ \gamma^k, & i \notin \mathcal{W}^k \text{ and } c^k_i < 0, \end{cases}$$

where $\alpha_m d_m$ will be defined below. This definition ensures that the terms in the quadratic model for violated constraints not covered by the working set constraints are a quadratic approximation of the corresponding constraint contribution in the penalized objective.

To solve the trust-region EQP

$$\begin{aligned} \min_d \quad & \langle \nabla f^k, d \rangle + \tfrac{1}{2} \langle d, H^{\mathrm{EQP}}_k d \rangle \\ \text{s.t.} \quad & A_k d + c^k_{\mathcal{W}^k} = 0, \\ & \|d\| \leq \Delta^k, \end{aligned}$$

we use a combination of the inexact decomposition approach of Byrd and Omojokun [Byr87; Omo89; LNP98] and the Krylov subspace approach for equality constrained problems of Gould et al. [GHN01]. A solution of the of equality constraint $A_k d = -c^k_{\mathcal{W}^k}$ is decomposed $d = \alpha_m d_m + d_z$ into the minimum norm solution

$$d_m = \begin{pmatrix} I & 0 \end{pmatrix} \begin{pmatrix} I & A_k^T \\ A_k & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ -c^k_{\mathcal{W}^k} \end{pmatrix}$$

and step d_z in the nullspace of A_k that is normal to d_m. Choosing the stepsize $\alpha_m = \min\{1, \tau\frac{\Delta}{\|d_m\|}\}$ with $0 < \tau < 1$ satisfies the trust-region constraint in the interior of the trust-region sphere. The normal step d_z is now given as solution of

$$\begin{aligned} \min_{d_z} \quad & \langle \nabla f^k + \alpha_m A_k d_m, d_z\rangle + \tfrac{1}{2}\langle d_z, H_k^{\mathrm{EQP}} d_z\rangle \\ \text{s.t.} \quad & A_k d_z = 0, \\ & \|d_z\|^2 \leq (\Delta^k)^2 - \alpha_m^2 \|d_m\|^2. \end{aligned}$$

This nullspace trust-region problem is solved with the Generalized Lanczos method that is discussed in detail in Chapter 8. This is a preconditioned Krylov subspace method, in which the nullspace constraint is satisfied by using the orthogonal projection onto $\ker A_k$ given by

$$P_I = \begin{pmatrix} I & 0 \end{pmatrix} \begin{pmatrix} I & A_k^T \\ A_k & 0 \end{pmatrix}^{-1} \begin{pmatrix} I \\ 0 \end{pmatrix}$$

as preconditioner. Computing the projection requires factorizing $\begin{pmatrix} I & A_k^T \\ A_k & 0 \end{pmatrix}$, the same matrix that was required to compute the Least-Squares Multiplier Estimation and the step d_m. An additional precondition can be applied by using the M^{-1} orthogonal projection

$$P_M = \begin{pmatrix} I & 0 \end{pmatrix} \begin{pmatrix} M & A_k^T \\ A_k & 0 \end{pmatrix}^{-1} \begin{pmatrix} I \\ 0 \end{pmatrix}$$

instead of P_I for a matrix M that is positive definite. This implies replacing the trust-region norm $\|\cdot\|$ by $\|\cdot\|_M : x \mapsto \sqrt{\langle x, Mx\rangle}$. By norm-equivalence on a finite-dimensional space, the convergence proof still works mutatis mutandis and the convergence result holds. The least-squares multiplier estimation must still be computed as described in the step c. if weak stationarity is measured in the $\|\cdot\|$ norm, and must be computed by

$$\Lambda^M_{\mathcal{W}^k} = \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} M & A_k^T \\ A_k & 0 \end{pmatrix}^{-1} \begin{pmatrix} -\nabla f^k \\ 0 \end{pmatrix}$$

if weak stationarity is measured in the $\|\cdot\|_{M^{-1}}$ norm.

Having computed the EQP step $d_{\mathrm{EQP}} = \alpha_m d_m + z$, a trial step d^k that makes progress at least as good as the Cauchy step has to be selected. To this end, a back-tracking linesearch of q_k along the segment $\alpha \in [0,1] \mapsto d_{\mathrm{C}}^k + \alpha(d_{\mathrm{EQP}} - d_{\mathrm{C}}^k)$ is used. The trial-step is accepted as defined in steps iv. and vii. of Algorithm 7.1.

In our implementation we use $\tau = 0.8$ and $M = I$ with the exception of the Gauß-Newton NMPC applications. We employ $\|\cdot\|$ always to measure weak stationarity in all cases. The step acceptance constants chosen are $\rho^{\mathrm{u}} = \rho^{\mathrm{s}} = 10^{-8}$.

Second Order Correction Step (Step e)

If the trial step is rejected by vii. of Algorithm 7.1, we compute another trial step by second order correction as suggested by Fletcher [Fle87]. This is a safeguard against the Maratos effect [Mar78] describing the phenomenon that steps that make good progress towards the solution can be rejected. The second order correction step that updates the trial direction is given by minimum norm solution of $A_k d + c_{\mathcal{W}^k}(z^k + d^k) = 0$ computed by $d_s = \begin{pmatrix} I & 0 \end{pmatrix} \begin{pmatrix} M & A_k^T \\ A_k & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ -c_{\mathcal{W}^k}(z^k + d^k) \end{pmatrix}$.

Trust-Region Radii Update (Step f and g)

In our implementation the following trust-region updates are used that satisfy the requirements of steps v. and vi. of Algorithm 7.1:

$$\Delta^{k+1} = \begin{cases} \max\{\Delta^k, 7\|d^k\|\}, & \rho^k \geq 0.9, \\ \max\{\Delta^k, 2\|d^k\|\}, & 0.9 > \rho^k \geq 0.3, \\ \Delta^k, & 0.3 \geq \rho^k \geq 10^{-8}, \\ \frac{1}{2}\min\{\Delta^k, \|d^k\|\}, & \rho^k < 10^{-8} \end{cases}$$

$$\Delta_\ell^{k+1} = \begin{cases} \min\{\max\{1.2\|d^k\|_\infty, 1.2\|d_C^k, 0.1\Delta_\ell^k\}, 7\Delta_\ell^k\}, & \rho^k \geq 10^{-8} \text{ and } d_C^k = d_\ell^k, \\ \min\{\max\{1.2\|d^k\|_\infty, 1.2\|d_C^k, 0.1\Delta_\ell^k\}, \Delta_\ell^k\}, & \rho^k \geq 10^{-8} \text{ and } d_C^k \neq d_\ell^k, \\ \min\{\max\{0.5\|d^k\|_\infty, 0.1\Delta_\ell^k\}, \Delta_\ell^k\}, & \rho^k < 10^{-8}. \end{cases}$$

7.6. Remarks

The practical algorithm is a member of the algorithms described by the framework 7.1 and consequently shares the convergence property towards B-stationary points under assumptions T, C, L and B. It incorporates a Newton-like step where exact and possibly indefinite Hessians can be used leading to quadratic convergence if the active set estimation $\mathcal{W}^k$ becomes stationary. The method to solve the quadratic subproblem is iterative and thus requires access to the Hessian only in operator form, i.e. via evaluations $v \mapsto Hv$ which can be favorly exploited in optimal control applications.

In computational experience it turned out that the usage of the Hessian with modified Lagrange multipliers incorporating information about violated constraints not covered by the working set estimate is important to make quick progress towards feasibility. If the penalty parameter γ^k becomes large this may however force the algorithm to prioritize establishing feasibility over optimality.

The algorithm is built on a trust-region globalization framework of the penalized problem (PenEC) and thus has the drawback of not being affine invariant [Deu74; Deu06]. We have observed that it is very sensitive to variable and constraint scaling, and it is crucial that applications treated with the algorithm are carefully scaled.

7.7. Summary

In this chapter, we have extended the `SLIQUE` algorithm of Byrd et al. towards (MPEC). The presented algorithm constitutes a novel method for solving (PenEC), which includes (MPEC) via penalty reformulation of nonlinear constraints. The algorithm employs a trust-region globalization and makes use of linear phase and a step determination phase that finds a step that is at least as good as the Cauchy step. Convergence to B-stationary points under assumptions T, C, L and B is proven, which distinguishes this algorithm from the majority of methods for (MPEC) that only ensure satisfaction of weaker stationarity concepts.

A practical method of the algorithm is presented and its implementation forms the basis for the numerical results presented in Part III.

8. Trust Region Problems in Hilbert Space

A fast and reliable method for solving the trust region subproblem is an important ingredient in the SLPECEQP algorithm of Chapter 7.5. In this chapter, we present Gould's Generalized Lanczos Method and extend it towards trust region problems in Hilbert spaces. We have developed an additional heuristic addressing ill-conditioning.

The results of this chapter are published in [LKP16].

8.1. Trust Region Subproblem

In this chapter, we are concerned with solving the trust region problem, as it frequently arises as a subproblem in sequential algorithms for nonlinear optimization.

8.1 Definition (Trust Region Subproblem).
Let $(\mathcal{H}, \langle\cdot,\cdot\rangle)$ be a Hilbert space. Let $H : \mathcal{H} \to \mathcal{H}$ be self-adjoint and $M : \mathcal{H} \to \mathcal{H}$ be self-adjoint, bounded and coercive. Let $\langle\cdot,\cdot\rangle_M$ be the inner product induced by M via $\langle x, y\rangle_M := \langle x, My\rangle$ and $\|\cdot\|_M$ the corresponding norm. Let $\mathcal{X} \subseteq \mathcal{H}$ be a closed subspace. Let $\Delta > 0$ and $g \in \mathcal{H}$.

Then the *trust region subproblem* defined by H, g, M, Δ and $\mathcal{X}$ is

$$\begin{aligned} \min_{x\in\mathcal{H}} \quad & \tfrac{1}{2}\langle x, Hx\rangle + \langle x, g\rangle \\ \text{s.t.} \quad & \|x\|_M \le \Delta, \\ & x \in \mathcal{X}. \end{aligned} \tag{TR$(H, g, M, \Delta, \mathcal{X})$}$$

We denote by $q(x) := \frac{1}{2}\langle x, hx\rangle + \langle x, g\rangle$ the objective function. △

8.2. Survey on Unconstrained Trust Region Problems

Trust Region Subproblems are an important ingredient in modern optimization algorithms as globalization mechanism. The monograph [CGT00] provides an exhaustive overview on Trust Region Methods for nonlinear programming, mainly for problems

formulated in finite-dimensional spaces. For trust region algorithms in Hilbert spaces, we refer to [KS87; Toi88; Hei93; UU00]. In [ABG07] applications of trust region subproblems formulated on Riemannian manifolds are considered. Recently, trust region-like algorithms with guaranteed complexity estimates in relation to the KKT tolerance have been proposed [CGT11a; CGT11b; CS16]. The necessary ingredients in the subproblem solver for the algorithm investigated by Curtis and Samadi [CS16] have been implemented our implementation `trlib` presented in Chapter 11.

Solution algorithms for trust region problems can be classified into direct algorithms that make use of matrix factorizations and iterative methods that access the operators H and M only via evaluations $x \mapsto Hx$ and $x \mapsto Mx$ or $x \mapsto M^{-1}x$. For the Hilbert space context and the application to the SLPECEQP algorithm of Chapter 7.5, we are interested in the latter class of algorithms and in particular to Krylov subspace based algorithms as these can ensure equality constraints via preconditioner.

We refer to [CGT00] and the references therein for a survey of direct algorithms, but point out the algorithm of Moré and Sorensen [MS83] that will be used on a specific tridiagonal subproblem, as well as the work of Gould et al. [GT10], who use higher order Taylor models to obtain high order convergence results. The first iterative method was based on the conjugate gradient process, and was proposed independently by Toint [Toi81] and Steihaug [Ste83]. Gould et al. [Gou+99] proposed an extension of the Steihaug-Toint algorithm. There, the Lanczos algorithm is used to build up a nested sequence of Krylov spaces, and tri-diagonal trust region subproblems are solved with a direct method. Hager [Hag01] considers an approach that builds on solving the problem restricted to a sequence of subspaces that use SQP iterates to accelerate and ensure quadratic convergence. Erway et al. [EGG09; EG10] investigate a method that also builds on a sequence of subspaces built from accelerator directions satisfying optimality conditions of a primal-dual interior point method. In the methods of Steihaug-Toint and Gould, the operator M is used to define the trust region norm and acts as preconditioner in the Krylov subspace algorithm. The method of Erway et al. allows to use a preconditioner that is independent of the operator used for defining the trust region norm. The trust region problem can equivalently be stated as generalized eigenvalue problem. Approaches based on this characterization are studied by Sorensen [Sor97], Rendl and Wolkowicz [RW97] Rojas et al. [RSS01; RSS08] and Adachi et al. [Ada+17].

8.3. Existence and Uniqueness of Minimizers

In this section, we briefly summarize the main results about existence and uniqueness of solutions of the trust region subproblem. To prove existence of minimizers, we

have to impose a certain compactness condition on H:

8.2 Definition (Compact negative part).
A self-adjoint, bounded operator $H : \mathcal{H} \to \mathcal{H}$ has *compact negative part*, if there are self-adjoint, bounded operators P and K with $H = P - K$ and $\langle x, Px\rangle \geq 0$ and K is compact. △

From now on we impose the following setting that will ensure existence of a solution:

8.3 Assumption.
$(\mathcal{H}, \langle\cdot,\cdot\rangle)$ is a Hilbert space. The operator $H : \mathcal{H} \to \mathcal{H}$ is self-adjoint, bounded and with compact negative part. The operator $M : \mathcal{H} \to \mathcal{H}$ is self-adjoint, bounded and coercive. The trust region radius $\Delta > 0$ is positive, $g \in \mathcal{H}$ and $\mathcal{X} \subseteq \mathcal{H}$ is a closed subspace. △

8.4 Lemma (Properties of $\mathrm{TR}(H, g, M, \Delta, \mathcal{X})$**).**

(1) The mapping $x \mapsto \langle x, Hx\rangle$ is sequentially weakly lower semicontinuous, and Fréchet differentiable for every $x \in \mathcal{H}$.

(2) The feasible set $\mathcal{F} := \{x \in \mathcal{H} \mid \|x\|_M \leq \Delta\}$ is bounded and weakly closed.

(3) The operator M is surjective.

Proof. $H = P - K$ with compact K, so (1) follows from [Hes51, Thm 8.2]. Fréchet differentiability follows from boundedness of H. Boundedness of $\mathcal{F}$ follows from coercivity of M. Furthermore, $\mathcal{F}$ is obviously convex and strongly closed, hence weakly closed. Finally, (3) follows by the Lax-Milgram Theorem [Cla13, ex. 7.19]: By boundedness of M, there is $C > 0$ with $|\langle x, My\rangle \leq C\|x\|\,\|y\|$. The coercivity assumption implies existence of $c > 0$ such that $\langle x, Mx\rangle \leq c\|x\|^2$ for all $x, y \in \mathcal{H}$. Then, M satisfies the assumptions of the Lax-Milgram Theorem. Given $z \in \mathcal{H}$, application of this Theorem yields $\xi \in \mathcal{H}$ with $\langle x, M\xi\rangle = \langle x, z\rangle$ for all $x \in \mathcal{H}$. Thus $M\xi = z$. □

8.5 Lemma (Existence of a solution).
Problem $(\mathrm{TR}(H, g, M, \Delta, \mathcal{X}))$ *has a solution.*

Proof. By Lemma 8.4, the objective functional q is sequentially weakly lower semicontinuous and the feasible set $\mathcal{F}$ is weakly closed and bounded, the claim follows then from a generalized Weierstrass Theorem [KZ06, Ch. 7]. □

To present optimality conditions for the trust region subproblem, we first present a helpful lemma on the change of the objective function between two points on the trust region boundary.

8.6 Lemma (Objective Change on Trust Region Boundary).
Let $x^0, x^1 \in \mathcal{H}$ with $\|x^i\|_M = \Delta$ for $i = 0, 1$ be boundary points of $(\mathrm{TR}(H, g, M, \Delta, \mathcal{X}))$*, and let $\lambda \geq 0$ satisfy $(H + \lambda M)x^0 + g = 0$. Then $d = x^1 - x^0$ satisfies $q(x^1) - q(x^0) = \frac{1}{2}\langle d, (H + \lambda M)d\rangle$.*

Proof. Using $0 = \|x^1\|_M^2 - \|x^0\|_M^2 = \langle x^0 + d, M(x^0 + d)\rangle - \langle x^0, Mx^0\rangle = \langle d, Md\rangle + 2\langle x^0, Md\rangle$ and $g = -(H + \lambda M)x^0$ we find

$$\begin{aligned} q(x^1) - q(x^0) &= \tfrac{1}{2}\langle d, Hd\rangle + \langle d, Hx^0\rangle + \langle g, d\rangle = \tfrac{1}{2}\langle d, Hd\rangle \overbrace{\lambda\langle x^0, Md\rangle}^{-\frac{1}{2}\lambda\langle d, Md\rangle} \\ &= \tfrac{1}{2}\langle d, (H + \lambda M)d\rangle. \end{aligned}$$

□

Necessary optimality conditions for the finite dimensional problem, see e.g. [CGT00], generalize in a natural way to the Hilbert space context.

8.7 Theorem (Necessary Optimality Conditions).
Let $x^* \in \mathcal{H}$ be a global solution of $(\mathrm{TR}(H, g, M, \Delta, \mathcal{H}))$*. Then there is $\lambda^* \geq 0$ such that*

(1) $(H + \lambda^* M)x^* + g = 0$,

(2) $\|x^*\|_M - \Delta \leq 0$,

(3) $\lambda^*(\|x^*\|_M - \Delta) = 0$,

(4) $\langle d, (H + \lambda^* M)d\rangle \geq 0$ *for all* $d \in \mathcal{H}$.

Proof. Let $\sigma : \mathcal{H} \to \mathbb{R}, \sigma(x) := \langle x, Mx\rangle - \Delta^2$, so that the trust region constraint becomes $\sigma(x) \leq 0$. The function σ is Fréchet-differentiable for all $x \in \mathcal{H}$ with surjective differential provided $x \neq 0$ and satisfies constraint qualifications in that case. We may assume $x^* \neq 0$ as the Theorem holds for $x^* = 0$ (then $g = 0$) for elementary reasons.

Now if x^* is a solution of $(\mathrm{TR}(H, g, M, \Delta, \mathcal{H}))$, conditions (1)–(3) are necessary optimality conditions, cf. [Cla13, Thm 9.1].

To prove (4), we distinguish three cases:

First, suppose $\|x\|_M = \Delta$ and $d \in \mathcal{H}$ with $\langle d, Mx^*\rangle \neq 0$: Given such d, there is $\alpha \in \mathbb{R} \setminus \{0\}$ with $\|x^* + \alpha d\|_M = \Delta$. Using Lemma 8.6 yields $\langle d, (H + \lambda^* M)d\rangle = \frac{2}{\alpha^2}(q(x^* + \alpha d) - q(x^*)) \geq 0$ since x^* is a solution.

Second, assume $\|x\|_M = \Delta$ and $d \in \mathcal{H}$ with $\langle d, Mx^*\rangle = 0$: Since $x^* \neq 0$ and M is surjective, there is $p \in H$ with $\langle p, Mx^*\rangle \neq 0$, let $d(\tau) := d + \tau p$ for $\tau \in \mathbb{R}$. Then $\langle d(\tau), Mx^*\rangle \neq 0$ for $\tau \neq 0$, by the previous case

$$\begin{aligned} 0 &\leq \langle d(\tau), (H + \lambda^* M)d(\tau)\rangle \\ &= \langle d, (H + \lambda^* M)d\rangle + \tau\langle p, (H + \lambda^* M)d\rangle + \tau^2\langle p, (H + \lambda^* M)p\rangle. \end{aligned}$$

Passing to the limit $\tau \to 0$ shows $\langle d, (H + \lambda^* M)d\rangle \geq 0$.

In the final third case assume $\|x\|_M < \Delta$: Then $\lambda^* = 0$ by (c). Let $d \in \mathcal{H}$ and consider $x(\tau) = x^* + \tau d$, which is feasible for sufficiently small τ. By optimality and stationarity (a):

$$0 \leq q(x(\tau)) - q(x^*) = \tau\langle x^*, Hd\rangle + \tfrac{\tau^2}{2}\langle d, Hd\rangle + \tau\langle g, d\rangle = \tfrac{\tau^2}{2}\langle d, Hd\rangle,$$

thus $\langle d, Hd\rangle \geq 0$. □

8.8 Corollary (Sufficient Optimality Condition).
Let $x^* \in \mathcal{H}$ and $\lambda^* \geq 0$ such that (a)–(c) of Theorem 8.7 hold and $\langle d, (H + \lambda^* M)d\rangle > 0$ holds for all $d \in \mathcal{H}$. Then x^* is the unique global solution of (TR($H, g, M, \Delta, \mathcal{H}$)).

Proof. This is an immediate consequence of Lemma 8.6. □

8.4. The Generalized Lanczos Method

The Generalized Lanczos Trust Region (GLTR) method is an iterative method to approximately solve (TR($H, g, M, \Delta, \mathcal{H}$)) and has first been described in Gould et al. [Gou+99]. In every iteration of the GLTR process, problem TR($H, g, M, \Delta, \mathcal{H}$) is restricted to the Krylov subspace $\mathcal{K}_i := \operatorname{span}\{(M^{-1}H)^j M^{-1} g \mid 0 \leq j \leq i\}$,

$$\begin{array}{ll} \min\limits_{x \in \mathcal{H}} & \frac{1}{2}\langle x, Hx\rangle + \langle x, g\rangle \\ \text{s.t.} & \|x\|_M \leq \Delta, \\ & x \in \mathcal{K}_i. \end{array} \qquad (\text{TR}(H, g, M, \Delta, \mathcal{K}_i))$$

The following Lemma relates solutions of (TR($H, g, M, \Delta, \mathcal{K}_i$)) to those of the original problem TR($H, g, M, \Delta, \mathcal{H}$).

8.9 Lemma (Solution of the Krylov subspace trust region problem).
Let x^i be a global minimizer of (TR($H, g, M, \Delta, \mathcal{K}_i$)) *and λ^i the corresponding Lagrange multiplier. Then (x^i, λ^i) satisfies the global optimality conditions of* (TR($H, g, M, \Delta, \mathcal{H}$)), *Theorem 8.7, in the following sense:*

(1) $(H + \lambda^i M)x^i + g \perp_M \mathcal{K}_i$,

(2) $\|x^i\|_M - \Delta \leq 0$,

(3) $\lambda^i(\|x^i\|_M - \Delta) = 0$,

(4) $\langle d, (H + \lambda^i M)d\rangle \geq 0$ *for all* $d \in \mathcal{K}_i$.

PROOF. (2)–(4) are immediately obtained from Theorem 8.7 as $\mathcal{K}_i \subseteq \mathcal{H}$ is a Hilbert space. Assertion (1) follows from $x^* = x^i + x^\perp$ with $x^i \in \mathcal{K}_i$, $x^\perp \perp \mathcal{K}_i$ and Theorem 8.7 for x^i. □

Solving problem (TR($H, g, M, \Delta, \mathcal{H}$)) may thus be achieved by iterating the following Krylov subspace process. Each iteration requires the solution of an instance of the truncated trust region subproblem (TR($H, g, M, \Delta, \mathcal{K}_i$)).

Algorithm 8.1: Krylov subspace process for solving (TR($H, g, M, \Delta, \mathcal{H}$)).

input : H, M, g, Δ, `tol`
output: i, x^i, λ^i

for $i \geq 0$ **do**
  Construct a basis for the i-th Krylov subspace $\mathcal{K}_i$
  Compute a representation of $q(x)$ restricted to $\mathcal{K}_i$
  Solve the subproblem (TR($H, g, M, \Delta, \mathcal{K}_i$)) to obtain (x^i, λ^i)
  if $\|(H + \lambda^i M)x^i + g\|_{M^{-1}} \leq$ `tol` **then return**
end

Algorithm 8.1 stops the subspace iteration as soon as $\|(H + \lambda^i M)x^i + g\|_{M^{-1}}$ is small enough. The norm $\|\cdot\|_{M^{-1}}$ is used in the termination criterion since it is the norm belonging to the dual of $(\mathcal{H}, \|\cdot\|_M)$ and the Lagrange gradient $(H + \lambda^i M)x^i + g$ should be regarded as element of the dual.

8.4.1. Krylov Subspace Buildup

In this section, we present the preconditioned conjugate gradient (pCG) process and the preconditioned Lanczos process (pL) for construction of Krylov subspace bases. We discuss the transition from pCG to pL upon breakdown of the pCG process.

Preconditioned Conjugate Gradient Process

An H-conjugate basis $(\hat{p}_j)_{0 \leq j \leq i}$ of $\mathcal{K}_i$ may be obtained using preconditioned conjugate gradient (pCG) iterations, Algorithm 8.2.

The stationary point s^i of $q(x)$ restricted to the Krylov subspace $\mathcal{K}_i$ is given by $s^i = \sum_{j=0}^{i} \alpha^j \hat{p}^j$ and can thus be computed using the recurrence

$$s^0 \leftarrow \alpha^0 \hat{p}^0, \quad s^{j+1} \leftarrow s^j + \alpha^{j+1} \hat{p}^{j+1}, \ 0 \leq j \leq N - 1$$

as an extension of Algorithm 8.2. The iterates' M-norms $\|s^i\|_M$ are monotonically increasing [Ste83, Thm 2.1]. Hence, as long as H is coercive on the subspace $\mathcal{K}_i$ (this

Algorithm 8.2: Preconditioned conjugate gradient (pCG) process.

input : H, M, g^0, $i \in \mathbb{N}$
output: v^i, g^i, p^i, α^{i-1}, β^{i-1}

Initialize $\hat{v}^0 \leftarrow M^{-1}g^0$, $\hat{p}^0 \leftarrow -\hat{v}^0$
for $j \leftarrow 0$ **to** $i-1$ **do**
  $\alpha^j \leftarrow \langle \hat{g}^j, \hat{v}^j \rangle / \langle \hat{p}^j, H\hat{p}^j \rangle = \|\hat{v}^j\|_M / \langle \hat{p}^j, H\hat{p}^j \rangle$
  $\hat{g}^{j+1} \leftarrow \hat{g}^j + \alpha^j H\hat{p}_j$
  $\hat{v}^{j+1} \leftarrow M^{-1}\hat{g}^{j+1}$
  $\beta^j \leftarrow \langle \hat{g}^{j+1}, \hat{v}^{j+1} \rangle / \langle \hat{g}^j, \hat{v}^j \rangle = \|\hat{v}^{j+1}\|_M^2 / \|\hat{v}^j\|_M^2$
  $\hat{p}^{j+1} \leftarrow -\hat{v}^{j+1} + \beta^j \hat{p}^j$
end

implies $\alpha_j > 0$ for all $0 \le j \le i$) and $\|s^i\|_M \le \Delta$, the solution to (TR$(H, g, M, \Delta, \mathcal{K}_i)$) is directly given by s^i. Breakdown of the pCG process occurs if $\alpha^i = 0$. In computational practice, if the criterion $|\alpha^i| \le \varepsilon$ is violated, where $\varepsilon \ge 0$ is a suitable small tolerance, it is possible – and necessary – to continue with Lanczos iterations, described next.

Preconditioned Lanczos Process

An M-orthogonal basis $(p_j)_{0 \le j \le i}$ of $\mathcal{K}_i$ may be obtained using the preconditioned Lanczos (pL) process, Algorithm 8.3, and permits to continue subspace iterations even after pCG breakdown.

Algorithm 8.3: Preconditioned Lanczos (pL) process.

input : H, M, g^0, $j \in \mathbb{N}$
output: v^i, g^i, p^{i-1}, γ^{i-1}, δ^{i-1}

Initialize $g^{-1} \leftarrow 0$, $\gamma^{-1} \leftarrow 1$, $v^0 \leftarrow M^{-1}g^0$, $p^0 \leftarrow v^0$
for $i \leftarrow 0$ **to** $j-1$ **do**
  $\gamma^j \leftarrow \sqrt{\langle g^j, v^j \rangle} = \|g^j\|_{M^{-1}} = \|v^j\|_M$
  $p^j \leftarrow (1/\gamma^j)v^j = (1/\|v^j\|_M)v^j$
  $\delta^j \leftarrow \langle p^j, Hp^j \rangle$
  $g^{j+1} \leftarrow Hp^j - (\delta^j/\gamma^j)g^j - (\gamma^j/\gamma^{j-1})g^{j-1}$
  $v^{j+1} \leftarrow M^{-1}g^{j+1}$
end

The following simple relationship holds between the Lanczos iteration data and the pCG iteration data, and may be used to initialize the pL process from the final pCG iterate before breakdown:

$$\gamma^i = \begin{cases} \|\hat{v}^0\|_M, & i = 0 \\ \sqrt{\beta^{i-1}}/|\alpha^{i-1}|, & i \geq 1 \end{cases}, \qquad \delta^i = \begin{cases} 1/\alpha^0, & i = 0 \\ 1/\alpha^i + \beta^{i-1}/\alpha^i, & i \geq 1 \end{cases},$$
$$p^i = 1/\|\hat{v}_i\|_M \left[\prod_{j=0}^{i-1}(-\operatorname{sgn}\alpha^j)\right] \hat{v}_i, \qquad g^i = \gamma^j/\|\hat{v}_i\|_M \left[\prod_{j=0}^{i-1}(-\operatorname{sgn}\alpha^j)\right] \hat{g}_i.$$

In turn, breakdown of the pL process occurs if an invariant subspace of H is exhausted, in which case $\gamma^i = 0$. If this subspace does not span $\mathcal{H}$, the pL process may be restarted if provided with a vector g^0 that is M-orthogonal to the exhausted subspace.

The pL process may also be expressed in compact matrix form as

$$HP_i - MP_iT_i = g^{i+1}\boldsymbol{e}_{\boldsymbol{i+1}}^T, \ \langle P_i, MP_i\rangle = I,$$

with P_i being the matrix composed from columns $p_0, \ldots, p_i$, and T_i the symmetric tridiagonal matrix with diagonal elements $\delta^0, \ldots, \delta^i$ and off-diagonal elements $\gamma^1, \ldots, \gamma^i$.

As P_i is a basis for $\mathcal{K}_i$, every $x \in \mathcal{K}_i$ can be written as $x = P_i\boldsymbol{h}$ with a coordinate vector $\boldsymbol{h} \in \mathbb{R}^{i+1}$. Using the compact form of the Lanczos iteration, one can immediately express the quadratic form in this basis as $q(x) = \frac{1}{2}\langle \boldsymbol{h}, T_i\boldsymbol{h}\rangle + \gamma^0\langle \boldsymbol{e_1}, \boldsymbol{h}\rangle$. Similarly, $\|x\|_M = \|\boldsymbol{h}\|_2$. Solving (TR$(H, g, M, \Delta, \mathcal{K}_i)$) thus reduces to solving TR$(T_i, \gamma^0\boldsymbol{e_1}, I, \Delta, \mathbb{R}^{i+1})$ on $\mathbb{R}^{i+1}$ and recovering $x = P_i\boldsymbol{h}$.

8.4.2. Easy and Hard case of the Tridiagonal Subproblem

As just described, using the tridiagonal representation T_i of H on the basis P_i of the i-th iteration of the pL process, the trust-region subproblem TR$(T_i, \gamma^0\boldsymbol{e_1}, I, \Delta, \mathbb{R}^{i+1})$ needs to be solved. For notational convenience, we drop the iteration index i from T_i in the following. Considering the necessary optimality conditions of Theorem 8.7, it is natural to define the mapping

$$\lambda \mapsto \boldsymbol{x}(\lambda) := (T + \lambda I)^+(-\gamma^0\boldsymbol{e_1}) \text{ for } \lambda \in I := [\max\{0, -\theta_{\min}\}, \infty),$$

where $\theta_{\min}$ denotes the smallest eigenvalue of T, and the superscript + denotes the Moore-Penrose pseudo-inverse. On I, $T + \lambda I$ is positive semidefinite. The following definition relates $\boldsymbol{x}(\lambda^*)$ to a global minimizer $(\boldsymbol{x^*}, \lambda^*)$ of TR$(T_i, \gamma^0\boldsymbol{e_1}, I, \Delta, \mathbb{R}^{i+1})$.

8.10 Definition (Easy Case and Hard Case).
Let $(\boldsymbol{x}^*, \lambda^*)$ satisfy the necessary optimality conditions of Theorem 8.7.
If $\langle \gamma^0 \boldsymbol{e_1}, \mathrm{Eig}(\theta_{\min})\rangle \neq 0$, we say that T satisfies the *easy case*. Then, $\boldsymbol{x}^* = \boldsymbol{x}(\lambda^*)$.
If $\langle \gamma^0 \boldsymbol{e_1}, \mathrm{Eig}(\theta_{\min})\rangle = 0$, we say that T satisfies the *hard case*. Then, $\boldsymbol{x}^* = \boldsymbol{x}(\lambda^*) + \boldsymbol{v}$ with suitable $\boldsymbol{v} \in \mathrm{Eig}(\theta_{\min})$. Here $\mathrm{Eig}(\theta) = \{\boldsymbol{v} \in \mathbb{R}^{i+1} \mid T\boldsymbol{v} = \theta\boldsymbol{v}\}$ denotes the eigenspace of T associated to θ. △

With the following Theorem, Gould et al. in [Gou+99] use the the irreducible components of T to give a full description of the solution $x(\lambda^*) + v$ in the hard case.

8.11 Theorem (Global Minimizer in the Hard Case).
Let $T = \mathrm{diag}(R_1, \ldots, R_k)$ with irreducible tridiagonal matrices R_j and let $1 \leq \ell \leq k$ be the smallest index for which $\theta_{\min}(R_\ell) = \theta_{\min}(T)$ holds. Further, let $\boldsymbol{x_1}(\theta) = (R_1 + \theta I)^+(-\gamma^0 \boldsymbol{e_1})$ and let $(\boldsymbol{x_1^*}, \lambda_1^*)$ be a KKT-tuple corresponding to a global minimum of $\mathrm{TR}(R_1, \gamma^0 \boldsymbol{e_1}, I, \Delta, \mathbb{R}^{r_1})$, $\boldsymbol{x_1^*} = \boldsymbol{x_1}(\lambda_1^*)$.

If $\lambda_1^* \geq -\theta_{\min}$, then $x^* = (\boldsymbol{x_1}(\lambda_1^*)^T,\ \boldsymbol{0},\ \ldots,\ \boldsymbol{0})^T$ satisfies Theorem 8.7 for the problem $\mathrm{TR}(T, \gamma^0 \boldsymbol{e_1}, I, \Delta, \mathbb{R}^{i+1})$.

If $\lambda_1^* < -\theta_{\min}$, then $x^* = (\boldsymbol{x_1}(-\theta_{\min})^T,\ \boldsymbol{0},\ \ldots,\ \boldsymbol{0},\ \boldsymbol{v}^T, \boldsymbol{0},\ \ldots,\ \boldsymbol{0})^T$, with an eigenvector $v \in \mathrm{Eig}(R_\ell, \theta_{\min})$ such that $\|\boldsymbol{x}^*\|_2^2 = \|\boldsymbol{x_1}(-\theta_{\min})\|_2^2 + \|\boldsymbol{v}\|_2^2 = \Delta^2$ satisfies Theorem 8.7 for $\mathrm{TR}(T, \gamma^0 \boldsymbol{e_1}, I, \Delta, \mathbb{R}^{i+1})$. △

In particular, as long as T is irreducible, the hard case does not occur. For the tridiagonal matrices arising from Krylov subspace iterations, this is the case as long as the pL process does not break down.

8.4.3. Solving the Tridiagonal Subproblem in the Easy Case

Assume that T is irreducible, and thus satisfies the easy case.. Solving the tridiagonal subproblem amounts to checking whether the problem admits an interior solution and, if not, to finding a value $\lambda^* \geq \max\{0, -\theta_{\min}\}$ with $\|x(\lambda^*)\| = \Delta$.

We follow Moré and Sorensen [MS83], who define $\sigma_p(\lambda) := \|\boldsymbol{x}(\lambda)\|^p - \Delta^p$ and propose the Newton iteration

$$\lambda^{i+1} \leftarrow \lambda^i - \sigma_p(\lambda^i)/\sigma_p'(\lambda^i) = \lambda^i - \frac{\|\boldsymbol{x}(\lambda^i)\|^p - \Delta^p}{p\|\boldsymbol{x}(\lambda^i)\|^{p-2}\langle \boldsymbol{x}(\lambda^i), \boldsymbol{x}'(\lambda^i)\rangle},\ i \geq 0,$$

with $\boldsymbol{x}'(\lambda) = -(T + \lambda I)^+ \boldsymbol{x}(\lambda)$, to find a root of $\sigma_{-1}(\lambda)$. Provided that the initial value λ^0 lies in the interval $[\max\{0, -\theta_{\min}\}, \lambda^*]$, such that $(T + \lambda^0 I)$ is positive semidefinite, $\|\boldsymbol{x}(\lambda^0)\| \geq \Delta$, and no safeguarding of the Newton iteration is necessary, it can be shown that this leads to a sequence of iterates in the same interval that converges to λ^* at globally linear and locally quadratic rate, cf. [Gou+99].

Note that $\lambda^* > -\theta_{\min}$ as $\sigma_{-1}(\lambda)$ has a singularity in $-\theta_{\min}$ but $\sigma_{-1}(\lambda^*) = 1/\Delta$ and it thus suffices to consider $\lambda > \max\{0, -\theta_{\min}\}$.

Both the function value and derivative require the solution of a linear system of the form $(T + \lambda I)\boldsymbol{w} = \boldsymbol{b}$. As $T + \lambda I$ is tridiagonal, symmetric positive definite, and of reasonably small dimension, it is computationally feasible to use a tridiagonal Cholesky decomposition for this.

