Recent Advances in Optimization and its Applications in Engineering

Moritz Diehl
François Glineur
Elias Jarlebring
Wim Michiels
Editors

Springer

Recent Advances in Optimization and its Applications in Engineering

Moritz Diehl · François Glineur · Elias Jarlebring ·
Wim Michiels
Editors

Recent Advances in Optimization and its Applications in Engineering

Springer

Editors

Prof. Dr. Moritz Diehl
Katholieke Universiteit Leuven
Dept. Electrical Engineering (ESAT)
and Optimization in Engineering
Center (OPTEC)
Kasteelpark Arenberg 10
3001 Leuven
Belgium
moritz.diehl@esat.kuleuven.be

Prof. Dr. François Glineur
CORE-UCL
Voie du Roman Pays 34
1348 Louvain-la-Neuve
Belgium
glineur@core.ucl.ac.be

Dr. Elias Jarlebring
Katholieke Universiteit Leuven
Department Computerwetenschappen
and Optimization in Engineering
Center (OPTEC)
Celestijnenlaan 200 A
3001 Heverlee
Belgium
Elias.Jarlebring@cs.kuleuven.be

Prof. Dr. Wim Michiels
Katholieke Universiteit Leuven
Department Computerwetenschappen
and Optimization in Engineering
Center (OPTEC)
Celestijnenlaan 200 A
3001 Heverlee
Belgium
Wim.Michiels@cs.kuleuven.be

ISBN 978-3-642-12597-3 e-ISBN 978-3-642-12598-0
DOI 10.1007/978-3-642-12598-0
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2010933976

Cover design: WMXDesign GmbH, Heidelberg

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The field of mathematical optimization combines a rich body of fundamental theory with a powerful collection of computational methods and a variety of exciting applications. The field evolves fast, and the last decade has been characterized by major developments in optimization theory, improvements in numerical methods also benefiting from the parallel developments in computational hardware, and emergence of novel applications.

Many of these novel applications belong to engineering, as globalization of the economy, increased competition, limited natural resources and ever stronger environmental constraints call for better performance of industrial products and higher productivity of production processes. This new environment poses real challenges to engineering research, design and development. Adequate translation of the conflicting objectives into optimization problems, availability of efficient and reliable numerical algorithms and correct interpretation of the results are essential in addressing these challenges. We are convinced that significant advances can be achieved by cross-fertilization and integration of the recent developments in the mathematical optimization community on the one hand and the different engineering communities on the other hand.

The present volume contains a careful selection of articles on recent advances in optimization theory, numerical methods, and their applications in engineering. The authors met at the 14th Belgian-French-German Conference on Optimization (BFG09) that took place in Leuven in September 14–18, 2009. The conference was organized by the Optimization in Engineering Center OPTEC at the Katholieke Universiteit Leuven (K.U.Leuven), in collaboration with the Center for Operations Research and Econometrics (CORE) at the Université Catholique de Louvain (UCLouvain).

In the context sketched above, BFG09's special topic was "Optimization in Engineering", aiming at deepening the contacts between engineering optimizers and mathematical optimizers. We believe that this aim has been reached and that it is well reflected in the present volume, which is divided into the following chapters: convex optimization, nonlinear optimization, optimization on manifolds, optimal control, model predictive control, PDE-constrained optimization and engineering applications of optimization. We want in particular to point out the overview articles by three of the invited speakers at the BFG09 (M. Dür, P.-A. Absil, J.-B. Caillau) as well as by the winners of the best talk and best poster prizes (A. Potschka, M. Ishteva). These overview articles can be found at the beginning of their respective chapters.

This book would not have been possible without the substantial help of many anonymous reviewers whom we want to thank at this place. Acceptance decisions for each submitted article were based on at least two reviews, which also helped the authors to further improve their contributions. We also gratefully acknowledge financial support by the Fonds Wetenschappelijk Onderzoek – Vlaanderen (FWO) and the Fonds de la Recherche Scientifique (F.R.S.-FNRS).

We are particularly indebted to Jacqueline De bruyn and Ioanna Stamati for the numerous hours spent communicating with the authors of this volume on technical questions, and in particular to Ioanna Stamati for compiling the final LaTeX manuscript. Last but not least, we want to thank the staff at Springer, in particular Birgit Kollmar-Thoni and Eva Hestermann-Beyerle, for their efficient and professional support, including the design of an innovative cover, which features a word cloud (obtained from the web site wordle.net) reflecting importance of the most frequent terms used throughout the book.

We wish all readers of this book the same pleasure we had in compiling it!

Leuven and Louvain-La-Neuve,
Belgium, July 2010

Moritz Diehl
François Glineur
Elias Jarlebring
Wim Michiels

Contents

Part III Optimization on Manifolds

Part IV Optimal Control

Part V Model Predictive Control

Part VI PDE-Constrained Optimization

Part VII Engineering Applications of Optimization

Part I

Convex Optimization

Copositive Programming – a Survey

Mirjam Dür

Johann Bernoulli Institute of Mathematics and Computer Science, University of Groningen, P.O. Box 407, 9700 AK Groningen, The Netherlands. M.E.Dur@rug.nl

Summary. Copositive programming is a relatively young field in mathematical optimization. It can be seen as a generalization of semidefinite programming, since it means optimizing over the cone of so called copositive matrices. Like semidefinite programming, it has proved particularly useful in combinatorial and quadratic optimization. The purpose of this survey is to introduce the field to interested readers in the optimization community who wish to get an understanding of the basic concepts and recent developments in copositive programming, including modeling issues and applications, the connection to semidefinite programming and sum-of-squares approaches, as well as algorithmic solution approaches for copositive programs.

1 Introduction

A copositive program is a linear optimization problem in matrix variables of the following form:

$$\begin{aligned} &\min \langle C, X\rangle \\ &\text{s.t. } \langle A_i, X\rangle = b_i \quad (i = 1, \dots, m), \\ &\qquad X \in \mathcal{C}, \end{aligned} \tag{1}$$

where $\mathcal{C}$ is the cone of so-called copositive matrices, that is, the matrices whose quadratic form takes nonnegative values on the nonnegative orthant $\mathbb{R}^n_+$:

$$\mathcal{C} = \{A \in \mathcal{S} : x^T Ax \geq 0 \text{ for all } x \in \mathbb{R}^n_+\}$$

(here $\mathcal{S}$ is the set of symmetric $n \times n$ matrices, and the inner product of two matrices in (1) is $\langle A, B\rangle := \operatorname{trace}(BA) = \sum_{i,j=1}^n a_{ij}b_{ij}$, as usual). Obviously, every positive semidefinite matrix is copositive, and so is every entrywise nonnegative matrix, but the copositive cone is significantly larger than both the semidefinite and the nonnegative matrix cones.

Interpreting (1) as the primal program, one can associate a corresponding dual program which is a maximization problem over the dual cone. For an arbitrary given cone $\mathcal{K} \subseteq \mathcal{S}$, the dual cone $\mathcal{K}^*$ is defined as

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_1, © Springer-Verlag Berlin Heidelberg 2010

$$\mathcal{K}^* := \{A \in \mathcal{S} : \langle A, B\rangle \geq 0 \text{ for all } B \in \mathcal{K}\}.$$

In contrast to the semidefinite and nonnegative matrix cones, the cone $\mathcal{C}$ is not selfdual. It can be shown (see e.g. [6]) that $\mathcal{C}^*$ is the cone of so-called completely positive matrices

$$\mathcal{C}^* = \operatorname{conv}\{xx^T : x \in \mathbb{R}^n_+\}.$$

Using this, the dual of (1) can be derived through the usual Lagrangian approach and is easily seen to be

$$\begin{aligned} &\max \sum_{i=1}^m b_i y_i \\ &\text{s.t. } C - \sum_{i=1}^m y_i A_i \in \mathcal{C}^*,\ y_i \in \mathbb{R}. \end{aligned} \tag{2}$$

Since both $\mathcal{C}$ and $\mathcal{C}^*$ are convex cones, (1) and (2) are convex optimization problems. KKT optimality conditions hold if Slater's condition is satisfied, as shown by [28], and imposing a constraint qualification guarantees strong duality, i.e., equality of the optimal values of (1) and (2). The most common constraint qualification is to assume that both problems are feasible and one of them strictly feasible (meaning that there exists a strictly feasible point, i.e., a solution to the linear constraints in the interior of the cone).

Copositive programming is closely related to quadratic and combinatorial optimization. We illustrate this connection by means of the standard quadratic problem

$$(\text{StQP}) \qquad \begin{aligned} &\min x^T Q x \\ &\text{s.t. } e^T x = 1, \\ &\qquad x \geq 0, \end{aligned}$$

where e denotes the all-ones vector. This optimization problem asks for the minimum of a (not necessarily convex) quadratic function over the standard simplex. Easy manipulations show that the objective function can be written as $x^T Q x = \langle Q, xx^T\rangle$. Analogously the constraint $e^T x = 1$ transforms to $\langle E, xx^T\rangle = 1$, with $E = ee^T$. Hence, the problem

$$\begin{aligned} &\min \langle Q, X\rangle \\ &\text{s.t. } \langle E, X\rangle = 1, \\ &\qquad X \in \mathcal{C}^* \end{aligned} \tag{3}$$

is obviously a relaxation of (StQP). Since the objective is now linear, an optimal solution must be attained in an extremal point of the convex feasible set. It can be shown that these extremal points are exactly the rank-one matrices xx^T with $x \geq 0$ and $e^T x = 1$. Together, these results imply that (3) is in fact an exact reformulation of (StQP).

The standard quadratic problem is an NP-hard optimization problem, since the maximum clique problem can be reduced to an (StQP). Indeed, denoting by $\omega(G)$ the clique number of a graph G and by A_G its adjacency matrix, Motzkin and Straus [43] showed that

$$\frac{1}{\omega(G)} = \min\{x^T(E - A_G)x : e^T x = 1, x \geq 0\}. \tag{4}$$

Nevertheless, (3) is a convex formulation of this NP-hard problem. This shows that NP-hard convex optimization problems do exist. The complexity has moved entirely into the cone-constraint $X \in \mathcal{C}^*$. It is known that testing whether a given matrix is in $\mathcal{C}$ is co-NP-complete (cf. [44]). Consequently, it is not tractable to do a line-search in $\mathcal{C}$. The cones $\mathcal{C}$ and $\mathcal{C}^*$ do allow self-concordant barrier functions (see [46]), but these functions can not be evaluated in polynomial time. Thus, the classical interior point methodology does not work. Optimizing over either $\mathcal{C}$ or $\mathcal{C}^*$ is thus NP-hard, and restating a problem as an optimization problem over one of these cones does not resolve the difficulty of that problem. However, studying properties of $\mathcal{C}$ and $\mathcal{C}^*$ and using the conic formulations of quadratic and combinatorial problems does provide new insights and also computational improvements.

Historical remarks

The concept of copositivity seems to go back to Motzkin [42] in the year 1952. Since then, numerous papers on both copositivity and complete positivity have emerged in the linear algebra literature, see [6] or [36] for surveys. Using these cones in optimization has been studied only in the last decade.

An early paper relating the solution of a certain quadratic optimization problem to copositivity is Preisig [52] from 1996. Preisig describes properties and derives an algorithm for what we would now call the dual problem of (3) with E replaced by a strictly copositive matrix B. However, he just analyzes this particular problem and does not provide the conic programming framework outlined above. It seems that his paper has been widely ignored by the optimization community.

Quist et al. [53] suggested in 1998 that semidefinite relaxations of quadratic problems may be tightened by looking at the copositive cone. They were the first to formulate problems with the conic constraints $X \in \mathcal{C}$ and $X \in \mathcal{C}^*$.

Bomze et al. [11] were the first to establish an equivalent copositive formulation of an NP-hard problem, namely the standard quadratic problem. Their paper from 2000 also coined the term "copositive programming".

Since [11] appeared, a number of other quadratic and combinatorial problems have been shown to admit an exact copositive reformulation. Although these formulations remain NP-hard, they have inspired better bounds than previously available. Through sum-of-squares approximations (cf. Section 5 below) they have opened a new way to solve these problems. Finally, new solution algorithms for copositive and completely positive problems have been developed and proved very successful in some settings, as we describe in Section 6.

2 Applications

Binary quadratic problems

We have seen in Section 1 that the standard quadratic problem can be rewritten as a completely positive program. This can be extended to so-called multi-StQPs, where one seeks to optimize a quadratic form over the cartesian product of simplices, see [15].

Burer [19] showed the much more general result that every quadratic problem with linear and binary constraints can be rewritten as such a problem. More precisely, he showed that a quadratic binary problem of the form

$$\begin{aligned} \min\ & x^TQx + 2c^Tx \\ \text{s.t. } & a_i^Tx = b_i \quad (i = 1,\dots,m) \\ & x \geq 0 \\ & x_j \in \{0,1\} \quad (j \in B) \end{aligned} \tag{5}$$

can equivalently be reformulated as the following completely positive problem:

$$\begin{aligned} \min\ & \langle Q, X\rangle + 2c^Tx \\ \text{s.t. } & a_i^Tx = b_i \quad (i = 1,\dots,m) \\ & \langle a_ia_i^T, X\rangle = b_i^2 \quad (i = 1,\dots,m) \\ & x_j = X_{jj} \quad (j \in B) \\ & \begin{pmatrix} 1 & x \\ x & X \end{pmatrix} \in \mathcal{C}^*, \end{aligned}$$

provided that (5) satisfies the so-called key condition, i.e., $a_i^Tx = b_i$ for all i and $x \geq 0$ implies $x_j \leq 1$ for all $j \in B$. As noted by Burer, this condition can be enforced without loss of generality.

It is still an open question whether problems with general quadratic constraints can similarly be restated as completely positive problems. Only special cases like complementarity constraints have been solved [19]. For a comment on Burer's result see [13]. Natarajan et al. [45] consider (5) in the setting where $Q = 0$ and c is a random vector, and derive a completely positive formulation for the expected optimal value.

Fractional quadratic problems

Consider a matrix A whose quadratic form x^TAx does not have zeros in the standard simplex, i.e., consider without loss of generality a strictly copositive matrix A. Preisig [52] observed that then the problem of maximizing the ratio of two quadratic forms over the standard simplex

$$\min\left\{\frac{x^TQx}{x^TAx} : e^Tx = 1, x \geq 0\right\}$$

is equivalent to

$$\min\{x^TQx : x^TAx = 1, x \geq 0\}$$

and hence, by similar arguments as used to derive (3), is equivalent to the completely positive program

$$\min\{\langle Q, X\rangle : \langle A, X\rangle = 1, x \in \mathcal{C}^*\}$$

For a thorough discussion, see also [16].

Combinatorial problems

For the problem of determining the **clique number** $\omega(G)$ of a graph G, we can combine the Motzkin-Straus formulation (4) with the completely positive formulation (3) of the standard quadratic problem. Taking the dual of that problem, we arrive at

$$\tfrac{1}{\omega(G)} = \max\{\lambda : \lambda(E - A_G) - E \in \mathcal{C}\}.$$

Using a somewhat different approach, De Klerk and Pasechnik [23] derive the following formulation for the **stability number** $\alpha(G)$:

$$\alpha(G) = \min\{\lambda : \lambda(I + A_G) - E \in \mathcal{C}\}$$

(I the identity matrix), or, in the dual formulation,

$$\alpha(G) = \max\{\langle E, X\rangle : \langle A_G + I, X\rangle = 1, X \in \mathcal{C}^*\}.$$

The last formulation can be seen as a strengthening of the Lovász ϑ number, which is obtained by optimizing over the cone $\mathcal{S}^+ \cap \mathcal{N}$ of entrywise nonnegative and positive semidefinite matrices instead of $\mathcal{C}^*$ in the above problem.

Dukanovic and Rendl [26] introduce a related copositivity-inspired strengthening of the Lovász ϑ number toward the chromatic number of G, which is shown to be equal to the fractional chromatic number.

For the **chromatic number** $\chi(G)$ of a graph G with n nodes, a copositive formulation has been found by Gvozdenović and Laurent in [30]:

$$\begin{aligned}\chi(G) = \max\ & y \\ \text{s. t. }\ & \tfrac{1}{n^2}(ty)E + z(n(I + A_{G_t})E) \in \mathcal{C} \qquad t = 1, \ldots, n \\ & y, z \in \mathbb{R}.\end{aligned}$$

where A_{G_t} denotes the adjacency matrix of the graph G_t, the cartesian product of the graphs K_t (the complete graph on t nodes) and G. This product graph G_t has node set $V(K_t) \times V(G) = \bigcup_{p=1}^{t} V_p$, where $V_p := \{pi : i \in V(G)\}$. An edge (pi, qj) is present in G_t if ($p \neq q$ and $i = j$) or if ($p = q$ and (ij) is an edge in G).

A completely positive formulation of the related problem of computing the **fractional chromatic number** can be found in [26].

A completely positive formulation for the **quadratic assignment problem** (QAP) was developed in [50]. Introducing it requires some notation: let A, B, C be the matrices describing the QAP instance. $B \otimes A$ denotes the Kronecker product of B and A, i.e., the $n^2 \times n^2$ matrix $(b_{ij}A)$. Let $c = \mathrm{vec}(C)$ be the vector derived from C by stacking the columns of C on top of each other, and let $\mathrm{Diag}(c)$ be the $n^2 \times n^2$ diagonal matrix with the entries of c on its diagonal. The variable Y of the completely positive problem is also an $n^2 \times n^2$ matrix. Its $n \times n$ component blocks are addressed by Y^{ij} with $i, j = 1, \ldots, n$. Finally, δ_{ij} is the Kronecker-delta.

Using this notation, Povh and Rendl [50] show that the optimal value of QAP is the solution of the following completely positive program of order n^2:

$$\begin{aligned} OPT_{QAP} = \ & \min \ \langle B \otimes A + \mathrm{Diag}(c), Y \rangle \\ & \text{s.t.} \ \textstyle\sum_i Y^{ii} = I \\ & \qquad \langle I, Y^{ij} \rangle = \delta_{ij} \quad (i, j = 1, \ldots, n) \\ & \qquad \langle E, Y \rangle = n^2 \\ & \qquad Y \in \mathcal{C}^*. \end{aligned}$$

The problem of finding a **3-partitioning** of the vertices of a graph G was studied by Povh and Rendl in [51]. Consider a graph on n vertices with weights $a_{ij} \geq 0$ on its edges. The problem is to partition the vertices of G into subsets S_1, S_2, and S_3 with given cardinalities m_1, m_2, and m_3 (with $\sum_i m_i = n$) in such a way that the total weight of edges between S_1 and S_2 is minimal. Note that this problem contains the classical graph bisection problem as a special case.

The completely positive formulation requires some notation again. Letting e_i denote the ith unit vector in appropriate dimension, take $E_{ij} = e_i e_j^T$ and B_{ij} its symmetrized version $B_{ij} = 1/2(E_{ij} + E_{ji})$. For $j = 1, \ldots, n$, define matrices $W_j \in \mathbb{R}^{n \times n}$ by $W_j = e_j e^T$. Moreover, define the following 3×3 matrices: E_3 the all-ones matrix in $\mathbb{R}^{3\times 3}$, $B = 2B_{12}$ in $\mathbb{R}^{3\times 3}$ and for $i = 1, 2, 3$ define $V_i \in \mathbb{R}^{3\times 3}$ as $V_i = e_i e^T$.

With these notations, Povh and Rendl derive the following completely positive formulation of order $3n$:

$$\begin{aligned} & \min \ \tfrac{1}{2} \langle B^T \otimes A, Y \rangle \\ & \text{s.t.} \ \langle B_{ij} \otimes I, Y \rangle = m_i \delta_{ij} \qquad 1 \leq i \leq j \leq 3 \\ & \qquad \langle E_3 \otimes E_{ii}, Y \rangle = 1 \qquad i = 1, \ldots, n \\ & \qquad \langle V_i \otimes W_j^T, Y \rangle = m \qquad i = 1, 2, 3; \ j = 1, \ldots, n \\ & \qquad \langle B_{ij} \otimes E, Y \rangle = m_i m_j \qquad 1 \leq i \leq j \leq 3 \\ & \qquad Y \in \mathcal{C}^*. \end{aligned}$$

As far as we are aware, the above list comprises all problem classes for which an equivalent copositive or completely positive formulation has been established up to now. It illustrates that copositive programming is a powerful modelling tool which interlinks the quadratic and binary worlds. In the next sections, we will discuss properties of the cones as well as algorithmic approaches to tackle copositive programs.

3 The cones $\mathcal{C}$ and $\mathcal{C}^*$

Topological properties

Both $\mathcal{C}$ and $\mathcal{C}^*$ are full-dimensional closed, convex, pointed, non-polyhedral matrix cones. The interior of $\mathcal{C}$ is the set of strictly copositive matrices:

$$\operatorname{int}(\mathcal{C}) = \{A : x^T Ax > 0 \text{ for all } x \geq 0, x \neq 0\}.$$

The extremal rays of $\mathcal{C}^*$ are the rank-one completely positive matrices

$$\operatorname{Ext}(\mathcal{C}^*) = \{xx^T : x \geq 0\}.$$

Proofs of all these statements can be found in [6]. The interior of the completely positive cone has first been characterized in [27]. Dickinson [24] gave an improved characterization which reads as follows:

$$\operatorname{int}(\mathcal{C}^*) = \{AA^T : \operatorname{rank}(A) = n \text{ and } A = [a|B] \text{ with } a \in \mathbb{R}^n_{++}, B \geq 0\}.$$

Here the notation $[a|B]$ describes the matrix whose first column is a and whose other columns are the columns of B. An alternative characterization is

$$\operatorname{int}(\mathcal{C}^*) = \{AA^T : \operatorname{rank}(A) = n \text{ and } A > 0\}.$$

A full characterization of the extremal rays of $\mathcal{C}$ (or equivalently, a complete "outer" description of $\mathcal{C}^*$ in terms of supporting hyperplanes) is an open problem. Partial results can be found in [3, 4, 5, 32, 34].

Small dimensions

The cones $\mathcal{C}$ and $\mathcal{C}^*$ are closely related to the cones $\mathcal{S}^+$ of positive semidefinite matrices and $\mathcal{N}$ of entrywise nonnegative matrices, since we immediately get from the definitions that

$$\mathcal{C}^* \subseteq \mathcal{S}^+ \cap \mathcal{N} \quad \text{and} \quad \mathcal{C} \supseteq \mathcal{S}^+ + \mathcal{N}.$$

Matrices in $\mathcal{S}^+ \cap \mathcal{N}$ are sometimes called doubly nonnegative. It is a very interesting fact (cf. [41]) that for $n \times n$-matrices of order $n \leq 4$, we have equality in the above relations, whereas for $n \geq 5$, both inclusions are strict. A counterexample that illustrates $\mathcal{C} \neq \mathcal{S}^+ + \mathcal{N}$ is the so-called Horn-matrix, cf. [31]:

$$H = \begin{pmatrix} 1 & -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 \\ -1 & 1 & 1 & -1 & 1 \end{pmatrix}.$$

To see that H is copositive, write

$$\begin{aligned} x^T Hx &= (x_1 - x_2 + x_3 + x_4 - x_5)^2 + 4x_2x_4 + 4x_3(x_5 - x_4) \\ &= (x_1 - x_2 + x_3 - x_4 + x_5)^2 + 4x_2x_5 + 4x_1(x_4 - x_5). \end{aligned}$$

The first expression shows that $x^T Hx \geq 0$ for nonnegative x with $x_5 \geq x_4$, whereas the second shows $x^T Hx \geq 0$ for nonnegative x with $x_5 < x_4$. It can be shown [31] that H is extremal for $\mathcal{C}$, and consequently H can not be decomposed into $H = S + N$ with $S \in \mathcal{S}^+$ and $N \in \mathcal{N}$.

Why is this jump when the size of A changes from 4×4 to 5×5? This question was answered by Kogan and Berman [40] using graph theoretic arguments: associate to a given symmetric matrix $A \in \mathbb{R}^{n \times n}$ a graph G with n vertices, such that an edge (i, j) is present in G if and only if $A_{ij} \neq 0$. Kogan and Berman [40] define a graph G to be completely positive, if every matrix $A \in \mathcal{S}^+ \cap \mathcal{N}$ whose graph is G is completely positive, and they show that a graph is completely positive if and only if it does not contain a long odd cycle, i.e., a cycle of length greater than 4. Obviously, this can not happen in graphs on four vertices, which shows that for small dimensions $\mathcal{C}^* = \mathcal{S}^+ \cap \mathcal{N}$. Observe that the Horn-matrix is related to the 5-cycle via $H = E - 2A_5$, where A_5 the adjacency matrix of the 5-cycle.

The case of 5×5 copositive and completely positive matrices has therefore attracted special interest, and several papers have dealt with this setting, see [20] and references therein.

4 Testing copositivity and complete positivity

Complexity

It has been shown by Murty and Kabadi [44] that checking whether a given matrix $A \in \mathcal{C}$ is a co-NP-complete decision problem. Intuitively, checking $A \in \mathcal{C}^*$ should have the same computational complexity. It seems, however, that a formal proof of this statement has not yet been given.

This general complexity result does not exclude that for special matrix classes checking copositivity is cheaper. For example, for diagonal matrices one only needs to verify nonnegativity of the diagonal elements, evidently a linear-time task. This can be generalized: For tridiagonal matrices [10] and for acyclic matrices [35], testing copositivity is possible in linear time.

Complete positivity

There are several conditions, necessary and sufficient ones, for complete positivity of a matrix. Most of them use linear algebraic arguments or rely on properties of the graph associated to the matrix, and it seems unclear how they can be used for algorithmic methods to solve optimization problems over $\mathcal{C}^*$. For a comprehensible survey of these conditions, we refer to [6]. We just mention two sufficient conditions: a sufficient condition shown in [39] is that A is nonnegative and diagonally dominant. Another sufficient condition for $A \in \mathcal{S}^+ \cap \mathcal{N}$ to be in $\mathcal{C}^*$ is that A is tridiagonal or acyclic, as shown in [8].

Decomposing a given matrix $A \in \mathcal{C}^*$ into $A = \sum_{i=1}^{k} b_i b_i^T$ is also a nontrivial task. Since this is equivalent to finding a nonnegative matrix $B \in \mathbb{R}^{n \times k}$ (whose columns are b_i) with $A = BB^T$, this is sometimes called nonnegative factorization of A. A major line of research in the linear algebra literature is concerned with determining the minimal number k of factors necessary in such a decomposition. This quantity is called the cp-rank, and is conjectured [25] to be $\lfloor n^2/4 \rfloor$ if n is the order of the matrix. See [6] for more details on the cp-rank. Berman and Rothblum [7] proposed a non-polynomial algorithm to compute the cp-rank (and thus to determine whether a matrix is completely positive). Their method, however, does not provide a factorization. Jarre and Schmallowsky [37] also propose a procedure which for a given matrix A either determines a certificate proving $A \in \mathcal{C}^*$ or converges to a matrix $S \in \mathcal{C}^*$ which is in some sense "close" to A. Bomze [9] shows how a factorization of A can be used to construct a factorization of $\begin{pmatrix} 1 & b^T \\ b & A \end{pmatrix}$.

Copositivity criteria based on structural matrix properties

Obviously, copositivity of a matrix can not be checked through its eigenvalues. It can be checked by means of the so-called Pareto eigenvalues [33], but computing those is not doable in polynomial time. Spectral properties of copositive matrices provide some information and are discussed in [38].

For dimensions up to four, explicit descriptions are available [33]. For example, a symmetric 2×2 matrix A is copositive if and only if its entries fulfill $a_{11} \geq 0, a_{22} \geq 0$ and $a_{12} + \sqrt{a_{11}a_{22}} \geq 0$, see [1]. As this description indicates, the boundary of the cone $\mathcal{C}$ has both "flat parts" and "curved parts", so the cone is neither polyhedral nor strictly nonpolyhedral everywhere. This geometry and the facial structure of $\mathcal{C}$ is, however, not well-understood.

In all dimensions, copositive matrices necessarily have nonnegative diagonal elements: if $a_{ii} < 0$ for some i, then the corresponding coordinate vector e_i would provide $e_i^T A e_i = a_{ii} < 0$, thus contradicting copositivity of A.

A condition similar to the Schur-complement also holds for copositive matrices, as shown in [29]: Consider

$$A = \begin{pmatrix} a & b^T \\ b & C \end{pmatrix}$$

with $a \in \mathbb{R}$, $b \in \mathbb{R}^n$ and $C \in \mathbb{R}^{n \times n}$. Then A is copositive iff $a \geq 0$, C is copositive, and $y^T(aC - bb^T)y \geq 0$ for all $y \in \mathbb{R}^n_+$ such that $b^T y \leq 0$.

Numerous criteria for copositivity in terms of structural properties of the matrix have been given, many of them in terms of properties of principal submatrices. We name just one example stated in [21] but attributed to Motzkin: a symmetric matrix is strictly copositive iff each principal submatrix for which the cofactors of the last row are all positive has a positive determinant. Many conditions of the same flavor can be found in the literature. Again, it seems doubtful whether those conditions will prove useful for optimization purposes, so we refer to the surveys [33] and [36] for a more thorough treatment.

A recursive method to determine copositivity of a matrix has been proposed by Danninger [22].

An algorithmic approach

A conceptually different approach to copositivity testing which essentially uses global optimization techniques has been proposed in [18]. This approach relies on the observation that A is copositive iff the quadratic form $x^T Ax \geq 0$ on the standard simplex. If $v_1, \dots, v_n$ denote the vertices of a simplex, we can write a point x in the simplex in barycentric coordinates as $x = \sum_{i=1}^n \lambda_i v_i$ with $\lambda_i \geq 0$ and $\sum_{i=1}^n \lambda_i = 1$. This gives

$$x^T Ax = \sum_{i,j=1}^{n} v_i^T A v_j \lambda_i \lambda_j.$$

Hence, a necessary condition for $x^T Ax$ to be nonnegative on the simplex is that

$$v_i^T A v_j \geq 0 \text{ for all } i, j. \tag{6}$$

This condition can be refined by studying simplicial partitions of the standard simplex. As the partition gets finer, stronger and stronger necessary conditions are derived which, in the limit, capture all strictly copositive matrices. This approach gives very good numerical results for many matrices. It can be generalized in such a way that cones between $\mathcal{N}$ and $\mathcal{S}^+ + \mathcal{N}$ are used as certificates, see [54].

5 Approximation hierarchies

A matrix is copositive if its quadratic form is nonnegative for nonnegative arguments. Based on this definition, various approaches have used conditions which ensure positivity of polynomials.

For a given matrix $A \in \mathcal{S}$, consider the polynomial

$$P_A(x) := \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i^2 x_j^2.$$

Clearly, $A \in \mathcal{C}$ if and only if $P_A(x) \geq 0$ for all $x \in \mathbb{R}^n$. A sufficient condition for this is that $P_A(x)$ has a representation as a sum of squares (sos) of polynomials. Parrilo [47] showed that $P_A(x)$ allows a sum of squares decomposition if and only if $A \in \mathcal{S}^+ + \mathcal{N}$, yielding again the relation $\mathcal{S}^+ + \mathcal{N} \subseteq \mathcal{C}$.

A theorem by Pólya [49] states that if $f(x_1,\ldots,x_n)$ is a homogeneous polynomial which is positive on the standard simplex, then for sufficiently large $r \in \mathbb{N}$ the polynomial

$$f(x_1,\ldots,x_n)\cdot\left(\sum_{i=1}^n x_i^2\right)^r$$

has positive coefficients. Inspired by this result, Parrilo [47] (cf. also [23] and [12]) defined the following hierarchy of cones for $r \in \mathbb{N}$:

$$\mathcal{K}^r := \left\{A \in \mathcal{S} : P_A(x)\left(\sum_{i=1}^n x_i^2\right)^r \text{ has an sos decomposition }\right\}.$$

Parrilo showed $\mathcal{S}^+ + \mathcal{N} = \mathcal{K}^0 \subset \mathcal{K}^1 \subset \ldots$, and $\operatorname{int}(\mathcal{C}) \subseteq \bigcup_{r\in\mathbb{N}} \mathcal{K}^r$, so the cones $\mathcal{K}^r$ approximate $\mathcal{C}$ from the interior. Since the sos condition can be written as a system of linear matrix inequalities (LMIs), optimizing over $\mathcal{K}^r$ amounts to solving a semidefinite program.

Exploiting a different sufficient condition for nonnegativity of a polynomial, De Klerk and Pasechnik [23], cf. also Bomze and De Klerk [12], define

$$\mathcal{C}^r := \left\{A \in \mathcal{S} : P_A(x)\left(\sum_{i=1}^n x_i^2\right)^r \text{ has nonnegative coefficients }\right\}.$$

De Klerk and Pasechnik showed that $\mathcal{N} = \mathcal{C}^0 \subset \mathcal{C}^1 \subset \ldots$, and $\operatorname{int}(\mathcal{C}) \subseteq \bigcup_{r\in\mathbb{N}} \mathcal{C}^r$. Each of these cones is polyhedral, so optimizing over one of them is solving an LP.

Refining these approaches, Peña et al. [48] derive yet another hierarchy of cones approximating $\mathcal{C}$. Adopting standard multiindex notation, where for a given multiindex $\beta \in \mathbb{N}^n$ we have $|\beta| := \beta_1 + \cdots + \beta_n$ and $x^\beta := x_1^{\beta_1}\cdots x_n^{\beta_n}$, they define the following set of polynomials

$$\mathcal{E}^r := \left\{ \sum_{\beta\in\mathbb{N}^n, |\beta|=r} x^\beta x^T(S_\beta + N_\beta)x : S_\beta \in \mathcal{S}^+, N_\beta \in \mathcal{N} \right\}.$$

With this, they define the cones

$$\mathcal{Q}^r := \left\{A \in \mathcal{S} : x^TAx\left(\sum_{i=1}^n x_i^2\right)^r \in \mathcal{E}^r\right\}.$$

They show that $\mathcal{C}^r \subseteq \mathcal{Q}^r \subseteq \mathcal{K}^r$ for all $r \in \mathbb{N}$, with $\mathcal{Q}^r = \mathcal{K}^r$ for $r = 0, 1$. Similar to $\mathcal{K}^r$, the condition $A \in \mathcal{Q}^r$ can be rewritten as a system of LMIs. Optimizing over $\mathcal{Q}^r$ is therefore again an SDP.

All these approximation hierarchies approximate $\mathcal{C}$ uniformly and thus do not take into account any information provided by the objective function of an optimization problem. Moreover, in all these approaches the system of LMIs (resp. linear inequalities) gets large quickly as r increases. Thus, dimension of the SDPs increases so quickly that current SDP-solvers can only solve problems over those cones for small values of r, i.e., $r \leq 3$ at most.

We are not aware of comparable approximation schemes that approximate the completely positive cone $\mathcal{C}^*$ from the interior.

6 Algorithms

The approximation hierarchies described in the last section can be used to approximate a copositive program, and in many settings this gives very good results and strong bounds. However, the size of the problems increases exponentially as one goes through the approximation levels, so only low-level approximations are tractable.

As far as we are aware, there are two approaches to solve copositive programs directly: one is a feasible descent method in the completely positive cone $\mathcal{C}^*$, the other one approximates the copositive cone $\mathcal{C}$ by a sequence of polyhedral inner and outer approximations. In the sequel we briefly describe both methods.

Optimizing over $\mathcal{C}^*$

A recent attempt to solve optimization problems over $\mathcal{C}^*$ is a feasible descent method by Bomze et al. [14], who approximate the steepest descent path from a feasible starting point in $\mathcal{C}^*$. They study the problem

$$\begin{aligned} \min\ & \langle C, X\rangle \\ \text{s.t. } & \langle A_i, X\rangle = b_i \quad (i=1,\ldots,m), \\ & X \in \mathcal{C}^*. \end{aligned} \tag{7}$$

The optimal solution is approximated by a sequence of feasible solutions, and in this sense the algorithm resembles an interior point method. Starting from an initial feasible solution X^0 of which a factorization $X^0 = (V^0)(V^0)^T$ is assumed to be available, the next iteration point is $X^{j+1} = X^j + \Delta X^j$, where ΔX^j is a solution of the following regularized version of (7):

$$\begin{aligned} \min\ & \varepsilon\langle C, \Delta X\rangle + (1-\varepsilon)\|\Delta X\|_j^2 \\ \text{s.t. } & \langle A_i, \Delta X\rangle = 0 \quad (i=1,\ldots,m), \\ & X^j + \Delta X \in \mathcal{C}^*. \end{aligned}$$

The norm $\|\cdot\|_j$ used in iteration j depends on the current iterate X^j. Setting $X^{j+1} = (V + \Delta V)(V + \Delta V)^T$, they show the regularized problem to be equivalent to

$$\begin{aligned} \min\ & \varepsilon\langle C, V(\varDelta V)^T + (\varDelta V)V^T + (\varDelta V)(\varDelta V)^T\rangle \\ & \qquad +(1-\varepsilon)\|V(\varDelta V)^T + (\varDelta V)V^T + (\varDelta V)(\varDelta V)^T\|_j^2 \\ \text{s. t. } & \langle A_i, V(\varDelta V)^T + (\varDelta V)V^T + (\varDelta V)(\varDelta V)^T\rangle = 0 \quad (i = 1, \dots, m), \\ & V + \varDelta V \in \mathcal{N}. \end{aligned}$$

This problem now involves the tractable cone $\mathcal{N}$ instead of $\mathcal{C}^*$, but the objective is now a nonconvex quadratic function, and the equivalence statement only holds for the global optimum. Using linearization techniques and Tikhonov regularization for this last problem in V-space, the authors arrive at an implementable algorithm which shows promising numerical performance for the max-clique problem as well as box-constrained quadratic problems.

Convergence of this method is not guaranteed. Moreover, the algorithm requires knowledge of a feasible starting point together with its factorization. Finding a feasible point is in general as difficult as solving the original problem, and given the point, finding the factorization is highly nontrivial. In special settings, however, the factorized starting point comes for free.

Optimizing over $\mathcal{C}$

An algorithm for the copositive optimization problem (1) has been proposed in [17]. We also refer to [16] for a detailed elaboration. The method is based on the copositivity conditions developed in [18] which we briefly described in Section 4. Recall condition (6). Consider a simplicial partition $\mathcal{P}$ of the standard simplex $\varDelta$ into smaller simplices, i.e., a family $\mathcal{P} = \{\varDelta^1, \dots, \varDelta^m\}$ of simplices satisfying $\varDelta = \bigcup_{i=1}^m \varDelta^i$ and $\operatorname{int}(\varDelta^i) \cap \operatorname{int}(\varDelta^j) = \emptyset$ for $i \neq j$. We denote the set of all vertices of simplices in $\mathcal{P}$ by

$$V_{\mathcal{P}} = \{v : v \text{ is a vertex of some simplex in } \mathcal{P}\},$$

and the set of all edges of simplices in $\mathcal{P}$ by

$$E_{\mathcal{P}} = \{(u, v) : u \neq v \text{ are vertices of the same simplex in } \mathcal{P}\}.$$

In this notation, the necessary copositivity condition from [18] reads: a matrix A is copositive if $v^T Av \geq 0$ for all $v \in V_{\mathcal{P}}$ and $u^T Av \geq 0$ for all $(u, v) \in E_{\mathcal{P}}$, cf. (6). This motivates to define the following set corresponding to a given partition $\mathcal{P}$:

$$\begin{aligned} \mathcal{I}_{\mathcal{P}} := \{A \in \mathcal{S} : \ & v^T Av \geq 0 \ \text{ for all } v \in V_{\mathcal{P}}, \\ & u^T Av \geq 0 \ \text{ for all } (u, v) \in E_{\mathcal{P}}\}. \end{aligned}$$

It is not difficult so see that for each partition $\mathcal{P}$ the set $\mathcal{I}_{\mathcal{P}}$ is a closed, convex, polyhedral cone which approximates $\mathcal{C}$ from the interior. Likewise, define the sets

$$\mathcal{O}_{\mathcal{P}} := \{A \in \mathcal{S} : v^T Av \geq 0 \ \text{ for all } v \in V_{\mathcal{P}}\}.$$

These sets can be shown to be closed, convex, polyhedral cones which approximate $\mathcal{C}$ from the exterior. For both inner and outer approximating cones the approximation of $\mathcal{C}$ gets monotonically better if the partitions get finer. In the limit (i.e., if the diameter $\delta(\mathcal{P}) := \max_{\{u,v\}\in E_{\mathcal{P}}} \|u - v\|$ of the partitions goes to zero), the cones $\mathcal{I}_{\mathcal{P}}$ converge to $\mathcal{C}$ from the interior, and the $\mathcal{O}_{\mathcal{P}}$ converge to $\mathcal{C}$ from the exterior.

Note that due to their polyhedrality optimizing over $\mathcal{I}_{\mathcal{P}}$ or $\mathcal{O}_{\mathcal{P}}$ amounts to solving an LP. Now replacing the cone $\mathcal{C}$ in (1) by $\mathcal{I}_{\mathcal{P}}$ and $\mathcal{O}_{\mathcal{P}}$, respectively, results in two sequences of LPs whose solutions are upper, resp. lower, bounds of the optimal value of (1). Under standard assumptions, this algorithm is provably convergent.

The performance of this method relies on suitable strategies to derive simplicial partitions $\mathcal{P}$ of the standard simplex, and in this sense the approach resembles a Branch-and-Bound algorithm. The partitioning strategy can be guided adaptively through the objective function, yielding a good approximation of $\mathcal{C}$ in those parts of the cone that are relevant for the optimization and only a coarse approximation in those parts that are not.

A drawback is that the number of constraints in the auxiliary LPs grows very quickly and the constraint systems contain a lot of redundancy. This necessitates rather involved strategies to keep the size of the systems reasonable, but nonetheless computer memory (not cpu-time) remains the limiting factor for this algorithm.

The algorithm is not adequate for general models derived from Burer's result [19], and provides only poor results for box-constrained quadratic problems. However, the method turns out to be very successful for the standard quadratic problem: while a standard global optimization solver like BARON [55] solves StQPs in 30 variables in about 1000 seconds, this method solves problems in 2000 variables in 30 seconds (on average). This shows that the copositive approach to StQPs outperforms all other available methods.

A variant of this approach can be found in [56].

Conclusion and outlook

Copositive programming is a new versatile research direction in conic optimization. It is a powerful modelling tool and allows to formulate many combinatorial as well as nonconvex quadratic problems. In the copositive formulation, all intractable constraints (binary as well as quadratic constraints) get packed entirely in the cone constraint. Studying the structure of the copositive and completely positive cones thus provides new insight to both combinatorial and quadratic problems. Though formally very similar to semidefinite programs, copositive programs are NP-hard. Nonetheless, the copositive formulations have lead to new and tighter bounds for some combinatorial problems. Algorithmic approaches to directly solve copositive and completely positive problems have been proposed and given encouraging numerical results.

Copositive optimization continues to be a highly active research field. Future research will deal with both modeling issues and algorithmic improvements. For example, it would be intersting to extend Burer's result to problems with general quadratic constraints. The now available algorithms are not successful for all copositive models, so we need other, better models for some problem classes. It will also be very interesting to see new copositivity driven cutting planes for various combinatorial problems which will emerge from a better understanding of the facial geometry of $\mathcal{C}$.

On the algorithmic side, the methods need to be improved and adapted to different problem classes. Since now a very good algorithm for StQPs is available, a natural next step is to tailor this algorithm to QPs with arbitrary linear constraints or box constraints.

References

1. Andersson L.E., Chang G.Z., Elfving T., Criteria for copositive matrices using simplices and barycentric coordinates. *Linear Algebra and its Applications* 220(1995): 9–30.
2. Anstreicher K.M., Burer S., D.C. versus copositive bounds for standard QP. *Journal of Global Optimization* 33(2005): 199–312.
3. Baston V.J., Extreme copositive quadratic forms. *Acta Arithmetica* 15(1969): 319–327.
4. Baumert L.D., Extreme copositive quadratic forms. *Pacific Journal of Mathematics* 19(1966): 197–204.
5. Baumert L.D., Extreme copositive quadratic forms II. *Pacific Journal of Mathematics* 20(1967): 1–20.
6. Berman A., Shaked-Monderer N., *Completely positive matrices*, World Scientific, 2003.
7. Berman A., Rothblum, U., A note on the computation of the cp-rank. *Linear Algebra and its Applications* 419(2006), 1–7.
8. Berman A., Hershkowitz D., Combinatorial results on completely positive matrices. *Linear Algebra and its Applications* 95(1987), 111–125.
9. Bomze I.M., Building a completely positive factorization. *Technical Report* TR-ISDS 2009-06, Department of Statistics and Decision Support Systems, University of Vienna, Austria. Online at `http://www.optimization-online.org/DB_HTML/2009/08/2381.html`
10. Bomze I.M., Linear-time copositivity detection for tridiagonal matrices and extension to block-tridiagonality, *SIAM Journal on Matrix Analysis and Applications* 21(2000): 840–848.
11. Bomze I.M., Dür M., de Klerk E., Roos C., Quist A.J., Terlaky T., On copositive programming and standard quadratic optimization problems, *Journal of Global Optimization* 18(2000): 301–320.
12. Bomze I.M., de Klerk E., Solving standard quadratic optimization problems via linear, semidefinite and copositive programming. *Journal of Global Optimization* 24(2002): 163–185.
13. Bomze I.M., Jarre F., A note on Burer's copositive representation of mixed-binary QPs. *Technical Report* TR-ISDS 2009-04, Department of Statistics and

Decision Support Systems, University of Vienna, Austria. Online at `http://www.optimization-online.org/DB_HTML/2009/08/2368.html`

14. Bomze I.M., Jarre F., Rendl F., Quadratic factorization heuristics for copositive programming. *Technical Report* TR-ISDS 2009-08, Department of Statistics and Decision Support Systems, University of Vienna, Austria. Online at `http://www.optimization-online.org/DB_HTML/2009/10/2426.html`
15. Bomze I.M., Schachinger W., Multi-standard quadratic optimization: interior point methods and cone programming reformulation. *Computational Optimization and Applications* 45(2009): 237–256.
16. Bundfuss S., Copositive matrices, copositive programming, and applications. *Ph.D. Dissertation*, TU Darmstadt 2009. Online at `http://www3.mathematik.tu-darmstadt.de/index.php?id=483`
17. Bundfuss S., Dür M.: An adaptive linear approximation algorithm for copositive programs. *SIAM Journal on Optimization* 20(2009): 30–53.
18. Bundfuss S., Dür M.: Algorithmic copositivity detection by simplicial partition. *Linear Algebra and its Applications* 428(2008): 1511–1523.
19. Burer S., On the copositive representation of binary and continuous nonconvex quadratic programs. *Mathematical Programming* 120(2009): 479–495.
20. Burer S., Anstreicher K.M., Dür M., The difference between 5 × 5 doubly nonnegative and completely positive matrices. *Linear Algebra and its Applications* 431(2009): 1539–1552.
21. Cottle R.W., Habetler G.J., Lemke C.E., On classes of copositive matrices. *Linear Algebra and its Applications* 3(1970): 295–310.
22. Danninger G., A recursive algorithm for determining (strict) copositivity of a symmetric matrix. *Methods of Operations Research* 62(1990): 45–52.
23. de Klerk E., Pasechnik D.V., Approximation of the stability number of a graph via copositive programming, *SIAM Journal on Optimiaztion* 12(2002): 875–892.
24. Dickinson P.J.C., An improved characterisation of the interior of the completely positive cone. *Preprint* (2010).
25. Drew J.H., Johnson C.R., Loewy R., Completely positive matrices associated with M-matrices *Linear Algebra and Multilinear Algebra* 37(1994): 303–310.
26. Dukanovic I., Rendl F., Copositive programming motivated bounds on the stability and the chromatic numbers. *Mathematical Programming* 121(2010): 249–268.
27. Dür M., Still G., Interior points of the completely positive cone. *Electronic Journal of Linear Algebra* 17(2008): 48–53.
28. Eichfelder G., Jahn J., Set-semidefinite optimization. *Journal of Convex Analysis* 15(2008): 767–801.
29. Feng Y.Y., Li P., Criteria for copositive matrices of order four. *Linear Algebra and its Applications* 194(1993), 109–124.
30. Gvozdenović N., Laurent M., The operator Ψ for the chromatic number of a graph. *SIAM Journal on Optimization* 19(2008), 572–591.
31. Hall Jr. M., Newman M., Copositive and completely positive quadratic forms. *Proceedings of the Cambridge Philosophical Society* 59(1963): 329–33.
32. Haynsworth E., Hoffman A.J., Two remarks on copositive matrices. *Linear Algebra and its Applications* 2(1969): 387–392.
33. Hiriart-Urruty J.B., Seeger A., A variational approach to copositive matrices. Forthcoming in *SIAM Review.*
34. Hoffman A.J., Pereira F., On copositive matrices with $-1, 0, 1$ entries. *Journal of Combinatorial Theory* 14(1973): 302–309.

35. Ikramov K.D., Linear-time algorithm for verifying the copositivity of an acyclic matrix. *Computational Mathematics and Mathematical Physics* 42(2002): 1701–1703.
36. Ikramov K.D., Savel'eva N., Conditionally definite matrices. *Journal of Mathematical Sciences* 99(2000): 1–50.
37. Jarre F., Schmallowsky K., On the computation of $\mathcal{C}^*$ certificates. *Journal of Global Optimization* 45(2009): 281–296.
38. Johnson C.R., Reams R., Spectral theory of copositive matrices. *Linear Algebra and its applications* 395(2005): 275–281.
39. Kaykobad M., On nonnegative factorization of matrices. *Linear Algebra and its applications* 96(1987), 27–33.
40. Kogan N., Berman A., Characterization of completely positive graphs. *Discrete Mathematics* 114(1993): 297–304.
41. Maxfield J.E., Minc H., On the matrix equation $X'X = A$, *Proceedings of the Edinburgh Mathematical Society*, 13:125–129, 1962/1963.
42. National Bureau of Standards, Report 1818. *Quarterly Report, April through June 1952.*
43. Motzkin, T.S., Straus, E.G., Maxima for graphs and a new proof of a theorem of Turan, *Canadian Journal of Mathematics* 17(1965), 533–540.
44. Murty K.G., Kabadi S.N., Some NP-complete problems in quadratic and nonlinear programming. *Mathematical Programming* 39(1987): 117–129.
45. Natarajan K., Teo C.P., Zheng Z., Mixed zero-one linear programs under objective uncertainty: a completely positive representation. *Preprint.* Online at `http://www.optimization-online.org/DB_HTML/2009/08/2365.html`
46. Nesterov Y., Nemirovskii A., *Interior-point polynomial algorithms in convex programming*, SIAM Studies in Applied Mathematics, 13.
47. Parrilo P., Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization, *Ph.D. Dissertation*, California Institute of Technology, 2000. Available at:
`http://etd.caltech.edu/etd/available/etd-05062004-055516/`
48. Peña J., Vera J., Zuluaga L., Computing the stability number of a graph via linear and semidefinite programming, *SIAM Journal on Optimization* 18(2007): 87–105.
49. Pólya G., Über positive Darstellung von Polynomen. *Vierteljahresschrift der naturforschenden Gesellschaft in Zürich* 73(1928): 141–145.
50. Povh J., Rendl F., Copositive and semidefinite relaxations of the Quadratic Assignment Problem. *Discrete Optimization* 6(2009): 231–241.
51. Povh J., Rendl F., A copositive programming approach to graph partitioning. *SIAM Journal on Optimization* 18(2007): 223–241.
52. Preisig J.C., Copositivity and the minimization of quadratic functions with nonnegativity and quadratic equality constraints. *SIAM Journal on Control and Optimization* 34(1996): 1135–1150.
53. Quist A.J., de Klerk E., Roos C., Terlaky T., Copositive relaxation for general quadratic programming. *Optimization Methods and Software* 9(1998): 185–208.
54. Sponsel J., Bundfuss S., Dür, M., Testing copositivity using semidefinitness. *Manuscript in preparation* (2009).
55. Tawarmalani M., Sahinidis N.V., Global optimization of mixed-integer nonlinear programs: A theoretical and computational study. *Mathematical Programming* 99(2004): 563–591.

56. Yıldırım E.A., On the accuracy of uniform polyhedral approximations of the copositive cone. *Preprint.* Online at
`http://www.optimization-online.org/DB_HTML/2009/07/2342.html`

A Robust H_∞ Quasi-LPV Approach for Designing Nonlinear Observers

Daniel F. Coutinho[1] and Alain Vande Wouwer[2]

[1] Group of Automation and Control Systems, PUCRS, Av. Ipiranga 6681, Porto Alegre-RS, 90619-900, Brazil. dcoutinho@pucrs.br

[2] Service d'Automatique, Université de Mons (UMONS), 31 Boulevard Dolez, B-7000 Mons, Belgium. Alain.VandeWouwer@umons.ac.be

Summary. This work applies the quasi-LPV technique to the design of robust observers for a class of bioreactors. The system nonlinearities are modeled in terms of two time varying parameter vectors, $\theta(t)$ and $\delta(t)$. The vector $\theta(t)$ contains all nonlinear terms that are only function of the measurements, whereas the remaining terms are lumped into the vector $\delta(t)$. Then, a $\theta(t)$ parameter-dependent Luenberger-like observer is proposed, where the design conditions are given in terms of linear matrix inequality constraints. These conditions ensure regional stability w.r.t. to a set of admissible initial conditions and also minimizes an upper-bound on the $\mathcal{L}_2$-gain of the error system. These results are applied to a high cell density bioreator.

1 Introduction

Since the seminal works of Kalman [1] and Luenberger [2], state estimation of dynamical systems has been an active topic of research in control theory, fault detection and information fusion. State estimation can be defined as the task of estimating a function of the states of a dynamical system based on a (usually uncertain) model and the measurements of its outputs which may be corrupted by disturbance signals. Popular state estimators for linear systems are the Kalman Filter and Luenberger observer, in which a certain level of accuracy on the system model is required. When the model is uncertain, the observer may have poor performance or even assume an erratic behavior. Moreover, in many practical situations the signal to be observed results from a nonlinear map, and only approximate solutions can be obtained based on system linearization, as used in the extended Kalman filter (EKF) [3]. Certainly, the design of nonlinear observers is much more involved than the linear counterpart, and has led to a wide diversity of approaches, see, for instance, [4], [5], [6], [7] and [8].

On the other hand, the problem of robustness and disturbance rejection in control theory has been addressed by means of convex optimization techniques. To this end, the control problem is recast as a set of linear matrix

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*, DOI 10.1007/978-3-642-12598-0_2, © Springer-Verlag Berlin Heidelberg 2010

inequalities (LMIs) through the Lyapunov theory and a solution is then obtained using very efficient interior-point method algorithms [9]. However, the LMI framework cannot be applied in a straightforward way to deal with nonlinear dynamical systems. Recently, several authors have modeled the system nonlinearities as time-varying parameters giving rise to the (quasi)-LPV approach [10]. However, the majority of these approaches does not properly address the stability problem (note that the stability properties only hold locally when dealing with nonlinear systems).

In this work, we propose a convex optimization problem for designing a Luenberger-like observer for uncertain nonlinear systems. First, the system nonlinearities are modeled as bounded time-varying parameters leading to a quasi-LPV representation of the system. Then, the design conditions are expressed in terms of a set of parameter-dependent LMIs, which can be numerically solved [9]. The proposed LMI conditions ensure regional stability of the error system and disturbance attenuation performance in an $\mathcal{H}_\infty$ setting. This approach is applied to the estimation of the dead biomass concentration in a high cell density bioreactor.