Gould et al. in [GT10] improve upon the convergence result by considering higher order Taylor expansions of $\sigma_p(\lambda)$ and values $p \neq -1$ to obtain a method with locally quartic convergence.

8.4.4. The Newton initializer

Cheap oracles for a suitable initial value λ^0 may be available, including, for example, zero or the value λ^* of the previous iteration of the pL process. If these fail, it becomes necessary to compute $\theta_{\min}$. To this end, we follow Gould et al. [Gou+99] and Parlett and Reid [PR81], who define the Parlett-Reid Last-Pivot function $d(\theta)$:

8.12 Definition (Parlett-Reid Last-Pivot Function).

$$d(\theta) := \begin{cases} d_i, & \text{if there exists } (d_0, \dots, d_i) \in (0,\infty)^i \times \mathbb{R}, \text{ and } L \text{ unit lower} \\ & \text{triangular such that } T - \theta I = L \operatorname{diag}(d_0, \dots, d_i)\, L^T \\ -\infty, & \text{otherwise.} \end{cases}$$

△

Since T is irreducible, its eigenvalues are simple [GL96, Thm 8.5.1] and $\theta_{\min}$ is given by the unique value $\theta \in \mathbb{R}$ with $T - \theta I$ singular and positive semidefinite, or, equivalently, $d(\theta) = 0$.

A safeguarded root-finding method is used to determine $\theta_{\min}$ by finding the root of $d(\theta)$. An interval of safety $[\theta_\ell^k, \theta_u^k]$ is used in each iteration and a guess $\theta^k \in [\theta_\ell^k, \theta_u^k]$ is chosen. Gershgorin bounds may be used to provide an initial interval [GL96, Thm 7.2.1]. Depending on the sign of $d(\theta)$ the interval of safety is then contracted to $[\theta_\ell^k, \theta^k]$ if $d(\theta^k) < 0$ and to $[\theta^k, \theta_u^k]$ if $d(\theta^k) \geq 0$ as the interval of safety for the next iteration. One choice for θ^k is bisection. Newton steps as previously described may be taken advantage of if they remain inside the interval of safety.

For successive pL iterations, the fact that the tridiagonal matrices grow by one column and row in each iteration may be exploited to save most of the computational effort involved. As noted by Parlett and Reid [PR81], the recurrence to compute the d_i via Cholesky decomposition of $T - \theta I$ in Def. 8.12 is identical with the recurrence that results from applying a Laplace expansion for the determinant of tridiagonal

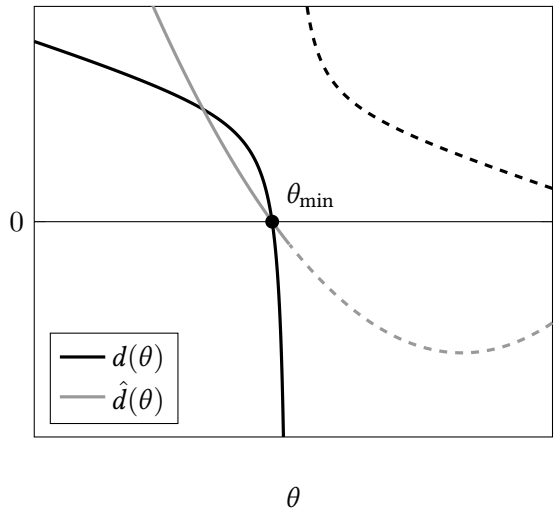


Figure 8.1.: The Parlett-Reid last-pivot function $d(\theta)$ and the lifted function $\hat{d}(\theta)$ have the common zero $\theta_{\min}$. Dashed lines show the analytic continuation of the right hand side of $d(\theta) = \prod_j(\theta - \theta_j)/\prod_j(\theta - \hat{\theta}_j)$ into the region where $d(\theta) = -\infty$.

matrices [GL96, §2.1.4]. Comparing the recurrences thus yields the explicit formula

$$d(\theta) = \frac{\det(T - \theta I)}{\det(\hat{T} - \theta I)} = -\frac{\prod_j(\theta - \theta_j)}{\prod_j(\theta - \hat{\theta}_j)}, \tag{8.1}$$

where $\hat{T}$ denotes the principal submatrix of T obtained by erasing the last column and row, and θ_j and $\hat{\theta}_j$ enumerate the eigenvalues of T and $\hat{T}$, respectively. The right hand side is obtained by identifying numerator and denominator with the characteristic polynomials of T and $\hat{T}$, and by factorizing these.

It becomes apparent that $d(\theta)$ has a pole of first order in $\hat{\theta}_{\min}$. After lifting this pole, the function $\hat{d}(\theta) := (\theta - \hat{\theta}_{\min})d(\theta)$ is smooth on a larger interval. When iteratively constructing the tridiagonal matrices in successive pL iterations, the value $\hat{\theta}_{\min}$ is readily available and it becomes preferable to use $\hat{d}(\theta)$ instead of $d(\theta)$ for root finding.

8.4.5. Solving the Tridiagonal Subproblem in the Hard Case

If the hard case is present, the decomposition of T into irreducible components has to be determined. This is given in a natural way by Lanczos breakdown. Every time the Lanczos process breaks down and is restarted with a vector M-orthogonal to

the previously considered Krylov subspaces, a new tridiagonal block is obtained. Solving the problem in the hard case then amounts to applying Theorem 8.11: First all smallest eigenvalue θ_i of the irreducible blocks R_i have to be determined as well as the KKT tuple $(\boldsymbol{x_1^*}, \lambda_1^*)$ by solving the easy case for $\mathrm{TR}(R_1, \gamma^0 \boldsymbol{e_1}, I, \Delta, \mathbb{R}^{r_1})$. Again, let ℓ be the smallest index i with minimal θ_i. In the case $\lambda_1^* \geq -\theta_\ell$, the global solution is given by $x^* = ((\boldsymbol{x_1^*})^T,\ \mathbf{0},\ \ldots, \mathbf{0})^T$. On the other hand if $\lambda_1^* < -\theta_\ell$ the eigenspace of R_ℓ corresponding to θ_ℓ has to be obtained. As R_ℓ is irreducible, all eigenvalues of R_ℓ are simple and an eigenvector $\tilde{\boldsymbol{v}}$ spanning the desired eigenspace can be obtained for example by inverse iteration [GL96, §8.2.2]. The solution is now given by $\boldsymbol{x^*} = (\boldsymbol{x_1(-\theta_\ell)}^T,\ \mathbf{0},\ \boldsymbol{v}^T,\ \mathbf{0})^T$ with $\boldsymbol{x_1}(-\theta_{\min}) = (R_1 - \theta_\ell I)^{-1}(-\gamma^0 \boldsymbol{e_1})$ and $\boldsymbol{v} := \alpha \tilde{\boldsymbol{v}}$ where α has been chosen as the root of the scalar quadratic equation $\Delta^2 = \|\boldsymbol{x_1}(-\theta_{\min})\|^2 + \alpha^2 \|\tilde{\boldsymbol{v}}\|^2$ that leads to the smaller objective value.

8.4.6. Heuristic addressing ill-conditioning

The pL directions P_i are M-orthogonal if computed using exact arithmetic. It is well known that, in finite precision and if H is ill-conditioned, M-orthogonality may be lost due to propagation of roundoff errors . An indication that this happened may be by disproving

$$\tfrac{1}{2}\langle \boldsymbol{h}, T_i \boldsymbol{h}\rangle + \gamma^0 \langle \boldsymbol{h}, \boldsymbol{e_1}\rangle = q(P_i \boldsymbol{h}),$$

which holds if P_i indeed is M-orthogonal. On several badly scaled instances, for example `ARGLINB` of the `CUTEst` test set, we have seen that both quantities above may even differ in sign, in which case the solution of the trust-region subproblem would yield a direction of ascent. This issue becomes especially severe if H has small, but positive eigenvalues and admits an interior solution of the trust region subproblem. Then, the Ritz values computed as eigenvalues of T_i may very well be negative due to the introduction of roundoff errors, and enforce a convergence to a boundary point of the trust region subproblem. Finally, if the trust region radius Δ is large, the two "solutions" can differ significantly.

To address this observation, we have developed a heuristic that, by convexification, permits to obtain a descent direction of progress even if P_i has lost M-orthogonality. For this, let $\underline{\rho} := \min_j \frac{\langle p_j, H p_j\rangle}{\langle p_j, M p_j\rangle}$ and $\overline{\rho} := \max_j \frac{\langle p_j, H p_j\rangle}{\langle p_j, M p_j\rangle}$ be the minimal and maximal Rayleigh quotients used as estimates of extremal eigenvalues of H. Both are cheap to compute during the Krylov subspace iterations.

(1) If algorithm 8.1 has converged with a boundary solution such that

$$\lambda \geq 10^{-2} \max\{1, \rho_{\max}\} \quad \text{and} \quad |\rho_{\min}| \leq 10^{-8} \rho_{\max},$$

the case described above may be at hand. We compute $q_x := q(P_i \boldsymbol{h})$ in addition to $q_h := \frac{1}{2}\langle \boldsymbol{h}, T_i \boldsymbol{h}\rangle + \gamma^0 \langle \boldsymbol{h}, \boldsymbol{e_1}\rangle$. If either $q_x > 0$ or $|q_x - q_h| > 10^{-7} \max\{1, |q_x|\}$, we resolve with a convexified problem.

(2) The convexification heuristic we use is obtained by adding a positive diagonal matrix D to T_i, where D is chosen such that $T_i + D$ is positive definite. We then resolve then the tridiagonal problem with $T_i + D$ as the new convexified tridiagonal matrix. We obtain D by attempting to compute a Cholesky factor T_i. Monitoring the pivots in the Cholesky factorization, we choose d_j such that the pivots π_j are at least slightly positive. The formal procedure is given in algorithm 8.4. In our implementation we use the constants $\varepsilon = 10^{-12}$ and $\sigma = 10$.

Algorithm 8.4: Convexification heuristic for the tridiagonal matrix T_i.

input : T_i, $\varepsilon > 0$, $\sigma > 0$
output: D such that $T_i + D$ is positive definite

for $j = 0, \ldots, i$ **do**

$$\hat{\pi}_j := \begin{cases} \delta_0, & j = 0 \\ \delta_j - \gamma_j^2 / \pi_{j-1}, & j > 0 \end{cases}$$

$$d_j := \begin{cases} 0, & \hat{\pi}_j \geq \varepsilon \\ \sigma |\gamma_j^2 / \pi_{j-1} - \delta_j|, & \hat{\pi}_j < \varepsilon \end{cases}$$

$\pi_j := \hat{\pi}_j + d_j$

end

8.5. Summary

We have analyzed trust-region subproblems in Hilbert space and showed existence under appropriate assumptions. We have extended Gould's Generalized Lanczos Method for trust region problems in Hilbert spaces. We have developed a novel heuristic addressing ill-conditioned problems. The implementation `trlib` of this algorithm is presented in Chapter 11.

9. Gauß-Newton Preconditioner for Model Predictive Control

Real-time control of processes mandate the development of computationally *fast* algorithms that quickly yield good approximations to solutions of optimal control problems. We give a short overview on Nonlinear Model Predictive Control (NMPC) and develop a Gauß-Newton Preconditioner to accelerate the SLPECEQP algorithm of Chapter 7 in nonlinear model predictive control applications.

9.1. Online Optimal Control

In real-time optimal feedback control scheme, a family of optimal control problems parameterized by time is considered. At a given time, an optimal control problem describing the system under consideration is set up, solved instantaneously and the obtained optimal control evaluated at this time point is applied to the system.

As it is not possible to solve optimal control problems instantaneously, such an idealized scheme cannot be applied in practice. Instead, a set of discrete sampling times is selected at which the optimal control problem is solved and feedback control can be given afterwards. The duration between two subsequent sampling times has to be longer as the computations for solving the optimal control problem take. The computed feedback control is available with a delay bounded by the time it takes to solve the optimal control problem, which is hard to estimate a priori.

9.2. Real-Time Iterations

It is desirable to ensure that the duration between two subsequent samples is small as this ensures that the computed feedback control matches the process state that has been assumed upon computation. To this end, Diehl [Die01; Die+02] has developed the concept of real-time iterations, which we sketch in the following.

Suppose that at a time instance t an approximation to the optimal control problem corresponding to the process state $x(t)$ has to be computed and that previous to t a prediction $x^{\mathrm{pred}}(t)$ of $x(t)$ is available. If a direct method for discretization of

the optimal control problem and an iterative method for solution of the resulting nonlinear program is used, the iterative method can be prepared and initialized with $x^{\text{pred}}(t)$ instead of $x(t)$ previously to t in a *preparation phase*. Once $x(t)$ becomes available at t, the iteration data is updated by replacing the prediction $x^{\text{pred}}(t)$ with $x(t)$ and an approximate solution to the updated nonlinear program can be quickly computed in a *feedback phase*. With this solution a prediction for the state at the next time instance can be obtained and it can be used to initialize the subsequent nonlinear program in a *transition phase*.

By including an initial value embedding into the optimal control problem, Diehl showed, using techniques from parametric nonlinear programming and the local contraction Theorem of Bock [Boc87], that a full-step SQP real-time iteration scheme is contractible and provides a *tangential predictor* to the solution of the optimal control problem. In particular, only one SQP iteration is performed per problem in the feedback phase. In the transition phase, a *shift strategy* is employed for moving horizon problems. Crucial is the usage of the SQP method, as it is an active-set method with excellent warm starting capabilities. In comparison with interior point methods, it has been shown by Diehl [DFH09] and Zavala [ZLB07] that the computed tangential predictors by an interior point method are inferior to those of an active set method.

Kirches [Kir10; Kir+13a] has extended the real-time iteration scheme to the case of mixed-integer nonlinear model predictive control and proved contractibility of a scheme based on a full-step SQPVC method even in the presence of rounding. Bock et al. [BKS07], Wirsching et al. [WBD06; Kir+10b] and Frasch et al. [Fra+12] have developed a *multi-level iteration* and *mixed-level iteration* framework with adaptive reevaluation and relinearization. Wynn et al. [WVD14] establish conditions for convergence of real-time moving horizon schemes. Gros et al. [Gro+16] provide a comparison of nonlinear model predictive control and linear model predictive control.

9.3. SLPECEQP Real-Time Iteration Scheme

In this section, we present a real-time iteration scheme based on the SLPECEQP algorithm of Chapter 7. We assume that the SLPECEQP algorithm has been initialized with a solution of the offline problem. The real-time iteration scheme is outlined in Algorithm 9.1. The following three phases are iterated in the scheme:

Preparation Phase. In the preparation phase, the most expensive computations can be set up already without knowledge of the initial system state x^0, that is included in the optimal control problem via the initial value embedding constraint $0 = x(t_0) - x^0$. In this phase, the solution is prepared as far as possible without knowledge of x^0. In

Algorithm 9.1: SLPECEQP Real-Time Iteration Scheme. Lower case alphabetic enumeration corresponds to steps in Algorithm 7.2

I. Preparation. Function and derivative evaluation.

II. Feedback. Insert x^0 into initial value embedding $0 = s^0 - x^0$.
IIa. Compute penalty choice γ, linear step d_ℓ, working set guess $\mathcal{W}$.
IIb. Compute least-squares multiplier estimation $\lambda_{\mathrm{LS}}, \mu_{\mathrm{LS}}, \nu_{\mathrm{LS}}, \sigma_{\mathrm{LS}}$.
IIc. Compute Cauchy step $d_{\mathrm{C}} = \alpha d_\ell$.
IId. Compute step d as solution of an EQP on $\mathcal{W}$.
II. Provide feedback control $u(t_0)$.

III. Transition. Shift time horizon and variables.

particular, all function and derivative evaluations necessary are executed.

Feedback Phase. Once the system state x^0 is known, it is included in the evaluation of the initial value embedding. For the Direct Multiple Shooting Discretization as presented in Chapter 3.1.1, the initial value embedding constraint is given by $0 = s^0 - x^0$ with the shooting variable s^0 on the first multiple shooting node. Afterwards, one iteration of the SLPECEQP method is performed. This can be done cheaply as the solution of the LPEC with an active-set method from a neighboring point usually requires very few active-set iterations due to warm-starting capabilities of active-set LPEC solution algorithms. To ensure fast solution of the EQP we rely on the Gauß-Newton Preconditioner discussed in the next section.

The solution estimate obtained bye one iteration of the SLPECEQP method is used to compute $u(t_0)$ that can be immediately applied to the process. If the working set estimate $\mathcal{W}$ identifies the active set at the solution, $u(t_0)$ provides a tangential predictor to the optimal feedback control.

Transition Phase. In the transition phase the problem for the next sampling time is prepared. This involves shifting the time horizon and the shooting variables motivated by the principle of optimality of subarcs. In a moving horizon setting, new estimates for the variables on the last shooting node are required as these are lost upon shifting. Several strategies exist to this end. One possibility is to reuse the values of the variables of the previous problem, which may render the matching condition on the last shooting interval infeasible. Another possibility is to integrate the initial value problem on the last shooting interval from the shifted state variable on the penultimate shooting node.

9.4. Gauß-Newton Preconditioner

The dominating computational effort in solving multiple-shooting discretizations of optimal control problems with the SLPECEQP algorithm is the iterative solution of the EQP. The computational effort for solving the EQP is in essence determined by two factors: The number of preconditioned Krylov subspace iterations required and the time it requires to evaluate one matrix-vector product with Hessian of the Lagrangian. For online optimal control applications it is crucial to solve the EQP as fast as possible. While the time to evaluate one matrix-vector product is fixed, the number of Krylov subspace iterations can be influenced by the choice of a suitable preconditioner that either is a good approximation to the inverse or clusters the eigenvalues of the Hessian of the Lagrangian.

In online optimal control, objective functions often are of tracking type and thus have a Least-Squares Lagrange type objective function. For this case of problems with Least-Squares Lagrange type objective function, there is a natural preconditioner given in the form of the Gauß-Newton approximation to the Hessian matrix. Direct multiple shooting discretization of such a problem yields an objective function that can be represented as

$$f(z) = \tfrac{1}{2}\|r(z)\|_2^2, \tag{9.1}$$

with a twice continuously differentiable function $r(z)$.

9.1 Definition (Gauß-Newton preconditioner).
The Gauß-Newton preconditioner of (9.1) in z is defined by $M := J^T J$, $J = \frac{\mathrm{d}r(z)}{\mathrm{d}z}$. The corresponding projection in the EQP solution is defined by

$$P_{J^TJ} = \begin{pmatrix} I & 0 \end{pmatrix} \begin{pmatrix} J^TJ & A^T \\ A & 0 \end{pmatrix}^{-1} \begin{pmatrix} I \\ 0 \end{pmatrix}. \qquad \triangle$$

Gauß-Newton methods based on the Gauß-Newton approximation $J^T J$ of the Hessian $J^T J + \sum_i r_i(z)\nabla^2 r_i(z)$ for nonlinear least-squares problems are analyzed in detail by Bock [Boc87]. The following lemma gives a rank conditions for the existence of P_{J^TJ}:

9.2 Lemma ([Boc87, 3.1.25,3.1.28]).
Let $J \in \mathbb{R}^{m_1 \times n}$ and $A \in \mathbb{R}^{m_2 \times n}$ with $m_2 \leq n$ and $n \leq m_1 + m_2$. Suppose that the rank conditions $\operatorname{rank} A = m_2$ and $\operatorname{rank} \begin{pmatrix} J \\ A \end{pmatrix} = n$ hold.

Then $\begin{pmatrix} J^TJ & A^T \\ A & 0 \end{pmatrix}$ is non-singular and P_{J^TJ} is well-defined.

Furthermore, $J^T J$ is positive definite on the null-space of A. $\triangle$

For our purposes we use the Gauss-Newton Hessian approximation as preconditioner. An preconditioner is efficient if it either ensures clustering of eigenvalues or reduces the condition number. An optimal preconditioner that would yield convergence within one iteration is given by the Hessian of the problem, provided that is positive definite. The Gauss-Newton Hessian approximation is cheap to compute and provides a good approximation to the Hessian that is always positive semi-definite. A quantitative description of the quality of the approximation can be found in the context of the estimation of κ and ω-conditions of Bock's local contraction Theorem [Boc87], see also [Deu06] and [Pot11, Ch. 5.2].

A computational example that demonstrates the efficiency of the Gauß-Newton preconditioner is given in Chapter 16.2.

9.5. Summary

In this chapter, we gave a short introduction into online optimal control and introduced a Gauß-Newton preconditioner that is effectively applied in 16.2.

Part III.

Implementations and Numerical Results

10. Benchmarking Optimization Software

In this chapter, we introduce Dolan-Moré Performance Profiles used to benchmark optimization software and the benchmark collections `CUTEr` and `CUTEst`.

10.1. Performance Profiles

Dolan and Moré [DM02] have introduced *performance profiles* to assess the performance of different solvers on a collection of benchmark problems.

10.1 Definition (Performance Profile).
Assume a finite set $\mathcal{P}$ of benchmark problems and a finite set $\mathcal{S}$ of solvers is given and $t : \mathcal{P} \times \mathcal{S} \to \mathbb{R}_{>0}, (p, s) \mapsto t_{p,s}$ a performance mapping.

The performance ratio $r : \mathcal{P} \times \mathcal{S} \to \mathbb{R}_{>0}$ is then defined by normalization to best performance,

$$\hat{r}_{p,s} := \frac{t_{p,s}}{\min_{\sigma \in \mathcal{S}} t_{p,\sigma}}.$$

The *performance profile* $\hat{\rho}_s : \mathbb{R}_{>0} \to [0, 1]$ of solver $s \in \mathcal{S}$ is defined as cumulative distribution of $\hat{r}_{\cdot,s}$,

$$\hat{\rho}_s(\tau) := \frac{|\{p \in \mathcal{P} \mid \hat{r}_{p,s} \leq \tau\}|}{|\mathcal{P}|}.$$

△

By definition, $\hat{\rho}_s(1)$ gives the ratio of problems solver s solves fastest, while $\lim_{\tau \to \infty} \hat{\rho}_s(\tau)$ gives the fraction of problems that solver s can solve at all. For $\tau > 1$, $\hat{\rho}_s(\tau)$ that can be solved by s within a factor τ of the fastest solver.

Mahajan et al. [MLK11] noted that this lacks the information if a solver s is faster than any other solver by a factor of at most τ and to this end introduced *extended performance profiles*:

10.2 Definition (Extended Performance Profile).
Assume a finite set $\mathcal{P}$ of benchmark problems and a finite set $\mathcal{S}$ of solvers is given and $t : \mathcal{P} \times \mathcal{S} \to \mathbb{R}_{>0}, (p, s) \mapsto t_{p,s}$ a performance mapping.

The extended performance ratio $r : \mathcal{P} \times \mathcal{S} \to \mathbb{R}_{>0}$ is then defined by

$$r_{p,s} := \frac{t_{p,s}}{\min_{\sigma \in \mathcal{S}, \sigma \neq i} t_{p,\sigma}}.$$

The *extended performance profile* $\rho_s : \mathbb{R}_{>0} \to [0,1]$ of solver $s \in \mathcal{S}$ is defined as cumulative distribution of $r_{\cdot,s}$,

$$\rho_s(\tau) := \frac{|\{p \in \mathcal{P} \mid r_{p,s} \leq \tau\}|}{|\mathcal{P}|}.$$

△

Extended performance profiles extend the notion of performance profiles. It holds $\hat{\rho}_s(\tau) = \rho_s(\tau)$ for all $\tau \geq 1$. Now $\lim_{\tau \searrow 0} \rho_s(\tau)$ gives the fraction of problems that can be solved only by s.

10.2. Benchmark Sets `CUTEr` and `CUTEst`

`CUTEr` [GOT02] and its successor `CUTEst` [GOT15] are established collections of benchmark problems for nonlinear programming. The problems stem both from real-world applications and from academic examples and represent a variety of different classes of problems. All problems have objective and constraints functions that are at least twice continuously differentiable and are without integrality constraints. Problems are categorized in unconstrained and constrained and in linear, quadratic or general nonlinear and span sizes from one up to 250,000 variables and constraints. They are distributed together with a Fortran library that provide routines for evaluation of objective and constraint functions and their derivatives.

Benson [Ben01] has translated the `CUTEr` collection as of 2001 into `AMPL` [FGK90; FGK02] which provides a convenient tool for comparison involving a wide range of solvers, as most solver have interfaces to `AMPL`. The AMPL translation consists of a 924 instance subset of current `CUTEr` that may slightly differ from its `CUTEr`/`CUTEst` counterpart in supplied start points, parameter values, and choices for variable sized problems.

10.3. Summary

We have introduced *performance profiles* as a tool to compare different solvers on benchmark sets and the collections of benchmark problems `CUTEr` and `CUTEst`.

11. Implementation and Benchmark of Generalized Lanczos Method

We introduce `trlib`, which is a vector-free implementation of the GLTR (Generalized Lanczos Trust Region) method for solving the trust region subproblem described in chapter 8. We assess the performance of this implementation on trust region problems obtained from the set of unconstrained nonlinear minimization problems of the `CUTEst` benchmark library, as well as on a number of examples formulated in Hilbert space that arise from PDE-constrained optimal control.

The results of this chapter are published in [LKP16].

11.1. Implementation `trlib`

In this section, we present details of our implementation `trlib` of the GLTR method.

11.1.1. Existing Implementation

The GLTR reference implementation is the software package `GLTR` in the optimization library `GALAHAD` [GOT04]. This Fortran 90 implementation uses conjugate gradient iterations exclusively to build up the Krylov subspace, and provides a reverse communication interface that requires to exchange vector data to be stored as contiguous arrays in memory.

11.1.2. `trlib` Implementation

Our implementation is called `trlib`, short for *trust region library*. It is written in plain ANSI C99 code, and has been made available as open source [LKP16]. We provide a reverse communication interface in which only scalar data and requests for vector operations are exchanged, allowing for great flexibility in applications. `trlib` has been added to the `SciPy` library package in version 1.0 as core optimization solver.

Beside the stable and efficient conjugate gradient iteration we also implemented the Lanczos iteration and a crossover mechanism to expand the Krylov subspace, as we frequently found applications in the context of constrained optimization with

an SLPECEQP method where conjugate gradient iterations broke down whenever directions of tiny curvature had been encountered.

11.1.3. Vector Free Reverse Communication Interface

The implementation is built around a reverse communication calling paradigm. To solve a trust region subproblem, the according library function has to be repeatedly called by the user and, after each call, the user has to perform a specific action indicated by the value of an output variable. Only scalar data representing dot products and coefficients in axpy operations as well as integer and floating point workspace to hold data for the tridiagonal subproblems is passed between the user and the library. In particular, all vector data has to be managed by the user, who must be able to compute dot products $\langle x, y\rangle$, perform axpy $y := \alpha x + y$ on them and implement operator vector products $x \mapsto Hx, x \mapsto M^{-1}x$ with the Hessian and the preconditioner.

Thus, no assumption about representation and storage of vectorial data is made, as well as no assumption on the discretization of $\mathcal{H}$ if $\mathcal{H}$ is not finite-dimensional. This is beneficial in problems arising from optimization problems stated in function space that may not be stored naturally as contiguous vectors in memory or where adaptivity regarding the discretization may be used along the solution of the trust region subproblem. It also gives a trivial mechanism for exploiting parallelism in vector operations as vector data may be stored and operations may be performed on GPU without any changes in the trust region library.

In particular, this interface allows for easy interfacing with the PDE-constrained optimization software DOLFIN-adjoint [Far+13; FF13] within the finite element framework FEniCS [Aln+15; LW10; Aln+14] without having to rely on assumptions on how the finite element discretization is stored.

11.1.4. Conjugate Gradient Breakdown

Per default, conjugate gradient iterations are used to build the Krylov subspace. The algorithm switches to Lanczos iterations if the magnitude of the curvature $|\langle \hat{p}, H\hat{p}\rangle| \leq$ tol_curvature, with a user defined tolerance tol_curvature ≥ 0.

11.1.5. Easy Case

In the easy case after the Krylov space has been assembled in a particular iteration, it remains to solve $(\mathrm{TR}(T_i, \gamma^0 \boldsymbol{e}_1, I, \Delta, \mathbb{R}^{i+1}))$, which we do as outlined in Chapter 8.4.3. As mentioned there, an improved convergence order can be obtained by higher order Taylor expansions of $\sigma_p(\lambda)$ and values $p \neq -1$, see [GT10]. However, in our case the computational cost for solving the tridiagonal subproblem — often warm started in a

suitable way — is negligible in comparison to the cost of computing matrix vector products $x \mapsto Hx$. We thus decided to stick to the simpler Newton root-finding on $\sigma_{-1}(\lambda)$.

To obtain a suitable initial value λ^0 for the Newton iteration, we first try λ^* obtained in the previous Krylov iteration if available, and $\lambda^0 = 0$ otherwise. If these fail, we use $\lambda^0 = -\theta_{\min}$ computed as outlined in Chapter 8.4.4 by zero-finding on $d(\theta)$ or $\hat{d}(\theta)$. This requires suitable models for $\hat{d}(\theta)$. Gould et al. [Gou+99] propose to use a quadratic model $\theta^2 + a\theta + b$ for $\hat{d}(\theta)$ that captures the asymptotics $t \longrightarrow -\infty$ obtained by fitting function value and derivative in a point in the root finding process. We have also had good success with the linear Newton model $a\theta + b$, and with using a second order quadratic model $a\theta^2 + b\theta + c$, that makes use of an additional second derivative. Derivatives of $d(\theta)$ or $\hat{d}(\theta)$ are easily obtained by differentiating the recurrence $d_{k+1} = \delta_{k+1} - \theta - \frac{\gamma_{k+1}^2}{d_k}$ for the Cholesky decomposition, $\dot{d}_{k+1} = -1 + \frac{\gamma_{k+1}^2}{d_k^2}\dot{d}_k$ and $\ddot{d}_{k+1} = \frac{\gamma_{k+1}^2}{d_k^2}(\ddot{d}_k - 2\frac{\dot{d}_k^2}{d_k})$ and are cheaply to compute together with the Cholesky decomposition. In our implementation, a heuristic is used to select the option that is inside the interval of safety and promises good progress. The heuristic is given by using $\theta^2 + a\theta + b$ in case that the bracket width $\theta_{\mathrm{u}}^k - \theta_\ell^k$ satisfies $\theta_{\mathrm{u}}^k - \theta_\ell^k \geq 0.1 \max\{1, |\theta^k|\}$ and $a\theta^2 + b\theta + c$ otherwise. The motivation behind this is that in the former case it is not guaranteed that θ^k has been determined to high accuracy as zero of $d(\theta)$ and thus the model that captures the global behavior might be better suited. In the latter case, θ^k has been confirmed to be a zero of $d(\theta)$ to a certain accuracy and it is safe to use the model representing local behavior.

11.1.6. Hard Case

We now discuss the so-called hard case of the trust region problem, which we have found to be of critical importance for the performance of trust region subproblem solvers in general nonlinear non-convex programming. We discuss algorithmic and numerical choices made in `trlib` that we have found to help improve performance and stability.

Exact Hard Case

The function for the solution of the tridiagonal subproblem implements the algorithm as given by Theorem 8.11 if provided with a decomposition in irreducible blocks.

However, from local information it is not possible to distinguish between convergence to a global solution of the original problem and the case in which an invariant Krylov subspace is exhausted that may not contain the global minimizer as, in both

cases, the gradient vanishes.

The handling of the hard case is thus left to the user in the reverse communication calling scheme if arrived at a point where the gradient norm is sufficiently small. The user then has to decide if the solution in the Krylov subspaces investigated so far is accepted, or further Krylov subspaces should be investigated. In that case it is left to the user to determine a new nonzero initial vector for the Lanczos iteration that is M-orthogonal to the previous Krylov subspaces. One possibility to obtain such a vector is using a random vector and M-orthogonalizing it with respect to the previous Lanczos directions using the modified Gram-Schmidt algorithm.

Near Hard Case

The near hard case arises if $\langle \gamma^0 \boldsymbol{e_1}, \frac{\tilde{\boldsymbol{v}}}{\|\tilde{\boldsymbol{v}}\|} \rangle$ is tiny, where $\tilde{\boldsymbol{v}}$ spans the eigenspace $\mathrm{Eig}(\theta_{\min})$.

Numerically this is detected if there is no $\lambda \geq \max\{0, -\theta_{\min}\}$ such that $\|\boldsymbol{x}(\lambda)\| \geq \Delta$ holds in floating point arithmetic. In that case we use the heuristic $\lambda^* = -\theta_{\min}$ and $x^* = x(-\theta_{\min}) + \alpha \boldsymbol{v}$ with $\boldsymbol{v} \in \mathrm{Eig}(\theta_{\min})$ where α is determined such that $\|\boldsymbol{x^*}\| = \Delta$.

Another possibility would be to modify the tridiagonal matrix T by dropping off-diagonal elements below a specified threshold and work on the obtained decomposition into irreducible blocks. However, we have not investigated this possibility as the heuristic delivers satisfactory results in practice.

11.1.7. Reentry with New Trust Region Radius

In nonlinear programming applications it is common that, after a rejected step, another closely related trust region subproblem has to be solved with the only changed data being the trust region radius. As this has no influence on the Krylov subspace but only on the solution of the tridiagonal subproblem, efficient hot-starting has been implemented. Here, the tridiagonal subproblem is solved again with exchanged radius and termination tested. If this point does not satisfy the termination criterion, conjugate gradient or Lanczos iterations are resumed until convergence. However, we rarely observed the need to resume the Krylov iterations in practice.

An explanation is offered based on the use of the convergence criterion $\|\nabla L\|_{M^{-1}} \leq$ `tol` as follows: In the Krylov subspace $\mathcal{K}_i$,

$$\|\nabla L\|_{M^{-1}} = \gamma^{i+1} |\langle \boldsymbol{x}(\lambda), \boldsymbol{e_{i+1}} \rangle| \leq \gamma^{i+1} \|\boldsymbol{x}(\lambda)\|_2 = \gamma^{i+1} \Delta,$$

convergence occurs thus if either γ^{i+1} or the last component of $\boldsymbol{x}(\lambda) \leq \Delta$ are small. Reducing the trust region radius also reduces the upper bound for $\|\nabla L\|_{M^{-1}}$, so convergence is likely to occur, especially if γ^{i+1} turns out to be small.

If the trust region radius is small enough, or equivalently the Lagrange multiplier large enough, it can be proven that a decrease in the trust region radius leads to a decrease in $\|\nabla L\|_{M^{-1}}$:

11.1 Lemma.
There exists $\hat{\lambda} > \max_i |\lambda_i(T)|$ *such that* $\lambda \mapsto \gamma^{i+1}|\langle \boldsymbol{x}(\lambda), \boldsymbol{e_{i+1}}\rangle$ *is a decreasing function for* $\lambda \geq \hat{\lambda}$.

Proof. Using the expansion $(T_i + \lambda I)^{-1} = \sum_{k\geq 0}(-1)^k \frac{1}{\lambda^{k+1}} T^k$, which holds for $\lambda > \max_i |\lambda_i(T)|$, we find:

$$\begin{aligned}\|\nabla L\|_{M^{-1}} &= \gamma^{i+1}|\langle \boldsymbol{x}(\lambda), \boldsymbol{e_{i+1}}\rangle| = \gamma^{i+1}\gamma^0|\langle (T_i + \lambda I)^{-1}\boldsymbol{e_1}, \boldsymbol{e_{i+1}}\rangle| \\ &= \gamma^{i+1}\gamma^0 \left|\sum_{k\geq 0}(-1)^k \tfrac{1}{\lambda^{k+1}} \boldsymbol{e}_{\boldsymbol{i+1}}^T T^k \boldsymbol{e_1}\right| = \frac{\prod_{j=0}^{i+1}\gamma^j}{\lambda^{i+1}} + O((\tfrac{1}{\lambda})^{i+2}),\end{aligned}$$

where we have made use of the facts that $\boldsymbol{e}_{\boldsymbol{i+1}}^T T^k \boldsymbol{e_0}$ vanishes for $k < i$, and that $\boldsymbol{e}_{\boldsymbol{i+1}}^T T^k \boldsymbol{e_0} = \prod_{j=1}^{i}\gamma^j$, which can be easily proved using the relation $T\boldsymbol{e_j} = \gamma^{j-1}\boldsymbol{e_{j-1}} + \gamma^{j+1}\boldsymbol{e_{j+1}} + \delta_j \boldsymbol{e_j}$. The claim now holds if λ is large enough such that higher order terms in this expansion can be neglected. □

11.1.8. Termination criterion

Convergence is reported as soon as the Lagrangian gradient satisfies

$$\|\nabla L\|_{M^{-1}} \leq \begin{cases}\max\{\texttt{tol\_abs\_i}, \texttt{tol\_rel\_i}\,\|g\|_{M^{-1}}\}, & \text{if } \lambda = 0 \\ \max\{\texttt{tol\_abs\_b}, \texttt{tol\_rel\_b}\,\|g\|_{M^{-1}}\}, & \text{if } \lambda > 0\end{cases}.$$

The rationale for using possibly different tolerances in the interior and boundary case is motivated from applications in nonlinear optimization where trust region subproblems are used as globalization mechanism. There a local minimizer of the nonlinear problem will be an interior solution to the trust region subproblem and it is thus not necessary to solve the trust region subproblem in the boundary case to highest accuracy.

11.1.9. TRACE

In the recently proposed TRACE algorithm [CS16], trust region problems are also used. In addition to solving trust region problems, the following operations have to be performed:

- $\min_x \frac{1}{2}\langle x, (H + \lambda M)x\rangle + \langle g, x\rangle$,

- Given constants $\sigma_\mathrm{l}, \sigma_\mathrm{u}$ compute λ such that the solution point of $\min_x \frac{1}{2}\langle x, (H + \lambda M)x\rangle + \langle g, x\rangle$ satisfies $\sigma_\mathrm{l} \leq \frac{\lambda}{\|x\|_M} \leq \sigma_\mathrm{u}$.

These operations have to be performed after a trust region problem has been solved and can be efficiently implemented using the Krylov subspaces already built up.

We have implemented these as suggested in [CS16], where the first operation requires one backsolve with tridiagonal data and the second one is implemented as root finding on $\lambda \mapsto \frac{\lambda}{\|x(\lambda)\|} - \sigma$ with a certain $\sigma \in [\sigma_\mathrm{l}, \sigma_\mathrm{u}]$ that is terminated as soon as $\frac{\lambda}{\|x(\lambda)\|} \in [\sigma_\mathrm{l}, \sigma_\mathrm{u}]$.

11.1.10. C11 Interface

The algorithm has been implemented in C11. The user is responsible for holding vector-data and invokes the algorithm by repeated calls to the function `trlib_krylov_min` with integer and floating point workspace and dot products $\langle v, g\rangle$, $\langle p, Hp\rangle$ as arguments and in return receives status information and instructions to be performed on the vectorial data. A detailed reference is provided in the `Doxygen` documentation to the code.

11.1.11. Python Interface

A low-level python interface to the C library has been created using `Cython` that closely resembles the C API and allows for easy integration into more user-friendly, high-level interfaces.

As a particular example, a trust region solver for PDE-constrained optimization problems has been developed to be used from `DOLFIN-adjoint` [Far+13; FF13] within `FEniCS` [Aln+15; LW10; Aln+14]. Here vectorial data is only considered as `FEniCS`-objects and no numerical data except for dot products is used of these objects.

11.2. Performance on `CUTEst` Benchmark Collection

In this section, we present an assessment of the computational performance of our implementation `trlib` of the GLTR method, and compare it to the reference implementation `GLTR` as well as several competing methods for solving the trust region problem and their respective implementations.

11.2.1. Generation of Trust-Region Subproblems

For want of a reference benchmark set of non-convex trust region subproblems, we resorted to the subset of unconstrained nonlinear programming problems of the

CUTEst benchmark library, and use a standard trust region algorithm, e.g. Gould et al. [Gou+99], for solving $\min_{x \in \mathbb{R}^n} f(x)$, as a generator of trust-region subproblems. The algorithm starts from a given initial point $x^0 \in \mathbb{R}^n$ and trust region radius $\Delta^0 > 0$, and iterates for $k \geq 0$:

Algorithm 11.1: Standard trust region algorithm for unconstrained nonlinear programming, used to generate trust region subproblems from CUTEst.

input : f, x^0, Δ^0, ρ_{acc}, ρ_{inc}, γ^+, γ^-, tol_abs
output: k, x^k

for $k \geq 0$ **do**
- Evaluate $g^k := \nabla f(x^k)$
- Test for termination: Stop if $\|g^k\| \leq$ tol_abs
- Evaluate $H^k := \nabla^2_{xx} f(x^k)$
- Compute (approximate) minimizer d^k to $\text{TR}(H^k, g^k, I, \Delta^k)$
- Assess the performance $\rho^k := (f(x^k + d^k) - f(x^k))/q(d^k)$ of the step
- Update step: $x^{k+1} := \begin{cases} x^k + d^k, & \rho^k \geq \rho_{\text{acc}} \\ x^k, & \rho^k < \rho_{\text{acc}} \end{cases}$,
- Update trust region radius: $\Delta^{k+1} := \begin{cases} \gamma^+ \Delta^k, & \rho^k \geq \rho_{\text{inc}} \\ \Delta^k, & \rho_{\text{acc}} \leq \rho^k < \rho_{\text{inc}} \\ \gamma^- \Delta^k, & \rho^k < \rho_{\text{acc}} \end{cases}$

end

In a first study, we compared our implementation trlib of the GLTR method to the reference implementation GLTR as well as several competing methods for solving the trust region problem, and their respective implementations, as follows:

- GLTR [Gou+99] in the GALAHAD library implements the GLTR method.
- LSTRS [RSS08] uses an eigenvalue based approach. The implementation uses MATLAB and makes use of the direct ARPACK [LSY98] reverse communication interface, which is deprecated in recent versions of MATLAB and lead to crashes within MATLAB 2013b used by us. We thus resorted to the standard eigs eigenvalue solver provided by MATLAB which might severely impact the behavior of the algorithm.
- SSM [Hag01] implements a sequential subspace method that may use an SQP accelerated step.

solver	τ interior convergence	τ boundary convergence
`GLTR`	$\min\{0.5, \|\boldsymbol{g}^{\boldsymbol{k}}\|_{M^{-1}}\}\|\boldsymbol{g}^{\boldsymbol{k}}\|_{M^{-1}}$	identical to interior
`LSTRS`	defined in dependence of convergence of implicit restarted Arnoldi method	
`SSM`	$\min\{0.5, \|\boldsymbol{g}^{\boldsymbol{k}}\|\}_{M^{-1}}\|\boldsymbol{g}^{\boldsymbol{k}}\|_{M^{-1}}$	identical to interior
`ST`	$\min\{0.5, \|\boldsymbol{g}^{\boldsymbol{k}}\|\}_{M^{-1}}\|\boldsymbol{g}^{\boldsymbol{k}}\|_{M^{-1}}$	method heuristic in that case
`trlib`	$\min\{0.5, \|\boldsymbol{g}^{\boldsymbol{k}}\|\}_{M^{-1}}\|\boldsymbol{g}^{\boldsymbol{k}}\|_{M^{-1}}$	$\max\{10^{-6}, \min\{0.5, \|\boldsymbol{g}^{\boldsymbol{k}}\|_{M^{-1}}^{1/2}\}\}\|\boldsymbol{g}^{\boldsymbol{k}}\|_{M^{-1}}$

Table 11.1.: Convergence criteria for subproblem solvers $\|\nabla L\|_{M^{-1}} \leq \tau$

- `ST` is an implementation of the truncated conjugate gradient method proposed independently by Steihaug [Ste83] and Toint [Toi81].
- `trlib` is our implementation of the GLTR method.

All codes, with the exception of `LSTRS`, have been implemented in a compiled language, Fortran 90 in case of `GLTR` and C in for all other codes, by their respective authors. `LSTRS` has been implemented in interpreted MATLAB code. The benchmark code used to run this comparison has also been made open source and is available as `trbench` [Len16].

In our test case the parameters $\Delta^0 = \frac{1}{\sqrt{n}}$, $\texttt{tol\_abs} = 10^{-7}$, $\rho_{\text{acc}} = 10^{-2}$, $\rho_{\text{inc}} = 0.95$, $\gamma^+ = 2$ and $\gamma^- = \frac{1}{2}$ have been used. We used the subproblem convergence criteria as specified in Table 11.1 for the different solvers, trying to have as comparable convergence criteria as possible within the available applications. Our rationale for the interior convergence criterion to request $\|\nabla L\|_{M^{-1}} = O(\|\boldsymbol{g}^{\boldsymbol{k}}\|_{M^{-1}}^2)$ is that it defines an inexact Newton method with q-quadratic convergence rate, [NW06, Thm 7.2]. As `LSTRS` is a method based on solving a generalized eigenvalue problem, its convergence criterion depends on the convergence criterion of the generalized eigensolver and is incomparable with the other termination criteria. With the exception of `trlib`, no other solver allows to specify different convergence criteria for interior and boundary convergence.

Figure 11.1 shows extended performance profiles. It can be seen that `GLTR` and `trlib` are the most robust solvers on the subset of unconstrained problems from `CUTEst` in the sense that they eventually solve the largest fraction of problems among all solvers and that they are also among the fastest solvers. That `GLTR` and `trlib` show similar performance is to be expected as they implement the identical GLTR algorithm, where `trlib` is slightly more robust and faster. We attribute this to the implementation of efficient hot-start capabilities and also the Lanczos process to build up the Krylov subspaces once directions of zero curvature are encountered.

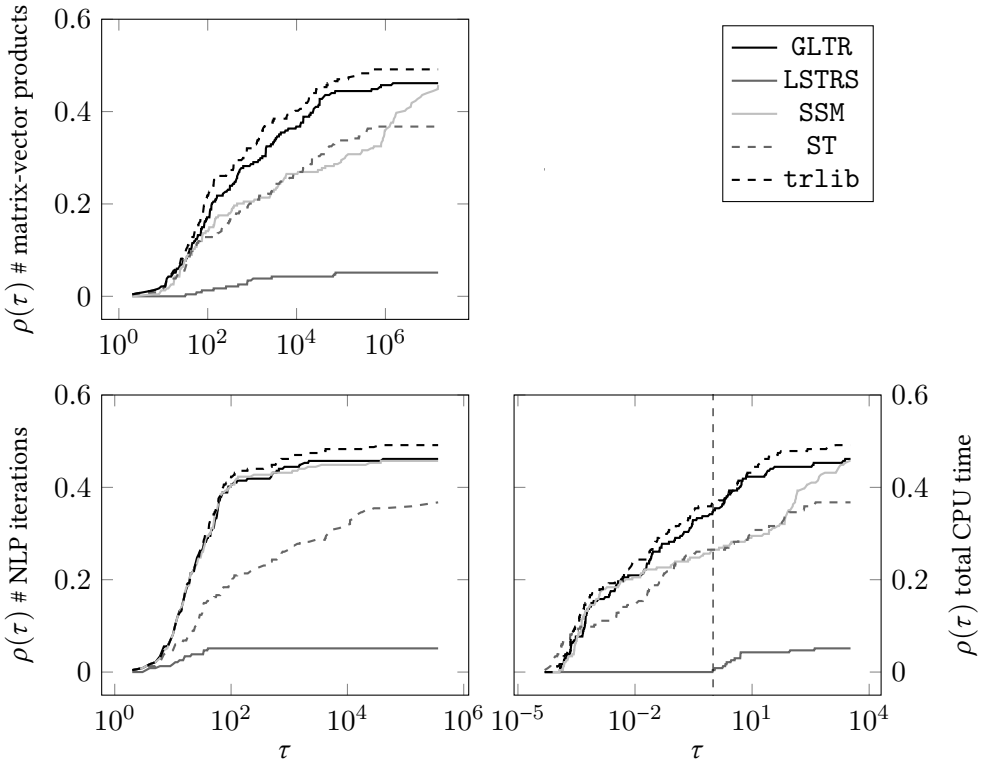


Figure 11.1.: Performance Profiles for matrix-vector products, NLP iterations and total CPU time for different trust region subproblem solvers when used in a standard trust region algorithm for unconstrained minimization evaluated on the set of all unconstrained minimization problems from the CUTEst library.

11.3. PDE constrained Trust Region Problem in Hilbert Space

We solved a modified variant of `SCDIST1` [Cas86; MRT06] of the `OPTPDE` benchmark library [Her+; Her+14] for PDE constrained optimal control problems. The state constraint has been dropped and a trust region constraint added in order to obtain the following function space trust region problem:

$$\begin{aligned} \min_{y \in W^{1,2}(\Omega), u \in L^2(\Omega)} \quad & \tfrac{1}{2}\|y - y_\mathrm{d}\|^2_{L^2(\Omega)} + \tfrac{\beta}{2}\|u - u_\mathrm{d}\|^2_{L^2(\Omega)} \\ \text{s.t.} \quad & -\triangle y + y = u, \qquad x \in \Omega, \\ & \partial_n y = 0, \qquad x \in \partial\Omega, \\ & \|y\|^2_{L^2(\Omega)} + \|u\|^2_{L^2(\Omega)} \leq \Delta^2. \end{aligned}$$

Here $\Omega \subseteq \mathbb{R}^n$ is a domain and $\triangle$ is the Laplace operator $\triangle = \sum_{i=1}^n \partial^2_{ii}$.

Tracking data $y_\mathrm{d}, u_\mathrm{d}$ has been used as specified in `OPTPDE` where typical regularization parameters have been considered in the range $10^{-8} \leq \beta \leq 10^{-3}$. Different geometries $\Omega \in \{(0,1)^2, (0,1)^3, \{x \in \mathbb{R}^2 \mid \|x\| \leq 1\}, \{x \in \mathbb{R}^3 \mid \|x\| \leq 1\}\}$ in two and three dimensions have been studied.

The finite element software `FEnICS` has been used to obtain a finite element discretization of the problem:

$$\begin{aligned} \min_{\boldsymbol{y} \in \mathbb{R}^{n_y}, \boldsymbol{u} \in \mathbb{R}^{n_u}} \quad & \tfrac{1}{2}\|\boldsymbol{y} - \boldsymbol{y}_\mathrm{d}\|^2_M + \tfrac{\beta}{2}\|\boldsymbol{u} - \boldsymbol{u}_\mathrm{d}\|^2_M \\ \text{s.t.} \quad & A\boldsymbol{y} - M\boldsymbol{u} = 0, \\ & \|\boldsymbol{y}\|^2_M + \|\boldsymbol{u}\|^2_M \leq \Delta^2, \end{aligned}$$

where M denotes the mass matrix and $A = K + M$ with K being the stiffness matrix.

We used the approach suggested by Gould et al. [GHN01] to solve this equality constrained trust region problem:

(1) A null-space projection in the preconditioning step of the Krylov subspace iteration is used to satisfy the discretized PDE constraint. The required preconditioner is given by

$$\begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{u} \end{pmatrix} \mapsto \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \end{pmatrix} \begin{pmatrix} M & 0 & A \\ 0 & M & -M \\ A & -M & 0 \end{pmatrix}^{-1} \begin{pmatrix} I & 0 \\ 0 & I \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{u} \end{pmatrix}.$$

(2) We used MINRES [PS75] for solving with the linear system arising in this preconditioner to high accuracy. MINRES iterations themselves are preconditioned

using the approximate Schur-complement preconditioner

$$\begin{pmatrix} \tilde{M} & & \\ & \tilde{M} & \\ & & \tilde{A}M^{-1}\tilde{A} \end{pmatrix}^{-1},$$

as proposed by [RDW10]. This preconditioner is an approximation to the optimal preconditioner

$$\begin{pmatrix} M & & \\ & M & \\ & & AM^{-1}A + M \end{pmatrix}^{-1}$$

that would lead to mesh-independent MINRES convergence in three iterations, provided that exact arithmetic [Kuz95; MGW00] is used.

(3) In the MINRES preconditioner of step (2), products with $\tilde{M}^{-1}$ and $\tilde{A}^{-1}$ are computed using truncated conjugate gradients (CG) to high accuracy, again preconditioned using an algebraic multigrid as preconditioner.

In Figure 11.2, it can be seen that using the GLTR method for these function space problems yields a solver with mesh-independent convergence behavior. The number of outer iterations is virtually constant on a wide range of different meshes and varies at most by one iteration. The number of inner (MINRES) iterations varies only slightly, as is to be expected due to the use of an approximately optimal preconditioner in step (2).

11.4. Summary

We have presented `trlib` which implements Gould's Generalized Lanczos Method for trust region problems and is now part of the core optimization library of the scientific computing python package `SciPy`. Distinct features of the implementation are by the choice of a reverse communication interface that does not need access to vector data but only to dot products between vectors and by the implementation of preconditioned Lanczos iterations to build up the Krylov subspace. The package `trbench`, which relies on `CUTEst`, has been introduced as a test bench for trust region problem solvers. Our implementation `trlib` shows similar and favorable performance in comparison to the `GLTR` implementation of the Generalized Lanczos Method and also in comparison to other iterative methods for solving the trust region problem.

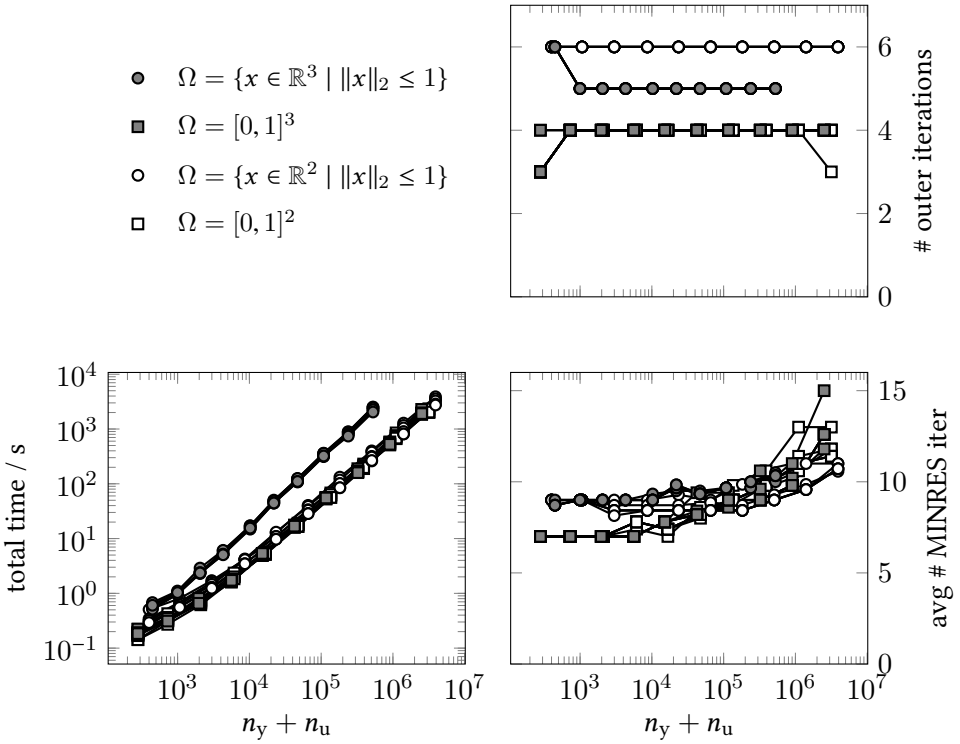


Figure 11.2.: Results for distributed control trust region problem for different mesh sizes. Results are shown for four different geometries. Regularization parameters $\beta \in \{10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}, 10^{-8}\}$ have been considered, however computational results for a fixed geometry hardly change with β leading to near-identical plots.

Moreover, we have solved an example from PDE constrained optimization to show that the implementation can be used for problems stated in Hilbert space as a function space solver with almost discretization independent behavior in that example.

12. SLPECEQP Implementation and Benchmark

In this chapter, we analyze the performance of the proposed SLPECEQP algorithm on the CUTEr and CUTEst set of benchmark problems for nonlinear programming.

The results of this chapter are published in [LKB17].

12.1. Implementation Details

We have implemented the SLPECEQP algorithm described in Chapter 7.5 using the Python scripting language and made the implementation open source [LKB17]. Similar to Matlab, Python is an interpreted language that provides fast methods to work on numerical data with the NumPy and SciPy packages [JOP+15]. Via the Cython package [Beh+11], C, C++, and Fortran code can be used directly from Python. Thus, rapid prototyping is possible while time critical components of the algorithm can be implemented in a compiled language.

For nonlinear programs, the LPEC of step a. of the SLPECEQP algorithm is a linear program and we use the dual simplex method of GuRoBi 6.0 [Gur15] to solve it. The Generalized Lanczos method described in Chapter 8 is used to solve the trust-region subproblem in the EQP step, where the projection systems in the preconditioned Krylov method are solved with the sparse indefinite solver MA57 [Duf04].

The implementation has been realized as a Python module. We allow slightly more general formulations for (NLP) with two-sided bounds for variables and constraints.

We use the following termination criterion that quantifies satisfaction of (KKT) relative to the initial point. To this end define

$$\begin{aligned}
\texttt{stat}^k &:= \|\nabla_x L(x^k, \lambda_{\mathrm{LS}}^k, \mu_{\mathrm{LS}}^k)\|_\infty, \\
\texttt{compl}^k &:= \|\mu_{\mathrm{LS}}^k * c_{\mathcal{I}}(x^k)\|_\infty, \\
\texttt{opt}^k &:= \max\{\texttt{stat}^k, \texttt{compl}^k\}, \\
\texttt{feas}_{\mathcal{E}}^k &:= \|c_{\mathcal{E}}(x^k)\|_\infty, \\
\texttt{feas}_{\mathcal{I}}^k &:= \|\min\{0, c_{\mathcal{I}}(x^k)\}\|_\infty, \\
\texttt{feas}^k &:= \max\{\texttt{feas}_{\mathcal{E}}^k, \texttt{feas}_{\mathcal{I}}^k\}.
\end{aligned}$$

Step of Algorithm	▼ mean %	var. %
Active Set Determination	55.5	3.4
EQP Setup	12.6	1.8
Line search d_{EQP}	11.4	0.7
EQP Solution	11.3	4.7
Factorization pCG	2.9	0.1
Penalty, Term. Test	2.6	0
Line search d_{C}	2.0	0.1
Second Order Corr.	1.7	0.1

Table 12.1.: Distribution of CPU time, excluding function evaluations.

The termination criterion requires now satisfaction of the following two conditions

$$\texttt{opt}^k \leq \begin{cases} 10^{-6} \max\{1, \min\{|f(x^k)|, \|\nabla f(x^0)\|_\infty\}\}, & \text{unconstrained case,} \\ 10^{-6} \max\{1, \|\nabla f(x^k)\|_\infty\}, & \text{constrained case,} \end{cases}$$
$$\texttt{feas}^k \leq 10^{-6} \max\{1, \|c_{\mathcal{E}}(x^0)\|_\infty, \|c_{\mathcal{I}}(x^0)\|_\infty\}.$$

12.2. Performance on CUTEst Benchmark Collection

We used CUTEst to analyze the performance of our implementation. 40 instances have been omitted from CUTEst for which evaluations fail due to, e.g., starting points for which functions are not well-defined, the remaining benchmark set then consists of 1109 instances that are considered. Figure 12.1 shows the ratio of instances that could be solved within a wall time limit of t, respective an iteration limit n or an limit n_{Hv} on the number of Hessian vector products.

Table 12.1 gives a breakdown of the relative computational cost of the steps of the algorithm, excluding function evaluations. As can be seen, LP solution to obtain an active set estimate and EQP solution to obtain a Newton-type step dominate the computational effort. Significant amounts of interpreted Python code are executed during EQP setup, trust region ratio computation, penalty function evaluation, and in the termination test. Here, one may expect speed-ups after reimplementation in a compiled language.

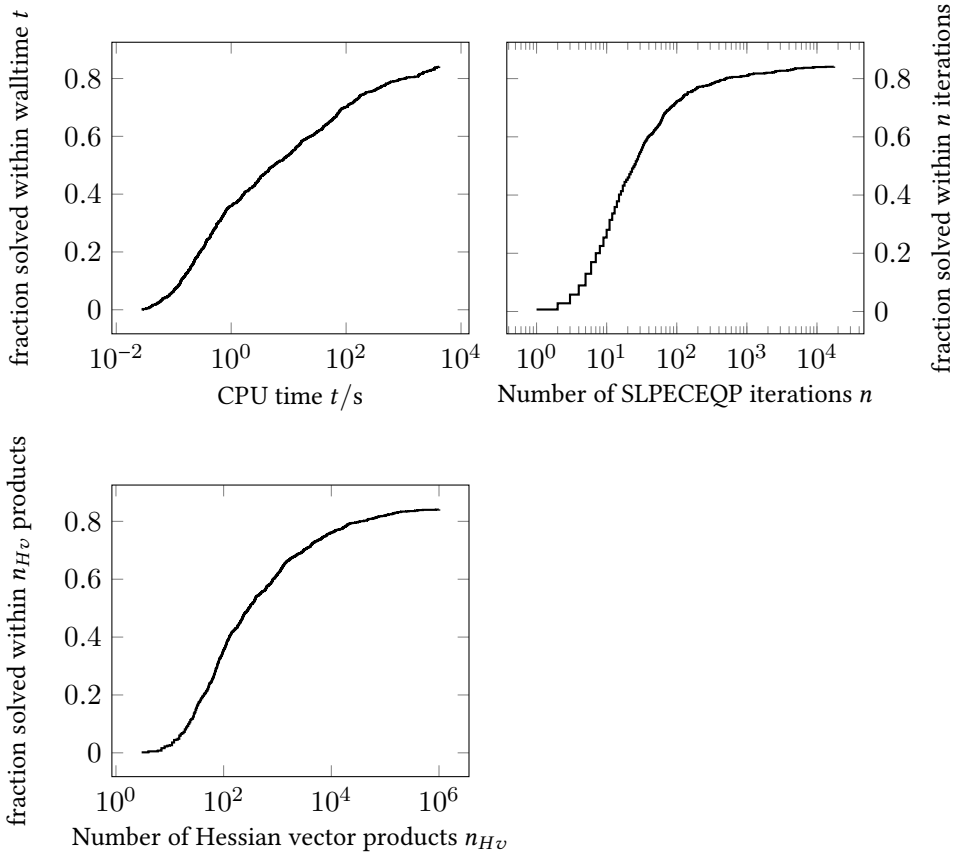


Figure 12.1.: Performance of implementation on CUTEst. Plots show the ratio of problems of the CUTEst benchmark collection solved within at most t seconds respective at most n iterations respective at most n_{Hv} products with the Hessian of the Lagrangian.

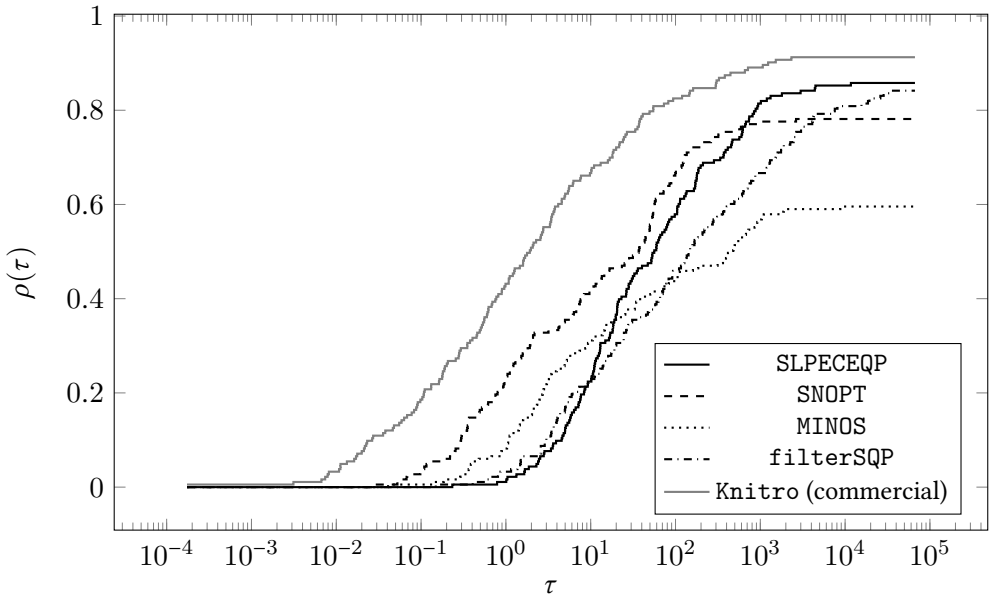


Figure 12.2.: Extended performance profile comparing the SLPECEQP implementation with state-of-the-art active set nonlinear programming solvers on AMPL translation of CUTEr.

12.3. Comparison with Active-Set Solvers on CUTEr Benchmark Collection

Not all the solvers we are using to compare our implementation directly support CUTEst. Hence, we have chosen to use the AMPL translation of CUTEr for that purpose. Like done for CUTEst, we omit all instances for which evaluations fail. We also omit instances that could be solved by any solver in less than 0.11 seconds which constitutes a test set including 183 problems. For such tiny instances, the overhead time required for starting the Python interpreter (0.11 seconds) dominates the actual solution time. Again, we imposed a wall time limit of one hour on the solution time per instance.

To compare our SLPECEQP implementation with the established active set solvers filterSQP [FL02], SNOPT [GMS02], MINOS [MS03], and the active-set solver of the commercial Knitro package [BNW06], we compute an extended performance profile. The result for the above subset of the CUTEr benchmark collection is shown in Figure 12.2. It can be seen that together with Knitro and filterSQP our implemen-

tation is among the most robust of the five solvers, in the sense that they solve the largest fraction of problems within the wall time limit. Our `SLPECEQP` implementation is in the interpreted language Python that incurs some speed limitations, and is hence not the overall fastest solver. Still, it achieves a performance that is competitive with other solvers that have been implemented in the compiled languages Fortran/C++.

12.4. Summary

In this chapter, we have considered the performance of our Python implementation of a `SLPECEQP` method for solving the nonlinear programming problem. On the well-established CUTEr benchmark collection, the implementation has been show to deliver competitive performance and to be more robust than three popular NLP solvers examined.

13. Implementation of Multiple Shooting Discretization

This chapter introduces the python implementation `OptimIND` that evaluates a multiple shooting discretization and their derivatives of an optimal control problem following the principle of internal numeric differentiation [Boc81a; Boc83] and automatic differentiation.

13.1. Problem Formulation

Let $t_0 = \tau_0 < \ldots < \tau_{N-1} \leq \tau_{N-1} = t_f$ be a grid of N nodes partitioning the time horizon $[t_0, t_f]$. The class of problems treated by `OptimIND` is then given by

$$\begin{aligned}
\min_{x,u,p} \quad & \int_{t_0}^{t_f} \|\ell(t, x, u, p)\|_2^2 + m(x(t_f), p) \\
\text{s.t.} \quad & \dot{x}(t) = f(x(t), u(t), p) && \text{a.e. } t \in [t_0, t_f], \\
& 0 = \textstyle\sum_{i \in [N]} d_i(\tau_i, x(\tau_i), u(\tau_i), p), \\
& \underline{c} \leq c(t, x(t), u(t), p) \leq \overline{c}, && \text{a.e. } t \in [t_0, t_f], \\
& \underline{x} \leq x(t) \leq \overline{x}, && \text{a.e. } t \in [t_0, t_f], \\
& \underline{u} \leq u(t) \leq \overline{u}, && \text{a.e. } t \in [t_0, t_f], \\
& \underline{p} \leq p \leq \overline{p}.
\end{aligned}$$

Here $x : [t_0, t_f] \to \mathbb{R}^{n_x}$ denotes states, $u : [t_0, t_f] \to \mathbb{R}^{n_u}$ controls, $p \in \mathbb{R}^{n_p}$ parameters. The function ℓ denotes a least-squares Lagrangian objective term, m a Mayer objective term. The dynamics exhibited by the states is defined by the function f, the function c is used to describe path constraints and the functions d_i to describe multi-point boundary constraints.

13.2. Multiple Shooting Discretization

We use Bock's Direct Multiple Shooting Method [Pli81; BP84; Lei95; Lei99; Lei+03a] to transform the optimal control problem into a nonlinear program. Let $V_i \subseteq L^\infty([\tau_i, \tau_{i+1}], \mathbb{R}^{n_u})$ be a finite dimensional vector space of control functions with a

chosen basis $\{\xi_{ij}, j \in [\dim V_i]\}$, denote $\xi_i : \mathbb{R}^{\dim V_i} \to V_i, q \mapsto \sum_{j\in[\dim V_i]} q_j \xi_{ij}$ the coordinate isomorphism. The shooting discretization introduces now on every shooting node $\tau_i, i \in [N]$ an initial value s^i and a control parameterization value q^i and requires continuity of the solution trajectory along the states. By $x^{i+1}(s^i, q^i, p)$ we denote the solution of the initial value problem $\dot{x} = f(t, x(t), \xi_i(q^i)(t), p), x(\tau_i) = s^i$ evaluated at τ_{i+1}. The least-squares Lagrangian objective function is discretized by a trapezoidal rule approximation in the shooting nodes with a weights vector σ. Path constraints are discretized by enforcing them at shooting nodes.

The resulting nonlinear program is now given by

$$
\begin{aligned}
\min_{s,q,p} \quad & \sum_{i\in[N]} \sigma_i \|\ell(\tau_i, s^i, \xi_i(q^i)(\tau_i), p)\|_2^2 + m(s^{N-1}, p) \\
\text{s.t.} \quad & 0 = x^{i+1}(s^i, q^i, p) - s^{i+1}, && i \in [N-1] \\
& 0 = \xi_{N-2}(q^{N-2})(\tau_{N-1}) - \xi_{N-1}(q^{N-1})(\tau_{N-1}), \\
& 0 = \textstyle\sum_{i\in[N]} d_i(\tau_i, s^i, \xi_i(q^i)(\tau_i), p), \\
& \underline{c} \le c(\tau_i, s^i, \xi_i(q^i)(\tau_i), p) \le \overline{c}, && i \in [N], \\
& \underline{x}^i \le s^i \le \overline{x}^i, && i \in [N], \\
& \underline{u}^i \le \xi_i(q^i)(\tau_i) \le \overline{u}^i, && i \in [N-1], \\
& \underline{p} \le p \le \overline{p}.
\end{aligned}
$$

13.3. Implementation `OptimIND`

For the control ansatz spaces, we used constant controls per shooting interval, i.e. $\xi_i(q_i)(t) = q_i$. To solve the initial value problems, we use the integrator package `SolvIND` that provides an adaptive BDF method `DAESOL-II` [Alb10] for stiff problems and a Runge-Kutta-Fehlberg method `RKFSWT` [Feh69; Feh70; Kir06] for non-stiff problems. For a definition of stiff and non-stiff problems and a description of the methods, we refer to the monographs by Hairer, Nørsett and Wanner, [HNW93; HW96]. A distinctive feature of `SolvIND` is the ability to compute sensitivities satisfying the principle of internal numerical differentiation [Boc81a; Boc83] combined with automatic differentiation [GW08; Spe80] and Taylor-coefficient propagation [BCG93; GW08].

`OptimIND` allows the specification of the functions ℓ, m, f, r, d as `SolvIND` model file in `C++` and provides a python interface to compute evaluations of the nonlinear program function evaluations and their first and second order derivatives.

13.4. Derivative Computation

For the SLPECEQP algorithm to be applied, first-order derivatives of all functions and second-order derivatives in the form of matrix-vector products with the Hessian of the Lagrangian of the nonlinear problem are required. Furthermore, to obtain derivatives of the nonlinear program, it is necessary to compute derivatives of the solutions x_{i+1} of the initial value problems with respect to initial values s_i and parameters p.

It is crucial that second-order derivative evaluations are consistent in the following sense: If $H(v) = Hv$ denotes the matrix-vector product with a direction v and exact Hessian H and evaluations compute the approximation $\hat{H}(v)$, then $v \mapsto \hat{H}(v)$ must be linear. Otherwise, rapid loss of orthogonality may happen in the Krylov subspace algorithm solving the equality constrained subproblem.

13.4.1. Automatic Differentiation and Taylor Coefficient Propagation

The Direct Multiple Shooting Discretization yields a *factorable programming* formulation [McC74; Sha80; Jac01] where all involved functions are *factorable functions*. Evaluating a factorable function f in a point x giving $y = f(x)$ on a computer is done by formulating an algorithm that computes $y = f(x)$ as a sequence of elemental operations like addition, subtraction, multiplication, division, exponential function etc, where every elemental function is locally an analytic function. This constitutes an evaluation graph with intermediate results as vertices and elemental operations as edges. Every analytic function has by definition a power series expansion and composition of convergent power series yields again a convergent power series that converges to the composition of the corresponding analytical functions. Using composition of power series as edges allows to define a lifted computational graph with nodes given by intermediate power series. Projection onto constant coefficient yields the original computational graph.

Forward mode. The forward mode of automatic differentiation uses the lift of the computational graph of f to power series and truncates it up to a certain order k. Evaluating $x + td$ for a given direction d yields $\sum_{i=0}^{k} \frac{f^{(i)}(x)d^i}{i!} t^i$ where $f^{(i)}(x)d^i$ is to be understood as i-fold contraction of $f^{(i)}$ with d.

Forward/reverse mode. Using forward mode with truncation order k and input $x + td$ and storing all intermediate results on a *tape* allows the application of the reverse mode where a direction y of dependent variables is required as additional input. Traversing the computational graph backwards after forward evaluation from

dependent to independent variables while accumulating derivative information allows to compute $\frac{\mathrm{d}}{\mathrm{d}x}\sum_{i=0}^{k+1}\frac{y^T f^{(i)}(x)d^i}{i!}t^i$.

Forward and forward/reverse mode only require a small multiple of the computational effort to compute $f(x)$, forward/reverse mode may require in addition a substantial amount of memory to store the tape. Using this approach it is possible to compute the Taylor coefficients $\frac{f^{(i)}(x)d^i}{i!}$ and $\frac{\mathrm{d}}{\mathrm{d}x}\frac{y^T f^{(i)}(x)d^i}{i!}$ respectively up to machine precision.

First-order derivatives. We require Jacobians $\frac{\mathrm{d}f}{\mathrm{d}x}$. These can be computed using the forward mode with $k = 1$ and $d = e_i$ for all basis vectors e_i, yielding $\frac{\mathrm{d}f}{\mathrm{d}x}e_i$ by extraction of the first Taylor coefficient. If f is defined on a subset of $\mathbb{R}^n$, n evaluations of forward directional derivatives are needed.

Second-order derivatives. We require Hessian-vector products $\frac{\mathrm{d}^2}{\mathrm{d}x^2}\lambda^T f v$. These can be computed using the forward/reverse mode with $k = 1$, $d = v$ and $y = \lambda$, the desired vector product is obtained by extraction of the first Taylor coefficient.

We use the implementation `ADOL-C` [WKG05] for automatic differentiation. It provides Taylor coefficient propagation that relies on operator overloading techniques and can be applied to algorithms defined in `C++` composed from smooth elemental functions. This approach allows to compute the necessary derivatives up to machine precision. In particular, matrix-vector products are consistent.

13.4.2. Derivatives of solutions to initial value problems

The solutions $x_{i+1}(s_i, q_i, p)$ of the initial value problems $\dot{x}(t) = f(t, x(t), \xi_i(q_i)(t), p)$, $x(\tau_i) = s_i$ evaluated at τ_{i+1} are computed using an adaptive discretization scheme for ordinary differential equations. Their first- and second-order derivatives with respect to initial values s_i and parameters q_i, p are required as well.