2 Problem Statement

Consider the following nonlinear system

$$\dot{x} = f(x,q) + G(x,q)u + B_w w \ , \ y = C_y x + D_w w \ , \ x(0) = x_0 \tag{1}$$

where $x \in \mathcal{X} \subset \Re^n$ is the state vector; $q \in \mathcal{Q} \subset \Re^{n_q}$ is the vector of parametric uncertainties; $y \in \mathcal{Y} \subset \Re^{n_y}$ is the measurement vector; $u \in \mathcal{U} \in \Re^{n_u}$ is the control input; $w \in \mathcal{L}_{2,[0,T]}^{n_w}$ is a vector of disturbance signals with bounded energy in finite horizon; $f(\cdot) \in \Re^n$ is a locally Lipschitz vector function of (x,q); $G(\cdot) \in \Re^{n \times n_u}$ is a continuous and bounded matrix function of (x,q); and $B_w \in \Re^{n \times n_w}$, $C_y \in \Re^{n_y \times n}$ and $D_w \in \Re^{n_y \times n_w}$ are constant matrices. We assume w.r.t. system (1) that $\mathcal{X}$, $\mathcal{Q}$ and $\mathcal{U}$ are given polytopic sets. Let $\mathcal{Y}$ be a polytopic covering of the set $\{y : y = C_y x \ , \ x \in \mathcal{X}\}$.

The objective of this work is to estimate a vector

$$\xi = C_\xi x \ , \ \xi \in \Re^{n_\xi} \ , \tag{2}$$

from the measurement of y, where $C_\xi \in \Re^{n_\xi \times n}$ is a given constant matrix. To this end, we propose the following observer

$$\dot{\hat{x}} = f(\hat{x},0) + G(\hat{x},0)u + L(y,u)(y - \hat{y}) \ , \ \hat{y} = C_y \hat{x} \ , \ \hat{\xi} = C_\xi \hat{x} \tag{3}$$

where $\hat{x} \in \hat{\mathcal{X}} \subset \Re^n$ is the observer state; $\hat{y} \in \Re^{n_y}$ and $\hat{\xi} \in \Re^{n_\xi}$ are estimates of y and ξ, respectively; $L(\cdot)\Re^{n \times n_y}$ is a matrix function of (y,u) to be determined; and $f(\hat{x},0)$ and $G(\hat{x},0)$ are the same functions as $f(x,q)$ and $G(x,q)$ in (1) obtained by setting $x = \hat{x}$ and $q = 0$ (nominal model).

For designing the matrix $L(y,u)$, we model the nonlinearities of system (1) using output and input-dependent parameters, e.g., $\theta_i = \theta_i(y,u)$, for $i = 1,\ldots,n_\theta$, as well as parameters affected by the uncertainties (possibly state and input dependent), e.g., $\delta_j = \delta_j(x,u,q)$, for $j = 1,\ldots,n_\delta$, such that the original system dynamics can be cast in the following form:

$$\dot{x} = (A_1(\theta) + A_2(\delta))x + (B_1(\theta) + B_2(\delta))u + B_w w \tag{4}$$

where $\theta := [\theta_1 \cdots \theta_{n_\theta}]' \in \Re^{n_\theta}$; $\delta := [\delta_1 \cdots \delta_{n_\delta}]' \in \Re^{n_\delta}$; $A_1(\cdot)$ and $B_1(\cdot)$ are affine matrix functions of θ; and $A_2(\cdot)$ and $B_2(\cdot)$ are linear matrix functions of δ. Let $\Theta := \{\theta : \underline{\alpha_i} \leq \theta_i \leq \overline{\alpha}_i; \underline{\alpha}_i, \overline{\alpha}_i \in \Re; i = 1,\ldots,n_\theta\}$ and $\Delta := \{\delta : \underline{\beta}_i \leq \delta_i \leq \overline{\beta}_i; \underline{\beta}_i, \overline{\beta}_i \in \Re; i = 1,\ldots,n_\delta\}$ be given polytopes such that $\theta = \theta(y,u) \in \Theta$ and $\delta = \delta(x,u,q) \in \Delta$ for all $(x,y,u,q) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{U} \times \mathcal{Q}$. We stress that there is no limitation in rewriting the original system in (1) as the time-varying parameter dependent representation in (4). However, we are likely to be conservative when dealing with very complex dynamics.

In light of (4), we redefine the observer dynamics as follows:

$$\dot{\hat{x}} = A_1(\theta)\hat{x} + B_1(\theta)u + L(\theta)(y - \hat{y}) \ , \ \hat{y} = C_y\hat{x} \ , \ \hat{\xi} = C_\xi\hat{x} \tag{5}$$

where the observer gain $L(\theta) \in \Re^{n \times n_y}$ is constrained to be an affine matrix function of θ.

In this work, we aim at designing the matrix $L(\theta)$ such that the observer has regional stability properties and some performance w.r.t. to disturbance signals. Let $e := x - \hat{x}$ be the state vector of the following error system

$$\dot{e} = (A_1(\theta) - L(\theta)C_y)e + A_2(\delta)x + B_2(\delta)u + (B_w - L(\theta)D_w)w \tag{6}$$

and let $z := \xi - \hat{\xi} = C_\xi e$ be the observation error. We assume w.r.t. the above error system that the pair $(A_1(\theta), C_y)$ is observable for all $\theta \in \Theta$ and there exists a polytopic covering $\mathcal{E}$ such that $e \in \mathcal{E}$ for all $x \in \mathcal{X}$ and $\hat{x} \in \hat{\mathcal{X}}$.

Notice that the error dynamics in (6) is also a function of x and u due the non-cancelation of the terms that are dependent on the uncertain parameters δ. To overcome this problem, let $\nu := [x' \ \ u']' \in \Re^{n_\nu}$, $n_\nu = n + n_u$, be a fictitious disturbance signal. The vector ν belongs to $\mathcal{L}_{2,[0,T]}^{n_\nu}$, as x and u are assumed bounded.

Now, we introduce the following definition of regional stability to be applied in the sequel.

Definition 1. *Consider system (6). Let $\mathcal{W} := \{w : \|w\|_{2,[0,T]}^2 \leq \rho_w, \rho_w \in \Re_+\}$ and $\mathcal{N} := \{\nu : \|\nu\|_{2,[0,T]}^2 \leq \rho_\nu, \rho_\nu \in \Re_+\}$. The system (6) is regionally stable w.r.t. $\mathcal{R} \subset \mathcal{E}$ and $\mathcal{W} \times \mathcal{N}$, if for any $e(0) \in \mathcal{R}_0 \subset \mathcal{R}$, $w \in \mathcal{W}$ and $\nu \in \mathcal{N}$, the trajectory $e(t)$ remains in $\mathcal{R}$ for all $t \in [0,T]$, $\theta \in \Theta$ and $\delta \in \Delta$.*

In view of the above, we can state the problem to be addressed in this work as follows: *design the matrix $L(\theta)$ such that the system (6) is regionally stable* *w.r.t. $\mathcal{R}$ and $\mathcal{W} \times \mathcal{N}$ for all $\theta \in \Theta$ and $\delta \in \Delta$ and such that an upper bound γ on the finite horizon $\mathcal{L}_2$-gain from w_a to z is minimized, where $w_a := [\, \nu' \ w' \,]'$.*

We end this section recalling the following results of the Lyapunov theory [11].

Lemma 1. *Consider a nonlinear system $\dot{x} = f(x, w)$, where $x \in \mathcal{X} \subset \Re^n$, $w \in \mathcal{W} \subset \mathcal{L}^{n_w}_{2,[0,T]}$, $\mathcal{W} := \{w : \|w\|^2_{2,[0,T]} \le \kappa \ , \ \kappa \in \Re_+\}$, $f(0, 0) = 0$ and $f(x, w)$ satisfies the conditions for existence and uniqueness of solution for all $(x, w) \in \mathcal{X} \times \mathcal{W}$. Suppose there exist a continuously differentiable function $V : \mathcal{X} \mapsto \Re$ and positive scalars ϵ_1 and ϵ_2 satisfying the following conditions:*

$$\epsilon_1 x'x \le V(x) \le \epsilon_2 x'x \ , \ \dot{V}(x) - w'w < 0 \ , \ \mathcal{R} \subset \mathcal{X} \ , \ \forall \ x \in \mathcal{X} \tag{7}$$

where $\mathcal{R} = \{x : V(x) \le 1 + \kappa\}$. Then, the nonlinear system is regionally exponentially stable w.r.t. $\mathcal{R} \subset \mathcal{X}$ and $\mathcal{W}$. That is, for any $x(0) \in \mathcal{R}_0 \subset \mathcal{R}$, where $\mathcal{R}_0 := \{x : V(x) \le 1\}$, and $w \in \mathcal{W}$, the state trajectory $x(t)$ remains in $\mathcal{R}$ for all $t \in [0, T]$.

Lemma 2. *Consider a nonlinear map $z = h(x, w)$, $\dot{x} = f(x, w)$, where $x \in \mathcal{X} \subset \Re^n$, $z \in \Re^{n_z}$, $w \in \mathcal{W} \subset \mathcal{L}^{n_w}_{2,[0,T]}$, $f(0, 0) = 0$, $f(x, w)$ satisfies the conditions for existence and uniqueness of solution and $h(x, w)$ is a continuous and bounded vector function, for all $(x, w) \in \mathcal{X} \times \mathcal{W}$. Assume the system $\dot{x} = f(x, w)$ is regionally exponentially stable w.r.t. to $\mathcal{R} \subset \mathcal{X}$ and $\mathcal{W}$. Let $\gamma \in \Re_+$ be a given scalar. Suppose there exist positive scalars ϵ_1 and ϵ_2 and a continuously differentiable function $V : \mathcal{X} \mapsto \Re$ satisfying the following inequalities:*

$$\epsilon_1 x'x \le V(x) \le \epsilon_2 x'x \ , \ \dot{V}(x) + \frac{z'z}{\gamma} - \gamma w'w < 0 \ , \ \forall \ (x, w) \in \mathcal{X} \times \mathcal{W} \tag{8}$$

Then, the finite horizon $\mathcal{L}_2$-gain from w to z is bounded by γ, i.e.

$$\|\mathcal{G}_{wz}\|_{\infty,[0,T]} := \sup_{0 \ne w \in \mathcal{W}, x \in \mathcal{R}} \frac{\|z\|_{2,[0,T]}}{\|w\|_{2,[0,T]}} \le \gamma \tag{9}$$

3 Observer Design

In this section, we devise a convex optimization problem for determining the matrix $L(\theta)$. Basically, we translate the stability and performance conditions of Lemmas 1 and 2 into parameter-dependent LMI constraints which can be numerically solved.

To this end, let $B_\nu(\delta) := [\, A_2(\delta) \ \ B_2(\delta) \,]$. Thus, we can recast the error system in (6) in the following compact form:

$$\dot{e} = (A_1(\theta) - L(\theta) C_y) e + B_\nu(\delta) \nu + (B_w - L(\theta) D_w) w \ , \ z = C_\xi e \ . \tag{10}$$

Now, consider the following Lyapunov function candidate:

$$V(e) = e'Pe \ , \ P = P' > 0 \tag{11}$$

Remark 1. It turns out that the above quadratic function may be conservative for assessing the stability of parameter-dependent systems. However, the stability conditions will also depend on $\dot{\theta}$ and $\dot{\delta}$ if we consider a parameter-dependent Lyapunov function [12]. To avoid the estimation of bounding sets for $\dot{\theta}$ and $\dot{\delta}$, the Lyapunov function is constrained to be parameter independent. In addition, we provide an estimate of the system stability region based on a level set of $V(e)$, and a parameter dependent estimate would not be practical.

In view of (10) and (11), the time derivative of $V(e)$ is as follows:

$$\dot{V} = 2e'P\dot{e} = \zeta' \Pi(P,L)\zeta \tag{12}$$

where $\zeta = [\,e' \ \nu' \ w'\,]'$ and

$$\Pi(P,L) = \begin{bmatrix} \mathrm{Her}(PA_1(\theta) - PL(\theta)C_y) & PB_\nu(\delta) & PB_w - PL(\theta)D_w \\ B_\nu(\delta)'P & 0 & 0 \\ B_w'P - D_w'L(\theta)'P & 0 & 0 \end{bmatrix}$$

with $\mathrm{Her}(\cdot) = (\cdot) + (\cdot)'$.

Notice in Lemma 1 that for ensuring regional stability properties we have to guarantee that $\mathcal{R} \subset \mathcal{E}$. To this end, we first consider that the polytope $\mathcal{E}$ can be represented by the following set of inequalities

$$\mathcal{E} = \{e : m_j'e \le 1 \ , \ j = 1, \ldots, n_m\} \tag{13}$$

where $m_1, \ldots, m_{n_m} \in \Re^n$ are constant vectors defining the n_m edges of $\mathcal{E}$. Then, the condition $\mathcal{R} \subset \mathcal{E}$ can be written as follows:

$$2 - m_j'e - e'm_j \ge 0 \ , \ \forall \ e \ : V(e) - (1+\kappa) \le 0$$

Applying the $\mathcal{S}$-procedure [9], the above is guaranteed if the following holds for some scalars $\tau_j \in \Re_+$:

$$\begin{bmatrix} 1 \\ -e \end{bmatrix}' \begin{bmatrix} (2\tau_j - 1 - \kappa) & \tau_j m_j' \\ \tau_j m_j & P \end{bmatrix} \begin{bmatrix} 1 \\ -e \end{bmatrix} \ge 0 \ , \ j = 1, \ldots, n_m \tag{14}$$

In view of the above, we propose the following Theorem.

Theorem 1. *Consider the error system in (10). Let $\mathcal{U}$, Θ and Δ be given polytopic sets. Let $\mathcal{W} := \{w : \|w\|^2_{2,[0,T]} \le \rho_w \ , \ \rho_w \in \Re_+\}$ and $\mathcal{N} := \{\nu : \|\nu\|^2_{2,[0,T]} \le \rho_\nu \ , \ \rho_\nu \in \Re_+\}$ be given admissible sets of disturbances w and ν, respectively. Let $\mathcal{E}$ be a given polytope as defined in (13). Let $\mathcal{R}_0 := \{e : e'P_0e \le 1 \ , \ P_0 = P_0' > 0\} \subset \mathcal{E}$ be a given set of admissible initial conditions for the error system. Suppose the matrices $P = P'$, $R_0, R_1, \ldots, R_{n_\theta}$, with appropriate dimensions, and scalars $\tau_1, \ldots, \tau_{n_m}$ and γ are a solution to the following optimization problem, where the LMIs are constructed at all $(\theta, \delta) \in \mathcal{V}(\Theta \times \Delta)$ and $j = 1, \ldots, n_m$*

$$\min_{P,\dots,\gamma} \gamma \; : \; \begin{cases} P_0 - P \geq 0 \;, \; \begin{bmatrix} (2\tau_j - 1 - \rho_w - \rho_\nu) & \tau_j m_j' \\ \tau_j m_j & P \end{bmatrix} > 0 \\ \\ \Gamma_a(P, R_0, \dots, R_{n_\theta}) < 0 \;, \; \Gamma_b(P, R_0, \dots, R_{n_\theta}, \gamma) < 0 \end{cases} \tag{15}$$

where $\mathcal{V}(\cdot)$ is the set of vertices of $(\cdot)$, $R(\theta) = R_0 + \theta_1 R_1 + \cdots + \theta_{n_\theta} R_{n_\theta}$ and

$$\Gamma_a(P, R_0, \dots, R_{n_\theta}) = \begin{bmatrix} \mathrm{Her}(PA_1(\theta) - R(\theta)C_y) & \star & \star \\ B_\nu(\delta)'P & -I_{n_\nu} & 0 \\ (B_w'P - D_w'R(\theta)') & 0 & -I_{n_w} \end{bmatrix} , \tag{16}$$

$$\Gamma_b(P, R_0, \dots, R_{n_\theta}, \gamma) = \begin{bmatrix} \mathrm{Her}(PA_1(\theta) - R(\theta)C_y) & \star & \star & C_\xi' \\ B_\nu(\delta)'P & -\gamma I_{n_\nu} & 0 & 0 \\ (B_w'P - D_w'R(\theta)') & 0 & -\gamma I_{n_w} & 0 \\ C_\xi & 0 & 0 & -\gamma I_{n_\xi} \end{bmatrix}$$

with $\star$ denoting symmetric block matrices. Then, the error system in (10), with $L(\theta) = P^{-1}R(\theta)$, is regionally exponentially stable w.r.t. $\mathcal{R} := \{e : e'Pe \leq 1 + \rho_w + \rho_\nu\} \subset \mathcal{E}$ and $\mathcal{W} \times \mathcal{N}$. That is, for any $e(0) \in \mathcal{R}_0 := \{e : e'P_0 e \leq 1\} \subset \mathcal{R}$, $w \in \mathcal{W}$ and $\nu \in \mathcal{N}$, the trajectory $e(t)$ remains in $\mathcal{R}$ for all $t \in [0, T]$. Moreover, $\|\mathcal{G}_{w_a z}\|_{\infty,[0,T]} \leq \gamma$, where $w_a = [\,w' \; \nu'\,]'$.

Proof. Firstly, note that the last two LMIs in (15) are affine in (θ, δ). If they hold for all $(\theta, \delta) \in \mathcal{V}(\Theta \times \Delta)$, then by convexity they also hold for all $(\theta, \delta) \in \Theta \times \Delta$.

Secondly, consider the second LMI in (15). From the Schur complement [9], we obtain that $P > 0$. Let $\underline{\lambda}$ and $\overline{\lambda}$ be the smallest and largest eigenvalues of P, respectively. We have $\underline{\lambda} e'e \leq e'Pe \leq \overline{\lambda} e'e$ In addition, pre- and post-multiplying the second LMI in (15) by $[\,1 \; -e'\,]$ and its transpose, respectively, leads to (14) with $\kappa = \rho_w + \rho_\nu$. Hence, $\mathcal{R} := \{e : e'Pe \leq 1 + \rho_w + \rho_\nu\} \in \mathcal{E}$. Notice that the first LMI in (15) implies $\mathcal{R}_0 := \{e : e'P_0 e \leq 1\} \subset \mathcal{R}$.

Thirdly, consider the notation in (12) and let $w_a = [\,\nu' \; w'\,]'$. Pre- and post-multiplying the third LMI in (15) by ζ' and ζ, respectively, leads to $\dot{V} - w_a' w_a < 0$, since $\dot{V} = \zeta' \Pi(P, L) \zeta$ by noting that $R(\theta) = PL(\theta)$.

Finally, applying the Schur complement to the fourth LMI in (15) yields

$$\begin{bmatrix} \mathrm{Her}(PA_1(\theta) - R(\theta)C_y) - \gamma^{-1} C_\xi' C_\xi & \star & \star \\ B_\nu(\delta)'P & -\gamma I_{n_\nu} & 0 \\ (B_w'P - D_w'R(\theta)') & 0 & -\gamma I_{n_w} \end{bmatrix} < 0$$

Pre- and post-multiplying the above by ζ' and ζ, respectively leads to $\dot{V} + \gamma^{-1} z'z - \gamma w_a' w_a < 0$, and the rest of this proof follows immediately from Lemmas 1 and 2.

4 Case Study: A High Cell Density Bioreactor

To analyze the proposed approach for observer design, we consider the high-cell-density perfusion bioreactor recently studied in [13]. The general bioreactor setup is illustrated in Figure 1. The fresh culture medium with concentration S_{in} is fed to the bioreactor through the input flow F_{in} which can be diluted by the flow F_w (with null glucose concentration). The culture medium is drained from the bioreactor by the bleed flow F_1 or by the perfusion flow F_2. A filtering device (for instance, an acoustic device for sono-perfusion) allows the biomass to be recirculated to the bioreactor, so that F_2 can be considered as a cell-free stream.

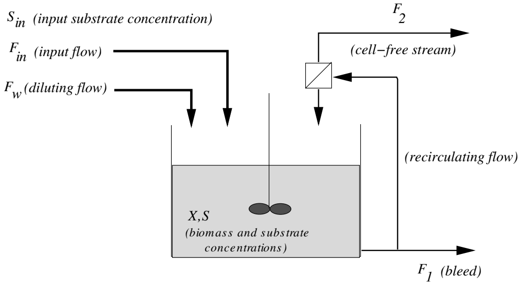


Fig. 1. A high-cell-density perfusion/chemostat bioreactor.

For this system, the main objective is to control the biomass and substrate concentrations in the culture medium by means of the inputs F_{in} and F_1. The flows F_w and F_2 are employed to guarantee a constant volume operation. If $F_{in} \geq F_1$ we have to drain the tank by means of F_2 (and then F_w is null). Otherwise, i.e., in the case of $F_1 > F_{in}$, we have to fill the tank by adding the glucose free flow F_w to maintain a constant volume in the tank where $F_2 = 0$. In the following, we assume that the flows F_w and F_2 are perfectly regulated by an additional controller, so that either $F_w + F_{in} = F_1$ or $F_{in} = F_1 + F_2$.

The bioreactor dynamics can be expressed in terms of the living biomass $y_1 = X$, the substrate $y_2 = S$ and the dead biomass ξ concentrations leading to the following state space representation [13]:

$$\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{\xi} \end{bmatrix} = \begin{bmatrix} 1 & -k_2 \\ -k_1 & 0 \\ 0 & k_2 \end{bmatrix} \begin{bmatrix} \psi_1(y,\xi) \\ \psi_2(y,\xi) \end{bmatrix} - \begin{bmatrix} y_1 & 0 \\ \sigma y_2 & (1-\sigma)y_2 - S_{in} \\ \xi & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{17}$$

where $y = [y_1\ y_2]'$ is the measurement; $u_1 = F_1/v$ and $u_2 = F_{in}/v$ are the dilution rates with v denoting the reactor volume; k_1 and k_2 are the pseudo-stoichiometric coefficients; S_{in} is the input substrate concentration; σ is a binary number standing for the reactor mode of operation; and $\psi_1(\cdot)$ and $\psi_2(\cdot)$ are nonlinear functions representing the reaction kinetics. In [13], the reaction kinetics ψ_1 and ψ_2 are modeled by the following functions:

$$\psi_1 = \mu \frac{y_1 y_2}{K_c y_1 + y_2} \quad \text{and} \quad \psi_2 = (y_1 + \xi) y_1 \tag{18}$$

where μ is the maximal biomass growth rate, and K_c is the saturation coefficient.

The numerical value of the constants k_1, k_2, K_c and μ are experimentally difficult to estimate [14]. To overcome this problem, we consider a robust feedback linearizing control law as given below:

$$u_1 = \frac{1}{y_1}\left(\frac{\bar{\mu} y_1 y_2}{\bar{K}_c + y_2} - \bar{k}_2(y_1+\xi)y_1 - 1.2(\tilde{y}_1^{sp} - y_1)\right) \ , \ \tilde{y}_1^{sp} = \int (y_1^{sp} - y_1)dt \quad (19)$$

$$u_2(\sigma) = \frac{1}{S_{in}}\left(\frac{\bar{\mu}\bar{k}_1 y_1 y_2}{\bar{K}_c + y_2} + y_2 u_{(1+\sigma)} + 2.3(\tilde{y}_2^{sp} - y_2)\right) \ , \ \tilde{y}_2^{sp} = \int (y_2^{sp} - y_2)dt \quad (20)$$

where y_1^{sp}, y_2^{sp} are the set-points of y_1 and y_2; $u_{(1+\sigma)}$ is a switched control input, which can be either u_1 when $\sigma = 0$ or u_2 for $\sigma = 1$; and $\bar{k}_1, \bar{k}_2, \bar{K}_c$ and $\bar{\mu}$ represent the nominal values of the uncertain parameters $k_i(q) = \bar{k}_i(1+q_i)$, $i = 1, 2$, $\mu(q) = \bar{\mu}(1+q_\mu)$, $K_c(q) = \bar{K}_c(1+q_c)$ with $q = [q_1 \ q_2 \ q_\mu \ q_c]'$ being the uncertainty vector (i.e., admissible deviations from the nominal values).

To implement the control law, we need the information on the dead biomass concentration which is difficult to measure on-line. Hence, we design an observer to estimate the state ξ from the measurements of y_1 and y_2 considering the results given in Section 3.

To simplify the design of the observer, we consider the following representation for system (17)

$$\dot{y}_1 = \bar{\mu}(\theta_4 + \delta_2)y_1 - \bar{k}_2(\theta_1 + \delta_1)(y_1 + \xi) - \theta_1 u_1 \quad (21)$$

$$\dot{y}_2 = -\bar{k}_1\bar{\mu}(\theta_4 + \delta_3)y_1 + S_{in}u_2 - \theta_2 u_{(1+\sigma)} \quad (22)$$

$$\dot{\xi} = \bar{k}_2(\theta_1 + \delta_1)(y_1 + \xi) - \theta_3\xi \quad (23)$$

with the time-varying parameters $\theta_1 = y_1$, $\theta_2 = y_2$, $\theta_3 = u_1$, $\delta_1 = q_2 y_1$, and

$$\theta_4 = \frac{y_2}{\bar{K}_c y_1 + y_2} \ , \ \delta_2 = y_2\frac{q_\mu(\bar{K}_c y_1 + y_2) - \bar{K}_c q_c y_1}{(\bar{K}_c(1+q_c)y_1 + y_2)(\bar{K}_c y_1 + y_2)}$$

$$\delta_3 = y_2\frac{(\bar{K}_c y_1 + y_2)(q_1 + q_\mu + q_1 q_\mu) - \bar{K}_c q_c y_1}{(\bar{K}_c(1+q_c)y_1 + y_2)(\bar{K}_c y_1 + y_2)} \quad (24)$$

For the above system, we consider the following numerical data taken from [13]: $\bar{k}_1 = 5.2$, $\bar{k}_2 = 7.6 \times 10^{-5}$, $\bar{K}_c = 8.0$, and $\bar{\mu} = 5.04 \times 10^{-2}$, where we assume $\mathcal{Q} = \{q : |q_1| \leq 0.2, |q_2| \leq 0.2, |q_\mu| \leq 0.5, |q_c| \leq 0.5\}$. In addition, we consider that $y^{sp} := [y_1^{sp} \ y_2^{sp}]'$ may assume values in the following set $\mathcal{Y}^{sp} = \{y^{sp} : 0.5 \leq y_1^{sp} \leq 10, 10 \leq y_2^{sp} \leq 25\}$. Assuming the measurement of ξ and applying a gridding technique to the parameter vector $q \in \mathcal{Q}$, we have performed exhaustive simulations where we have observed that the controller in (19)-(20) leads to a zero closed-loop steady-state tracking error with no overshoot (or undershoot) for step changes in $y^{sp} \in \mathcal{Y}^{sp}$.

In view of the above observations, we have defined the polytope of admissible states as $\mathcal{X} = \{y_1, y_2, \xi : 0.5 \leq y_1 \leq 10, 10 \leq y_2 \leq 25, 0 \leq$

$\xi \leq 1\}$. Taking the definitions of θ and δ in (24) into account, we have $\Theta = \{\theta : 1 \leq \theta_1 \leq 6, 5 \leq \theta_2 \leq 20, 0.01 \leq \theta_3 \leq 0.2, 0.2 \leq \theta_4 \leq 0.8\}$ and $\Delta = \{\delta : |\delta_1| \leq 0.6, 0.01 \leq \delta_2 \leq 0.12, 0.01 \leq \delta_3 \leq 0.15\}$. In addition, we have assumed $\mathcal{N} = \{\nu : \|\nu\|_{2,[0,T]} \leq 1\}$, $\mathcal{E} = \{e : |e_1| \leq 4, |e_2| \leq 4, |e_3| \leq 1\}$, and $\mathcal{R}_0 = \{e : e_1^2 \leq 9, e_2^2 \leq 9, e_3^2 \leq 1/4\}$. Thus, we get the following results from Theorem 1:

$$L = \begin{bmatrix} 825.416 & 0.000 \\ 0.000 & 814.835 \\ 2.5 \times 10^{-5} & 0.000 \end{bmatrix}, \quad P = \begin{bmatrix} 0.0625 & 0.0000 & 0.0000 \\ 0.0000 & 0.0625 & 0.0000 \\ 0.0000 & 0.0000 & 0.9999 \end{bmatrix}, \quad \gamma = 10.820\,,$$

where $L(\theta)$ was constrained to be constant, and the computation has been performed using YALMIP/SDPT3 package [15, 16]. The total CPU time was 0.3 [sec] running in a Core2 Duo CPU (@ 2.26GHz) with 21 LMI constraints and 14 decision variables.

Figure 2 shows the (living) biomass (y_1), the dead biomass (ξ) and its estimate ($\hat{\xi}$) concentrations for the following step variations on the reference signals

$$y_1^{sp}(t) = \begin{cases} 6 & 0 \leq t \leq 5\ [hours] \\ & \text{for} \\ 2 & t > 5\ [hours] \end{cases}, \quad y_2^{sp}(t) = \begin{cases} 16 & 0 \leq t \leq 15\ [hours] \\ & \text{for} \\ 20 & t > 15\ [hours] \end{cases}$$

where we have considered a band limited white noise (with noise power 0.1) on the measurement of the living biomass, the worst case values for k_1, k_2, K_c and μ, the control law in (19) and (20) with $\hat{\xi}$, and the following observer:

$$\begin{bmatrix} \dot{\hat{y}}_1 \\ \dot{\hat{y}}_2 \\ \dot{\hat{\xi}} \end{bmatrix} = \begin{bmatrix} 1 & -\bar{k}_2 \\ -\bar{k}_1 & 0 \\ 0 & \bar{k}_2 \end{bmatrix} \begin{bmatrix} \hat{\psi}_1 \\ \hat{\psi}_2 \end{bmatrix} - \begin{bmatrix} \hat{y}_1 & 0 \\ \sigma y_2 & (1-\sigma) y_2 - S_{in} \\ \hat{\xi} & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} - L(y - \hat{y})$$

with $\hat{\psi}_1 = \bar{\mu}\hat{y}_1 y_2/(\bar{K}_c\hat{y}_1 + y_2)$ and $\hat{\psi}_2 = (\hat{y}_1 + \hat{\xi})\hat{y}_1$.

It turns out that there is a constant steady-state error (ultimate boundness stability [17]) on the estimate of ξ due to uncertainties on the model parameters. However, the controller in (19) and (20) is robust against constant errors leading to a response similar to the observer-free closed-loop system.

5 Conclusion

This work proposes a convex optimization approach for designing nonlinear observers with guaranteed regional stability and disturbance attenuation performance for the error system. Basically, the uncertain nonlinear system is modeled by means of a quasi-LPV representation and then an LMI-based formulation is derived to determine the observer gain. The proposed approach is successfully applied to estimate the dead biomass in a high-cell-density bioreactor recently studied in the process control literature.

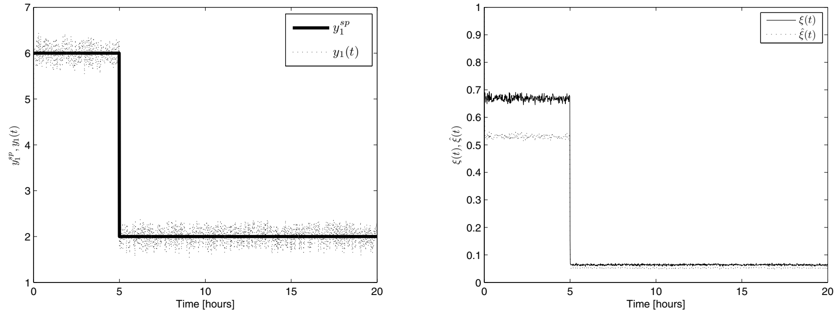


Fig. 2. Living, y_1, and dead biomass, ξ, concentrations and the estimate $\hat{\xi}$.

Acknowledgments

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian Federal Science Policy Office (BELSPO). The scientific responsibility rests with its authors. D. Coutinho has been beneficiary of a post-doc fellowship granted by BELSPO.

References

1. Kalman R, Bucy R (1961) Trans ASME 83(D):95–108
2. Luenberger D (1964) Trans Military Electron MIL-8:74–80
3. Anderson B, Moore J (1979) Optimal Filtering. Prentice Hall, Upper Saddle River
4. Nijmeijer H, Fossen T (1999) New Directions in Nonlinear Observer Design (LNCIS 244). Springer-Verlag, London
5. Jo NH, Seo JH (2000) IEEE T Automat Contr 45(5):968–973
6. Arcak M, Kokotović P (2001) Automatica 37(12):1923–1930
7. Liu Y, Li X-Y (2003) IEEE T Automat Contr 48(6):1041–1045
8. Arulampalam MS, Maskell S, Gordon N, Clapp T (2002) IEEE T Signal Process 50(2):174–188
9. Boyd S, El Ghaoui L, Feron E, Balakrishnan V (1994) Linear Matrix Inequalities in System and Control Theory. SIAM, Philadelphia
10. Leith D, Leithead W (2000) Int J Control 73(11):1001–1025
11. Khalil H (1996) Nonlinear Systems. Prentice Hall, Upper Saddle River
12. Souza C, Trofino A (2006) Int J Robust Nonlinear Control 16(5):243–257
13. Deschenes J, Desbiens A, Perrier M, Kamen A (2006) Ind Eng Chem Res 45(26):8985–8997
14. Bastin G, Dochain D (1990) On-line Estimation and Adaptive Control of Bioreactors. Elsevier, Amsterdam
15. Löfberg J (2004) YALMIP: A Toolbox for Modeling and Optimization in MATLAB. Proc CACSD Conf, Taipei - Taiwan
16. Tutuncu R, Toh K, Todd M (2003) Mathem Program Ser B 95:189–217
17. Dawson D, Qu Z, Carroll J (1992) Systems & Contr Letters 18(3):217–222

Solving Infinite-dimensional Optimization Problems by Polynomial Approximation

Olivier Devolder[1], François Glineur[1], and Yurii Nesterov[1]

ICTEAM & IMMAQ, Université catholique de Louvain,
CORE, Voie du Roman Pays, 34, Louvain-la-Neuve, B-1348, Belgium,
{Olivier.Devolder,Francois.Glineur,Yurii.Nesterov}@uclouvain.be[†]

Summary. We solve a class of convex infinite-dimensional optimization problems using a numerical approximation method that does not rely on discretization. Instead, we restrict the decision variable to a sequence of finite-dimensional linear subspaces of the original infinite-dimensional space and solve the corresponding finite-dimensional problems in a efficient way using structured convex optimization techniques. We prove that, under some reasonable assumptions, the sequence of these optimal values converges to the optimal value of the original infinite-dimensional problem and give an explicit description of the corresponding rate of convergence.

1 Introduction

Optimization problems in infinite-dimensional spaces, and in particular in functional spaces, were already considered in the 17th century: the development of the calculus of variations, motivated by physical problems, focused on the development of necessary and sufficient optimality conditions and finding closed-form solutions. Much later, the advent of computers in the mid-20th century led to the consideration of finite-dimensional optimization from an algorithmic point of view, with linear and nonlinear programming. Finally, a general theory of optimization in normed spaces began to appear in the 70's [8, 2], leading to a more systematic and algorithmic approach to infinite-dimensional optimization.

Nowadays, infinite-dimensional optimization problems appear in a lot of active fields of optimization, such as PDE-constrained optimization [7], with applications to optimal control, shape optimization or topology optimization. Moreover, the generalization of many classical finite optimization problems to a continuous time setting lead to infinite-dimensional problems.

[†] The first author is a F.R.S.-FNRS Research Fellow. This text presents research results of the Belgian Program on Interuniversity Poles of Attraction initiated by the Belgian State, Prime Minister's Office, Science Policy Programming. The scientific responsibility is assumed by the authors.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_3, © Springer-Verlag Berlin Heidelberg 2010

From the algorithmic point of view, these problems are often solved using discretization techniques (either discretization of the problem or discretization of the algorithm). In this work, we consider a different method of resolution that does not rely on discretization: instead, we restrict the decision variables to a sequence of finite-dimensional linear subspaces of the original infinite-dimensional space, and solve the corresponding finite-dimensional problems.

2 Problem class and examples

Consider a normed vector space $(X, \|.\|_X)$ of infinite dimension and its topological dual $(X', \|.\|_{X'})$ equipped with the dual norm. We focus on the following class of convex infinite-dimensional optimization problems:

$$P^* = \inf_{x \in X} \langle c, x\rangle \text{ s.t } \langle a_i, x\rangle = b_i \quad \forall i = 1, \ldots, L \text{ and } \|x\|_X \leq M \qquad \text{(P)}$$

where $c \in X', a_i \in X'$, $M \in \mathbb{R}_{++}$, $b_i \in \mathbb{R}$ for all $i = 1, \ldots, L$ (L is finite) and P^* denotes the optimal objective value. This problem class, with a linear objective, linear equalities and a single nonlinear inequality bounding the norm of the decision variable, is one of the simplest that allows us to outline and analyze our approximation technique. Nevertheless, it can be used to model many applications, among which the following simple continuous-time supply problem, which we describe for the sake of illustration.

A company buys a specific substance (for example oil or gas) in continuous time. Assume that this substance is made of L different constituents and that its composition continuously changes with time. In the same way, the price of this substance follows a market rate and therefore also changes in continuous time.

The finite time interval $[0, T]$ represents one production day. Assume that, for each constituent i, a specific daily demand b_i must be satisfied at the end of the day. We want to compute a purchase plan $x(t)$, i.e. the quantity of substance to buy at each time t, such that it meets the daily demands for a minimal total cost. For this application, the decision functional space X can be taken as the space of continuous functions on $[0, T]$ (see also Section 5 for other examples of suitable functional spaces). Denoting the price of the substance at time t by $\gamma(t)$, the amount of constituent i in the substance at time t by $\alpha_i(t)$, we obtain the following infinite-dimensional problem

$$\inf_{x \in X} \int_0^T \gamma(t)x(t)dt \text{ s.t } \int_0^T \alpha_i(t)x(t)dt = b_i \; \forall i \text{ and } 0 \leq x(t) \leq K \; \forall t \in T$$

where we also impose a bound for the maximal quantity that we can buy at each moment of time. The objective function and equality constraints are linear, so that we only need to model the bound constraints as a norm constraint. This is easily done with a linear change of variable: letting $x(t) = \frac{1}{2}K + \bar{x}(t) \; \forall t$, the bound constraint becomes $\|\bar{x}\|_\infty \leq \frac{1}{2}K$, which now fits the format of (P).

This model can also be used to compute how to modify an existing purchase plan when changes in the demands occur. Denote the modification of the daily demand for the constituent i by Δb_i and the change in our purchase quantity at time t by $\tilde{x}(t)$, and assume we do not want to modify the existing planning too much, so that we impose the constraint $\|\tilde{x}\| \leq M$ for a given norm. We obtain the following infinite-dimensional problem:

$$\inf_{x \in X} \int_0^T \gamma(t)\tilde{x}(t)dt \text{ s.t } \int_0^T \alpha_i(t)\tilde{x}(t)dt = \Delta b_i \ \forall i = 1, \ldots, L \text{ and } \|\tilde{x}\| \leq M$$

which also belongs to problem class (P). Finally, note that this problem class allows the formulation of continuous linear programs (CLPs, see [2]), such as

$$\inf_{x \in X} \int \gamma(t)x(t)dt \text{ s.t} \int \alpha_i(t)x(t)dt = b_i \ \forall i \text{ and } x(t) \geq 0 \ \forall t$$

provided we know an upper bound K on the supremum of $x(t)$, so that the nonnegativity constraint can be replaced by $0 \leq x(t) \leq K \ \forall t$, which can be rewritten using the infinity norm with a linear change of variables as in first example above.

3 Finite-dimensional approximations

We propose to approximate infinite-dimensional problem (P) by a sequence of finite-dimensional approximations. Let $\{p_1, \ldots, p_n, \ldots\} \subset X$ be an infinite family of linearly independent elements of X and denote by $X_n = \operatorname{span}\{p_1, \ldots, p_n\}$, the finite-dimensional linear subspace generated by the first n elements of this family.

Replacing the infinite-dimensional space X in (P) by X_n, we obtain the following family of problems with optimal values P_n^*

$$P_n^* = \inf_{x \in X_n} \langle c, x \rangle \text{ s.t } \langle a_i, x \rangle = b_i \ \forall i = 1, \ldots, L \text{ and } \|x\|_X \leq M \ . \qquad (\mathrm{P}_n)$$

Expressing function x as $x = \sum_{i=1}^n x_i p_i$ and denoting the finite vector of variables x_i by $\mathbf{x}$ leads to the following family of equivalent finite-dimensional formulations

$$P_n^* = \inf_{\mathbf{x} \in \mathbb{R}^n} \langle c^{(n)}, \mathbf{x} \rangle \text{ s.t } \langle a_i^{(n)}, \mathbf{x} \rangle = b_i \ \forall i = 1, \ldots, L \text{ and } \Big\| \sum_{i=1}^n x_i p_i \Big\|_X \leq M \ ,$$

where $c^{(n)}$ and $a_i^{(n)}$ are vectors in $\mathbb{R}_n$ whose components are defined by $[c^{(n)}]_j = \langle c, p_j \rangle$ and $[a_i^{(n)}]_j = \langle a_i, p_j \rangle \ \forall j = 1, \ldots, n$ and $\forall i = 1, \ldots, L$.

For our approach to be effective, these problems must be solvable by existing algorithms for finite-dimensional optimization. In particular, we would like to ensure that the bounded norm inequality can be handled by existing efficient optimization methods (the other components of the problem, namely the linear objective and linear equalities, are usually easily handled). We now list some explicit situations where this is indeed the case.

1. The easiest case corresponds to situations where X is a Hilbert space. Indeed, if we choose in that case $\{p_1, \ldots, p_n\}$ to be an orthonormal basis of X_n, we have that $\|\sum_{i=1}^n x_i p_i\|_X = \|\mathbf{x}\|_2$, where the last norm is the standard Euclidean norm of $\mathbb{R}^n$. The bounded norm inequality becomes a simple convex quadratic constraint, hence the approximation problem (P_n) can easily be solved (in fact, it admits a solution in closed form).

In the rest of this list, we focus on situations where functional space X is a Lebesgue or Sobolev space (see Section 5 for some properties) and where the basis elements $p_1, p_2, \ldots$ are polynomials (hence the title of this work), because this leads in many situations to problems that can be efficiently solved. We can take for example the monomial basis $X_n = \text{span}\{1, t, \ldots, t^{n-1}\}$, which means that variable x in problem (P_n) can be written $x(t) = \sum_{i=0}^{n-1} x_i t^i$ and becomes a polynomial of degree $n-1$.

2. Let $[a,b]$ denote a bounded, semi-infinite or an infinite interval. When $X = L^\infty([a,b])$, the norm inequality $\|x\|_\infty \leq M$ can be formulated as $-M \leq x(t) \leq M \; \forall t \in [a,b]$, which is equivalent to requiring positivity of both polynomials $x(t) + M$ and $M - x(t)$ on interval $[a,b]$. This in turn can be formulated as a semidefinite constraint, using the sum of squares approach (see e.g. [10]). Therefore, problems (P_n) can be efficiently solved as a semidefinite programming problem, using interior-point methods with polynomial-time worst-case algorithmic complexity.
3. When X is the Sobolev space $W^{k,\infty}([a,b])$, we have that constraint $\|x\|_{k,\infty} \leq M$ is equivalent to $-M \leq x^{(l)}(t) \leq M \forall t \in [a,b]$ and $\forall l \leq k$, where $x^{(l)}(t)$ is the l^{th} derivative of $x(t)$, whose coefficients depend linearly on those of vector $\mathbf{x}$. Therefore, as in the previous case, we solve the corresponding (P_n) as a semidefinite programming problem.
4. In the case of $X = L^q([a,b])$ where q is an *even* integer, we use Gaussian quadrature to obtain an suitable finite-dimensional representation of the constraint $\|x\|_q = (\int_a^b |x(t)|^q dt)^{1/q} \leq M$. We use the following result (see e.g. [6]):

   Theorem 1. *Given an integer m, there exists a set of m abscissas $\{z_1, z_2, \ldots, z_m\}$ and a set of m positive weights $\{w_1, w_2, \ldots, w_m\}$ such that the quadrature formula $\int_a^b f(x)dx \approx \sum_{i=1}^m w_i f(z_i)$ is exact for all polynomials of degree less or equal to $2m-1$.*

   We now use the fact that, since $x(t)$ is a polynomial of degree at most $n-1$, $|x(t)|^q$ is a polynomial of degree at most $q(n-1)$, so that we can choose $m = \frac{1}{2}q(n-1)+1$ and have $\int_a^b |x(t)|^q dt = \sum_{i=1}^{\frac{1}{2}q(n-1)+1} w_i \lambda_i^q$ where $\lambda_i = x(z_i)$; note that quantities λ_i depend linearly on the coefficients $\mathbf{x}$ of polynomial $x(t)$. The bound constraint can now be written as $\sum_{i=1}^{\frac{1}{2}q(n-1)+1} w_i \lambda_i^q \leq M^q$, which is a structured convex constraint on vector of variables $\mathbf{x}$. Because a self-concordant barrier is known for this set [9, Ch. 6], it can be solved in polynomial time with an interior-point method. The same kind of approach can be used to obtain an explicit translation in finite dimension of the polynomial approximation when $X = W^{k,q}([a,b])$ for even integers q.

Now that we know how to solve problems (P_n) efficiently, we show in the next section that, under some reasonable assumptions, the sequence of their optimal values P_n^* converges to the optimal value of the original infinite-dimensional problem P^* when $n \to +\infty$.

4 Convergence of the approximations

The optimal values of problems (P) and (P$_n$) clearly satisfy $P^* \leq P_n^*$; moreover, $P_{n+1}^* \leq P_n^*$ holds for all $n \geq 0$. In order to prove that P_n^* converges to P^*, we also need to find an upper bound on the difference $P_n^* - P^*$. Our proof requires the introduction of a third problem, a relaxation of problem (P$_n$) where the equality constraints are only satisfied approximately. More specifically, we define the linear operator $A : X \to \mathbb{R}^L$ by $[Ax]_i = \langle a_i, x \rangle$, form the vector $b = (b_1, b_2, \ldots, b_L)$ and impose that the norm of the residual vector $Ax - b$ is bounded by a positive parameter ϵ:

$$P_{n,\epsilon}^* = \inf_{x \in X_n} \langle c, x \rangle \text{ s.t } \|Ax - b\|_q \leq \epsilon \text{ and } \|x\|_X \leq M \qquad (\mathrm{P}_{n,\epsilon})$$

(we equip $\mathbb{R}^L$, the space of residuals, with the classical q-norm $\|.\|_q$ norm and define the conjugate exponent q' by $\frac{1}{q} + \frac{1}{q'} = 1$). We clearly have $P_{n,\epsilon}^* \leq P_n$. Our proof of an upper bound for the quantity $P_n^* - P^* = (P_n^* - P_{n,\epsilon}^*) + (P_{n,\epsilon}^* - P^*)$ proceeds in two steps: we first prove an upper bound on $P_{n,\epsilon}^* - P^*$ for a specific value of ϵ depending on n, and then use a general regularity theorem to establish a bound on the difference $P_n^* - P_{n,\epsilon}^*$.

We use the following notations: for $x \in X$, an element of best approximation of x in X_n is denoted by $P_{X_n}(x) = \arg\min_{p \in X_n} \|x - p\|_X$ (such kind of elements exists as X is a normed vector space and X_n is a finite-dimensional linear subspace see e.g. section 1.6 in [4]; in case it is not unique, it is enough to select one of these best approximations in the following developments), while the corresponding best approximation error is $E_n(x) = \min_{p \in X_n} \|x - p\|_X = \|x - P_{X_n}(x)\|_X$.

4.1 Upper bound on $P_{n,\epsilon}^* - P^*$

Assume that problem (P) is solvable. This is true for example if X is a reflexive Banach space or the topological dual of a separable Banach space (see [12] and Section 5 for further comments on this issue). Let x_{opt} be an optimal solution to (P) and let us consider $P_{X_n}(x_{\text{opt}})$, its best approximation in X_n (note that if (P) is not solvable, we can consider for all $\mu > 0$ a μ-solution x_μ of this problem, i.e. a feasible solution such that $< c, x_\mu > \leq P^* + \mu$, and replace P^* by $P^* + \mu$ in the following developments).

First, $\|P_{X_n}(x_{\text{opt}})\|_X$ can be bigger than $\|x_{\text{opt}}\|_X$, and does not necessarily satisfy the norm inequality constraint, but we have $\|P_{X_n}(x_{\text{opt}})\|_X \leq \|x_{\text{opt}}\|_X + \|x_{\text{opt}} - P_{X_n}(x_{\text{opt}})\|_X \leq M + E_n(x_{\text{opt}})$. Therefore, if we choose $\lambda = \frac{M}{M + E_n(x_{\text{opt}})}$ and $\overline{p} = \lambda P_{X_n}(x_{\text{opt}})$, we obtain $\|\overline{p}\|_X \leq M$. Moreover, we have $\|\overline{p} - x_{\text{opt}}\|_X \leq 2E_n(x_{\text{opt}})$ because we can write $\|\overline{p} - x_{\text{opt}}\|_X \leq \|\overline{p} - P_{X_n}(x_{\text{opt}})\|_X + \|P_{X_n}(x_{\text{opt}}) - x_{\text{opt}}\|_X$, and $\|\overline{p} - P_{X_n}(x_{\text{opt}})\|_X = \|(\lambda - 1)P_{X_n}(x_{\text{opt})}\|_X \leq (1 - \lambda)(M + E_n(x_{\text{opt}})) \leq E_n(x_{\text{opt}})$.

On the other hand, we have for all $i = 1, \ldots, L$ that $|\langle a_i, \overline{p} \rangle - b_i| = |\langle a_i, \overline{p} - x_{\text{opt}} \rangle| \leq \|a_i\|_{X'} \|\overline{p} - x_{\text{opt}}\|_X \leq \|a_i\|_{X'} 2E_n(x_{\text{opt}})$. Therefore, choosing $\epsilon(n) = 2E_n(x_{\text{opt}}) \left(\sum_{i=1}^L \|a_i\|_{X'}^q\right)^{1/q}$, we obtain that $\overline{p}$ is feasible for the problem ($\mathrm{P}_{n,\epsilon(n)}$). Similarly, we have $|\langle c, x_{\mathrm{opt}}\rangle - \langle c, \overline{p}\rangle| \leq \|c\|_{X'} 2E_n(x_{\mathrm{opt}})$, and we have proved the following lemma:

Lemma 1. *For* $\epsilon(n) = 2E_n(x_{opt})\left(\sum_{i=1}^{L} \|a_i\|_{X'}^q\right)^{1/q}$*, the optimal values of problems* (P) *and* ($\mathrm{P}_{n,\epsilon(n)}$) *satisfy* $P^*_{n,\epsilon(n)} - P^* \leq \|c\|_{X'} 2E_n(x_{opt})$.

4.2 Upper bound on $P^*_n - P^*_{n,\epsilon}$

We first introduce a general regularity theorem that compares the optimal value of a problem with linear equality constraints with the optimal value of its relaxation, and then apply it to the specific pair of problems (P_n) and ($\mathrm{P}_{n,\epsilon}$).

Regularity Theorem

Let $(Z, \|.\|_Z)$ and $(Y, \|.\|_Y)$ be two normed vector space, $A : Z \to Y$ be a linear operator, $Q \subset Z$ be a convex bounded closed subset of Z with nonempty interior, $b \in Y$ and $\mathcal{L} = \{z \in Z : Az = b\}$. We denote the distance between a point z and subspace $\mathcal{L}$ by $d(z, \mathcal{L}) = \inf_{y \in \mathcal{L}} \|z - y\|_Z$.

Lemma 2. *Assume that there exists a point* $\hat{z} \in Z$ *such that* $A\hat{z} = b$ *and* $B(\hat{z}, \rho) \subset Q \subset B(\hat{z}, R)$ *for some* $0 < \rho \leq R$*. Then, for every point* $z \in Q$ *such that* $d(z, \mathcal{L}) \leq \delta$*, there exists* $\tilde{z} \in \mathcal{L} \cap Q$ *such that* $\|z - \tilde{z}\|_Z \leq \delta \left(1 + \frac{R}{\rho}\right)$.

Proof. Denote $Q_z = \mathrm{conv}(z, B(\hat{z}, \rho)) \subset Q$. The support function of this set is $\sigma_{Q_z}(s) = \sup_{y \in Q_z} \langle s, y\rangle = \max\{\langle s, z\rangle, \langle s, \hat{z}\rangle + \rho \|s\|_{Z'}\}$. Let π be any element of best approximation of the point z into $\mathcal{L}$. Define $\alpha = \frac{\rho}{\rho+\delta}$ and consider $\tilde{z} = \alpha\pi + (1-\alpha)\hat{z}$. Then we have for any $s \in Z'$ that

$$\begin{aligned}
\langle s, \tilde{z}\rangle &= \alpha\langle s, z\rangle + (1-\alpha)\langle s, \hat{z}\rangle + \alpha\langle s, \pi - z\rangle \\
&\leq \alpha\langle s, z\rangle + (1-\alpha)\left[\langle s, \hat{z}\rangle + \frac{\alpha\delta}{1-\alpha}\|s\|_{Z'}\right] \\
&= \alpha\langle s, z\rangle + (1-\alpha)\left[\langle s, \hat{z}\rangle + \rho\|s\|_{Z'}\right] \leq \sigma_{Q_z}(s)
\end{aligned}$$

and hence $\tilde{z} \in Q_z \subset Q$. Since we also have $\tilde{z} \in \mathcal{L}$, it remains to note that

$$\begin{aligned}
\|z - \tilde{z}\|_Z &\leq \|z - \pi\|_Z + \|\pi - \tilde{z}\|_Z = \delta + (1-\alpha)\|\pi - \hat{z}\|_Z \\
&\leq \delta + (1-\alpha)(\|\pi - z\|_Z + \|z - \hat{z}\|_Z) \leq \delta + (1-\alpha)(\delta + R) \\
&= \delta\left(1 + \frac{R+\delta}{\rho+\delta}\right) \leq \delta\left(1 + \frac{R}{\rho}\right)
\end{aligned}$$

□

We consider now the following optimization problem:

$$g^* = \inf_{z \in Z} \langle c, z\rangle \text{ s.t } Az = b \text{ and } z \in Q \tag{G}$$

and its relaxed version

$$g^*_\epsilon = \inf_{z \in Z} \langle c, z\rangle \text{ s.t } \|Az - b\|_Y \leq \epsilon \text{ and } z \in Q. \tag{G$_\epsilon$}$$

The following Regularity Theorem links the optimal values of these two problems.

Theorem 2 (Regularity Theorem). *Assume that*

A1. (G_ϵ) *is solvable,*
A2. *there exists* $\hat{z} \in Z$ *s.t* $A\hat{z} = b$ *and* $B(\hat{z}, \rho) \subset Q \subset B(\hat{z}, R)$ *for* $0 < \rho \leq R$,
A3. *the operator* $A : Z \to Y$ *is non degenerate, i.e. there exists a constant* $\sigma > 0$ *such that* $\|Az - b\|_Y \geq \sigma d(z, \mathcal{L})$ $\forall z \in Z$.

Then $g^* \geq g^*_\epsilon \geq g^* - \frac{\epsilon\|c\|_{Z'}}{\sigma}\left(1 + \frac{R}{\rho}\right)$.