Several strategies can be considered to acquire these derivatives, where we choose the strategy of Internal Numerical Differentiation. We briefly discuss those strategies and their merits and disadvantages.

External Numerical Differentiation. Treating the adaptive discretization scheme as composition of smooth functions we could consider it as black box and apply automatic differentiation or finite difference approximation to obtain sensitivities. External Numerical Differentiation has the advantage that it is easy to implement. However, derivatives computed by external numerical differentiation are not consistent in the sense that they do not necessarily converge to the exact derivatives if integrator

accuracy is increased. Furthermore adaptive components are usually implemented by conditional statements that are non-smooth, so that $x_{i+1}^{\text{discretized}}(s_i, q_i, p)$ is *not a smooth function*! This does not allow the usage of automatic differentiation and, while it is still possible to compute finite difference approximations, these are inaccurate due to non-smooth behavior of $x_{i+1}^{\text{discretized}}$. For these reasons, it is not desirable to use external numerical differentiation, especially as we require second-order matrix-vector products to describe a linear operator which would be lost in this approach due to non-smoothness.

Variational Differential Equations. Variational differential equations in forward and adjoint mode can be formulated using a differentiate-then-discretize approach. They allow computations of the derivatives by solving augmented initial value problems. These yield consistent results even after discretization for the forward form, but are not necessarily consistent for the adjoint form as adaptive error control for the adjoint equation in general gives a different discretization scheme. For efficient computation of second-order derivatives the adjoint form is required, due to the inconsistency of second-order matrix-vector products it is not desirable to use variational differential equations.

Internal Numerical Differentiation. The principle of Internal Numerical Differentiation (IND) [Boc81a; Boc83] states that derivatives of an adaptive discretization scheme must be computed from the discretization scheme with all adaptive components kept frozen and must be convergent for the nominal value as well as for the derivative. Following the principle of IND solves the aforementioned problems. The integrator package `SolvIND` used in `OptimIND` implements IND by using automatic differentiation and skipping differentiation of adaptive components.

13.5. Summary

In this chapter, we introduced the implementation `OptimIND` of a Multiple Shooting Discretization framework. It makes use of internal numerical differentiation and automatic differentiation for sensitivity generation and in particular computes consistent Hessian-vector products.

14. Optimal Control Case Study: Re-entry of Apollo type space shuttle

In this chapter, we assess the performance of our SLPECEQP algorithm on the well-studied and challenging problem given by re-entry of a space shuttle of Apollo type. We compare using the SLPECEQP algorithm with the multiple shooting package `MUSCOD-II`.

14.1. Reentry problem

A well studied and interesting benchmark optimal control problem is given by re-entry of a space shuttle of Apollo type [Pli81; BP84; Pes89; SB02; Pot06] to be maneuvered into a position that is suitable for splashdown in the Pacific. The Space Shuttle Columbia disintegrated upon reentering Earth's atmosphere on February 1, 2003 killing all seven crew members, the incident is known as *Space Shuttle Columbia disaster* and demonstrates the importance to compute reliable solutions to this problem.

The problem is given by finding a trajectory for the space shuttle that allows splashdown at a defined favorable position in the Pacific while minimizing convective heating during the flight through the earth's atmosphere. This problem is challenging as the differential equations are highly nonlinear with solutions that are very sensitive to changes in initial values and controls and have moving singularities with solutions to the initial value problem only in a small vicinity of the solution of the optimal control problem.

Using the quantities defined in Table 14.1, the optimal control problem is stated as

Symbol	Description	Value
v	tangential velocity	
		$v_0 = 0.36 \cdot 10^5 \frac{\text{ft}}{\text{s}}$
		$v_f = 0.27 \cdot 10^5 \frac{\text{ft}}{\text{s}}$
γ	flight path angle	
		$\gamma_0 = -\frac{8.1^\circ \pi}{180^\circ}$
		$\gamma_f = 0$
R	earth's radius	$209 \cdot 10^5$ ft
ξ	normalized altitude above earth's surface	$\frac{h}{R}$
		$\xi_0 = \frac{4}{R}$
		$\xi_f = \frac{2.5}{R}$
ϱ	atmospheric density	$\varrho = \varrho_0 e^{-\beta R \xi}$
		$\varrho_0 = 2.704 \cdot 10^{-3} \frac{\text{slug}}{\text{ft}^3}$
		$\beta = 4.26 \cdot \frac{1}{10^5 \text{ ft}}$
u	angle of attack	
C_W	aerodynamical drag coefficient	$1.174 - 0.9 \cos u$
C_A	aerodynamical lift coefficient	$0.6 \sin u$
S/m	frontal area over mass of vehicle	$53200 \frac{\text{ft}^2}{\text{slug}}$
g	gravitational acceleration	$3.2172 \cdot 10^{-4} \frac{10^5 \text{ ft}}{\text{s}^2}$

Table 14.1.: Definition of quantities in re-entry problem

follows:

$$
\begin{aligned}
\min_{v,\gamma,\xi,u,T} \quad & \int_0^T 10\ v^3 \sqrt{\varrho}\ \mathrm{d}t \\
\text{s.t.} \quad & \dot{v} = -\tfrac{S\varrho v^2}{2m} C_W(u) - \tfrac{g \sin\gamma}{(1+\xi)^2}, \\
& \dot{\gamma} = \tfrac{S\varrho v}{2m} C_A(u) + \tfrac{v\cos\gamma}{R(1+\xi)} - \tfrac{g\sin\gamma}{v(1+\xi)^2}, \\
& \dot{\xi} = \tfrac{v\sin\gamma}{R}, \\
& v(0) = v_0, \qquad v(T) = v_f, \\
& \gamma(0) = \gamma_0, \qquad \gamma(T) = \gamma_f, \\
& \xi(0) = \xi_0, \qquad \xi(T) = \xi_f.
\end{aligned}
$$

To obtain the complete trajectory requires in addition integrating the distance ζ on the earth's surface, which satisfies the initial value problem

$$\dot{\zeta} = \tfrac{v}{1+\xi} \cos\gamma, \quad \zeta(0) = 0.$$

# shooting intervals	objective	constraint violation	# SLPECEQP iter.
14	$2.9487 \cdot 10^{-2}$	$1.13 \cdot 10^{-8}$	27
100	$2.7808 \cdot 10^{-2}$	$2.4 \cdot 10^{-13}$	38

Table 14.2.: Computational results in re-entry problem.

As ζ is decoupled from the objective function, the constraints and the differential equations for v, γ, ξ, it is not necessary to treat it in the optimal control problem.

14.2. Computational results using OptimIND and SLPECEQP

We used the Software Package OptimIND to solve the optimal control problem with integrator DAESOL-II and SLPECEQP using an equidistant multiple shooting grid with piecewise constant controls, that are constant on every shooting interval. The problem has been initialized with a constant control $u \equiv 0.1$ and initial values 0 on the multiple shooting nodes with the exception of the initial and final shooting node, where $(v_0, \gamma_0, \xi_0)^T$ and $(v_f, \gamma_f, \xi_f)^T$ respectively have been used.

We found that the problem is infeasible if less than 14 multiple shooting intervals are used and report objective function value, constraint satisfaction and number of SLPECEQP iterations in Table 14.2. On the finest discretization with 100 shooting intervals, the SLPECEQP algorithm converged with 38 iterations with 24 accepted trust region attempts and 14 discarded attempts. This illustrates the high nonlinearity of the problem, as in about a third of the cases the prediction of the quadratic model did not capture the real behavior. In total 946 Hessian vector products had to be computed, where the maximum of Hessian vector products in an iteration was 88. 98.8% of the CPU time was spent in evaluation, 0.73% on the solution of the linear programs, 0.2% on the solution of the equality constraint quadratic programs, 0.19% on the factorization of the projection matrix and 0.07% on the remaining computations necessary.

The optimal value T of the free end time has been determined to $T = 225.00$. Figure 14.1 shows the optimal trajectories and control.

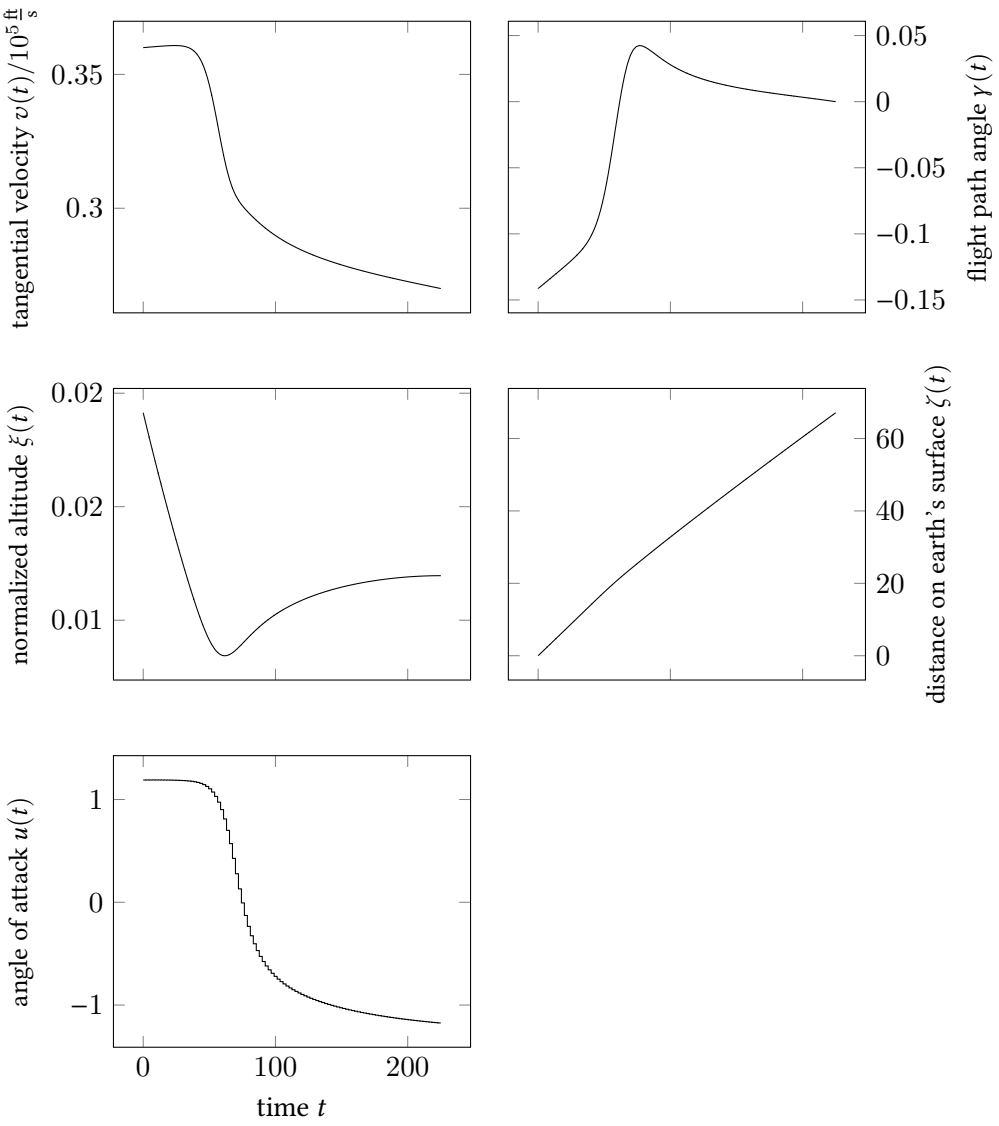


Figure 14.1.: Optimal control and trajectory in re-entry problem on fine grid with 100 shooting intervals computed by `OptimIND` and `SLPECEQP`.

14.3. Computational results with MUSCOD-II

We first note that the Open Source solver SLSQP [Kra88] failed on a multiple shooting discretization of the problem and the Open Source solver IpOpt [Wäc02; WB06] required a huge number of interior point iterations and are thus no viable option to solve this problem. For comparison purposes we thus have solved the optimal control problem also with MUSCOD-II [Lei99; Die+16]. MUSCOD-II employs a similar multiple shooting discretization and uses a structure exploiting SQP algorithm to solve the discretized problem. The re-entry problem is part of the examples implemented in MUSCOD-II, where a coarse control discretization on the shooting grid

$$0, 0.25T, 0.375T, 0.5T, 0.675T, 0.75T, T$$

with continuous piecewise linear controls is used. A Runge-Kutta-Fehlberg integrator of order 4/5 with internal numerical differentiation is used for discretization of the ordinary differential equation and finite difference approximations for the Hessian approximation in the SQP method.

It was not possible to solve the optimal control problem with MUSCOD-II using the identical discretization, as the SQP algorithm failed on a discretization with equidistant grid and either piecewise constant or piecewise controls for all grid sizes up to 100 multiple shooting nodes. The failure was always due to the employed quadratic program subproblem solver QPOPT [GMS95] not being able to solve the quadratic program within the increased iteration limit of 10^8 QP iterations, regardless if condensing for structure exploitation is used or not.

The only comparison is thus possible with the solution using the discretization with continuous piecewise linear controls on the above mentioned non-equidistant shooting grid. Initial point in MUSCOD-II was chosen by the standard option given by linear interpolation between $(v_0, \gamma_0, \xi_0)^T$ and $(v_f, \gamma_f, \xi_f)^T$ for the initial values of the states on the shooting nodes and $u \equiv 0.1$ as in the SLPECEQP case. The SQP methods converges with objective function value 0.027827 and terminal time $T = 225.35$ in 9 iterations with 82% of the CPU time spent in evaluations, 4.9% on condensing, 2.5% on solution of the condensed quadratic program and 10% for remaining calculations. Figure 14.2 shows the optimal trajectories and control.

Comparing MUSCOD-II and solving the discretized optimal control problem with the SLPECEQP algorithm, it turns out that the SLPECEQP method works for a wider variety of discretizations. To set up and solve the re-entry problem in MUSCOD-II, expert knowledge is required to find the non-equidistant grid used in the discretization.

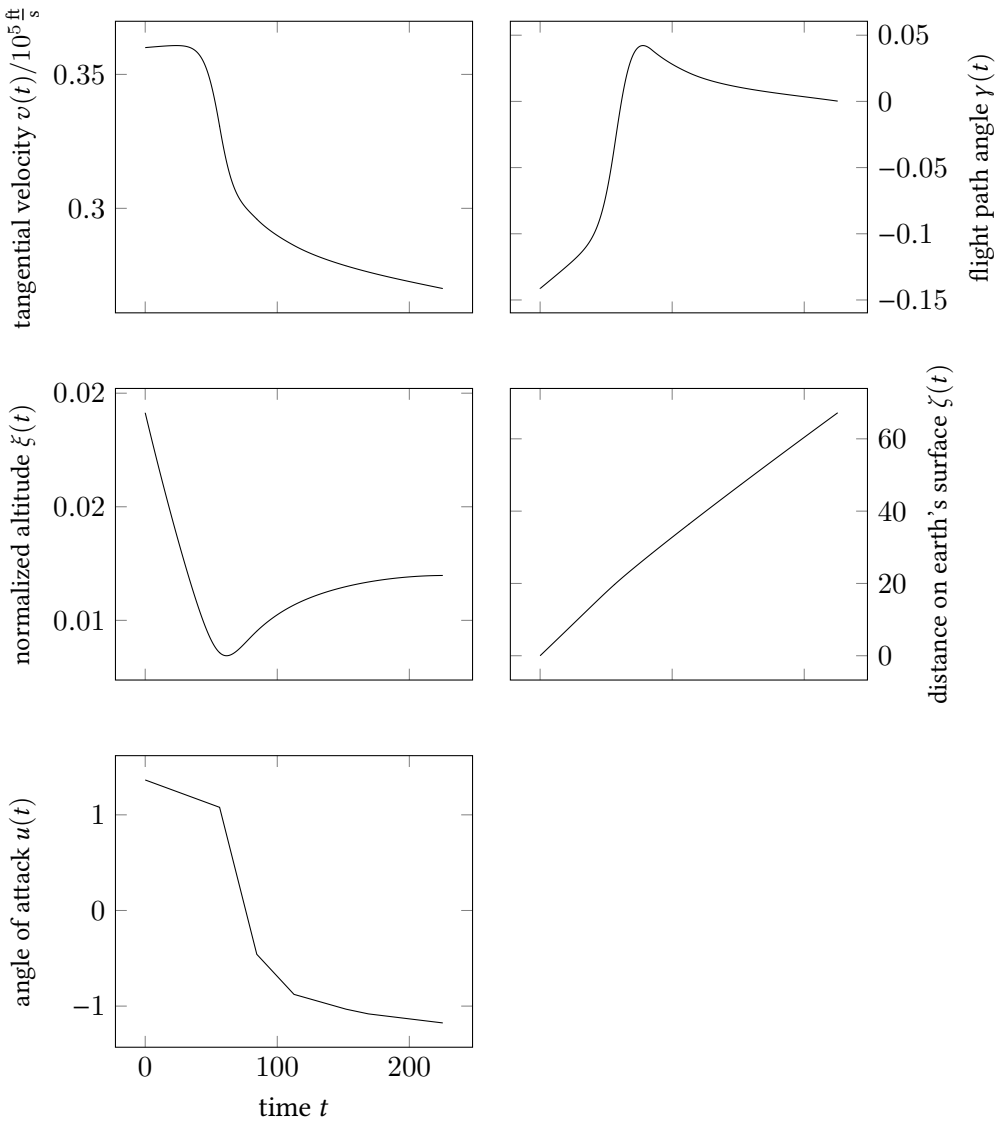


Figure 14.2.: Optimal Control and Trajectory in re-entry problem as computed by MUSCOD-II.

14.4. Summary

In this chapter, we have tested the SLPECEQP algorithm on the challenging re-entry problem and compared it to both packages found comparable solutions. We could employ the SLPECEQP on a range of discretizations while `MUSCOD-II` could only be used with an expert provided discretization on a non-equidistant grid.

15. Mixed-Integer Optimal Control Case Study: Egerstedt Example

In this chapter, we study an extension of a problem due to Egerstedt as benchmark case for mixed-integer optimal control. We apply the partial outer convexification and relaxation approach of Chapter 6 and the SLPECEQP algorithm on the discretized problem.

Parts of the results of this chapter are published in [KL16].

15.1. Problem Formulation

We use a modified version of an example originally due to Egerstedt [EWD03; EWA06] that has been extensively studied as benchmark example for mixed-integer optimal control by Sager [SBD12; Bur11].

Example by Sager and relationship to the example of Egerstedt

In [SBD12] the example is formulated as the following problem:

$$\begin{aligned}
\min_{x,\omega} \quad & \int_0^1 \|x\|^2 \, \mathrm{d}t \\
\text{s.t.} \quad & \dot{x}_0 = -x_0\omega_0 + (x_0 + x_1)\omega_1 + (x_0 - x_1)\omega_2, \\
& \dot{x}_1 = (x_0 + 2x_1)\omega_0 + (x_0 - 2x_1)\omega_1 + (x_0 + x_1)\omega_2, \\
& x(0) = (\tfrac{1}{2}, \tfrac{1}{2})^T, \\
& 0.4 \leq x_0, \\
& 1 = \omega_0 + \omega_1 + \omega_2, \quad \omega(t) \in \{0,1\}^3.
\end{aligned}$$

In contrast, Egerstedt originally formulated the problem

$$\begin{aligned}
\min_{x} \quad & \int_0^1 \|x\|^2 \, \mathrm{d}t \\
\text{s.t.} \quad & \dot{x} \in \{A_1 x, A_2 x\}, \\
& x(0) = (1,0)^T.
\end{aligned}$$

which has been modified by Szymkat and Korytowski [SK08] by replacing the time horizon $[0, 10]$ with $[0, 1]$ and the differential inclusion initial value problem with $\dot{x} \in \{A_1 x, A_2 x, A_3 x\}$, $x(0) = (\frac{1}{2}, \frac{1}{2})$. The matrices are given by

$$A_1 = \begin{pmatrix} -1 & 0 \\ 1 & 2 \end{pmatrix},$$
$$A_2 = \begin{pmatrix} 1 & 1 \\ 1 & -2 \end{pmatrix},$$
$$A_3 = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}.$$

The formulation of Sager arises if binary indicator functions $\omega_0, \omega_1, \omega_2$ are introduced for the choices in the differential inclusion and the additional path constraint $x_0(t) \geq 0.4$ is introduced. Note that the computation in [SBD12] has been performed with the constraint $x_0(t) \geq 0.4$, while the paper states $x_1(t) \geq 0.4$.

To study the effect of different convexification schemes, Sager presents also a version of the problem with nonlinear dependence on the controls.

Modified Example with Vanishing Constraint

As a benchmark problem for mixed-integer optimal control with control dependent constraint we modify the problem by adding the constraint $x_0(t)(\omega_0(t) + 2\omega_1(t)) + x_1(t)\omega_2(t) - 1 \geq 0$ and express it as mixed-integer optimal control problem with vanishing constraint formulation:

$$\begin{aligned}
\min_{x,\omega} \quad & \int_0^1 \|x\|^2 \, \mathrm{d}t \\
\text{s.t.} \quad & \dot{x}_0 = -x_0\omega_0 + (x_0 + x_1)\omega_1 + (x_0 - x_1)\omega_2, \\
& \dot{x}_1 = (x_0 + 2x_1)\omega_0 + (x_0 - 2x_1)\omega_1 + (x_0 + x_1)\omega_2, \\
& x(0) = (\tfrac{1}{2}, \tfrac{1}{2})^T, \\
& 0.4 \leq x_0, \\
& 0 \leq \omega_0(x_0 - 1), \\
& 0 \leq \omega_1(2x_0 - 1), \\
& 0 \leq \omega_2(x_1 - 1), \\
& 1 = \omega_0 + \omega_1 + \omega_2, \quad \omega(t) \in \{0, 1\}^3.
\end{aligned}$$

15.2. Comparing SLPECEQP with Hoheisel's Regularization and IpOpt

15.2.1. Comparing SLPECEQP and MUSCOD-II on Problem without Vanishing Constraint

We study solving the relaxed problem without vanishing constraint that has been considered by Sager [SBD12]:

$$
\begin{aligned}
\min_{x,\alpha} \quad & \int_0^1 \|x\|^2 \, \mathrm{d}t \\
\text{s.t.} \quad & \dot{x}_0 = -x_0\alpha_0 + (x_0 + x_1)\alpha_1 + (x_0 - x_1)\alpha_2, \\
& \dot{x}_1 = (x_0 + 2x_1)\alpha_0 + (x_0 - 2x_1)\alpha_1 + (x_0 + x_1)\alpha_2, \\
& x(0) = (\tfrac{1}{2}, \tfrac{1}{2})^T, \\
& 0.4 \leq x_0, \\
& 1 = \alpha_0 + \alpha_1 + \alpha_2, \quad \alpha(t) \in [0,1]^3.
\end{aligned}
$$

This problem is bilinear in states and controls as the right hand side depends linearly on controls and constraints and the objective dependence is quadratic on states. SLPECEQP makes use of the exact Hessian which is singular. We have observed that this singular dependence leads to inefficiencies and it is better to solve the equivalent problem given by the identification $\alpha_i := \beta_i^2$.

We used the same equidistant shooting discretizations with 80 shooting intervals as in [SBD12] with Gauß-Newton approximation to the objective Hessian matrix. After 58 SLPECEQP iterations with a total number of 214 Hessian vector products, we found a solution satisfying KKT conditions with residual $1.6 \cdot 10^{-5}$ with objective function value 0.995590. It was not possible to compute solutions to higher accuracy as the Jacobian matrix in this point has a condition number of $2.4 \cdot 10^{10}$ and the KKT matrix of $2.4 \cdot 10^{11}$, indicating that the problem may be singular in the solution.

We also solved the problem using MUSCOD-II where the original problem formulation could be used by employing a positive definite structured BFGS Hessian approximation [Bro70; Fle70; Gol70; Sha70; BP84] that is not as affected by the singularity problems as the exact Hessian SLPECEQP algorithm. Trying to solve the problem with finite difference exact Hessian approximation within MUSCOD-II failed upon solving the first QP subproblem due to the singularity. MUSCOD-II converged with the same shooting discretization in 40 SQP iterations with an objective function value of 0.995593, that is comparable within the integration tolerance but slightly worse to the function value computed with the SLPECEQP algorithm. Sager reports a slightly better objective function value of 0.995569 which is comparable to the objec-

tive function values computed by SLPECEQP and MUSCOD-II within the integration tolerances. Controls and Trajectories are shown in Figure 15.1.

15.2.2. Solution of Relaxed Problem with Vanishing Constraints

We employed a discretization with 512 shooting intervals for the problem with vanishing constraints within the shooting framework OptimIND and used again a Gauß-Newton approximation for the objective Hessian matrix. A slack reformulation has been used to formulate the vanishing constraints as complementarity constraint.

We used the proposed SLPECEQP algorithm within in the SLPECEQP implementation with the Augmented Lagrangian Gradient Projection solver ALGRAPS to solve the LPEC subproblem to solve the relaxed problem with vanishing problems, that has been reformulated in a complementarity constraint form.

Solving the problem on a fine discretization using $(x_0, x_1, \alpha_0, \alpha_1, \alpha_2) \equiv (\frac{1}{2}, \frac{1}{2}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ as initial point on all shooting nodes lead the algorithm frequently to points of local infeasibility with the penalty parameter rising to it's upper bound $\mu = 10^{20}$. Due to the high condition numbers of the problem's constraint Jacobian, no further progress in decreasing infeasibility was possible. We thus used a bootstrapping and shooting based restoration strategy to solve the problem on a fine discretization by solving the problem on a sequence of grids G_i with 2^i shooting intervals for $i = 3, \ldots, 9$ and initializing the problem on the grid G_{i+1} with the control obtained by the solution on the grid G_i and states obtained by forward integration using the initial value constraint. If the algorithm terminated in a point of local infeasibility, we used forward integration as restoration mechanism.

Using this scheme and a grid with $512 = 2^9$, we could solve the problem equidistant multiple shooting intervals to a KKT tolerance of 10^{-3} with the integrator DAESOL-II. Solving the problem on the finest grid G_9 required 12 SLPECEQP iterations with a total number of 27 Hessian vector product evaluations. A higher accuracy could not be achieved due to the condition of the problems constraints with Jacobian condition number of 10^{10}.

For comparison purposes, we used CasADi [AÅD12] and IpOpt [Wäc02; WB06] to solve a smoothened vanishing constraint formulation while ensuring that the smoothened problem satisfies constraint qualifications. We use the smoothing proposed by Hoheisel [Hoh09], where the vanishing constraints of type $0 \leq xc(x)$ are replaced by $\phi^\tau(-c(x), x) \leq \tau$, where the smoothing function for $\tau > 0$ is given by

$$\phi^\tau(a, b) := \tfrac{1}{2}(ab + \sqrt{a^2 b^2 + \tau^2} + \sqrt{b^2 + \tau^2} - b).$$

We used values $\tau = 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}, 10^{-8}$, for values $\tau < 10^{-9}$ we could not achieve convergence with IpOpt. Computational results are listed in Table 15.1,

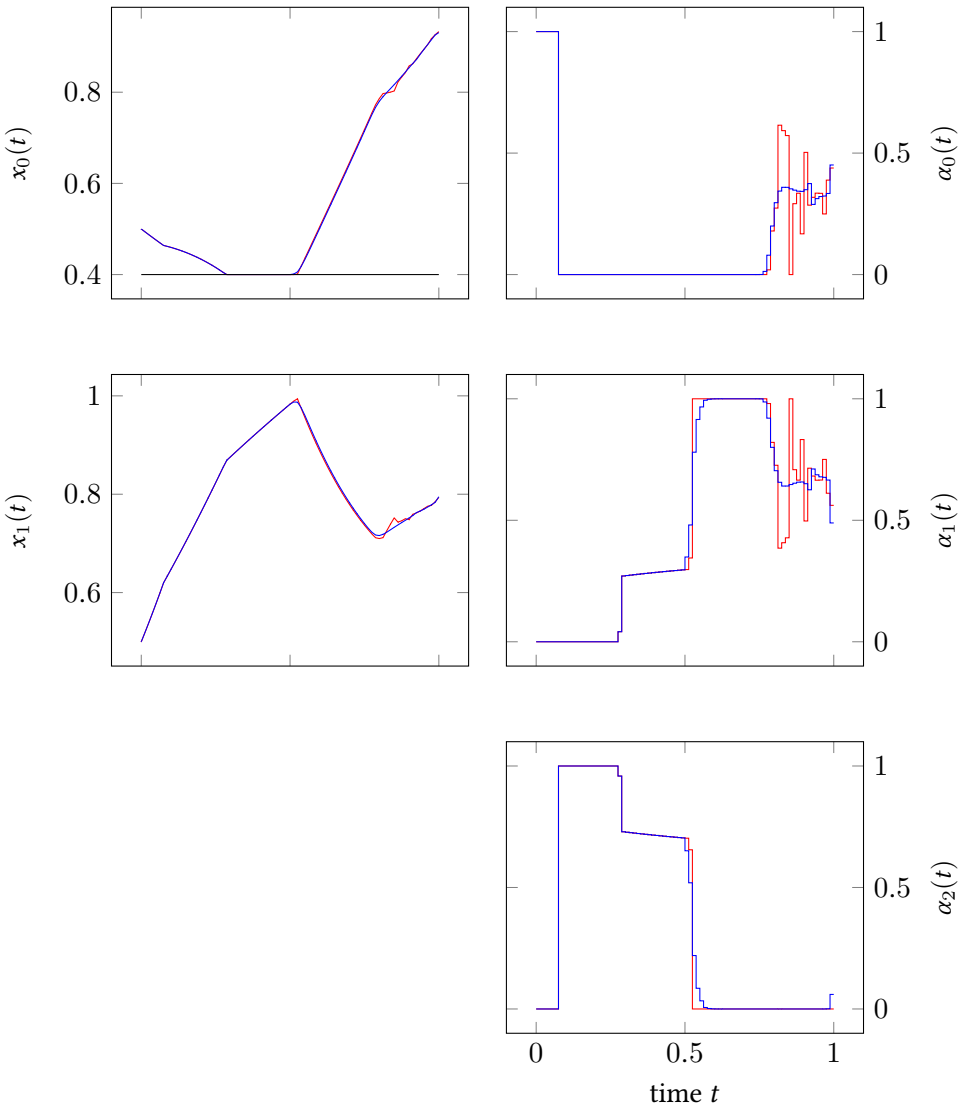


Figure 15.1.: Optimal Control and Trajectory in relaxed Egerstedt problem on 80 shooting intervals. Control and Trajectory computed using SLPECEQP are shown in blue and control and trajectory computed using MUSCOD-II in red.

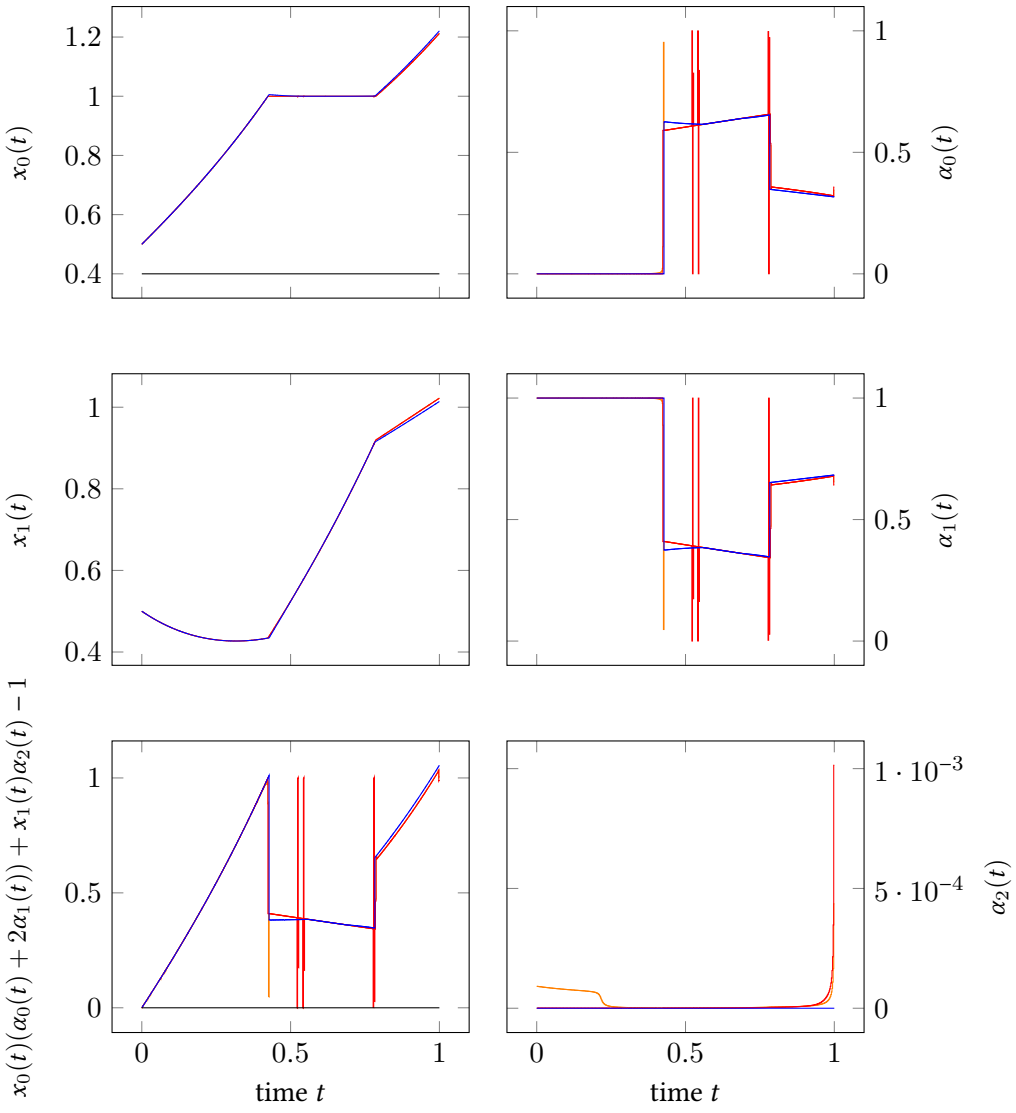


Figure 15.2.: Optimal Control and Trajectory in relaxed Egerstedt problem with vanishing constraint on 512 shooting intervals. Control and Trajectory computed using SLPECEQP are shown in blue and control and trajectory computed with `IpOpt` and Hoheisel smoothing in orange ($\tau = 10^{-4}$) and red ($\tau = 10^{-6}$).

solver	problem formulation	objective	VC violation
SLPECEQP	MPEC	1.313759	0
IpOpt	smoothened MPVC $\tau = 10^{-4}$	1.312688	$7.9 \cdot 10^{-5}$
IpOpt	smoothened MPVC $\tau = 10^{-5}$	1.312859	$1.3 \cdot 10^{-5}$
IpOpt	smoothened MPVC $\tau = 10^{-6}$	1.312873	$6.1 \cdot 10^{-6}$
IpOpt	smoothened MPVC $\tau = 10^{-7}$	1.313684	$2.4 \cdot 10^{-4}$
IpOpt	smoothened MPVC $\tau = 10^{-8}$	1.313515	$3.9 \cdot 10^{-6}$

Table 15.1.: Computational results for relaxed problem with vanishing constraints, comparing SLPECEQP solution with smoothed MPVC solution for different values of the smoothing parameter τ. Vanishing Constraint violation denotes the largest constraint violation of any vanishing constraint on any shooting interval.

controls and trajectories shown in Figure 15.2. Using the approach with smoothing of the vanishing constraint yields a solution with better objective function value, but constraint violation in the range 10^{-5}–10^{-6} for all values of τ that have been tested. In contrast, the SLPECEQP approach yields a solution that maintains strict feasibility with respect to the vanishing constraint. Decreasing the value of τ yields highly oscillatory solutions as can be seen in the plot of the controls in Figure 15.2. The orange lines corresponding to $\tau = 10^{-4}$ are much smoother than the red lines corresponding to $\tau = 10^{-6}$.

15.3. Comparison of Rounding Scheme with Branch & Bound Solver Bonmin

We used (VC-SOS-SUR) introduced in Chapter 6 to compute binary feasible solutions by rounding on N equidistant intervals from the solution of the relaxed problem on the finest shooting grid.

In addition, we compare the VC-SOS-SUR solution to the solution obtained with state-of-the-art mixed-integer nonlinear programming solver Bonmin [Bon+08]. As Bonmin requires definition of the optimization problems in the modeling language AMPL [FGK90; FGK02] which does not provide integrators, a similar multiple shooting discretization was not possible. We thus used a discretization partitioning the time interval into N equidistant intervals with constant controls on each interval and discretized the differential equation with an implicit Euler discretization using 400

N	objective			constraint violation	
	BB	VC-SOS-SUR	gap	shooting nodes	along trajectory
8	1.37148	1.350888	$3.7 \cdot 10^{-2}$	$1.3 \cdot 10^{-1}$	$1.3 \cdot 10^{-1}$
16	1.32679	1.315212	$1.5 \cdot 10^{-3}$	$6.6 \cdot 10^{-2}$	$6.8 \cdot 10^{-2}$
32	1.31965	1.316310	$2.5 \cdot 10^{-3}$	$3.7 \cdot 10^{-2}$	$3.8 \cdot 10^{-2}$
64	timeout	1.315079	$1.3 \cdot 10^{-3}$	$1.6 \cdot 10^{-2}$	$1.8 \cdot 10^{-2}$
128	timeout	1.313661	$-9.8 \cdot 10^{-5}$	$8.3 \cdot 10^{-3}$	$1.0 \cdot 10^{-2}$
256	timeout	1.313700	$-5.8 \cdot 10^{-5}$	$3.3 \cdot 10^{-3}$	$5.2 \cdot 10^{-3}$
512	timeout	1.313747	$-1.2 \cdot 10^{-5}$	$6.4 \cdot 10^{-4}$	$2.6 \cdot 10^{-3}$

Table 15.2.: Binary feasible solution obtained with branch-and-bound and VC-SOS-SUR scheme. With branch-and-bound using the `Bonmin`, computations terminated only for $N \leq 32$ within a walltime limit of 24 hours. Objective gap denotes difference between objective function value of relaxed solution and of rounded solution obtained with VC-SOS-SUR scheme. A negative gap may occur by the slight constraint violation.

steps on each interval.

Figure 15.3 and Figure 15.4 show the relaxed solution and the binary feasible point for $N = 32$ and $N = 128$. Table 15.1 and Figure 15.5 show the computational results. Using `Bonmin`, we have been able to compute the solution within a walltime of 24 hours only for $N \leq 32$. In contrast, the solution obtained by the VC-SOS-SUR scheme has a computational effort linear in N and can be computed in milliseconds once the relaxed problem has been solved. The solutions obtained via the VC-SOS-SUR are approximately feasible with feasibility violation showing $1/N$ behavior and approximately optimal with optimality gap in comparison to relaxed problem showing $1/N^2$ behavior.

15.4. Summary

We considered an extended variant of the Egerstedt benchmark example for mixed-integer optimal control. We compared the SLPECEQP algorithm and `MUSCOD-II` on a relaxed problem variant without vanishing constraints and SLPECEQP and `IpOpt` with Hoheisel's regularization scheme on the relaxed problem with vanishing constraint. We demonstrated the effectiveness and efficiency of the rounding scheme by comparison with the branch-and-bound solver `Bonmin`.

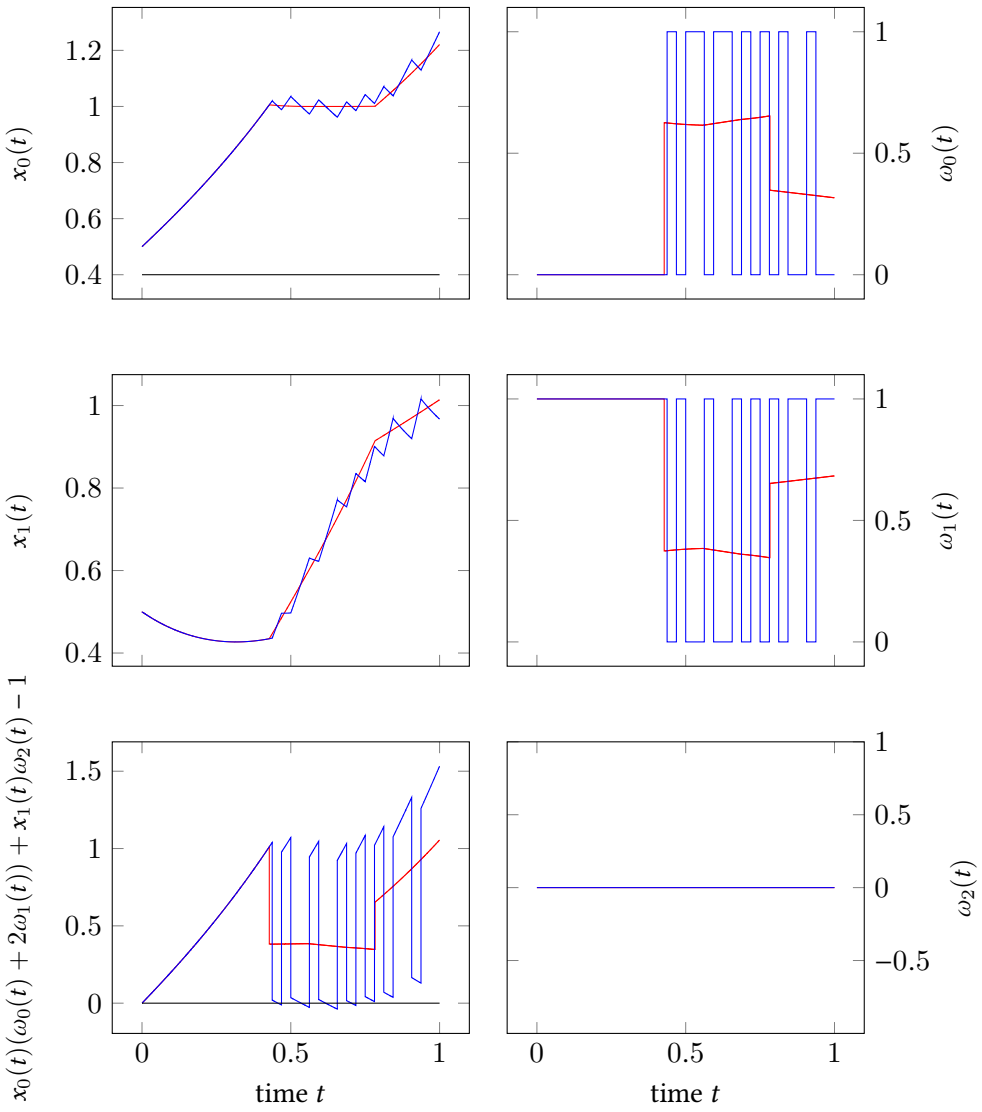


Figure 15.3.: Relaxed solution in red and binary feasible solution in blue obtained by rounding with VC-SOS-SUR scheme on 32 intervals.

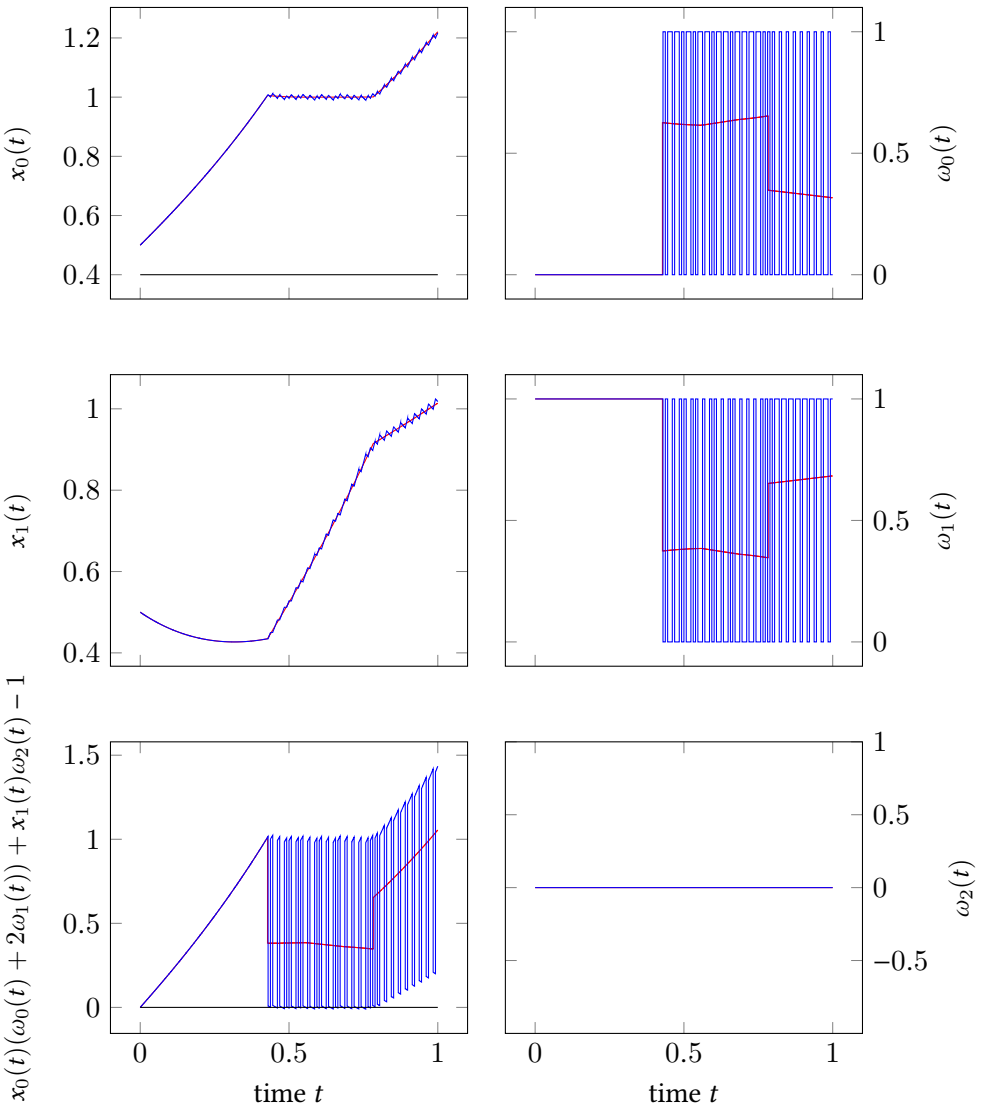


Figure 15.4.: Relaxed solution in red and binary feasible solution in blue obtained by rounding with VC-SOS-SUR scheme on 128 intervals.

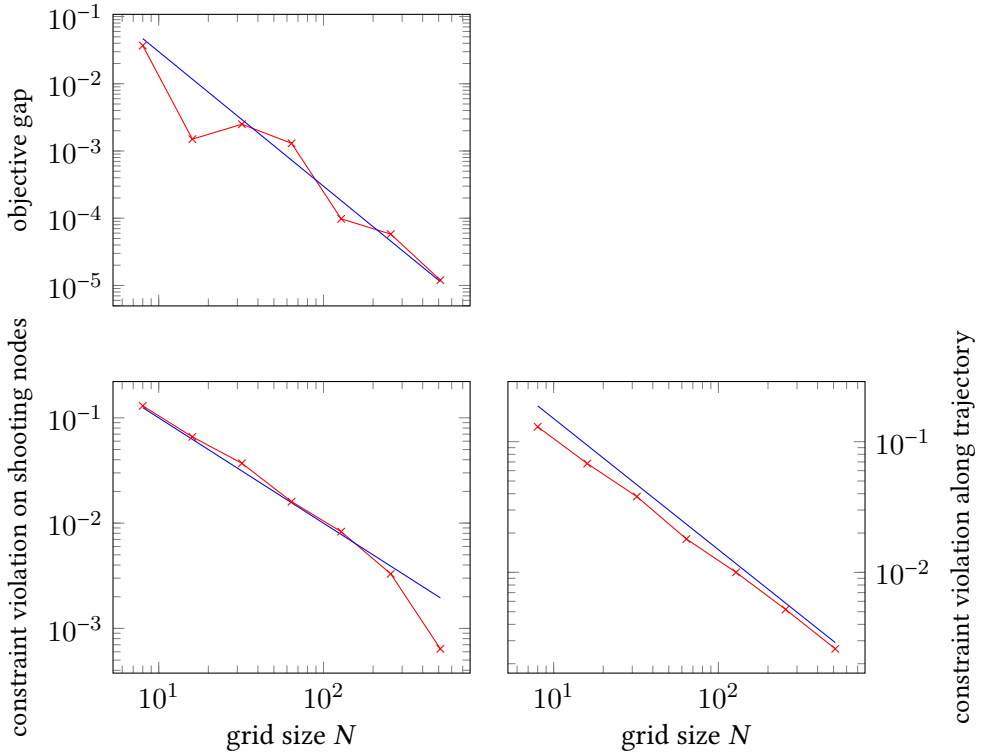


Figure 15.5.: Approximation properties of solution obtained via VC-SOS-SUR scheme. It can be seen that for this example the constraint violation has $1/N$ behavior and the objective gap has $1/N^2$ behavior. Blue lines show $1/N^2$ dependence for the objective gap respectively $1/N$ dependence for the constraint violation.

16. Nonlinear Model Predictive Case Studies

In this chapter, we study a nonlinear batch reactor and a continuous stirred tank reactor as case studies for the applicability of the SLPECEQP algorithm for model predictive control. The continuous stirred tank reactor has a least-squares tracking type objective that allows application of the Gauß-Newton preconditioner introduced in Chapter 9 and study it's performance at the hand of this example. We use Pontryagin's Maximum Principle to compute reference solutions to the offline problems.

16.1. Real-Time Feasibility for Nonlinear Batch Reactor

Optimal control and model predictive control of chemical batch reactor with nonlinear dynamics is analyzed in section. The model has been described by Biegler [Bie84] and has been studied as benchmark problem for optimal control by Biegler, Renfro [Ren86], Logsdon and Biegler [LB89; LB92] and Leineweber [DLS01] and as benchmark problem for model predictive control by Kirches [Kir+12]. Biegler and Logsdon note that this problem is interesting as optimal control benchmark problem, as the control becomes saturated it is difficult to compute an accurate solution if a direct method is used on a fixed grid without adaptive grid refinement.

The batch reactor is assumed to operate over a one hour period producing two products B and C in parallel reactions $A \to B$ and $B \to C$ that are irreversible and first order in A. Reaction rates are given by $k_{A\to i} = k^0_{A\to i} \exp(-E_{A\to i}/RT)$ with $k^0_{A\to B} = 10^6 \frac{1}{\mathrm{h}}, k^0_{A\to C} = 5 \cdot 10^{11} \frac{1}{\mathrm{h}}, E_{A\to B} = 10^4 \frac{\mathrm{cal}}{\mathrm{g\,mol}}, E_{A\to C} = 2 \cdot 10^4 \frac{\mathrm{cal}}{\mathrm{g\,mol}}$. Note that the reference [Bie84] states units $1/\mathrm{s}$ for the pre-exponential factors, which does not match the optimal control problem stated in hours without time transformation.

Dynamics of the reactor are then described by

$$\begin{aligned}
\dot{c}_A &= -k_{A\to B} c_A - k_{A\to C} c_A, \\
\dot{c}_B &= k_{A\to B} c_A, \\
\dot{c}_C &= k_{A\to C} c_A.
\end{aligned}$$

The goal is to find a temperature control $T \le 412\,K$ that maximizes the yield of B after one hour. Introducing normalized concentrations $x_A := \frac{1}{c_A(0)} c_A, x_B := \frac{1}{c_A(0)} c_B$

and noting that $T \mapsto u := k_{A\to B} = k^0_{A\to B}\exp(-E_{A\to B}/RT)$ is bijective with $k_{A\to C} = \frac{1}{2}k^2_{B\to C} = pu^2$ with $p := \frac{1}{2}$, the problem under consideration is the following:

$$\begin{aligned} \min_{x_A, x_B, u} \quad & x_B(1\,\mathrm{h}) \\ \text{s.t.} \quad & \dot{x}_A = -ux_A - pu^2x_A, \\ & \dot{x}_B = ux_A, \\ & x_A(0) = 1, \quad x_B(0) = 0, \\ & 0 \le u \le 5\,\tfrac{1}{\mathrm{h}}. \end{aligned}$$

Solution to the optimal control problem using Maximum Principle

As reference we have computed the function space solution satisfying necessary conditions given by Pontryagin's Maximum Principle [Pon+61].

Defining the Hamilton function $H(x, \lambda, u) := -\lambda_A x_A(u + \frac{p}{u}^2) + \lambda_B x_A u$, the Maximum Principle asserts that if x^*, u^* is a solution to the problem there are costates $\lambda^* = (\lambda^*_A, \lambda^*_B)^T$ such that the following conditions are met:

(1) $\dot{x}^* = \nabla_\lambda H(x^*, \lambda^*, u^*)$,

(2) $\dot{\lambda}^* = -\nabla_x H(x^*, \lambda^*, u^*)$,

(3) $H(x^*, \lambda^*, u^*) = \max_{u\in[0,5]} H(x^*, \lambda^*, u)$,

(4) $\lambda^*(1\,\mathrm{h}) = (0, 1)^T$,

(5) $t \mapsto H(x^*(t), \lambda^*(t), u^*(t))$ is constant.

Expanding the Hamiltonian and using (1)–(4) leads to the following two-point boundary value problem:

$$\begin{aligned} u &= \begin{cases} \frac{1}{2p}(\frac{\lambda_B}{\lambda_A} - 1), & \lambda_A x_A \ge 0 \text{ and } 1 \le \frac{\lambda_B}{\lambda_A} \le 6, \\ 0 \text{ or } 5, & \text{else}, \end{cases}, \\ \dot{x}_A &= -ux_A - pu^2x_A, \quad \dot{x}_B = ux_A, \\ \dot{\lambda}_A &= \lambda_A(u + pu^2) - \lambda_B u, \quad \dot{\lambda}_B = 0, \\ x_A(0) &= 1, \quad x_B(0) = 0, \quad \lambda_A(1\,\mathrm{h}) = 0, \quad \lambda_B(1\,\mathrm{h}) = 1. \end{aligned}$$

Using $\dot{\lambda}_B = 0$ and $\lambda_B(1\,\mathrm{h}) = 1$ allows immediately to eliminate $\lambda_B \equiv 1$. Using a computation with the direct multiple shooting method, we estimated that there is one switching point $\tau \in [0, 1\,\mathrm{h}]$ where u is free in $[0, \tau]$ and is at its upper bound in $[\tau, 1\,\mathrm{h}]$, which is confirmed a posteriori by the solution of the boundary value problem.

We denote by $I(t_1, t_0, \xi_A, \xi_B, \xi_\lambda)$ the solution evaluated at t_1 of the initial value problem

$$\begin{aligned} u &= \begin{cases} \frac{1}{2p}(\frac{\lambda_B}{\lambda_A} - 1), & \lambda_A x_A \geq 0 \text{ and } 1 \leq \frac{\lambda_B}{\lambda_A} \leq 6, \\ 0 \text{ or } 5, & \text{else}, \end{cases}, \\ \dot{x}_A &= -ux_A - pu^2 x_A, \quad \dot{x}_B = ux_A, \\ \dot{\lambda}_A &= \lambda_A(u + pu^2) - \lambda_B u, \\ x_A(t_0) &= \xi_A, \quad x_B(t_0) = \xi_B, \quad \lambda_A(t_0) = \xi_\lambda. \end{aligned}$$

Using this notation we solved the boundary value problem by a two stage shooting approach as zero-finding problem of the mapping

$$\mathbb{R}^5 \to \mathbb{R}^5, \quad \begin{pmatrix} \tau \\ \lambda_A^0 \\ x_A^1 \\ x_B^1 \\ \lambda_A^1 \end{pmatrix} \mapsto \begin{pmatrix} I(\tau, 0, 1, 0, \lambda_A^0) - (x_A^1, x_B^1, \lambda_A^1)^T \\ I_{\lambda_A}(1\,\mathrm{h}, \tau, x_A^1, x_B^1, \lambda_A^1) \\ H(1\,\mathrm{h}) - H(0) \end{pmatrix}$$

and used Newton's method to compute the solution to the root-finding problem. To initialize the root-finding algorithm, we used estimations from the computation with the direct multiple shooting method, namely $\tau \approx 0.9, x_A(\tau) \approx 0.1, x_B(\tau) \approx 0.5$. Missing are multiplier estimates λ_A^0 and λ_A^1. An estimate for $u(0) \approx 0.75$ is available from the direct method, from which it is possible to estimate $\lambda_A^0 \approx 0.57$ as $\lambda_B^0 = 1$. Finally, λ_A^1 could be easily found by backward integrating $\dot{\lambda}_A = \lambda_A(u + pu^2) - u\lambda_B$ over $[\tau, 1\,\mathrm{h}]$ with terminal value $\lambda_A = 0$, using $u \equiv 5, \lambda_B \equiv 1$. We used a coarse estimate $\lambda_A^1 = 0.15$. We used the integrator `DAESOL-II` with integration tolerance 10^{-10} to solve the initial value problems and compute derivatives. With this initial point, Newton's method with exact derivatives converged in 21 iterations with a final residual of $4 \cdot 10^{-14}$, further progress was not possible as the Jacobian in the solution point has a condition number of order 10^5 and is not desirable, as the integration result has not been computed to this accuracy. The solution point is given by

$$\begin{aligned} \tau &= 9.4995765858621 \cdot 10^{-1}, \\ \lambda_A^0 &= 5.7354506073338 \cdot 10^{-1}, \\ x_A^1 &= 7.6121655731059 \cdot 10^{-2}, \\ x_B^1 &= 5.6085564700249 \cdot 10^{-1}, \\ \lambda_A^1 &= 1.6669907171844 \cdot 10^{-1}, \end{aligned}$$

and the objective function value by 0.573545056322502 which compares well with the objective value of 0.57353 reported by Logsdon and Biegler computed using a

N	objective $x_B(1\,\mathrm{h})$	objective residual $x_B - x_B^{\mathrm{indirect}}$	# SLPECEQP	# Hv
5	0.5683867	$5 \cdot 10^{-3}$	18	151
10	0.5722421	$1 \cdot 10^{-3}$	16	190
20	0.5732976	$2 \cdot 10^{-4}$	14	207
40	0.5734790	$7 \cdot 10^{-5}$	15	295
80	0.5735282	$1 \cdot 10^{-5}$	105	1317
160	0.5735380	$7 \cdot 10^{-6}$	165	1787
320	0.5735440	$1 \cdot 10^{-6}$	40	612
550	0.5735447	$4 \cdot 10^{-7}$	42	695

Table 16.1.: Computational results for offline nonlinear batch control problem for a different number of shooting intervals N. # SLPECEQP denotes the number of SLPECEQP iterations, # Hv the number of Hessian vector products computed. Objective residual compares the objective as determined by the direct shooting approach with the objective obtained from the solution satisfying necessary conditions of Pontryagin's Maximum Principle.

direct approach with orthogonal collocation. The Hamiltonian has been confirmed to be constant along the trajectory up to a residual of $8 \cdot 10^{-9}$. Figure 16.1 shows the trajectory, costates, control and Hamiltonian along the trajectory.

Solution to the optimal control problem with shooting discretization

We used a multiple shooting discretization with piecewise constant controls along shooting intervals within the shooting framework `OptimIND` with the integrator `DAESOL-II` and solved the resulting nonlinear problem with SLPECEQP. As initial point $(x_A, x_B, u) \equiv (1.0, 0.0, 2.5)$ was used. We report computational results for a different number of equidistant shooting intervals N in Table 16.1. Figure 16.2 shows the computed control and trajectory for $N = 550$. Comparing with the solution of the indirect approach, the objective function matches and we find empirically a N^{-2} dependence for the gap between objective computed by direct shooting approach on N intervals and the objective computed using the indirect approach, compare Figure 16.3.

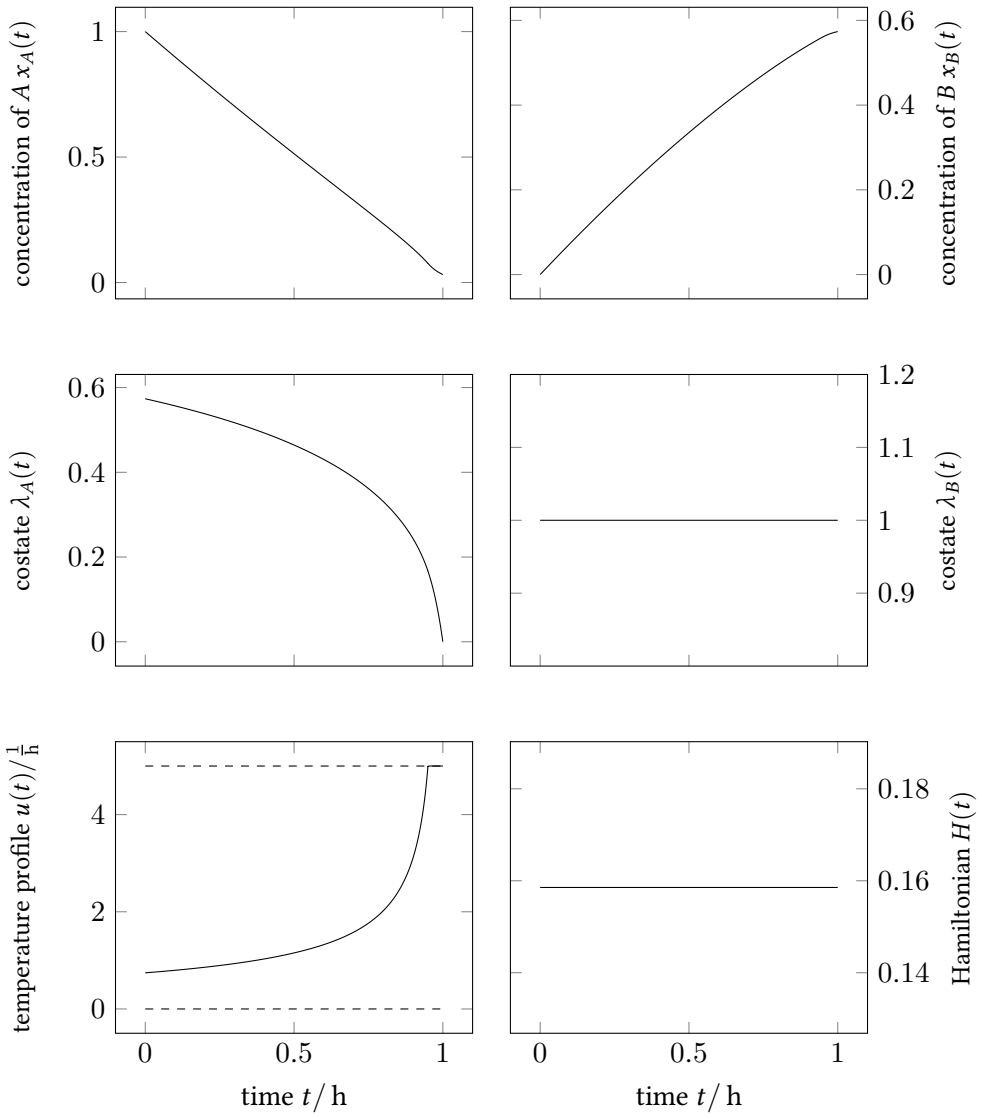


Figure 16.1.: Optimal control, trajectory, costates and Hamiltonian in offline nonlinear batch control problem.

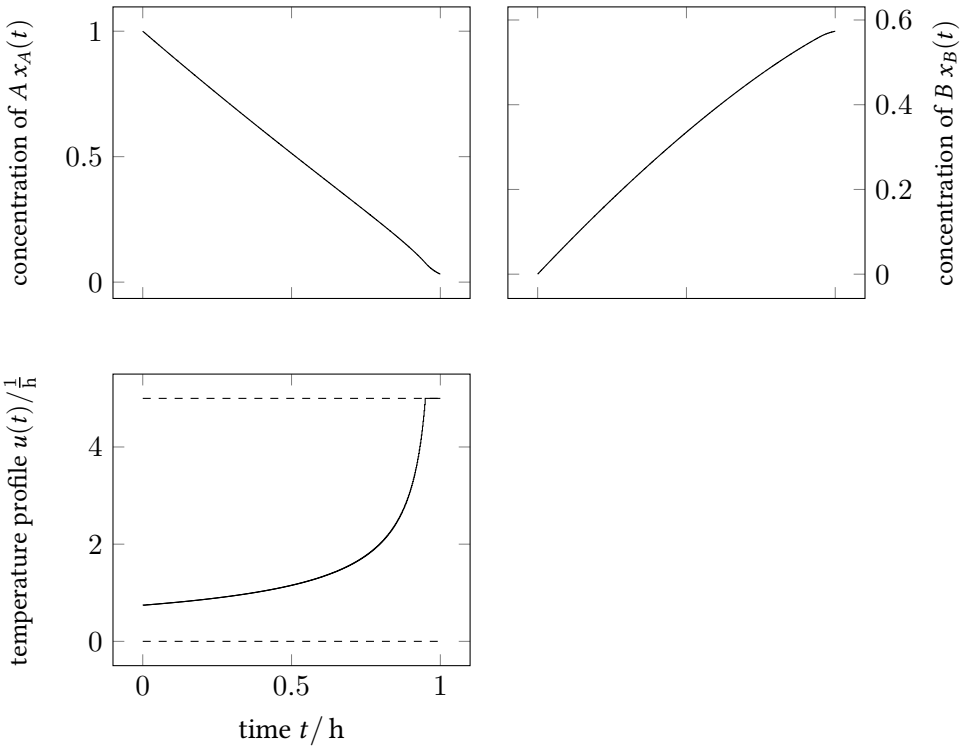


Figure 16.2.: Optimal control and trajectory in offline nonlinear batch control problem on 550 shooting intervals.

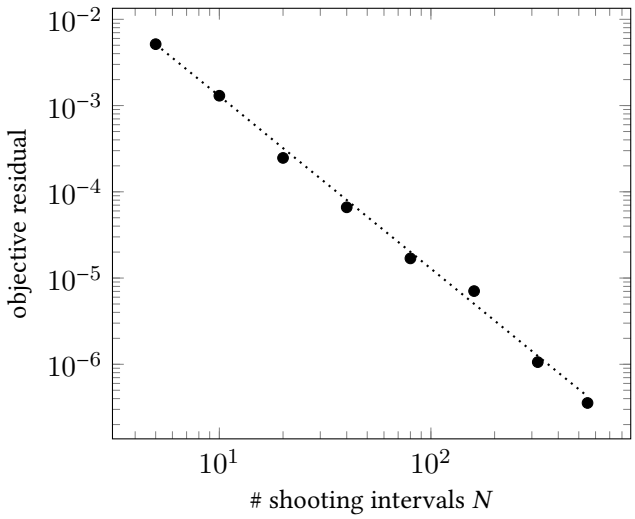


Figure 16.3.: Objective residual comparing objective as determined by the direct shooting approach with the objective obtained from the solution satisfying necessary conditions of Pontryagin's Maximum Principle in dependence on number of shooting intervals N. The dotted line shows N^{-2} dependence.

Disturbance Scenario

We consider the parameter perturbation scenario described by Kirches [Kir+12] given by

$$p = p(t) = \begin{cases} 0.5, & t < 0.5\,\mathrm{h} \text{ or } t > 0.55\,\mathrm{h}, \\ 1.2, & 0.5\,\mathrm{h} \leq t \leq 0.55\,\mathrm{h}. \end{cases}$$

This is a large perturbation of the pre-exponential factor $k_{B\to C}$ which could be interpreted as a temporary impurity in the tank modifying the reaction rate.

We compare a model predictive controller that does not assume this parameter perturbation a priori with the offline optimal control that accounts for the set-point change. For the offline optimal control problem we used the same direct shooting discretization as in the case without parameter perturbation on 500 shooting intervals and found an objective function value of 0.5688503. As reference we have computed the solution with the indirect approach again, using the same technique as for the case without disturbance. Newton's method converged within 26 iteration from the same initial point with a final residual of $5 \cdot 10^{-13}$ and a condition number of the final Jacobian of 10^5. The objective function value determined by necessary conditions from Pontryagin's Maximum Principle is 0.5688508264711951146, so there is an objective gap between shooting discretization on 500 shooting intervals and solution via indirect method of $4 \cdot 10^{-7}$. In the model predictive control scenario, we used a shrinking horizon control where 160 shooting intervals have been used for the complete time horizon. Using this controller that does not account for the parameter perturbation a priori, we find an objective function value of 0.5658556, an optimality loss of about 0.5%. The controller required an average of 1.9 Hessian matrix-vector evaluations per NMPC iteration with a maximum of 22 Hessian matrix-vector products, making it feasible for fast feedback. Kirches reports an objective function value of 0.564731 using a full SQP predictive controller with the same discretization and 0.563029 using an adjoint SQP controller, an optimality loss of about 0.7% and 1% respectively.

Computational results are shown in Figure 16.4.

16.2. Gauß-Newton Preconditioner for Stirred Tank Reactor

As second case study we consider the nonlinear process modeling a continuous stirred tank reactor. The basic model was first considered by Seborg et al. [SEM89; Seb+10, Ch. 2.4] with constant tank level and has augmented by modeling the tank level by Pannochia and Rawlings [PR03]. It is frequently used as a benchmark for model predictive control, see for example Klatt and Engell [KE93], Henson and Seborg [HS97,

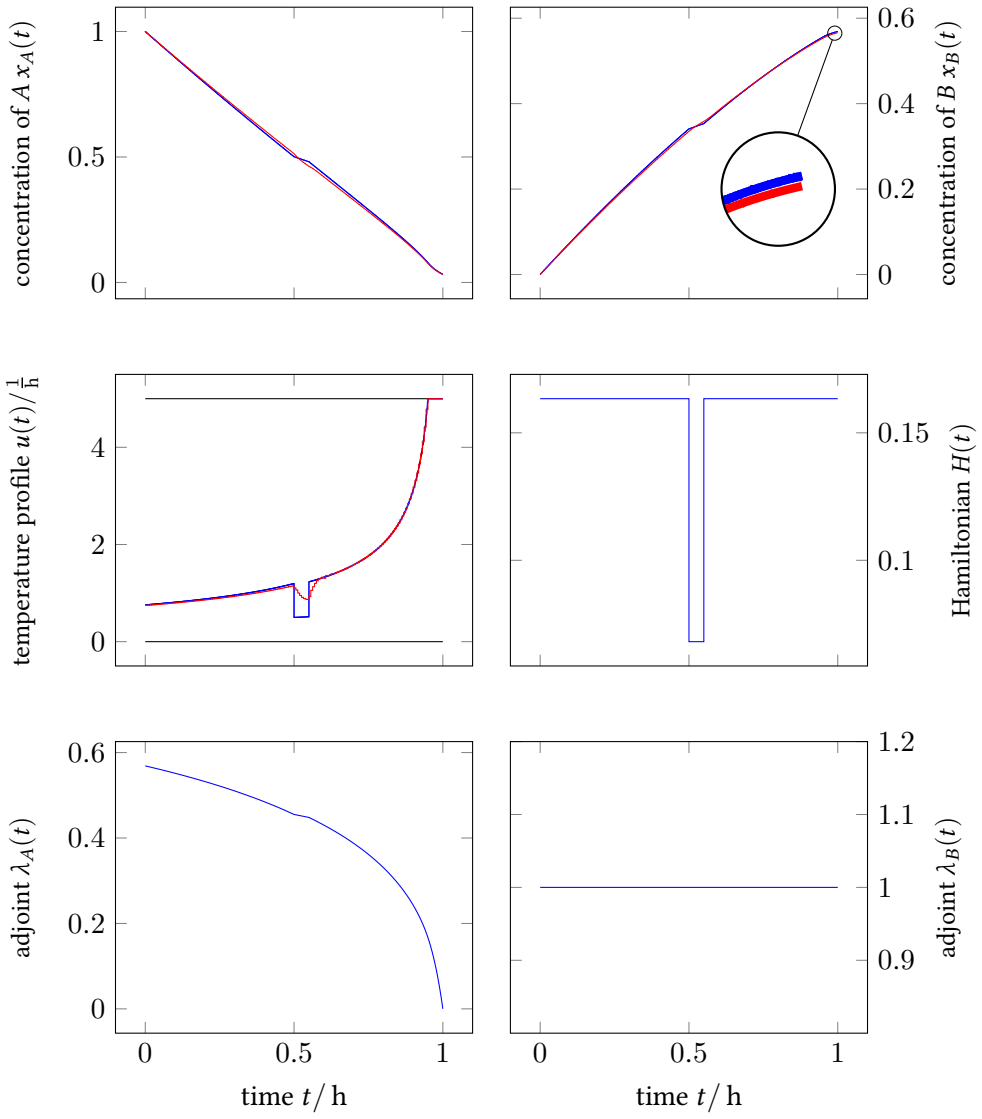


Figure 16.4.: Optimal control and trajectory in parameter disturbance scenario for $0.5 \leq t \leq 0.55$. In blue the offline control and trajectory is shown, both the one computed by the shooting method on 500 intervals and the one computed using Pontryagin's Maximum Principle. They cannot be distinguished in plotting resolution. In red is the answer of the model predictive shrinking horizon controller on 160 shooting intervals that does not anticipate the disturbance, leading to a suboptimality in yield of x_B at the final time of only 0.5%. The Hamiltonian is only piecewise constant as the differential right hand side is discontinuous.

Symbol	Description	Value
h	tank level	
c_A	molar concentration of A	
T	reactor temperature	
F	outlet flow rate	
T_c	coolant liquid temperature	
T_0	inflow temperature	$3.5 \cdot 10^2\,\mathrm{K}$
r	tank radius	$2.19 \cdot 10^{-1}\,\mathrm{m}$
F_0	inlet flow rate	$10^{-1}\,\frac{\mathrm{m}^3}{\mathrm{min}}$
c_0	inlet molar concentration of A	$10^3\,\frac{\mathrm{mol}}{\mathrm{m}^3}$
k_0	pre-exponential factor in Arrhenius law	$7.2 \cdot 10^{10}\,\frac{1}{\mathrm{min}}$
$\beta = E/R$	E activation energy, R universal gas constant	$8.75 \cdot 10^3\,\mathrm{K}$
U	heat transfer coefficient	$5.4936 \cdot 10^4\,\frac{\mathrm{J}}{\mathrm{min\,m^2\,K}}$
ρ	mass density of feed and product stream	$10^3\,\frac{\mathrm{kg}}{\mathrm{m}^3}$
C_p	constant pressure heat capacity	$2.39 \cdot 10^2\,\frac{\mathrm{J}}{\mathrm{kg\,K}}$
ΔH	heat of reaction	$-5 \cdot 10^4\,\frac{\mathrm{J}}{\mathrm{mol}}$

Table 16.2.: Definition of quantities in stirred tank reactor dynamics

Ch. 1] and Diehl [Die01]. We follow Kirches et al. [Kir+12] in choice of parameters and scenarios.