Proof. The first inequality is evident. For the second one, consider z^*_ϵ, an optimal value of the problem (G_ϵ). Since $d(z^*_\epsilon, \mathcal{L}) \leq \delta := \frac{\epsilon}{\sigma}$, in view of Lemma 2, there exists a point $\tilde{z} \in \mathcal{L} \cap Q$ such that $\|z^*_\epsilon - \tilde{z}\|_Z \leq \delta\left(1 + \frac{R}{\rho}\right)$. Therefore, we can conclude $g^*_\epsilon = \langle c, z^*_\epsilon\rangle = \langle c, \tilde{z}\rangle + \langle c, z^*_\epsilon - \tilde{z}\rangle \geq g^* - \|c\|_{Z'}\,\delta\left(1 + \frac{R}{\rho}\right)$ □

Satisfying the hypotheses of the Regularity Theorem

We want to apply the Regularity Theorem to the pair of problems (P_n) and $(\mathrm{P}_{n,\epsilon})$. First, we note that, as X_n is finite-dimensional, the set $\{x \in X_n : \|x\|_X \leq M, \|Ax - b\|_q \leq \epsilon\}$ is compact. As the functional c is continuous, we conclude that problem $(\mathrm{P}_{n,\epsilon})$ is solvable, i.e. hypothesis A1 is satisfied.

In order to prove hypothesis A2, we assume that there exists $\hat{x} \in X_n$ such that $A\hat{x} = b$ and $\|\hat{x}\|_X < M$ (a kind of Slater condition). If we denote $\mathcal{R}_n = \min_{x\in X_n, Ax=b}\|x\|_X$, $\tilde{x} = \arg\min_{x\in X_n, Ax=b}\|x\|_X$ and $Q_n = \{x \in X_n : \|x\|_X \leq M\} = B_{X_n}(0, M)$, we have : $B_{X_n}(\tilde{x}, M - \mathcal{R}_n) \subset Q_n \subset B_{X_n}(\tilde{x}, 2M)$.

Regarding hypothesis A3, denote $\mathcal{L}_n = \{x \in X_n : Ax = b\}$ and write

$$\begin{aligned} d(x, \mathcal{L}_n) &= \min_{u\in X_n, Au=b}\|x - u\|_X = \min_{\lambda\in\mathbb{R}^n, A^{(n)}\lambda=b}\left\|x - \sum_{i=1}^n \lambda_i p_i\right\|_X \\ &= \min_{u\in X_n}\max_{y\in\mathbb{R}^L}\left[\|x - u\|_X + \langle y, -Au + b\rangle\right] \end{aligned}$$

where we defined $[A^{(n)}]_{i,j} = \langle a_i, p_j\rangle$. Since a linearly constrained optimization problem in $\mathbb{R}^n$ with convex objective function always admits a zero duality gap (see e.g. [3]), we have

$$\begin{aligned} d(x, \mathcal{L}_n) &= \max_{y\in\mathbb{R}^L}\min_{u\in X_n}\left[\|x - u\|_X + \langle y, -Au + b\rangle\right] \\ &= \max_{y\in\mathbb{R}^L}\left(\langle y, b - Ax\rangle + \min_{u\in X_n}\|x - u\|_X + \langle y, A(x - u)\rangle\right). \end{aligned}$$

Consider now the Lagrangian dual functional $\gamma(y) = \min_{u\in X_n}\|x - u\|_X + \langle y, A(x - u)\rangle$. If we define $A' : \mathbb{R}^L \to X'$ by $\langle y, Ax\rangle = \langle A'y, x\rangle$ $\forall x \in X, \forall y \in \mathbb{R}^L$, we can check that $A'y = \sum_{i=1}^L y_i a_i$. Denoting $\|A'y\|_{X',n} = \sup_{w\in X_n}\frac{|\langle A'y, w\rangle|}{\|w\|_X}$, it follows from the definition of the dual norm that $\gamma(y) = 0$ if $\|A'y\|_{X',n} \leq 1$ and $-\infty$ otherwise. We conclude that

$$\begin{aligned} d(x, \mathcal{L}_n) &= \max_{\{y \in \mathbb{R}^L \text{ s.t. } \|A'y\|_{X',n} \leq 1\}} \langle y, b - Ax \rangle \\ &\leq \max_{\{y \in \mathbb{R}^L \text{ s.t. } \|A'y\|_{X',n} \leq 1\}} \|y\|_{q'} \|b - Ax\|_q \ . \end{aligned}$$

Therefore, choosing a $\sigma_n > 0$ such that $\frac{1}{\sigma_n} = \max_{\{y \in \mathbb{R}^L \text{ s.t. } \|A'y\|_{X',n} \leq 1\}} \|y\|_{q'}$ ensures degeneracy of A, and we have

Lemma 3. *If $\sigma_n = \min_{\{y \in \mathbb{R}^L, \|y\|_{q'} = 1\}} \|A'y\|_{X',n}$ is strictly positive then operator $A \ : \ X_n \to \mathbb{R}^L$ is non-degenerate with constant σ_n.*

Remark 1. If X is a Hilbert space and if we work with the Euclidean norm for $\mathbb{R}^L$, we can obtain a more explicit non-degeneracy condition, by identifying all $x' \in X'$ with the corresponding element of X given by the Riesz representation theorem such that X' is identified with X. Suppose $\{p_1, \ldots, p_n\}$ is an orthonormal basis of X_n. Using $A^{(n)}$ as defined above, we have $\sup_{w \in X_n} \frac{|\langle A'y, w \rangle|}{\|w\|_X} = \sup_{\mathbf{w} \in \mathbb{R}^n} \frac{|\langle A^{(n)T}y, \mathbf{w} \rangle|}{\|\mathbf{w}\|_2} = \left\|A^{(n)T}y\right\|_2$. Furthermore, if $\cup X_n$ is dense in X, $\left\|A^{(n)T}y\right\|_2^2 = \sum_{j=1}^n \left(\langle \sum_{i=1}^L a_i y_i, p_j \rangle\right)^2$ converges to $\left\|A^T y\right\|_X^2 = \left\|\sum_{i=1}^L a_i y_i\right\|_X^2 = \lambda_{\min}(AA^T)$ when n tends to infinity. Operator $AA^T : \mathbb{R}^L \to \mathbb{R}^L$ is positive semidefinite and corresponds to a matrix with components $[AA^T]_{i,j} = \langle a_i, a_j \rangle$. It is therefore enough to assume it is nonsingular or, equivalently, the linear independence of all a_i in $X' = X$, to show that there exists N such that for all $n \geq N$, $\sigma_n > 0$.

We are now able to apply the Regularity Theorem to (P_n) and $(\mathrm{P}_{n,\epsilon})$.

Lemma 4. *Assume that*

1. *there exists $\hat{x} \in X_n$ such that $A\hat{x} = b$ and $\|\hat{x}\|_X < M$,*
2. *$\sigma_n = \min_{\{y \in \mathbb{R}^L, \|y\|_{q'} = 1\}} \|A'y\|_{X',n} > 0$.*

Then the optimal values of the problems (P_n) and $(\mathrm{P}_{n,\epsilon})$ satisfy for all $\epsilon > 0$ $P_n^* - \frac{\epsilon \|c\|_{X'}}{\sigma_n}\left(1 + \frac{2M}{M - \mathcal{R}_n}\right) \leq P_{n,\epsilon}^* \leq P_n^*$ with $\mathcal{R}_n = \min_{x \in X_n, Ax = b} \|x\|_X$.

4.3 Convergence result

In order to combine the two bounds we have obtained, we need to assume that hypotheses of Lemmas 1 and 4 are satisfied for some values of n. In fact,

- If there exists N_1 such that $\mathcal{R}_{N_1} < M$ then $\mathcal{R}_n < M$ for all $n \geq N_1$.
- If there exists N_2 such that $\sigma_{N_2} > 0$ then $\sigma_n > 0$ for all $n \geq N_2$.

Therefore, we have proved the following convergence result:

Theorem 3. *Assume that*

1. *the infinite-dimensional problem* (P) *is solvable*

2. there exists N_1 *and* $\hat{x} \in X_{N_1}$ *such that* $A\hat{x} = b$ *and* $\|\hat{x}\|_X < M$
3. there exists N_2 *such that* $\sigma_{N_2} > 0$.

Then we have for all $n \geq N = \max\{N_1, N_2\}$ *that*

$$P^* \leq P_n^* \leq P^* + 2E_n(x_{opt}) \|c\|_{X'} \left(1 + \frac{\left(\sum_{i=1}^{L} \|a_i\|_{X'}^q\right)^{1/q}}{\sigma_n} \left(1 + \frac{2M}{M - \mathcal{R}_n}\right)\right)$$

where x_{opt} *is an optimal solution of* (P) *and* $\mathcal{R}_n = \min_{x \in X_n, Ax=b} \|x\|_X$.

To summarize, we have obtained a convergence result for our polynomial approximation scheme provided that $E_n(x_{\text{opt}})$, the best approximation error of the optimal solution of (P), converges to zero when n goes to infinity, which is is a natural condition from the practical point of view. This holds for example if the linear subspace $\text{span}\{p_1, \ldots, p_n, \ldots\} = \cup_n X_n$ is dense in X.

5 Specific classes of infinite-dimensional problems

To conclude, we provide a few examples of specific functional spaces X and comment on their solvability and the expected rate of convergence described by Theorem 3.

X is the Lebesgue space L^q

These functional spaces are suitable for use in the supply problems considered in Section 2. Let Ω be a domain of $\mathbb{R}^N$ and $1 \leq q \leq \infty$. Let X be the Lebesgue space $L^q(\Omega) = \{u \in \mathcal{M}(\Omega) : \int_\Omega |u(t)|^q \, dt < +\infty\}$ with norm $\|u\|_X = \|u\|_q = \left(\int_\Omega |u(t)|^q \, dt\right)^{1/q}$ in the case $1 \leq q < \infty$, and $\|u\|_X = \|u\|_\infty = \text{ess}\sup_{t \in \Omega} |u(t)|$ when $q = \infty$. We take as linear and continuous functionals $c : L^q(\Omega) \to \mathbb{R},\ u \to \int_\Omega u(t)\gamma(t)dt$ and $a_i : L^q(\Omega) \to \mathbb{R},\ u \to \int_\Omega u(t)\alpha_i(t)dt$ where γ and $\alpha_i \in L^{q'}(\Omega)$ for all $i = 1, \ldots, L$.

Concerning the solvability of this problem, note that $L^q(\Omega)$ is reflexive for all $1 < q < \infty$ and that $L^\infty(\Omega) = (L^1(\Omega))'$ where $L^1(\Omega)$ is separable ([1]). Therefore, we can conclude that the infinite-dimensional problem has at least one optimal solution for all $1 < q \leq \infty$. Similar results can be otained if we consider the sequence space l^q.

If Ω is a bounded interval $[a, b]$ and $X_n = \text{span}\{1, t, \ldots, t^{n-1}\}$, we have the following well-known results about the convergence of the best polynomial approximation error of a function $u \in X$, see e.g. [11, 5]:

- $E_n(u)_q \to 0$ iff $u \in L^q([a, b])$ for all $1 \leq q < \infty$
- $E_n(u)_\infty \to 0$ iff $u \in \mathcal{C}([a, b])$
- $E_n(u)_q = O(\frac{1}{n^r})$ if $u \in \mathcal{C}^{r-1,r-1}([a, b])$ for all $1 < q \leq \infty$

where $E_n(u)_q = \inf_{v \in X_n} \|u - v\|_q$ and $\mathcal{C}^{k,r} = \{u \in \mathcal{C}^k([a, b])$ s.t $u^{(r)}$ is Lipschitz continuous $\}$ with $r \leq k$.

Recall that these quantities, that describe the best approximation error of the optimal solution of (P), have a direct influence on the convergence rate of P_n^* to P^* (cf. Theorem 3).

X is the Sobolev space $W^{k,q}$

If we want to include derivatives of our variable in the constraints or in the objective, we need to work in Sobolev spaces. Let Ω be a domain of $\mathbb{R}^N$, $1 \leq q \leq \infty$ and $k \in \mathbb{N}$. For all multi-indices $(\beta_1, \ldots, \beta_N) \in \mathbb{N}^N$, we note $|\beta| = \sum_{i=1}^{N} \beta_i$ and $D^\beta u = \frac{\partial^{|\beta|} u}{\partial t_1^{\beta_1} \ldots \partial t_N^{\beta_N}}$ in the weak sense. We choose for X the Sobolev space $W^{k,q}(\Omega) = \{u \in \mathcal{M}(\Omega) : D^\beta u \in L^q(\Omega) \quad \forall 0 \leq |\beta| \leq k\}$ with the norm $\|u\|_X = \|u\|_{k,q} = \left(\sum_{0 \leq |\beta| \leq k} \left\|D^\beta u\right\|_q^q\right)^{1/q}$ in the case $1 \leq q < \infty$ and $\|u\|_{k,\infty} = \max_{0 \leq |\beta| \leq k} \left\|D^\beta u\right\|_\infty$ when $q = \infty$. Our linear and continuous functionals are $c : W^{k,q}(\Omega) \to \mathbb{R},\ u \to \sum_{0 \leq |\beta| \leq k} \int_\Omega D^\beta u(t) \gamma_\beta(t) dt$ and $a_i : W^{k,q}(\Omega) \to \mathbb{R},\ u \to \sum_{0 \leq |\beta| \leq k} \int_\Omega D^\beta u(t) \alpha_{i,\beta}(t) dt$ where γ_β and $\alpha_{i,\beta} \in L^{q'}(\Omega)$ for all $i = 1, \ldots, L$ and for all $0 \leq |\beta| \leq k$.

Since the space $W^{k,q}$ is reflexive for all $k \in \mathbb{N}$ and for all $1 < q < \infty$ [1], existence of an optimal solution to (P) is guaranteed. Furthermore, when Ω is a bounded interval $[a, b]$, it is well-known that the polynomials are dense in the Sobolev space $W^{k,q}([a, b])$ for all $k \in \mathbb{N}$ and for all $1 \leq q < \infty$. Therefore, Theorem 3 guarantees convergence of the polynomial approximation scheme in this case.

References

1. R.A. Adams (2003) Sobolev Spaces: 2nd edition. Academic Press.
2. E.J. Anderson and P. Nash (1987) Linear Programming in infinite-dimensional spaces. Wiley.
3. S. Boyd and L. Vandenbergh (2009) Convex Optimization (7th printing with corrections). Cambdrige University Press.
4. E.W. Cheney (1982) Introduction to Approximation Theory (second edition). AMS Chelsea Publishing.
5. Z. Ditzian and V. Totik (1987) Moduli of smoothness. Springer-Verlag.
6. A. Gil, J. Segura and N.M. Temme (2007) Numerical Methods for special functions. SIAM.
7. M. Hinze, R. Pinnau, M. Ulbrich and S. Ulbrich (2009) Optimization with PDE constraints. Springer-Verlag.
8. D.G. Luenberger (1969) Optimization by vector space methods. Wiley.
9. Y.E. Nesterov and A.S. Nemirovskii (1994) Interior-Point Polynomial Algorithms in Convex Programming. SIAM.
10. Y.E. Nesterov (2000) Squared functional systems and optimization problems. In High performance Optimization, pages 405-440. Kluwer Academic Publishers.
11. A.F. Timan (1963) Theory of approximation of functions of a real variable. Pergamon Press.
12. E. Zeidler (1985) Nonlinear Functional Analysis and its Applications (Part 3): Variational Methods and Optimization. Springer-Verlag.

Abstract Cones of Positive Polynomials and Their Sums of Squares Relaxations

Roland Hildebrand[1]

[1] LJK, Université Grenoble 1 / CNRS, 51 rue des Mathématiques, BP53, 38041 Grenoble cedex, France roland.hildebrand@imag.fr

Summary. We present a new family of sums of squares (SOS) relaxations to cones of positive polynomials. The SOS relaxations employed in the literature are cones of polynomials which can be represented as ratios, with an SOS as numerator and a fixed positive polynomial as denominator. We employ nonlinear transformations of the arguments instead. A fixed cone of positive polynomials, considered as a subset in an abstract coefficient space, corresponds to an infinite, partially ordered set of concrete cones of positive polynomials of different degrees and in a different number of variables. To each such concrete cone corresponds its own SOS cone, leading to a hierarchy of increasingly tighter SOS relaxations for the abstract cone.

1 Introduction

Many optimization problems can be recast as conic programs over a cone of positive polynomials on $\mathbf{R}^n$. Cones of positive polynomials cannot be described efficiently in general, and the corresponding conic programs are NP-hard. Hence approximations have to be employed to obtain suboptimal solutions. A standard approach is to approximate the cone of positive polynomials from inside by the cone of sums of squares (SOS), i.e. the cone of those polynomials which are representable as a sum of squares of polynomials of lower degree. The SOS cone is semidefinite representable, and conic programs over this cone can be cast as efficiently solvable semidefinite programs. This approximation is not exact, however, even for polynomials of degree 6 in two variables [6], as the famous example of the Motzkin polynomial [1] shows. Tighter approximations can be obtained when using the cone of polynomials which can be represented as ratios, with the numerator being a sum of squares of polynomials, and the denominator a fixed positive polynomial. Usually this fixed polynomial is chosen to be $\left(\sum_{k=1}^n x_k^2\right)^d$ for some integer $d > 0$ [2].

We propose another family of SOS based relaxations of cones of positive polynomials. We consider the cone of positive polynomials not as a cone of functions, but rather as a subset in an abstract coefficient space. The same

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*, DOI 10.1007/978-3-642-12598-0_4, © Springer-Verlag Berlin Heidelberg 2010

abstract cone then corresponds to an infinite number of concrete cones of positive polynomials of different degrees and in a different number of variables. To each such concrete cone corresponds its own SOS cone, and these SOS cones are in general different for different realizations of the abstract cone. We present a computationally efficient criterion to compare the different SOS cones and introduce a corresponding equivalence relation and a partial order on the set of these SOS cones. This allows us to build hierarchies of increasingly tighter semidefinite relaxations for the abstract cone, and thus also for the original cone of positive polynomials. We show on the example of the cone of positive polynomials containing the Motzkin polynomial that our hierarchy of relaxations possesses the capability of being exact at a finite step.

The remainder of the contribution is structured as follows. In the next section we define notation that will be used in the paper. In Sect. 3 we define and analyze the considered cones of positive polynomials. In Sect. 4 we consider sums of squares relaxations of these cones and study their properties. In Sect. 5 we define the abstract cones of positive polynomials and their SOS relaxations and establish a hierarchical structure on the set of these relaxations. Finally, we demonstrate the developed apparatus on the example of the cone containing the Motzkin polynomial in Sect. 6.

2 Notation

For a finite set S, denote by $\# S$ the cardinality of S.

For a subset A of a real vector space V, denote by $\mathrm{cl}\, A$ the closure, by $\mathrm{int}\, A$ the interior, by $\mathrm{aff}\, A$ the affine hull, by $\mathrm{conv}\, A$ the convex hull, and by $\mathrm{con}\,\mathrm{cl}\, A$ the set $\mathrm{cl} \cup_{\alpha \geq 0} \alpha A$. If A is a convex polytope, denote by $\mathrm{extr}\, A$ the set of its vertices.

Let $\mathcal{S}(m)$ denote the space of real symmetric matrices of size $m \times m$, and $\mathcal{S}_+(m) \subset \mathcal{S}(m)$ the cone of positive semidefinite (PSD) matrices. By I_n denote the $n \times n$ identity matrix. Let $\pi_2 : \mathbf{Z} \to \mathbf{F}_2$ be the ring homomorphism from the integers onto the field $\mathbf{F}_2 = (\{0,1\}, +, \cdot)$ (mapping even integers to 0 and odd ones to 1), and $\pi_2^n : \mathbf{Z}^n \to \mathbf{F}_2^n$ the corresponding homomorphism of the product rings, acting as $\pi_2^n : (a_1, \dots, a_n) \mapsto (\pi_2(a_1), \dots, \pi_2(a_n))$. For an integer matrix M, let $\pi_2[M]$ be the matrix obtained by element-wise application of π_2 to M. The corresponding $\mathbf{F}_2$-linear map will also be denoted by $\pi_2[M]$. For a linear map M, let $Im\, M$ be the image of M in the target space.

Let $\mathcal{A} \subset \mathbf{N}^n$ be an ordered finite set of multi-indices of length n, considered as row vectors. Denote by $\Gamma_{\mathcal{A}} = \{\sum_{\alpha\in\mathcal{A}} a_\alpha \alpha \,|\, a_\alpha \in \mathbf{Z}\ \forall\ \alpha \in \mathcal{A}\} \subset \mathbf{Z}^n$ the lattice generated by $\mathcal{A}$ in $\mathrm{aff}\, \mathcal{A}$, and let $\Gamma_{\mathcal{A}}^e \subset \Gamma_{\mathcal{A}}$ be the sublattice of even points. For $x = (x_1, \dots, x_n)^T \in \mathbf{R}^n$, denote by $X_{\mathcal{A}}(x)$ the corresponding vector of monomials $(x^\alpha)_{\alpha\in\mathcal{A}}$, and define the set $\mathcal{X}_{\mathcal{A}} = \{X_{\mathcal{A}}(x) \,|\, x \in \mathbf{R}^n\}$. By $\mathcal{L}_{\mathcal{A}}$ we denote the real vector space of polynomials $p(x) = \sum_{\alpha\in\mathcal{A}} c_\alpha x^\alpha$. There exists a canonical isomorphism $\mathcal{I}_{\mathcal{A}} : \mathcal{L}_{\mathcal{A}} \to \mathbf{R}^{\#\mathcal{A}}$, which maps a polynomial $p \in \mathcal{L}_{\mathcal{A}}$ to its coefficient vector $\mathcal{I}_{\mathcal{A}}(p) = (c_\alpha(p))_{\alpha\in\mathcal{A}}$.

3 Cones of positive polynomials

Let $\mathcal{A} \subset \mathbf{N}^n$ be an ordered finite set of multi-indices. We call a polynomial $p \in \mathcal{L}_\mathcal{A}$ *positive* if $p(x) = \langle \mathcal{I}_\mathcal{A}(p), X_\mathcal{A}(x) \rangle \geq 0$ for all $x \in \mathbf{R}^n$. The positive polynomials form a closed convex cone $\mathcal{P}_\mathcal{A}$. This cone cannot contain a line, otherwise the monomials x^α, $\alpha \in \mathcal{A}$, would be linearly dependent.

Let $p \in \mathcal{L}_\mathcal{A}$ be a polynomial. The convex hull of all indices $\alpha \in \mathcal{A}$ such that $c_\alpha(p) \neq 0$, viewed as vectors in $\mathbf{R}^n$, forms a convex polytope. This polytope is called the *Newton polytope* of p and is denoted by $N(p)$. The convex hull of the whole multi-index set $\mathcal{A}$, viewed as a subset of the integer lattice in $\mathbf{R}^n$, will be called the *Newton polytope* associated with the linear space $\mathcal{L}_\mathcal{A}$ and denoted by $N_\mathcal{A}$. Obviously we have the relation $N_\mathcal{A} = \cup_{p \in \mathcal{L}_\mathcal{A}} N(p)$. Newton polytopes of polynomials in $p \in \mathcal{L}_\mathcal{A}$ have the following property.

Lemma 1. *[4, p.365] Assume above notation and let $p \in \mathcal{P}_\mathcal{A}$. If $\alpha \in \mathcal{A}$ is an extremal point of $N(p)$, then α is even and $c_\alpha(p) > 0$.*

Without restriction of generality we henceforth assume that

$$\text{all indices in extr}\, N_\mathcal{A} \text{ have even entries,} \tag{1}$$

otherwise the cone $\mathcal{P}_\mathcal{A}$ is contained in a proper subspace of $\mathcal{L}_\mathcal{A}$.

Lemma 2. *Under assumption (1), the cone $\mathcal{P}_\mathcal{A}$ has nonempty interior.*

Proof. Let us show that the polynomial $p(x) = \sum_{\alpha \in \text{extr} N_\mathcal{A}} x^\alpha$ is an interior point of $\mathcal{P}_\mathcal{A}$.

Since the logarithm is a concave function, we have for every integer $N > 0$, every set of reals $\lambda_1, \dots, \lambda_N \geq 0$ such that $\sum_{k=1}^N \lambda_k = 1$, and every set of reals $a_1, \dots, a_N > 0$ that $\log \sum_{k=1}^N \lambda_k a_k \geq \sum_{k=1}^N \lambda_k \log a_k$. It follows that $\log \sum_{k=1}^N a_k \geq \sum_{k=1}^N \lambda_k \log a_k$ and therefore $\sum_{k=1}^N a_k \geq \prod_{k=1}^N a_k^{\lambda_k}$.

Let now $\alpha^1, \dots, \alpha^N$ be the extremal points of $N_\mathcal{A}$, and let $\alpha = \sum_{k=1}^N \lambda_k \alpha^k \in \mathcal{A}$ be an arbitrary index, represented as a convex combination of the extremal points. By the above, we then have for every $x \in \mathbf{R}^n$ satisfying $\Pi_{l=1}^n x_l \neq 0$ that $\sum_{k=1}^N x^{\alpha^k} \geq \prod_{k=1}^N (x^{\alpha^k})^{\lambda_k} = |x|^\alpha = |x^\alpha|$. By continuity this holds also for x such that $\Pi_{l=1}^n x_l = 0$. Thus the polynomial $p(x) + q(x)$ is positive, as long as the 1-norm of the coefficient vector $\mathcal{I}_\mathcal{A}(q)$ does not exceed 1.

It follows that both the cone $\mathcal{P}_\mathcal{A}$ and its dual are regular cones, i.e. closed convex cones with nonempty interior, containing no lines.

Lemma 3. *Under assumption (1), $(\mathcal{I}_\mathcal{A}[\mathcal{P}_\mathcal{A}])^* = \text{conv}(\text{con}\,\text{cl}\mathcal{X}_\mathcal{A})$.*

Proof. Clearly $p \in \mathcal{P}_\mathcal{A}$ if and only if for all $y \in \text{con}\,\text{cl}\mathcal{X}_\mathcal{A}$ we have $\langle \mathcal{I}_\mathcal{A}(p), y \rangle \geq 0$. Hence $\mathcal{I}_\mathcal{A}[\mathcal{P}_\mathcal{A}]$ is the dual cone of the convex hull $\text{conv}(\text{con}\,\text{cl}\mathcal{X}_\mathcal{A})$.

It rests to show that this convex hull is closed. Let z be a vector in the interior of the cone $\mathcal{I}_\mathcal{A}[\mathcal{P}_\mathcal{A}]$. Such a vector exists by the preceding lemma. Then the set $C = \{y \in \mathrm{con}\,\mathrm{cl}\mathcal{X}_{\mathcal{A}} \mid \langle y, z\rangle = 1\}$ is compact, and hence its convex hull $\mathrm{conv}\,C$ is closed. But $\mathrm{conv}(\mathrm{con}\,\mathrm{cl}\mathcal{X}_{\mathcal{A}})$ is the conic hull of $\mathrm{conv}\,C$, and therefore also closed. Thus $(\mathcal{I}_{\mathcal{A}}[\mathcal{P}_{\mathcal{A}}])^* = (\mathrm{conv}(\mathrm{con}\,\mathrm{cl}\mathcal{X}_{\mathcal{A}}))^{**} = \mathrm{conv}(\mathrm{con}\,\mathrm{cl}\mathcal{X}_{\mathcal{A}})$.

We shall now analyze the set $\mathrm{con}\,\mathrm{cl}\mathcal{X}_{\mathcal{A}}$, which is, as can be seen from the previous lemma, determining the cone $\mathcal{P}_{\mathcal{A}}$.

For every ordered index set $\mathcal{A}$ with elements $\alpha^1, \dots, \alpha^m \in \mathbf{N}^n$, where each multi-index is represented by a row vector $\alpha^k = (\alpha^k_1, \dots, \alpha^k_n)$, define the $m \times n$ matrix $M_{\mathcal{A}} = (\alpha^k_l)_{k=1,\dots,m;l=1,\dots,n}$. Further define $\alpha^k_0 = 1$, $k = 1, \dots, m$ and the $m \times (n+1)$ matrix $M'_{\mathcal{A}} = (\alpha^k_l)_{k=1,\dots,m;l=0,\dots,n}$.

Lemma 4. *Assume above notation. Then*

$$\mathrm{con}\,\mathrm{cl}\mathcal{X}_{\mathcal{A}} = \mathrm{cl}\{(-1)^\delta \circ \exp(y) \mid \delta \in Im\,\pi_2[M_{\mathcal{A}}],\ y \in Im\,M'_{\mathcal{A}}\},$$

where both $(-1)^\delta$ and $\exp(y)$ are understood element-wise, and $\circ$ denotes the Hadamard product of vectors.

Proof. The space $\mathbf{R}^n$ is composed of 2^n orthants O_γ, which can be indexed by the vectors in $\mathbf{F}_2^n$. Here the index $\gamma = (\gamma_1, \dots, \gamma_n)^T$ of the orthant O_γ is defined such that $\mathrm{sgn}\,x = (-1)^\gamma$ for all $x \in \mathrm{int}\,O_\gamma$, where both $\mathrm{sgn}\,x$ and $(-1)^\gamma$ have to be understood element-wise. In a similar way, the 2^m orthants of $\mathbf{R}^m$ are indexed by the elements of $\mathbf{F}_2^m$.

We shall now compute the set $T_\gamma = \{\beta X_{\mathcal{A}}(x) \mid \beta > 0,\ x \in \mathrm{int}O_\gamma\} \subset \mathbf{R}^m$.

First observe that the signs of the components of $\beta X_{\mathcal{A}}(x)$ do not depend on β and on $x \in \mathrm{int}O_\gamma$. Namely, the k-th component equals $\beta \prod_{l=1}^n x_l^{\alpha_l^k}$, and its sign is $(-1)^{\delta_k}$, where $\delta_k = \sum_{l=1}^n \pi_2(\alpha^k_l)\gamma_l$. Therefore, T_γ is contained in the interior of the orthant O_δ, where $\delta = (\delta_1, \dots, \delta_m)^T = \pi_2[M_{\mathcal{A}}](\gamma) \in \mathbf{F}_2^m$. Thus, if γ runs through $\mathbf{F}_2^n$, then the indices of the orthants containing T_γ run through $Im\,\pi_2[M_{\mathcal{A}}]$.

Consider the absolute values of the components of $\beta X_{\mathcal{A}}(x)$. The logarithm of the modulus of the k-th component is given by $\log\beta + \sum_{l=1}^n \alpha^k_l \log|x_l|$. Now the vector $(\log\beta, \log|x_1|, \dots, \log|x_n|)^T$ runs through $\mathbf{R}^{n+1}$ if (β, x) runs through $\mathrm{int}\mathbf{R}_+ \times \mathrm{int}O_\gamma$, and therefore the element-wise logarithm of the absolute values of $\beta X_{\mathcal{A}}(x)$ runs through $Im\,M_{\mathcal{A}}$, independently of γ.

We have proven the relation

$$\{\beta X_{\mathcal{A}}(x) \mid \beta > 0,\ \prod_{l=1}^n x_l \neq 0\} = \{(-1)^\delta \circ e^y \mid \delta \in Im\,\pi_2[M_{\mathcal{A}}],\ y \in Im\,M'_{\mathcal{A}}\} \tag{2}$$

It rests to show that the closure of the left-hand side equals $\mathrm{con}\,\mathrm{cl}\mathcal{X}_{\mathcal{A}}$. Clearly this closure is contained in $\mathrm{con}\,\mathrm{cl}\mathcal{X}_{\mathcal{A}}$. The converse inclusion follows from the continuity of the map $(\beta, x) \mapsto \beta X_{\mathcal{A}}(x)$ on $\mathbf{R} \times \mathbf{R}^n$ and the fact that the set $\{(\beta, x) \mid \beta > 0,\ \prod_{l=1}^n x_l \neq 0\}$ is dense in $\mathbf{R}_+ \times \mathbf{R}^n$. This concludes the proof.

The description of $\mathrm{con}\,\mathrm{cl}\mathcal{X}_{\mathcal{A}}$ given by Lemma 4 allows us to relate these sets for different index sets $\mathcal{A}$.

Theorem 1. *Assume above notation. Let $\mathcal{A} = \{\alpha^1, \dots, \alpha^m\} \subset \mathbf{N}^n$, $\mathcal{A}' = \{\alpha'^1, \dots, \alpha'^m\} \subset \mathbf{N}^{n'}$ be nonempty ordered multi-index sets satisfying assumption (1). Then the following are equivalent.*

1) $\operatorname{con}\operatorname{cl}\mathcal{X}_{\mathcal{A}} = \operatorname{con}\operatorname{cl}\mathcal{X}_{\mathcal{A}'}$*,*

2) $\mathit{Im}\, M'_{\mathcal{A}} = \mathit{Im}\, M'_{\mathcal{A}'}$ *and* $\mathit{Im}\, \pi_2[M_{\mathcal{A}}] = \mathit{Im}\, \pi_2[M_{\mathcal{A}'}]$*,*

3) the order isomorphism $I_A : \mathcal{A} \to \mathcal{A}'$ can be extended to a bijective, affine map $R : \operatorname{aff}\mathcal{A} \to \operatorname{aff}\mathcal{A}'$, and there exists a bijective linear map Z : $\operatorname{span}(\pi_2^n[\mathcal{A}]) \to \operatorname{span}(\pi_2^{n'}[\mathcal{A}'])$ *such that* $(Z \circ \pi_2^n)(\alpha^k) = \pi_2^{n'}(\alpha'^k)$*,* $k = 1, \dots, m$*,*

4) the order isomorphism $I_A : \mathcal{A} \to \mathcal{A}'$ can be extended to a lattice isomorphism $I_\Gamma : \Gamma_{\mathcal{A}} \to \Gamma_{\mathcal{A}'}$, and $I_\Gamma[\Gamma^e_{\mathcal{A}}] = \Gamma^e_{\mathcal{A}'}$.

Moreover, the following is a consequence of 1) — 4).

5) $\mathcal{I}_{\mathcal{A}}[\mathcal{P}_{\mathcal{A}}] = \mathcal{I}_{\mathcal{A}'}[\mathcal{P}_{\mathcal{A}'}]$*.*

Proof. 1) ⇔ 2): Denote set (2) by $S(\mathcal{A})$. This set is contained in $\cup_{\delta \in \mathbf{F}_2^m} O_\delta$ and is closed in its relative topology. Hence $S(\mathcal{A}) = (\operatorname{cl} S(\mathcal{A})) \cap \left(\cup_{\delta \in \mathbf{F}_2^m} O_\delta\right)$. Therefore, if condition 2) is not satisfied, then $S(\mathcal{A}) \neq S(\mathcal{A}')$, and hence $\operatorname{cl} S(\mathcal{A}) \neq \operatorname{cl} S(\mathcal{A}')$, which implies by Lemma 4 that condition 1) is not satisfied. On the other hand, if condition 2) is satisfied, then $S(\mathcal{A}) = S(\mathcal{A}')$, and again by Lemma 4 condition 1) is satisfied.

2) ⇔ 3): The first relation in condition 2) is equivalent to the coincidence of the kernels of ${M'}^T_{\mathcal{A}}$ and ${M'}^T_{\mathcal{A}'}$. But these kernels define exactly all affine dependencies between the elements of $\mathcal{A}$ and $\mathcal{A}'$, respectively. Therefore $\ker {M'}^T_{\mathcal{A}} = \ker {M'}^T_{\mathcal{A}'}$ if and only if there exists an isomorphism R between the affine spaces $\operatorname{aff}\mathcal{A}$ and $\operatorname{aff}\mathcal{A}'$ that takes α^k to α'^k, $k = 1, \dots, m$. The equivalence of the second relation in condition 2) and the existence of the map Z is proven similarly.

3) ⇔ 4): Clearly the map R in 3) defines the sought lattice isomorphism $I_\Gamma : \Gamma_{\mathcal{A}} \to \Gamma_{\mathcal{A}'}$. On the other hand, the existence of I_Γ implies that $\ker {M'}^T_{\mathcal{A}} \cap \mathbf{Z}^m = \ker {M'}^T_{\mathcal{A}'} \cap \mathbf{Z}^m$. For the kernel of an integer matrix, however, one can always find an integer basis. Therefore it follows that $\ker {M'}^T_{\mathcal{A}} = \ker {M'}^T_{\mathcal{A}'}$, and I_Γ can be extended to an affine isomorphism $R : \operatorname{aff}\mathcal{A} \to \operatorname{aff}\mathcal{A}'$. We have shown equivalence of the first conditions in 3) and 4).

Note that π_2^n maps the lattice $\Gamma_{\mathcal{A}}$ to $\operatorname{aff}(\pi_2^n[\mathcal{A}])$, and likewise, $\pi_2^{n'}$ maps $\Gamma_{\mathcal{A}'}$ to $\operatorname{aff}(\pi_2^{n'}[\mathcal{A}'])$. Since both $\mathcal{A}, \mathcal{A}'$ satisfy (1), these sets contain at least one even point. Hence the images $\pi_2^n[\mathcal{A}], \pi_2^{n'}[\mathcal{A}']$ contain the origin, and the affine spans of these images are actually linear spans. Let us now assume that I_Γ exists and consider the diagram

$$\begin{tikzcd}
\Gamma_{\mathcal{A}} \arrow[r, "I_\Gamma"] \arrow[d, "\pi_2^n"'] & \Gamma_{\mathcal{A}'} \arrow[d, "\pi_2^{n'}"'] \\
\operatorname{span}(\pi_2^n[\mathcal{A}]) \arrow[r, "Z"] & \operatorname{span}(\pi_2^{n'}[\mathcal{A}'])
\end{tikzcd}$$

If there exists a linear map Z as in 3), then it makes the diagram commute. The relation $I_\Gamma[\Gamma^e_{\mathcal{A}}] = \Gamma^e_{\mathcal{A}'}$ now follows from the fact that Z maps the origin to the origin. On the other hand, let $I_\Gamma[\Gamma^e_{\mathcal{A}}] = \Gamma^e_{\mathcal{A}'}$. Since $\Gamma^e_{\mathcal{A}} \neq \emptyset$, we have that I_Γ takes pairs of points with even difference to pairs of points with even difference. This implies that there exists a well-defined map Z which makes the diagram commute. Moreover, Z takes the origin to the origin. Since I_Γ is affine, Z must also be affine and hence linear. Finally, repeating the argument with I_Γ^{-1} instead of I_Γ, we see that Z must be invertible.

Finally, the implication 1) ⇒ 5) is a direct consequence of Lemma 3.

4 Sums of squares relaxations

Let $\mathcal{A} = \{\alpha^1, \dots, \alpha^m\} \subset \mathbf{N}^n$ be an ordered multi-index set satisfying (1). A polynomial $p \in \mathcal{L}_{\mathcal{A}}$ is certainly positive if it can be represented as a finite sum of squares of other polynomials. The set of polynomials representable in this way forms a closed convex cone [5], the sums of squares cone

$$\Sigma_{\mathcal{A}} = \left\{ p \in \mathcal{L}_{\mathcal{A}} \,|\, \exists\; N,\; q_1, \dots, q_N :\; p = \sum_{k=1}^N q_k^2 \right\} \subset \mathcal{P}_{\mathcal{A}}. \qquad (3)$$

The SOS cone is semidefinite representable, and therefore a semidefinite relaxation of the cone $\mathcal{P}_{\mathcal{A}}$. We will henceforth call the cone $\Sigma_{\mathcal{A}}$ the *standard* SOS cone, or the *standard* SOS relaxation. In general we have $\Sigma_{\mathcal{A}} \neq \mathcal{P}_{\mathcal{A}}$, and we will see in Subsection 4.1 that we might even have $\dim \Sigma_{\mathcal{A}} \neq \dim \mathcal{P}_{\mathcal{A}}$.

We shall now generalize the notion of the SOS cone $\Sigma_{\mathcal{A}}$. Let $\mathcal{F} = \{\beta^1, \dots, \beta^{m'}\} \subset \mathbf{N}^n$ be an ordered multi-index set. We then define the set

$$\begin{aligned} \Sigma_{\mathcal{F},\mathcal{A}} &= \left\{ p \in \mathcal{L}_{\mathcal{A}} \,|\, \exists\; N,\; q_1, \dots, q_N \in \mathcal{L}_{\mathcal{F}} : p = \sum_{k=1}^N q_k^2 \right\}, \\ &= \left\{ p \in \mathcal{L}_{\mathcal{A}} \,|\, \exists\; C = C^T \succeq 0 : p(x) = X_{\mathcal{F}}^T(x) C X_{\mathcal{F}}(x) \right\}, \end{aligned} \qquad (4)$$

which is also a semidefinite representable closed convex cone. Obviously we have the inclusion $\Sigma_{\mathcal{F},\mathcal{A}} \subset \Sigma_{\mathcal{A}}$. The next result shows that the standard SOS cone $\Sigma_{\mathcal{A}}$ is actually an element of the family $\{\Sigma_{\mathcal{F},\mathcal{A}}\}_{\mathcal{F} \subset \mathbf{N}^n}$ of cones.

Lemma 5. *[4, p.365] If the polynomial $p(x) = \sum_{k=1}^N q_k^2(x)$ is a sum of squares, then for every polynomial q_k participating in the SOS decomposition of p we have $2N(q_k) \subset N(p)$.*

It follows that for every $p(x) = \sum_{k=1}^N q_k(x)^2 \in \Sigma_{\mathcal{A}}$, the nonzero coefficients of every polynomial q_k have multi-indices lying in the polytope $\frac{1}{2}N_{\mathcal{A}}$. Thus $\Sigma_{\mathcal{A}} = \Sigma_{\mathcal{F}_{\max}(\mathcal{A}),\mathcal{A}}$, where $\mathcal{F}_{\max}(\mathcal{A}) = (\frac{1}{2}N_{\mathcal{A}}) \cap \mathbf{N}^n$. We get the following result.

Proposition 1. *Assume above notation. Then for every finite multi-index set $\mathcal{F} \subset \mathbf{N}^n$ such that $\mathcal{F}_{\max}(\mathcal{A}) \subset \mathcal{F}$ we have $\Sigma_{\mathcal{F},\mathcal{A}} = \Sigma_{\mathcal{A}}$.*

In general, the smaller $\mathcal{F}$, the weaker will be the relaxation $\Sigma_{\mathcal{F},\mathcal{A}}$. It does not make sense, however, to choose $\mathcal{F}$ larger than $\mathcal{F}_{\max}(\mathcal{A})$. Let us define the following partial order on the relaxations $\Sigma_{\mathcal{F},\mathcal{A}}$.

Definition 1. *Assume above notation and let the multi-index sets $\mathcal{F}_1, \mathcal{F}_2$ be subsets of $\mathcal{F}_{\max}(\mathcal{A})$. If $\mathcal{F}_1 \subset \mathcal{F}_2$, then we say that the relaxation $\Sigma_{\mathcal{F}_1,\mathcal{A}}$ of the cone $\mathcal{P}_{\mathcal{A}}$ is* coarser *than $\Sigma_{\mathcal{F}_2,\mathcal{A}}$, or $\Sigma_{\mathcal{F}_2,\mathcal{A}}$ is* finer *than $\Sigma_{\mathcal{F}_1,\mathcal{A}}$.*

A finer relaxation is tighter, but a strictly finer relaxation does not a priori need to be strictly tighter. The standard SOS relaxation $\Sigma_{\mathcal{A}}$ is then the finest relaxation among all relaxations of type (4).

We can make the semidefinite representation of $\Sigma_{\mathcal{F},\mathcal{A}}$ explicit by comparing the coefficients in the relation $p(x) = X_{\mathcal{F}}^T(x)CX_{\mathcal{F}}(x)$ appearing in definition (4). As it stands, this relation determines the polynomial $p(x)$ as a function of the symmetric matrix C, thus defining a linear map $L_{\mathcal{F},\mathcal{A}} : \mathcal{S}(m') \to \mathcal{L}_{(\mathcal{F}+\mathcal{F})\cup\mathcal{A}}$ by

$$c_\alpha(p) = \sum_{k,k' :\, \beta^k + \beta^{k'} = \alpha} C_{kk'}, \qquad \alpha \in (\mathcal{F}+\mathcal{F}) \cup \mathcal{A}.$$

Thus we obtain the description

$$\Sigma_{\mathcal{F},\mathcal{A}} = \mathcal{L}_{\mathcal{A}} \cap L_{\mathcal{F},\mathcal{A}}[\mathcal{S}_+(m')], \tag{5}$$

revealing $\Sigma_{\mathcal{F},\mathcal{A}}$ as a linear section of a linear image of the PSD cone $\mathcal{S}_+(m')$.

Note that the linear map $L_{\mathcal{F},\mathcal{A}}$ is completely determined by the map $s_{\mathcal{F},\mathcal{A}} : \mathcal{F} \times \mathcal{F} \to (\mathcal{F}+\mathcal{F}) \cup \mathcal{A}$ defined by $s_{\mathcal{F},\mathcal{A}}(\beta^k, \beta^{k'}) = \beta^k + \beta^{k'}$. Denote by $\mathrm{incl}_{\mathcal{A}} : \mathcal{A} \to (\mathcal{F}+\mathcal{F}) \cup \mathcal{A}$ the inclusion map. We then have the following result.

Theorem 2. *Let $\mathcal{F} = \{\beta^1, \dots, \beta^{m'}\}, \mathcal{A} = \{\alpha^1, \dots, \alpha^m\} \subset \mathbf{N}^n$, $\mathcal{F}' = \{\beta'^1, \dots, \beta'^{m'}\}, \mathcal{A}' = \{\alpha'^1, \dots, \alpha'^m\} \subset \mathbf{N}^{n'}$, be ordered multi-index sets satisfying $\mathcal{F} \subset \mathcal{F}_{\max}(\mathcal{A})$, $\mathcal{F}' \subset \mathcal{F}_{\max}(\mathcal{A}')$, and let $I_F : \mathcal{F} \to \mathcal{F}'$, $I_A : \mathcal{A} \to \mathcal{A}'$ be the order isomorphisms. Suppose that there exists a bijective map I that makes the following diagram commutative:*

$$\begin{tikzcd}
\mathcal{F} \times \mathcal{F} \arrow[r, "s_{\mathcal{F},\mathcal{A}}"] \arrow[d, "I_F \times I_F"'] & (\mathcal{F}+\mathcal{F}) \cup \mathcal{A} \arrow[d, "I"'] & \mathcal{A} \arrow[l, "\mathrm{incl}_{\mathcal{A}}"'] \arrow[d, "I_A"'] \\
\mathcal{F}' \times \mathcal{F}' \arrow[r, "s_{\mathcal{F}',\mathcal{A}'}"] & (\mathcal{F}'+\mathcal{F}') \cup \mathcal{A}' & \mathcal{A}' \arrow[l, "\mathrm{incl}_{\mathcal{A}'}"']
\end{tikzcd}$$

Then $\mathcal{I}_{\mathcal{A}}[\Sigma_{\mathcal{F},\mathcal{A}}] = \mathcal{I}_{\mathcal{A}'}[\Sigma_{\mathcal{F}',\mathcal{A}'}]$.

4.1 Dimensional considerations

From (5) it follows that the cone $\Sigma_{\mathcal{F},\mathcal{A}}$ is always contained in the linear subspace $\mathcal{L}_{(\mathcal{F}+\mathcal{F})\cap\mathcal{A}} \subset \mathcal{L}_{\mathcal{A}}$ and thus is better viewed as a relaxation of the cone $\mathcal{P}_{(\mathcal{F}+\mathcal{F})\cap\mathcal{A}}$ rather than of $\mathcal{P}_{\mathcal{A}}$ itself. In view of Lemma 2 a necessary condition for the cone $\Sigma_{\mathcal{F},\mathcal{A}}$ to have the same dimension as $\mathcal{P}_\mathcal{A}$ is thus the inclusion $\mathcal{A} \subset \mathcal{F} + \mathcal{F}$.

A natural question is now whether this inclusion is always satisfied by the multi-index set $\mathcal{F} := \mathcal{F}_{\max}(\mathcal{A}) = (\frac{1}{2}N_\mathcal{A}) \cap \mathbf{N}^n$, which gives rise to the standard SOS cone $\Sigma_\mathcal{A}$. The answer to this question is negative, as the example $\mathcal{A} = \{(2,0,0),(0,2,0),(2,2,0),(0,0,4),(1,1,1)\}$ taken from [4, p.373] shows.

It is, however, not hard to show that if $\mathcal{A}$ is contained in a 2-dimensional affine plane, then $\mathcal{A} \subset \mathcal{F}_{\max}(\mathcal{A}) + \mathcal{F}_{\max}(\mathcal{A})$.

5 Hierarchies of relaxations

In this section we construct hierarchies of semidefinite relaxations of the cone of positive polynomials which are tighter than the standard SOS relaxation.

Conditions 2) — 4) of Theorem 1 define an easily verifiable equivalence relation $\sim_P$ on the class of finite ordered multi-index sets satisfying (1). By Theorem 1, we have for any two equivalent multi-index sets $\mathcal{A} \sim_P \mathcal{A}'$ that $\mathcal{I}_\mathcal{A}[\mathcal{P}_\mathcal{A}] = \mathcal{I}_{\mathcal{A}'}[\mathcal{P}_{\mathcal{A}'}]$. It is therefore meaningful to define the abstract cone

$$\mathcal{P}_{[\mathcal{A}]} = \mathcal{I}_\mathcal{A}[\mathcal{P}_\mathcal{A}] = \{\mathcal{I}_\mathcal{A}(p) \,|\, p \in \mathcal{P}_\mathcal{A}\} \subset \mathbf{R}^m,$$

where $[\mathcal{A}]$ is the equivalence class of $\mathcal{A}$ with respect to the relation $\sim_P$. The points of this cone cannot anymore be considered as polynomials on $\mathbf{R}^n$. A cone of positive inhomogeneous polynomials on $\mathbf{R}^n$, e.g., corresponds to the same abstract cone as the cone of their homogenizations, which are defined on $\mathbf{R}^{n+1}$. For every concrete choice of a representative $\mathcal{A}' \in [\mathcal{A}]$, however, the map $\mathcal{I}_{\mathcal{A}'}^{-1}$ puts them in correspondence with positive polynomials in $\mathcal{P}_{\mathcal{A}'}$.

Similarly, the existence of the bijective map I in Theorem 2 defines an easily verifiable equivalence relation $\sim_\Sigma$ on the class of pairs $(\mathcal{F},\mathcal{A})$ of ordered finite multi-index sets satisfying $\mathcal{F} \subset \mathcal{F}_{\max}(\mathcal{A})$. By Theorem 2, for any two equivalent pairs $(\mathcal{F},\mathcal{A}) \sim_\Sigma (\mathcal{F}',\mathcal{A}')$ we have $\mathcal{I}_\mathcal{A}[\Sigma_{\mathcal{F},\mathcal{A}}] = \mathcal{I}_{\mathcal{A}'}[\Sigma_{\mathcal{F}',\mathcal{A}'}]$. We can then define the abstract cone $\Sigma_{[(\mathcal{F},\mathcal{A})]} = \mathcal{I}_\mathcal{A}[\Sigma_{\mathcal{F},\mathcal{A}}] \subset \mathbf{R}^m$, where $[(\mathcal{F},\mathcal{A})]$ is the equivalence class of the pair $(\mathcal{F},\mathcal{A})$ with respect to the relation $\sim_\Sigma$. For every concrete choice of a representative $(\mathcal{F}',\mathcal{A}') \in [(\mathcal{F},\mathcal{A})]$ the map $\mathcal{I}_{\mathcal{A}'}^{-1}$ takes the abstract cone $\Sigma_{[(\mathcal{F},\mathcal{A})]}$ to the cone $\Sigma_{\mathcal{F}',\mathcal{A}'}$ of SOS polynomials.

For different, but equivalent, multi-index sets $\mathcal{A} \sim_P \mathcal{A}'$ the standard SOS relaxations $\Sigma_\mathcal{A}, \Sigma_{\mathcal{A}'}$ defined by (3) will in general not be equivalent. It is therefore meaningless to speak of a standard SOS relaxation of the cone $\mathcal{P}_{[\mathcal{A}']}$. For every representative $\mathcal{A} \in [\mathcal{A}']$ we have, however, a finite hierarchy of SOS relaxations $\Sigma_{\mathcal{F},\mathcal{A}}$ defined by (4). This allows us to define SOS relaxations of the abstract cone $\mathcal{P}_{[\mathcal{A}']}$.

Definition 2. *Let C be an equivalence class of finite ordered multi-index sets with respect to the equivalence relation $\sim_P$, and $\mathcal{P}_C$ the corresponding abstract cone of positive polynomials. For every pair $(\mathcal{F},\mathcal{A})$ of finite ordered multi-index sets such that $\mathcal{A} \in C$ and $\mathcal{F} \subset \mathcal{F}_{\max}(\mathcal{A})$, we call the abstract cone $\Sigma_{[(\mathcal{F},\mathcal{A})]}$ an SOS relaxation of $\mathcal{P}_C$.*

Clearly the SOS relaxations of the cone $\mathcal{P}_C$ are inner semidefinite relaxations. The set of SOS relaxations inherits the partial order defined in Definition 1.

Definition 3. *Let C be an equivalence class of finite ordered multi-index sets with respect to the equivalence relation $\sim_P$, and let $\Sigma_{C_1}, \Sigma_{C_2}$ be SOS relaxations of the cone $\mathcal{P}_C$, where C_1, C_2 are equivalence classes of the relation $\sim_\Sigma$. If there exist multi-index sets $\mathcal{F}_1, \mathcal{F}_2, \mathcal{A}$ such that $(\mathcal{F}_1, \mathcal{A}) \in C_1$, $(\mathcal{F}_2, \mathcal{A}) \in C_2$, and $\mathcal{F}_1 \subset \mathcal{F}_2$, then we say that the relaxation Σ_{C_1} is* coarser *than Σ_{C_2}, or Σ_{C_2} is* finer *than Σ_{C_1}.*

It is not hard to see that the relation defined in Definition 3 is indeed a partial order. A finer relaxation is tighter, but a strictly finer relaxation does not a priori need to be strictly tighter. Note that we do not require $\mathcal{A} \in C$. This implies that if, e.g., both $\Sigma_{C_1}, \Sigma_{C_2}$ are SOS relaxations for two different abstract cones $\mathcal{P}_C, \mathcal{P}_{C'}$, and Σ_{C_2} is a finer relaxation of $\mathcal{P}_C$ than Σ_{C_1}, then Σ_{C_2} is also a finer relaxation of $\mathcal{P}_{C'}$ than Σ_{C_1}.

Theorem 3. *Let $\mathcal{F}, \mathcal{A} \subset \mathbf{N}^n$ be finite ordered multi-index sets satisfying $\mathcal{F} \subset \mathcal{F}_{\max}(\mathcal{A})$, and suppose that $\mathcal{A}$ satisfies (1). Let further M be an $n \times n$ integer matrix with odd determinant, and let $v \in \mathbf{Z}^n$ be an arbitrary integer row vector. Let now $\mathcal{F}'$ be the multi-index set obtained from $\mathcal{F}$ by application of the affine map $R' : \beta \mapsto \beta M + v$, and $\mathcal{A}'$ the set obtained from $\mathcal{A}$ by application of the affine map $R : \alpha \mapsto \alpha M + 2v$. Then $\mathcal{A} \sim_P \mathcal{A}'$, provided the elements of $\mathcal{A}'$ have nonnegative entries, and $(\mathcal{F}, \mathcal{A}) \sim_\Sigma (\mathcal{F}', \mathcal{A}')$, provided the elements of $\mathcal{F}', \mathcal{A}'$ have nonnegative entries.*

Proof. Assume the conditions of the theorem. Then we have $N_{\mathcal{A}'} = R[N_\mathcal{A}]$, and therefore $\frac{1}{2}N_{\mathcal{A}'} = R'[\frac{1}{2}N_\mathcal{A}]$. It follows that $\mathcal{F}' \subset \mathcal{F}_{\max}(\mathcal{A}')$.