In the liquid phase of the tank reactor, a irreversible exothermic reaction $A \longrightarrow B$ takes place in the liquid phase of the reactor which is subject to external cooling. Formulating mass and energy balances leads to the following dynamic description:

$$\begin{aligned}
\dot{h} &= \tfrac{1}{\pi r^2}(F_0 - F),\\
\dot{c}_A &= \tfrac{F_0}{\pi r^2 h}(c_0 - c_A) - k_0 c_A \exp(-\tfrac{\beta}{T}),\\
\dot{T} &= \tfrac{F_0}{\pi r^2 h}(T_0 - T) + \tfrac{-\Delta H}{\rho C_p} k_0 c_A \exp(-\tfrac{\beta}{T}) + \tfrac{2U}{r\rho C_p}(T_c - T).
\end{aligned}$$

Set-Point and Steady State

We consider a scenario where it is desired to run the batch reactor in a set-point given by $h^{\mathrm{set}} = 6.59 \cdot 10^{-1}\,\mathrm{m}$, $c_A^{\mathrm{set}} = 8.77 \cdot 10^2\,\frac{\mathrm{mol}}{\mathrm{m}^3}$, $T^{\mathrm{set}} = 3.245 \cdot 10^2\,\mathrm{K}$, $F^{\mathrm{set}} = 10^{-1}\,\frac{\mathrm{m}^3}{\mathrm{min}}$, $T_c^{\mathrm{set}} = 3 \cdot 10^2\,\mathrm{K}$. This set-point is in a neighborhood of a steady state of the system if the controls F and T_c are at their respective set-points. Using Newton's method

we computed the steady state for $F = F^{\text{set}}$, $T = T_c^{\text{set}}$ and found $h^{\text{steady}} = 6.59 \cdot 10^{-1}\,\text{m}$, $c_A^{\text{steady}} = 8.77798024197374729738 \cdot 10^2\,\frac{\text{mol}}{\text{m}^3}$, $T^{\text{steady}} = 3.24499656537196187855 \cdot 10^2\,\text{K}$, satisfying $\|(\dot{h}, \dot{c}_A, \dot{T})^T\| \leq 10^{-12}$.

16.2.1. Offline scenario with inlet molar concentration disturbance

As an optimal control test problem, we consider a scenario that involves a known disturbance of the inlet molar concentration, so that the data c_0 now has to be considered as a function of time:

$$c_0(t) = \begin{cases} 10^3\,\frac{\text{mol}}{\text{m}^3}, & t < 9\,\text{min} \\ 1.05 \cdot 10^3\,\frac{\text{mol}}{\text{m}^3}, & t \geq 9\,\text{min}. \end{cases}$$

The process is started in the set-point and the objective aims at steering tank level and concentration into the set-point as well as it violates too large deviations of the outlet flow rate and coolant liquid temperate from the control set-points.

The problem under consideration is thus

$$\begin{aligned} \min_{h, c_A, T, F, T_c} \quad & \int_0^{50\,\text{min}} \gamma_h (h(t) - h^{\text{set}})^2 + \gamma_{c_A} (c_A(t) - c_A^{\text{set}})^2 \\ & \qquad \gamma_F (F(t) - F^{\text{set}})^2 + \gamma_{T_c} (T_c(t) - T_c^{\text{set}})^2 \; \mathrm{d}t \\ \text{s.t.} \quad & \dot{h} = \tfrac{1}{\pi r^2}(F_0 - F), \\ & \dot{c}_A = \tfrac{F_0}{\pi r^2 h}(c_0 - c_A) - k_0 c_A \exp(-\tfrac{\beta}{T}), \\ & \dot{T} = \tfrac{F_0}{\pi r^2 h}(T_0 - T) + \tfrac{-\Delta H}{\rho C_p} k_0 c_A \exp(-\tfrac{\beta}{T}) + \tfrac{2U}{r \rho C_p}(T_c - T), \\ & h(0) = h^{\text{set}}, \quad c_A(0) = c_A^{\text{set}}, \quad T(0) = T^{\text{set}}, \\ & F \in [0.085\,\tfrac{\text{m}^3}{\text{min}}, 0.115\,\tfrac{\text{m}^3}{\text{min}}], \quad T_c \in [299\,\text{K}, 301\,\text{K}]. \end{aligned}$$

Tracking weights used are $\gamma_h = 1\,\frac{1}{\text{m}^2}, \gamma_{c_A} = 10^{-4}\,\frac{\text{m}^6}{\text{mol}^2}, \gamma_F = 10^5\,\frac{\text{min}^2}{\text{m}^6}, \gamma_{T_c} = 10^{-1}\,\frac{1}{\text{K}^2}$.

We used a multiple shooting discretization on 50 shooting intervals with the integrator `RKFSWT` within the shooting framework `OptimIND` and solved the resulting nonlinear program with SLPECEQP. We used a least squares approximation of the objective function on shooting nodes and a Gauß-Newton approximation to the objective Hessian.

We have considered solving the nonlinear program with and without variable scaling and with and without Gauß-Newton preconditioning. Scaling factors have been chosen as $s_h = 1, s_{c_A} = 10^3, s_T = 10^2, s_F = 10^{-1}, s_{T_c} = 10^2$. Computational results are shown in Table 16.3, trajectories and controls in Figure 16.5. Results clearly demonstrate the effectiveness of the Gauß-Newton preconditioner and also the lack

of affine contravariance of the SLPECEQP algorithm. In the case without variable scaling, the lack of variable scaling can be also understood as a bad preconditioner so that the effect of bad variable scaling is twofold. First, it affects the nonlinear programming solver that is not affine contravariant due to penalty function and trust region globalization, which can be seen by the fact that the maximum penalty parameter has been increased to 10^4 from its initial value 10. Second, it acts as a bad preconditioner in the CG method for the trust region subproblem solver, forcing a high number of CG iterations with too early termination for good nonlinear progress, as the convergence criterion of the subproblem solver depends on the preconditioner. All variants terminate in the same point with identical function value satisfying KKT conditions with a residual $\leq 10^{-8}$.

Offline scenario comparison with indirect approach

For comparison purposes we have computed the offline solution again using the indirect approach. The boundary value problem resulting from Pontryagin's Maximum Principle is given in this case by

$$
\begin{aligned}
F &= \begin{cases} F^s + \frac{\lambda_h}{2w_F \pi r^2}, & \left|\frac{\lambda_h}{2w_F \pi r^2}\right| \leq 0.15 \, \frac{\mathrm{m}^3}{\mathrm{min}} \\ 0.085 \, \frac{\mathrm{m}^3}{\mathrm{min}} \text{ or } 0.115 \, \frac{\mathrm{m}^3}{\mathrm{min}}, & \text{else,} \end{cases}, \\
T_c &= \begin{cases} T_c^s - \frac{\lambda_T U}{w_{T_c} r \rho C_p}, & \left|\frac{\lambda_T U}{w_{T_c} r \rho C_p}\right| \leq 1 \, \mathrm{K} \\ 299 \, \mathrm{K} \text{ or } 301 \, \mathrm{K}, & \text{else,} \end{cases}, \\
\dot{h} &= \tfrac{1}{\pi r^2}(F_0 - F), \\
\dot{c}_A &= \tfrac{F_0}{\pi r^2 h}(c_0 - c_A) - k_0 c_A \exp(-\tfrac{\beta}{T}), \\
\dot{T} &= \tfrac{F_0}{\pi r^2 h}(T_0 - T) + \tfrac{-\Delta H}{\rho C_p} k_0 c_A \exp(-\tfrac{\beta}{T}) + \tfrac{2U}{r \rho C_p}(T_c - T), \\
\dot{\lambda}_h &= -2w_h(h - h^s) + \tfrac{F_0}{\pi r^2 h^2}\left(\lambda_c(c_0 - c) + \lambda_T(T_0 - T)\right), \\
\dot{\lambda}_{c_A} &= -2w_{c_A}(c_A - c^s) + \lambda_{c_A}\left(\tfrac{F_0}{\pi r^2 h} + k_0 \exp(-\tfrac{\beta}{T})\right) + \lambda_T \tfrac{\Delta H}{\rho C_p} k_0 \exp(-\tfrac{\beta}{T}), \\
\dot{\lambda}_T &= \lambda_{c_A} \tfrac{k_0 \beta c_A}{T^2} \exp(-\tfrac{\beta}{T}) + \lambda_T\left(\tfrac{F_0}{\pi r^2 h} + \tfrac{\Delta H k_0 \beta c_A}{\rho C_p T^2} \exp(-\tfrac{\beta}{T}) + \tfrac{2U}{r \rho C_p}\right), \\
h(0) &= h^{\mathrm{set}}, \\
c_A(0) &= c_A^{\mathrm{set}}, \\
T(0) &= T^{\mathrm{set}}, \\
\lambda(50 \, \mathrm{min}) &= (0, 0, 0)^T.
\end{aligned}
$$

We have solved the boundary value problem using a multiple shooting method with shooting nodes at times 0, 5, 9, 10, 15, 20, 25, 30, 35, 40, 45, 50 (in minutes) and

N	Scaling	PC	# SLPECEQP	# Hv	Objective		max. penalty
					NLP	OCP	
17	Yes	GN	7	58	0.6415932	1.0054737	10^2
17	Yes	I	8	202	0.6415932	1.0054737	10^2
17	No	GN	8	73	0.6415932	1.0054737	10^2
17	No	I	25	1070	0.6415932	1.0054737	10^2
34	Yes	GN	8	66	0.7603741	0.9625429	10^2
34	Yes	I	7	224	0.7602741	0.9625429	10^2
34	No	GN	14	143	0.7602741	0.9625429	10^2
34	No	I	50	2681	0.7602741	0.9625429	10^3
68	Yes	GN	6	49	0.8548115	0.9340243	10^2
68	Yes	I	6	223	0.8584115	0.9340243	10^2
68	No	GN	10	98	0.8548115	0.9340243	10^2
68	No	I	94	1936	0.8548115	0.9340243	10^3
137	Yes	GN	7	49	0.8985745	0.9045564	10^2
137	Yes	I	5	223	0.8985745	0.9045564	10^2
137	No	GN	187	1739	0.8985745	0.9045564	10^2
137	No	I	9	1073	0.8985745	0.9045564	10^2
275	Yes	GN	7	49	0.8978458	0.9046565	10^2
275	Yes	I	46	719	0.8978458	0.9046565	10^2
275	No	GN	214	1749	0.8978458	0.9046565	10^3
275	No	I	289	4675	0.8978458	0.9046565	10^4
550	Yes	GN	77	623	0.9011404	0.9008656	10^2
550	Yes	I	15	334	0.9011404	0.9008656	10^3
550	No	GN	> 500				
550	No	I	failure				

Table 16.3.: Computational results with different variants of the nonlinear program. N is the number of shooting intervals. PC denotes the used preconditioner with I being identity and GN being the Gauß-Newton preconditioner. # SLPECEQP denotes the number of SLPECEQP iterations, # Hv the number of Hessian vector product evaluations. NLP objective denotes the approximation of the least-squares objective using a trapezoidal rule in shooting nodes, OCP objective the least-squares integral as defined for the optimal control problem obtained by a posteriori integration. Failure occurred during integration if the maximum number of allowed integrator steps was reached.
Using Gauß-Newton preconditioning and proper scaling yields a nearly discretization independent behavior.

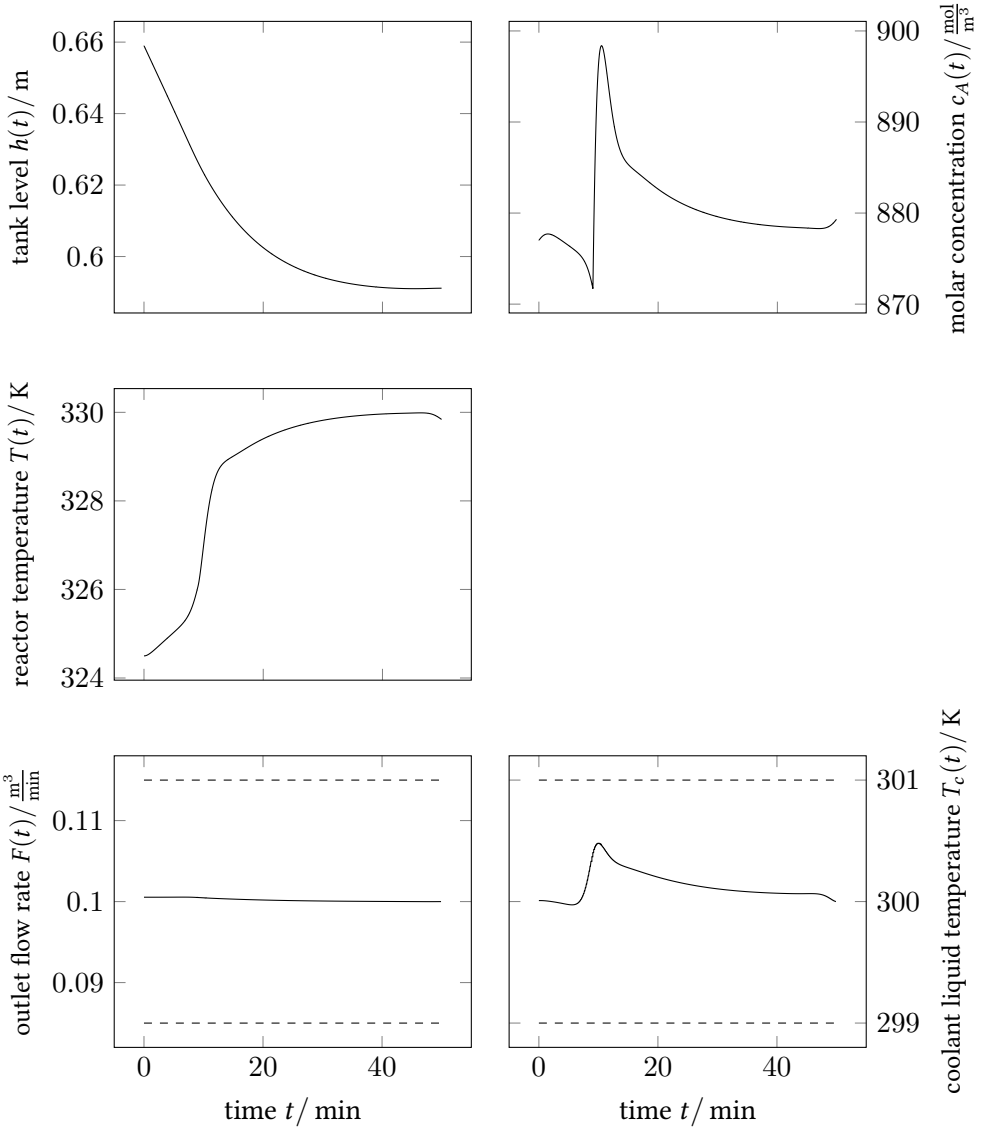


Figure 16.5.: Optimal control and trajectory in offline disturbance scenario with inlet molar concentration disturbance at $t = 9$ min on 320 shooting intervals.

used Newton's method to solve the resulting root-finding problem. As initial point we used the values of h, c_A, T at the shooting nodes obtained from the solution computed with the direct shooting method and $\lambda_h = 2w_F\pi r^2(F - F^s), \lambda_{c_A} = 0, \lambda_{T_c} = -w_{T_c}\frac{r\rho C_p}{U}(T_c - T_c^s)$ with F, T_c as controls obtained from the solution computed with the direct method. Newton's method converged with a final residual of $1.0 \cdot 10^{-9}$ and condition of the Jacobian of $3.2 \cdot 10^8$ and yields an objective function value of 0.9008569215. The computed trajectory, costates, controls and Hamiltonian along the trajectory are shown in Figure 16.6. A comparison of the optimal control objective between the solutions computed with the direct approach using an trapezoidal rule for objective approximation in shooting nodes and objective function value obtained from indirect approach are shown in Figure 16.7.

Using the starting point given by the steady state or the set point and the same costate initialization strategy does not allow integration of the trajectories. Without the introduction of a finer shooting grid and further advanced globalization strategies, knowledge of the solution obtained by the direct method is thus crucial for initialization of the indirect method. Without this knowledge, it would have not been possible using, the described grid of 21 shooting nodes to compute the solution satisfying the necessary conditions of Pontryagin's Maximum Principle.

Online scenario with inlet flow rate disturbance

As second disturbance scenario we consider a set-point change in inlet flow rate given by

$$F(t) = \begin{cases} 0.1\ \frac{\mathrm{m}^3}{\mathrm{min}}, & t < 5\,\mathrm{min} \\ 0.11\ \frac{\mathrm{m}^3}{\mathrm{min}}, & t \geq 5\,\mathrm{min}. \end{cases}$$

We follow [Kir+12] in choice of tracking weights $\gamma_h = 1\ \frac{1}{\mathrm{m}^2}, \gamma_{c_A} = 10^{-4}\ \frac{\mathrm{m}^6}{\mathrm{mol}^2}, \gamma_F = 10^{-8}\ \frac{\mathrm{min}^2}{\mathrm{m}^6}, \gamma_{T_c} = 10^{-4}\ \frac{1}{\mathrm{K}^2}$ as the tracking weights used in the offline scenario enforce a control regularization that is too strong.

We use a moving-horizon nonlinear model predictive controller with prediction horizon of 5 min that is initialized in the steady state and consider the controller reaction on the time horizon $[0, 10\,\mathrm{min}]$. Figure 16.8 shows the reaction of the SLPECEQP-NMPC controller to the disturbance scenario. Using the Gauß-Newton preconditioner, an average of 2.9 Hessian-vector products is computed per NMPC iteration with a maximum of 23 Hessian-vector products which constitute the main computation cost during the NMPC feedback phase. An objective function value $\leq 10^{-2}$ is reached after 55 s after the disturbance happened which demonstrates the contractivity of the scheme in that case. Using the identity instead of the Gauß-Newton preconditioner, an average of 43 Hessian-vector products are required per NMPC iteration with a

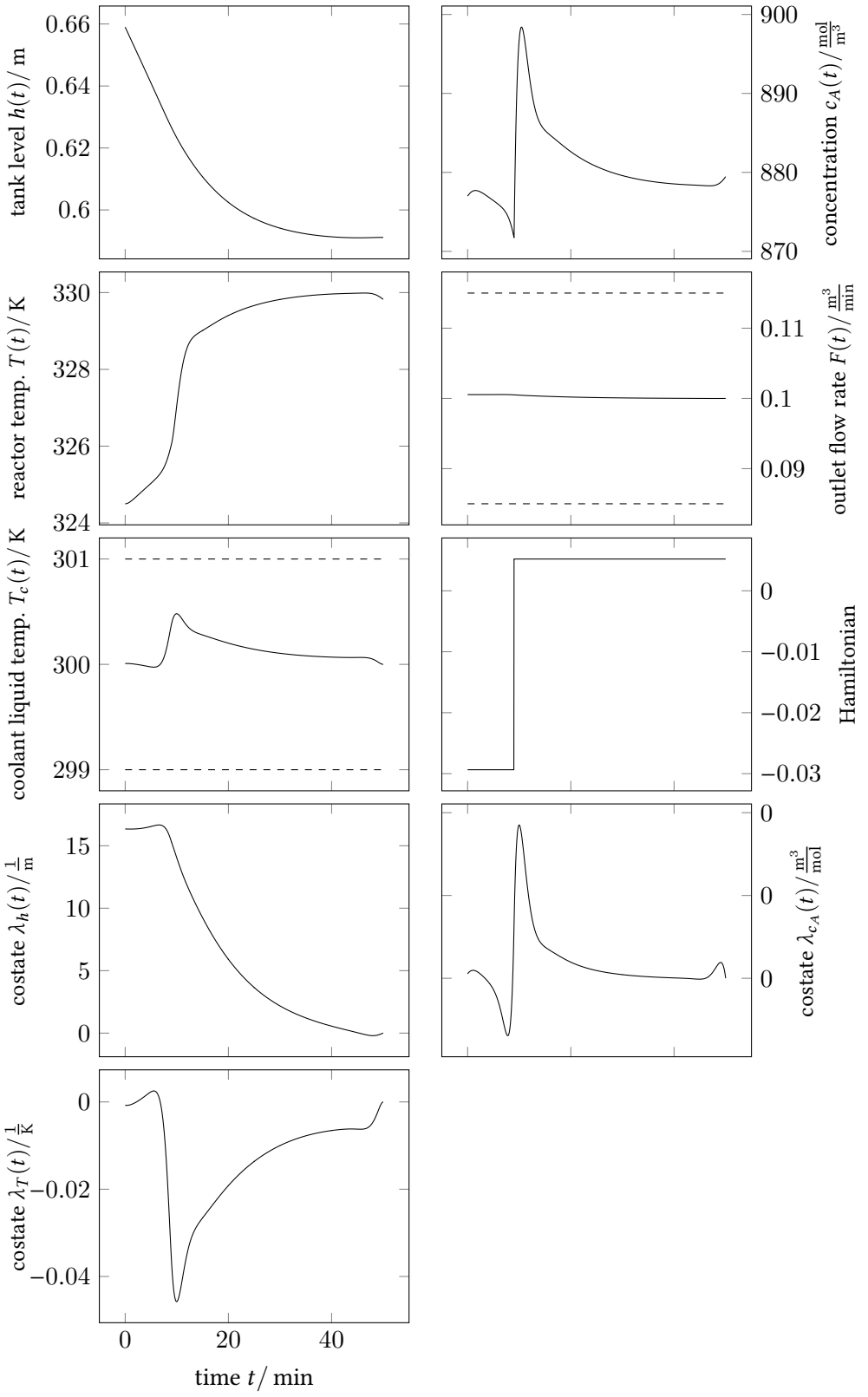


Figure 16.6.: Optimal control and trajectory in offline disturbance scenario with inlet molar concentration disturbance at $t = 9\,\text{min}$ computed by indirect multiple shooting approach.

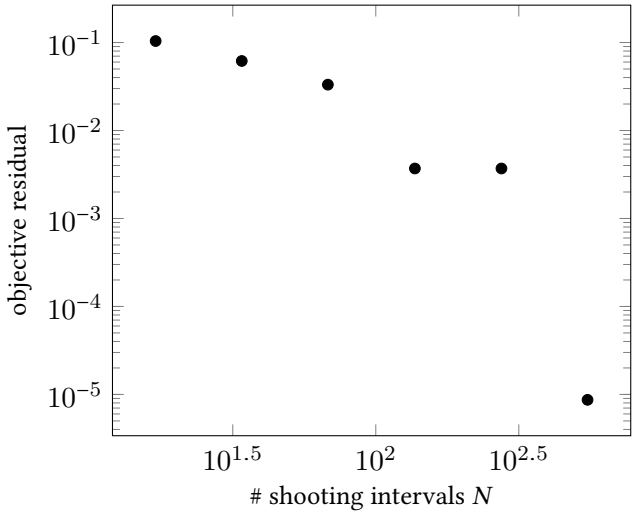


Figure 16.7.: Objective residual comparing optimal control objective as determined by the direct shooting approach with the objective obtained from the solution satisfying necessary conditions of Pontryagin's Maximum Principle in dependence on number of shooting intervals N.

maximum of 233 products in one iteration, which severely degrades the applicability of the scheme. The preconditioner is thus essential to guarantee fast feedback in a real-time application setting.

16.3. Summary

In this chapter, we studied a nonlinear batch reactor and a continuous stirred tank reactor as benchmark problems for nonlinear model predictive control. We found that a real-time iteration scheme based on the SLPECEQP algorithm shows satisfactory performance and demonstrated the effectiveness of the Gauß-Newton preconditioner in the example of the continuous stirred tank reactor. As a reference we computed solutions to the offline problems that satisfy the necessary conditions of Pontryagin's Maximum Principle.

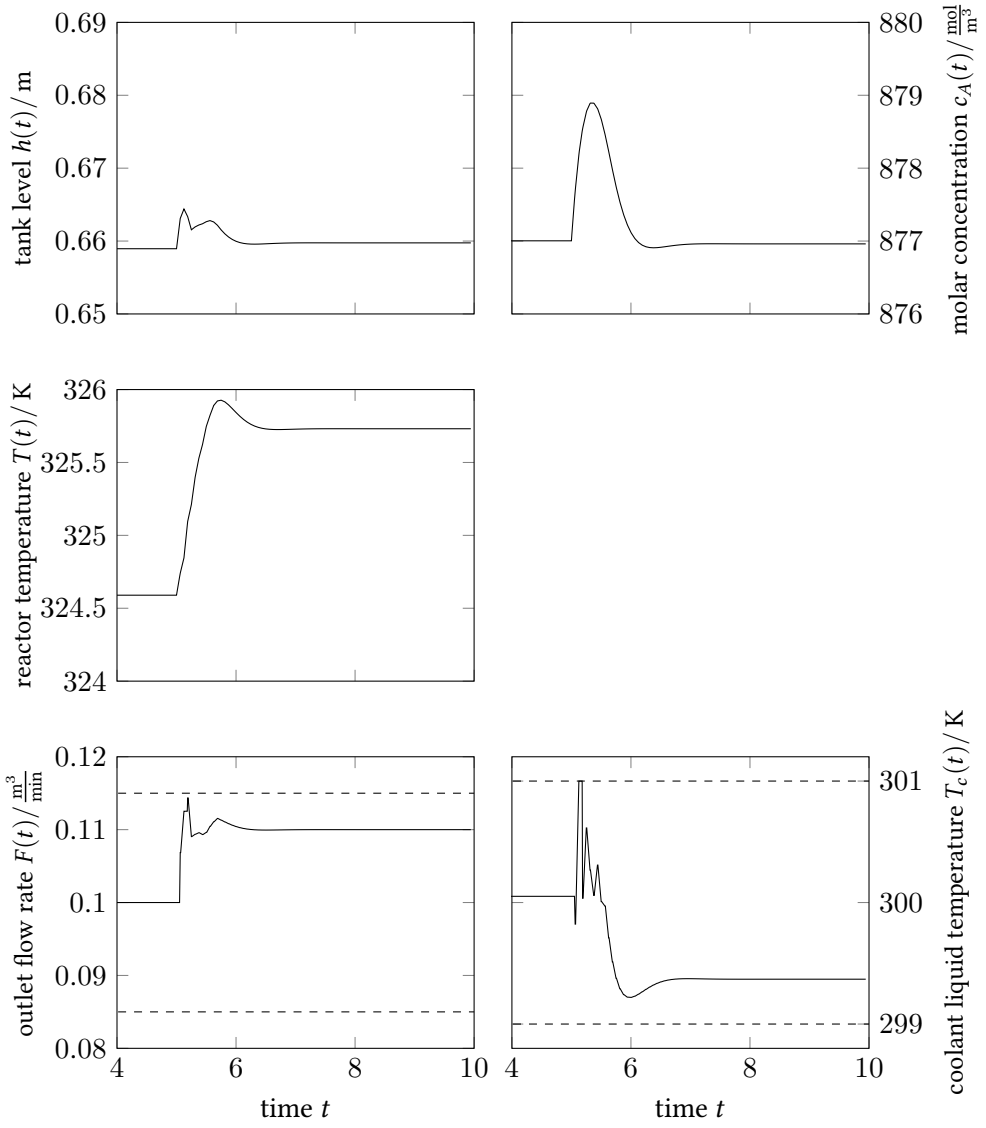


Figure 16.8.: Optimal control and trajectory in model predictive control disturbance scenario with inlet flow rate disturbance at $t = 5\,\text{min}$.

17. Conclusion and Outlook

In this thesis we contributed to numerical methods for Mixed-Integer Optimal Problems with Combinatorial Constraints. Our results generalize findings of Sager [Sag06; SBD12] by allowing combinatorial constraints that depend on integer control variables.

We considered a reformulation based on partial outer convexification and relaxation and established an approximation result connecting these problems. We found that suboptimal solutions to a Mixed-Integer Optimal Control Problem with arbitrary small infeasibility and optimality loss can be computed by solving the continuous relaxation of the partial outer convexification of the problem. With VC-SOS-SUR we introduced a new rounding scheme that computationally exploits this approximation property.

Direct discretizations of the relaxed convexified problem mandate the solution of non-smooth and non-convex Mathematical Programs with Vanishing Constraints. We established a Sequential LPEC EQP method for Mathematical Programs with Vanishing Constraints that extends the SLEQP algorithm for nonlinear programming of Nocedal and Waltz [Byr+05]. We proved global convergence of the method to Bouligand stationary points. Fast convergence of the algorithm is promoted via Newton-type steps computed from EQP trust region subproblems.

We generalized Gould's Generalized Lanczos method for the trust region subproblem to a Hilbert space setting. The Hilbert space setting covers the application in the SLEQP algorithm as well as, for example, applications from PDE constrained optimal control. To achieve real-time feasibility in an online optimal control context, we developed a Gauß-Newton preconditioner for effective iterative solution of the trust region subproblem.

We implemented the proposed methods and demonstrated the applicability and efficiency on a set of benchmark problems and found performance competitive with state-of-the art solvers.

Our research can be extended in several directions.

We have developed our method for Mathematical Programs with Vanishing Constraints for the discretization of an Optimal Control Problem with Vanishing Constraints. It is appealing to establish and study the method in a function space context.

While we have established partial results concerning the trust-region part of the algorithm, two obstacles have to be addressed. First, algorithms for LPEC have to be generalized into a function space context. This requires establishing a suitable theoretical framework and defining corresponding notions from the finite-dimensional case, compare Anderson and Nash [AN87] for partial results concerning the LP case. Second, our convergence proof relied on norm-equivalence in finite-dimensional space. Care must be exercised in defining a function space framework that ensures that the embedding into the space defining the linear model is continuous.

Furthermore, an interesting algorithmic enhancement can be obtained by considering a SLPECEQP algorithm with Newton lifting for problems with many intermediate variables. The lifted Newton approach [AD10] has been proven to be advantageous for optimization problem with a tree-structure of intermediate variables. Such a structure is present in many real-world problems and in particular in a natural way in problems arising as multiple shooting discretizations of optimal control problems. Efficient exploitation of the structure of lifted Newton problems is possible at both the level of the LPEC part and the EQP part of the algorithm.

Considering the Generalized Lanczos method used to solve the trust region subproblems, we have added a convexification heuristic to handle ill-posed problems. However, it is not necessary to use conjugate gradient or Lanczos iterations to build Krylov subspaces. Using the minimal residual method [PS75] for Krylov subspace generation provides a method that we conjecture to be robust on ill-posed problems without additional convexification. On the same time, the increase in computational effort promises to be modest.

Having established these methodological and algorithmic extensions and generalizations, it is interesting to apply the proposed methods to further real-world examples. Especially via MPEC reformulation of bi-level Optimal Control Problems, our methods are applicable to a wide range of interesting problems as parameter estimation in human gait analysis [Hat14] or robust model predictive control [DBK06; Die+08; HD13].

Bibliography

[AÅD12] J. Andersson, J. Åkesson, and M. Diehl. "CasADi: A Symbolic Package for Automatic Differentiation and Optimal Control." In: *Recent Advances in Algorithmic Differentiation*. Ed. by S. Forth et al. Lecture Notes in Computational Science and Engineering. Berlin: Springer, 2012, pp. 297–307. DOI: 10.1007/978-3-642-30023-3_2.

[Aba67] J. Abadie. "On the Kuhn–Tucker theorem." In: *Nonlinear Programming*. Ed. by J. Abadie and S. Vajda. New York, NY: John Wiley & Sons, Inc., 1967, pp. 21–36.

[ABG07] P.-A. Absil, C. Baker, and K. Gallivan. "Trust-Region Methods on Riemannian Manifolds." In: *Foundations of Computational Mathematics* 7.3 (July 2007), pp. 303–330. ISSN: 1615-3383. DOI: 10.1007/s10208-005-0179-9.

[Ach07] T. Achterberg. "Constraint Integer Programming." Dissertation. Technical University of Berlin, 2007. DOI: 10.14279/depositonce-1634.

[ACR79] U. Ascher, J. Christiansen, and R. Russell. "A collocation solver for mixed order systems of boundary value problems." In: *Mathematics of Computation* 33 (1979), pp. 659–679. DOI: 10.1090/S0025-5718-1979-0521281-7.

[AD10] J. Albersmeyer and M. Diehl. "The Lifted Newton Method and its Application in Optimization." In: *SIAM Journal on Optimization* 20.3 (2010), pp. 1655–1684. DOI: 10.1137/080724885.

[Ada+17] S. Adachi, S. Iwata, Y. Nakatsukasa, and A. Takeda. "Solving the Trust-Region Subproblem By a Generalized Eigenvalue Problem." In: *SIAM Journal on Optimization* 27.1 (2017), pp. 269–291. DOI: 10.1137/16M1058200.

[AF03] R. Adams and J. Fournier. *Sobolev Spaces*. Second Edition. Vol. 140. Pure and Applied Mathematics. Elsevier/Academic Press, Amsterdam, 2003.

[AK08] W. Achtziger and C. Kanzow. "Mathematical programs with vanishing constraints: optimality conditions and constraint qualifications." In: *Mathematical Programming, Series A* 114 (2008), pp. 69–99. DOI: 10.1007/s10107-006-0083-3.

[Ala40] L. Alaoglu. "Weak Topologies of Normed Linear Spaces." In: *Annals of Mathematics* 41.1 (1940), pp. 252–267. ISSN: 0003486X.

[Alb10] J. Albersmeyer. "Adjoint based algorithms and numerical methods for sensitivity generation and optimization of large scale dynamic systems." Dissertation. Heidelberg University, 2010. DOI: 10.11588/heidok.00011651.

[Aln+14] M. S. Alnæs et al. "Unified Form Language: A Domain-specific Language for Weak Formulations of Partial Differential Equations." In: *ACM Transactions on Mathematical Software* 40.2 (2014), 9:1–9:37. ISSN: 0098-3500. DOI: 10.1145/2566630.

[Aln+15] M. Alnæs et al. "The FEniCS Project Version 1.5." In: *Archive of Numerical Software* 3.100 (2015). ISSN: 2197-8263. DOI: 10.11588/ans.2015.100.20553.

[AMR88] U. Ascher, R. Mattheij, and R. Russell. *Numerical Solution of Boundary Value Problems for Differential Equations*. Engelwood Cliffs, NJ: Prentice Hall, 1988.

[AN87] E. Anderson and P. Nash. *Linear Programming in Infinite-Dimensional Spaces: Theory and Applications*. Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley, 1987. ISBN: 9780608009926.

[Ani05a] M. Anitescu. "Global Convergence of an Elastic Mode Approach for a Class of Mathematical Programs with Complementarity Constraints." In: *SIAM Journal on Optimization* 16.1 (2005), pp. 120–145. DOI: 10.1137/040606855.

[Ani05b] M. Anitescu. "On Using the Elastic Mode in Nonlinear Programming Approaches to Mathematical Programs with Complementarity Constraints." In: *SIAM Journal on Optimization* 15.4 (2005), pp. 1203–1236. DOI: 10.1137/S1052623402401221.

[ATW07] M. Anitescu, P. Tseng, and S. Wright. "Elastic-mode algorithms for mathematical programs with equilibrium constraints: global convergence and stationarity properties." In: *Mathematical Programming, Series A* 110 (2007), pp. 337–371. DOI: 10.1007/s10107-006-0005-4.

[Ban32] S. Banach. *Théorie des opérations linéaires*. fre. Warszawa: Instytut Matematyczny Polskiej Akademi Nauk, 1932. URL: http://eudml.org/doc/268537.

[Bär83] V. Bär. "Ein Kollokationsverfahren zur numerischen Lösung allgemeiner Mehrpunktrandwertaufgaben mit Schalt– und Sprungbedingungen mit Anwendungen in der optimalen Steuerung und der Parameteridentifizierung." Diplomarbeit. Rheinische Friedrich–Wilhelms–Universität zu

Bonn, 1983. URL: https://bonnus.ulb.uni-bonn.de/SummonRecord/FETCH-bonn_catalog_21300842.

[Bar88] J. F. Bard. "Convex two-level optimization." In: *Mathematical Programming* 40.1 (Jan. 1988), pp. 15–27. ISSN: 1436-4646. DOI: 10.1007/BF01580720.

[BB09] B. Baumrucker and L. Biegler. "MPEC strategies for optimization of a class of hybrid dynamic systems." In: *Journal of Process Control* 19.8 (2009). Special Section on Hybrid Systems: Modeling, Simulation and Optimization, pp. 1248–1256. ISSN: 0959-1524. DOI: 10.1016/j.jprocont.2009.02.006.

[BCG93] C. Bischof, G. Corliss, and A. Griewank. "Structured Second- and Higher-Order Derivatives Through Univariate Taylor Series." In: *Optimization Methods and Software* (1993), pp. 211–232. DOI: 10.1080/10556789308805543.

[Beh+11] S. Behnel et al. "Cython: The Best of Both Worlds." In: *Computing in Science Engineering* 13.2 (2011), pp. 31–39. ISSN: 1521-9615. DOI: 10.1109/MCSE.2010.118.

[Bel+13] P. Belotti et al. "Mixed-Integer Nonlinear Optimization." In: *Acta Numerica*. Ed. by A. Iserles. Vol. 22. Cambridge University Press, 2013, pp. 1–131. DOI: 10.1017/S0962492913000032.

[Bel57] R. Bellman. *Dynamic Programming*. 6th. ISBN 0-486-42809-5 (paperback). Princeton, N.J.: University Press, 1957.

[Ben+06] H. Y. . Benson, A. Sen, D. F. Shanno, and R. J. Vanderbei. "Interior-Point Algorithms, Penalty Methods and Equilibrium Problems." In: *Computational Optimization and Applications* 34.2 (June 2006), pp. 155–182. ISSN: 1573-2894. DOI: 10.1007/s10589-005-3908-8.

[Ben01] H. Benson. *CUTEr AMPL models*. 2001. URL: http://orfe.princeton.edu/~rvdb/ampl/nlmodels/cute/.

[Ber05] D. Bertsekas. *Dynamic programming and optimal control, Volume 1*. eng. 3. ed. Belmont, Mass.: Athena Scientific, 2005, XV, 543 S. ISBN: 1-886529-26-4 ; 978-1-886529-26-7.

[Ber74] L. Berkovitz. *Optimal Control Theory*. Vol. 12. Applied Mathematical Sciences. New York: Springer-Verlag, 1974.

[BG16] M. Benko and H. Gfrerer. "An SQP method for mathematical programs with complementarity constraints with strong convergence properties." eng. In: *Kybernetika* 52.2 (2016), pp. 169–208. DOI: 10.14736/kyb-2016-2-0169.

[Bie84] L. Biegler. "Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation." In: *Computers & Chemical Engineering* 8 (1984), pp. 243–248. DOI: 10.1016/0098-1354(84)87012-X.

[Bin+01] T. Binder et al. "Introduction to Model Based Optimization of Chemical Processes on Moving Horizons." In: *Online Optimization of Large Scale Systems: State of the Art*. Ed. by M. Grötschel, S. Krumke, and J. Rambau. Springer, 2001, pp. 295–340. DOI: 10.1007/978-3-662-04331-8_18.

[BKS00] H. Bock, E. Kostina, and J. Schlöder. "On the Role of Natural Level Functions to Achieve Global Convergence for Damped Newton Methods." In: *System Modelling and Optimization. Methods, Theory and Applications*. Ed. by M. Powell and S. Scholtes. Kluwer, 2000, pp. 51–74. DOI: 10.1007/978-0-387-35514-6_3.

[BKS07] H. Bock, E. Kostina, and J. Schlöder. "Numerical Methods for Parameter Estimation in Nonlinear Differential Algebraic Equations." In: *GAMM Mitteilungen* 30/2 (2007), pp. 376–408. DOI: 10.1002/gamm.200790024.

[BL80] H. Bock and R. Longman. "Optimal Control of Velocity Profiles for Minimization of Energy Consumption in the New York Subway System." In: *Proceedings of the Second IFAC Workshop on Control Applications of Nonlinear Programming and Optimization*. International Federation of Automatic Control. 1980, pp. 34–43.

[BL82] H. Bock and R. Longman. "Computation of optimal controls on disjoint control sets for minimum energy subway operation." In: *Proceedings of the American Astronomical Society. Symposium on Engineering Science and Mechanics*. Taiwan, 1982.

[BL85] H. Bock and R. Longman. "Computation of optimal controls on disjoint control sets for minimum energy subway operation." In: *Advances in the Astronautical Sciences* 50 (1985), pp. 949–972.

[BM12] L. Berkovitz and N. Medhin. *Nonlinear Optimal Control Theory*. Chapman & Hall/CRC Applied Mathematics & Nonlinear Science. CRC Press, 2012. ISBN: 9781466560277.

[BNW06] R. Byrd, J. Nocedal, and R. Waltz. "Knitro: An Integrated Package for Nonlinear Optimization." In: *Large-Scale Nonlinear Optimization*. Ed. by G. Pillo and M. Roma. Vol. 83. Nonconvex Optimization and Its Applications. Springer US, 2006, pp. 35–59. ISBN: 978-0-387-30063-4. DOI: 10.1007/0-387-30065-1_4.

[BNW08] R. H. Byrd, J. Nocedal, and R. A. Waltz. "Steering exact penalty methods for nonlinear programming." In: *Optimization Methods and Software* 23.2 (2008), pp. 197–213. DOI: 10.1080/10556780701394169.

[Boc77] H. Bock. "Zur numerischen Behandlung zustandsbeschränkter Steuerungsprobleme mit Mehrzielmethode und Homotopieverfahren." In: *Zeitschrift für Angewandte Mathematik und Mechanik* 57.4 (1977), T266–T268.

[Boc78a] H. Bock. *Numerische Berechnung zustandsbeschränkter optimaler Steuerungen mit der Mehrziehlmethode*. Heidelberg: Carl-Cranz-Gesellschaft, 1978.

[Boc78b] H. Bock. "Numerical Solution of Nonlinear Multipoint Boundary Value Problems with Applications to Optimal Control." In: *Zeitschrift für Angewandte Mathematik und Mechanik* 58 (1978), T407–T409. DOI: 10.1002/zamm.19780580706.

[Boc81a] H. Bock. "Numerical treatment of inverse problems in chemical reaction kinetics." In: *Modelling of Chemical Reaction Systems*. Ed. by K. Ebert, P. Deuflhard, and W. Jäger. Vol. 18. Springer Series in Chemical Physics. Heidelberg: Springer, 1981, pp. 102–125. DOI: 10.1007/978-3-642-68220-9_8.

[Boc81b] H. Bock. *Numerische Behandlung von zustandsbeschränkten und Chebyshev-Steuerungsproblemen*. Tech. rep. R106/81/11. Heidelberg: Carl Cranz Gesellschaft, 1981. URL: http://www.iwr.uni-heidelberg.de/groups/agbock/FILES/Bock1981b.pdf.

[Boc83] H. Bock. "Recent advances in parameter identification techniques for ODE." In: *Numerical Treatment of Inverse Problems in Differential and Integral Equations*. Ed. by P. Deuflhard and E. Hairer. Boston: Birkhäuser, 1983, pp. 95–121. DOI: 10.1007/978-1-4684-7324-7_7.

[Boc87] H. Bock. *Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*. Ed. by E. Brieskorn et al. Vol. 183. Bonner Mathematische Schriften. 1987. URL: http://www.iwr.uni-heidelberg.de/groups/agbock/FILES/Bock1987.pdf.

[Bon+08] P. Bonami et al. "An Algorithmic Framework for Convex Mixed Integer Nonlinear Programs." In: *Discrete Optimization* 5.2 (2008), pp. 186–204. DOI: 10.1016/j.disopt.2006.10.011.

[Bou38] N. Bourbaki. "Sur les espaces de Banach." In: *Comptes rendus de l'Académie des sciences* 206 (1938), pp. 1701–1704.

[BP84] H. Bock and K. Plitt. "A Multiple Shooting algorithm for direct solution of optimal control problems." In: *Proceedings of the 9th IFAC World Congress*. Budapest: Pergamon Press, 1984, pp. 242–247. URL: http://www.iwr.uni-heidelberg.de/groups/agbock/FILES/Bock1984.pdf.

[BRB08] B. Baumrucker, J. Renfro, and L. Biegler. "MPEC problem formulations and solution strategies with chemical engineering applications." In: *Computers & Chemical Engineering* 32 (2008), pp. 2903–2913. DOI: 10.1016/j.compchemeng.2008.02.010.

[Bro70] C. G. Broyden. "The convergence of a class of double–rank minimization algorithms." In: *Journal of the Institute of Mathematics and its Applications* 6 (1970), pp. 76–90. DOI: 10.1093/imamat/6.1.76.

[Buc10] A. Buchner. "Auf Dynamischer Programmierung basierende nichtlineare modellprädiktive Regelung für LKW." Diplomarbeit. Heidelberg University, Jan. 2010. URL: http://mathopt.de/PUBLICATIONS/Buchner2010.pdf.

[Bul71] R. Bulirsch. *Die Mehrzielmethode zur numerischen Lösung von nichtlinearen Randwertproblemen und Aufgaben der optimalen Steuerung*. Tech. rep. Oberpfaffenhofen: Carl-Cranz-Gesellschaft, 1971.

[Bur11] H. Burgdörfer. "Strukturausnutzende Algorithmen der Quadratischen Programmierung für gemischt-ganzzahlige Optimalsteuerung." Diplomarbeit. Heidelberg University, 2011.

[Byr+03] R. Byrd, N. Gould, J. Nocedal, and R. Waltz. "An algorithm for nonlinear optimization using linear programming and equality constrained subproblems." In: *Mathematical Programming, Series B* 100.1 (2003), pp. 27–48. ISSN: 0025-5610. DOI: 10.1007/s10107-003-0485-4.

[Byr+05] R. Byrd, N. Gould, J. Nocedal, and R. Waltz. "On the Convergence of Successive Linear-Quadratic Programming Algorithms." In: *SIAM Journal on Optimization* 16.2 (2005), pp. 471–489. DOI: 10.1137/S1052623403426532.

[Byr+13] R. Byrd, J. Nocedal, R. Waltz, and Y. Wu. "On the use of piecewise linear models in nonlinear programming." In: *Mathematical Programming, Series A* 137.1-2 (2013), pp. 289–324. ISSN: 0025-5610. DOI: 10.1007/s10107-011-0492-9.

[Byr87] R. H. Byrd. "Robust trust region methods for constrained optimization." In: *Third SIAM Conference on Optimization*. 1987.

[Cas86] E. Casas. "Control of an Elliptic Problem with Pointwise State Constraints." In: *SIAM Journal on Control and Optimization* 24.6 (1986), pp. 1309–1318. DOI: 10.1137/0324078.

[Ces83] L. Cesari. *Optimization — Theory and Applications*. Springer Verlag, 1983. DOI: 10.1007/978-1-4613-8165-5.

[CF03] C. Chin and R. Fletcher. "On the global convergence of an SLP-filter algorithm that takes EQP steps." In: *Mathematical Programming, Series A* 96.1 (2003), pp. 161–177. ISSN: 0025-5610. DOI: 10.1007/s10107-003-0378-6.

[CGT00] A. Conn, N. Gould, and P. Toint. *Trust Region Methods*. Society for Industrial and Applied Mathematics, 2000. DOI: 10.1137/1.9780898719857.

[CGT11a] C. Cartis, N. Gould, and P. Toint. "Adaptive cubic regularisation methods for unconstrained optimization. Part I: motivation, convergence and numerical results." In: *Mathematical Programming, Series A* 127.2 (2011), pp. 245–295. ISSN: 1436-4646. DOI: 10.1007/s10107-009-0286-5.

[CGT11b] C. Cartis, N. Gould, and P. Toint. "Adaptive cubic regularisation methods for unconstrained optimization. Part II: worst-case function- and derivative-evaluation complexity." In: *Mathematical Programming, Series A* 130.2 (2011), pp. 295–319. ISSN: 1436-4646. DOI: 10.1007/s10107-009-0337-y.

[Che+06] Y. Chen, B. F. Hobbs, S. Leyffer, and T. S. Munson. "Leader-Follower Equilibria for Electric Power and NO_x Allowances Markets." In: *Computational Management Science* 3.4 (Sept. 2006), pp. 307–330. ISSN: 1619-6988. DOI: 10.1007/s10287-006-0020-1.

[CKA95] H. Chen, A. Kremling, and F. Allgöwer. "Nonlinear predictive control of a benchmark CSTR." In: *Proc. 3rd European Control Conference ECC'95*. Rome, 1995, pp. 3247–3252.

[Cla13] F. Clarke. *Functional Analysis, Calculus of Variations and Optimal Control*. Vol. 264. Graduate Texts in Mathematics. Springer-Verlag London, 2013. DOI: 10.1007/978-1-4471-4820-3.

[CS16] F. Curtis and D. R. M. Samadi. "A trust region algorithm with a worst-case iteration complexity of $O(\varepsilon^{-3/2})$ for nonconvex optimization." In: *Mathematical Programming, Series A* (2016), pp. 1–32. ISSN: 1436-4646. DOI: 10.1007/s10107-016-1026-2.

[DBK06] M. Diehl, H. Bock, and E. Kostina. "An approximation technique for robust nonlinear optimization." In: *Mathematical Programming, Series B* 107 (2006), pp. 213–230. DOI: 10.1007/s10107-005-0685-1.

[DeM+05] V. DeMiguel, M. P. Friedlander, F. J. Nogales, and S. Scholtes. "A two-sided relaxation scheme for Mathematical Programs with Equilibrium Constraints." In: *SIAM Journal on Optimization* 16.2 (2005), pp. 587–609. DOI: 10.1137/04060754x.

[Dem02] S. Dempe. *Foundations of Bilevel Programming*. Springer US, 2002. DOI: 10.1007/b101970.

[Deu06] P. Deuflhard. *Newton Methods for Nonlinear Problems. Affine Invariance and Adaptive Algorithms*. Vol. 35. Springer Series in Computational Mathematics. Springer, 2006.

[Deu74] P. Deuflhard. "A Modified Newton Method for the Solution of Ill-conditioned Systems of Nonlinear Equations with Applications to Multiple Shooting." In: *Numerische Mathematik* 22 (1974), pp. 289–311. DOI: 10.1007/BF01406969.

[DF95] S. P. Dirkse and M. C. Ferris. "The path solver: a nommonotone stabilization scheme for mixed complementarity problems." In: *Optimization Methods and Software* 5.2 (1995), pp. 123–156. DOI: 10.1080/10556789508805606.

[DFH09] M. Diehl, H. Ferreau, and N. Haverbeke. "Efficient Numerical Methods for Nonlinear MPC and Moving Horizon Estimation." In: *Nonlinear Model Predictive Control*. Ed. by L. Magni, D. Raimondo, and F. Allgöwer. Vol. 384. Springer Lecture Notes in Control and Information Sciences. Berlin, Heidelberg, New York: Springer-Verlag, 2009, pp. 391–417. DOI: 10.1007/978-3-642-01094-1.

[Die+02] M. Diehl et al. "Real-time optimization and Nonlinear Model Predictive Control of Processes governed by differential-algebraic equations." In: *Journal of Process Control* 12.4 (2002), pp. 577–585. DOI: 10.1016/S0959-1524(01)00023-3.

[Die+08] M. Diehl, J. Gerhard, W. Marquardt, and M. Mönnigmann. "Numerical solution approaches for robust optimal control problems." In: *Computers & Chemical Engineering* 32 (2008), pp. 1279–1292. DOI: 10.1016/j.compchemeng.2007.06.002.

[Die+16] M. Diehl et al. *MUSCOD-II Users' Manual*. Tech. rep. Universität Heidelberg, 2016.

[Die01] M. Diehl. "Real-Time Optimization for Large Scale Nonlinear Processes." Dissertation. Heidelberg University, 2001. DOI: 10.11588/heidok.00001659.

[DLS01] M. Diehl, D. Leineweber, and A. Schäfer. *MUSCOD-II Users' Manual*. IWR-Preprint 2001-25. Universität Heidelberg, 2001.

[DM02] E. Dolan and J. Moré. "Benchmarking optimization software with performance profiles." English. In: *Mathematical Programming, Series A* 91.2 (2002), pp. 201–213. ISSN: 0025-5610. DOI: 10.1007/s101070100263.

[Duf04] I. Duff. "MA57 — a code for the solution of sparse symmetric definite and indefinite systems." In: *ACM Transactions on Mathematical Software* 30.2 (2004), pp. 118–144. DOI: 10.1145/992200.992202.

[DW75] E. D. Dickmanns and K. H. Well. "Approximate solution of optimal control problems using third order hermite polynomial functions." In: *Optimization Techniques IFIP Technical Conference Novosibirsk, July 1–7, 1974*. Ed. by G. I. Marchuk. Berlin, Heidelberg: Springer Berlin Heidelberg, 1975, pp. 158–166. ISBN: 978-3-540-37497-8. DOI: 10.1007/3-540-07165-2_21.

[DZ13] S. Dempe and A. B. Zemkoho. "The bilevel programming problem: reformulations, constraint qualifications and optimality conditions." In: *Mathematical Programming, Series A* 138.1 (Apr. 2013), pp. 447–473. ISSN: 1436-4646. DOI: 10.1007/s10107-011-0508-5.

[EG10] J. Erway and P. Gill. "A Subspace Minimization Method for the Trust-Region Step." In: *SIAM Journal on Optimization* 20.3 (2010), pp. 1439–1461. DOI: 10.1137/08072440X.

[EGG09] J. Erway, P. Gill, and J. Griffin. "Iterative Methods for Finding a Trust-region Step." In: *SIAM Journal on Optimization* 20.2 (2009), pp. 1110–1131. DOI: 10.1137/070708494.

[EWA06] M. Egerstedt, Y. Wardi, and H. Axelsson. "Transition-time optimization for switched-mode dynamical systems." In: *IEEE Transactions on Automatic Control* 51.1 (2006), pp. 110–115. DOI: 10.1109/TAC.2005.861711.

[EWD03] M. Egerstedt, Y. Wardi, and F. Delmotte. "Optimal Control of Switching Times in Switched Dynamical Systems." In: *Proceedings of the 42nd IEEE Conference of Decision and Control*. 2003.

[Far+13] P. E. Farrell, D. A. Ham, S. W. Funke, and M. E. Rognes. "Automated Derivation of the Adjoint of High-Level Transient Finite Element Programs." In: *SIAM Journal on Scientific Computing* 35.4 (2013), pp. C369–C393. DOI: 10.1137/120873558.

[Far02] J. Farkas. "Theorie der einfachen Ungleichungen." In: *Journal für die reine und angewandte Mathematik* 124 (1902), pp. 1–27.

[FDM05] M. C. Ferris, S. P. Dirkse, and A. Meeraus. "Mathematical Programs with Equilibrium Constraints: Automatic Reformulation and Solution via Constrained Optimization." In: *Frontiers in Applied General Equilibrium Modeling: In Honor of Herbert Scarf*. Ed. by T. J. Kehoe and T. N. S. J. Whalley. Cambridge University Press, 2005, pp. 67–94. DOI: 10.1017/CBO9780511614330.005.

[Feh69] E. Fehlberg. "Klassische Runge-Kutta-Formeln fünfter und siebenter Ordnung mit Schrittweiten-Kontrolle." In: *Computing* 4 (1969), pp. 93–106. DOI: 10.1007/BF02234758.

[Feh70] E. Fehlberg. "Klassische Runge-Kutta-Formeln vierter und niedrigerer Ordnung mit Schrittweiten-Kontrolle und ihre Anwendung auf Wärmeleitungsprobleme." In: *Computing* 6 (1970), pp. 61–71. DOI: 10.1007/BF02241732.

[FF13] S. W. Funke and P. E. Farrell. *A framework for automated PDE-constrained optimisation*. Tech. rep. 2013. arXiv: 1302.3894.

[FFG99] M. Ferris, R. Fourer, and D. Gay. "Eexpressing Complementarity Problems in an Algebraic Modeling Language and Communicating Tthem to Solvers." In: *SIAM Journal on Optimization* 9.4 (1999), pp. 991–1009. DOI: 10.1137/S105262349833338X.

[FGK02] R. Fourer, D. Gay, and B. Kernighan. *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press, 2002.

[FGK90] R. Fourer, D. Gay, and B. Kernighan. "A Modeling Language for Mathematical Programming." In: *Management Science* 36 (1990), pp. 519–554. DOI: 10.1287/mnsc.36.5.519.

[FJQ99] F. Facchinei, H. Jiang, and L. Qi. "A smoothing method for mathematical programs with equilibrium constraints." In: *Mathematical Programming* 85 (1999), pp. 107–134. DOI: 10.1007/s10107990015a.

[FK03] M. L. Flegel and C. Kanzow. "A Fritz John Approach to First Order Optimality Conditions for Mathematical Programs with Equilibrium Constraints." In: *Optimization* 52.3 (2003), pp. 277–286. DOI: 10.1080/0233193031000120020.

[FL02] R. Fletcher and S. Leyffer. "Nonlinear programming without a penalty function." In: *Mathematical Programming, Series A* 91.2 (2002), pp. 239–269. ISSN: 00255610. DOI: 10.1007/s101070100244.

[FL04] R. Fletcher and S. Leyffer. "Ssolving mathematical programs with complementarity constraints as nonlinear programs." In: *Optimization Methods and Software* 19.1 (2004), pp. 15–40. DOI: 10.1080/10556780410001654241.

[Fle+06] R. Fletcher, S. Leyffer, D. Ralph, and S. Scholtes. "Local Convergence of SQP Methods for Mathematical Programs with Equilibrium Constraints." In: *SIAM Journal on Optimization* 17 (2006), pp. 259–286. DOI: 10.1137/S1052623402407382.

[Fle05] M. Flegel. "Constraint Qualifications and Stationarity Concepts for Mathematical Programs with Equilibrium Constraints." PhD thesis. University of Würzburg, 2005. URL: http://nbn-resolving.org/urn:nbn:de:bvb:20-opus-12453.

[Fle70] R. Fletcher. "A new approach to variable metric algorithms." In: *Computer Journal* 13 (1970), pp. 317–322. DOI: 10.1093/comjnl/13.3.317.

[Fle87] R. Fletcher. *Practical Methods of Optimization*. Second Edition. Chichester: Wiley, 1987. DOI: 10.1002/9781118723203.

[FLM12] H. Fang, S. Leyffer, and T. Munson. "A pivoting algorithm for linear programming with linear complementarity constraints." In: *Optimization Methods and Software* 27.1 (2012), pp. 89–114. DOI: 10.1080/10556788.2010.512956.

[FLP98] M. Fukushima, Z.-Q. Luo, and J.-S. Pang. "A Globally Convergent Sequential Quadratic Programming Algorithm for Mathematical Programs with Linear Complementarity Constraints." In: *Computational Optimization and Applications* 10.1 (Apr. 1998), pp. 5–34. ISSN: 1573-2894. DOI: 10.1023/A:1018359900133.

[FM89] R. Fletcher and E. S. de la Maza. "Nonlinear programming and nonsmooth optimization by successive linear programming." In: *Mathematical Programming* 43.1–3 (1989), pp. 235–256. ISSN: 0025-5610. DOI: 10.1007/BF01582292.

[FMT02] R. Franke, M. Meyer, and P. Terwiesch. "Optimal Control of the Driving of Trains." In: *Automatisierungstechnik* 50.12 (2002), pp. 606–614. DOI: 10.1524/auto.2002.50.12.606.

[Fox60] L. Fox. "Some numerical experiments with eigenvalue problems in ordinary differential equations." In: *Boundary Value Problems in Differential Equations*. Ed. by R. Langer. 1960.

[FP97] M. C. Ferris and J. S. Pang. "Engineering and Economic Applications of Complementarity Problems." In: *SIAM Review* 39.4 (1997), pp. 669–713. DOI: 10.1137/S0036144595285963.

[FP99] M. Fukushima and J.-S. Pang. "Convergence of a Smoothing Continuation Method for Mathematical Progams with Complementarity Constraints." In: *Ill-posed Variational Problems and Regularization Techniques*. Berlin, Heidelberg: Springer, 1999, pp. 99–110. ISBN: 978-3-642-45780-7. DOI: 10.1007/978-3-642-45780-7_7.

[Fra+12] J. Frasch, L. Wirsching, S. Sager, and H. Bock. "Mixed-Level Iteration Schemes for Nonlinear Model Predictive Control." In: *Proceedings of the IFAC Conference on Nonlinear Model Predictive Control*. 2012. DOI: 10.3182/20120823-5-NL-3013.00085.

[FT99] M. Fukushima and P. Tseng. "An Implementable Active–Set Algorithm for Computing a B–Stationary Point of a Mathematical Program with Linear Complementarity Constraints." In: *SIAM Journal on Optimization* 12 (1999), pp. 724–739. DOI: 10.1137/S1052623499363232.

[Gal97] J. Gallitzendörfer. *Parallele Algorithmen für Optimierungsrandwertprobleme*. Fortschritt-Berichte / VDI : Reihe 10, Informatik, Kommunikationstechnik. VDI-Verlag, 1997. ISBN: 9783183514106.

[GAM] GAMS. *GAMS homepage*. URL: http://www.gams.com.

[GB94] J. Gallitzendörfer and H. Bock. "Parallel Algorithms for optimization boundary value problems in DAE." In: *Praxisorientierte Parallelverarbeitung*. Ed. by H. Langendörfer. Hanser, München, 1994.

[Ger05] M. Gerdts. "Solving mixed-integer optimal control problems by branch& bound: a case study from automobile test-driving with gear shift." In: *Optimal Control Applications and Methods* 26 (2005), pp. 1–18. DOI: 10.1002/oca.751.

[Gfr14] H. Gfrerer. "Optimality Conditions for Disjunctive Programs Based on Generalized Differentiation with Application to Mathematical Programs with Equilibrium Constraints." In: *SIAM Journal on Optimization* 24.2 (2014), pp. 898–931. DOI: 10.1137/130914449.

[GHN01] N. Gould, M. Hribar, and J. Nocedal. "On the solution of equality constrained quadratic programming problems arising in optimization." In: *SIAM Journal on Scientific Computing* 23.4 (2001), pp. 1376–1395. DOI: 10.1137/S1064827598345667.

[GJ79] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: W.H. Freeman, 1979.

[GL96] G. Golub and C. van Loan. *Matrix Computations*. 3rd. Baltimore: Johns Hopkins University Press, 1996.

[GMS02] P. Gill, W. Murray, and M. Saunders. "SNOPT: An SQP algorithm for large-scale constrained optimization." In: *SIAM Journal on Optimization* 12 (2002), pp. 979–1006. DOI: 10.1137/S0036144504446096.

[GMS95] P. Gill, W. Murray, and M. Saunders. *User's Guide For QPOPT 1.0: A Fortran Package For Quadratic Programming*. 1995. URL: http://www.sbsi-sol-optimize.com/manuals/QPOPT%20Manual.pdf.

[Gol70] D. Goldfarb. "A family of variable metric updates derived by variational means." In: *Mathematics of Computation* 24 (1970), pp. 23–26. DOI: 10.2307/2004873.

[GOT02] N. Gould, D. Orban, and P. Toint. *CUTEr testing environment for optimization and linear algebra solvers*. http://cuter.rl.ac.uk/cuter-www/. 2002. URL: http://cuter.rl.ac.uk/cuter-www/.

[GOT04] N. I. Gould, D. Orban, and P. L. Toint. "GALAHAD, a library of thread-safe Fortran 90 packages for large-scale nonlinear optimization." In: *ACM Transactions on Mathematical Software* 29.4 (2004), pp. 353–372. DOI: 10.1145/962437.962438.

[GOT15] N. I. Gould, D. Orban, and P. L. Toint. "CUTEst: a Constrained and Unconstrained Testing Environment with safe threads for mathematical optimization." In: *Computational Optimization and Applications* 60.3 (Apr. 2015), pp. 545–557. ISSN: 1573-2894. DOI: 10.1007/s10589-014-9687-3.

[Gou+99] N. Gould, S. Lucidi, M. Roma, and P. Toint. "Solving the Trust-Region Subproblem using the Lanczos Method." In: *SIAM Journal on Optimization* 9.2 (1999), pp. 504–525. DOI: 10.1137/S1052623497322735.

[GR08] G. Giallombardo and D. Ralph. "Multiplier convergence in trust-region methods with application to convergence of decomposition methods for MPECs." In: *Mathematical Programming, Series A* 112.2 (Apr. 2008), pp. 335–369. ISSN: 1436-4646. DOI: 10.1007/s10107-006-0020-5.

[Gro+16] S. Gros et al. "From Linear to Nonlinear MPC: bridging the gap via the Real-Time Iteration." In: *International Journal of Control* (2016), pp. 1–19. DOI: 10.1080/00207179.2016.1222553.

[Grö19] T. H. Grönwall. "Note on the Derivatives with Respect to a Parameter of the Solutions of a System of Differential Equations." In: *The Annals of Mathematics* 20 (1919), pp. 292–296. DOI: 10.2307/1967124.

[GT10] N. Gould and P. Toint. "Nonlinear programming without a penalty function or a filter." In: *Mathematical Programming, Series A* 122 (2010), pp. 155–196. DOI: 10.1007/s10107-008-0244-7.

[Gui69] M. Guignard. "Generalized Kuhn–Tucker conditions for mathematical programming problems in a Banach space." In: *SIAM Journal on Control* 7.2 (1969), pp. 232–241. DOI: 10.1137/0307016.

[Gur15] Gurobi. *Gurobi Optimizer Reference Manual*. 2015. URL: http://www.gurobi.com.

[GW08] A. Griewank and A. Walther. *Evaluating Derivatives*. Second. SIAM, 2008. DOI: 10.1137/1.9780898717761.

[Hag01] W. Hager. "Minimizing a Quadratic Over a Sphere." In: *SIAM Journal on Optimization* 12.1 (2001), pp. 188–208. DOI: 10.1137/S1052623499356071.

[Hat+13] K. Hatz, S. Leyffer, J. Schlöder, and H. Bock. *Regularizing bilevel nonlinear programs by lifting*. Tech. rep. ANL/MCS-P4076-0613. Argonne National Laboratory, Mathematics and Computer Science Division, 2013.

[Hat14] K. Hatz. "Efficient numerical methods for hierarchical dynamic optimization with application to cerebral palsy gait modeling." Dissertation. Heidelberg University, 2014. DOI: 10.11588/heidok.00016803.

[HD13] B. Houska and M. Diehl. "Nonlinear robust optimization via sequential convex bilevel programming." In: *Mathematical Programming, Series A* 142 (2013), pp. 539–577. DOI: 10.1007/s10107-012-0591-2.

[Hei93] M. Heinkenschloss. "Mesh Independence for Nonlinear Least Squares Problems with Norm Constraints." In: *SIAM Journal on Optimization* 3.1 (1993), pp. 81–117. DOI: 10.1137/0803005.

[Hel+09] E. Hellström, M. Ivarsson, J. Aslund, and L. Nielsen. "Look-ahead control for heavy trucks to minimize trip time and fuel consumption." In: *Control Engineering Practice* 17 (2009), pp. 245–254. DOI: 10.1016/j.conengprac.2008.07.005.

[Her+] R. Herzog, A. Rösch, S. Ulbrich, and W. Wollner, eds. *OPTPDE — A Collection of Problems in PDE-Constrained Optimization*. URL: http://www.optpde.net.

[Her+14] R. Herzog, A. Rösch, S. Ulbrich, and W. Wollner. "OPTPDE: A Collection of Problems in PDE-Constrained Optimization." In: *Trends in PDE Constrained Optimization*. Ed. by G. Leugering et al. Vol. 165. International Series of Numerical Mathematics. Springer International Publishing, 2014, pp. 539–543. ISBN: 978-3-319-05082-9. DOI: 10.1007/978-3-319-05083-6_34.

[Hes51] M. R. Hestenes. "Applications of the theory of quadratic forms in Hilbert space to the calculus of variations." In: *Pacific Journal of Mathematics* 1.4 (1951), pp. 525–581. URL: http://projecteuclid.org/euclid.pjm/1103052021.

[Hes66] M. Hestenes. *Calculus of variations and optimal control theory*. New York: Wiley, 1966.

[HFD11] B. Houska, H. Ferreau, and M. Diehl. "ACADO Toolkit – An Open Source Framework for Automatic Control and Dynamic Optimization." In: *Optimal Control Applications and Methods* 32.3 (2011), pp. 298–312. DOI: 10.1002/oca.939/abstract.

[HK07] T. Hoheisel and C. Kanzow. "First- and second-order optimality conditions for mathematical programs with vanishing constraints." In: *Applications of Mathematics* 52.6 (2007), pp. 459–514.

[HK08] T. Hoheisel and C. Kanzow. "Stationary Conditions for Mathematical Programs with Vanishing Constraints using Weak Constraint Qualifications." In: *Journal of Mathematical Analysis and Applications* 337 (2008), pp. 292–310. DOI: 10.1016/j.jmaa.2007.03.087.

[HK09a] M. Hintermüller and I. Kopacka. "Mathematical Programs with Complementarity Constraints in Function Space: C- and Strong Stationarity and a Path-Following Algorithm." English. In: *SIAM Journal on Optimization* 20 (2009), pp. 868–902. DOI: 10.1137/080720681.

[HK09b] T. Hoheisel and C. Kanzow. "On the Abadie and Guignard constraint qualifications for mathematical programs with vanishing constraints." In: *Optimization* 58.4 (May 2009), pp. 431–448. DOI: 10.1080/02331930701763405.

[HKS13] T. Hoheisel, C. Kanzow, and A. Schwartz. "Theoretical and numerical comparison of relaxation methods for mathematical programs with complementarity constraints." In: *Mathematical Programming, Series A* 137.1–2 (Feb. 2013), pp. 257–288. DOI: 10.1007/s10107-011-0488-5.

[HL69] H. Hermes and J. Lasalle. *Functional analysis and time optimal control*. Ed. by R. Bellmann. Vol. 56. Mathematics in science and engineering. New York and London: Academic Press, 1969.

[HM79] S. Han and O. Mangasarian. "Exact penalty functions in nonlinear programming." In: *Mathematical Programming* 17.1 (1979), pp. 251–269. ISSN: 0025-5610. DOI: 10.1007/BF01588250.

[HNW93] E. Hairer, S. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I.* Second Edition. Vol. 8. Springer Series in Computational Mathematics. Berlin: Springer-Verlag, 1993. DOI: 10.1007/978-3-540-78862-1.

[Hoh09] T. Hoheisel. "Mathematical Programs with Vanishing Constraints." Dissertation. University of Würzburg, July 2009. URL: http://nbn-resolving.org/urn:nbn:de:bvb:20-opus-40790.

[HR04] X. Hu and D. Ralph. "Convergence of a Penalty Method for Mathematical Programming with Complementarity Constraints." In: *Journal of Optimization Theory and Applications* 123.2 (2004), pp. 365–290. DOI: 10.1007/s10957-004-5154-0.

[HS11] M. Hintermüller and T. Surowiec. "First-Order Optimality Conditions for Elliptic Mathematical Programs with Equilibrium Constraints via Variational Analysis." In: *SIAM Journal on Optimization* 21.4 (2011), pp. 1561–1593. DOI: 10.1137/100802396.

[HS97] M. Henson and D. Seborg. *Nonlinear Process Control.* Prentice Hall PTR, 1997. ISBN: 9780136251798.

[Hu+12] J. Hu, J. E. Mitchell, J.-S. Pang, and B. Yu. "On linear programs with linear complementarity constraints." In: *Journal of Global Optimization* 53.1 (May 2012), pp. 29–51. ISSN: 1573-2916. DOI: 10.1007/s10898-010-9644-3.

[Hu08] J. Hu. "On linear programs with linear complementarity constraints." PhD thesis. Rensselaer Polytechnic Institute, 2008. URL: http://www.rpi.edu/~mitchj/phdtheses/jing/rpithes.pdf.

[HW96] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II.* Second Edition. Vol. 14. Springer Series in Computational Mathematics. Berlin: Springer, 1996. DOI: 10.1007/978-3-642-05221-7.

[IPS12] A. Izmailov, A. Pogosyan, and M. Solodov. "Semismooth Newton method for the lifted reformulation of mathematical programs with complementarity constraints." In: *Computational Optimization Applications* 51 (2012), pp. 199–221. DOI: 10.1007/s10589-010-9341-7.

[IS09] A. Izmailov and M. Solodov. "Mathematical Programs with Vanishing Constraints: Optimality Conditions, Sensitivity, and a Relaxation Method." In: *Journal of Optimization Theory and Applications* 142 (2009), pp. 501–532. DOI: 10.1007/s10957-009-9517-4.

[Jac01] R. H. F. Jackson. "Factorable programming." In: *Encyclopedia of Operations Research and Management Science*. Boston, MA: Springer US, 2001, pp. 285–288. ISBN: 978-1-4020-0611-1. DOI: 10.1007/1-4020-0611-X_328.

[Jan10] D. Janka. "Optimum Experimental Design and Multiple Shooting." Diplomarbeit. Heidelberg University, 2010.

[Jan15] D. Janka. "Sequential quadratic programming with indefinite Hessian approximations for nonlinear optimum experimental design for parameter estimation in differential–algebraic equations." Dissertation. Heidelberg University, 2015. DOI: 10.11588/heidok.00019170.

[Jan84] R. Janin. "Directional derivative of the marginal function in nonlinear programming." In: *Sensitivity, Stability and Parametric Analysis*. Ed. by A. V. Fiacco. Berlin, Heidelberg: Springer, 1984, pp. 110–126. ISBN: 978-3-642-00913-6. DOI: 10.1007/BFb0121214.

[JKS13] M. Jung, C. Kirches, and S. Sager. "On Perspective Functions and Vanishing Constraints in Mixed-Integer Nonlinear Optimal Control." In: *Facets of Combinatorial Optimization – Festschrift for Martin Grötschel*. Ed. by M. Jünger and G. Reinelt. Springer Berlin Heidelberg, 2013, pp. 387–417. DOI: 10.1007/978-3-642-38189-8_16.

[JOP+15] E. Jones, T. Oliphant, P. Peterson, et al. *SciPy: Open source scientific tools for Python*. 2001–2015. URL: http://www.scipy.org/.

[JR00] H. Jiang and D. Ralph. "Smooth SQP Methods for Mathematical Programs with Nonlinear Complementarity Constraints." In: *SIAM Journal on Optimization* 10.3 (2000), pp. 779–808. DOI: 10.1137/S1052623497332329.

[Júd+07] J. Júdice, H. Sherali, I. Ribeiro, and A. Faustino. "Complementarity Active-Set Algorithm for Mathematical Programming Problems with Equilibrium Constraints." In: *Journal of Optimization Theory and Applications* 134 (2007), pp. 467–481. DOI: 10.1007/s10957-007-9231-z.

[Júd12] J. J. Júdice. "Algorithms for linear programming with linear complementarity constraints." In: *TOP* 20.1 (Apr. 2012), pp. 4–25. ISSN: 1863-8279. DOI: 10.1007/s11750-011-0228-2.

[Jun13] M. Jung. "Relaxations and Approximations for Mixed-Integer Optimal Control." Dissertation. Heidelberg Univeristy, 2013. DOI: 10.11588/heidok.00016036.

[Kar39] W. Karush. "Minima of functions of several variables with inequalities as side conditions." MA thesis. Department of Mathematics, University of Chicago, 1939.

[KB06] S. Kameswaran and L. Biegler. "Simultaneous dynamic optimization strategies: Recent advances and challenges." In: *Computers & Chemical Engineering* 30 (2006), pp. 1560–1575. DOI: 10.1016/j.compchemeng.2006.05.034.

[KDB09] A. Kadrani, J.-P. Dussault, and A. Benchakroun. "A New Regularization Scheme for Mathematical Programs with Complementarity Constraints." In: *SIAM Journal on Optimization* 20.1 (2009), pp. 78–103. DOI: 10.1137/070705490.