Since det $M \neq 0$, the map R is invertible. Further, the matrix $\pi_2[M]$ defines an invertible linear map Z on $\mathbf{F}_2^n$, because $\det \pi_2[M] = \pi_2(\det M) = 1$. Moreover, the projection π_2^n intertwines the maps R and Z, because the translational part of R is even. It is then easily seen that the restrictions $R|_{\operatorname{aff} \mathcal{A}} : \operatorname{aff} \mathcal{A} \to \operatorname{aff} \mathcal{A}'$ and $Z|_{\operatorname{span}(\pi_2^n[\mathcal{A}])} : \operatorname{span}(\pi_2^n[\mathcal{A}]) \to \operatorname{span}(\pi_2^n[\mathcal{A}'])$ satisfy condition 3) of Theorem 1. This proves the relation $\mathcal{A} \sim_P \mathcal{A}'$.

Likewise, the restriction $I = R|_{(\mathcal{F}+\mathcal{F})\cup\mathcal{A}}$ makes the diagram in Theorem 2 commute, which proves the relation $(\mathcal{F}, \mathcal{A}) \sim_\Sigma (\mathcal{F}', \mathcal{A}')$.

We will use this result to construct strictly finer relaxations from a given standard SOS relaxation.

If the determinant of the matrix M in Theorem 3 equals ±1, then the maps R', R define isomorphisms of $\mathbf{Z}^n$. Then $\#\mathcal{F}_{\max}(\mathcal{A}) = \#\mathcal{F}_{\max}(\mathcal{A}')$, and the standard relaxations $\Sigma_\mathcal{A}, \Sigma_{\mathcal{A}'}$ are equivalent. If, however, $|\det M| > 1$, then $\#\mathcal{F}_{\max}(\mathcal{A}')$ might be strictly bigger than $\#\mathcal{F}_{\max}(\mathcal{A})$, and then $\mathcal{I}_{\mathcal{A}'}[\Sigma_{\mathcal{A}'}]$ will be strictly finer than $\mathcal{I}_\mathcal{A}[\Sigma_\mathcal{A}]$. In particular, this happens if $\#\mathcal{A} > 1$ and the sets $\mathcal{F}', \mathcal{A}'$ are obtained from $\mathcal{F}, \mathcal{A}$ by multiplying every multi-index with a fixed odd natural number $k > 1$. Thus, unlike the hierarchy of relaxations (4), for abstract cones $\mathcal{P}_C$ of dimension $m > 1$ the hierarchy of SOS relaxations is infinite, and the corresponding partial order does not have a finest element.

6 Example

Consider the inhomogeneous Motzkin polynomial $p_M(x,y) = x^4y^2 + x^2y^4 + 1 - 3x^2y^2 \in \mathcal{P}_{\mathcal{A}}$ with $\mathcal{A} = \{(4,2),(2,4),(0,0),(2,2)\}$. Its Newton polytope is the triangle given by $N(p_M) = N_{\mathcal{A}} = \operatorname{conv}\{(4,2),(2,4),(0,0)\}$, and therefore $\mathcal{F}_{\max}(\mathcal{A}) = \{(2,1),(1,2),(0,0),(1,1)\}$. It is easily checked that the standard SOS cone $\Sigma_{\mathcal{A}}$ obtained from (4) by setting $\mathcal{F} = \mathcal{F}_{\max}(\mathcal{A})$ consists of those polynomials in $\mathcal{L}_{\mathcal{A}}$ all whose coefficients are nonnegative. The corresponding abstract SOS cone is therefore given by $\Sigma_{[(\mathcal{F},\mathcal{A})]} = \mathbf{R}^4_+$.

Using Lemma 4, it is a little exercise to show $\operatorname{con}\operatorname{cl}\mathcal{X}_{\mathcal{A}} = \{(y_1,y_2,y_3,y_4)^T \in \mathbf{R}^4_+ \,|\, y_4 = \sqrt[3]{y_1y_2y_3}\}$. By Lemma 3 we then easily obtain $\mathcal{P}_{[\mathcal{A}]} = \{c = (c_1,c_2,c_3,c_4)^T \,|\, c_1,c_2,c_3 \geq 0,\ c_4 \geq -3\sqrt[3]{c_1c_2c_3}\}$. Let us now apply the construction provided in Theorem 3, setting $M = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$ and $v = 0$. Then $\mathcal{F}$ is mapped to $\mathcal{F}' = \{(3,0),(0,3),(0,0),(1,1)\}$, and $\mathcal{A}$ to $\mathcal{A}' = \{(6,0),(0,6),(0,0),(2,2)\}$. By Theorem 3 we then have $\mathcal{A} \sim_P \mathcal{A}'$ and $(\mathcal{F},\mathcal{A}) \sim_\Sigma (\mathcal{F}',\mathcal{A}')$. The set $\mathcal{F}'' = \mathcal{F}_{\max}(\mathcal{A}')$, however, is now composed of 10 points and is hence strictly larger than $\mathcal{F}'$. Therefore the relaxation $\Sigma_{[(\mathcal{F}'',\mathcal{A}')]}$ of the cone $\mathcal{P}_{[\mathcal{A}]}$ is strictly finer than $\Sigma_{[(\mathcal{F},\mathcal{A})]}$. Moreover, with $e_3 = (1,1,1)^T$, $v(x,y) = (\sqrt[3]{c_1}x^2, \sqrt[3]{c_2}y^2, \sqrt[3]{c_3})^T$ every polynomial $p_c(x,y) = c_1x^6 + c_2y^6 + c_3 + c_4x^2y^2 \in \mathcal{P}_{\mathcal{A}'}$, i.e. satisfying $c_1,c_2,c_3 \geq 0$ and $c_4 \geq -3\sqrt[3]{c_1c_2c_3}$, can be written as

$$p_c(x,y) = e^T v(x,y) \cdot v(x,y)^T \frac{3I_3 - ee^T}{2} v(x,y) + (c_4 + 3\sqrt[3]{c_1c_2c_3})x^2y^2,$$

which obviously is a sum of squares. Thus the relaxation $\Sigma_{[(\mathcal{F}'',\mathcal{A}')]}$ is *exact.*

From the proof of [3, Theorem 1] it follows that there does not exist a fixed integer $d > 0$ such that with $h(x) = \left(\sum_{k=1}^n x_k^2\right)^d$ the product hp is a sum of squares for every polynomial $p \in \mathcal{P}_{\mathcal{A}}$. Thus this commonly used hierarchy of SOS relaxations is not capable of representing the cone $\mathcal{P}_{[\mathcal{A}]}$ at any finite step.

References

1. Motzkin T S (1967) The arithmetic-geometric inequality. In: Shisha O (ed) Inequalities Proc of Sympos at Wright-Patterson AFB. Acad Press, New York
2. Lasserre J B (2009) Moments, positive polynomials and their applications. Imperial College Press, London
3. Reznick B (2005) Proc Amer Math Soc 133:2829–2834
4. Reznick B (1978) Duke Math Journal 45(2):363–372
5. Ellison W J (1969) Math Proc Cambridge Phil Soc 65:663–672
6. Hilbert D (1888) Math Ann 32:342–350.

Asynchronous Gossip Algorithm for Stochastic Optimization: Constant Stepsize Analysis*

S. Sundhar Ram[1], Angelia Nedić[2], and Venu V. Veeravalli[3]

[1] Electrical and Computer Engineering Department, University of Illinois at Urbana-Champaign, Urbana IL 61801, USA ssriniv5@illinois.edu
[2] Industrial and Enterprise Systems Engineering Department, University of Illinois at Urbana-Champaign, Urbana IL 61801, USA angelia@illinois.edu
[3] Electrical and Computer Engineering Department, University of Illinois at Urbana-Champaign, Urbana IL 61801, USA vvv@illinois.edu

Summary. We consider the problem of minimizing the sum of convex functions over a network when each component function is known (with stochastic errors) to a specific network agent. We discuss a gossip based algorithm of [2], and we analyze its error bounds for a constant stepsize that is uncoordinated across the agents.

1 Introduction

The gossip optimization algorithm proposed in [2] minimizes a sum of functions when each component function is known (with stochastic errors) to a specific network agent. The algorithm is reliant on the gossip-consensus scheme of [1], which serves as a main mechanism for the decentralization of the overall network optimization problem. The gossip-based optimization algorithm is *distributed* and *totally asynchronous* since there is no central coordinator and the agents do not have a common notion of time. Furthermore, the algorithm is *completely local* since each agent knows only its neighbors, and relies on its own local information and some limited information received from its neighbors. Agents have no information about the global network.

In [2], the convergence properties of the algorithm with a (random) diminishing uncoordinated stepsize was studied. In this paper we study the properties of the algorithm when the agents use deterministic *uncoordinated constant stepsizes*. Our primary interest is in establishing the limiting error bounds for the method. We provide such error bounds for strongly convex functions and for general convex functions (through the use of the running averages of the iterates). The bounds are given explicitly in terms of the problem data, the network connectivity parameters and the agent stepsize values. The bounds scale linearly in the number of agents.

* This work is supported by NSF Career grant 0742538.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_5, © Springer-Verlag Berlin Heidelberg 2010

2 Problem, algorithm and assumptions

Throughout this paper, we use $\|x\|$ to denote the Euclidean norm of a vector x. We write $\mathbf{1}$ to denote the vector with all entries equal to 1. The matrix norm $\|M\|$ of a matrix M is the norm induced by the Euclidean vector norm. We use x^T and M^T to denote the transpose of a vector x and a matrix M, respectively. We write $[x]_i$ to denote the i-th component of a vector x. Similarly, we write $[M]_{i,j}$ or $M_{i,j}$ to indicate the (i,j)-th component of a matrix M. We write $|S|$ to denote the cardinality of a set S with finitely many elements.

Consider a network of m agents that are indexed by $1,\ldots,m$, and let $V = \{1,\ldots,m\}$. The agents communicate over a network with a static topology represented by an undirected graph $(V,\mathscr{E})$, where $\mathscr{E}$ is the set of undirected links $\{i,j\}$ with $i \neq j$ and $\{i,j\} \in \mathscr{E}$ only if agents i and j can communicate.

We are interested in solving the following problem over the network:

$$\begin{aligned} &\text{minimize} && f(x) \triangleq \sum_{i=1}^{m} f_i(x) \\ &\text{subject to} && x \in X, \end{aligned} \tag{1}$$

where each f_i is a function defined over the set $X \subseteq \mathbb{R}^n$. The problem (1) is to be solved under the following restrictions on the network information. Each agent i knows only its own objective function f_i and it can compute the (sub)gradients ∇f_i with stochastic errors. Furthermore, each agent can communicate and exchange some information with its local neighbors only.

To solve problem (1), we consider an algorithm that is based on the gossip consensus model in [1]. Let $N(i)$ be the set of all neighbors of agent i, i.e., $N(i) = \{j \in V \mid \{i,j\} \in \mathscr{E}\}$. Each agent has its local clock that ticks at a Poisson rate of 1 independently of the clocks of the other agents. At each tick of its clock, agent i communicates with a randomly selected neighbor $j \in N(i)$ with probability $P_{ij} > 0$, where $P_{ij} = 0$ for $j \notin N(i)$. Then, agent i and the selected neighbor j exchange their current estimates of the optimal solution, and each of these agents performs an update using the received estimate and the erroneous (sub)gradient direction of its objective function.

Consider a single virtual clock that ticks whenever any of the local Poisson clocks tick. Let Z_k be the time of the k-th tick of the virtual Poisson clock, and let the time be discretized according to the intervals $[Z_{k-1}, Z_k)$, $k \geq 1$. Let I_k denote the index of the agent that wakes up at time k, and let J_k denote the index of a neighbor that is selected for communication. Let $x_{i,k}$ denote the iterate of agent i at time k. The iterates are generated according to the following rule. Agents other than I_k and J_k do not update:

$$x_{i,k} = x_{i,k-1} \qquad \text{for } i \notin \{I_k, J_k\}. \tag{2}$$

Agents I_k and J_k average their current iterate and update independently using subgradient steps as follows:

$$\begin{aligned} v_{i,k} &= \left(x_{I_k,k-1} + x_{J_k,k-1}\right)/2, \\ x_{i,k} &= P_X[v_{i,k} - \alpha_i(\nabla f_i(v_{i,k}) + \epsilon_{i,k})], \end{aligned} \tag{3}$$

where P_X denotes the Euclidean projection on the set X, $\nabla f_i(x)$ is a subgradient of f_i at x, α_i is a positive stepsize, and $\epsilon_{i,k}$ is *stochastic error* in computing $\nabla f_i(v_{i,k})$. The updates are initialized with random vectors $x_{i,0}$, $i \in V$, which are assumed to be mutually independent and also independent of all the other random variables in the process.

The key difference between the work in [2] and this paper is in the stepsize. The work in [2] considers a diminishing (random) stepsize $\alpha_{i,k}$, which is defined in terms of the frequency of agent i updates. In contrast, in this paper, we consider the method with a deterministic constant stepsize $\alpha_i > 0$ for all i. As the stepsizes across agents need not to be the same, the algorithm does not require any coordination among the agents.

We next discuss our assumptions.

Assumption 1 *The underlying communication graph* $(V, \mathscr{E})$ *is connected.*

Assumption 1 ensures that, through the gossip strategy, the information of each agent reaches every other agent frequently enough. However, to ensure that the common vector solves problem (1), some additional assumptions are needed for the set X and the functions f_i. We use the following.

Assumption 2 *The set* $X \subseteq \mathbb{R}^n$ *is compact and convex. Each function* f_i *is defined and convex over an open set containing the set* X*.*

Differentiability of the functions f_i is not assumed. At points where the gradient does not exist, we use a subgradient. Under the compactness of X, the subgradients are uniformly bounded over X, i.e., for some $C > 0$ we have

$$\sup_{x \in X} \|\nabla f_i(x)\| \leq C \qquad \text{for all } i \in V.$$

Furthermore, the following *approximate subgradient relation* holds:

$$\nabla f_i(v)^T(v-x) \geq f_i(y) - f_i(x) - C\|v-y\| \quad \text{for any } x, y, v \in X \text{ and } i \in V. \tag{4}$$

We now discuss the random errors $\epsilon_{i,k}$ in computing the subgradients $\nabla f_i(x)^T$ at points $x = v_{i,k}$. Let $\mathscr{F}_k$ be the σ-algebra generated by the entire history of the algorithm up to time k inclusively, i.e.,

$$\mathscr{F}_k = \{x_{i,0},\, i \in V\} \cup \{I_\ell, J_\ell, \epsilon_{I_\ell,\ell}, \epsilon_{J_\ell,\ell};\ 1 \leq \ell \leq k\} \qquad \text{for all } k \geq 1,$$

where $\mathscr{F}_0 = \{x_{i,0},\, i \in V\}$. We use the following assumption on the errors.

Assumption 3 *With probability 1, for all* $i \in \{I_k, J_k\}$ *and* $k \geq 1$*, the errors satisfy* $\mathsf{E}\left[\epsilon_{i,k} \mid \mathscr{F}_{k-1}, I_k, J_k\right] = 0$ *and* $\mathsf{E}\left[\|\epsilon_{i,k}\|^2 \mid \mathscr{F}_{k-1}, I_k, J_k\right] \leq \nu^2$ *for some* ν*.*

When X and each f_i are convex, every vector $v_{i,k}$ is a convex combination of $x_{j,k} \in X$ (see Eq. (3)), implying that $v_{i,k} \in X$. In view of subgradient boundedness and Assumption 3, it follows that for $k \geq 1$,

$$\mathsf{E}\left[\|\nabla f_i^T(v_{i,k}) + \epsilon_{i,k}\|^2 \mid \mathscr{F}_{k-1}, I_k, J_k\right] \leq (C+\nu)^2 \quad \text{for } i \in \{I_k, J_k\}. \tag{5}$$

3 Preliminaries

We provide an alternative description of the algorithm, and study the properties of the agent's disagreements. Define the matrix W_k as follows:

$$W_k = I - \frac{1}{2}(e_{I_k} - e_{J_k})(e_{I_k} - e_{J_k})^T \qquad \text{for all } k, \tag{6}$$

where $e_i \in \mathbb{R}^m$ has its i-th entry equal to 1, and the other entries equal to 0. Using W_k, we can write method (2)–(3) as follows: for all $k \geq 1$ and $i \in V$,

$$\begin{aligned} x_{i,k} &= v_{i,k} + p_{i,k}\chi_{\{i\in\{I_k,J_k\}\}}, \\ v_{i,k} &= \sum_{j=1}^{m}[W_k]_{ij}\; x_{j,k-1}, \\ p_{i,k} &= P_X[v_{i,k} - \alpha_i\,(\nabla f_i(v_{i,k}) + \epsilon_{i,k})] - v_{i,k}, \end{aligned} \tag{7}$$

where $\chi_{\mathscr{C}}$ is the characteristic function of an event $\mathscr{C}$. The matrices W_k are symmetric and stochastic, implying that each $\mathsf{E}\,[W_k]$ is doubly stochastic. Thus, by the definition of the method in (7), we can see that

$$\sum_{i=1}^{m} \mathsf{E}\left[\|v_{i,k} - x\|^2 \mid \mathcal{F}_{k-1}\right] \leq \sum_{j=1}^{m} \|x_{j,k-1} - x\|^2 \qquad \text{for all } x \in \mathbb{R}^n \text{ and } k, \tag{8}$$

$$\sum_{i=1}^{m} \mathsf{E}\left[\|v_{i,k} - x\| \mid \mathcal{F}_{k-1}\right] \leq \sum_{j=1}^{m} \|x_{j,k-1} - x\| \qquad \text{for all } x \in \mathbb{R}^n \text{ and } k. \tag{9}$$

In our analysis, we use the fact that $W_k^2 = W_k$, $(W_k - \frac{1}{m}\mathbf{1}\mathbf{1}^T)^2 = W_k - \frac{1}{m}\mathbf{1}\mathbf{1}^T$ and that the norm of the matrices $\mathsf{E}[W_k] - \frac{1}{m}\mathbf{1}\mathbf{1}^T$ is equal to the second largest eigenvalue of $\mathsf{E}[W_k]$. We let λ denote the square of this eigenvalue, i.e., $\lambda = \|\mathsf{E}[W_k] - \frac{1}{m}\mathbf{1}\mathbf{1}^T\|^2$. We have the following lemma.

Lemma 1. *Let Assumption 1 hold. Then, we have $\lambda < 1$.*

We next provide an estimate for the disagreement among the agents.

Lemma 2. *Let Assumptions 1–3 hold*[4]*, and let $\{x_{i,k}\}$, $i = 1, \ldots, m$, be the iterate sequences generated by algorithm (7). Then, we have for all i,*

$$\limsup_{k\to\infty} \sum_{i=1}^{m} \mathsf{E}\left[\|x_{i,k} - \bar{y}_k\|\right] \leq \frac{\sqrt{2m}\,\bar{\alpha}}{1 - \sqrt{\lambda}}\,(C + \nu),$$

where $\bar{y}_k = \frac{1}{m}\sum_{j=1}^{m} x_{j,k}$ for all k, and $\bar{\alpha} = \max_{1\leq j\leq m} \alpha_j$.

[4] Here, we only need the error boundedness from Assumption 3.

Proof. We will consider coordinate-wise relations by defining the vector $z_k^\ell \in \mathbb{R}^m$, for each $\ell \in \{1, \dots, n\}$, as the vector with entries $[x_{i,k}]_\ell$, $i = 1, \dots, m$. From the definition of the method in (7), we have

$$z_k^\ell = W_k\, z_{k-1}^\ell + \zeta_k^\ell \qquad \text{for } k \geq 1, \tag{10}$$

where $\zeta_k^\ell \in \mathbb{R}^m$ is a vector with coordinates $[\zeta_k^\ell]_i$ given by

$$[\zeta_k^\ell]_i = \begin{cases} [P_X[v_{i,k} - \alpha_i\,(\nabla f_i(v_{i,k}) + \epsilon_{i,k})] - v_{i,k}]_\ell & \text{if } i \in \{I_k, J_k\}, \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

Furthermore, note that $[\bar{y}_k]_\ell$ is the average of the entries of the vector z_k^ℓ, i.e.,

$$[\bar{y}_k]_\ell = \frac{1}{m}\, \mathbf{1}^T z_k^\ell \qquad \text{for all } k \geq 0. \tag{12}$$

By Eqs. (10) and (12), we have $[\bar{y}_k]_\ell = \frac{1}{m}\left(\mathbf{1}^T W_k z_{k-1}^\ell + \mathbf{1}^T \zeta_k^\ell\right)$, implying

$$\begin{aligned} z_k^\ell - \mathbf{1}[\bar{y}_k]_\ell &= W_k z_{k-1}^\ell + \zeta_k^\ell - \frac{1}{m}\,\mathbf{1}\mathbf{1}^T(W_k z_{k-1}^\ell + \zeta_k^\ell) \\ &= \left(W_k - \frac{1}{m}\,\mathbf{1}\mathbf{1}^T\right) z_{k-1}^\ell + \left(I - \frac{1}{m}\,\mathbf{1}\mathbf{1}^T\right) \zeta_k^\ell, \end{aligned}$$

where I denotes the identity matrix, and the last equality follow by the doubly stochasticity of W_k, i.e., $\mathbf{1}^T W_k = \mathbf{1}^T$. Since the matrices W_k are stochastic, i.e., $W_k \mathbf{1} = \mathbf{1}$, it follows $\left(W_k - \frac{1}{m}\,\mathbf{1}\mathbf{1}^T\right)\mathbf{1} = 0$, implying that $\left(W_k - \frac{1}{m}\,\mathbf{1}\mathbf{1}^T\right)[\bar{y}_{k-1}]_\ell \mathbf{1} = 0$. Hence,

$$z_k^\ell - [\bar{y}_k]_\ell \mathbf{1} = D_k(z_{k-1}^\ell - [\bar{y}_{k-1}]_\ell \mathbf{1}) + M\zeta_k^\ell \qquad \text{for all } k \geq 1,$$

where $D_k = W_k - \frac{1}{m}\,\mathbf{1}\mathbf{1}^T$ and $M = I - \frac{1}{m}\,\mathbf{1}\mathbf{1}^T$. Thus, we have for $\ell = 1, \dots, n$ and all $k \geq 1$,

$$\|z_k^\ell - [\bar{y}_k]_\ell \mathbf{1}\|^2 \leq \|D_k(z_{k-1}^\ell - [\bar{y}_{k-1}]_\ell \mathbf{1})\|^2 + \|M\zeta_k^\ell\|^2 + 2\|D_k(z_{k-1}^\ell - [\bar{y}_{k-1}]_\ell \mathbf{1})\|\,\|M\zeta_k^\ell\|.$$

By summing these relations over $\ell = 1, \dots, n$, and then taking the expectation and using Hölder's inequality we obtain for all $k \geq 1$,

$$\sum_{\ell=1}^n \mathsf{E}\left[\|z_k^\ell - [\bar{y}_k]_\ell \mathbf{1}\|^2\right] \leq \left(\sqrt{\sum_{\ell=1}^n \mathsf{E}\left[\|D_k(z_{k-1}^\ell - [\bar{y}_{k-1}]_\ell \mathbf{1})\|^2\right]} + \sqrt{\sum_{\ell=1}^n \mathsf{E}\left[\|M\zeta_k^\ell\|^2\right]}\right)^2 \tag{13}$$

Using the fact the matrix W_k is independent of the past $\mathcal{F}_{k-1}$, we have

$$\sum_{\ell=1}^n \mathsf{E}\left[\left\|D_k(z_{k-1}^\ell - [\bar{y}_{k-1}]_\ell \mathbf{1})\right\|^2 \mid \mathcal{F}_{k-1}\right] \leq \lambda \sum_{\ell=1}^n \|z_{k-1}^\ell - [\bar{y}_{k-1}]_\ell \mathbf{1}\|^2, \tag{14}$$

where $\lambda = \|\mathsf{E}[D_k^T D_k]\|^2 = \|\mathsf{E}[D_k]\|^2$, and $\lambda < 1$ from Lemma 1.

We next estimate the second term in (13). The matrix $M = I - \frac{1}{m}\mathbf{1}\mathbf{1}^T$ is a projection matrix (it projects on the subspace orthogonal to the vector $\mathbf{1}$), so that we have $\|M\|^2 = 1$, implying that $\|M\zeta_k^\ell\|^2 \le \|\zeta_k^\ell\|^2$ for all k. Using this and the definition of ζ_k^ℓ in (11), we obtain

$$\|M\zeta_k^\ell\|^2 \le 2 \sum_{i\in\{I_k,J_k\}} \left|[P_X[v_{i,k} - \alpha_i\left(\nabla f_i(v_{i,k}) + \epsilon_{i,k}\right)] - v_{i,k}]_\ell\right|^2.$$

Therefore,

$$\sum_{\ell=1}^n \mathsf{E}\left[\|M\zeta_k^\ell\|^2\right] \le 2\mathsf{E}\left[\mathsf{E}\left[\sum_{i\in\{I_k,J_k\}} \alpha_i^2\|\nabla f_i(v_{i,k}) + \epsilon_{i,k}\|^2 \mid \mathcal{F}_{k-1}, I_k, J_k\right]\right]$$
$$\le 2\bar{\alpha}^2(C+\nu)^2,$$

where in the last inequality we use $\bar{\alpha} = \max_i \alpha_i$ and relation (5). Combining the preceding relation with Eqs. (13) and (14), we obtain

$$\sqrt{\sum_{\ell=1}^n \mathsf{E}\left[\|z_k^\ell - [\bar{y}_k]_\ell\mathbf{1}\|^2\right]} \le \sqrt{\lambda}\sqrt{\sum_{\ell=1}^n \mathsf{E}\left[\|z_{k-1}^\ell - [\bar{y}_{k-1}]_\ell\mathbf{1}\|^2\right]} + \sqrt{2}\,\bar{\alpha}(C+\nu).$$

Since $\lambda < 1$, by recursively using the preceding relation, we have

$$\limsup_{k\to\infty}\sqrt{\sum_{\ell=1}^n \mathsf{E}\left[\|z_k^\ell - [\bar{y}_k]_\ell\mathbf{1}\|^2\right]} \le \frac{\sqrt{2}\,\bar{\alpha}}{1-\sqrt{\lambda}}(C+\nu).$$

The result now follows by $\sum_{i=1}^m \mathsf{E}\left[\|x_{i,k} - \bar{y}_k\|^2\right] = \sum_{\ell=1}^n \mathsf{E}\left[\|z_k^\ell - \mathbf{1}[\bar{y}_k]_\ell\|^2\right]$ and $\sum_{i=1}^m \mathsf{E}\left[\|x_{i,k} - \bar{y}_k\|\right] \le \sqrt{m}\,\sqrt{\sum_{i=1}^m \mathsf{E}\left[\|x_{i,k} - \bar{y}_k\|^2\right]}$. □

The bound in Lemma 2 captures the dependence of the differences between $x_{i,k}$ and their current average $\bar{y}_k$ in terms of the maximum stepsize and the communication graph. The impact of the communication graph $(V, \mathscr{E})$ is captured by the spectral radius λ of the expected matrices $\mathsf{E}\left[(W_k - \frac{1}{m}\mathbf{1}\mathbf{1}^T)^2\right]$.

4 Error Bounds

We have the following result for strongly convex functions.

Proposition 1. *Let Assumptions 1–3 hold. Let each function f_i be strongly convex over the set X with a constant σ_i, and let α_i be such that $2\alpha_i\sigma_i < 1$. Then, for the sequences $\{x_{i,k}\}$, $i \in V$, generated by (7), we have for all i,*

$$\limsup_{k\to\infty}\sum_{i=1}^m \mathsf{E}[\|x_k^i - x^*\|^2] \le \frac{\bar{\omega} - \underline{\omega}}{1-q}\,2mCC_X + \frac{\bar{\alpha}\bar{\omega}}{1-q}\left(m + \frac{2\sqrt{2m}}{1-\sqrt{\lambda}}\right)(C+\nu)^2,$$

where x^* is the optimal solution of problem (1), $q = \max_i\{1 - 2\gamma_i\alpha_i\sigma_i\}$, $\gamma_i = \frac{1}{m}\left(1 + \sum_{j\in N(i)} P_{ji}\right)$, $C_X = \max_{x,y\in X}\|x-y\|$, $\bar{\alpha} = \max_i \alpha_i$, $\bar{\omega} = \max_i \gamma_i\alpha_i$, and $\underline{\omega} = \min_i \gamma_i\alpha_i$.

Proof. The sum $f = \sum_{i=1}^m f_i$ is strongly convex with constant $\sigma = \sum_{i=1}^m \sigma_i$. Thus, problem (1) has a unique optimal solution $x^* \in X$. From relation (7), the nonexpansive property of the projection operation, and relation (5) we obtain for the optimal point x^*, and any k and $i \in \{I_k, J_k\}$,

$$\begin{aligned}\mathsf{E}\left[\|x_{i,k} - x^*\|^2 \mid \mathcal{F}_{k-1}, I_k, J_k\right] \le \|v_{i,k} - x^*\|^2 - 2\alpha_i \nabla f_i(v_{i,k})^T (v_{i,k} - x^*) \\ +\alpha_i^2 (C+\nu)^2. \qquad (15)\end{aligned}$$

By the strong convexity of f_i, it follows

$$\nabla f_i(v_{i,k})^T (v_{i,k} - x^*) \ge \sigma_i \|v_{i,k} - x^*\|^2 + \nabla f_i(x^*)^T (v_{i,k} - x^*).$$

Using $\bar{y}_{k-1} = \frac{1}{m}\sum_{j=1}^m x_{j,k-1}$, we have $\nabla f_i(x^*)^T (v_{i,k} - x^*) = \nabla f_i(x^*)^T (\bar{y}_{k-1} - x^*) + \nabla f_i(x^*)^T (v_{i,k} - \bar{y}_{k-1})$, which in view of $\|\nabla f_i(x^*)\| \le C$ implies

$$\nabla f_i(x^*)^T (v_{i,k} - x^*) \ge \nabla f_i(x^*)^T (\bar{y}_{k-1} - x^*) - C\,\|v_{i,k} - \bar{y}_{k-1}\|. \qquad (16)$$

By combining the preceding two relations with inequality (15), we obtain

$$\begin{aligned}\mathsf{E}\left[\|x_{i,k} - x^*\|^2 \mid \mathcal{F}_{k-1}, I_k, J_k\right] \le (1 - 2\alpha_i\sigma_i)\|v_{i,k} - x^*\|^2 \\ +\alpha_i^2 (C+\nu)^2 - 2\alpha_i \nabla f_i(x^*)^T (\bar{y}_{k-1} - x^*) + 2\alpha_i C \|v_{i,k} - \bar{y}_{k-1}\|.\end{aligned}$$

Taking the expectation with respect to F_{k-1} and using the fact the preceding inequality holds with probability γ_i (the probability that agent i updates at time k), and $x_{i,k} = v_{i,k}$ with probability $1 - \gamma_i$, we obtain for any i and k,

$$\begin{aligned}&\mathsf{E}\left[\|x_{i,k} - x^*\|^2 \mid \mathcal{F}_{k-1}\right] \le (1 - 2\gamma_i\alpha_i\sigma_i)\mathsf{E}\left[\|v_{i,k} - x^*\|^2 \mid \mathcal{F}_{k-1}\right] \\ &+ \gamma_i\alpha_i^2 (C+\nu)^2 - 2\gamma_i\alpha_i \nabla f_i(x^*)^T (\bar{y}_{k-1} - x^*) + 2\gamma_i\alpha_i C \mathsf{E}\left[\|v_{i,k} - \bar{y}_{k-1}\| \mid \mathcal{F}_{k-1}\right].\end{aligned}$$

Adding and subtracting $(\min_i \gamma_i\alpha_i)\,\nabla f_i(x^*)^T (\bar{y}_{k-1} - x^*)$, and using $\bar{y}_{k-1} \in X$ and the compactness of X, we obtain

$$\begin{aligned}&\mathsf{E}\left[\|x_{i,k} - x^*\|^2 \mid \mathcal{F}_{k-1}\right] \le q\mathsf{E}\left[\|v_{i,k} - x^*\|^2 \mid \mathcal{F}_{k-1}\right] + \bar{\omega}\bar{\alpha}(C+\nu)^2 \\ &+ 2(\bar{\omega} - \underline{\omega})CC_X - 2\underline{\omega}\nabla f_i(x^*)^T (\bar{y}_{k-1} - x^*) + 2\bar{\omega}C\mathsf{E}\left[\|v_{i,k} - \bar{y}_{k-1}\| \mid \mathcal{F}_{k-1}\right],\end{aligned}$$

where $q = \max_i\{1 - 2\gamma_i\alpha_i\sigma_i\}$, $\underline{\omega} = \min_i \gamma_i\alpha_i$, $\bar{\omega} = \max_i \gamma_i\alpha_i$, $\bar{\alpha} = \max_i \alpha_i$ and $C_X = \max_{x,y\in X}\|x - y\|$. Now, by summing the preceding relations over i, by using $\sum_{i=1}^m \nabla f_i(x^*)^T (\bar{y}_{k-1} - x^*) \ge 0$ and using relation (8) (with $x = x^*$) and relation (9) (with $x = \bar{y}_{k-1}$), we obtain

$$\begin{aligned}\sum_{i=1}^m \mathsf{E}[\|x_{i,k} - x^*\|^2] \le q\sum_{j=1}^m \mathsf{E}[\|x_{j,k} - x^*\|^2] + m\bar{\omega}\bar{\alpha}(C+\nu)^2 \\ + 2m(\bar{\omega} - \underline{\omega})CC_X + 2\bar{\omega}C\sum_{j=1}^m \mathsf{E}[\|x_{j,k} - \bar{y}_{k-1}\|].\end{aligned}$$

The desired estimate follows from the preceding relation by noting that $q < 1$, by taking the limit superior and by using Lemma 2 and $C(C+\nu) \le (C+\nu)^2$. □

Proposition 1 requires each node to select a stepsize α_i so that $2\alpha_i\sigma_i < 1$, which can be done since each node knows its strong convexity constant σ_i. Furthermore, note that the relation $q = \max_{1\le i\le m}\{1-\gamma_i\alpha_i\sigma_i\} < 1$ can be ensured globally over the network without any coordination among the agents.

The following error estimate holds without strong convexity.

Proposition 2. *Let Assumptions 1–3 hold. Then, for the sequences* $\{x_{i,k}\}$, $i \in V$, *generated by (7), we have for all* i,

$$\limsup_{k\to\infty} \frac{1}{k}\sum_{t=1}^{k} \mathsf{E}[f(x_{i,t-1})] \le f^* + m\,(\rho-1)\,CC_X \\ +\bar{\alpha}\left((\rho+m)\frac{\sqrt{2m}}{1-\sqrt{\lambda}} + \frac{m}{2}\rho\right)(C+\nu)^2,$$

where f^* *is the optimal value of problem* (1), $C_X = \max_{x,y\in X}\|x-y\|$, $\rho = \frac{\max_i \gamma_i\alpha_i}{\min_i \gamma_i\alpha_i}$, $\gamma_i = \frac{1}{m}\left(1+\sum_{j\in N(i)} P_{ji}\right)$ *and* $\bar{\alpha} = \max_i \alpha_i$.

Proof. The optimal set X^* is nonempty. Thus, Eq. (15) holds for any $x^* \in X^*$. From approximate subgradient relation (4) it follows

$$\nabla f_i(v_{i,k})^T(v_{i,k}-x^*) \ge f_i(\bar{y}_{k-1}) - f_i(x^*) - C\|v_{i,k}-\bar{y}_{k-1}\|.$$

The preceding relation and Eq. (15) yield for all $i \in \{I_k, J_k\}$ and $k \ge 1$,

$$\mathsf{E}\left[\|x_{i,k}-x^*\|^2 \mid \mathcal{F}_{k-1}, I_k, J_k\right] \le \|v_{i,k}-x^*\|^2 - 2\alpha_i(f_i(\bar{y}_{k-1}) - f_i(x^*)) \\ +2\alpha_i C\|v_{i,k}-\bar{y}_{k-1}\| + \alpha_i^2(C+\nu)^2,$$

where $C_X = \max_{x,y\in X}\|x-y\|$. The preceding relation holds when $i \in \{I_k, J_k\}$, which happens with probability γ_i. When $i \notin \{I_k, J_k\}$, we have $x_{i,k} = v_{i,k}$ (see Eq. (7)), which happens with probability $1-\gamma_i$. Thus, by taking the expectation conditioned on $\mathcal{F}_{k-1}$, we obtain

$$\mathsf{E}\left[\|x_{i,k}-x^*\|^2 \mid \mathcal{F}_{k-1}\right] \le \mathsf{E}\left[\|v_{i,k}-x^*\|^2 \mid \mathcal{F}_{k-1}\right] - 2\gamma_i\alpha_i(f_i(\bar{y}_{k-1}) - f_i(x^*)) \\ +2\gamma_i\alpha_i C\mathsf{E}\left[\|v_{i,k}-\bar{y}_{k-1}\| \mid \mathcal{F}_{k-1}\right] + \gamma_i\alpha_i^2(C+\nu)^2.$$

Letting $\underline{\omega} = \min_{1\le i\le m}\{\gamma_i\alpha_i\}$ and $\bar{\omega} = \max_{1\le i\le m}\{\gamma_i\alpha_i\}$, and using

$$|f_i(\bar{y}_{k-1}) - f_i(x^*)| \le C\|\bar{y}_{k-1}-x^*\| \le CC_X,$$

which holds by the subgradient boundedness and the fact $\bar{y}_k \in X$, we see that

$$\mathsf{E}\left[\|x_{i,k}-x^*\|^2 \mid \mathcal{F}_{k-1}\right] \le \mathsf{E}\left[\|v_{i,k}-x^*\|^2 \mid \mathcal{F}_{k-1}\right] - 2\underline{\omega}(f_i(\bar{y}_{k-1}) - f_i(x^*)) \\ +2(\bar{\omega}-\underline{\omega})CC_X + 2\bar{\omega}C\mathsf{E}\left[\|v_{i,k}-\bar{y}_{k-1}\| \mid \mathcal{F}_{k-1}\right] + \bar{\omega}\bar{\alpha}(C+\nu)^2,$$

where $\bar{\alpha} = \max_{1\le i\le m}\alpha_i$. By summing the preceding inequalities over i, and using Eq. (8) with $x = x^*$ and Eq. (9) with $x = \bar{y}_{k-1} \in X$, we obtain

$$2\underline{\omega}\mathsf{E}[f(\bar{y}_{k-1}) - f(x^*)] \le \sum_{j=1}^{m} \mathsf{E}[\|x_{j,k-1} - x^*\|^2] - \sum_{i=1}^{m} \mathsf{E}[\|x_{i,k} - x^*\|^2]$$
$$+2m(\bar{\omega} - \underline{\omega})CC_X + 2\bar{\omega}C\sum_{j=1}^{m} \mathsf{E}[\|x_{j,k-1} - \bar{y}_{k-1}\|] + m\bar{\omega}\bar{\alpha}(C+\nu)^2,$$

where $f = \sum_{i=1}^{m} f_i$. Next, after dividing the preceding relation by $2\underline{\omega}$ and noting that by convexity and the boundedness of the subgradients of each f_i, we have

$$f(x_{i,k-1}) - f^* \le f(\bar{y}_{k-1}) - f^* + mC\|x_{i,k-1} - \bar{y}_{k-1}\|,$$

we obtain for all i,

$$\mathsf{E}[f(x_{i,k-1}) - f(x^*)] \le \frac{1}{2\underline{\omega}}\left(\sum_{j=1}^{m} \mathsf{E}[\|x_{j,k-1} - x^*\|^2] - \sum_{i=1}^{m} \mathsf{E}[\|x_{i,k} - x^*\|^2]\right)$$
$$+m(\rho - 1)CC_X + (\rho + m)\,C\sum_{j=1}^{m} \mathsf{E}[\|x_{j,k-1} - \bar{y}_{k-1}\|] + \frac{m}{2}\rho\bar{\alpha}(C+\nu)^2,$$

where $\rho = \frac{\bar{\omega}}{\underline{\omega}}$. By summing these relations from time 1 to time k, and then averaging with respect to k, we obtain

$$\frac{1}{k}\sum_{t=1}^{k} \mathsf{E}[f(x_{i,t-1}) - f(x^*)] \le \frac{1}{2k\underline{\omega}}\left(\sum_{j=1}^{m} \mathsf{E}[\|x_{j,0} - x^*\|^2] - \sum_{i=1}^{m} \mathsf{E}[\|x_{i,k} - x^*\|^2]\right)$$
$$+m(\rho - 1)CC_X + (\rho + m)C\frac{1}{k}\sum_{t=1}^{k}\sum_{j=1}^{m} \mathsf{E}[\|x_{j,t-1} - \bar{y}_{t-1}\|] + \frac{m}{2}\rho\bar{\alpha}(C+\nu)^2.$$

Letting $k \to \infty$ and using the relation

$$\limsup_{k\to\infty} \frac{1}{k}\sum_{t=1}^{k}\left(\sum_{j=1}^{m} \mathsf{E}[\|x_{j,k-1} - \bar{y}_{k-1}\|]\right) \le \limsup_{k\to\infty}\sum_{j=1}^{m} \mathsf{E}[\|x_{j,k-1} - \bar{y}_{k-1}\|],$$

we have for any i,

$$\limsup_{k\to\infty} \frac{1}{k}\sum_{t=1}^{k} \mathsf{E}[f(x_{i,t-1}) - f(x^*)] \le m(\rho - 1)CC_X$$
$$+(\rho + m)C\limsup_{k\to\infty}\sum_{j=1}^{m} \mathsf{E}[\|x_{j,k-1} - \bar{y}_{k-1}\|] + \frac{m}{2}\rho\bar{\alpha}(C+\nu)^2.$$

By Lemma 2 we have

$$\limsup_{k\to\infty}\sum_{j=1}^{m} \mathsf{E}[\|x_{j,k-1} - \bar{y}_{k-1}\|] \le \frac{\sqrt{2m}\,\bar{\alpha}}{1 - \sqrt{\lambda}}\,(C+\nu),$$

which together with the preceding relation yields for all i,

$$\limsup_{k\to\infty} \frac{1}{k}\sum_{t=1}^{k} (\mathsf{E}[f(x_{i,t-1})] - f(x^*)) \leq m\,(\rho-1)\,CC_X$$
$$+(\rho+m)C\frac{\sqrt{2m}\,\bar{\alpha}}{1-\sqrt{\lambda}}\,(C+\nu) + \frac{m}{2}\rho\bar{\alpha}(C+\nu)^2.$$

By using $C(C+\nu) \leq (C+\nu)^2$ and grouping the terms accordingly, we obtain the desired relation. □

By Proposition 2 and the convexity of f, we have for $u_{i,k} = \frac{1}{k}\sum_{t=1}^{k} x_{i,t-1}$,

$$\limsup_{k\to\infty} \mathsf{E}[f\,(u_{i,k})] \leq f^* + B,$$

where $B = m\,(\rho-1)\,CC_X + \bar{\alpha}\left((\rho+m)\frac{\sqrt{2m}}{1-\sqrt{\lambda}} + \frac{m}{2}\rho\right)(C+\nu)^2$. When the ratio $\rho = \frac{\max_i \gamma_i\alpha_i}{\min_i \gamma_i\alpha_i}$ is close to value 1, the bound is approximately given by $B \approx \bar{\alpha}\left((1+m)\frac{\sqrt{2m}}{1-\sqrt{\lambda}} + \frac{m}{2}\right)(C+\nu)^2$. In this case, the bound scales in the size m of the network as $m^{3/2}$, which is by order $1/2$ less than the scaling of the bound for the distributed consensus-based subgradient algorithm of [3], which scales at best as m^2.

5 Discussion

The bounds scale well with the size of the network. For strongly convex functions, the bound in Proposition 1 scales independently of the size of the network if the degrees of the nodes are about the same order and do not change with the size of the network. The bound in Proposition 2 scales as $m\sqrt{m}$ with the size m of the network. In our development, we have assumed that the network topology is static, which may not be realistic in some applications. Of future interest is to investigate the algorithm for dynamic network topology.

References

1. Boyd S, Ghosh A, Prabhakar B, Shah D (2006) Randomized gossip algorithms. IEEE Trans. on Information Theory 52:2508–2530
2. Sundhar RS, Nedić A, Veeravalli VV (2009) Asynchronous Gossip Algorithm for stochastic optimization. Proc. of IEEE Conf. on Decision and Control
3. Sundhar RS, Veeravalli VV, Nedić A (2009) Distributed and non-autonomous power control through distributed convex optimization. IEEE INFOCOM

Part II

Nonlinear Optimization

On Hessian- and Jacobian-Free SQP Methods - a Total Quasi-Newton Scheme with Compact Storage

Torsten Bosse[1], Andreas Griewank[2], Lutz Lehmann[3], and
Volker Schloßhauer[4]

[1] † Humboldt-Universität zu Berlin, Institut für Mathematik, Unter den Linden 6, 10099 Berlin, Germany, bosse@math.hu-berlin.de

[2] † Humboldt-Universität zu Berlin, Institut für Mathematik, Unter den Linden 6, 10099 Berlin, Germany, griewank@math.hu-berlin.de

[3] Humboldt-Universität zu Berlin, Institut für Mathematik, Unter den Linden 6, 10099 Berlin, Germany, llehmann@math.hu-berlin.de

[4] † Weierstraß-Institut für Angewandte Analysis und Stochastik, Mohrenstr. 39, 10117 Berlin, Germany, schlosshauer@wias-berlin.de

Keywords: NLP; Total quasi-Newton method; Limited-memory; Symmetric rank-one update; Compact representation (with damping); Algorithmic Differentiation; BFGS

Summary. In this paper we describe several modifications to reduce the memory requirement of the total quasi-Newton method proposed by Andreas Griewank et al..

The idea is based on application of the compact representation formulae for the well-known BFGS and SR1 update for unconstrained optimization. It is shown how these definitions can be extended to a total quasi-Newton approach for the constrained case.

A brief introduction to the limited-memory approach is described in the present paper using an updated null-space factorization for the KKT system as well as an efficient numerical implementation of the null-space method in which the null-space representation is not stored directly. It can be proven that the number of operations per iteration is bounded by a bilinear order $\mathcal{O}(n \cdot \max(m, l))$ instead of a cubic order $\mathcal{O}(m \cdot n^2)$ for standard SQP methods. Here n denotes the number of variables, m the maximal number of active constraints, and l the user-selected number of stored update vectors.

† Supported by the DFG Research Center MATHEON "Mathematics for Key technologies", Straße des 17. Juni 136, 10623 Berlin, Germany, www.matheon.de

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_6, © Springer-Verlag Berlin Heidelberg 2010

1 Introduction

The main goal of this work is to sketch an efficient approach to solve *nonlinear programs* of the form:

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

Here $c_{\mathcal{I}} = (c_i)_{i\in\mathcal{I}}$ and $c_{\mathcal{E}} = (c_i)_{i\in\mathcal{E}}$ denote the mappings composed of the *inequality constraint* functions $c_i : \mathbb{R}^n \to \mathbb{R}$, $i \in \mathcal{I}$, and *equality constraint* functions $c_i : \mathbb{R}^n \to \mathbb{R}, i \in \mathcal{E}$, where f and c_i, $i \in \mathcal{I} \cup \mathcal{J}$, are at least C^2. Also, the existence of a regular solution $x^* \in \mathbb{R}^n$ where LICQ holds is assumed.

An active-set strategy can be used to handle the inequalities. Assume that the active set $\mathcal{A}(x^*)$ of such a solution x^* is known and denote by $c_{\mathcal{A}} : \mathbb{R}^n \to \mathbb{R}^m$ the corresponding constraint mapping. Then solving the NLP is equivalent to finding the solution of the *reduced equality constraint problem*:

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} f(x) \\ \text{s.t.}\ &c_{\mathcal{A}}(x) = 0. \end{aligned}$$

Let λ_i be the Lagrange multiplier for the constraint c_i. The *Lagrangian*

$$\mathcal{L}(x,\lambda) := f(x) + \sum_{i\in\mathcal{A}} \lambda_i c_i(x)$$

associated with the reduced equality-constrained problem can be used to state the first-order optimality condition for the stationary point (x^*, λ^*):

$$\nabla_{x,\lambda_{\mathcal{A}}} \mathcal{L}(x^*, \lambda^*) = 0. \tag{1}$$

According to [4], a total quasi-Newton approach can be applied to determine (x^*, λ^*). In such an approach the reduced Hessian of the Lagrangian and the constraint Jacobian are approximated by some matrices $B \approx \nabla^2_{xx}\mathcal{L}(x,\lambda)$ and $A \approx c'_{\mathcal{A}}(x)$. Applying a null-space method based on an *extended QR factorization* $A = [L, 0][Y, Z]^\top$, one obtains the approximating *null-space factorized KKT system*

$$\begin{pmatrix} Y^\top BY & Y^\top BZ & L^\top \\ Z^\top BY & Z^\top BZ & 0 \\ L & 0 & 0 \end{pmatrix} \begin{pmatrix} Y^\top s \\ Z^\top s \\ \sigma \end{pmatrix} = - \begin{pmatrix} Y^\top \nabla_x \mathcal{L}(x,\lambda) \\ Z^\top \nabla_x \mathcal{L}(x,\lambda) \\ c_{\mathcal{A}}(x) \end{pmatrix}$$

for (1) that can be efficiently updated by low-rank formulae. Here, the matrix $Z \in \mathbb{R}^n \times \mathbb{R}^{n-m}$ contains an orthonormal null-space basis of A. The right-hand side of the equation is obtained exactly by use of the backward mode in Algorithmic Differentiation (cf. [3]). The approximate *projected Hessian* $Z^\top BZ$ is kept positive definite throughout the optimization procedure, since the exact one will have this property near local minima where second-order sufficiency conditions hold.

2 A Limited-Memory Approach for the SR1 Method

2.1 Compact Representation Formula

Consider a symmetric rank-one update (SR1) of the Hessian B defined by

$$B_+ = B + \beta \frac{(w - Bs)(w - Bs)^\top}{(w - Bs)^\top s} \quad \text{with} \quad (w - Bs)^\top s \neq 0$$

where $\beta \in (0, 1]$ is a damping parameter. In order to avoid the complete fill-in caused by the addition of low-rank terms, one prefers to store the triples $(s, w, \beta) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}$ where $s := x_+ - x$ and $w := \nabla_x \mathcal{L}(x_+, \lambda) - \nabla_x \mathcal{L}(x, \lambda)$. Unless $w^\top s = 0$ the pairs (s, w) are scaled throughout such that $|w^\top s| = 1$, which leaves the secant condition $w = B_+ s$ unaffected.

In the following a sequence of damped SR1 updates identified with (s_j, w_j, β_j), $j \in \{0, \dots, l-1\}$, is applied to $B^{(0)} := \gamma I$ using a compact representation formula, which is well-known for many quasi-Newton updates. The scaled update vectors and scalar products are, therefore, arranged in matrices

$$\begin{aligned} S &:= \big(s_0 \cdots s_{l-1}\big) \in \mathbb{R}^{n \times l}, \quad W := \big(w_0 \cdots w_{l-1}\big) \in \mathbb{R}^{n \times l}, \\ Q &\in \mathbb{R}^{l \times l} \quad \text{with} \quad Q_{ih} := Q_{hi} = w_{i-1}^\top s_{h-1} (i \geq h), \\ P &\in \mathbb{R}^{l \times l} \quad \text{with} \quad P_{ih} := P_{hi} = s_{i-1}^\top w_{h-1} (i \geq h). \end{aligned}$$

Theorem 1 (SR1 - Compact representation formula).
Let l be the number of damped regular SR1 updates $(s_j, w_j, \beta_j)_{j=0}^{l-1}$, i.e.

$$(w_j - B^{(j)} s_j)^\top s_j \neq 0, \; \beta_j \neq 0 \quad \forall j \in \{0, \dots, l-1\},$$

applied to the initial matrix $B^{(0)} = \gamma I$ with $B^{(j)}$ defined as the intermediate matrix after applying the first $j \leq l$ updates. Then $M := P - D - \gamma S^\top S \in \mathbb{R}^{l \times l}$ is invertible and $B = B^{(l)}$ is given by

$$B = \gamma I + (W - \gamma S) M^{-1} (W - \gamma S)^\top \tag{2}$$

where D denotes the diagonal matrix $D = \operatorname{diag}(D_{jj})_{j=0}^{l-1}$ with

$$D_{jj} := (1 - \beta_j^{-1})(w_j - B^{(j)} s_j)^\top s_j.$$

A compact representation formula for the BFGS update can be found in [2].

Remark: Equation (2) represents a generalization of the usual SR1 update formula in [2]. In the undamped case, i.e. $(\beta_j)_{j=0}^{l-1} = 1$, D vanishes.

Due to the Sherman Morrison Woodbury formula, one obtains a similar formula for the inverse. Therefore, define $N := Q + D - \gamma^{-1} W^\top W$ and verify

$$B^{-1} = \gamma^{-1} I + (S - \gamma^{-1} W) N^{-1} (S - \gamma^{-1} W)^\top .$$

The compact representation formulae offer a number of advantages over the full-storage implementation. First and foremost the space for storing B is reduced to a pair of low-rank matrices S and W and the scalar γ that ideally represents the average eigenvalue of B. In a limited-memory approach the number l of updates is fixed, so only the most recent update vectors are kept inside S and W. The computational effort for adding (or replacing) update vectors for B is bounded by $\mathcal{O}(l \cdot n)$ compared to $\mathcal{O}(n^2)$ for SR1 updates. The bound $\mathcal{O}(l \cdot n + l^3)$ holds for multiplying vectors by B or its inverse B^{-1}. If $l \ll \sqrt{n}$ is small, the factorization effort for M and N stays negligible.

On the other hand, not storing all updates causes the loss of superlinear convergence (see [6]), which may possibly increase the overall computational effort.

2.2 Maintaining the Positive Definiteness of the Hessian

Positive definiteness of $Z^\top B Z$ and maximal rank of A imply unique solvability of the KKT system. Unlike the BFGS update, the SR1 update does not necessarily preserve the positive definiteness of $Z^\top B Z$. A remedy is proposed in [7] for the limited-memory approach. It consists of both determining suitable values for the damping parameters β_i and adapting the scaling parameter γ. More specifically, one obtains the following statement (cf. [7]) for $\bar{Q} := Q + D \in \mathbb{R}^{l \times l}$ as defined before including a constructive proof for γ:

Lemma 1. *If $\bar{Q}$ is positive definite,*[5] *then there exists $\Gamma > 0$ such that B becomes positive definite for all $\gamma > \Gamma$.*

Proof. Consider auxiliary matrices $T_1, T_2, T_3 \in \mathbb{R}^{(n+l)\times(n+l)}$ defined by

$$T_1 := \begin{pmatrix} \gamma I & U \\ U^\top & -M \end{pmatrix} \quad \text{with} \quad U = (W - \gamma S),$$

$$T_2 := \begin{pmatrix} I & UM^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} \gamma I & U \\ U^\top & -M \end{pmatrix} \begin{pmatrix} I & 0 \\ M^{-1}U^\top & I \end{pmatrix} = \begin{pmatrix} B & 0 \\ 0 & -M \end{pmatrix},$$

$$T_3 := \begin{pmatrix} I & 0 \\ -\gamma^{-1}U^\top & I \end{pmatrix} \begin{pmatrix} \gamma I & U \\ U^\top & -M \end{pmatrix} \begin{pmatrix} I & -\gamma^{-1}U \\ 0 & I \end{pmatrix} = \begin{pmatrix} \gamma I & 0 \\ 0 & -M - \gamma^{-1}U^\top U \end{pmatrix}.$$

Simplifying the last equation one recovers the middle term N of B^{-1}:

$$-M - \gamma^{-1} U^\top U = W^\top S + S^\top W - P + D - \gamma^{-1} W^\top W = \bar{Q} - \gamma^{-1} W^\top W.$$

Due to Sylvester's law, the inertias of T_1, T_2 and T_3 coincide. So, one can deduce: B is positive definite (as $B^{(0)} = \gamma I$) if and only if $-M$ and N have the same inertia. Furthermore, if $\bar{Q}$ is positive definite, then there exists $\Gamma > 0$ such that N, T_3 and B become positive definite. □

[5] The assumption is reasonable, as in quadratic programming without damping one retrieves: $\bar{Q} = Q = S^\top \nabla^2_{xx} \mathcal{L}(x, \lambda) S$ is positive definite.

The assumption of the previous lemma can be guaranteed by damping a new secant update pair (s_{new}, w_{new}) to prevent the compact representation (2) of the reduced Hessian losing its positive definiteness property. Therefore, consider the rank-two update formula that describes the replacement of a single update (s_h, w_h, β_h) by $(s_{new}, w_{new}, \beta_{new})$ for $h \in \{0, \ldots, l-1\}$ in $\bar{Q}$:

$$\bar{Q}_{new} := \bar{Q} + \frac{1}{2}(e_h + d)(e_h + d)^\top - \frac{1}{2}(e_h - d)(e_h - d)^\top + \beta_{new} e_h e_h^\top \qquad (3)$$
$$\text{and} \quad (d_j)_{j=0}^{l-1} := \begin{cases} s_j^\top w_{new} - \bar{Q}_{hj} & \text{if } (j \neq h) \\ \frac{1}{2}(s_{new}^\top w_{new} - \bar{Q}_{hh}) & \text{otherwise.} \end{cases}$$

Since the largest eigenvalue of $\bar{Q}_{new}$ cannot grow rapidly but its smallest one could become zero or even negative one can control its conditioning by a Powell-like test on the determinant. A suitable choice for the damping parameter β_{new} can then be derived by investigating the determinant ratio for $\bar{Q}$ and the updated version $\bar{Q}_{new}$:

Lemma 2 (Determinant ratio for damping parameters).
Let $(s_i, w_i, \beta_i)_{i=0}^{l-1}$ be a sequence of l regular SR1 updates and $(s_{new}, w_{new}, \beta_{new})$ be a regular SR1 update replacing (s_h, w_h, β_h), $h \in \{0, \ldots, l-1\}$. Define the determinant ratio function $q : \mathbb{R} \to \mathbb{R}$ as

$$q(\beta_{new}) := \frac{\det \bar{Q}_{new}}{\det \bar{Q}}.$$

Then it holds for $b := \bar{Q}^{-1} e_h$ and $c := \bar{Q}^{-1} d$:

$$q(\beta_{new}) = b_h \beta_{new} + c_h^2 + 2c_h - b_h c^\top d + 1.$$

Choosing β_{new} such that $q(\beta_{new}) \in [1/\mu, \mu]$ maintains the positive definiteness of $\bar{Q}_{new}$ after the update (3) where $1 < \mu$ is a fixed constant. In the next step one can numerically try ascending values for γ and verify the positive definiteness of B by analyzing the inertias of $-M$ and N according to the first Lemma.

2.3 Constrained Optimization and Limited-Memory

Consider again the factorized KKT system of the equality-constrained problem for computing a total quasi-Newton step, as described in Section 1:

$$\begin{pmatrix} Y^\top BY & Y^\top BZ & L^\top \\ Z^\top BY & Z^\top BZ & 0 \\ L & 0 & 0 \end{pmatrix} \begin{pmatrix} Y^\top s \\ Z^\top s \\ \sigma \end{pmatrix} = - \begin{pmatrix} Y^\top \nabla_x \mathcal{L}(x, \lambda) \\ Z^\top \nabla_x \mathcal{L}(x, \lambda) \\ c_{\mathcal{A}}(x) \end{pmatrix}. \qquad (4)$$

Then the limited-memory approach can easily be incorporated by replacing B with the compact representation formula. Hence, instead of storing the factors $Y^\top BY$, $Z^\top BY$, and $Z^\top BZ$, it is sufficient to store and update only the matrices W, S, and two smaller matrices in $\mathbb{R}^{l\times l}$. In addition, the necessary matrix-vector products can be calculated directly by multiplication from right to left using the reformulation

$$\begin{aligned}
Y^\top BY &= \gamma I + (Y^\top W - \gamma Y^\top S)M^{-1}(Y^\top W - \gamma Y^\top S)^\top,\\
Y^\top BZ &= (Y^\top W - \gamma Y^\top S)M^{-1}(Z^\top W - \gamma Z^\top S)^\top,\\
Z^\top BZ &= \gamma I + (Z^\top W - \gamma Z^\top S)M^{-1}(Z^\top W - \gamma Z^\top S)^\top, \text{ and}\\
(Z^\top BZ)^{-1} &= \gamma^{-1} I + \gamma^{-2}(Z^\top W - \gamma Z^\top S)N^{-1}(Z^\top W - \gamma Z^\top S)^\top
\end{aligned}$$

where the middle matrices M, $N \in \mathbb{R}^{l\times l}$ are now defined as follows:

$$M := P - D - \gamma S^\top S \text{ and } N := -M - \gamma^{-1}(W - \gamma S)^\top ZZ^\top (W - \gamma S).$$

Since the damping of the update and the choice of γ discussed in section 2.2 ensures the positive definiteness of B, this property will be shared by the reduced Hessian $Z^\top BZ$. A major concern now is to handle the matrices Y and Z of the extended QR-factorization, which also need to be stored. Consequently, one needs at least a complexity of order $\mathcal{O}(n^2)$ to store the Jacobian factorization, even for problems with a few active constraints. The next section gives a possible solution to this drawback.

2.4 Avoidance of the Null-space Factor *Z*

When using a partial limited-memory method in conjunction with a total quasi-Newton approach and a null-space factorized KKT system, a significant amount of memory is expended on the matrix Z containing the null-space basis of the Jacobian. This fact reduces the benefits of the limited-memory approach, especially, if only a small number of constraints is active. The following summarizes how the partial limited-memory approach can be improved by utilizing the orthonormality relation $ZZ^\top + YY^\top = I$ for the range- and null-space representation $[Y, Z]$.

In this case the storage of the $(n - m) \times n$ matrix Z can be avoided without any loss in theory. According to [1], Z is necessary neither to get a total quasi-Newton step nor for the update of the factorized KKT system (4) itself. Thus, a further reduction of the computational effort in a realization of an algorithm is possible by eliminating Z. Also a bilinear upper bound on memory allocation and the operation count per iteration is obtained.

Theorem 2 (Solving KKT without Z).
The solution of the approximated null-space factorized KKT system (4)

$$\begin{aligned}
s &= -YL^{-1}c_{\mathcal{A}}(x) - Z(Z^\top BZ)^{-1}(Z^\top \nabla_x \mathcal{L}(x,\lambda) - Z^\top BYL^{-1}c_{\mathcal{A}}(x))\\
\sigma &= -L^{-\top}(Y^\top \nabla_x \mathcal{L}(x,\lambda) + Y^\top BYY^\top s + Y^\top BZZ^\top s)
\end{aligned}$$

can be computed ***without using Z*** *if the Hessian approximation B is given as a low-rank perturbation of a multiple of the identity matrix.*

Proof. Consider the computation of the vector s, which can be written as

$$s = -YL^{-1}c_{\mathcal{A}}(x) - Z(Z^\top BZ)^{-1}Z^\top \left[\nabla_x \mathcal{L}(x,\lambda) - BYL^{-1}c_{\mathcal{A}}(x)\right].$$

Here only the factor $Z(Z^\top BZ)^{-1}Z^\top$ is interesting, as it depends on Z. With reference to section 2.3, $(Z^\top BZ)^{-1}$ is given by

$$\begin{aligned}(Z^\top BZ)^{-1} = \gamma^{-1}I + \gamma^{-2}(Z^\top W - \gamma Z^\top S)[-M \\ -\gamma^{-1}(W-\gamma S)ZZ^\top(W-\gamma S)]^{-1}(Z^\top W - \gamma Z^\top S)^\top.\end{aligned}$$

Multiplication on left and right by Z and its transpose, respectively, yields

$$\begin{aligned}Z(Z^\top BZ)^{-1}Z^\top = \gamma^{-1}ZZ^\top + \gamma^{-2}ZZ^\top(W-\gamma S)\,[-M \\ -\gamma^{-1}(W-\gamma S)^\top ZZ^\top(W-\gamma S)]^{-1}(W-\gamma S)^\top ZZ^\top.\end{aligned}$$

Applying the identity $ZZ^\top = I - YY^\top$ to the equation above as well as to the formula for the Lagrange multiplier step via

$$\begin{aligned}\sigma &= -L^{-\top}\left[Y^\top \nabla_x \mathcal{L}(x,\lambda) + Y^\top BYY^\top s + Y^\top BZZ^\top s\right] \\ &= -L^{-\top}Y^\top\left[\nabla_x \mathcal{L}(x,\lambda) + Bs\right]\end{aligned}$$

concludes the proof. □

2.5 Improving Computational Efficiency

From a numerical point of view the most time-consuming part per iteration is the step computation. Here several matrix-matrix products of order no less than $\mathcal{O}(n \cdot m \cdot l)$ would be necessary, since the reformulation

$$Z(Z^\top BZ)^{-1}Z^\top = (\gamma^{-1}I + \gamma^{-2}ZZ^\top(W-\gamma S)N^{-1}(W-\gamma S)^\top)ZZ^\top$$

involves a computation and factorization of the middle matrix N:

$$N = -M - \gamma^{-1}(W-\gamma S)(I - YY^\top)(W-\gamma S) \in \mathbb{R}^{l\times l}.$$

As proven in [1], the basic idea to overcome this drawback is to avoid recomputation of N from scratch and to apply Hessian and Jacobian updates directly to the matrix N.

Hence, one can show by multiplication from right to left that the whole step-computation has bilinear complexity $\mathcal{O}(n \cdot \max(m,l))$ because the remaining matrix-matrix additions as well as matrix-vector products can be considered as cheap, i.e. of bilinear complexity. Note that N can be factorized from scratch without exceeding $\mathcal{O}(n \cdot l)$ operations for $l \ll n$ sufficiently small.

The update of the matrix N due to changes of the Hessian, the Jacobian, and the scaling parameter γ is examined in three propositions, where it is proven that the effort is bounded by $\mathcal{O}(n \cdot max(m,l))$ operations. Since the proofs are quite analogous, only the one for the Hessian updates is given.

Proposition 1 (Updating N - Hessian updates).
The matrix N can be directly updated with $\mathcal{O}(n \cdot max(m,l))$ operations if the Hessian is subject to a rank-one modification.

Proof. Three different actions can be performed if the Hessian is updated in the limited-memory case:

1. A new secant pair (s_i, w_i) is added to (S, W),
2. an old pair (s_i, w_i) is removed from (S, W), or
3. an old update (s_i, w_i) is exchanged by a new one (s_{new}, w_{new}).

In all these cases the matrix N needs to be modified as it depends on (S, W). The basic idea of the proof is to represent these changes as a constant number of low-rank updates. Therefore, not only the matrices S and W will be stored and updated but also $S^\top Y$, $W^\top Y$, and all summands of N up to transpositions. All the three cases will be illustrated on $S^\top W$.

1. Appending a new update pair (s_{new}, w_{new}) to the set (S, W) by setting $(S, W)_+ = ((s_1, \ldots, s_{i-1}, s_i = s_{new}), (w_1, \ldots, w_{i-1}, w_i = w_{new}))$ results in an extended matrix plus a rank-two update:

$$\begin{aligned}(S^\top W)_+ &= \begin{bmatrix} S^\top W & S^\top w_i \\ s_i^\top W & s_i^\top w_i \end{bmatrix} \\ &= \begin{bmatrix} W^\top S & 0 \\ 0 & 0 \end{bmatrix} + (S^\top w_i)e_i^\top + e_i(s_i^\top W)^\top + w_i^\top s_i(e_i e_i^\top).\end{aligned}$$

2. Assume the secant pair (s_i, w_i) that shall be deleted is in last position in (S, W), i.e. $(S, W) = ((s_1, \ldots, s_i), (w_1, \ldots, w_i))$. Otherwise use the routine described in the next point to exchange it with the last one. Then the secant pair can be removed by erasing the last row and column of $S^\top W$.
3. Exchanging a secant pair (s_i, w_i) by a new one can be realized by a rank-two update on (S, W) with $\tilde{s} := (s_{new} - s_i)$ and $\tilde{w} := (w_{new} - w_i)$:

$$(S^\top W)_+ = S^\top W + e_i \tilde{s}^\top W + S^\top \tilde{w} e_i^\top + \tilde{s}^\top \tilde{w}(e_i e_i^\top).$$

Obviously, the operation count for the updates of all summands is dominated by two extra calculations including the Y-factor, i.e. $s_{new}^\top Y$ and $w_{new}^\top Y$, where the numerical effort is of order $\mathcal{O}(n \cdot m)$. Evaluating the remaining expressions without Y is cheap, e.g. the expression $e_i(s_i^\top W)^\top$ can be computed by first evaluating $s_i^\top W$ and storing this vector. In these cases the complexity bound $\mathcal{O}(n \cdot l)$ is not exceeded. Applying the results on N, one derives that the new middle matrix N_+ is given by a sequence of rank-one updates:

1. Appending a new secant pair (s_i, w_i) to (S, W) gives:

$$N_+ = \begin{bmatrix} N & 0 \\ 0 & 0 \end{bmatrix} + \sum_{j=1}^{8} \lambda_j (e_i v_j^\top + v_j e_i^\top) + \sum_{j=9}^{16} \lambda_j (e_i e_i^\top),$$

2. using MATLAB-like notation, the deletion of (s_i, w_i) in (S, W) yields:

$$N_+ = \left(N + \sum_{j=1}^{8} \lambda_j (e_i v_j^\top + v_j e_i^\top) + \sum_{j=9}^{16} \lambda_j (e_i e_i^\top) \right) [1:k-1; 1:k-1],$$

3. and exchanging (s_i, w_i) with (s_{new}, w_{new}) results in:

$$N_+ = N + \sum_{j=1}^{8} \lambda_j (e_i v_j^\top + v_j e_i^\top) + \sum_{j=9}^{16} \lambda_j (e_i e_i^\top)$$

where the vectors v_j and scalars λ_j are defined by the performed action. □

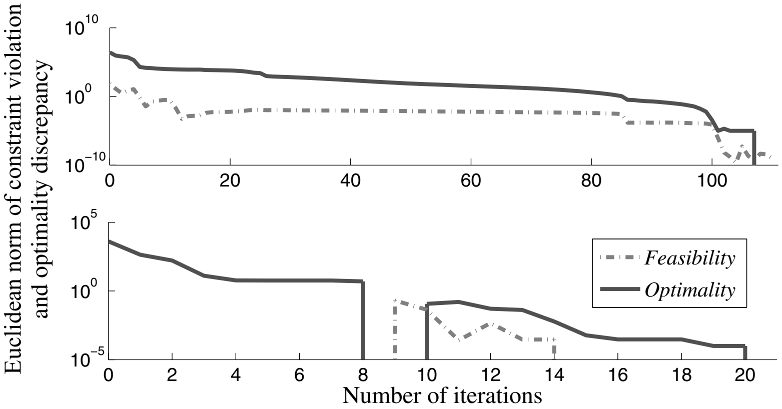
Hence, the following result is obtained by a careful implementation of the linear algebra for the updating of the factorized KKT system.

Theorem 3 (Bosse).
For a partial limited-memory approach on a total quasi-Newton method with an updated null-space factorized KKT system, the needed memory size and computational effort per iteration are both of order $\mathcal{O}(nm + nl + l^3)$.

More details on the proof can be found in [1], Chapter 6: 'Zed is Dead'.

3 Examples

The effectiveness of the presented method has been verified on the two examples LUKVLE3 (top) and LUKVLI9 (bottom) from the CUTEr test set.



Here the number of constraints is small ($m = 2$, $m = 6$) , whereas the number of variables is comparatively large ($n \approx 10000$). For $l = 4$ secant pairs in storage, the two problems were solved within 111 and 20 iterations, respectively. Thus, the overall effort $\sim 100 \cdot 6 \cdot 10^4$ arithmetic operations was less than that for one null-space factorization of the full KKT system. IPOPT takes 9 and 33 steps, respectively, using full first- and second-order derivative information!

4 Conclusion

This article summarizes our recent research on total quasi-Newton methods for nonlinear programming. A practical implementation of the limited-memory SR1 method is presented. It avoids the explicit storage of the Hessian and reduces the computational effort for quasi-Newton updates to about $\mathcal{O}(l \cdot n)$ operations. A null-space factorized KKT system in the constrained case is reformulated by means of compact representation formulae and solved efficiently using an updated QR decomposition of the Jacobian. The new approach circumvents the necessity of storing the matrix Z for the solution of the system while reducing the computational effort per iteration to the bilinear complexity $\mathcal{O}(n \cdot \max(l, m))$. This should be particularly beneficial on dense large-scale problems with a small set of active constraints $m \ll n$.

The quoted results for the large-scale problems LUKVLE3 and LUKVLI9 indicate acceptable linear convergence rates even for a small number of stored secant pairs ($l = 4$) with drastic reduction in computational effort per iteration. More runs on the CUTEr test set have shown as a rule of thumb that the choice of l between ~ 5 and ~ 15 results in a good balance between an acceptable linear convergence rate and an effective step computation.

A further reduction in storage and operations count is envisioned by a *semi-normal* approach that is based on a range-space method. In this case also the storage of the range-space basis Y is omitted. The matrix-vector products including Y are replaced by an extra Algorithmic Differentiation operation. A smart updating of the triangular matrix L reduces the effort to the order $\mathcal{O}(m^2/2 + n \cdot l)$.

References

1. Bosse T (2009) A Derivative-matrix-free NLP Solver without Explicit Nullspace Representation. Diploma Thesis, Humboldt Universität zu Berlin, Berlin
2. Byrd R, et al. (1994) Representations of quasi-Newton matrices and their use in limited-memory methods, Math. Programming 63:129–156
3. Griewank A, Walther A (2008) Evaluating derivatives. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA
4. Griewank A, Walther A, Korzec M (2007) Maintaining factorized KKT systems subject to rank-one updates of Hessians and Jacobians, Optimization Methods & Software 22:279–295
5. Korzec M (2006) A General-Low-Rank Update-Based Quadratic Programming Solver. Diploma Thesis, Humboldt Universität zu Berlin, Berlin
6. Nocedal J, Wright S (2006) Numerical Optimization, Springer Series in Operations Research, 2nd Edt.
7. Schloßhauer V (2009) Strukturausnutzung und Speicherplatzbegrenzung für hochdimensionale, nichtlineare Optimierung. Diploma Thesis, Humboldt Universität zu Berlin, Berlin

Approximate Geometric Ellipsoid Fitting: A CG-Approach

Martin Kleinsteuber[1] and Knut Hüper[2]

[1] Institute of Data Processing, Technische Universität München, Germany,
Kleinsteuber@tum.de

[2] Department of Mathematics, University of Würzburg, Germany
Hueper@mathematik.uni-wuerzburg.de

Summary. The problem of geometric ellipsoid fitting is considered. In connection with a conjugate gradient procedure a suitable approximation for the Euclidean distance of a point to an ellipsoid is used to calculate the fitting parameters. The approach we follow here ensures optimization over the set of all ellipsoids with codimension one rather than allowing for different conics as well. The distance function is analyzed in some detail and a numerical example supports our theoretical considerations.

1 Introduction

The approximation of a set of data by an ellipsoid is an important problem in computer science and engineering, e.g. in computer vision or computer graphics, or more specifically, in 3D-reconstruction and virtual reality generation. Moreover, there are further applications in robotics [13], astronomy [18] and in metrology [2, 5, 17], as well.

Mathematically, the problem of fitting can often be expressed by a set of implicit equations depending on a set of parameters. For fixed parameters the set of equations often describes implicitly a smooth manifold, e.g. in those cases where the regular value theorem applies. The task then is to find a parameter vector, such that the corresponding manifold best fits a given set of data. As it is studied in the computer vision community, e.g. see [9, 8], a large class of computer vision problems actually falls into this category.

Certainly, there exists a variety of different ways to measure the quality of a fit, dependent on the application context. Here we focus on a certain problem of *geometric* fitting, namely, minimizing the sum of the squared Euclidean distances between the data points and the manifold. In a natural way this is a generalization of the well known linear orthogonal regression problem.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_7, © Springer-Verlag Berlin Heidelberg 2010

A quite different approach to geometric fitting comes under the name of *algebraic* fitting which we do not follow here. It turns out that in many cases the algebraic approach has to be distinguished from the geometric one. Firstly, it seems that the numerical treatment of the former is more feasible, mainly due to the fact that the underlying optimization problem is based on a vector space model, rather than modelled in a nonlinear differential manifold setting. This might be the reason why it was preferably studied in much detail in the past, see e.g. [1, 4, 6, 10, 14, 15, 19]. Secondly, geometric fitting does not necessarily support a traditional straightforward statistical interpretation, again typical for a computer vision application, see [9] for a thorough discussion of this aspect.

For early work in the spirit of our approach, see however [11].

As already mentioned above the parameter vector might vary itself over a smooth manifold. E.g. fitting an ellipsoid of codimension one in $\mathbb{R}^n$ to a set of data points sitting in $\mathbb{R}^n$ as well, amounts in an optimization problem over the set of *all* codimension one ellipsoids. As we will see below this set can be neatly parameterized by the product of $\mathbb{R}^n$ with the set $\mathcal{P}_n$ of symmetric positive definite $n \times n$-matrices, or equivalently, by the product of $\mathbb{R}^n$ with the set $\mathcal{R}_+^{n\times n}$ of $n \times n$ upper triangular matrices with *positive* diagonal entries.

In general, there exists no explicit formula for the Euclidean distance of a point to a set. We therefore will use a suitable approximation together with a conjugate-gradient-type procedure to compute the fitting parameters.

In this paper we will put an emphasis on the geometric fitting of ellipsoids of codimension one to data points. The approach we follow here ensures that we actually optimize over all *ellipsoids* of codimension one, rather than allowing for other or even all *conics* of codimension one, or even conics of any codimension as well.

The paper is organized as follows. In the next section we motivate the quality measure we use, namely a distance function which approximates the Euclidean distance of a point to an ellipsoid in a consistent manner, in a way made precise below. We investigate the local properties of this function and compare it with the Euclidean distance and with algebraic fitting.

Differentiability of the square of this function allows for a smooth optimization procedure. In the third section we briefly describe the global parameterization of the smooth manifold of all ellipsoids of codimension one in $\mathbb{R}^n$ and set the ground for a conjugate gradient algorithm living on this manifold. The last section briefly discusses the CG-method used here, supported by a numerical example.

2 Motivation of the Distance Function

In this section we introduce a new distance measure as an approximation of the Euclidean distance from a point to an ellipsoid. This measure has the advantage that, in contrast to the Euclidean distance, it can be expressed explicitly in terms of the ellipsoid parameters and is therefore suitable for optimization tasks. Moreover, it does not have the drawback of the measure that underlies algebraic fitting, where it might happen that, given a set of points, any ellipsoid that is large enough drives the corresponding cost arbitrarily small. We specify this phenomenon in Proposition 1 below.

Let $(\cdot)^\top$ denote transposition and let

$$\mathcal{E}_{Q,\tau} := \{q \in \mathbb{R}^n \mid (q-\tau)^\top Q(q-\tau) = 1\} \tag{1}$$

be an ellipsoid with center $\tau \in \mathbb{R}^n$ and positive definite $Q \in \mathcal{P}_n$. For ellipsoids centered at the origin we shortly write $\mathcal{E}_Q := \mathcal{E}_{Q,0}$. In order to fit an ellipsoid to a given set of data $y_i \in \mathbb{R}^n$, $i = 1, \ldots N$, a quality measure is required that reflects *how well* an ellipsoid fits the y_i's. There are two measures that arise in a natural way: the Euclidean distance and, since any ellipsoid defines a metric by considering it as a unit ball, the corresponding distance induced by Q. For $x, y \in \mathbb{R}^n$ denote by

$$\langle x, y\rangle_Q := x^\top Q y \tag{2}$$

the induced scalar product, the associated norm by $\|x\|_Q = (x^\top Q x)^{\frac{1}{2}}$, and the induced distance measure by

$$d_Q(x,y) := \|x-y\|_Q. \tag{3}$$

Lemma 1. *Let $x \in \mathbb{R}^n$. Then the Q-distance between x and $\mathcal{E}_Q$ is given by*

$$d_Q(x, \mathcal{E}_Q) = |1 - \|x\|_Q|. \tag{4}$$

The point of lowest Q-distance to x on $\mathcal{E}_Q$ is $\hat{x} = \frac{x}{\|x\|_Q}$.

Proof. Without loss of generality we might assume that $x \neq 0$. We compute the critical points of the function

$$a \colon \mathcal{E}_Q \to \mathbb{R}, \quad q \mapsto \|q-x\|_Q^2, \tag{5}$$

as follows. The tangent space $T_q\mathcal{E}_Q$ of $\mathcal{E}_Q$ at $q \in \mathcal{E}_Q$ is given by

$$T_q\mathcal{E}_Q := \{\xi \in \mathbb{R}^n \mid \xi^\top Q q = 0\}, \tag{6}$$

hence

$$\mathrm{D}\, a(q)\xi = 2\xi^\top Q(q-x) = -2\xi^\top Q x. \tag{7}$$

The derivative vanishes if and only if $q \in \mathbb{R}x$. A simple calculation then shows, that the minimum of a is given by

$$\hat{x} := \tfrac{x}{\|x\|_Q}. \tag{8}$$

Consequently,

$$d_Q(x, \mathcal{E}_Q) = d_Q(x, \hat{x}) = \|x - \hat{x}\|_Q = |1 - \|x\|_Q|. \tag{9}$$

□

The quality measure used in *algebraic fitting* is closely related to the Q-distance. It is defined by

$$d_{\mathrm{alg}}(x, \mathcal{E}_Q) = \left|1 - \|x\|_Q^2\right| \tag{10}$$

or, for general ellipsoids,

$$d_{\mathrm{alg}}(x, \mathcal{E}_{Q,\tau}) = \left|1 - \|x - \tau\|_Q^2\right|, \tag{11}$$

cf. [10]. Although this is easy to compute, minimizing the sum of squares of d_{alg} for a given set of noisy data points may not yield a desired result as the following proposition is stating.

Proposition 1. *Let $y_1, \ldots, y_N \in \mathbb{R}^n$ be given. Then for all $\varepsilon > 0$ there exists $\delta > 0$ and $\tau \in \mathbb{R}^n$ such that*

$$\sum_{i=1}^{N} d_{\mathrm{alg}}^2(y_i, \mathcal{E}_{\delta I_n,\tau}) < \varepsilon. \tag{12}$$

Proof. Let $\delta = \delta(\tau) = \frac{1}{\|\tau\|^2}$. The claim follows since

$$\sum_{i=1}^{N} d_{\mathrm{alg}}^2(y_i, \mathcal{E}_{\delta I_n,\tau}) = \sum_{i=1}^{N} (1 - \delta\|y_i - \tau\|^2)^2 = \sum_{i=1}^{N} (1 - \tfrac{\|y_i-\tau\|^2}{\|\tau\|^2})^2 \xrightarrow{\|\tau\|\to\infty} 0.$$

□

Given a convex set $\mathcal{C} \subset \mathbb{R}^n$ and a point $x \in \mathbb{R}^n$ outside $\mathcal{C}$, it is well known that there is a unique point $q \in \partial\mathcal{C}$ on the boundary of $\mathcal{C}$ such that $d(x, \partial\mathcal{C}) = d(x, q)$, cf. Chapter 2 in [3]. If x lies in the interior of $\mathcal{C}$, this needs not to be true anymore. However, in the case where $\partial\mathcal{C} = \mathcal{E}_Q$ is an ellipsoid, q depends smoothly on x in a neighborhood of $\mathcal{E}_Q$.

Lemma 2. *Let $x \in \mathbb{R}^n$ and let $\pi\colon \mathbb{R}^n \to \mathcal{E}_Q$ be such that $d(x, \mathcal{E}_Q) = d(x, \pi(x))$. Then π is smooth in a neighborhood of $\mathcal{E}_Q$ and*

$$\mathrm{D}\,\pi(x)|_{x=q} h = \left(\mathrm{id} - \tfrac{Qqq^\top Q}{q^\top Q^2 q}\right) h. \tag{13}$$

Proof. Let $x \in \mathbb{R}^n$ be arbitrary but fixed and let $e\colon \mathcal{E}_Q \to \mathbb{R}$ with $e(q) = \frac{1}{2}\|x - q\|^2$. The minimal value of e then is $d(x, \mathcal{E}_q)$. Differentiating yields the critical point condition, namely

$$\mathrm{D}e(q)\xi = \xi^\top(q - x) = 0 \quad \text{for all} \quad \xi \in T_q\mathcal{E}_Q = \{\xi \in \mathbb{R}^n \mid \xi^\top Qq = 0\}. \tag{14}$$

Now since $T_q\mathcal{E}_Q = (\mathrm{im}(Qq))^\perp = \mathrm{im}\left(\mathrm{id} - \frac{Qqq^\top Q^\top}{q^\top Q^2 q}\right)$, the critical point condition is equivalent to

$$\left(\mathrm{id} - \tfrac{Qqq^\top Q^\top}{q^\top Q^2 q}\right)(q - x) = 0. \tag{15}$$

Using $q^\top Qq = 1$ yields

$$(q^\top Q^2 q)(q - x) - Qq + Qqq^\top Qx = 0. \tag{16}$$

Consider now the function

$$F\colon \mathcal{E}_Q \times \mathbb{R}^n \to \mathbb{R}^n, \quad (q, x) \mapsto (q^\top Q^2 q)(q - x) - Qq + Qqq^\top Qx. \tag{17}$$

Then F is smooth and $F(q, q) = 0$ for all $q \in \mathcal{E}_Q$. We use the implicit function theorem to complete the proof. The derivatives of F with respect to the first and second argument, respectively, are

$$\begin{aligned} \mathrm{D}_1 F(q,x)\xi &= (2\xi^\top Q^2 q)q + (q^\top Q^2 q)\xi - Q\xi - (2\xi^\top Q^2 q)x + Q\xi q^\top Qx + Qq\xi^\top Qx \\ \mathrm{D}_2 F(q,x)h &= Qqq^\top Qh - (q^\top Q^2 q)h. \end{aligned} \tag{18}$$

Hence $\mathrm{D}_1\, F(q,q)\xi = q^\top Q^2 q\xi$ and notice that $q^\top Q^2 q > 0$. The implicit function theorem yields the existence of a neighborhood U around q and a unique smooth function $\tilde{\pi}\colon U \to \mathcal{E}_Q$ such that $F(\tilde{\pi}(x), x) = 0$. Using π defined as above, we get $F(\pi(x), x) = 0$. Moreover, the uniqueness of $\tilde{\pi}$ implies $\tilde{\pi}|_U = \pi|_U$. Furthermore,

$$0 = \mathrm{D}\, F(\pi(x), x)h = \mathrm{D}_1\, F(\pi(x), x)\, \mathrm{D}\, \pi(x)h + \mathrm{D}_2\, F(\pi(x), x)h \tag{19}$$

and hence

$$\begin{aligned} \mathrm{D}\, \pi(x)|_{x=q} h &= -(\mathrm{D}_1\, F(\pi(q), q)^{-1}\, \mathrm{D}_2\, F(\pi(q), q)h \\ &= -(q^\top Q^2 q)^{-1}(Qqq^\top Qh - q^\top Q^2 qh) = \left(\mathrm{id} - \tfrac{Qqq^\top Q}{q^\top Q^2 q}\right) h. \end{aligned} \tag{20}$$

□

As an approximation of the Euclidean distance $d(x, \mathcal{E}_Q)$, we consider the Euclidean distance between x and $\frac{x}{\|x\|_Q}$, cf. Figure 1, i.e.

$$\tilde{d}\colon \mathbb{R}^n \setminus \{0\} \to \mathbb{R}, \quad x \mapsto \left|1 - \|x\|_Q^{-1}\right| \|x\|. \tag{21}$$

The definition of $d(x, \mathcal{E}_Q)$ immediately yields

$$d(x, \mathcal{E}_Q) \leq \tilde{d}(x, \mathcal{E}_Q). \tag{22}$$

For large $\|x\|$ both d and $\tilde{d}$ tend to the same value, i.e.

$$\lim_{\|x\| \to \infty} \tfrac{\tilde{d}(x, \mathcal{E}_Q)}{d(x, \mathcal{E}_Q)} = 1. \tag{23}$$

An investigation of the derivatives yields the local behavior of $d, \tilde{d}$ and d_Q around some $q \in \mathcal{E}_Q$. It allows in particular to compare the first order approximations of the three distances: locally, $\tilde{d}$ behaves similar to the Euclidean

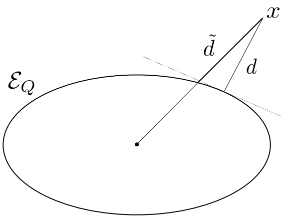


Fig. 1. Illustration of the distance measure $\tilde{d}$.

distance the more the ellipsoid becomes similar to a sphere. Moreover it shares the nice property with the Euclidean distance that it is invariant under scaling of Q, whereas the local behavior of d_Q depends on the absolute values of the eigenvalues of Q.

Proposition 2. *Let $x \in \mathbb{R}^n \setminus \{0\}$ and let $q \in \mathcal{E}_Q$. Let $\lambda_{\min}, \lambda_{\max}$ be the smallest, resp. largest eigenvalue of Q. Then*

$$\lim_{x \to q, x \notin \mathcal{E}_Q} \| \mathrm{D}\, d(x, \mathcal{E}_Q) \| = 1, \tag{24}$$

$$1 \leq \lim_{x \to q, x \notin \mathcal{E}_Q} \| \mathrm{D}\, \tilde{d}(x, \mathcal{E}_Q) \| \leq \sqrt{\tfrac{\lambda_{\max}}{\lambda_{\min}}}, \tag{25}$$

$$\sqrt{\lambda_{\min}} \leq \| \mathrm{D}\, d_Q(x, \mathcal{E}_Q) \| \leq \sqrt{\lambda_{\max}}, \quad \textit{for all } x \notin \mathcal{E}_Q, \tag{26}$$

where equality holds in the last equation either in the case of $Qx = \lambda_{\min} x$, or for $Qx = \lambda_{\max} x$.

Proof. Let $\pi(x)$ be defined as in Lemma (2), let $q \in \mathcal{E}_Q$ and let $U \subset \mathbb{R}^n$ be a neighborhood of q such that $\pi(x)$ is smooth. For $x \in U \setminus \mathcal{E}_Q$,

$$\begin{aligned} \mathrm{D}\, d(x, \mathcal{E}_Q) h &= \mathrm{D} \langle x - \pi(x), x - \pi(x) \rangle^{\frac{1}{2}} h = \left\langle h - \mathrm{D}\, \pi(x) h, \tfrac{x - \pi(x)}{\|x - \pi(x)\|} \right\rangle \\ &= \left\langle h, (\mathrm{id} - \mathrm{D}\, \pi(x))^\top \tfrac{x - \pi(x)}{\|x - \pi(x)\|} \right\rangle . \end{aligned} \tag{27}$$

Hence

$$\| \mathrm{D}\, d(x, \mathcal{E}_Q) \| = \left\| (\mathrm{id} - \mathrm{D}\, \pi(x))^\top \tfrac{x - \pi(x)}{\|x - \pi(x)\|} \right\| \leq \| (\mathrm{id} - \mathrm{D}\, \pi(x)) \|_{\mathrm{Frob}},$$

by submultiplicativity of the Frobenius norm. Therefore, using Eq. (13),

$$\lim_{x \to q, x \notin \mathcal{E}_Q} \| \mathrm{D}\, d(x, \mathcal{E}_Q) \| \leq \lim_{x \to q, x \notin \mathcal{E}_Q} \| (\mathrm{id} - \mathrm{D}\, \pi(x)) \|_{\mathrm{Frob}} = \left\| \tfrac{Q q q^\top Q}{q^\top Q^2 q} \right\|_{\mathrm{Frob}} = 1.$$

Now let

$$\gamma_x(t) = \tfrac{t x + (1 - t) \pi(x)}{\|x - \pi(x)\|}.$$

Then $\pi(\gamma_x(t)) = \pi(x)$ for all $t \in (0, 1)$ and

$$d(\gamma_x(t), \mathcal{E}_Q) = d(\gamma_x(t), \pi(x)) = |t|. \tag{28}$$

Therefore, by the Cauchy-Schwarz inequality and using $\|\dot{\gamma}_x(t)\| = 1$,

$$\begin{aligned} 1 = |\tfrac{\mathrm{d}}{\mathrm{d}\, t} d(\gamma_x(t), \mathcal{E}_Q)| &= |\,\mathrm{D}\, d(\gamma_x(t), \mathcal{E}_Q) \cdot \dot{\gamma}_x(t)| \\ &\leq \|\,\mathrm{D}\, d(\gamma_x(t), \mathcal{E}_Q)\| \|\dot{\gamma}_x(t)\| = \|\,\mathrm{D}\, d(\gamma_x(t), \mathcal{E}_Q)\|. \end{aligned} \tag{29}$$

This proves equation (24). For Eq. (25) note that

$$\|\,\mathrm{D}\, \tilde{d}(x, \mathcal{E}_Q)\| = \left\| \tfrac{x}{\|x\|}(1 - \|x\|_Q^{-1}) - \|x\| \tfrac{Qx}{\|x\|_Q^3} \right\|. \tag{30}$$

The first term tends to 0 for $x \to q$ and $\|x\|_Q$ tends to 1. It is therefore sufficient to consider the term $\|x\|\|Qx\|$. Substituting $y := Q^{\frac{1}{2}} x$, which implies $\|y\|^2 \to 1$ as $x \to q$, we obtain

$$\|x\|\|Qx\| = \tfrac{(y^\top Q^{-1} y)^{\frac{1}{2}}}{\|y\|} \tfrac{(y^\top Q y)^{\frac{1}{2}}}{\|y\|} \|y\|^2 \leq \sqrt{\tfrac{\lambda_{\max}}{\lambda_{\min}}} \|y\|^2, \tag{31}$$

hence

$$\lim_{x \to q, x \notin \mathcal{E}_Q} \|\,\mathrm{D}\, \tilde{d}(x, \mathcal{E}_Q)\| \leq \sqrt{\tfrac{\lambda_{\max}}{\lambda_{\min}}}. \tag{32}$$

On the other hand, the Cauchy-Schwarz inequality implies

$$\lim_{x \to q} \|x\|\|Qx\| \geq \lim_{x \to q} x^\top Q x = 1. \tag{33}$$

Finally, equation (26) follows since

$$\|\,\mathrm{D}\, d_Q(x, \mathcal{E}_Q)\| = \left\| \tfrac{Qx}{\|x\|_Q} \right\| = \left(\tfrac{x^\top Q^2 x}{x^\top Q x} \right)^{\frac{1}{2}}. \tag{34}$$

□

3 Parameterization of the set of ellipsoids

Given a set of data points $y_1, \ldots y_N$, our aim is to minimize the sum of the squares of the individual distance measures $\tilde{d}(y_i, \mathcal{E}_{Q,\tau})$ over the set of all ellipsoids $\mathcal{E}_{Q,\tau}$, i.e. over the set

$$E := \mathcal{P}_n \times \mathbb{R}^n. \tag{35}$$

Each positive definite matrix $Q \in \mathcal{P}_n$ possesses a unique Cholesky decomposition $Q = S^\top S$, with $S \in \mathcal{R}_+^{n \times n}$, and $\mathcal{R}_+^{n \times n}$ being the set of upper triangular $n \times n$-matrices with *positive* diagonal entries. Explicit formulas for computing the Cholesky decomposition, cf. [7], imply that

$$\mathcal{R}_+^{n \times n} \to \mathcal{P}_n, \quad S \mapsto S^\top S \tag{36}$$

is a diffeomorphism. We exploit this fact to obtain a global parameterization of E. Let $\mathcal{R}^{n\times n}$ be the set of upper triangular matrices. Then $\mathcal{R}^{n\times n} \simeq \mathbb{R}^{\frac{n(n+1)}{2}}$ and

$$\phi\colon \mathcal{R}^{n\times n} \to \mathcal{R}_+^{n\times n}, \qquad \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ 0 & r_{22} & \cdots & r_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & r_{nn} \end{bmatrix} \mapsto \begin{bmatrix} \mathrm{e}^{r_{11}} & r_{12} & \cdots & r_{1n} \\ 0 & \mathrm{e}^{r_{22}} & \cdots & r_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \mathrm{e}^{r_{nn}} \end{bmatrix} \tag{37}$$

is a diffeomorphism as well. Thus

$$\mathcal{R}^{n\times n} \times \mathbb{R}^n \to E, \quad (R,\tau) \mapsto (\phi(R)^\top \phi(R), \tau) \tag{38}$$

is a *global* parameterization of the set E of codimension one ellipsoids.

4 CG-method for fitting ellipsoids to data

Using the parameterization derived in the last section and recalling that

$$\tilde{d}(x, \mathcal{E}_{Q,\tau}) = |1 - \|x-\tau\|_Q^{-1}| \cdot \|x - \tau\|,$$

a conjugate gradient method was implemented for the following problem. Given a set of data points $y_1, \ldots y_N \in \mathbb{R}^n$, minimize

$$\begin{aligned} &f\colon \mathcal{R}^{n\times n} \times \mathbb{R}^n \to \mathbb{R}, \\ &(R,\tau) \mapsto \sum_{i=1}^{N} \left(1 - \left((y_i-\tau)^\top \phi(R)^\top \phi(R)(y_i-\tau)\right)^{-\frac{1}{2}}\right)^2 \|y_i - \tau\|^2. \end{aligned} \tag{39}$$

The step-size selection was chosen using a modified one dimensional Newton step, i.e. given a point $(R,\tau) \in \mathcal{R}^{n\times n} \times \mathbb{R}^n$ and a direction $(\xi, h) \in \mathcal{R}^{n\times n} \times \mathbb{R}^n$, we have chosen the step-size

$$t^* = -\frac{\frac{\mathrm{d}}{\mathrm{d}\,t} f(R+t\xi, \tau+th)}{\left|\frac{\mathrm{d}^2}{\mathrm{d}\,t^2} f(R+t\xi, \tau+th)\right|}. \tag{40}$$

The absolute value in the denominator has the advantage, that in a neighborhood of a nondegenerated minimum the step-size coincides with the common Newton step, whereas t^* is equal to the negative of the Newton step-size if $\frac{\mathrm{d}^2}{\mathrm{d}\,t^2} f(R+t\xi, \tau+th) > 0$. Our step-size selection is also supported by simulations showing that this modification is essential for not getting stuck in local maxima or saddle points. To derive the gradient of f, for convenience we define

$$\mu_i(t) := \phi(R+t\xi)(y_i - \tau + th). \tag{41}$$

Let $\operatorname{diag}(X)$ be the diagonal matrix having the same diagonal as the matrix X and let $\operatorname{off}(X)$ be the strictly upper triangular matrix having the same upper diagonal entries as X. Then

$$\dot{\mu}_i(0) = \left(\operatorname{diag}(\xi)\,\mathrm{e}^{\operatorname{diag}(R)} + \operatorname{off}(\xi)\right)(y_i - \tau) + \phi(R)h. \tag{42}$$

Lemma 3. *Let $\mu_i := \mu_i(0)$ and let $c_i := (\mu_i^\top \mu_i)^{-\frac{1}{2}}$. The gradient of f evaluated at (R, τ) is given by*

$$\nabla f(R,\tau) = \Big(\nabla_1 f(R,\tau), \nabla_2 f(R,\tau)\Big) \tag{43}$$

where

$$\begin{aligned}\nabla_1 f(R,\tau) =& 2\sum_{i=1}^{N}(1-c_i)c_i^3\left(\operatorname{diag}((y_i-\tau)\mu_i^\top) + \operatorname{off}(\mu_i(y_i-\tau)^\top)\right),\\ \nabla_2 f(R,\tau) =& \sum_{i=1}^{N}\left(2(1-c_i)c_i^3\phi(R)^\top\mu_i + (1-c_i)^2(y_i-\tau)\right).\end{aligned} \tag{44}$$

□

The proof is lengthy but straightforward and is therefore omitted.

The algorithm was implemented using a direction update according to the formula by Polak and Ribière with restart after $n_0 := \dim E = \frac{n(n+1)}{2} + n$ steps, cf. [12]. The algorithm has the n_0-*step quadratic termination property.* That is, being a CG-method in a space diffeomorphic to a Euclidean space, it could be applied equally well to the strictly convex quadratic function $\tilde{f}(x) = x^\top Cx$ for $C \in \mathcal{P}_{n_0}$ and therefore would terminate after at most n_0 steps at the minimum of $\tilde{f}$. Consequently, under the assumption that the unique minimum of our function f is nondegenerated, the implemented CG-method is an n_0-step locally quadratic convergent algorithm, cf. [16].

In Figure 2, eight data points $y_1, \ldots, y_8$ have been generated in the following way. First, an ellipsoid $\mathcal{E}_{Q_0,\tau_0}$ has been specified and eight randomly chosen points have been normalized to $\hat{y}_1, \ldots, \hat{y}_8$, such that $\hat{y}_1, \ldots, \hat{y}_8 \in \mathcal{E}_{Q_0,\tau_0}$. Then noise has been added to obtain $y_i = \hat{y}_i + \Delta\hat{y}_i$. The figure compares the minimum of our cost function with the result of an algebraic fit (dotted line) of the y_i's. Due to Proposition 1 the algebraic fit might have a long tail.

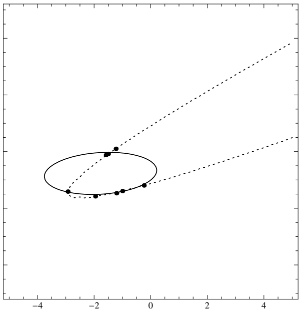


Fig. 2. Algebraic fitting (dotted line) vs. the method proposed here.

Acknowledgments

This work has been supported in parts by *CoTeSys* - Cognition for Technical Systems, Cluster of Excellence, funded by Deutsche Forschungsgemeinschaft.

References

1. F. L. Bookstein. Fitting conic sections to scattered data. *Computer Graphics and Image Processing*, 9:56–71, 1981.
2. P. Ciarlini, M. Cox, F. Pavese, and D. Richter, editors. *Advanced mathematical tools in metrology II. Proceedings of the 2nd international workshop, Oxford, UK, September 27–30, 1995.* Singapore: World Scientific, 1996.
3. F. Deutsch. *Best approximation in inner product spaces.* NY Springer, 2001.
4. A. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct least square fitting of ellipses. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(5):476–480, 1999.
5. A. Forbes. Generalised regression problems in metrology. *Numer. Algorithms*, 5(1-4):523–533, 1993.
6. W. Gander, G. Golub, and R. Strebel. Least-squares fitting of circles and ellipses. *BIT*, 34(4):558–578, 1994.
7. G. Golub and C. F. Van Loan. *Matrix computations. 3rd ed.* Baltimore: The Johns Hopkins Univ. Press, 1996.
8. K. Kanatani. *Statistical optimization for geometric computation: theory and practice.* Amsterdam: North-Holland, 1996.
9. K. Kanatani. Statistical optimization for geometric fitting: theoretical accuracy bound and high order error analysis. *Int. J. of Comp. Vision*, 80:167–188, 2008.
10. I. Markovsky, A. Kukush, and S. Van Huffel. Consistent least squares fitting of ellipsoids. *Numer. Math.*, 98(1):177–194, 2004.
11. M.Baeg, H. Hashimoto, F. Harashima, and J. Moore. Pose estimation of quadratic surface using surface fitting technique. In *Proc. of the International Conference on Intelligent Robots and Systems*, vol. 3, pages 204–209, 1995.
12. J. Nocedal and S. J. Wright. *Numerical optimization.* New York: Springer, 1999.
13. E. Rimon and S. P. Boyd. Obstacle collision detection using best ellipsoid fit. *J. Intell. Robotics Syst.*, 18(2):105–126, 1997.
14. P. D. Sampson. Fitting conic sections to 'very scattered' data: An iterative refinement of the Bookstein algorithm. *Computer Graphics and Image Processing*, 18(1):97–108, 1982.
15. S. Shklyar, A. Kukush, I. Markovsky, and S. van Huffel. On the conic section fitting problem. *J. Multivariate Anal.*, 98(3):588–624, 2007.
16. J. Stoer. On the relation between quadratic termination and convergence properties of minimization algorithms. *Numer. Math.*, 28:343–366, 1977.
17. R. Strebel, D. Sourlier, and W. Gander. A comparison of orthogonal least squares fitting in coordinate metrology. Van Huffel, Sabine (ed.), Recent advances in total least squares techniques and errors-in-variables modeling. Philadelphia: SIAM. 249-258, 1997.
18. P. Thomas, R. Binzel, M. Gaffey, B. Zellner, A. Storrs, and E. Wells. Vesta: Spin pole, size, and shape from Hubble Space Telescope images. *Icarus*, 128, 1997.
19. J. Varah. Least squares data fitting with implicit functions. *BIT*, 36(4):842–854, 1996.

Continuous Reformulation of MINLP Problems

Korbinian Kraemer and Wolfgang Marquardt

Aachener Verfahrenstechnik - Process Systems Engineering, RWTH Aachen University, Templergraben 55, 52056 Aachen, Germany
korbinian.kraemer;wolfgang.marquardt@avt.rwth-aachen.de

Summary. The solution of mixed-integer nonlinear programming (MINLP) problems often suffers from a lack of robustness, reliability, and efficiency due to the combined computational challenges of the discrete nature of the decision variables and the nonlinearity or even nonconvexity of the equations. By means of a continuous reformulation, the discrete decision variables can be replaced by continuous decision variables and the MINLP can then be solved by reliable NLP solvers. In this work, we reformulate 98 representative test problems of the MINLP library MINLPLib with the help of Fischer-Burmeister (FB) NCP-functions and solve the reformulated problems in a series of NLP steps while a relaxation parameter is reduced. The solution properties are compared to the MINLP solution with branch & bound and outer approximation solvers. Since a large portion of the reformulated problems yield local optima of poor quality or cannot even be solved to a discrete solution, we propose a reinitialization and a post-processing procedure. Extended with these procedures, the reformulation achieved a comparable performance to the MINLP solvers SBB and DICOPT for the 98 test problems. Finally, we present a large-scale example from synthesis of distillation systems which we were able to solve more efficiently by continuous reformulation compared to MINLP solvers.

1 Introduction

Optimization problems in engineering are often of discrete-continuous nature and usually nonlinear or even nonconvex. In the field of chemical engineering for example, typical examples include the synthesis of reactor or heat exchanger networks, and unit or flowsheet structure optimization. The discrete variables in these examples usually stem from the structural decisions whereas typical continuous variables are compositions or energies, etc.. In addition, thermodynamics, reaction kinetics and economic objective functions add strong nonlinearities. Due to the combined computational challenges from both the discrete nature and the nonlinearity, these problems are particularly hard to solve. Specifically, the solution performance often suffers from the lack of robust solution algorithms, the necessity of a proper initialization with good

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_8, © Springer-Verlag Berlin Heidelberg 2010

starting points and long computational times. In the light of these challenges it is comprehensible that only few applications of large-scale discrete-continuous nonlinear optimization have been realized in industry.

Discrete-continuous nonlinear optimization problems are usually formulated as MINLP problems. Lastusilta et al. [1] give a comparison of the performances of different MINLP solvers, including recent developments such as CoinBonmin [2]. In recent years, global MINLP solvers for nonconvex problems have been developed and successfully applied to problems of small to medium scale. The high computational effort however still prohibits the use of these solvers for large-scale problems. Local optimization algorithms for MINLP problems are usually based on decomposition methods or tree-search algorithms. Decomposition methods, e.g. outer approximation [3], rely on an iteration between overestimating nonlinear programming (NLP) subproblems and underestimating mixed-integer linear programming (MILP) subproblems. Tree search algorithms like branch & bound [4] perform a search in the space of the NLP subproblems with intelligent node selection and elimination. While these local MINLP solvers have been applied to large-scale problems, the solution robustness, reliability, and efficiency still remain issues.

In recent years, discrete-continuous nonlinear optimization problems have also been reformulated as purely continuous optimization problems. The resulting nonconvex NLP problems can then locally be solved with NLP solvers. Continuous reformulation was first successfully applied to optimization problems in the form of mathematical programs with equilibrium constraints (MPEC) [5]. Here, the equilibrium conditions in the MPEC problems are replaced by nonconvex continuous formulations enforcing the discrete decisions. More recently, general MINLP problems have also been reformulated as purely continuous problems by replacing the discrete variable set with continuous variables [6, 7]. Comparable to MPECs, the discrete decisions are then reached by adding special nonconvex constraints.

2 Continuous Reformulation

Certain discrete-continuous problems can be formulated as MPEC problems where discrete decisions are represented by equilibrium conditions. The equilibrium condition implies that either a constraint is enforced or a decision variable is at its bounds. MPEC problems are often reformulated as NLP problems and solved by NLP solvers. One way to reformulate the equilibrium constraint (EC) is to introduce a penalty function in the objective which penalizes non-discrete solutions. The EC can also be modeled by complementarity constraints in the form of binary multiplications. Various authors suggest to use NCP-functions for the formulation of the EC [5]. However, all these reformulation strategies share one drawback: They violate the linear independence constraint qualification (LICQ) and the Mangasarian-Fromovitz constraint qualification (MFCQ) [8]. It was therefore proposed to relax the reformulations by adding a relaxation parameter μ to the EC. The problem is then solved in a series of successive NLPs as the relaxation parameter μ is reduced to zero. Stein et al. [6] transferred the continuous reformulation approach to MINLP problems, which were derived from general disjunctive programs via big-M constraints. The Fischer-Burmeister (FB) NCP-function was employed to enforce the discrete decisions. Later, Kraemer et al. [7] proposed an extension of the continuous reformulation approach to include general formulations of MINLP problems with binary variables, which are given by

$$\min_{x,y} \quad f(\mathbf{x},\mathbf{y}), \qquad \text{s.t.} \quad g(\mathbf{x},\mathbf{y}) \leq 0, \qquad \mathbf{x} \in \Re^n,\ \mathbf{y} \in \{0,1\}^m. \tag{1}$$

For the continuous reformulation, the binary variables $y \in [0,1]$ were relaxed. FB NCP-functions were used to force the relaxed binary variables to take on binary values:

$$1 - \sqrt{{y_i}^2 + (1-y_i)^2} \leq \mu, \qquad i \in [1,m]. \tag{2}$$

Note that the FB NCP-function was relaxed by the relaxation parameter μ which was reduced to zero in a series of successive NLPs. A discrete solution is returned by the last NLP where $\mu = 0$.