[KE93] K.-U. Klatt and S. Engell. "Rührkesselreaktor mit Parallel- und Folgereaktion." In: *Nichtlineare Regelung – Methoden, Werkzeuge, Anwendungen. VDI-Berichte Nr. 1026*. Ed. by S. Engell. Düsseldorf: VDI-Verlag, 1993, pp. 101–108.

[Kel68] H. Keller. *Numerical Methods for Two-Point Boundary Value Problems*. Blaisdell, 1968.

[Kir+10a] C. Kirches, S. Sager, H. Bock, and J. Schlöder. "Time-optimal control of automobile test drives with gear shifts." In: *Optimal Control Applications and Methods* 31.2 (Mar. 2010), pp. 137–153. DOI: 10.1002/oca.892.

[Kir+10b] C. Kirches, L. Wirsching, S. Sager, and H. Bock. "Efficient Numerics for Nonlinear Model Predictive Control." In: *Recent Advances in Optimization and its Applications in Engineering*. Ed. by M. Diehl, F. Glineur, E. Jarlebring, and W. Michiels. ISBN 978-3-6421-2597-3. Springer, 2010. Chap. Recent Advances in Optimization and its Applications in Engineering, pp. 339–359. DOI: 10.1007/978-3-642-12598-0_30.

[Kir+11] C. Kirches, H. Bock, J. Schlöder, and S. Sager. "Block Structured Quadratic Programming for the Direct Multiple Shooting Method for Optimal Control." In: *Optimization Methods and Software* 26.2 (Apr. 2011), pp. 239–257. DOI: 10.1080/10556781003623891.

[Kir+12] C. Kirches, L. Wirsching, H. Bock, and J. Schlöder. "Efficient Direct Multiple Shooting for Nonlinear Model Predictive Control on Long Horizons." In: *Journal of Process Control* 22.3 (2012), pp. 540–550. DOI: 10.1016/j.jprocont.2012.01.008.

[Kir+13a] C. Kirches, H. Bock, J. Schlöder, and S. Sager. "Mixed-integer NMPC for predictive cruise control of heavy-duty trucks." In: *European Control Conference*. Zurich, Switzerland, July 2013, pp. 4118–4123. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6669210.

[Kir+13b] C. Kirches, A. Potschka, H. Bock, and S. Sager. "A Parametric Active Set Method for a Subclass of Quadratic Programs with Vanishing Constraints." In: *Pacific Journal of Optimization* 9.2 (2013), pp. 275–299. URL: http://www.ybook.co.jp/online2/pjov9n2.html.

[Kir+15] C. Kirches, M. Jung, F. Lenders, and S. Sager. "Approximation properties of complementarity problems from mixed-integer optimal control." In: *Mixed-integer Nonlinear Optimization: A Hatchery for Modern Mathematics*. Ed. by L. Liberti, S. Sager, and A. Wiegele. Vol. 12. Oberwolfach Reports 4. 2015, pp. 2736–2737. URL: https://www.mfo.de/document/1543/OWR_2015_46.pdf.

[Kir06] C. Kirches. "A Numerical Method for Nonlinear Robust Optimal Control with Implicit Discontinuities and an Application to Powertrain Oscillations." Diplomarbeit. Heidelberg University, Oct. 2006. URL: http://mathopt.de/PUBLICATIONS/Kirches2006.pdf.

[Kir10] C. Kirches. "Fast numerical methods for mixed-integer nonlinear model-predictive control." Dissertation. Heidelberg University, July 2010. DOI: 10.11588/heidok.00011636.

[Kir17] C. Kirches. *An augmented Lagrangian Gradient Projection Method for Solving Mathematical Programs with Complementarity Constraints*. Working Paper. 2017.

[KL16] C. Kirches and F. Lenders. "Approximation Properties and Tight Bounds for Constrained Mixed-Integer Optimal Control." In: *Optimization Online* (Apr. 2016). (submitted to Mathematical Programming). URL: http://www.optimization-online.org/DB_HTML/2016/04/5404.html.

[KM40] M. Krein and D. Milman. "On extreme points of regular convex sets." eng. In: *Studia Mathematica* 9.1 (1940), pp. 133–138. URL: http://eudml.org/doc/219061.

[KN03] C. Kaya and J. Noakes. "A Computational Method for Time-Optimal Control." In: *Journal of Optimization Theory and Applications* 117 (2003), pp. 69–92. DOI: 10.1023/A:1023600422807.

[Kra85] D. Kraft. "On converting optimal control problems into nonlinear programming problems." In: *Computational Mathematical Programming*. Ed. by K. Schittkowski. Vol. F15. NATO ASI. Springer, 1985, pp. 261–280. DOI: 10.1007/978-3-642-82450-0_9.

[Kra88] D. Kraft. "A software package for sequential quadratic programming." In: *Forschungsbericht- Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt* (1988).

[Kra94] D. Kraft. "Algorithm 733: TOMP–Fortran Modules for Optimal Control Calculations." In: *ACM Transactions on Mathematical Software* 20.3 (Sept. 1994), pp. 262–281. ISSN: 0098-3500. DOI: 10.1145/192115.192124.

[KS13] C. Kanzow and A. Schwartz. "A New Regularization Method for Mathematical Programs with Complementarity Constraints with Strong Convergence Properties." In: *SIAM Journal on Optimization* 23.2 (2013), pp. 770–798. DOI: 10.1137/100802487.

[KS87] C. T. Kelley and E. W. Sachs. "Quasi-Newton Methods and Unconstrained Optimal Control Problems." In: *SIAM Journal on Control and Optimization* 25.6 (1987), pp. 1503–1516. DOI: 10.1137/0325083.

[KT51] H. Kuhn and A. Tucker. "Nonlinear programming." In: *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*. Ed. by J. Neyman. Berkeley: University of California Press, 1951, pp. 481–492.

[Kuz95] Y. A. Kuznetsov. "Efficient iterative solvers for elliptic finite element problems on nonmatching grids." In: *Russian Journal of Numerical Analysis and Mathematical Modelling* 10.3 (1995), pp. 187–212. DOI: 10.1515/rnam.1995.10.3.187.

[KZ06] A. Kurdila and M. Zabarankin. *Convex Functional Analysis*. Systems & Control: Foundations & Applications. Birkhäuser Basel, 2006. ISBN: 9783764373573. DOI: 10.1007/3-7643-7357-1.

[LB89] J. S. Logsdon and L. T. Biegler. "Accurate solution of differential-algebraic optimization problems." In: *Industrial & Engineering Chemistry Research* 28.11 (1989), pp. 1628–1639. DOI: 10.1021/ie00095a010.

[LB92] J. Logsdon and L. Biegler. "Decomposition strategies for large-scale dynamic optimization problems." In: *Chemical Engineering Science* 47.4 (1992), pp. 851–864. DOI: 10.1016/0009-2509(92)80272-E.

[Lei+03a] D. Leineweber, I. Bauer, H. Bock, and J. Schlöder. "An Efficient Multiple Shooting Based Reduced SQP Strategy for Large-Scale Dynamic Process Optimization. Part I: Theoretical Aspects." In: *Computers & Chemical Engineering* 27 (2003), pp. 157–166. DOI: 10.1016/S0098-1354(02)00158-8.

[Lei+03b] D. Leineweber, A. Schäfer, H. Bock, and J. Schlöder. "An Efficient Multiple Shooting Based Reduced SQP Strategy for Large-Scale Dynamic Process Optimization. Part II: Software Aspects and Applications." In: *Computers & Chemical Engineering* 27 (2003), pp. 167–174. DOI: 10.1016/S0098-1354(02)00195-3.

[Lei95] D. Leineweber. "Analyse und Restrukturierung eines Verfahrens zur direkten Lösung von Optimal-Steuerungsproblemen." Diplomarbeit. Heidelberg University, 1995.

[Lei99] D. Leineweber. *Efficient reduced SQP methods for the optimization of chemical processes described by large sparse DAE models*. Vol. 613. Fortschritt-Berichte VDI Reihe 3, Verfahrenstechnik. Düsseldorf: VDI Verlag, 1999.

[Len16] F. Lenders. *trbench: Benchmarking Trust Region Problems*. 2016. URL: https://github.com/felixlen/trbench.

[Ley05] S. Leyffer. "The penalty interior-point method fails to converge." In: *Optimization Methods and Software* 20.4-5 (2005), pp. 559–568. DOI: 10.1080/10556780500140078.

[Ley06] S. Leyffer. "Complementarity constraints as nonlinear equations: Theory and numerical experience." In: *Optimization with Multivalued Mappings: Theory, Applications, and Algorithms*. Ed. by S. Dempe and V. Kalashnikov. Boston, MA: Springer US, 2006, pp. 169–208. ISBN: 978-0-387-34221-4. DOI: 10.1007/0-387-34221-4_9.

[LF03] G. Lin and M. Fukushima. "New Relaxation Method for Mathematical Programs with Complementarity Constraints." In: *Journal of Optimization Theory and Applications* 118.1 (July 2003), pp. 81–116. ISSN: 1573-2878. DOI: 10.1023/A:1024739508603.

[LF05] G. Lin and M. Fukushima. "A Modified Relaxation Scheme for Mathematical Programs with Complementarity Constraints." In: *Annals of Operations Research* 133.1 (Jan. 2005), pp. 63–84. ISSN: 1572-9338. DOI: 10.1007/s10479-004-5024-z.

[LKB17] F. Lenders, C. Kirches, and H. G. Bock. "pySLEQP: A Sequential Linear Quadratic Programming Method Implemented in Python." In: *Modeling, Simulation and Optimization of Complex Processes*. Ed. by H. G. Bock, H. X. Phu, R. Rannacher, and J. P. Schlöder. Springer Verlag, 2017, pp. 103–113. DOI: 10.1007/978-3-319-67168-0_9.

[LKP16] F. Lenders, C. Kirches, and A. Potschka. "trlib: A vector-free implementation of the GLTR method for iterative solution of the trust region problem." In: *Optimization Online* (Nov. 2016). (submitted to Optimization Methods and Software). URL: http://www.optimization-online.org/DB_HTML/2016/11/5724.html.

[LLN06] S. Leyffer, G. López-Calva, and J. Nocedal. "Interior Methods for Mathematical Programs with Complementarity Constraints." In: *SIAM Journal on Optimization* 17.1 (2006), pp. 52–77. DOI: 10.1137/040621065.

[LM07] S. Leyffer and T. Munson. *A Globally Convergent Filter Method for MPECs.* Preprint ANL/MCS-P1457-0907. 9700 South Cass Avenue, Argonne, IL 60439, USA: Mathematics and Computer Science Division, Argonne National Laboratory, Oct. 2007.

[LNP98] M. Lalee, J. Nocedal, and T. Plantenga. "On the implementation of an algorithm for large-scale equality constrained optimization." In: *SIAM Journal on Optimization* 8.3 (1998), pp. 682–706. DOI: 10.1137/S1052623493262993.

[LPR96] Z. Luo, J. Pang, and D. Ralph. *Mathematical Programs with Equlibrium Constraints.* Cambridge: Cambridge University Press, 1996. DOI: 10.1017/CBO9780511983658.

[LPR98] Z.-Q. Luo, J.-S. Pang, and D. Ralph. "Piecewise Sequential Quadratic Programming for Mathematical Programs with Nonlinear Complementarity Constraints." In: *Multilevel Optimization: Algorithms and Applications.* Ed. by A. Migdalas et al. Boston, MA: Springer US, 1998, pp. 209–229. ISBN: 978-1-4613-0307-7. DOI: 10.1007/978-1-4613-0307-7_9.

[LS04] X. Liu and J. Sun. "Generalized stationary points and an interior-point method for mathematical programs with equilibrium constraints." In: *Mathematical Programming, Series B* 101.1 (Sept. 2004), pp. 231–261. ISSN: 1436-4646. DOI: 10.1007/s10107-004-0543-6.

[LSY98] R. Lehoucq, D. Sorensen, and C. Yang. *ARPACK Users' Guide: Solution of Large–Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods.* Society for Industrial and Applied Mathematics (SIAM), 1998. DOI: 10.1137/1.9780898719628.

[Luo+96] Z.-Q. Luo, J.-S. Pang, D. Ralph, and S.-Q. Wu. "Exact penalization and stationarity conditions of mathematical programs with equilibrium constraints." In: *Mathematical Programming* 75.1 (Oct. 1996), pp. 19–76. ISSN: 1436-4646. DOI: 10.1007/BF02592205.

[LW10] A. Logg and G. N. Wells. "DOLFIN: Automated Finite Element Computing." In: *ACM Transactions on Mathematical Software* 37.2 (2010), 20:1–20:28. ISSN: 0098-3500. DOI: 10.1145/1731022.1731030.

[Mar78] N. Maratos. "Exact penalty function algorithms for finite-dimensional and control optimization problems." PhD thesis. London: Imperial College, 1978.

[McC74] G. P. McCormick. *A Minimanual for use of the SUMT computer program and the factorable programming language*. Tech. rep. SOL 74-15. Department of Operations Research, Stanford University, 1974.

[Meh17] P. Mehlitz. "Contributions to complementarity and bilevel programming in Banach spaces." Dissertation. Freiberg University of Mining and Technology, 2017. URL: http://nbn-resolving.de/urn:nbn:de:bsz:105-qucosa-227091.

[MF67] O. Mangasarian and S. Fromovitz. "Fritz John necessary optimality conditions in the presence of equality and inequality constraints." In: *Journal of Mathematical Analysis and Applications* 17 (1967), pp. 37–47. DOI: 10.1016/0022-247X(67)90163-1.

[MGW00] M. F. Murphy, G. H. Golub, and A. J. Wathen. "A Note on Preconditioning for Indefinite Linear Systems." In: *SIAM Journal on Scientific Computing* 21.6 (2000), pp. 1969–1972. DOI: 10.1137/S1064827599355153.

[MLK11] A. Mahajan, S. Leyffer, and C. Kirches. *Solving mixed-integer nonlinear programs by QP diving*. Technical Report ANL/MCS-P2071-0312. (submitted to Optimization Methods and Software). 9700 South Cass Avenue, Argonne, IL 60439, U.S.A.: Mathematics and Computer Science Division, Argonne National Laboratory, Mar. 2011. URL: http://www.optimization-online.org/DB_FILE/2012/03/3409.pdf.

[MR03] D. Q. Mayne and S. Rakovic. "Optimal Control of Constrained Piecewise Affine Discrete-Time Systems." In: *Computational Optimization and Applications* 25 (2003), pp. 167–191. DOI: 10.1023/A:1022905121198.

[MRT06] C. Meyer, A. Rösch, and F. Tröltzsch. "Optimal Control of PDEs with Regularized Pointwise State Constraints." In: *Computational Optimization and Applications* 33.2 (2006), pp. 209–228. ISSN: 1573-2894. DOI: 10.1007/s10589-005-3056-1.

[MS03] B. A. Murtagh and M. A. Saunders. *MINOS 5.51 User's Guide*. Tech. rep. Stanford University, 2003.

[MS64] N. G. Meyers and J. Serrin. "$H = W$." In: *Proceedings of the National Academy of Science* 51 (1964), pp. 1055–1056.

[MS83] J. J. Moré and D. C. Sorensen. "Computing a Trust Region Step." In: *SIAM Journal on Scientific and Statistical Computing* 4.3 (1983), pp. 553–572. DOI: 10.1137/0904038.

[Mun+01] T. S. Munson et al. "The Semismooth Algorithm for Large Scale Complementarity Problems." In: *INFORMS Journal on Computing* 13.4 (2001), pp. 294–311. DOI: 10.1287/ijoc.13.4.294.9734.

[NW06] J. Nocedal and S. Wright. *Numerical Optimization*. Second. ISBN 0-387-30303-0 (hardcover). Berlin Heidelberg New York: Springer Verlag, 2006. DOI: 10.1007/978-0-387-40065-5.

[Obe86] H. J. Oberle. "Numerical solution of minimax optimal control problems by multiple shooting technique." In: *Journal of Optimization Theory and Applications* 50.2 (Aug. 1986), pp. 331–357. ISSN: 1573-2878. DOI: 10.1007/BF00939277.

[OKZ98] J. Outrata, M. Kocvara, and J. Zowe. *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints: Theory, Applications and Numerical Results*. Nonconvex Optimization and Its Applications. Springer US, 1998. ISBN: 9781475728255. DOI: 10.1007/978-1-4757-2825-5.

[Omo89] E. Omojokun. "Trust region algorithms for optimization with nonlinear equality and Inequality restraints." PhD thesis. University of Colorado Boulder, 1989.

[Osb69] M. Osborne. "On shooting methods for boundary value problems." In: *Journal of Mathematical Analysis and Applications* 27 (1969), pp. 417–433. DOI: 10.1016/0022-247X(69)90059-6.

[Out00] J. V. Outrata. "A Generalized Mathematical Program with Equilibrium Constraints." In: *SIAM Journal on Control and Optimization* 38.5 (2000), pp. 1623–1638. DOI: 10.1137/S0363012999352911.

[Out99] J. V. Outrata. "Optimality Conditions for a Class of Mathematical Programs with Equilibrium Constraints." In: *Mathematics of Operations Research* 24.3 (1999), pp. 627–644. ISSN: 0364765X, 15265471. DOI: 10.1287/moor.24.3.627.

[PBS09] A. Potschka, H. Bock, and J. Schlöder. "A minima tracking variant of semi-infinite programming for the treatment of path constraints within direct solution of optimal control problems." In: *Optimization Methods and Software* 24.2 (2009), pp. 237–252. DOI: 10.1080/10556780902753098.

[Pes89] H. J. Pesch. "Real-time computation of feedback controls for constrained optimal control problems. part 2: A correction method based on multiple shooting." In: *Optimal Control Applications and Methods* 10.2 (1989), pp. 147–171. ISSN: 1099-1514. DOI: 10.1002/oca.4660100206.

[Pes94] H. Pesch. "A practical guide to the solution of real-life optimal control problems." In: *Control and Cybernetics* 23 (1994), pp. 7–60. URL: http://control.ibspan.waw.pl:3000/contents/export?filename=1994-1-2-03_pesch.pdf.

[Pet73] D. Peterson. "A review of constraint qualifications in finite dimensional spaces." In: *SIAM Review* 15.3 (July 1973), pp. 639–654. DOI: 10.1137/1015075.

[PF99] J.-S. Pang and M. Fukushima. "Complementarity Constraint Qualifications and Simplified B-Stationarity Conditions for Mathematical Programs with Equilibrium Constraints." In: *Computational Optimization and Applications* 13.1 (Apr. 1999), pp. 111–136. ISSN: 1573-2894. DOI: 10.1023/A:1008656806889.

[Pli81] K. Plitt. "Ein superlinear konvergentes Mehrzielverfahren zur direkten Berechnung beschränkter optimaler Steuerungen." Diplomarbeit. Rheinische Friedrich–Wilhelms–Universität Bonn, 1981. URL: https://bonnus.ulb.uni-bonn.de/SummonRecord/FETCH-bonn_catalog_21293522.

[Pon+61] L. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mischenko. *Matematicheskaya teoriya optimalnykh protsessov*. Moscow: Fizmatgiz, 1961.

[Pot06] A. Potschka. "Handling Path Constraints in a Direct Multiple Shooting Method for Optimal Control Problems." Diplomarbeit. Heidelberg University, 2006.

[Pot11] A. Potschka. "A direct method for the numerical solution of optimization problems with time-periodic PDE constraints." Dissertation. Heidelberg University, 2011. DOI: 10.11588/heidok.00012993.

[Pot16] A. Potschka. "Backward Step Control for Global Newton-Type Methods." In: *SIAM Journal on Numerical Analysis* 54.1 (2016), pp. 361–387. DOI: 10.1137/140968586.

[PP09] H. Pesch and M. Plail. "The Maximum Principle of optimal control: A history of ingenious ideas and missed opportunities." In: *Control and Cybernetics* 38.4A (2009), pp. 973–995. URL: http://control.ibspan.waw.pl:3000/contents/export?filename=2009-4-03_pesch_plail.pdf.

[PR03] G. Pannocchia and J. Rawlings. "Disturbance Models for Offset–Free Model-Predictive Control." In: *AIChE Journal* 49 (2003), pp. 426–437. DOI: 10.1002/aic.690490213.

[PR81] B. N. Parlett and J. K. Reid. "Tracking the Progress of the Lanczos Algorithm for Large Symmetric Eigenproblems." In: *IMA Journal of Numerical Analysis* 1.2 (1981), pp. 135–155. DOI: 10.1093/imanum/1.2.135.

[PS75] C. Paige and M. Saunders. "Solutions of sparse indefinite systems of linear equations." In: *SIAM Journal on Numerical Analysis* 12.4 (1975), pp. 617–629. DOI: 10.1137/0712047.

[PV14] M. Palladino and R. B. Vinter. "Minimizers That Are Not Also Relaxed Minimizers." In: *SIAM Journal on Control and Optimization* 52.4 (2014), pp. 2164–2179. DOI: 10.1137/130909627.

[Ral94] D. Ralph. "Global Convergence of Damped Newton's Method for Nonsmooth Equations via the Path Search." In: *Mathematics of Operations Research* 19.2 (1994), pp. 352–389. DOI: 10.1287/moor.19.2.352.

[RB03] A. Raghunathan and L. Biegler. "Mathematical programs with equilibrium constraints (MPECs) in process engineering." In: *Computers & Chemical Engineering* 27 (2003), pp. 1381–1392. DOI: 10.1016/S0098-1354(03)00092-9.

[RDB04] A. Raghunathan, M. Diaz, and L. Biegler. "An MPEC formulation for dynamic optimization of distillation operations." In: *Computers & Chemical Engineering* 28 (2004), pp. 2037–2052. DOI: 10.1016/j.compchemeng.2004.03.015.

[RDW10] T. Rees, H. S. Dollar, and A. J. Wathen. "Optimal Solvers for PDE-Constrained Optimization." In: *SIAM Journal on Scientific Computing* 32.1 (2010), pp. 271–298. DOI: 10.1137/080727154.

[Ren86] J. Renfro. "Computational studies in the optimization of systems described by differential/algebraic equations." PhD thesis. University of Houston, 1986.

[Roc70] R. Rockafellar. *Convex Analysis*. Princeton University Press, 1970. ISBN: 9781400873173.

[ROL17] M. Ringkamp, S. Ober-Blöbaum, and S. Leyendecker. "On the time transformation of mixed integer optimal control problems using a consistent fixed integer control function." In: *Mathematical Programming, Series A* 161.1 (Jan. 2017), pp. 551–581. ISSN: 1436-4646. DOI: 10.1007/s10107-016-1023-5.

[RS72] R. Russell and L. Shampine. "A collocation method for boundary value problems." In: *Numerische Mathematik* 19.1 (1972), pp. 1–28. DOI: 10.1007/BF01395926.

[RSS01] M. Rojas, S. A. Santos, and D. C. Sorensen. "A New Matrix-Free Algorithm for the Large-Scale Trust-Region Subproblem." In: *SIAM Journal on Optimization* 11.3 (2001), pp. 611–646. DOI: 10.1137/S105262349928887X.

[RSS08] M. Rojas, S. A. Santos, and D. C. Sorensen. "Algorithm 873: LSTRS: MATLAB Software for Large-scale Trust-region Subproblems and Regularization." In: *ACM Transactions on Mathematical Software* 34.2 (2008), 11:1–11:28. ISSN: 0098-3500. DOI: 10.1145/1326548.1326553.

[Rud66] W. Rudin. *Real and Complex Analysis*. McGraw-Hill, 1966.

[RW04] D. Ralph and S. J. Wright. "Some properties of regularization and penalization schemes for MPECs." In: *Optimization Methods and Software* 19 (2004), pp. 527–556. DOI: 10.1080/10556780410001709439.

[RW97] F. Rendl and H. Wolkowicz. "A semidefinite framework for trust region subproblems with applications to large scale minimization." In: *Mathematical Programming* 77.1 (Apr. 1997), pp. 273–299. ISSN: 1436-4646. DOI: 10.1007/BF02614438.

[Sag06] S. Sager. "Numerical methods for mixed–integer optimal control problems." Dissertation. Heidelberg University, 2006. URL: http://mathopt.de/PUBLICATIONS/Sager2005.pdf.

[SB02] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. New York, NY: Springer New York, 2002. DOI: 10.1007/978-0-387-21738-3_7.

[SBD12] S. Sager, H. Bock, and M. Diehl. "The Integer Approximation Error in Mixed-Integer Optimal Control." In: *Mathematical Programming, Series A* 133.1–2 (2012), pp. 1–23. DOI: 10.1007/s10107-010-0405-3.

[Sch01] S. Scholtes. "Convergence properties of a regularization scheme for mathematical programs with complementarity constraints." In: *SIAM Journal on Optimization* 11 (2001), pp. 918–936. DOI: 10.1137/S1052623499361233.

[Sch02] S. Scholtes. *Combinatorial Structures in Nonlinear Programming*. Tech. rep. University of Cambridge, 2002. URL: http://www.optimization-online.org/DB%5C_HTML/2002/05/477.html.

[Sch04] S. Scholtes. "Nonconvex Structures in Nonlinear Programming." In: *Operations Research* 52.3 (May 2004), pp. 368–383. DOI: 10.1287/opre.1030.0102.

[Sch05] A. Schäfer. "Efficient reduced Newton-type methods for solution of large-scale structured optimization problems with application to biological and chemical processes." Dissertation. Universität Heidelberg, 2005. DOI: 10.11588/heidok.00005264.

[Sch88] J. Schlöder. *Numerische Methoden zur Behandlung hochdimensionaler Aufgaben der Parameteridentifizierung*. Ed. by E. Brieskorn et al. Vol. 187. Bonner Mathematische Schriften. 1988.

[Sch90] V. Schulz. "Ein effizientes Kollokationsverfahren zur numerischen Behandlung von Mehrpunktrandwertaufgaben in der Parameteridentifizierung und Optimalen Steuerung." Diplomarbeit. University of Augsburg, 1990.

[Sch96] V. Schulz. "Reduced SQP methods for large-scale optimal control problems in DAE with application to path planning problems for satellite mounted robots." Dissertation. Heidelberg University, 1996.

[Seb+10] D. Seborg, D. Mellichamp, T. Edgar, and F. Doyle. *Process Dynamics and Control*. John Wiley & Sons, 2010. ISBN: 9780470128671.

[SEM89] D. Seborg, T. Edgar, and D. Mellichamp. *Process Dynamics and Control*. Chemical Engineering Series Bd. 1. Wiley, 1989. ISBN: 9780471863892.

[Sha70] D. F. Shanno. "Conditioning of Quasi–Newton methods for function minimization." In: *Mathematics of Computation* 24.111 (July 1970), pp. 647–656. DOI: 10.2307/2004840.

[Sha80] M. E. Shayan. "A methodology for comparing algorithms and a method of computing m^{th} order directional derivatives based on factorable programming." PhD thesis. George Washington University, 1980. URL: https://findit.library.gwu.edu/item/2357287.

[SK08] M. Szymkat and A. Korytowski. "The Method of Monotone Structural Evolution for Dynamic Optimization of Switched Systems." In: *Proceedings of the 47th IEEE Conference on Decision and Control*. 2008. DOI: 10.1109/CDC.2008.4739106.

[Sor97] D. C. Sorensen. "Minimization of a Large-Scale Quadratic FunctionSubject to a Spherical Constraint." In: *SIAM Journal on Optimization* 7.1 (1997), pp. 141–161. DOI: 10.1137/S1052623494274374.

[Spe80] B. Speelpenning. "Compiling fast partial derivatives of functions given by algorithms." PhD thesis. University of Illinois at Urbana-Champaign, 1980.

[SRB09] S. Sager, G. Reinelt, and H. Bock. "Direct methods with maximal lower bound for mixed-integer optimal control problems." In: *Mathematical Programming, Series A* 118.1 (2009), pp. 109–149. DOI: 10.1007/s10107-007-0185-6.

[SS00] H. Scheel and S. Scholtes. "Mathematical programs with complementarity constraints: Stationarity, optimality and sensitivity." In: *Mathematics of Operations Research* 25 (2000), pp. 1–22. DOI: 10.1287/moor.25.1.1.15213.

[SS78] R. Sargent and G. Sullivan. "The development of an efficient optimal control package." In: *Proceedings of the 8th IFIP Conference on Optimization Techniques (1977), Part 2*. Ed. by J. Stoer. Heidelberg: Springer, 1978. DOI: 10.1007/BFb0006520.

[SS99] S. Scholtes and M. Stöhr. "Exact Penalization of Mathematical Programs with Equilibrium Constraints." In: *SIAM Journal on Control and Optimization* 37.2 (1999), pp. 617–652. DOI: 10.1137/S0363012996306121.

[Ste12] O. Stein. "Lifting mathematical programs with complementarity constraints." In: *Mathematical Programming, Series A* 131.1 (Feb. 2012), pp. 71–94. ISSN: 1436-4646. DOI: 10.1007/s10107-010-0345-y.

[Ste83] T. Steihaug. "The Conjugate Gradient Method and Trust Regions in Large Scale Optimization." In: *SIAM Journal on Numerical Analysis* 20.3 (1983), pp. 626–637. DOI: 10.1137/0720042.

[Ste94] M. Steinbach. "A structured interior point SQP method for nonlinear optimal control problems." In: *Computational Optimal Control*. Ed. by R. Bulirsch and D. Kraft. Vol. 115. International Series of Numerical Mathematics. ISBN 0-8176-5015-6. Basel Boston Berlin: Birkhäuser, 1994, pp. 213–222. DOI: 10.1007/978-3-0348-8497-6_17.

[Ste95] M. Steinbach. "Fast recursive SQP methods for large-scale optimal control problems." Dissertation. Heidelberg University, 1995.

[Ste96] M. Steinbach. "Structured interior point SQP methods in optimal control." In: *Zeitschrift für Angewandte Mathematik und Mechanik* 76.S3 (1996), pp. 59–62.

[Stö00] M. Stöhr. "Nonsmooth trust region methods and their applications to mathematical programs with equilibrium constraints." Dissertation. University of Karlsruhe, 2000.

[Str93] O. Stryk. "Numerical solution of optimal control problems by direct collocation." In: *Optimal Control: Calculus of Variations, Optimal Control Theory and Numerical Methods*. Vol. 111. Bulirsch et al., 1993, pp. 129–143. DOI: 10.1007/978-3-0348-7539-4_10.

[Str95] O. Stryk. "Numerische Lösung optimaler Steuerungsprobleme: Diskretisierung, Parameteroptimierung und Berechnung der adjungierten Variablen." Dissertation. TU Munich, 1995.

[Su05] C.-L. Su. "Equilibrium Problems with Equilibrium Constraints: Stationarities, Algorithms, and Applications." PhD thesis. Stanford University, 2005. ISBN: 0-542-29676-4. URL: https://web.stanford.edu/group/SOL/dissertations/clsu-thesis.pdf.

[SU10] S. Steffensen and M. Ulbrich. "A New Relaxation Scheme for Mathematical Programs with Equilibrium Constraints." In: *SIAM Journal on Optimization* 20.5 (2010), pp. 2504–2539. DOI: 10.1137/090748883.

[SZ09] S. Subchan and R. Zbikowski. *Computational Optimal Control: Tools and Practice*. Wiley, 2009. ISBN: 9780470747681. DOI: 10.1002/9780470747674.

[THE75] T. Tsang, D. Himmelblau, and T. Edgar. "Optimal control via collocation and non-linear programming." In: *International Journal on Control* 21 (1975), pp. 763–768. DOI: 10.1080/00207177508922030.

[Toi81] P. L. Toint. "Towards an Efficient Sparsity Exploiting Newton Method for Minimization." In: *Sparse Matrices and Their Uses*. Ed. by I. S. Duff. London, England: Academic Press, 1981, pp. 57–88.

[Toi88] P. Toint. "Global Convergence of a a of Trust-Region Methods for Nonconvex Minimization in Hilbert Space." In: *IMA Journal of Numerical Analysis* 8.2 (1988), pp. 231–252. DOI: 10.1093/imanum/8.2.231.

[UU00] M. Ulbrich and S. Ulbrich. "Superlinear Convergence of Affine-Scaling Interior-Point Newton Methods for Infinite-Dimensional Nonlinear Problems with Pointwise Bounds." In: *SIAM Journal on Control and Optimization* 38.6 (2000), pp. 1938–1984. DOI: 10.1137/S0363012997325915.

[Vai65] G. Vainikko. "On the stability and convergence of the collocation method." In: *Differentsial'nye Uravneniya* 1 (1965). (In Russian. Translated in Differential Equations, 1 (1965), pp. 186–194), pp. 244–254.

[Vel03] V. Veliov. *Relaxation of Euler-Type Discrete-Time Control System*. Tech. rep. 273. TU-Wien, 2003. DOI: 10.1007/978-3-319-26520-9_14.

[Vel05] V. M. Veliov. "Error analysis of discrete approximations to bang-bang optimal control problems: the linear case." In: *Control and Cybernetics* 34.3 (2005), pp. 967–982. URL: http://control.ibspan.waw.pl:3000/contents/export?filename=2005-3-17_veliov.pdf.

[Vin10] R. Vinter. *Optimal Control*. Modern Birkhäuser Classics. Birkhäuser Boston, 2010. ISBN: 9780817680862. DOI: 10.1007/978-0-8176-8086-2.

[Wäc02] A. Wächter. "An Interior Point Algorithm for Large-Scale Nonlinear Optimization with Applications in Process Engineering." PhD thesis. Carnegie Mellon University, 2002.

[Wac13] G. Wachsmuth. "On LICQ and the uniqueness of Lagrange multipliers." In: *Operations Research Letters* 41 (Jan. 2013), pp. 78–80. DOI: 10.1016/j.orl.2012.11.009.

[Wac15] G. Wachsmuth. "Mathematical Programs with Complementarity Constraints in Banach Spaces." In: *Journal of Optimization Theory and Applications* 166.2 (Aug. 2015), pp. 480–507. ISSN: 1573-2878. DOI: 10.1007/s10957-014-0695-3.

[Wac16] G. Wachsmuth. "Optimization problems with complementarity constraints in infinite-dimensional spaces." Habilitationsschrift. Universität Chemnitz, 2016. URL: http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-227446.

[WB06] A. Wächter and L. Biegler. "On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming." In: *Mathematical Programming, Series A* 106.1 (2006), pp. 25–57. DOI: 10.1007/s10107-004-0559-y.

[WBD06] L. Wirsching, H. Bock, and M. Diehl. "Fast NMPC of a chain of masses connected by springs." In: *Proceedings of the 2006 IEEE International Conference on Control Applications (CCA)*. 2006, pp. 591–596. DOI: 10.1109/CACSD-CCA-ISIC.2006.4776712.

[WKG05] A. Walther, A. Kowarz, and A. Griewank. *ADOL-C: A Package for the Automatic Differentiation of Algorithms Written in C/C++*. Tech. rep. Institute of Scientific Computing, Technical University Dresden, 2005.

[Wlo71] J. Wloka. *Funktionalanalysis und Anwendungen*. de Gruyter Lehrbuch. Walter de Gruyter, Berlin-New York, 1971.

[WVD14] A. Wynn, M. Vukov, and M. Diehl. "Convergence Guarantees for Moving Horizon Estimation Based on the Real-Time Iteration Scheme." In: *IEEE Transactions on Automatic Control* 59.8 (2014), pp. 2215–2221. ISSN: 0018-9286. DOI: 10.1109/TAC.2014.2298984.

[Ye00] J. Ye. "Constraint Qualifications and Necessary Optimality Conditions for Optimization Problems with Variational Inequality Constraints." In: *SIAM Journal on Optimization* 10.4 (2000), pp. 943–962. DOI: 10.1137/S105262349834847X.

[Ye05] J. Ye. "Necessary and sufficient optimality conditions for mathematical programs with equilibrium constraints." In: *Journal on Mathematical Analysis and Applications* 307 (2005), pp. 350–369. DOI: S0022247X04008741.

[Ye95] J. Ye. "Necessary Conditions for Bilevel Dynamic Optimization Problems." In: *SIAM Journal on Control and Optimization* 33.4 (1995), pp. 1208–1223. DOI: 10.1137/S0363012993249717.

[YMP] B. Yu, J. E. Mitchell, and J.-S. Pang. *Solving Linear Programs with Complementarity Constraints using Branch-and-Cut*. Tech. rep.

[Yos78] K. Yosida. *Functional analysis*. Springer-Verlag, 1978. DOI: 10.1007/978-3-642-61859-8.

[You37] L. Young. *Generalized Curves and the Existence of an Attained Absolute Minimum in the Calculus of Variations*. 1937.

[Yua85] Y. Yuan. "Conditions for convergence of trust region algorithms for nonsmooth optimization." In: *Mathematical Programming* 31.2 (June 1985), pp. 220–228. ISSN: 1436-4646. DOI: 10.1007/BF02591750.

[YY97] J. J. Ye and X. Y. Ye. "Necessary Optimality Conditions for Optimization Problems with Variational Inequality Constraints." In: *Mathematics of Operations Research* 22.4 (1997), pp. 977–997. ISSN: 0364765X, 15265471.

[ZL01] J. Zhang and G. Liu. "A New Extreme Point Algorithm and Its Application in PSQP Algorithms for Solving Mathematical Programs with Linear Complementarity Constraints." In: *Journal of Global Optimization* 19.4 (Apr. 2001), pp. 345–361. ISSN: 1573-2916. DOI: 10.1023/A:1011226232107.

[ZLB07] V. Zavala, C. Laird, and L. Biegler. "A fast computational framework for large-scale moving-horizon estimation." In: *Proceedings of the 8th International Symposium on Dynamics and Control of Process Systems (DYCOPS)*. Cancun, Mexico, 2007. DOI: 10.3182/20070606-3-MX-2915.00122.