3 Results for MINLP Library

The continuous reformulation of MPECs and solution as NLPs has been applied to large MPEC problem libraries with good results [5, 9]. However, continuous reformulation strategies have not yet been applied to large MINLP problem libraries. Hence, it is the objective of this work to study the performance of continuous reformulation of MINLP problems empirically by means of a large MINLP test problem library.

For this study, the MINLPLib [10] library was chosen. The test problems in MINLPLib are supplied in GAMS [11] syntax by a large number of authors. At the time of the study, MINLPLib contained 271 test problems. Some problems occur in many similar versions which often only differ in a few parameters, variables or equations and have very similar solution properties. Obviously, the problems with many similar versions would have a disproportionate weight in the empirical study. In order to prevent such a distortion, the library was reduced to 98 representative MINLP problems by eliminating similar versions of a problem a priori, i.e. before the performance was checked.

The 98 MINLP problems of the reduced library were automatically reformulated with the help of FB NCP-functions as in equation (2). The FB NCP-functions are relaxed with the relaxation parameter μ and solved in a series of successive NLPs with μ reduced in nine steps from 1 to 0.3, 0.25, 0.2, 0.15, 0.1, 0.05, 0.025 and finally to $\mu = 0$. The solution properties of the reformulated problems, i.e. the value of objective and the solution time, are compared to the solution properties of the MINLP solution with the branch & bound solver SBB [12] and the outer approximation solver DICOPT [13], respectively. All optimization problems were solved in GAMS 22.7 [11] on a PC with a 3 GHz Dual-Core CPU (GAMS runs on one processor only). The NLP problems or subproblems in sections 3 and 4 were solved with the SQP-based solver SNOPT [14].

The continuous decision variables, which replace the binary variables in the reformulated problems, are initialized with a value of 0.5. In a few instances, the original MINLP program contains initial values for the binary variables. In these cases, the given initial values are carried over to the reformulated problems. It is however important to note that we did not assign any ”good” initial values to the decision variables other than those given in the original problem. The comparison of the solution quality, i.e. the value of the objective, for the 98 test problems is shown in the upper part of Fig. 1. More than half of the test problems yielded better solutions when solved with the classical MINLP solvers SBB or DICOPT. The poor performance of the continuous reformulation regarding the solution quality can in part be attributed to the high rate of infeasibility: 61% of the reformulated problems could not be solved to a discrete solution. The percentage of infeasible or non-converging problems is significantly lower for the MINLP solvers SBB (21%) and DICOPT (27%).

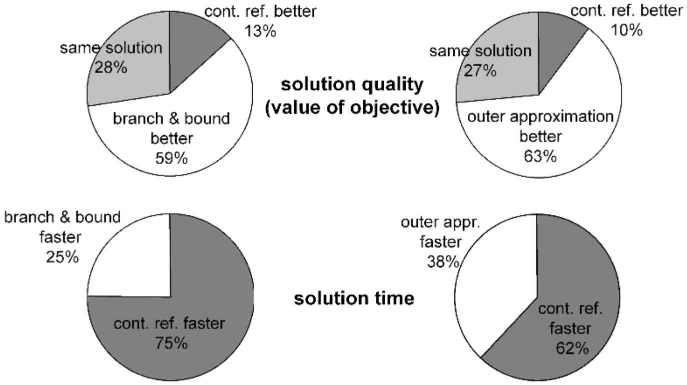


Fig. 1. Performance of continuous reformulation versus branch & bound solver SBB (left) and outer approximation solver DICOPT (right).

The solution times are compared in the lower part of Fig. 1. Note that here we only compare problems for which both compared solvers yield feasible solutions and at least one solution takes longer than 20 seconds (large-scale problems). The solution procedure for the reformulated problems requires the solution of only 9 NLPs regardless of the complexity of the original MINLP. It is therefore not surprising that most large-scale or complex problems converge faster when reformulated compared to the MINLP solution, where a large number of costly NLP subproblems have to be solved.

4 Extension of Continuous Reformulation

Reinitialization Procedure It was shown in the previous section that 61% of the reformulated problems turn infeasible when solved as described in Section 3. In most cases, the completely relaxed first NLP problem ($\mu = 1$) in the series of successive NLP problems can be solved but the solution becomes infeasible when the relaxation parameter μ is tightened in the subsequent NLPs. An illustration of this property is shown in the upper right of Fig. 2. Here, we demonstrate the solution procedure for one relaxed binary variable y_i. We assume that there is a bound $y_i < 0.8$ on the variable implied by the inequality constraints. ${y_i}^{opt} = 0.68$ is the value of the relaxed decision variable at the solution of the NLP. When the relaxation parameter μ is reduced in the successive solution steps, the feasible region for the relaxed decision variable y_i is split in two disjunct regions. As a consequence, ${y_i}^{opt}$ is pushed to the ”right” towards $y_i = 1$ in our example. When the bound imposed by the FB NCP-function and the bound $y_i < 0.8$ overlap for small values of μ, the feasible region on the right side vanishes. Very often, the NLP solver then does not move $y_i = 1$ to the feasible region at the left side but returns an infeasible solution. We therefore propose to reinitialize the decision variables,

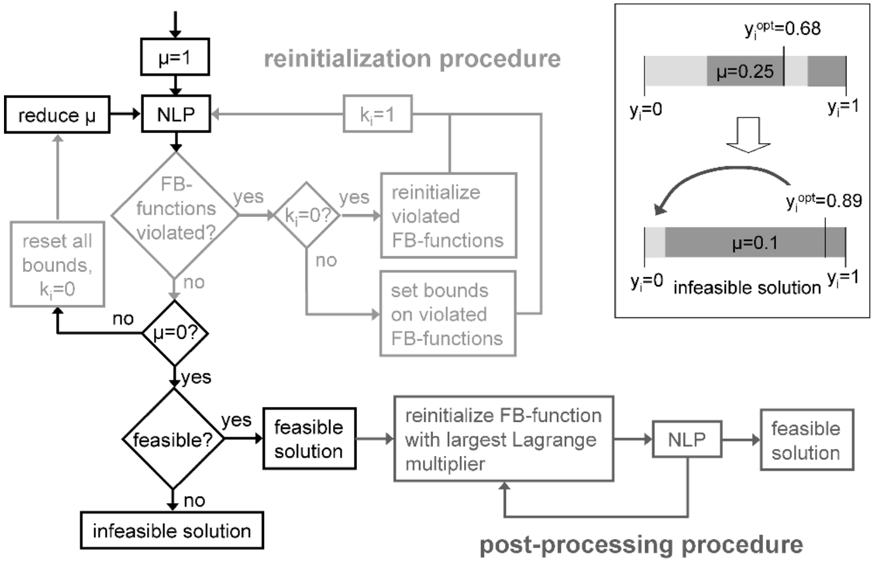


Fig. 2. Solution procedure with reinitialization and post-processing procedures.

which cause the infeasibility, in the feasible region at the opposite side of their domain. In our example, y_i would be reinitialized with $y_i = 0$.

We implemented this reinitialization strategy in the solution procedure as shown in Fig. 2. After each NLP, it is automatically checked whether any FB NCP-functions are violated. When this is the case, the violated FB NCP-functions are reinitialized by initializing the corresponding relaxed decision variables at the opposite side of their domain (i.e. 0 or 1) as described above. Then the NLP is solved again and when feasible, μ is reduced and the solution procedure is resumed. However, when the same FB NCP-functions are still violated and the reinitialized variables are still at the same side of their domain, these decision variables are forced to the opposite side of their domain by setting bounds on the variables. In our example, y_i would be bounded by $y_i \leq 0$. When the following NLP can be solved, all bounds are released again, μ is reduced, and the solution procedure is resumed. The number of reinitialized problems, which may be solved for each value of the relaxation parameter μ, is limited by an upper bound of m, i.e. the number of binary variables y_i.

Post-Processing Procedure In order to improve the solution quality (local optima) of the reformulated problems, a post-processing procedure was implemented as shown in Fig. 2. The post-processing procedure is started when $\mu = 0$ is reached. Then additional NLPs are solved, where single binary variables are fixed at the binary value which is complimentary to the value in the preceding NLP. In other words, the binary variable is fixed at 1 when it was 0 in the optimal solution of the preceding NLP and vice versa. The decision, which binary variable to fix in each post-processing NLP depends on the Lagrange multipliers of the preceding NLP: The binary variable bounded by the FB NCP-function with the largest associated Lagrange multiplier is chosen. The procedure is stopped when a maximum number of 10 NLPs in the post-processing is reached. Together with the reinitialization procedure for m binary variables and nine decreasing values of the relaxation parameter mu, a maximum number of $9 \cdot m + 10$ NLP subproblems need to be solved. Thus, the maximum number of NLP subproblems is identical to the number of combinations of binary variables for six binary variables and less for more than six binary variables. It is important to note that the maximum number of NLP subproblems was hardly ever reached in the solution of the test problems.

Results When extended by the reinitialization and post-processing procedures, only 17% of the 98 test problems could not be solved to a discrete solution. This is a significant reduction from the reformulation without the reinitialization and post-processing procedures (61%). In fact, the number is even lower than the number of problems which could not be solved by the MINLP solvers SBB (21%) and DICOPT (27%). The comparison of the solution quality for the 98 test problems is shown in the upper part of Fig. 3. With the help of the reinitialization and post-processing procedures, the continuous reformulation closed the gap to the classical MINLP solvers: The

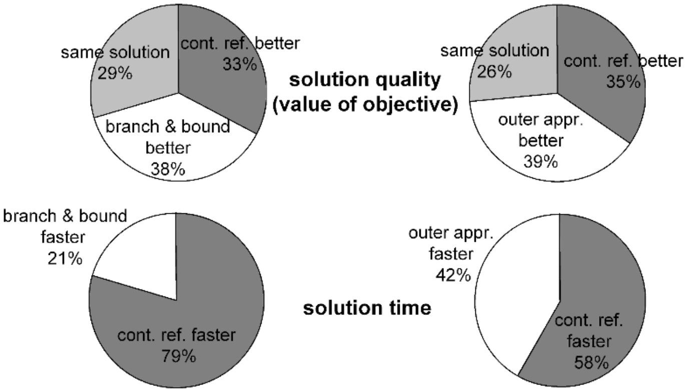


Fig. 3. Performance of extended continuous reformulation versus branch & bound solver SBB (left) and outer approximation solver DICOPT (right).

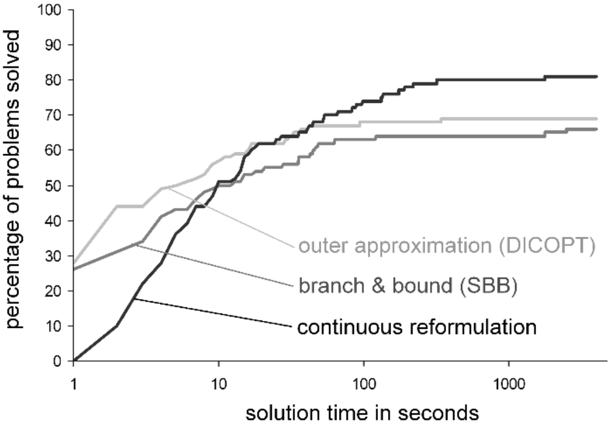


Fig. 4. Comparison of solver performances.

reformulation yielded better solutions almost as often as the MINLP solvers SBB and DICOPT.

Note that the post-processing procedure improved the solution of 54% of the reformulated test problems. However, the additional NLPs of the post-processing and reinitialization procedures extended the solution times for the reformulated problems. It becomes apparent in Fig. 4 that small-scale problems with few binary variables tend to demand longer solution times when they are reformulated. This is because a disproportionally large number of NLPs has to be solved within the reinitialization and post-processing procedures. It needs to be noted, however, that contrary to the subproblems in the fully implemented MINLP solvers, the reformulated problems are solved as consecutive separate NLP problems. As a consequence, GAMS performs a time consuming pre-solve step for each NLP which adds to the solution time especially for the small-scale problems.

Large-scale problems on the other hand, where the classical MINLP solvers need to solve a large number of NLP subproblems, often converge faster when they are reformulated. The solution times of the large-scale problems are compared in the lower part Fig. 3. Note that we only consider problems for which the solution took longer than 20 seconds with at least one of the compared solvers. We also exclude problems which solutions are infeasible by one or more solvers. Compared to the simple continuous reformulation in Section 3 the solution time advantage over the solver DICOPT has decreased slightly but is still noticeable. Obviously, there is a trade-off between robustness and reliability (quality of the solution) of the reformulation on the one and efficiency on the other hand. The extension with the reinitialization and post-processing procedures has shifted the balance slightly towards robustness and reliability.

It is certainly an important question, for which discrete continuous optimization problems the continuous reformulation performs better than the existing local MINLP solvers or vice versa. We cannot give definite answers to this question as this is still a topic of research. As indicated above, the reformulation offers the prospect of shorter solution times mostly for large-scale problems. Of course, these are in fact the problems were computational efficiency matters most. Regarding the robustness and reliability of the solution, the continuous reformulation tends to perform better for problems with low combinatorial complexity, i.e. problems which are not highly disjunct but where the local optima are located close together in the solution space. For these problems, the tightening of the NCP-functions works more reliably.

5 Large-Scale Example from Process Synthesis

In the last section, we present an example of a MINLP problem from process synthesis which fulfills all of the criteria for a superior performance of the reformulation as mentioned above: It is a problem of large scale with a large amount of variables and equations but it also displays a low combinatorial complexity because the local optima are located close together. The example is the rigorous economic optimization of a distillation process. Specifically, an equimolar feed of acetone, chloroform, benzene and toluene is to be separated into its pure components in a multicolumn process shown in Fig. 5. The process requires a recycle since the mixture exhibits an azeotrope between acetone

Table 1. Objective values and solution times.

	continuous reformulation	MINLP branch & bound	MINLP outer approximation
annualized cost (objective value)	417612 € /a	417768 € /a	no convergence
solution time optimization	146 s	1202 s	-

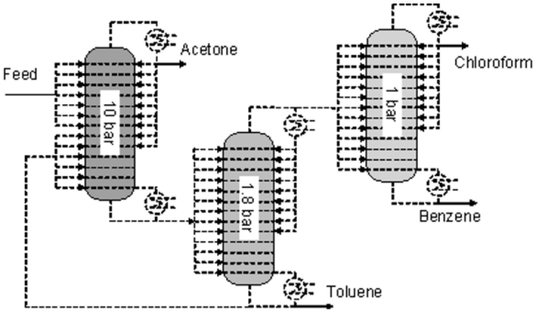


Fig. 5. Superstructure for the rigorous optimization of the multicolumn process.

and chloroform. We refer to [15] for a more detailed explanation of the separation process. The rigorous economic optimization provides information about the optimal number of column trays, the optimal feed tray locations, and the optimal operating point by minimizing a total cost function comprising capital and operating costs. The number of trays and the feed tray locations are discrete variables. Considering the large scale of the three-column process and the nonlinearity of the nonideal thermodynamics, it is obvious that this problem is particularly hard to solve.

Specifically, the optimization problem contains 3293 continuous variables, 376 binary variables, and 3305 mostly nonlinear equations. A detailed description of the column model formulation as well as the initialization can be found elsewhere [15]. The MINLP problem was reformulated as a continuous problem by introducing 376 relaxed FB NCP-functions as described in Section 3. Like many unit operations in chemical engineering, distillation is carried out in a cascade structure, i.e. a column. Here, the local optima are close together: The favorable trays for feeds or product draws are located side by side. In addition, the problem allows for a tight relaxation, i.e. the relaxed solution places the feed trays and product draws on neighboring trays. As a consequence of these favorable properties, the relaxation parameter could be reduced in only three steps to ($\mu = 1, 0.2, 0$). Since the NLPs in the stepwise solution procedure could all be solved, a reinitialization procedure was not necessary. The rigorous optimization of the large-scale example problem could be solved with excellent robustness, efficiency and reliability due to the continuous reformulation of the MINLP problem (and a suitable initialization procedure). Table 1 lists the respective objective values and solution times for a comparison of the optimization properties of the continuous reformulation versus the MINLP solvers SBB and DICOPT. The reformulated problem yielded a slightly better local optimum than the MINLP problem solved with SBB. The DICOPT solver did not converge for this problem. The solution time of the reformulated problem was significantly lower than the solution time of the MINLP problem, which also benefited from the same initialization.

6 Conclusion

In this work, 98 representative MINLP test problems of the library MINLPLib were reformulated as continuous problems with the help of FB NCP-functions. When solved in successive NLP steps with a gradually tightened relaxation parameter, the reformulated problems yielded considerably shorter solution times compared to the classical MINLP solvers SBB and DICOPT. As a drawback however, 61% of the reformulated problems could not be solved to a discrete solution. We therefore proposed an extension of the continuous reformulation by a reinitialization and a post-processing procedure. With this extension, the reformulation achieved a comparable performance to the MINLP solvers SBB and DICOPT for the 98 test problems: The reformulation identified better local optima for about the same percentage of problems as the MINLP solvers. Small-scale problems tend to be solved faster by the MINLP solvers whereas large-scale problems are often solved faster by the extended continuous reformulation. Apparently, it is very problem-specific which solver performs best. Finally, we presented an example from chemical process synthesis, which is of large scale but displays low combinatorial complexity. The continuous reformulation performs better than the MINLP solvers for this example. Obviously, it would be of great value to be able to predict a priori, whether a discrete-continuous optimization problem qualifies for continuous reformulation. Further research should therefore be directed towards a more detailed characterization of the problems which are suited for reformulation.

References

1. Lastusilta T, Bussieck M R, Westerlund T (2009) Ind Eng Chem Res 48:7337
2. Bonami P, Biegler L T, Conn A R, Cornuejols G, Grossmann I E, Laird C D, Lee J, Lodi A, Margot F, Sawaya N, Waechter A (2005) IBM Res Rep RC23771
3. Viswanathan J, Grossmann, I E (1990) Comput Chem Eng 14:769–782
4. Gupta O K, Ravindran V (1985) Manage Sci 31:1533–1546
5. Fletcher R, Leyffer S (2004) Optim Method Softw 19:15–40
6. Stein O, Oldenburg J, Marquardt W (2004) Comput Chem Eng 28:1951–1966
7. Kraemer K, Kossack S, Marquardt W (2007) In: Plesu V, Agachi P S (eds) 17th European Symposium on Computer Aided Process Engineering. Elsevier.
8. Scheel H, Scholtes S (2000) Math Oper Res 25:1–22
9. Baumrucker BT, Renfro JG, Biegler LT (2007) Comp Chem Eng 32:2903–2913
10. Bussieck M R, Drud A S, Meeraus A (2003) INFORMS J. Comput 15:114–119
11. Brooke A, Kendrick D, Meeraus A, Raman R (2005) GAMS - A Users Guide. GAMS Development Corporation, Washington
12. Drud A, (2005) SBB. In: GAMS - The Solver Manuals. GAMS Development Corporation, Washington
13. Grossman I E, Viswanathan J, Vecchietti A, Raman R, Kalvelagen E (2008) DICOPT. In: GAMS - The Solver Manuals. GAMS Corporation, Washington
14. Gill P, Murray W, Saunders M, Drud A, Kalvelagen E (2008) SNOPT. In: GAMS - The Solver Manuals. GAMS Development Corporation, Washington
15. Kraemer K, Kossack S, Marquardt W (2009) Ind Eng Chem Res 48:6749–6764

Local Convergence of Sequential Convex Programming for Nonconvex Optimization

Quoc Tran Dinh* and Moritz Diehl*

*Department of Electrical Engineering (SCD-ESAT) and OPTEC, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, 3001-Heverlee, Belgium
{quoc.trandinh, moritz.diehl}@esat.kuleuven.be

Summary. This paper introduces sequential convex programming (SCP), a local optimzation method for solving nonconvex optimization problems. A full-step SCP algorithm is presented. Under mild conditions the local convergence of the algorithm is proved as a main result of this paper. An application to optimal control illustrates the performance of the proposed algorithm.

1 Introduction and Problem Statement

Consider the following nonconvex optimization problem:

$$\begin{cases} \min\limits_{x} c^T x \\ \text{s.t. } g(x) = 0, \ x \in \Omega, \end{cases} \tag{P}$$

where $c \in \mathbf{R}^n$, $g : \mathbf{R}^n \to \mathbf{R}^m$ is non-linear and smooth on its domain, and Ω is a nonempty closed convex subset in $\mathbf{R}^n$.

This paper introduces *sequential convex programming* (SCP), a local optimization method for solving the nonconvex problem (P). We prove that under acceptable assumptions the SCP method locally converges to a KKT point[1] of (P) and the rate of convergence is linear.

Problems in the form of (P) conveniently formulate many problems of interest such as least squares problems, quadratically constrained quadratic programming, nonlinear semidefinite programming (SDP), and nonlinear second order cone programming problems (see, e.g., [1, 2, 5, 6, 10]). In nonlinear optimal control, by using direct transcription methods, the resulting problem is usually formulated as an optimization problem of the form (P) where the equality constraint $g(x) = 0$ originates from the dynamic system of an optimal control problem.

The main difficulty of the problem (P) is concentrated in the nonlinear constraint $g(x) = 0$ that can be overcome by linearizing it around the current

[1] KKT stands for "**K**arush-**K**uhn-**T**ucker".

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_9, © Springer-Verlag Berlin Heidelberg 2010

iteration point and maintaining the remaining convexity of the original problem. This approach differs from sequential quadratic programming, Gauss-Newton or interior point methods as it keeps even nonlinear constraints in the subproblems as long as they are convex.

Optimization algorithms using convex approximation approaches have been proposed and investigated by Fares *et al.* [4] for nonlinear SDP and Jarre [8] for nonlinear programming. Recently, Lewis and Wright [12] introduced a proximal point method for minimizing the composition of a general convex function h and a smooth function c using the convex approximation of $h(c(\cdot))$.

1.1. Contribution. In this paper, we first propose a *full-step SCP algorithm* for solving (P). Then we prove the local convergence of this method. The main contribution of this paper is Theorem 1, which estimates the local contraction and shows that the *full-step SCP algorithm* converges linearly to a KKT point of the problem (P). An application in optimal control is implemented in the last section.

1.2. Problem Statement. Throughout this paper, we assume that g is twice continuously differentiable on its domain. As usual, we define the Lagrange function of (P) by $L(x,\lambda) := c^T x + \lambda^T g(x)$ and the KKT condition associated with (P) becomes

$$\begin{cases} 0 \in c + \nabla g(x)\lambda + N_{\Omega}(x), \\ 0 = g(x), \end{cases} \tag{1}$$

where $\nabla g(x)$ denotes the Jacobian matrix of g at x. The multivalued mapping

$$N_{\Omega}(x) := \begin{cases} \{w \in \mathbf{R}^n \mid w^T(y-x) \leq 0,\ y \in \Omega\} & \text{if } x \in \Omega, \\ \emptyset & \text{otherwise} \end{cases} \tag{2}$$

is the normal cone of the convex set Ω at x. A pair $z^* := (x^*, \lambda^*)$ satisfying (1) is called a KKT point and x^* is called a stationary point of (P). We denote by Γ^* and S^* the sets of the KKT and the stationary points of (P), respectively. Note that the first line of (1) includes implicitly the condition $x \in \Omega$ due to definition (2). Let us define $K := \Omega \times \mathbf{R}^m$ and introduce a new mapping φ as follows

$$\varphi(z) := \begin{pmatrix} c + \nabla g(x)\lambda \\ g(x) \end{pmatrix}, \tag{3}$$

where z stands for (x,λ) in $\mathbf{R}^{n+m}$. Then the KKT condition (1) can be regarded as a generalized equation:

$$0 \in \varphi(z) + N_K(z), \tag{4}$$

where $N_K(z)$ is the normal cone of K at z.

The generalized equation (4) can be considered as a basic tool for studying variational inequalities, complementarity problems, fixed point problems and mathematical programs with equilibrium constraints. In the landmark paper

[13], Robinson introduced a condition for generalized equation (4), which is called *strong regularity.* This assumption is then used to investigate the solution of (4) under the influence of perturbations. *Strong regularity* becomes a standard condition in variational analysis as well as in optimization. It is important to note that (see [3]) the generalized equation (4) is strongly regular at $z^* \in \Gamma^*$ *if and only if* the strong second order sufficient condition (SOSC) of (P) holds at this point whenever Ω is polyhedral and the LICQ condition[2] is satisfied. Many research papers which have studied the stability and sensitivity in parametric optimization and optimal control also used the strong regularity property (see, e.g., [11, 14]).

1.4. Sequential Convex Programming Framework. The *full-step sequential convex programming algorithm* for solving (P) is an iterative method that generates a sequence $\{z^k\}_{k\geq 0}$ as follows:

1. Choose an initial point x^0 inside the convex set Ω and λ^0 in $\mathbf{R}^m$. Set $k := 0$.
2. For a given x^k, solve the following convex subproblem:

$$\begin{cases} \min\limits_{x} c^T x \\ \text{s.t. } g(x^k) + \nabla g(x^k)^T(x - x^k) = 0, \\ \qquad x \in \Omega, \end{cases} \tag{$\mathrm{P_{cvx}}(x^k)$}$$

   to obtain a solution $x_+(x^k)$ and the corresponding Lagrange multiplier $\lambda_+(x^k)$. Set $z_+(x^k) := (x_+(x^k), \lambda_+(x^k))$. If $\|z_+(x^k) - z^k\| \leq \varepsilon$ for a given tolerance $\varepsilon > 0$, then stop. Otherwise, set $z^{k+1} := z_+(x^k)$, increase k by 1 and go back to Step 2.

As we will show later, the iterative sequence $\{z^k\}$ generated by the *full-step SCP algorithm* converges to a KKT point z^* of the original problem (P), if it starts sufficiently close to z^* and the contraction property is satisfied (see Theorem 1 below).

In practice, this method should be combined with globalization strategies such as line search or trust region methods in order to ensure global convergence, if the starting point is arbitrary. Since Ω is convex, projection methods can be used to find an initial point x^0 in Ω.

Lemma 1. *If x^k is a stationary point of $P_{cvx}(x^k)$ then it is a stationary point of the problem (P).*

Proof. We note that x^k always belongs to Ω. Substituting x^k into the KKT condition of the subproblem $\mathrm{P_{cvx}}(x^k)$, it collapses to (1).

2 Local convergence of SCP methods

Suppose that $x^k \in \Omega$, $k \geq 0$, is the current iteration associated with $\lambda^k \in \mathbf{R}^m$. Then the KKT condition of the convex subproblem $\mathrm{P_{cvx}}(x^k)$ becomes

[2] LICQ stands for "**L**inear **I**ndependence **C**onstraint **Q**ualification".

$$\begin{cases} 0 \in c + \nabla g(x^k)\lambda + N_{\Omega}(x), \\ 0 = g(x^k) + \nabla g(x^k)^T(x - x^k), \end{cases} \tag{5}$$

where λ is the corresponding multiplier. Suppose that the Slater constraint qualification condition holds for $\mathrm{P}_{\mathrm{cvx}}(x^k)$, i.e.,

$$\mathrm{relint}\ \Omega \cap \{x \mid g(x^k) + \nabla g(x^k)^T(x - x^k) = 0\} \neq \emptyset,$$

where relintΩ is the set of the relative interior points of Ω. In other words, there exists a strictly feasible point of $\mathrm{P}_{\mathrm{cvx}}(x^k)$. Then by convexity of Ω, a point $(x_+(x^k), \lambda_+(x^k))$ is a KKT point of $\mathrm{P}_{\mathrm{cvx}}(x^k)$ if and only if $x_+(x^k)$ is a solution of (5) corresponding to the multiplier $\lambda_+(x^k)$. In the sequel, we use z for a pair (x, λ), z^* and $z_+(x^k)$ are a KKT point of (P) and $\mathrm{P}_{\mathrm{cvx}}(x^k)$, respectively. We denote by

$$\hat{\varphi}(z; x^k) := \begin{pmatrix} c + \nabla g(x^k)\lambda \\ g(x^k) + \nabla g(x^k)^T(x - x^k) \end{pmatrix}, \tag{6}$$

a linear mapping and $K := \Omega \times \mathbf{R}^m$. For each $x^* \in S^*$, we define a multivalued function:

$$L(z; x^*) := \hat{\varphi}(z; x^*) + N_K(z), \tag{7}$$

and $L^{-1}(\delta; x^*) := \{z \in \mathbf{R}^{n+m} : \delta \in L(z; x^*)\}$ for $\delta \in \mathbf{R}^{n+m}$ is its inverse mapping. To prove local convergence of the *full-step SCP algorithm*, we make the following assumptions:

(A1) The set of KKT points Γ^* of (P) is nonempty.

(A2) Let $z^* \in \Gamma^*$. There exists a neighborhood $U \subset \mathbf{R}^{n+m}$ of the origin and Z of z^* such that for each $\delta \in U$, $z^*(\delta) := L^{-1}(\delta; x^*) \cap Z$ is single valued. Moreover, the mapping $z^*(\cdot)$ is Lipschitz continuous on U with a Lipschitz constant $\gamma > 0$, i.e.,

$$\|z^*(\delta) - z^*(\delta')\| \leq \gamma\|\delta - \delta'\|, \quad \forall \delta, \delta' \in U. \tag{8}$$

(A3) There exists a constant $0 < \kappa < 1/\gamma$ such that $\|E_g(z^*)\| \leq \kappa$, where $E_g(z^*)$ is the Hessian of the Lagrange function L with respect to the argument x at $z^* = (x^*, \lambda^*)$ defined by

$$E_g(z) := \sum_{i=1}^{m} \lambda_i \nabla^2 g_i(x). \tag{9}$$

Remark 1. By definition of $\hat{\varphi}(\cdot;\cdot)$, we can refer to x^k as a parameter of this mapping and $\mathrm{P}_{\mathrm{cvx}}(x^k)$ can be considered as a parametric convex problem with respect to the parameter x^k.

i) It is easy to show that z^* is a solution to $0 \in \varphi(z) + N_K(z)$ if and only if it is a solution to $0 \in \hat{\varphi}(z; x^*) + N_K(z)$.

ii) Assumption (**A3**) implies that either the function g should be "weakly

nonlinear" (small second derivatives) in a neighborhood of a stationary point or the corresponding Lagrange multipliers are sufficiently small in the neighborhood of λ^*. The latter case occurs if the optimal objective value of (P) depends only weakly on perturbations of the nonlinear constraint $g(x) = 0$.
iii) Assumption (**A2**) is the strong regularity condition of the parametric generalized equation $0 \in \hat{\varphi}(z; x^k) + N_K(z)$ at (z^*, x^*) in the sense of Robinson [13].

For the assumption (**A2**), by linearity of $\hat{\varphi}$, we have $\hat{\varphi}(z; x^*) = \hat{\varphi}(z^*; x^*) + \nabla\hat{\varphi}(z^*; x^*)^T(z - z^*)$ where matrix $\nabla\hat{\varphi}(z)$ is defined by

$$\nabla\hat{\varphi}(z; x^*) := \begin{bmatrix} 0 & \nabla g(x^*) \\ \nabla g(x^*)^T & 0 \end{bmatrix}, \tag{10}$$

which may be singular even if $\nabla g(x^*)$ is full-rank. It is easy to see that $L(z; x^*)$ defined by (7) has the same form as $\hat{L}(z; x^*) := \hat{\varphi}(z^*, x^*) + \nabla\hat{\varphi}(z^*; x^*)(z - z^*) + N_K(z)$ a linearization of (4) at (z^*, x^*).
To make the strong regularity assumption clear in the sense of mathematical programming, for a given neighborhood U of 0 and Z of z^*, we define the following perturbed convex programming problem:

$$\begin{cases} \min\limits_{x} \ (c + \delta_{\mathrm{c}})^T(x - x^*) \\ \text{s.t.} \ \ g(x^*) + \delta_{\mathrm{g}} + \nabla g(x^*)^T(x - x^*) = 0, \\ \qquad x \in \Omega, \end{cases} \tag{$\mathrm{P_{cvx}}(x^*; \delta)$}$$

where $\delta = (\delta_{\mathrm{c}}, \delta_{\mathrm{g}})$ is a perturbation (or a parameter) vector. The Slater condition associated with $\mathrm{P_{cvx}}(x^*; \delta)$ becomes

$$\text{relint } \Omega \cap \{x \mid g(x^*) + \delta_{\mathrm{g}} + \nabla g(x^*)^T(x - x^*) = 0\} \neq \emptyset. \tag{11}$$

Then the assumption (A2) holds if and only if $z^*(\delta)$ is the unique KKT point of $\mathrm{P_{cvx}}(x^*; \delta)$, and this solution is Lipschitz continuous on U with a Lipschitz constant $\gamma > 0$ provided that (11) holds.

The *full-step SCP algorithm* is called to be well-defined if the convex subproblem $\mathrm{P_{cvx}}(x^k)$ has at least one KKT point $z_+(x^k)$ provided that z^k is sufficiently close to $z^* \in \Gamma^*$. In this case, the subproblem $\mathrm{P_{cvx}}(x^k)$ is said to be solvable.

Lemma 2. *Suppose that Assumptions (***A1***)-(***A3***) are satisfied, then the full-step SCP algorithm is well-defined.*

Proof. It follows from Remark 1 (i) that the parametric generalized equation $0 \in \hat{\varphi}(z; x^k) + N_K(z)$ is strongly regular at (z^*, x^*) according to Assumption (A2), where x^k is referred as a parameter. Applying Theorem 2.1 [13], we conclude that there exists a neighborhood X of x^* such that the generalized equation $0 \in \hat{\varphi}(z; x^k) + N_K(z)$ has unique solution $z_+(x^k)$ for all $x^k \in X$, which means that $z_+(x^k)$ is a KKT point of $\mathrm{P_{cvx}}(x^k)$. □

The main result of this paper is the following theorem.

Theorem 1. *[Local Contraction] Suppose that Assumptions (***A1***)-(***A3***) are satisfied. Suppose further for $z^* \in \Gamma^*$ that g is twice continuously differentiable on a neighborhood of x^*. Then the full-step SCP algorithm is well-defined and there exists $\rho > 0$ such that for all $z^k \in B(z^*, \rho)$ we have:*

$$\|z_+(x^k) - z^*\| \leq \alpha \|z^k - z^*\|, \tag{12}$$

where $\alpha \in (0, 1)$ does not depend on z^k and $z_+(x^k)$. Thus, if the initial point z^0 is sufficiently close to z^* then the sequence $\{z^k\}$ generated by full-step SCP algorithm converges to z^* linearly.

Proof. Note that $\Gamma^* \neq \emptyset$ by (**A1**), take any $z^* \in \Gamma^*$. Then the well-definedness of the *full-step SCP algorithm* follows from Lemma 2. By assumption (**A3**) that $\gamma\kappa < 1$ we can choose $\varepsilon := \frac{(1-\gamma\kappa)}{(4\sqrt{22}+2\sqrt{3})\gamma} > 0$. Since g is twice continuously differentiable on a neighborhood X of x^* and $E(x, \lambda)$ defined by (9) is linear with respect to λ, it implies that, for a given $\varepsilon > 0$ defined as above, there exists a positive number $r_0 > 0$ such that $\|\nabla g(x) - \nabla g(x^k)\| \leq \varepsilon$, $\|\nabla g(x) - \nabla g(x^*)\| \leq \varepsilon$, $\|E_g(z) - E_g(z^*)\| \leq \varepsilon$ and $\|E_g(z) - E_g(z^k)\| \leq \varepsilon$ for all $z = (x, \lambda) \in B(z^*, r_0)$ and $z^k = (x^k, \lambda^k) \in B(z^*, r_0)$, where $B(z^*, r_0)$ is the closed ball of radius r_0 centered at z^*.

Take any $z \in B(z^*, r_0) \subseteq Z$ and define the residual quantity

$$\delta(z; x^*, x^k) := \hat{\varphi}(z; x^*) - \hat{\varphi}(z; x^k). \tag{13}$$

This quantity can be expressed as

$$\begin{aligned}
\delta(z; x^*, x^k) &= [\hat{\varphi}(z; x^*) - \varphi(z^*)] + [\varphi(z^*) - \varphi(z)] \\
&\quad + [\varphi(z) - \varphi(z^k)] + [\varphi(z^k) - \hat{\varphi}(z; x^k)] \\
&= \int_0^1 M(z_t^k; x^k)(z - z^k)dt - \int_0^1 M(z_t^*; x^*)(z - z^*)dt \\
&= \int_0^1 [M(z_t^k; x^k) - M(z_t^*; x^*)](z - z^k)dt \\
&\quad - \int_0^1 M(z_t^*; x^*)(z^k - z^*)dt,
\end{aligned} \tag{14}$$

where $z_t^* := z^* + t(z - z^*)$, $z_t^k := z^k + t(z - z^k)$ with $t \in [0, 1]$, and the matrix M is defined by

$$M(\tilde{z}; \hat{x}) := \begin{bmatrix} E_g(\tilde{z}) & \nabla g(\tilde{x}) - \nabla g(\hat{x}) \\ \nabla g(\tilde{x})^T - \nabla g(\hat{x})^T & 0 \end{bmatrix}. \tag{15}$$

Since $t \in [0, 1]$, the points z_t^k and z_t^* must belong to $B(z^*, r_0)$. Using the following inequalities

$$\|E_g(z_t^k) - E_g(z_t^*)\| \leq \|E_g(z_t^k) - E_g(z^*)\| + \|E_g(z_t^*) - E_g(z^*)\| \leq 2\varepsilon,$$
$$\|\nabla g(x_t^k) - \nabla g(x_t^*)\| \leq \|\nabla g(x_t^k) - \nabla g(x^*)\| + \|\nabla g(x_t^*) - \nabla g(x^*)\| \leq 2\varepsilon,$$
and $\|\nabla g(x^k) - \nabla g(x^*)\| \leq \varepsilon$,

it follows that

$$\begin{aligned}\|M(z_t^k; x^k) - M(z_t^*; x^*)\|^2 &\leq \|E_g(z_t^k) - E_g(z_t^*)\|^2 \\ &\quad + 2[\|\nabla g(x_t^k) - \nabla g(x_t^*)\| + \|\nabla g(x^k) - \nabla g(x^*)\|]^2 \\ &\leq 22\varepsilon^2.\end{aligned}$$

This inequality implies that

$$\|M(z_t^*; x^*) - M(z_t^k; x^k)\| \leq \sqrt{22}\varepsilon. \tag{16}$$

Similarly, using Assumption (**A3**), we can estimate

$$\begin{aligned}\|M(z_t^*; x^*)\|^2 &\leq \|E_g(z_t^*)\|^2 + 2\|\nabla g(x_t^*) - \nabla g(x^*)\|^2 \\ &\leq 2\varepsilon^2 + [\|E_g(z_t^*) - E_g(z^*)\| + \|E_g(z^*)\|]^2 \\ &\leq 2\varepsilon^2 + (\varepsilon + \kappa)^2 \\ &\leq (\kappa + \sqrt{3}\varepsilon)^2.\end{aligned} \tag{17}$$

Combining (14), (16) and (17) together we obtain

$$\|\delta(z, x^*, x^k\| \leq (\kappa + \sqrt{3}\varepsilon)\|z^k - z^*\| + \sqrt{22}\varepsilon\|z - z^k\|. \tag{18}$$

Alternatively, we first shrink $B(z^*, r_0)$, if necessary, such that $\delta(z, x^*; x^k) \in U$ and then apply Assumption (**A2**) to imply that there exists $\tilde{z}(\delta) = (\tilde{x}(\delta), \tilde{\lambda}(\delta)) \in B(z^*, r_0)$ a solution of $\delta \in L(\cdot; z^*)$ for all $\delta \in U$ satisfying

$$\|\tilde{z}(\delta) - z^*\| \leq \gamma\|\delta\|. \tag{19}$$

If we recall $z_+(x^k)$ a KKT point of $\mathrm{P}_{\mathrm{cvx}}(x^k)$, one has $0 \in \hat{\varphi}(z_+(x^k); x^k) + N_K(z_+(x^k))$ which implies $\delta(z_+(x^k); x^*, x^k) \in \hat{\varphi}(z_+(x^k); x^*) + N_K(z_+(x^k))$ by definition of δ. Therefore, it follows from (19) that

$$\|z_+(x^k) - z^*\| \leq \gamma\|\delta(z_+(x^k); x^*, x^k)\|. \tag{20}$$

Substituting z by $z_+(x^k)$ into (18) and then merging with (20) we get

$$\|z_+(x^k) - z^*\| \leq (\gamma\kappa + \sqrt{3}\gamma\varepsilon)\|z^k - z^*\| + \sqrt{22}\gamma\varepsilon\|z_+(x^k) - z^k\|. \tag{21}$$

Using the triangle inequality $\|z_+(x^k) - z^k\| \leq \|z_+(x^k) - z^*\| + \|z^k - z^*\|$ for the right hand side of (21), after a simple rearrangement, the inequality (21) implies

$$\|z_+(x^k) - z^*\| \le \frac{[\gamma\kappa + (\sqrt{22}+\sqrt{3})\gamma\varepsilon]}{1-\sqrt{22}\gamma\varepsilon}\|z^k - z^*\|. \tag{22}$$

Let us denote $\alpha := \frac{[\gamma\kappa+(\sqrt{22}+\sqrt{3})\gamma\varepsilon]}{1-\sqrt{22}\gamma\varepsilon}$. From the choice of ε, it is easy to show that

$$\alpha = \frac{(3\sqrt{22}+\sqrt{3})\gamma\kappa + \sqrt{22}+\sqrt{3}}{3\sqrt{22}+2\sqrt{3}+\sqrt{22}\gamma\kappa} \in (0,1). \tag{23}$$

Thus the inequality (22) is rewritten as

$$\|z_+(x^k) - z^*\| \le \alpha\|z^k - z^*\|, \quad \alpha \in (0,1), \tag{24}$$

which proves (12).

If the starting point $z^0 \in B(z^*, r_0)$ then we have $\|z^1 - z^*\| \le \alpha\|z^0 - z^*\| \le \|z^0 - z^*\|$, which shows that $z^1 \in B(z^*, r_0)$. By induction, we conclude that the whole sequence $\{z^k\}$ is contained in $B(z^*, r_0)$. The remainder of the theorem follows directly from (12).

Remark 2. It is easy to see from (23) that $\alpha \in (\gamma\kappa, 1)$.

3 Numerical Results

In this section, we apply the SCP method to the optimal control problem arising from the optimal maneuvers of a rigid asymmetric spacecraft [7, 9]. The Euler equations for the angular velocity $\omega = (\omega_1, \omega_2, \omega_3)^T$ of the spacecraft are given by

$$\begin{cases} \dot\omega_1 = -\frac{(I_3-I_2)}{I_1}\omega_2\omega_3 + \frac{u_1}{I_1}, \\ \dot\omega_2 = -\frac{(I_1-I_3)}{I_2}\omega_1\omega_3 + \frac{u_2}{I_2}, \\ \dot\omega_3 = -\frac{(I_2-I_1)}{I_3}\omega_1\omega_2 + \frac{u_3}{I_3}, \end{cases} \tag{25}$$

where $u = (u_1, u_2, u_3)^T$ is the control torque; $I_1 = 86.24$ kg.m^2, $I_1 = 85.07$ kg.m^2 and $I_3 = 113.59$ kg.m^2 are the spacecraft principal moments of inertia. The performance index to be minimized is given by (see [7]):

$$J := \frac{1}{2}\int_0^{t_f} \|u(t)\|^2 dt. \tag{26}$$

The initial condition $\omega(0) = (0.01, 0.005, 0.001)^T$, and the terminal constraint is

$$\omega(t_f) = (0,0,0)^T \text{ (Case 1) or } \omega(t_f)^T S_f \omega(t_f) \le \rho_f \text{ (Case 2)}, \tag{27}$$

where matrix S_f is symmetric positive definite and $\rho_f > 0$. Matrix S_f is computed by using the discrete-time Riccati equation of the linearized form of (25) and ρ is taken by $\rho := 10^{-6} \times \lambda_{\max}(S_f)$, where $\lambda_{\max}(S_f)$ is the maximum eigenvalue of S_f. The additional inequality constraint is

$$\omega_1(t) - (5 \times 10^{-6} t^2 - 5 \times 10^{-4} t + 0.016) \leq 0, \tag{28}$$

for all $t \in [0, t_f]$ (see [7]).

In order to apply the SCP algorithm, we use the direct transcription method to transform the optimal control problem into a nonconvex optimization problem. The dynamic system is discretized based on the forward Euler scheme. With the time horizon $t_f = 100$, we implement the SCP algorithm for H_p (the number of the discretization points) from 100 to 500. The size (n, m, l) of the optimization problem goes from $(603, 300, 104)$ to $(3003, 1500, 504)$, where n is the number of variables, m is the number of equality constraints, and l is the number of inequality constraints.

We use an open source software (CVX) to solve the convex subproblems $\mathrm{P}_{\mathrm{cvx}}(x^k)$ and combine it with a line search strategy to ensure global convergence (not covered by this paper's theory). All the computational results are performed in Matlab 7.9.0 (2009) running on a desktop PC Pentium IV (2.6GHz, 512Mb RAM).

If we take the tolerance $\mathtt{TolX} = 10^{-7}$ then the number of iterations goes from 3 to 6 iterations depending on the size of the problem. Note that the resulting convex subproblems in Case 1 are convex quadratic, while, in Case 2, they are quadratically constrained quadratic programming problems.

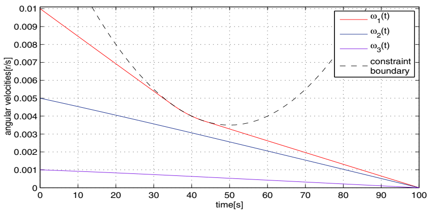


Fig1. Optimal angular velocities [Case 1]

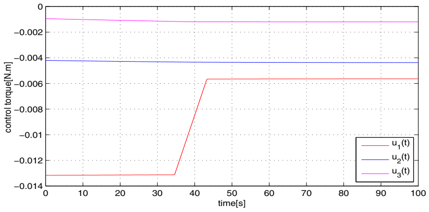


Fig2. Optimal control torques [Case 1]

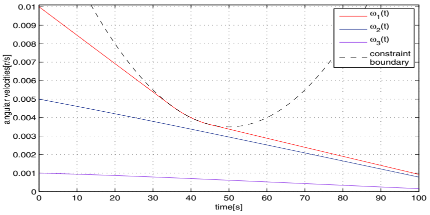


Fig3. Optimal angular velocities [Case 2]

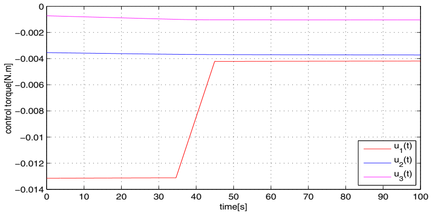


Fig4. Optimal control torques [Case 2]

Figure 1 (resp. Figure 3) shows the optimal angular velocity $\omega(t)$ of the rigid asymmetric spacecraft from 0 to $100s$ for Case 1 (resp. Case 2) with $H_p = 500$. The results show that $\omega_1(t)$ constrained by (28) touches its boundary around the point $t = 39s$ and $\omega(t)$ tends to zero at the end $(t = 100s)$ identical to the results in [7]. Figure 2 (resp. Figure 4) shows the optimal torque $u(t)$ of the rigid asymmetric spacecraft for Case 1 (resp. Case 2). The rate of convergence is illustrated in Figures 5 and 6 for Case 1 and Case 2,

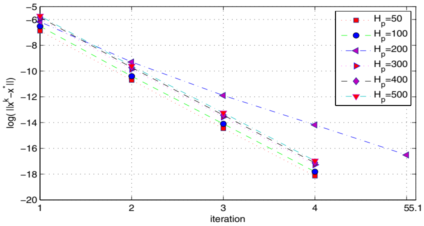


Fig5. Rate of Convergence [Case 1]

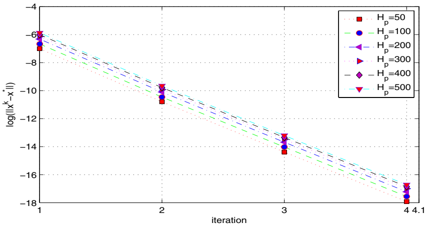


Fig6. Rate of Convergence [Case 2]

respectively. As predicted by the theoretical results in this paper, the rate of convergence shown in these figures is linear (with very fast contraction rate) for all the cases we implemented.

Acknowledgments. This research was supported by Research Council KUL: CoE EF/05/006 Optimization in Engineering(OPTEC), GOA AMBioRICS, IOF-SCORES4CHEM, several PhD/ postdoc & fellow grants; the Flemish Government via FWO: PhD/postdoc grants, projects G.0452.04, G.0499.04, G.0211.05, G.0226.06, G.0321.06, G.0302.07, G.0320.08 (convex MPC), G.0558.08 (Robust MHE), G.0557.08, G.0588.09, research communities (ICCoS, ANMMM, MLDM) and via IWT: PhD Grants, McKnow-E, Eureka-Flite+EU: ERNSI; FP7-HD-MPC (Collaborative Project STREP-grantnr. 223854), Erbocon, Contract Research: AMINAL, and Helmholtz Gemeinschaft: viCERP; Austria: ACCM, and the Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, Dynamical systems, control and optimization, 2007-2011).

The authors are very much thankful to the anonymous referees, who corrected numerous mistakes and suggested several improvements.

References

1. S. Boyd and L. Vandenberghe (2004). *Convex optimization.* University Press, Cambridge.
2. R. Correa and H. Ramirez C (2004). A global algorithm for nonlinear semidefinite programming. *SIAM J. Optim.*, 15(1):303–318.
3. A. L. Dontchev and T. R. Rockafellar (1996). Characterizations of strong regularity for variational inequalities over polyhedral convex sets. *SIAM J. Optim.*, 6(4):1087–1105.
4. B. Fares, D. Noll, and P. Apkarian (2002). Robust control via sequential semidefinite programming. *SIAM J. Control Optim.*, 40:1791–1820.
5. R. W. Freund, F. Jarre, and C. H. Vogelbusch (2007). Nonlinear semidefinite programming: sensitivity, convergence, and an application in passive reduced-order modeling. *Mathematical Programming*, Ser. B, 109:581–611.
6. M. Fukushima, Z.-Q. Luo, and P. Tseng (2003). A sequential quadratically constrained quadratic programming method for differentiable convex minimization. *SIAM J. Optimization*, 13(4):1098–1119.
7. H. Jaddu (2002). Direct solution of nonlinear optimal control problems using quasi-linearization and Chebyshev polynomials. *Journal of the Franklin Institute*, 339:479–498.
8. F. Jarre (2003). On an approximation of the Hessian of the Lagrangian. *Optimization Online* (http://www.optimization−online.org/DB_HTML/2003/12/800.html).
9. J.L. Junkins and J.D. Turner (1986). *Optimal spacecraft rotational maneuvers.* Elsevier, Amsterdam.
10. C. Kanzow, C. Nagel, H. Kato and M. Fukushima (2005). Successive linearization methods for nonlinear semidefinite programs. *Computational Optimization and Applications*, 31:251–273.
11. D. Klatte and B. Kummer (2001). *Nonsmooth equations in optimization: regularity, calculus, methods and applications.* Springer-Verlag, New York.
12. A. S. Lewis and S. J. Wright (2008). A proximal method for composite minimization. http://arxiv.org/abs/0812.0423.
13. S. M. Robinson (1980). Strong regularity generalized equations, *Mathematics of Operation Research*, 5(1):43–62.
14. R. T. Rockafellar and R. J-B. Wets (1997). *Variational analysis.* Springer-Verlag, New York.

Fixed-Order H-infinity Optimization of Time-Delay Systems

Suat Gumussoy[1] and Wim Michiels[2]

[1] Department of Computer Science, K. U. Leuven, Celestijnenlaan 200A, 3001 Heverlee, Belgium, suat.gumussoy@cs.kuleuven.be

[2] Department of Computer Science, K. U. Leuven, Celestijnenlaan 200A, 3001 Heverlee, Belgium, wim.michiels@cs.kuleuven.be

Summary. H-infinity controllers are frequently used in control theory due to their robust performance and stabilization. Classical H-infinity controller synthesis methods for finite dimensional LTI MIMO plants result in high-order controllers for high-order plants whereas low-order controllers are desired in practice. We design fixed-order H-infinity controllers for a class of time-delay systems based on a non-smooth, non-convex optimization method and a recently developed numerical method for H-infinity norm computations.

Robust control techniques are effective to achieve stability and performance requirements under model uncertainties and exogenous disturbances [16]. In robust control of linear systems, stability and performance criteria are often expressed by H-infinity norms of appropriately defined closed-loop functions including the plant, the controller and weights for uncertainties and disturbances. The optimal H-infinity controller minimizing the H-infinity norm of the closed-loop functions for finite dimensional multi-input-multi-output (MIMO) systems is computed by Riccati and linear matrix inequality (LMI) based methods [8, 9]. The order of the resulting controller is equal to the order of the plant and this is a restrictive condition for high-order plants. In practical implementations, fixed-order controllers are desired since they are cheap and easy to implement in hardware and non-restrictive in sampling rate and bandwidth. The fixed-order optimal H-infinity controller synthesis problem leads to a non-convex optimization problem. For certain closed-loop functions, this problem is converted to an interpolation problem and the interpolation function is computed based on continuation methods [1]. Recently fixed-order H-infinity controllers are successfully designed for finite dimensional LTI MIMO plants using a non-smooth, non-convex optimization method [10]. This approach allows the user to choose the controller order and tunes the parameters of the controller to minimize the H-infinity norm of the objective function using the norm value and its derivatives with respect to the controller parameters. In our work, we design fixed-order H-infinity controllers for a class of time-delay systems based on a non-smooth, non-convex

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_10, © Springer-Verlag Berlin Heidelberg 2010

optimization method and a recently developed H-infinity norm computation method [13]. H-infinity!optimization control!fixed-order

1 Problem Formulation

We consider time-delay plant G determined by equations of the form,

$$\dot{x}(t) = A_0 x(t) + \sum_{i=1}^{m} A_i x(t-\tau_i) + B_1 w(t) + B_2 u(t-\tau_{m+1}) \tag{1}$$
$$z(t) = C_1 x(t) + D_{11} w(t) + D_{12} u(t) \tag{2}$$
$$y(t) = C_2 x(t) + D_{21} w(t) + D_{22} u(t-\tau_{m+2}). \tag{3}$$

where all system matrices are real with compatible dimensions and $A_0 \in \mathbb{R}^{n\times n}$. The input signals are the exogenous disturbances w and the control signals u. The output signals are the controlled signals z and the measured signals y. All system matrices are real and the time-delays are positive real numbers. In robust control design, many design objectives can be expressed in terms of norms of closed-loop transfer functions between appropriately chosen signals w to z.

The controller K has a fixed-structure and its order n_K is chosen by the user *a priori* depending on design requirements,

$$\dot{x}_K(t) = A_K x_K(t) + B_K y(t) \tag{4}$$
$$u(t) = C_K x_K(t) \tag{5}$$

where all controller matrices are real with compatible dimensions and $A_K \in \mathbb{R}^{n_K \times n_K}$.

By connecting the plant G and the controller K, the equations of the closed-loop system from w to z are written as,

$$\begin{aligned} \dot{x}_{cl}(t) &= A_{cl,0} x_{cl}(t) + \sum_{i=1}^{m+2} A_{cl,i} x_{cl}(t-\tau_i) + B_{cl} w(t) \\ z(t) &= C_{cl} x_{cl}(t) + D_{cl} w(t) \end{aligned} \tag{6}$$

where

$$\begin{aligned} A_{cl,0} &= \begin{pmatrix} A_0 & 0 \\ B_K C_2 & A_K \end{pmatrix},\ A_{cl,i} = \begin{pmatrix} A_i & 0 \\ 0 & 0 \end{pmatrix} \text{ for i} = 1,\ldots,\text{m}, \\ A_{cl,m+1} &= \begin{pmatrix} 0 & B_2 C_K \\ 0 & 0 \end{pmatrix}, A_{cl,m+2} = \begin{pmatrix} 0 & 0 \\ 0 & B_K D_{22} C_K \end{pmatrix}, \\ B_{cl} &= \begin{pmatrix} B_1 \\ B_K D_{21} \end{pmatrix},\ C_{cl} = \begin{pmatrix} C_1 & D_{12} C_K \end{pmatrix},\ D_{cl} = D_{11}. \end{aligned} \tag{7}$$

The closed-loop matrices contain the controller matrices (A_K, B_K, C_K) and these matrices can be tuned to achieve desired closed-loop characteristics.

The transfer function from w to z is,

$$T_{zw}(s) = C_{cl} \left(sI - A_{cl,0} - \sum_{i=1}^{m+2} A_{cl,i} e^{-\tau_i s} \right)^{-1} B_{cl} + D_{cl} \tag{8}$$

and we define fixed-order H-infinity optimization problem as the following.

Problem Given a controller order n_K, find the controller matrices (A_K, B_K, C_K) stabilizing the system and minimizing the H-infinity norm of the transfer function T_{zw}.

2 Optimization Problem

2.1 Algorithm

The optimization algorithm consists of two steps:

1. **Stabilization:** minimizing the spectral abscissa, the maximum real part of the characteristic roots of the closed-loop system. The optimization process can be stopped when the controller parameters are found that stabilizes T_{zw} and these parameters are the feasible points for the H-infinity optimization of T_{zw}.
2. **H-infinity optimization:** minimizing the H-infinity norm of T_{zw} using the starting points from the stabilization step.

If the first step is successful, then a feasible point for the H-infinity optimization is found, i.e., a point where the closed-loop system is stable. If in the second step the H-infinity norm is reduced in a quasi-continuous way, then the feasible set cannot be left under mild controllability/observability conditions.

Both objective functions, the spectral abscissa and the H-infinity norm, are non-convex and not everywhere differentiable but smooth almost everywhere [15]. Therefore we choose a hybrid optimization method to solve a non-smooth and non-convex optimization problem, which has been successfully applied to design fixed-order controllers for the finite dimensional MIMO systems [10].

The optimization algorithm searches for the local minimizer of the objective function in three steps [5]:

1. A quasi-Newton algorithm (in particular, BFGS) provides a fast way to approximate a local minimizer [12],
2. A local bundle method attempts to verify local optimality for the best point found by BFGS,
3. If this does not succeed, gradient sampling [6] attempts to refine the approximation of the local minimizer, returning a rough optimality measure.

The non-smooth, non-convex optimization method requires the evaluation of the objective function -in the second step this is the H-infinity norm of T_{zw}- and the gradient of the objective function with respect to controller parameters where it exists. Recently a predictor-corrector algorithm has been developed to compute the H-infinity norm of time-delay systems [13]. We computed the gradients using the derivatives of singular values at frequencies where the H-infinity norm is achieved. Based on the evaluation of the objective function and its gradients, we apply the optimization method to compute fixed-order controllers. The computation of H-infinity norm of time-delay systems (8) is discussed in the following section. non-smooth optimization

2.2 Computation of the H-infinity Norm

We implemented a predictor-corrector type method to evaluate the H-infinity norm of T_{zw} in two steps (for details we refer to [13]): H-infinity!norm computationnorm!H-infinity

- **Prediction step:** we calculate the approximate H-infinity norm and corresponding frequencies where the highest peak values in the singular value plot occur.
- **Correction step:** we correct the approximate results from the prediction step.

Theoretical Foundation

The following theorem generalizes the well-known relation between the existence of singular values of the transfer function equal to a fixed value and the presence of imaginary axis eigenvalues of a corresponding Hamiltonian matrix [7] to time-delay systems:

Theorem 1. *[13] Let $\xi > 0$ be such that the matrix*

$$D_\xi := D_{cl}^T D_{cl} - \xi^2 I$$

is non-singular and define $\tau_{\max}$ as the maximum of the delays $(\tau_1, \ldots, \tau_{m+2})$. For $\omega \geq 0$, the matrix $T_{zw}(j\omega)$ has a singular value equal to $\xi > 0$ if and only if $\lambda = j\omega$ is an eigenvalue of the linear infinite dimensional operator $\mathcal{L}_\xi$ on $X := \mathcal{C}([-\tau_{\max},\ \tau_{\max}], \mathbb{C}^{2n})$ which is defined by

$$\mathcal{D}(\mathcal{L}_\xi) = \big\{\phi \in X : \phi' \in X,\ \phi'(0) = M_0\phi(0) + \sum_{i=1}^{m+2} (M_i\phi(-\tau_i) + M_{-i}\phi(\tau_i))\big\}, \qquad (9)$$

$$\mathcal{L}_\xi \phi = \phi',\ \phi \in \mathcal{D}(\mathcal{L}_\xi) \qquad (10)$$

with

$$M_0 = \begin{bmatrix} A_{cl,0} - B_{cl} D_\xi^{-1} D_{cl}^T C_{cl} & -B_{cl} D_\xi^{-1} B_{cl}^T \\ \xi^2 C_{cl}^T D_\xi^{-T} C_{cl} & -A_{cl,0}^T + C_{cl}^T D_{cl} D_\xi^{-1} B_{cl}^T \end{bmatrix},$$

$$M_i = \begin{bmatrix} A_{cl,i} & 0 \\ 0 & 0 \end{bmatrix},\quad M_{-i} = \begin{bmatrix} 0 & 0 \\ 0 & -A_{cl,i}^T \end{bmatrix},\quad 1 \leq i \leq m+2.$$

By Theorem 1, the computation of H-infinity norm of T_{zw} can be formulated as an eigenvalue problem for the linear operator $\mathcal{L}_\xi$.

Corollary 1.
$\|T_{zw}\|_\infty = \sup\{\xi > 0 :$ *operator $\mathcal{L}_\xi$ has an eigenvalue on the imaginary axis*$\}$

Conceptually Theorem 1 allows the computation of H-infinity norm via the well-known level set method [2, 4]. However, $\mathcal{L}_\xi$ is an infinite dimensional operator. Therefore, we compute the H-infinity norm of the transfer function T_{zw} in two steps:

1) The prediction step is based on a matrix approximation of $\mathcal{L}_\xi$.

2) The correction step is based on reformulation of the eigenvalue problem of $\mathcal{L}_\xi$ as a nonlinear eigenvalue problem of a finite dimension.

The approximation of the linear operator $\mathcal{L}_\xi$ and the corresponding standard eigenvalue problem for Corollary 1 is given in Section 2.3. The correction algorithm of the approximate results in the second step is explained in Section 2.4.

2.3 Prediction Step

The infinite dimensional operator $\mathcal{L}_\xi$ is approximated by a matrix $\mathcal{L}_\xi^N$. Based on the numerical methods for finite dimensional systems [2, 4], the H-infinity norm of the transfer function T_{zw} can be computed approximately as

Corollary 2.
$\|T_{zw}\|_\infty \approx \sup\{\xi > 0 : \textit{operator } \mathcal{L}_\xi^N \textit{ has an eigenvalue on the imaginary axis}\}$.

The infinite-dimensional operator $\mathcal{L}_\xi$ is approximated by a matrix using a *spectral method* (see, e.g. [3]). Given a positive integer N, we consider a mesh Ω_N of $2N+1$ distinct points in the interval $[-\tau_{\max},\ \tau_{\max}]$:

$$\Omega_N = \{\theta_{N,i},\ i = -N, \ldots, N\}, \tag{11}$$

where

$$-\tau_{\max} \leq \theta_{N,-N} < \ldots < \theta_{N,0} = 0 < \cdots < \theta_{N,N} \leq \tau_{\max}.$$

This allows to replace the continuous space X with the space X_N of discrete functions defined over the mesh Ω_N, i.e. any function $\phi \in X$ is discretized into a block vector $x = [x_{-N}^T \cdots\ x_N^T]^T \in X_N$ with components

$$x_i = \phi(\theta_{N,i}) \in \mathbb{C}^{2n}, \quad i = -N, \ldots, N.$$

Let $\mathcal{P}_N x,\ x \in X_N$ be the unique $\mathbb{C}^{2n}$ valued interpolating polynomial of degree $\leq 2N$ satisfying

$$\mathcal{P}_N x(\theta_{N,i}) = x_i, \quad i = -N, \ldots, N.$$

In this way, the operator $\mathcal{L}_\xi$ over X can be approximated with the matrix $\mathcal{L}_\xi^N :\ X_N \rightarrow X_N$, defined as

$$\begin{aligned}\left(\mathcal{L}_\xi^N\ x\right)_i &= \left(\mathcal{P}_N x\right)'(\theta_{N,i}), \quad i = -N, \ldots, -1, 1, \ldots, N,\\ \left(\mathcal{L}_\xi^N\ x\right)_0 &= M_0 \mathcal{P}_N x(0) + \sum_{i=1}^{m+2} (M_i \mathcal{P}_N x(-\tau_i) + M_{-i} \mathcal{P}_N x(\tau_i)).\end{aligned}$$

Using the Lagrange representation of $\mathcal{P}_N x$,

$$\mathcal{P}_N x = \textstyle\sum_{k=-N}^{N} l_{N,k}\ x_k,$$

where the Lagrange polynomials $l_{N,k}$ are real valued polynomials of degree $2N$ satisfying

$$l_{N,k}(\theta_{N,i}) = \begin{cases} 1 & i = k, \\ 0 & i \neq k, \end{cases}$$

we obtain the explicit form

$$\mathcal{L}_\xi^N = \begin{bmatrix} d_{-N,-N} & \dots & d_{-N,N} \\ \vdots & & \vdots \\ d_{-1,-N} & \dots & d_{-1,N} \\ a_{-N} & \dots & a_N \\ d_{1,-N} & \dots & d_{1,N} \\ \vdots & & \vdots \\ d_{N,-N} & \dots & d_{N,N} \end{bmatrix} \in \mathbb{R}^{(2N+1)(2n)\times(2N+1)2n},$$

where

$$\begin{aligned} d_{i,k} &= l'_{N,k}(\theta_{N,i})I, \quad i,k \in \{-N,\dots,N\},\ i \neq 0, \\ a_0 &= M_0\ x_0 + \textstyle\sum_{k=1}^{m+2} \left(M_k l_{N,0}(-\tau_k) + M_{-k} l_{N,0}(\tau_k)\right), \\ a_i &= \textstyle\sum_{k=1}^{m+2} \left(M_k l_{N,i}(-\tau_k) + M_{-k} l_{N,i}(\tau_k)\right),\ k \in \{-N,\dots,N\},\ k \neq 0. \end{aligned}$$

2.4 Correction Step

By using the finite dimensional level set methods, the largest level set ξ where $\mathcal{L}_\xi^N$ has imaginary axis eigenvalues and their corresponding frequencies are computed. In the correction step, these approximate results are corrected by using the property that the eigenvalues of the $\mathcal{L}_\xi$ appear as solutions of a finite dimensional nonlinear eigenvalue problem. The following theorem establishes the link between the linear infinite dimensional eigenvalue problem for $\mathcal{L}_\xi$ and the nonlinear eigenvalue problem.

Theorem 2. *[13] Let $\xi > 0$ be such that the matrix*

$$D_\xi := D_{cl}^T D_{cl} - \xi^2 I$$

is non-singular. Then, λ is an eigenvalue of linear operator $\mathcal{L}_\xi$ if and only if

$$\det H_\xi(\lambda) = 0, \tag{12}$$

where

$$H_\xi(\lambda) := \lambda I - M_0 - \sum_{i=1}^{m+2} \left(M_i e^{-\lambda\tau_i} + M_{-i} e^{\lambda\tau_i}\right) \tag{13}$$

and the matrices M_0, M_i, M_{-i} are defined in Theorem 1.

The correction method is based on the property that if $\hat{\xi} = \|T_{zw}(j\omega)\|_\infty$, then (13) has a multiple non-semisimple eigenvalue. If $\hat{\xi} \geq 0$ and $\hat{\omega} \geq 0$ are such that

$$\|T_{zw}(j\omega)\|_{\mathcal{H}_\infty} = \hat{\xi} = \sigma_1(T_{zw}(j\hat{\omega})), \tag{14}$$

then setting

$$h_\xi(\lambda) = \det H_\xi(\lambda),$$

the pair $(\hat{\omega}, \hat{\xi})$ satisfies

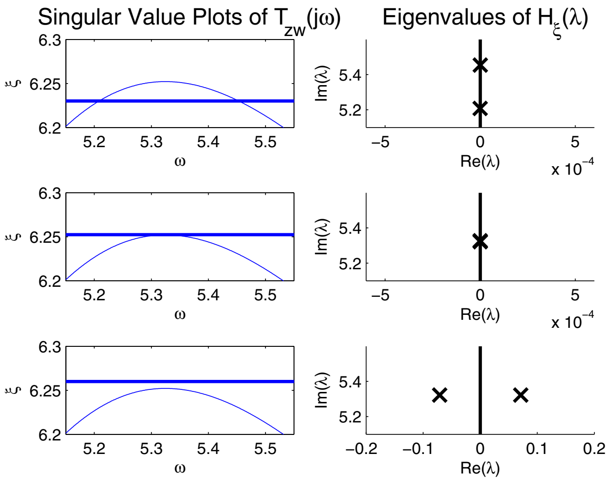


Fig. 1. (left) Intersections of the singular value plot of T_{zw} with the horizontal line $\xi = c$, for $c < \hat{\xi}$ (top), $c = \hat{\xi}$ (middle) and $c > \hat{\xi}$ (bottom). (right) Corresponding eigenvalues of $H_{\xi}(\lambda)$ (13).

$$h_{\xi}(j\omega) = 0, \quad h'_{\xi}(j\omega) = 0. \tag{15}$$

This property is clarified in Figure 1.

The drawback of working directly with (15) is that an explicit expression for the determinant of H_{ξ} is required. This scalar-valued conditions can be equivalently expressed in a matrix-based formulation.

$$\begin{cases} H(j\omega,\ \xi)\begin{bmatrix} u, \\ v \end{bmatrix} = 0, \quad n(u,v) = 0, \\ \Im\left\{v^*\left(I + \sum_{i=1}^{m+1} A_{cl,i}\tau_i e^{-j\omega\tau_i}\right)u\right\} = 0 \end{cases} \tag{16}$$

where $n(u,v) = 0$ is a normalizing condition. The approximate H-infinity norm and its corresponding frequencies can be corrected by solving (16). For further details, see [13].

2.5 Computing the Gradients

The optimization algorithm requires the derivatives of H-infinity norm of the transfer function T_{zw} with respect to the controller matrices whenever it is differentiable. Define the H-infinity norm of the function T_{zw} as

$$f(A_{cl,0}, \ldots, A_{cl,m+2}, B_{cl}, C_{cl}, D_{cl}) = \|T_{zw}(j\omega)\|_{\infty}.$$

These derivatives exist whenever there is a unique frequency $\hat{\omega}$ such that (14) holds, and, in addition, the largest singular value $\hat{\xi}$ of $T_{zw}(j\hat{\omega})$ has multiplicity one. Let w_l and w_r be the corresponding left and right singular vector, i.e.

$$\begin{aligned} T_{zw}(j\hat{\omega})\ w_r &= \hat{\xi}\ w_l, \\ w_l^*\ T_{zw}(j\hat{\omega}) &= \hat{\xi}\ w_r^*. \end{aligned} \tag{17}$$

When defining $\frac{\partial f}{\partial A_{cl,0}}$ as a n-by-n matrix whose (k,l)-th element is the derivative of f with respect to the (k,l)-th element of $A_{cl,0}$, and defining the other derivatives in a similar way, the following expressions are obtained [14]:

$$\begin{aligned} \frac{\partial f}{\partial A_{cl,0}} &= \frac{\Re\left(M(j\hat{\omega})^* C_{cl}^T w_l w_r^* B_{cl}^T M(j\hat{\omega})^*\right)}{w_r^* w_r}, \\ \frac{\partial f}{\partial A_{cl,i}} &= \frac{\Re\left(M(j\hat{\omega})^* C_{cl}^T w_l w_r^* B_{cl}^T M(j\hat{\omega})^* e^{j\omega\tau_i}\right)}{w_r^* w_r} \text{ for } i = 1, \ldots, m+2, \\ \frac{\partial f}{\partial B_{cl}} &= \frac{\Re(M(j\hat{\omega})^* C_{cl}^T w_l w_r^*)}{w_r^* w_r}, \frac{\partial f}{\partial C_{cl}} = \frac{\Re(w_l w_r^* B_{cl}^T M(j\hat{\omega})^*)}{w_r^* w_r}, \\ \frac{\partial f}{\partial D_{cl}} &= \frac{\Re\left(w_l w_r^*\right)}{w_r^* w_r} \end{aligned}$$

where $M(j\omega) = \left(j\omega I - A_{cl,0} - \sum_{i=1}^{m+2} A_{cl,i} e^{-j\omega\tau_i}\right)^{-1}$.

We compute the gradients with respect to the controller matrices as

$$\begin{aligned} \frac{\partial f}{\partial A_K} &= \left[0_{n_K \times n}\ I_{n_K}\right] \frac{\partial f}{\partial A_{cl,0}} \begin{bmatrix} 0_{n \times n_K} \\ I_{n_K} \end{bmatrix}, \\ \frac{\partial f}{\partial B_K} &= \left[0_{n_K \times n}\ I_{n_K}\right] \frac{\partial f}{\partial A_{cl,0}} \begin{bmatrix} I_n \\ 0_{n_K \times n} \end{bmatrix} C_2^T \\ &\quad + \left[0_{n_K \times n}\ I_{n_K}\right] \frac{\partial f}{\partial A_{cl,m+2}} \begin{bmatrix} 0_{n \times n_K} \\ I_{n_K} \end{bmatrix} C_K^T D_{22}^T + \left[0_{n_K \times n}\ I_{n_K}\right] \frac{\partial f}{\partial B_{cl}} D_{21}^T, \\ \frac{\partial f}{\partial C_K} &= B_2^T \left[I_n\ 0_{n \times n_K}\right] \frac{\partial f}{\partial A_{cl,m+1}} \begin{bmatrix} 0_{n \times n_K} \\ I_{n_K} \end{bmatrix} \\ &\quad + D_{22}^T B_K^T \left[0_{n_K \times n}\ I_{n_K}\right] \frac{\partial f}{\partial A_{cl,m+2}} \begin{bmatrix} 0_{n \times n_K} \\ I_{n_K} \end{bmatrix} + D_{12}^T \frac{\partial f}{\partial C_{cl}} \begin{bmatrix} 0_{n \times n_K} \\ I_{n_K} \end{bmatrix} \end{aligned}$$

where the matrices I_n, I_{n_K} and $0_{n \times n_K}$, $0_{n_K \times n}$ are identity and zero matrices.

3 Examples

We consider the time-delay system with the following state-space representation,

$$\begin{aligned} \dot{x}(t) &= -x(t) - 0.5x(t-1) + w(t) + u(t), \\ z(t) &= x(t) + u(t), \\ y(t) &= x(t) + w(t). \end{aligned}$$

We designed the first-order controller, $n_K = 1$,

$$\begin{aligned} \dot{x}_K(t) &= 3.61 x_K(t) + 1.39 y(t), \\ u(t) &= -0.83 x_K(t), \end{aligned}$$

achieving the closed-loop H-infinity norm 0.064. The closed-loop H-infinity norms of fixed-order controllers for $n_K = 2$ and $n_K = 3$ are 0.021 and 0.020 respectively.

Our second example is a 4^{th}-order time-delay system. The system contains 4 delays and has the following state-space representation,

$$\dot{x}(t) = \begin{pmatrix} -4.4656 & -0.4271 & 0.4427 & -0.1854 \\ -0.8601 & -5.6257 & 0.8577 & -0.5210 \\ 0.9001 & -0.7177 & -6.5358 & 0.0417 \\ -0.6836 & 0.0242 & 0.4997 & -3.5618 \end{pmatrix} x(t) + \begin{pmatrix} 0.6848 & -0.0618 & 0.5399 & 0.5057 \\ 0.3259 & -0.3810 & 0.6592 & -0.0066 \\ 0.6325 & 0.3752 & 0.4122 & 0.7303 \\ 0.5878 & 0.9737 & 0.1907 & -0.8639 \end{pmatrix} x(t-3.2)$$

$$+ \begin{pmatrix} 0.9371 & -0.7859 & 0.1332 & 0.7429 \\ -0.8025 & 0.4483 & 0.6226 & 0.0152 \\ 0.0940 & 0.2274 & 0.1536 & 0.5776 \\ -0.1941 & 0.5659 & 0.8881 & -0.0539 \end{pmatrix} x(t-3.4) + \begin{pmatrix} 0.6576 & -0.8543 & -0.3460 & 0.6415 \\ -0.3550 & 0.5024 & 0.6081 & 0.9038 \\ 0.9523 & 0.6624 & 0.0765 & -0.8475 \\ -0.4436 & 0.8447 & -0.0734 & 0.4173 \end{pmatrix} x(t-3.9)$$

$$+ \begin{pmatrix} 1 & 0 \\ -1.6 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} w(t) + \begin{pmatrix} 0.2 \\ -1 \\ 0.1 \\ -0.4 \end{pmatrix} u(t-0.2)$$

$$\begin{aligned} z(t) &= \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0.1 & 1 \\ -1 & 0.2 \end{pmatrix} w(t) + \begin{pmatrix} 1 \\ -1 \end{pmatrix} u(t) \\ y(t) &= \begin{pmatrix} 1 & 0 & -1 & 0 \end{pmatrix} x(t) + \begin{pmatrix} -2 & 0.1 \end{pmatrix} w(t) + 0.4u(t-0.2) \end{aligned}$$

When $n_K = 1$, our method finds the controller achieving the closed-loop H-infinity norm 1.2606,

$$\begin{aligned} \dot{x}_K(t) &= -0.712 x_K(t) - 0.1639 y(t), \\ u(t) &= -0.2858 x_K(t) \end{aligned}$$

and the results for $n_K = 2$ and $n_K = 3$ are 1.2573 and 1.2505 respectively.

4 Concluding Remarks

We successfully designed fixed-order H-infinity controllers for a class of time-delay systems. The method is based on non-smooth, non-convex optimization techniques and allows the user to choose the controller order as desired. Our approach can be extended to general time-delay systems. Although we illustrated our method for a dynamic controller, it can be applied to more general controller structures. The only requirement is that the closed-loop matrices should depend smoothly on the controller parameters. On the contrary, the existing controller design methods optimizing the closed-loop H-infinity norm are based on Lyapunov theory and linear matrix inequalities. These methods are conservative if the form of the Lyapunov functionals is restricted, and they require full state information.

5 Acknowledgements

This article presents results of the Belgian Programme on Interuniversity Poles of Attraction, initiated by the Belgian State, Prime Ministers Office for Science, Technology and Culture, the Optimization in Engineering Centre OPTEC of the K.U.Leuven, and the project STRT1-09/33 of the K.U.Leuven Research Foundation.

References

1. A. Blomqvist, A. Lindquist and R. Nagamune (2003) Matrix-valued Nevanlinna-Pick interpolation with complexity constraint: An optimization approach. IEEE Transactions on Automatic Control, 48:2172–2190.
2. S. Boyd and V. Balakrishnan (1990) A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its $\mathcal{L}_\infty$-norm. Systems & Control Letters, 15:1–7.
3. D. Breda, S. Maset and R. Vermiglio (2006) Pseudospectral approximation of eigenvalues of derivative operators with non-local boundary conditions. Applied Numerical Mathematics, 56:318–331.
4. N.A. Bruinsma and M. Steinbuch (1990) A fast algorithm to compute the $\mathcal{H}_\infty$-norm of a transfer function matrix. Systems & Control Letters, 14:287–293.
5. J.V. Burke, D. Henrion, A.S. Lewis and M.L. Overton (2006). Stabilization via nonsmooth, nonconvex optimization. IEEE Transactions on Automatic Control, 51:1760-1769.
6. J.V. Burke, A.S. Lewis and M.L. Overton, (2003) A robust gradient sampling algorithm for nonsmooth, nonconvex optimization. SIAM Journal on Optimization, 15:751–779.
7. R. Byers (1988) A bisection method for measuring the distance of a stable matrix to the unstable matrices. SIAM Journal on Scientific and Statistical Computing, 9:875–881.
8. J.C. Doyle, K. Glover, P.P. Khargonekar and B.A. Francis (1989) State-Space solutions to standard $\mathcal{H}^2$ and $\mathcal{H}^\infty$ control problems. IEEE Transactions on Automatic Control 46:1968–1972.
9. P. Gahinet and P. Apkarian (1994) An Linear Matrix Inequality Approach to $\mathcal{H}_\infty$ Control. International Journal of Robust and Nonlinear Control 4:421–448.
10. S. Gumussoy and M.L. Overton, (2008) Fixed-Order H-Infinity Controller Design via HIFOO, a Specialized Nonsmooth Optimization Package. Proceedings of the American Control Conference 2750-2754.
11. R. Hryniv and P. Lancaster (1999) On the perturbation of analytic matrix functions. Integral Equations and Operator Theory, 34:325–338.
12. A.S. Lewis and M.L.Overton, (2009) Nonsmooth optimization via BFGS. Submitted to SIAM Journal on Optimization.
13. W. Michiels and S. Gumussoy, (2009) Computation of H-infinity Norms for Time-Delay Systems. Accepted to SIAM Journal on Matrix Analysis and Applications. See also Technical Report TW551, Department of Computer Science, K.U.Leuven, 2009.
14. M. Millstone, (2006) HIFOO 1.5: Structured control of linear systems with a non-trivial feedthrough. Master's Thesis, New York University.
15. J. Vanbiervliet, K. Verheyden, W. Michiels and S. Vandewalle (2008) A nonsmooth optimization approach for the stabilization of time-delay systems. ESAIM Control, Optimisation and Calcalus of Variations 14:478–493.
16. K. Zhou, J.C. Doyle and K. Glover (1995) Robust and optimal control. Prentice Hall.

Using Model Order Reduction for the Parameter Optimization of Large Scale Dynamical Systems

Yao Yue and Karl Meerbergen

Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A , 3001 Heverlee, Belgium {yao.yue, karl.meerbergen}@cs.kuleuven.be

Summary. Optimization problems such as the parameter design of dynamical systems are often computationally expensive. In this paper, we apply Krylov based model order reduction techniques to the parameter design problem of an acoustic cavity to accelerate the computation of both function values and derivatives, and therefore, drastically improve the performance of the optimization algorithms. Two types of model reduction techniques are explored: conventional model reduction and parameterized model reduction. The moment matching properties of derivative computation via the reduced model are discussed. Numerical results show that both methods are efficient in reducing the optimization time.

1 Introduction

Numerical parameter studies of acoustic problems arising from applications such as airplane engines and insulation panels along motorways or in houses are often carried out in order to choose the 'optimal' values of the parameters to meet design objectives like reducing noise, and thus can be viewed as optimization problems. These problems are often computationally extremely expensive, since for each parameter value, an entire frequency response function (FRF) needs to be computed, which by itself is already quite expensive.

The computational cost for the FRF has been dramatically reduced by a factor of ten or more by using model order reduction (MOR) techniques [12]. The goal of MOR is to construct a low order model to approximate the original large-scale model with high accuracy to reduce the computational cost. It has been successfully applied to many different fields such as circuit simulations [6, 13] and (vibro) acoustics [12]. However, little work has been done to introduce MOR into optimization although optimization problems are more expensive in general because solving an FRF is only one iteration step in optimization.

In this paper, we first introduce a minimax problem arising from the parameter design of an acoustic cavity, propose an algorithm to solve it, and analyze its computational cost. Two types of Krylov based MOR methods

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_11, © Springer-Verlag Berlin Heidelberg 2010

are investigated to reduce the cost: conventional MOR on a single variable like SOAR (Second Order ARnoldi) [2, 14] and parameterized MOR (PMOR) on multiple variables like PIMTAP (Parameterized Interconnect Macromodeling via a Two-directional Arnoldi Process) [10]. We show how to integrate SOAR and PIMTAP into the minimax algorithm, especially analyzing the moment matching properties of the derivatives. Since computing the function values and the gradients is the most expensive part in our application, both SOAR and PIMTAP can drastically increase the performance of optimization algorithms. Numerical results show that MOR can significantly reduce the optimization time while still locating the optimizer with high accuracy.

2 Minimax Optimization

In this part, we introduce a minimax optimization problem arising from the optimal design of an acoustic cavity, which is a unit cube. We imposed homogeneous boundary conditions on all faces except one where we imposed an admittance boundary condition. The mathematical model of this problem is

$$\begin{cases} -\nabla^2 u + k^2 u = f, & \\ \frac{\partial u}{\partial n} + i\omega\gamma\, u = 0, & \text{for } u \in \Gamma_1, \\ u = 0, & \text{for } u \in \Gamma_2, \end{cases} \tag{1}$$

where u, k, f, n, ω and γ denote displacement, the wave number, the excitation, the normal direction, the frequency and the admittance ratio, respectively, Γ_1 denotes the face with admittance boundary condition and Γ_2 denotes the five other faces. We discretized the unit cube with finite differences and analyzed this system in the frequency domain. The output of the system is the displacement of a given point inside the cavity, which depends on both ω and γ. Therefore, we denote it by $y(\omega, \gamma)$. For any fixed γ_0, $|y(\omega, \gamma_0)|^2$ defines an FRF and we show three FRFs with different γ values in Fig. 1. Our design objective is to minimize the highest peak of the FRF by choosing the optimal γ. To study how the local maxima change w.r.t γ, we consider the necessary condition $\frac{\partial |y|^2}{\partial \omega} = 0$. This condition implicitly defines several curves, namely *Local MAximum Curve*s (*LMAC*s) and *Local MInimum Curve*s (*LMIC*s). Note that an LMAC or LMIC may discontinue at certain points because of its interaction with other LMACs or LMICs. To better understand this optimization problem, we project the LMACs onto the $|y|^2$–γ plane and get Fig. 2. In practice, these *Projected LMAC*s (*PLMAC*s) appear to be convex for γs near the optimizer. The *Projected Global MAximum Curve* (*PGMAC*) in Fig. 2 is the maximum of all the PLMACs. Although a PLMAC may discontinue at some point, the PGMAC is defined everywhere and continuous. However, it usually has non-differentiable kink points where different PLMACs intersect.

Mathematically, our optimization problem can be formulated as

$$\min_{\gamma} \max_{\omega_L \le \omega \le \omega_H} |y(\omega, \gamma)|^2 \qquad \text{s.t.} \quad \begin{cases} (K + i\omega\gamma\, C - \omega^2 M)x(\omega, \gamma) = f \\ y(\omega, \gamma) = l^* x(\omega, \gamma) \end{cases} \tag{2}$$

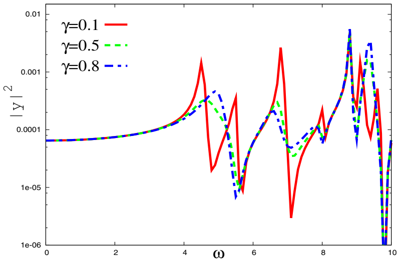


Fig. 1. FRFs with different γ values

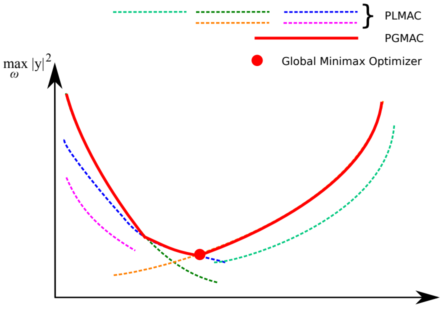


Fig. 2. Global Optimizer

where $K, C, M \in \mathbb{C}^{n \times n}$ are stiffness matrix, damping matrix and mass matrix respectively and $f, x, l \in \mathbb{C}^n$ are input vector, state vector and output vector respectively. Equivalently, the problem (2) can also be formulated as

$$\begin{aligned} &\min_{\gamma} \quad g(\gamma), && \text{(Min Phase or Outer Phase)}, \\ g(\gamma) = &\max_{\omega_L \le \omega \le \omega_H} |y(\omega, \gamma)|^2, && \text{(Max Phase or Inner Phase)}. \end{aligned} \tag{3}$$

For this system, the derivatives are cheap if y is already computed since

$$\begin{aligned} y &= l^*(K + i\omega\gamma C - \omega^2 M)^{-1} f, \\ \frac{\partial y}{\partial \omega} &= l^*(K + i\omega\gamma C - \omega^2 M)^{-1}(2\omega M - i\gamma C)(K + i\omega\gamma C - \omega^2 M)^{-1} f, \\ \frac{\partial y}{\partial \gamma} &= l^*(K + i\omega\gamma C - \omega^2 M)^{-1}(-i\omega C)(K + i\omega\gamma C - \omega^2 M)^{-1} f, \end{aligned} \tag{4}$$

and they share the same computationally dominant part, namely the LU factorization of the matrix $K + i\omega\gamma C - \omega^2 M$. Therefore, the derivatives should be exploited for fast convergence. On the other hand, to make MOR more applicable, we choose Quasi Newton type method because the first order derivatives can be approximated well with the reduced model as we will discuss in Section 4. We propose a Quasi Newton based algorithm in Algorithm 2.1.

Algorithm 2.1 (Minimax Optimization)

1. Initialization. *Select the initial admittance rate* γ_0 *and the error tolerance* τ. *Set* $k = 1$.

2. Min Phase: *For each* γ_k

2.1. Max Phase: *Compute* $g(\gamma_k) = \max_{\omega_L \le \omega \le \omega_H} |y(\omega, \gamma_k)|^2$ *using a grid search followed by a Quasi Newton refinement. Let the optimizer be* (ω_k, γ_k). *Compute the pseudo-gradient* $\frac{\partial g}{\partial \gamma}\Big|_{\gamma=\gamma_k}^{(pseudo)} \triangleq 2\Re\left\{ y^*(\omega_k, \gamma_k) \frac{\partial y}{\partial \gamma}\Big|_{\substack{\omega=\omega_k \\ \gamma=\gamma_k}} \right\}$.

2.2. Update: *Use the function value and pseudo-gradient of* $g(\gamma_k)$ *and* $g(\gamma_{k-1})$ *to do a Quasi Newton step to get* γ_{k+1}. *Set* $k = k + 1$. *If* $|\gamma_k - \gamma_{k-1}| \le \tau$, *return optimizer as* γ_k *and end the algorithm.*

All Quasi Newton steps in the algorithm above use a backtracking strategy with Armijo condition. In a Max Phase, an FRF may have many local maxima and we are only interested in the global maximum in our application, so we use grid search to try to avoid missing the highest peak and use a Quasi Newton step afterwards to increase the precision. In the Min Phase, however, the function $g(\gamma)$ is convex in our application and a local optimization algorithm suffices. The difficult part about the Min Phase is that $g(\gamma)$ may have kink points as is shown in Fig. 2. The pseudo-gradient equals the gradient at the differentiable points and equals the gradient of one of several intersecting PLMACs at a kink point. The direction obtained is always correct, and around the kink optimizer, the backtracking will terminate when $|\gamma_k - \gamma_{k-1}| \le \tau$.

In Algorithm 2.1, we need to compute y for each step of the Inner Phase, $\frac{\partial y}{\partial \omega}$ for each Quasi Newton step of the Inner Phase, and $\frac{\partial y}{\partial \gamma}$ for each step of the Outer Phase. When n is large, these computations are expensive due to the LU factorization of the large scale matrix $K + i\omega\gamma C - \omega^2 M$. A possible way to reduce the computational cost is MOR. However, if we want to do this efficiently, the gradient should also be computed via the reduced model; otherwise, the large scale matrix is factorized anyway and the acceleration effect is limited. We will introduce MOR in the next section and discuss derivative computations via the reduced model in section 4.

3 Krylov based MOR

3.1 Arnoldi Process on First Order System

Given $A \in \mathbb{C}^{n\times n}$ and $b \in \mathbb{C}^n$, the k-dimensional Krylov subspace is defined as

$$\mathcal{K}_k(A, b) = \text{span}\{b, Ab, A^2 b, \ldots, A^{k-1} b\},$$

where $k \le n$ and in most applications $k \ll n$. Krylov subspace methods are very suitable for large scale problems because only matrix-vector multiplication is required. The *Arnoldi process* (AP) [1] is a numerically stable scheme to generate an orthonormal basis of $\mathcal{K}_k(A, b)$.

First, we consider the simplest case: MOR on the first order linear system

$$\begin{cases} (K - \alpha M)x = b, \\ y = l^* x, \end{cases} \tag{5}$$

where K, $M \in \mathbb{C}^{n\times n}$, K is nonsingular and $b \in \mathbb{C}^n$. Given two matrices W_k, $V_k \in \mathbb{C}^{n\times k}$, we can approximate x with a vector in the subspace colspan$\{V_k\}$, namely $V_k z$ ($z \in \mathbb{C}^k$) and then left multiply the first equation by W_k^* to obtain the reduced model

$$\begin{cases} (\hat{K} - \alpha\hat{M})z = \hat{b}, \\ \hat{y} = \hat{l}^* z, \end{cases} \tag{6}$$

where $\hat{K} = W_k^* K V_k \in \mathbb{C}^{k\times k}$, $\hat{M} = W_k^* M V_k \in \mathbb{C}^{k\times k}$, $\hat{b} = W_k^* b \in \mathbb{C}^k$ and $\hat{l} = V_k^* l \in \mathbb{C}^k$. In this way, the order n system (5) is reduced to the order k system (6) and the remaining problem is how to choose W_k and V_k such that $\hat{y}$ is a good approximation of y. One approach to generate W_k and V_k is using a Krylov method. The key concept of Krylov based MOR is *moment matching*. Let $y = \sum_{i=0}^{\infty} m_i \alpha^i$ be the Taylor expansion of y, then m_i is called the i-th moment of y. For system (5), $m_i = l^*(K^{-1}M)^i K^{-1} b$. Theorem 3.1 shows that we can use Krylov methods to generate W_k and V_k with *moment matching property* and $\hat{y}$ is a Padé type approximation of y. So when α is small, the reduced system (6) is a good approximation of the original system (5) [14].

For (5), $\mathcal{K}_k(K^{-1}M, K^{-1}b)$ is called the k-dimensional *right Krylov subspace* and $\mathcal{K}_k(K^{-*}M^*, K^{-*}l)$ is called the k-dimensional *left Krylov subspace*. In a *one-sided method*, we use an AP on $\mathcal{K}_k(K^{-1}M, K^{-1}b)$ (or on $\mathcal{K}_k(K^{-*}M^*, K^{-*}l)$) to get the column vectors of V_k (or W_k), and set $W_k = V_k$ (or $V_k = W_k$). In a *two-sided method*, we use an AP on $\mathcal{K}_k(K^{-1}M, K^{-1}b)$ to get V_k and an AP on $\mathcal{K}_k(K^{-*}M^*, K^{-*}l)$ to get W_k.

Theorem 3.1

1. *Using a one-sided method to reduce* (5)*, the first k moments of y and $\hat{y}$ match if the left or the right Krylov subspace is of order k.*
2. *Using a two-sided method to reduce* (5)*, the first $2k$ moments of y and $\hat{y}$ match if both the left and the right Krylov subspaces are of order k.*

3.2 SOAR

Consider the second order system in the Max Phase optimization (3):

$$\begin{cases} (K + i\omega\gamma_0 C - \omega^2 M)x = f, \\ y = l^* x. \end{cases} \tag{7}$$

A straightforward method to reduce this system is to use the AP for an equivalent first order system [4], but the reduced model does not preserve the second order structure, which is often regarded as a disadvantage.

A solution to this problem is the SOAR method. SOAR builds the k-th *Second Order Krylov Subspace* that contains the k-th Krylov subspace generated by AP and thus inherits its moment matching properties. The k-th left (right) second order Krylov subspace is defined as $\mathcal{K}_k(-i\gamma_0 K^{-*}C^*, K^{-*}M^*, K^{-*}l)$ $(\mathcal{K}_k(-i\gamma_0 K^{-1}C, K^{-1}M, K^{-1}f))$ [14], where $\mathcal{K}_k(A, B, b) = \{p_1, p_2, \ldots, p_k\}$, $p_1 = b$, $p_2 = Ab$, and $p_i = Ap_{i-1} + Bp_{i-2}$ $(3 \le i \le k)$.

Instead of reducing the equivalent first order model, SOAR directly reduces the second order system (7) to get the reduced system

$$\begin{cases} (\hat{K} + i\omega\gamma_0 \hat{C} - \omega^2 \hat{M})z = \hat{f}, \\ \hat{y} = \hat{l}^* z, \end{cases} \tag{8}$$

where $[\hat{K}, \hat{C}, \hat{M}] = W^*[K, C, M]V$, $\hat{f} = W^* f$, $\hat{l} = V^* l$ and $W, V \in \mathbb{C}^{n\times k}$ contain the base vectors of the subspaces. If either colspan$\{W\}$ contains the left second order Krylov subspace or colspan$\{V\}$ contains the right second order Krylov subspace, the SOAR method is called *"one-sided"*, and if both these conditions are true, the SOAR method is called *"two-sided"*. In one-sided SOAR, it is common practice to set $W = V$. We summarize some important properties of SOAR in Theorem 3.2. See [4, 3, 2, 14] for more details.

Theorem 3.2 *Let* (8) *be the SOAR reduced model of the system* (7)*, we have the following moment matching result for SOAR:*

- *If one-sided SOAR is used and the dimension of the right (or left) second order Krylov space is k, then the first k moments of y and $\hat{y}$ match;*
- *If two-sided SOAR is used and the dimensions of both second order Krylov subspaces are k, then the first $2k$ moments of y and $\hat{y}$ match.*

3.3 PIMTAP

PMOR is a natural extension of MOR on one variable to accelerate the solution of parameterized linear systems. Pioneering work includes [16, 7, 15, 5], etc. Here we concentrate on PIMTAP method. Consider the following system

$$\begin{cases} \big(G_0 + \gamma G_1 + s(C_0 + \gamma C_1)\big)x = b, \\ \hfill y = l^* x, \end{cases} \tag{9}$$

where $G_0, G_1, C_0, C_1 \in \mathbb{C}^{n\times n}$ and $b, l \in \mathbb{C}^n$. Let the Taylor series of x in terms of s and γ be $x = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} r_i^j s^i \gamma^j$ $(r_i^j \in \mathbb{C}^n)$ and we define r_i^j as the (i, j)-th 2-parameter moment of x. The idea of Krylov based PMOR is to match the low order 2-parameter moments.

PIMTAP [10, 9, 8] provides a flexible, systematic and numerically stable way for reducing linear systems with multiple parameters. It defines

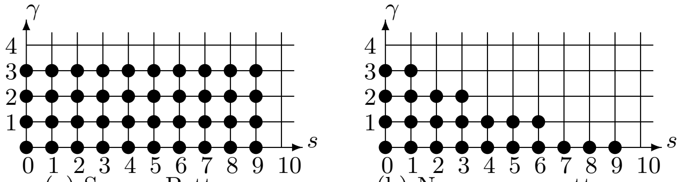
$$r_{[i]}^{[j]} = \begin{bmatrix} r_{i-1}^0 \\ r_{i-1}^1 \\ \vdots \\ r_{i-1}^{j-1} \end{bmatrix}, G_{[j]} = \underbrace{\begin{bmatrix} G_0 & & & & \\ G_1 & G_0 & & & \\ & G_1 & G_0 & & \\ & & \ddots & \ddots & \\ & & & G_1 & G_0 \end{bmatrix}}_{j \text{ blocks}}, C_{[j]} = \underbrace{\begin{bmatrix} C_0 & & & & \\ C_1 & C_0 & & & \\ & C_1 & C_0 & & \\ & & \ddots & \ddots & \\ & & & C_1 & C_0 \end{bmatrix}}_{j \text{ blocks}},$$

and finds the recursive relationship

$$r_{[i]}^{[j]} = -G_{[j]}^{-1} C_{[j]} r_{[i-1]}^{[j]}, \quad \text{for all } i > 1. \tag{10}$$

It is clear that $r_{[1]}^{[j]}, r_{[2]}^{[j]}, \ldots, r_{[k]}^{[j]}$ span the Krylov subspace $\mathcal{K}_k\left(-G_{[j]}^{-1}C_{[j]}, r_{[1]}^{[j]}\right)$ and we can use AP to generate its base vectors in a numerically stable way.

This method can generate rectangle *moment matching patterns* as is shown in Fig 3(a), in which a solid circle in (i,j) means the moment r_i^j is to be matched.



(a) Square Pattern (b) Non-square pattern.

Fig. 3. PIMTAP moment matching patterns.

In some applications, the high order cross-term moments are not so important and moment matching patterns like Fig. 3(b) are wanted. In the example of Fig. 3(b), we can build $\mathcal{K}_{10}\left(-G_{[1]}^{-1}C_{[1]}, r_{[1]}^{[1]}\right)$, $\mathcal{K}_7\left(-G_{[2]}^{-1}C_{[2]}, r_{[1]}^{[2]}\right)$, $\mathcal{K}_4\left(-G_{[3]}^{-1}C_{[3]}, r_{[1]}^{[3]}\right)$ and $\mathcal{K}_2\left(-G_{[4]}^{-1}C_{[4]}, r_{[1]}^{[4]}\right)$ to get the projection matrix V. PIMTAP recycles the moments that are already computed and thus avoids recomputation. See [10, 9, 8] for more details about PIMTAP.

After we obtain V from PIMTAP, we can project the system (9) on the subspace colspan$\{V\}$ to get the reduced model

$$\begin{cases} \left(\hat{G}_0 + \gamma\hat{G}_1 + s(\hat{C}_0 + \gamma\hat{C}_1)\right)z = \hat{b}, \\ \hfill \hat{y} = \hat{l}^* z, \end{cases} \tag{11}$$

where $[\hat{G}_0, \hat{G}_1, \hat{C}_0, \hat{C}_1] = V^*[G_0, G_1, C_0, C_1]V$ and $[\hat{b}, \hat{l}] = V^*[b, l]$. After this reduction, the moments specified by the moment matching pattern will be matched for y and $\hat{y}$.

4 Derivative Computation via the Reduced Model

The analysis in section 2 shows that computing derivatives via the reduced model is crucial in using MOR to improve the performance of the Minimax Algorithm 2.1. In this section, we analyze the moment matching properties for derivatives in two-sided SOAR and PIMTAP, which means that computing derivatives via the reduced model is feasible.

4.1 Computation of Derivatives w.r.t Free Variables

Theorem 4.1 *Let y and $\hat{y}$ be the output of the original system and the reduced system respectively, and $s_1, s_2, \ldots, s_l$ be l free parameters in both systems. Let*

$$y = \sum_{i_1=0}^{\infty}\sum_{i_2=0}^{\infty}\cdots\sum_{i_l=0}^{\infty} m(i_1, i_2, \ldots, i_l) s_1^{i_1} s_2^{i_2} \ldots s_l^{i_l},$$

$$\hat{y} = \sum_{i_1=0}^{\infty}\sum_{i_2=0}^{\infty}\cdots\sum_{i_l=0}^{\infty}\hat{m}(i_1,i_2,\ldots,i_l)s_1^{i_1}s_2^{i_2}\ldots s_l^{i_l}.$$

Then if $i_1 \geq 0$, $i_2 \geq 0$, …, $i_l \geq 0$, the $(i_1,i_2,\ldots,i_l)$-th moments of $\frac{\partial^{r_1+r_2+\cdots+r_l}y}{\partial s_1^{r_1}\partial s_2^{r_2}\partial s_l^{r_l}}$ and $\frac{\partial^{r_1+r_2+\cdots+r_l}\hat{y}}{\partial s_1^{r_1}\partial s_2^{r_2}\partial s_l^{r_l}}$ match iff the $(i_1+r_1,i_2+r_2,\ldots,i_l+r_l)$-th moments of y and $\hat{y}$ match.

From Theorem 4.1, it is clear that in 2-variable case, the (i,j)-th moment of ∇y and $\nabla\hat{y}$ match if the $(i+1,j)$-th and the $(i,j+1)$-th moments of y and $\hat{y}$ match.

4.2 Computation of Derivatives w.r.t Fixed Variables

For SOAR, Theorem 3.2 implies moment matching properties for $\frac{\partial y}{\partial\omega}$, but not for $\frac{\partial y}{\partial\gamma}$ since γ is a fixed variable rather than a free variable in the SOAR reduced model. In this section, we show that we can compute first order derivatives w.r.t fixed variables via the two-sided MOR reduced model with the moment matching property.

For the first order system (5), $\frac{dy}{d\alpha} = l^*(K-\alpha M)^{-1}M(K-\alpha M)^{-1}b$. Since $l^*(K-\alpha M)^{-1}$ can be approximated by the left-Krylov subspace and $(K-\alpha M)^{-1}b$ can be approximated by the right-Krylov subspace, we also has the following moment matching property for computing derivatives.

Theorem 4.2 *In system* (5)*, if both the left Krylov subspace match and the right Krylov subspace are of dimension k, the first k moments of $l^*(K-\alpha M)^{-1}A(K-\alpha M)^{-1}b$ and $\hat{l}^*(\hat{K}-\alpha\hat{M})^{-1}\hat{A}(\hat{K}-\alpha\hat{M})^{-1}\hat{b}$ match, where A is an arbitrary matrix and $\hat{A} = W^*AV$. If $A = \beta M$ ($\beta \in \mathbb{C}$), the first $2k-1$ moments match.*

The two-sided SOAR methods inherit the moment matching properties from two-sided AP.

Corollary 4.3 *For two-sided SOAR, if both the 2nd-order left-Krylov subspace and the 2nd-order right-Krylov subspace are of dimension k, the first $2k$ moments of y and $\hat{y}$ match, the first $2k-1$ moments of $\frac{\partial y}{\partial\omega}$ and $\frac{\partial\hat{y}}{\partial\omega}$ match and the first k moments of $\frac{\partial y}{\partial\gamma}$ and $\frac{\partial\hat{y}}{\partial\gamma}$ match.*

5 Numerical Results

Now we apply MOR in solving the minimax optimization problem 2. Both two-sided SOAR and PIMTAP reduced models can compute y, $\frac{\partial y}{\partial\omega}$ and $\frac{\partial\hat{y}}{\partial\gamma}$. Two-sided SOAR can be directly used in the Max Phase, and to use PIMTAP, we do the following substitution: $G_0 \leftarrow K$, $G_1 \leftarrow C$, $C_0 \leftarrow M$, $C_1 \leftarrow 0$, $\gamma \leftarrow i\omega\gamma$, $s \leftarrow -\omega^2$ and $b \leftarrow f$. The advantage of PIMTAP is that it can be used for several Max Phases. We compare three cases here: optimization with the original model, with one two-sided SOAR reduced model for each Max Phase, and with a PIMTAP reduced model for the whole Min Phase.

In the first example, we apply MOR to a system of order 15625. The numerical results in Table 1 show that MOR indeed drastically reduces the optimization time as we expect. In the second example, the system order is 216 and the numerical results in Table 2 show that MOR is effective even when the system order is relatively small. In both examples, the grid search interval is set to 0.01, backtracking factor equals 0.5, $\gamma_0 = 0.3$ and $\tau = 10^{-4}$. In both examples, the original models are very accurately approximated by the small order reduced system. How to choose the order of the reduced model is an open problem, but if the eigenvalues of the system are clustered away from zero, we can expect a low order reduced model to work well [11].

Table 1. Numerical Results for Example 1

	Direct method	Two-sided SOAR	PIMTAP
Matrix size	15625	50	61
Optimizer computed	(10.1245, 0.2671)	(10.1245, 0.2671)	(10.1245, 0.2671)
CPU time	3508s	128s	22s

Table 2. Numerical Results for Example 2

	Direct method	Two-sided SOAR	PIMTAP
Matrix size	216	35	44
Optimizer computed	(8.668, 0.2687)	(8.669, 0.2687)	(8.668, 0.2688)
CPU time	162s	27s	18s

6 Conclusions

In this paper, we have introduced MOR to a large scale minimax optimization problem that is computationally very expensive. We show that derivative computations, a key issue in many optimization algorithms, can also be computed with the reduced model with moment matching property. As both function values and the derivatives can be computed via the reduced model, the original large-scale model is no longer explicitly involved in the optimization, and the optimization time can be drastically reduced. Numerical results show that both SOAR and PIMTAP are effective in reducing optimization time, and PIMTAP is more efficient in our application.

AcknowledgmentsThis paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme,initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its author(s).

References

1. W. E. Arnoldi. The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quarterly of Applied Mathematics*, 9(17):17–29, 1951.

2. Z. Bai, K. Meerbergen, and Y. Su. Arnoldi methods for structure-preserving dimension reduction of second-order dynamical systems. In *Dimension reduction of large-scale systems*, pages 173–189. Springer, 2005.
3. Z. Bai and Y. Su. Dimension reduction of large-scale second-order dynamical systems via a second-order Arnoldi method. *SIAM Journal on Scientific Computing*, 26(5):1692–1709, 2005.
4. Z. Bai and Y. Su. SOAR: a second-order Arnoldi method for the solution of the quadratic eigenvalue problem. *SIAM Journal on Matrix Analysis and Applications*, 26(3):640–659, 2005.
5. L. Daniel, O. C. Siong, L. S. Chay, K. H. Lee, and J. White. A multiparameter moment-matching model-reduction approach for generating geometrically parameterized interconnect performance models. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 23(5):678–693, 2004.
6. P. Feldman and R. W. Freund. Efficient linear circuit analysis by Padé approximation via the Lanczos process. *IEEE Trans. Computer-Aided Design*, 14(5):639–649, 1995.
7. P. Gunupudi and M. Nakhla. Multi-dimensional model reduction of VLSI interconnects. In *Custom Integrated Circuits Conference, 2000. Proceedings of the IEEE 2000*, pages 499–502, 2000.
8. Y.-T. Li, Z. Bai, and Y. Su. A two-directional Arnoldi process and its application to parametric model order reduction. *Journal of Computational and Applied Mathematics*, 226(1):10–21, 2009.
9. Y.-T. Li, Z. Bai, Y. Su, and X. Zeng. Parameterized model order reduction via a two-directional Arnoldi process. In *Proceedings of the 2007 IEEE/ACM international conference on Computer-aided design*, pages 868–873, 2007.
10. Y.-T. Li, Z. Bai, Y. Su, and X. Zeng. Model order reduction of parameterized interconnect networks via a Two-Directional Arnoldi process. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 27(9):1571–1582, 2008.
11. K. Meerbergen. The solution of parametrized symmetric linear systems. *SIAM journal on matrix analysis and applications*, 24(4):1038–1059, 2003.
12. K. Meerbergen. Fast frequency response computation for Rayleigh damping. *International Journal for Numerical Methods in Engineering*, 73(1):96–106, 2008.
13. A. Odabasioglu, M. Celik, and L. T. Pileggi. PRIMA: passive reduced-order interconnect macromodeling algorithm. In *ICCAD '97: Proceedings of the 1997 IEEE/ACM international conference on Computer-aided design*, pages 58–65, Washington, DC, USA, 1997. IEEE Computer Society.
14. B. Salimbahrami and B. Lohmann. Order reduction of large scale second-order systems using Krylov subspace methods. *Linear Algebra and its Applications*, 415(2-3):385–405, 2006.
15. D. S. Weile, E. Michielssen, and K. Gallivan. Reduced-order modeling of multiscreen frequency-selective surfaces using krylov-based rational interpolation. *IEEE transactions on antennas and propagation*, 49(5):801–813.
16. D. S. Weile, E. Michielssen, and G. K. Grimme, E. and. A method for generating rational interpolant reduced order models of two-parameter linear systems. *Applied Mathematics Letters*, 12(5):93–102, 1999.

Part III

Optimization on Manifolds

Optimization On Manifolds: Methods and Applications

P.-A. Absil[1], R. Mahony[2], and R. Sepulchre[3]

[1] Department of Mathematical Engineering, Université catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium http://www.inma.ucl.ac.be/~absil/
[2] Australian National University, ACT, 0200, Australia
[3] Department of Electrical Engineering and Computer Science, Université de Liège, Belgium

Summary. This paper provides an introduction to the topic of optimization on manifolds. The approach taken uses the language of differential geometry, however, we choose to emphasise the intuition of the concepts and the structures that are important in generating practical numerical algorithms rather than the technical details of the formulation. There are a number of algorithms that can be applied to solve such problems and we discuss the steepest descent and Newton's method in some detail as well as referencing the more important of the other approaches. There are a wide range of potential applications that we are aware of, and we briefly discuss these applications, as well as explaining one or two in more detail.

1 Introduction

This paper is written as an invitation for the reader to the area of optimization on manifolds. It follows quite closely the structure of the plenary talk given by the first author at the 14th Belgian-French-German Conference on Optimization, Leuven, 14–18 September 2009. The style is rather on the informal side, and there is a definite bias towards the exposition given in the monograph [5], to which we refer for more details and for a larger bibliography. When we cite [5], we do not imply that it is the original reference for the topic in question and we refer the reader to the "Notes and References" sections of [5] for details of the history.

The general problem of optimization on manifolds is introduced in Section 2. A motivation for considering the problem in its most abstract form is given in Section 3. Manifolds are defined in more technical terms in Section 4. Several specific manifolds are presented in Section 5, along with pointers to applications where they are involved. Section 6 describes a steepest-descent optimization scheme on Riemannian manifolds. Its application to a simple problem is worked out in Section 7. Section 8 is dedicated to Newton's

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_12, © Springer-Verlag Berlin Heidelberg 2010

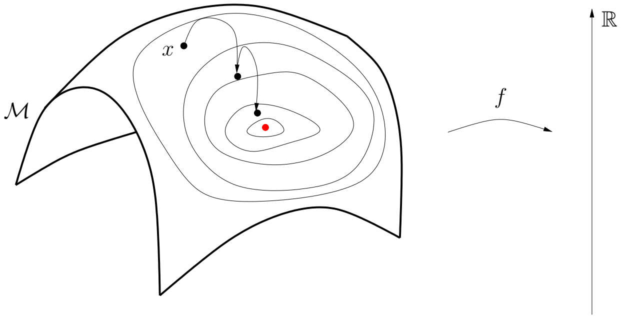


Fig. 1. Optimization on manifolds in one picture.

method on Riemannian manifolds. Other optimization methods on manifolds are briefly discussed in Section 9. Section 10 provides some conclusions.

2 Optimization on manifolds in one picture

The archetypal problem in optimization on manifolds is pictured in Figure 1. The set of feasible points is a manifold $\mathcal{M}$ that, for the sake of developing intuition, can be viewed as a smooth surface. We will argue in Section 3, however, that it is beneficial to depart from this restrictive representation. Anticipating Section 4, one can think of $\mathcal{M}$ as a collection of points, endowed with a yet-to-be-defined *manifold structure* that turns $\mathcal{M}$ into a topological set—so we can talk about neighborhoods—and that makes it possible to declare whether a real-valued function on $\mathcal{M}$ is smooth or not. The reader who cannot wait to get a more precise definition of the concept of a manifold is invited to take a peek at Section 4. It may also be reassuring to have a look at the list of specific manifolds in Section 5.

The smooth real-valued function f on the set $\mathcal{M}$ that defines the goal of the optimization problem is termed the *objective function.* A few of its level curves $f^{-1}(c)$, $c \in \mathbb{R}$, are represented in Figure 1. The dot inside the level curves is an optimal point of f, say, a minimizer of f.

Computing minimizers of f is our goal. More precisely, the problem is as follows:

Problem 1 (optimization on manifolds).
Given: a manifold $\mathcal{M}$ and a smooth function $f : \mathcal{M} \to \mathbb{R}$.
Sought: an element x_* of $\mathcal{M}$ such that there is a neighborhood V of x_* in $\mathcal{M}$ with $f(x_*) \leq f(x)$ for all $x \in V$.

Such an x_* is termed a *local minimizer* of f.

The methods we are interested in for solving Problem 1 are iterative algorithms on the manifold $\mathcal{M}$. Given a starting point $x_0 \in \mathcal{M}$, such an algorithm produces a sequence $(x_k)_{k\geq 0}$ in $\mathcal{M}$ that converges to x_* whenever x_0 is in a certain neighborhood, or basin of attraction, of x_*. As in classical optimization algorithms, the following properties are desirable: (i) the set of points x_0 for which convergence to x_* occurs should be large; (ii) convergence to x_* should be fast; (iii) the numerical effort required to compute each new iterate should be reasonable.

3 Why consider general manifolds?

A motivation for considering general manifolds—and not only manifolds that come to us as subsets of Euclidean spaces—is that they offer an adequate common framework for dealing with the following two problems.

Problem 2. Given a matrix $A = A^T \in \mathbb{R}^{n\times n}$ and a diagonal $n \times n$ matrix $N = \mathrm{diag}(1, \dots, p)$ with $p \leq n$, solve

$$\begin{aligned}\min f(X) &= \mathrm{trace}(X^T AXN)\\ \text{subj. to } X &\in \mathbb{R}^{n\times p}, X^T X = I.\end{aligned}$$

Solving this problem yields the p "leftmost" eigenvectors of A, i.e., those associated with the p algebraically smallest eigenvalues of A; see Section 7 or [5, §4.8] for details.

The optimization domain in Problem 2 is the set

$$\mathrm{St}(p, n) = \{X \in \mathbb{R}^{n\times p} : X^T X = I\}, \tag{1}$$

which is a subset of the Euclidean space $\mathbb{R}^{n\times p}$. If a subset of a Euclidean space can be locally smoothly straightened—the *submanifold property*—, then it admits one and only one "natural" manifold structure [5, Prop. 3.3.2]; see Figure 2 for an illustration. The set $\mathrm{St}(p, n)$ happens to be such a subset [5, §3.3.2]. Endowed with its natural manifold structure, $\mathrm{St}(p, n)$ is termed the *Stiefel manifold* of orthonormal p-frames in $\mathbb{R}^n$.[4]

Problem 3. Given a matrix $A = A^T \in \mathbb{R}^{n\times n}$, solve

$$\begin{aligned}\min f(Y) &= \mathrm{trace}\left((Y^T Y)^{-1} Y^T AY\right)\\ \text{subj. to } Y &\in \mathbb{R}^{n\times p}_*,\end{aligned}$$

where $\mathbb{R}^{n\times p}_*$ denotes the set of all full-rank $n \times p$ matrices ($p < n$).

[4] The Stiefel manifold is named in honor of Eduard L. Stiefel who studied its topology in [55]. Stiefel is perhaps better known for proposing with M. R. Hestenes the conjugate gradient method [25]. Incidentally, he was born 100 years ago.

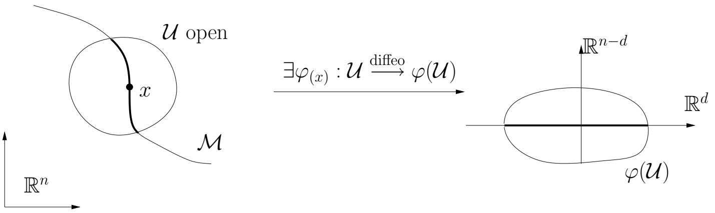


Fig. 2. The *set* $\mathcal{M} \subset \mathbb{R}^n$ is termed a *submanifold* of $\mathbb{R}^n$ if the situation described above holds for all $x \in \mathcal{M}$. Charts for $\mathcal{M}$ are obtained by extracting the d first coordinates.

The function f in Problem 3 has the following invariance property:

$$f(YM) = f(Y), \quad \text{for all } Y \in \mathbb{R}_*^{n\times p} \text{ and all } M \in \mathbb{R}_*^{p\times p}. \tag{2}$$

In other words, f is constant on each equivalence class

$$[Y] = \{YM : M \in \mathbb{R}_*^{p\times p}\}, \tag{3}$$

$Y \in \mathbb{R}_*^{n\times p}$. The equivalence class $[Y]$ is precisely the set of all $n \times p$ matrices that have the same column space as Y, and Y is a minimizer of f if and only if the column space of Y is a p-dimensional minor eigenspace of A (i.e., associated with the smallest eigenvalues); see [5, §2.1.1]. It is thus tempting to reconsider Problem 3 on a search space whose elements are the equivalence classes $[Y]$, $Y \in \mathbb{R}_*^{n\times p}$, and optimize the function

$$\check{f} : \{[Y] : Y \in \mathbb{R}_*^{n\times p}\} \to \mathbb{R} : [Y] \mapsto f(Y), \tag{4}$$

which is well defined in view of (2) and (3).

A major advantage of this reformulation of Problem 3 is that, for generic A, the minimizers of $\check{f}$ are isolated, while they are never isolated in the original formulation in view of the invariance property (2). The apparent downside is that the new search space, the quotient space

$$\mathrm{Gr}(p, n) = \{[Y] : Y \in \mathbb{R}_*^{n\times p}\}, \tag{5}$$

is no longer a Euclidean space. However, it turns out (see [5, §3.4.4]) that (5) admits one and only one "natural" manifold structure, which is inherited from the fact that, around every element of $\mathbb{R}_*^{n\times p}$, the bundle of equivalence classes can be smoothly straightened; see Figure 3 for an illustration. Endowed with this natural manifold structure, the set $\mathrm{Gr}(p, n)$ is termed the *Grassmann manifold*[5] of p-planes in $\mathbb{R}^{n\times p}$. (The set (5) is identified with the set of all p-dimensional subspaces of $\mathbb{R}^n$ because $[Y]$ is the set of all $n \times p$ matrices whose

[5] The Grassmann manifold is named in honor of Hermann Günther Graßmann who proposed a representation of the manifold known as *Plücker coordinates*. Graßmann is perhaps better known for his Sanskrit dictionary and his translation of the Rgveda [21]. Incidentally, he was born 200 years ago.

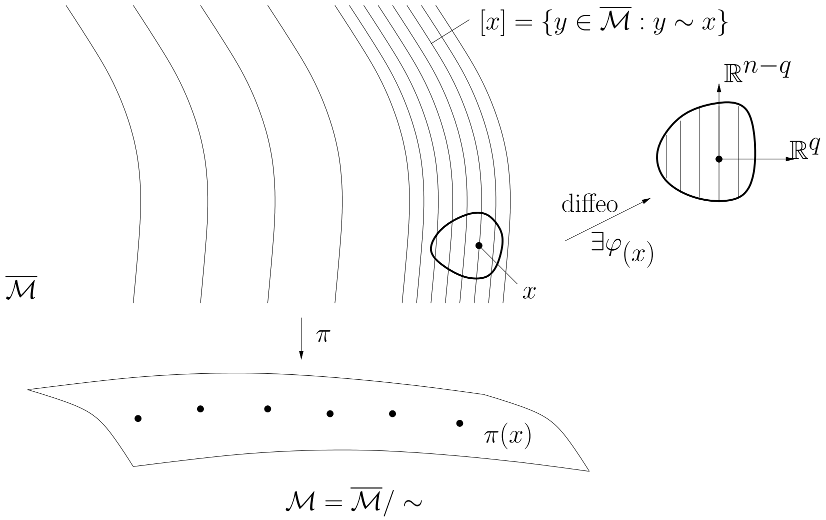


Fig. 3. The set $\overline{\mathcal{M}}/\sim := \{[x] : x \in \overline{\mathcal{M}}\}$ is termed a *quotient manifold* if the situation described above holds for all $x \in \overline{\mathcal{M}}$. Charts for $\overline{\mathcal{M}}/\sim$ are obtained by extracting the q first coordinates.

columns form a basis of the same p-dimensional subspace of $\mathbb{R}^n$.) Dealing with optimization problems such as the minimization of (4) is precisely what optimization on (quotient) manifolds is all about.

In summary, Problem 2 and the reformulated Problem 3 have the following properties in common: (i) their search space admits a natural manifold structure; (ii) in the sense of the manifold structure, the objective function is smooth, as a consequence of [33, Prop. 8.22] and [33, Prop. 7.17]. In the next section, we explain more technically what a manifold structure is, and what it means for a objective function on a manifold to be smooth.

4 Manifolds and smooth objective functions

The time has come to give an informal, application-driven, definition of a manifold structure. Details can be found in [5, §3.1.1] or in any textbook on differential geometry.

The intuition can be obtained from Figure 4. We are given a set $\mathcal{M}$, which initially is just a collection of points without any particular structure, and we are given a real-valued function f on the set $\mathcal{M}$. Since $\mathcal{M}$ does not have a vector space structure, the classical definition of differentiability of a function $f : \mathcal{M} \to \mathbb{R}$ at a point $x \in \mathcal{M}$ does not apply. The remedy is to consider a one-to-one correspondence φ between a subset $\mathcal{U}$ of $\mathcal{M}$ containing x and an open subset $\varphi(\mathcal{U})$ of some $\mathbb{R}^d$. Then f is declared to be differentiable at $x \in \mathcal{M}$ when

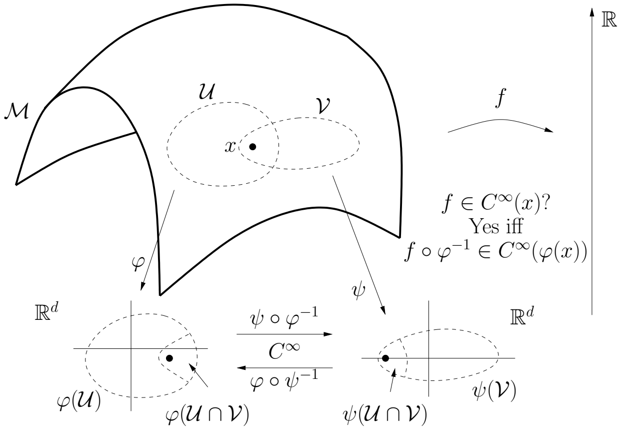


Fig. 4. Manifold structures and smoothness of objective functions.

the function $f \circ \varphi^{-1} : \varphi(\mathcal{U}) \to \mathbb{R} : y \mapsto f(\varphi^{-1}(y))$ is differentiable at $\varphi(x)$. Since $\varphi(\mathcal{U})$ is an open subset of $\mathbb{R}^d$, the usual definition of differentiability applies.

For this procedure to be applicable to every point of the set $\mathcal{M}$, we need to provide a collection of φ's such that the union of their domains is the whole set $\mathcal{M}$. Moreover, whenever the domains $\mathcal{U}$ and $\mathcal{V}$ of two correspondences φ and ψ overlap on a point $x \in \mathcal{M}$, we must require that, for all $f : \mathcal{M} \to \mathbb{R}$, $f \circ \varphi^{-1}$ is differentiable at $\varphi(x)$ if and only if $f \circ \psi^{-1}$ is differentiable at $\psi(x)$; otherwise differentiability of f at x is not well defined. This goal is achieved by imposing that the charts *overlap smoothly*, i.e., $\psi \circ \varphi^{-1}$ is a diffeomorphism—a smooth bijection with smooth inverse—between $\varphi(\mathcal{U} \cap \mathcal{V})$ and $\psi(\mathcal{U} \cap \mathcal{V})$. The collection of correspondences is then called an *atlas*, and the correspondences are called *charts*. The *maximal atlas* generated by an atlas is the collection of all charts that overlap smoothly with those of the given atlas. Finally, a *manifold* is a pair $(\mathcal{M}, \mathcal{A}^+)$, where $\mathcal{M}$ is a set and $\mathcal{A}^+$ is a maximal atlas on the set $\mathcal{M}$. In other words, a maximal atlas uniquely specifies a manifold structure on $\mathcal{M}$. For brevity, it is common to say "the manifold $\mathcal{M}$" when the maximal atlas is clear from the context or irrelevant.

Let us work out an example. When $p = 1$ and $n = 2$, the Stiefel manifold (1) reduces to the unit circle in $\mathbb{R}^2$. Let $\mathcal{U} = \mathrm{St}(1,2) \setminus \{(0,1),(0,-1)\}$, $\varphi : \mathcal{U} \to \mathbb{R} : x \mapsto x_2/x_1$, $\mathcal{V} = \mathrm{St}(1,2) \setminus \{(1,0),(-1,0)\}$, $\psi : \mathcal{V} \to \mathbb{R} : x \mapsto x_1/x_2$. Then $\{\varphi, \psi\}$ is an atlas of the set $\mathrm{St}(1,2)$. Moreover, it can be shown that this atlas induces the natural manifold structure mentioned in the previous section.

Let us show that the objective function f defined in Problem 2 is smooth. To this end, pick $x \in \mathrm{St}(1,2)$, and assume that $x \in \mathcal{U} \cap \mathcal{V}$. Observe that $\varphi^{-1}(y) = \frac{1}{\sqrt{1+y^2}} \begin{bmatrix} 1 & y \end{bmatrix}^T$ for all $y \in \mathbb{R}$. Hence $f \circ \varphi^{-1}(y) = \frac{1}{\sqrt{1+y^2}} \begin{bmatrix} 1 & y \end{bmatrix} A \begin{bmatrix} 1 \\ y \end{bmatrix}$, and we see that f is a smooth function on $\mathcal{U}$. A similar reasoning shows that f

is a smooth function on $\mathcal{V}$. Hence f is smooth on the whole manifold $\mathcal{M}$. (An alternate way of obtaining this result is by invoking the fact [33, Prop. 8.22] that the restriction of a smooth function to a submanifold is smooth.)

Looking back at the original Problem 1, we see that all the concepts involved therein are now well defined, except for "neighborhood". The notion of neighborhood in $\mathcal{M}$ is directly inherited from its manifold structure: a *neighborhood* of a point x in a manifold $\mathcal{M}$ is a subset of $\mathcal{M}$ that contains a set of the form $\varphi^{-1}(\Omega)$, where φ is a chart of the manifold $\mathcal{M}$ whose domain contains x and Ω is an open subset that contains $\varphi(x)$.

If all the charts of the maximal atlas are into the same $\mathbb{R}^d$, then d is called the *dimension* of the manifold. In particular, when the manifold is connected, its dimension is well defined.

Finally, we point out that the notion of smoothness extends to functions between two manifolds: the definition relies on expressing the function in charts and checking whether this expression is smooth. Note also that the Cartesian product of two manifolds admits a manifold structure in a natural way.

5 Specific manifolds, and where they appear

In this section, we present a few specific manifolds, and we discuss their use in science and engineering applications.

5.1 Stiefel manifold

The (compact) Stiefel manifold $\mathrm{St}(p,n)$ is the set of all p-tuples $(x_1,\ldots,x_p)$ of orthonormal vectors in $\mathbb{R}^n$. The notation $V_{n,p}$ or $V_p(\mathbb{R}^n)$ is also frequently encountered in the literature.

If we view $\mathbb{R}^n$ as the space of length-n column vectors and turn the p-tuples into $n \times p$ matrices,

$$(x_1,\ldots,x_p) \mapsto \begin{bmatrix} x_1 \cdots x_p \end{bmatrix},$$

we obtain the definition (1), i.e.,

$$\mathrm{St}(p,n) = \{X \in \mathbb{R}^{n\times p} : X^T X = I\}.$$

To relate this definition with the illustration in Figure 1, imagine that each point of $\mathcal{M}$ stands for an orthonormal p-frame $(x_1,\ldots,x_p)$, and that the objective function f assigns a real value to each orthonormal p-frame. We have already encountered such an f in Problem 2.

Here are a few domains of application for optimization methods on the Stiefel manifold, along with related references, which are by no means exhaustive: principal component analysis and the singular value decomposition [24, 8]; independent component analysis and the related problem of joint diagonalization of matrices [8, 42, 26, 56]; more generally, several applications related to machine learning [46, 57, 15]; Procrustes problems [20, 38]; computer vision [34, 59]; Lyapunov exponent computation for dynamical systems [14].

5.2 Sphere

When $p = 1$, the Stiefel manifold $\mathrm{St}(p, n)$ reduces to the unit sphere S^{n-1}, a particularly simple nonlinear manifold.

5.3 Orthogonal group

When $p = n$, the Stiefel manifold $\mathrm{St}(p, n)$ admits a group structure, where the group operation is the matrix product. This group is termed the *orthogonal group*, often denoted by O_n or $\mathrm{O}(n)$. Moreover, the group operation and its inverse are smooth in the sense of the manifold structure of $\mathrm{St}(p, n)$. This makes O_n a *Lie group*. For more information on Lie groups at an introductory level, see, e.g., [62].

The orthogonal group O_n has two connected components. The component that contains the identity matrix is called the *special orthogonal group* $\mathrm{SO}(n)$. The set $\mathrm{SO}(3)$ corresponds to the set of rotations.

5.4 Grassmann manifold

The Grassmann manifold $\mathrm{Gr}(p, n)$ is the set of all p-dimensional subspaces of $\mathbb{R}^n$. Most applications bear some relation with dimensionality reduction: [24, 8, 40, 53, 4, 39, 54, 12, 23, 52, 59, 15, 27].

5.5 Set of fixed-rank positive-semidefinite matrices

The differential geometry of the set

$$\mathrm{S}_+(p, n) = \{X \in \mathbb{R}^{n \times n} : X \succeq 0, \mathrm{rk}(X) = p\}$$

is a topic of interest, in view of its application in rank reduction of positive-definite matrices [13, 30, 60, 61].

5.6 Shape manifold

A quotient geometry arises because the notion of shape is invariant by rotation and by reparameterization; see, e.g., [31, 32, 29].

5.7 Oblique manifold and products of spheres

The oblique manifold $\{Y \in \mathbb{R}_*^{n \times p} : \mathrm{diag}(YY^T) = I_p\}$—where $\mathrm{diag}(YY^T) = I_p$ means that the rows of Y belong to the unit sphere—and Cartesian products of spheres appear, e.g., in the oblique Procrustes problem [58], in nonorthogonal joint diagonalization [3], and in time-varying system identification [49].

5.8 Flag manifold

Given $0 < p_1 < \ldots < p_k$, the flag manifold of type $(p_1, \ldots, p_k)$ is the collection of all k-tuples of linear subspaces of $\mathbb{R}^{p_k}$ $(V_1, \ldots, V_k)$ with $\dim(V_i) = p_i$ and V_i subspace of V_{i+1}. Flag manifolds are useful in the analysis of eigenvalue methods [9, 28] and in independent subspace analysis [47].

5.9 Essential manifold

An essential matrix is the product $E = \Omega R$ of a skew-symmetric matrix Ω and a rotation matrix R. The essential manifold appears in stereo vision processing [41, 22].

5.10 Other products of manifolds

Various Cartesian products of manifolds appear in applications. For example, the Euclidean group SE(3), an important manifold in computer vision and robotics, can be identified with $\mathrm{SO}(3) \times \mathbb{R}^3$. A product of 16 copies of SO(3) was used in [7] to specify the position of a human spine.

The next step is to consider products of infinitely many copies of a manifold, which brings us to curve fitting on manifolds; see [50] and references therein. See also [51] where the problem consists in finding a curve in the Euclidean group SE(3).

5.11 Other quotient manifolds

Quotient manifolds appear in several applications where the objective function has an invariance property that induces a regular equivalence relation; a characterization of regular equivalence relations can be found in [5, Prop. 3.4.2]. In fact, most of the manifolds above admit well-known quotient representations. For example, $\mathrm{St}(p, n)$ can be identified with $\mathrm{O}(n)/\mathrm{O}(n-p)$; see [18] for details.

6 Steepest descent: from $\mathbb{R}^n$ to manifolds

We now turn to optimization algorithms on manifolds. Amongst optimization methods on manifolds that exploit the smoothness of the cost function, the steepest-descent scheme is arguably the most basic.

The next table, where $\boldsymbol{\nabla} f(x) = \left[\partial_1 f(x) \cdots \partial_n f(x)\right]^T$ denotes the classical Euclidean gradient, sketches a comparison between steepest-descent in $\mathbb{R}^n$ and its generalization to manifolds. An illustration is given in Figure 5.

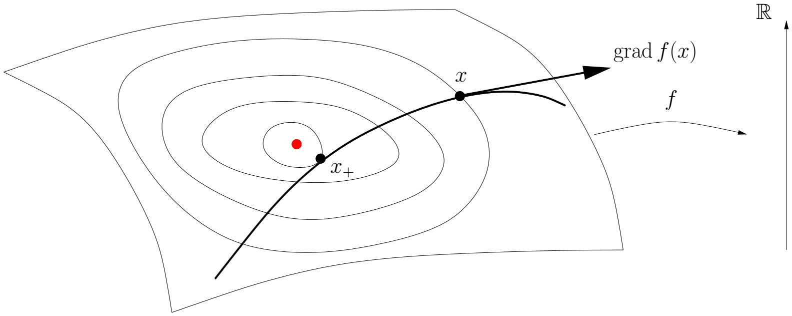


Fig. 5. Steepest descent on Riemannian manifolds.

	$\mathbb{R}^n$	Manifold
Search direction	Vector at x	Tangent vector at x
Steepest-desc. dir.	$-\boldsymbol{\nabla} f(x)$	$-\mathrm{grad}\, f(x)$
Search curve	$\gamma : t \mapsto x - t\,\boldsymbol{\nabla} f(x)$	γ s.t. $\gamma(0) = x$ and $\dot\gamma(0) = -\mathrm{grad}\, f(x)$

Figure 5 corresponds to a submanifold of a Euclidean space. However, we are interested in a theory that subsumes both submanifolds and quotient manifolds, for which we will need definitions of tangent vectors and gradients that are rather abstract. Nevertheless, the reader is invited to keep Figure 5 in mind, because it helps in developing the intuition.

The particularization of the abstract steepest-descent scheme to submanifolds of Euclidean spaces is rather simple and will be covered in this paper. For quotient manifolds, the situation is a bit more complicated, and we refer to [5] for details.

6.1 Tangent vectors and tangent spaces

The notion of a tangent vector at a point $x \in \mathcal{M}$ is intuitively clear when $\mathcal{M}$ is a submanifold of a Euclidean space $\mathcal{E}$. To obtain a tangent vector at x, take a smooth curve $\gamma : \mathbb{R} \to \mathcal{M}$ with $\gamma(0) = x$; then $\dot\gamma(0)$—the derivative of γ at $t = 0$—is a tangent vector to $\mathcal{M}$ at x. Here the derivative is the usual derivative: since $\mathcal{M}$ is a subset of the Euclidean space $\mathcal{E}$, γ can be viewed as a curve in $\mathcal{E}$, and the derivative of γ is understood in this sense. The set of all tangent vectors at x is termed the *tangent space* to $\mathcal{M}$ at x and denoted by $T_x\mathcal{M}$. Given $\xi_x \in T_x\mathcal{M}$, we say that a curve γ on $\mathcal{M}$ *realizes* ξ_x if $\gamma(0) = x$ and $\dot\gamma(0) = \xi_x$.

A tangent vector ξ_x can be paired with any smooth real-valued function f on $\mathcal{M}$ to yield the real number

$$\mathrm{D}f(x)[\xi_x] = \frac{\mathrm{d}}{\mathrm{d}t} f(\gamma(t))\Big|_{t=0}, \tag{6}$$

where γ is any curve that realizes ξ_x. This property is the key to generalizing tangent vectors to abstract manifolds. A mapping $\xi_x : f \mapsto \xi_x(f)$ is a *tangent vector* to $\mathcal{M}$ at x is there exists a curve γ on $\mathcal{M}$ such that $\gamma(0) = x$ and $\xi_x(f) = \frac{\mathrm{d}}{\mathrm{d}t} f(\gamma(t))\big|_{t=0}$ for all smooth real-valued functions f on $\mathcal{M}$. Again, the curve γ is said to *realize* ξ_x. An alternate notation for $\xi_x(f)$ is $\mathrm{D}f(x)[\xi_x]$, but one should bear in mind that it is only for $\mathcal{M}$ submanifold of a Euclidean space that $\mathrm{D}f(x)[\xi_x]$ is equal to $\lim_{t\to 0} \frac{\bar{f}(x+t\xi_x)-\bar{f}(x)}{t}$ for any smooth extension $\bar{f}$ of f.

The above is a curve-based definition of tangent vectors; several equivalent definitions can be found in the literature. We also point out that the disjoint union of the tangent spaces admits a natural manifold structure. This manifold is called the *tangent bundle* and denoted by $T\mathcal{M}$. This concept will reappear below when we introduce the notion of retraction.

6.2 Descent directions

With the notion of a tangent vector at hand, we can define a *descent direction* for an objective function f on a manifold $\mathcal{M}$ at a point x to be a tangent vector ξ_x at x such that $\mathrm{D}f(x)[\xi_x] < 0$. In this case, for any curve γ that realizes ξ_x, we have $\frac{\mathrm{d}}{\mathrm{d}t} f(\gamma(t))\big|_{t=0} < 0$. Hence, for all t positive and sufficiently small, $f(\gamma(t)) < f(x)$.

6.3 Steepest-descent direction and the gradient

By definition, the steepest ascent direction is along

$$\operatorname*{arg\,max}_{\substack{\xi_x \in T_x\mathcal{M} \\ \|\xi_x\|=1}} \mathrm{D}f(x)[\xi_x].$$

For this expression to be well-defined, we need a norm on $T_x\mathcal{M}$. The most convenient way of introducing such a norm is via an inner product. For all $x \in \mathcal{M}$, let g_x be an inner product in $T_x\mathcal{M}$, and define

$$\|\xi_x\| := \sqrt{g_x(\xi_x, \xi_x)}.$$

When g_x smoothly depends on x, $(\mathcal{M}, g)$ is termed a *Riemannian manifold.* As was the case with the maximal atlas, the notation $(\mathcal{M}, g)$ is often replaced by $\mathcal{M}$ when no confusion arises.

There is a unique element of $T_x\mathcal{M}$, called the *gradient* of f at x and denoted by $\operatorname{grad} f(x)$, such that

$$\begin{cases} \operatorname{grad} f(x) \in T_x\mathcal{M} \\ g_x(\operatorname{grad} f(x), \xi_x) = \mathrm{D}f(x)[\xi_x], \quad \forall \xi_x \in T_x\mathcal{M}. \end{cases}$$

The gradient of f at x, whose definition depends on the Riemannian metric, is along the steepest-ascent direction of f at x, whose definition also depends on the Riemannian metric:

$$\frac{\operatorname{grad} f(x)}{\|\operatorname{grad} f(x)\|} = \underset{\substack{\xi_x \in T_x\mathcal{M} \\ \|\xi_x\|=1}}{\arg\max}\, \mathrm{D}f(x)[\xi_x].$$

Hence, the steepest-descent direction is along $-\operatorname{grad} f(x)$.

Moreover, the norm of the gradient of f at x is equal to the slope at $t = 0$ of $t \mapsto f(\gamma(t))$, where γ is any curve that realizes $\frac{\operatorname{grad} f(x)}{\|\operatorname{grad} f(x)\|}$:

$$\|\operatorname{grad} f(x)\| = \mathrm{D}f(x)\left[\frac{\operatorname{grad} f(x)}{\|\operatorname{grad} f(x)\|}\right].$$

6.4 Gradient on submanifolds

Let $(\overline{\mathcal{M}}, \overline{g})$ be a Riemannian manifold and $\mathcal{M}$ be a submanifold of $\overline{\mathcal{M}}$. Then

$$g_x(\xi_x, \zeta_x) := \overline{g}_x(\xi_x, \eta_x),\ \forall \xi_x, \zeta_x \in T_x\mathcal{M}$$

defines a Riemannian metric g on $\mathcal{M}$. With this Riemannian metric, $\mathcal{M}$ is a *Riemannian submanifold* of $\overline{\mathcal{M}}$. Let $T_x^\perp\mathcal{M}$ denote the orthogonal complement of $T_x\mathcal{M}$ in $T_x\overline{\mathcal{M}}$ in the sense of $\overline{g}$. Every $z \in T_x\overline{\mathcal{M}}$ admits a decomposition $z = \mathrm{P}_x z + \mathrm{P}_x^\perp z$, where $\mathrm{P}_x z$ belongs to $T_x\mathcal{M}$ and $\mathrm{P}_x^\perp z$ to $T_x^\perp\mathcal{M}$. If $\bar{f} : \overline{\mathcal{M}} \to \mathbb{R}$ and $f = \bar{f}|_{\mathcal{M}}$, then

$$\operatorname{grad} f(x) = \mathrm{P}_x \operatorname{grad} \bar{f}(x). \tag{7}$$

6.5 Gradient on quotient manifolds

For the case of quotient manifolds, see [5, §3.6.2].

6.6 Choice of the search curve

The next task is to choose a curve γ through x at $t = 0$ such that

$$\dot{\gamma}(0) = -\operatorname{grad} f(x).$$

The curve selection process can be specified by a retraction. A *retraction* on $\mathcal{M}$ is a smooth mapping $R : T\mathcal{M} \to \mathcal{M}$ such that, for all $x \in \mathcal{M}$ and all $\xi_x \in T_x\mathcal{M}$,

1. $R(0_x) = x$, where 0_x denotes the origin of $T_x\mathcal{M}$;
2. $\frac{\mathrm{d}}{\mathrm{d}t} R(t\xi_x)\big|_{t=0} = \xi_x$.

Given a retraction R on $\mathcal{M}$, the curve $\gamma : t \mapsto R(-t\operatorname{grad} f(x))$ is a descent curve at $t = 0$ provided that $\operatorname{grad} f(x) \neq 0$.

Note that, in topology, a continuous map from a topological space X to a subspace A is termed a retraction if the restriction of the map to domain A is the identity map on A. In view of the property $R(0_x) = x$ and the natural inclusion of $\mathcal{M}$ in $T\mathcal{M}$, the differential-geometric retractions are topological retractions.

6.7 Line-search procedure

It remains to find t_* such that $f(\gamma(t_*))$ is sufficiently smaller than $f(\gamma(0))$. Since $t \mapsto f(\gamma(t))$ is simply a function from $\mathbb{R}$ to $\mathbb{R}$, we can use the step selection techniques that are available for classical line-search methods, e.g., exact minimization or Armijo backtracking.

The next iterate of the steepest-descent method is defined to be $x_+ = \gamma(t_*)$. Observe that the method can be tuned by modifying the Riemannian metric and the retraction.

7 A steepest-descent method for Problem 2

As an illustration, we apply the steepest-descent method of the previous section to Problem 2.

Let $A = A^T \in \mathbb{R}^{n\times n}$ with (unknown) eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n$. The goal is to compute the p dominant eigenvectors of A, i.e., those associated to $\lambda_1, \ldots, \lambda_p$, which are uniquely defined (up to sign reversal, assuming a unit-norm constraint) if $\lambda_1 > \cdots > \lambda_p$. To this end, we define $N = \mathrm{diag}(p, p-1, \cdots, 1)$ and solve

$$\max_{X^TX=I_p} \mathrm{trace}(X^TAXN). \tag{8}$$

The columns of the solution X (unique up to sign reversal) are the p dominant eigenvectors or A; see [24] or [5, §4.8].

Let us sketch the derivation of a steepest-ascent method on $\mathrm{St}(p,n) = \{X \in \mathbb{R}^{n\times p} : X^TX = I\}$ for solving (8). Details can be found in [5, §4.8]. Define $\bar{f} : \mathbb{R}^{n\times p} \to \mathbb{R} : X \mapsto \mathrm{trace}(X^TAXN)$ and $f = \bar{f}|_{\mathrm{St}(p,n)}$. We have $\frac{1}{2}\mathrm{grad}\,\bar{f}(X) = AXN$. Thus, in view of (7), $\frac{1}{2}\mathrm{grad}\,f(X) = \mathrm{P}_{T_X\mathrm{St}(p,n)}(AXN) = AXN - X\mathrm{sym}(X^TAXN)$, where $\mathrm{sym}(Z) := (Z+Z^T)/2$. This is the gradient in the sense of the Riemannian metric inherited from the embedding of $\mathrm{St}(p,n)$ in $\mathbb{R}^{n\times p}$. Possible choices for the retraction are given in [5, Ex. 4.1.3]. For example, the mapping given by $R(\xi_X) = \mathrm{qf}(X + \xi_X)$ is a retraction, where qf returns the Q factor of the QR decomposition of A.

This basic steepest-descent algorithm is given as an illustration; it is not meant to be competitive with state-of-the-art algorithms for eigenvalue computation. Competitive algorithms that stem from a Riemannian optimization approach can be found in [11, 10].

8 Newton's method on manifolds

We first present Newton's method on general manifolds. Then we particularize the algorithm to obtain an algorithm for Problem 2 with $p = 1$.

8.1 Newton on abstract manifolds

The central equation for Newton's method in $\mathbb{R}^n$ is

$$\mathrm{D}(\operatorname{grad} f)(x)[\eta_x] = -\operatorname{grad} f(x),$$

a linear equation in the update vector η_x. On a Riemannian manifold, it is clear that η_x becomes a tangent vector at x, and that $\operatorname{grad} f$ becomes the gradient vector field defined in Section 6.3. It remains to define the directional derivative of a vector field such as $\operatorname{grad} f$. A thoughtless extension of (6) would yield the formula $\lim_{t\to 0} \frac{\operatorname{grad} f(\gamma(t)) - \operatorname{grad} f(x)}{t}$, which is inapplicable to abstract manifolds since $\operatorname{grad} f(\gamma(t))$ and $\operatorname{grad} f(x)$ belong to $T_{\gamma(t)}\mathcal{M}$ and $T_x\mathcal{M}$, which are two different vector spaces. The remedy is given by endowing $\mathcal{M}$ with an object called an affine connection and denoted by ∇, that takes as argument a vector field and a tangent vector and returns the (covariant) derivative of the vector field along the tangent vector.

The Riemannian Newton method given below is formulated as in [7] (or see [5, §6.2]).

Required: Riemannian manifold $\mathcal{M}$; retraction R on $\mathcal{M}$; affine connection ∇ on $\mathcal{M}$; real-valued function f on $\mathcal{M}$.

Iteration $x_k \in \mathcal{M} \mapsto x_{k+1} \in \mathcal{M}$ defined by

1. Solve the Newton equation
$$\operatorname{Hess} f(x_k)\eta_k = -\operatorname{grad} f(x_k)$$
for the unknown $\eta_k \in T_{x_k}\mathcal{M}$, where
$$\operatorname{Hess} f(x_k)\eta_k := \nabla_{\eta_k} \operatorname{grad} f.$$
2. Set
$$x_{k+1} := R_{x_k}(\eta_k).$$

The algorithm has convergence properties akin to those of Newton's algorithm in $\mathbb{R}^n$ [5, §6.3].

8.2 Newton on submanifolds of $\mathbb{R}^n$

If $\mathcal{M}$ is a submanifold of $\mathbb{R}^n$, it naturally inherits a Riemannian metric by the restriction of the standard inner product of $\mathbb{R}^n$. If moreover the so-called Levi-Civita connection is chosen for ∇, the algorithm below is obtained.

Required: Riemannian submanifold $\mathcal{M}$ of $\mathbb{R}^n$; retraction R on $\mathcal{M}$; real-valued function f on $\mathcal{M}$.

Iteration $x_k \in \mathcal{M} \mapsto x_{k+1} \in \mathcal{M}$ defined by

1. Solve the Newton equation
$$\operatorname{Hess} f(x_k)\eta_k = -\operatorname{grad} f(x_k)$$

for the unknown $\eta_k \in T_{x_k}\mathcal{M}$, where

$$\operatorname{Hess} f(x_k)\eta_k := \mathrm{P}_{T_{x_k}\mathcal{M}}\mathrm{D}\operatorname{grad} f(x_k)[\eta_k].$$

2. Set

$$x_{k+1} := R_{x_k}(\eta_k).$$

8.3 Newton on the unit sphere S^{n-1}

Let us now particularize the algorithm to the case where $\mathcal{M}$ is the unit sphere S^{n-1}, viewed as a Riemannian submanifold of $\mathbb{R}^n$, with a particular choice for the retraction. We obtain a numerical algorithm that can be formulated without any reference to differential-geometric objects, and that inherits the desirable convergence properties of the abstract Riemannian Newton method.

Required: real-valued function f on S^{n-1}.

Iteration $x_k \in S^{n-1} \mapsto x_{k+1} \in S^{n-1}$ defined by

1. Solve the Newton equation

$$\begin{cases} \mathrm{P}_{x_k}\mathrm{D}(\operatorname{grad} f)(x_k)[\eta_k] = -\operatorname{grad} f(x_k) \\ x^T\eta_k = 0, \end{cases}$$

for the unknown $\eta_k \in \mathbb{R}^n$, where $\mathrm{P}_{x_k} = (I - x_k x_k^T)$.

2. Set

$$x_{k+1} := \frac{x_k + \eta_k}{\|x_k + \eta_k\|}.$$

In the algorithm above, $\operatorname{grad} f(x) = (I - xx^T)\operatorname{grad} \bar{f}(x)$, where $\bar{f}(x)$ is any smooth extension of f.

8.4 Newton for Rayleigh quotient optimization on unit sphere

Finally, if we apply the above algorithm to a specific objective function, such as the one given in Problem 2 with $p = 1$, we obtain a concrete numerical algorithm.

Iteration $x_k \in S^{n-1} \mapsto x_{k+1} \in S^{n-1}$ defined by

1. Solve the Newton equation

$$\begin{cases} \mathrm{P}_{x_k}A\mathrm{P}_{x_k}\eta_k - \eta_k x_k^T A x_k = -\mathrm{P}_{x_k}Ax_k, \\ x_k^T\eta_k = 0, \end{cases}$$

for the unknown $\eta_k \in \mathbb{R}^n$, where $\mathrm{P}_{x_k} = (I - x_k x_k^T)$.

2. Set

$$x_{k+1} := \frac{x_k + \eta_k}{\|x_k + \eta_k\|}.$$

Not surprisingly for such a fundamental problem, we fall back on a known eigenvalue algorithm, the Rayleigh quotient iteration. On several other problems, the Riemannian Newton method has led to novel numerical algorithms; see, e.g., [41, 4, 40, 58, 22, 19].

9 Other optimization methods on manifolds

Besides steepest descent and Newton, several other classical methods for unconstrained optimization admit a generalization to manifolds. Chapter 8 in [5] briefly mentions approximate Newton methods and conjugate gradient schemes. A Riemannian trust-region method was proposed in [2] (or see [5, Ch. 7]), which led to competitive algorithms for symmetric eigenvalue problems [11, 10]. For a Riemannian BFGS method, see [48] and references therein.

The relation between optimization methods on manifolds and feasible methods for equality-constrained optimization is investigated in [6]. This concerns in particular the theory of $\mathcal{U}$-Lagrangians, and the related $\mathcal{VU}$-decompositions and fast tracks [35, 43], as well as the theory of partly smooth functions [36], both of which coincide in the convex case [44, Th. 2.9]. The concepts of $\mathcal{U}$-Lagrangian and partly smooth functions led to several Newton-like algorithms whose iterates are constrained to a submanifold $\mathcal{M}$ such that the restriction $f_{|\mathcal{M}}$ is smooth. These algorithms are unified in [16] under a common two-step, predictor-corrector form, and connections with SQP and Riemannian Newton are studied in [44].

We also mention the literature on proximal point algorithms on Hadamard manifolds; see [37] and references therein.

10 Conclusion

We have proposed an introduction to the area of optimization on manifolds, written as a digest of [5] enhanced with references to the most recent literature. In summary, optimization on manifolds is about exploiting tools of differential geometry to build optimization schemes on abstract manifolds, then turning these abstract geometric algorithms into practical numerical methods for specific manifolds, with applications to problems that can be rephrased as optimizing a differentiable function over a manifold. This research program has shed new light on existing algorithms and produced novel numerical methods backed by a strong convergence analysis.

We close by pointing out that optimization of real-valued functions on manifolds, as formulated in Problem 1, is not the only place where numerical optimization and differential geometry meet. Noteworthy are the Riemannian geometry of the central path in linear programming [17, 45], and an intriguing continuous-time system on the Grassmann manifold associated with linear programs [63, 1].

Acknowledgements

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity

Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors. This work was supported in part by "Communauté française de Belgique - Actions de Recherche Concertées" and by the Australian Research Council through discovery grant DP0987411 "State Observers for Control Systems with Symmetry".

This work benefited from fruitful interaction with a fairly large number of colleagues, most of whom appear in the bibliography. The first author would like to thank especially Alan Edelman, Uwe Helmke, and Knut Hüper for helping him into this elegant area of research, and Kyle Gallivan and Paul Van Dooren for a valued and productive collaboration.

References

1. P.-A. Absil. Numerical representations of a universal subspace flow for linear programs. *Communications in Information and Systems*, 8(2):71–84, 2009.
2. P.-A. Absil, C. G. Baker, and K. A. Gallivan. Trust-region methods on Riemannian manifolds. *Found. Comput. Math.*, 7(3):303–330, July 2007.
3. P.-A. Absil and K. A. Gallivan. Joint diagonalization on the oblique manifold for independent component analysis. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 5, pages V–945–V–948, 2006.
4. P.-A. Absil, R. Mahony, and R. Sepulchre. Riemannian geometry of Grassmann manifolds with a view on algorithmic computation. *Acta Appl. Math.*, 80(2):199–220, January 2004.
5. P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, 2008.
6. P.-A. Absil, Jochen Trumpf, Robert Mahony, and Ben Andrews. All roads lead to Newton: Feasible second-order methods for equality-constrained optimization. Technical Report UCL-INMA-2009.024, UCLouvain, 2009.
7. Roy L. Adler, Jean-Pierre Dedieu, Joseph Y. Margulies, Marco Martens, and Mike Shub. Newton's method on Riemannian manifolds and a geometric model for the human spine. *IMA J. Numer. Anal.*, 22(3):359–390, July 2002.
8. Bijan Afsari and P. S. Krishnaprasad. Some gradient based joint diagonalization methods for ICA. In Springer LCNS Series, editor, *Proceedings of the 5th International Conference on Independent Component Analysis and Blind Source Separation*, 2004.
9. Gregory Ammar and Clyde Martin. The geometry of matrix eigenvalue methods. *Acta Appl. Math.*, 5(3):239–278, 1986.
10. C. G. Baker, P.-A. Absil, and K. A. Gallivan. An implicit trust-region method on Riemannian manifolds. *IMA J. Numer. Anal.*, 28(4):665–689, 2008.
11. Christopher G. Baker. *Riemannian manifold trust-region methods with applications to eigenproblems*. PhD thesis, School of Computational Science, Florida State University, Summer Semester 2008.
12. M. Baumann and U. Helmke. Riemannian subspace tracking algorithms on Grassmann manifolds. In *Proceedings of the 46th IEEE Conference on Decision and Control*, 2007.

13. Silvère Bonnabel and Rodolphe Sepulchre. Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank. *SIAM J. Matrix Anal. Appl.*, 31(3):1055–1070, 2009.
14. Thomas J. Bridges and Sebastian Reich. Computing Lyapunov exponents on a Stiefel manifold. *Phys. D*, 156(3-4):219–238, 2001.
15. Hasan Ertan Çetingül and René Vidal. Intrinsic mean shift for clustering on Stiefel and Grassmann manifolds. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'09)*, 2009.
16. Aris Daniilidis, Warren Hare, and Jérôme Malick. Geometrical interpretation of the predictor-corrector type algorithms in structured optimization problems. *Optimization*, 55(5-6):481–503, 2006.
17. Jean-Pierre Dedieu, Gregorio Malajovich, and Mike Shub. On the curvature of the central path of linear programming theory. *Found. Comput. Math.*, 5(2):145–171, 2005.
18. Alan Edelman, Tomás A. Arias, and Steven T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.*, 20(2):303–353, 1998.
19. L. Eldén and B. Savas. A Newton–Grassmann method for computing the best multi-linear rank-(r_1, r_2, r_3) approximation of a tensor. *SIAM J. Matrix Anal. Appl.*, 31:248–271, 2009.
20. Lars Eldén and Haesun Park. A Procrustes problem on the Stiefel manifold. *Numer. Math.*, 82(4):599–619, 1999.
21. Hermann Grassmann. Wörterbuch zum Rig-Veda, 1873. Leipzig.
22. Uwe Helmke, Knut Hüper, Pei Yean Lee, and John B. Moore. Essential matrix estimation using Gauss-Newton iterations on a manifold. *Int. J. Computer Vision*, 74(2):117–136, 2007.
23. Uwe Helmke, Knut Hüper, and Jochen Trumpf. Newton's method on Grassmann manifolds, September 2007. arXiv:0709.2205v2.
24. Uwe Helmke and John B. Moore. *Optimization and Dynamical Systems.* Communications and Control Engineering Series. Springer-Verlag London Ltd., London, 1994. With a foreword by R. Brockett.
25. Magnus R. Hestenes and Eduard Stiefel. Methods of conjugate gradients for solving linear systems. *J. Research Nat. Bur. Standards*, 49:409–436 (1953), 1952.
26. Knut Hüper, Hao Shen, and Abd-Krim Seghouane. Local convergence properties of FastICA and some generalisations. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 5, pages V–1009–V–1012, 2006.
27. M. Ishteva, L. De Lathauwer, P.-A. Absil, and S. Van Huffel. Best low multilinear rank approximation of higher-order tensors, based on the Riemannian trust-region scheme. Technical Report 09-142, ESAT-SISTA, K.U.Leuven, Belgium, 2009.
28. Jens Jordan and Uwe Helmke. Controllability of the QR-algorithm on Hessenberg flags. In David S. Gilliam and Joachim Rosenthal, editors, *Proceeding of the Fifteenth International Symposium on Mathematical Theory of Network and Systems (MTNS 2002)*, 2002.
29. Shantanu H. Joshi, Eric Klassen, Anuj Srivastava, and Ian Jermyn. A novel representation for Riemannian analysis of elastic curves in R^n. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

30. M. Journée, F. Bach, P.-A. Absil, and R. Sepulchre. Low-rank optimization for semidefinite convex problems, 2008. arXiv:0807.4423.
31. E. Klassen, A. Srivastava, M. Mio, and S.H. Joshi. Analysis of planar shapes using geodesic paths on shape spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):372–383, 2004.
32. Eric Klassen and Anuj Srivastava. Geodesics between 3D closed curves using path-straightening. In A. Leonardis, H. Bischof, and A. Pinz, editors, *ECCV 2006, Part I,*, volume 3951 of *LNCS*, pages 95–106. Springer-Verlag, Berlin Heidelberg, 2006.
33. John M. Lee. *Introduction to smooth manifolds*, volume 218 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2003.
34. Pei Yean Lee and John B. Moore. Pose estimation via a Gauss-Newton-on-manifold approach. In *Proceedings of the 16th International Symposium on Mathematical Theory of Network and System (MTNS), Leuven*, 2004.
35. Claude Lemaréchal, François Oustry, and Claudia Sagastizábal. The $\mathcal{U}$-Lagrangian of a convex function. *Trans. Amer. Math. Soc.*, 352(2):711–729, 2000.
36. A. S. Lewis. Active sets, nonsmoothness, and sensitivity. *SIAM J. Optim.*, 13(3):702–725 (electronic) (2003), 2002.
37. Chong Li, Genaro López, and Victoria Martín-Márquez. Monotone vector fields and the proximal point algorithm on Hadamard manifolds. *J. London Math. Soc.*, 79(3):663–683, 2009.
38. R. Lippert and A. Edelman. Nonlinear eigenvalue problems with orthogonality constraints (Section 9.4). In Zhaojun Bai, James Demmel, Jack Dongarra, Axel Ruhe, and Henk van der Vorst, editors, *Templates for the Solution of Algebraic Eigenvalue Problems*, pages 290–314. SIAM, Philadelphia, 2000.
39. Xiuwen Liu, Anuj Srivastava, and Kyle Gallivan. Optimal linear representations of images for object recognition. *IEEE Pattern Anal. and Mach. Intell.*, 26(5):662–666, May 2004.
40. Eva Lundström and Lars Eldén. Adaptive eigenvalue computations using Newton's method on the Grassmann manifold. *SIAM J. Matrix Anal. Appl.*, 23(3):819–839, 2001/02.
41. Yi Ma, Jana Kosecka, and Shankar S. Sastry. Optimization criteria and geometric algorithms for motion and structure estimation. *Int. J. Computer Vision*, 44(3):219–249, 2001.
42. Jonathan H. Manton. A centroid (Karcher mean) approach to the joint approximate diagonalization problem: The real symmetric case. *Digital Signal Processing*, 16(5):468–478, 2005.
43. Robert Mifflin and Claudia Sagastizábal. On $\mathcal{VU}$-theory for functions with primal-dual gradient structure. *SIAM J. Optim.*, 11(2):547–571 (electronic), 2000.
44. Scott A. Miller and Jérôme Malick. Newton methods for nonsmooth convex minimization: connections among $\mathcal{U}$-Lagrangian, Riemannian Newton and SQP methods. *Math. Program.*, 104(2-3, Ser. B):609–633, 2005.
45. Y. Nesterov and A. Nemirovski. Primal central paths and Riemannian distances for convex sets. *Found. Comput. Math.*, 8(5):533–560, 2008.
46. Yasunori Nishimori and Shotaro Akaho. Learning algorithms utilizing quasi-geodesic flows on the Stiefel manifold. *Neurocomputing*, 67:106–135, 2005.

47. Yasunori Nishimori, Shotaro Akaho, and Mark D. Plumbley. Natural conjugate gradient on complex flag manifolds for complex independent subspace analysis. In Vera Kurkova-Pohlova, Roman Neruda, and Jan Koutnik, editors, *Artificial Neural Networks - ICANN 2008*, volume 5163 of *LNCS*, pages 165–174. Springer, Berlin Heidelberg, 2008.
48. Chunhong Qi, Kyle A. Gallivan, and P.-A. Absil. Riemannian BFGS algorithm with applications. In *Recent Advances in Optimization and its Applications in Engineering*. Springer, 2010. To appear.
49. Quentin Rentmeesters, P.-A. Absil, and Paul Van Dooren. Identification method for time-varying ARX models. Submitted, 2009.
50. Chafik Samir, P.-A. Absil, Anuj Srivastava, and Eric Klassen. A gradient-descent method for curve fitting on Riemannian manifolds. Technical Report UCL-INMA-2009.023, UCLouvain, 2009.
51. Oliver Sander. Geodesic finite elements for Cosserat rods. submitted, 2009.
52. Berkant Savas and Lek-Heng Lim. Best multilinear rank approximation of tensors with quasi-Newton methods on Grassmannians. Technical Report LITH-MAT-R-2008-01-SE, Department of Mathematics, Linköpings Universitet, 2008.
53. Anuj Srivastava and Eric Klassen. Bayesian and geometric subspace tracking. *Adv. in Appl. Probab.*, 36(1):43–56, 2004.
54. Anuj Srivastava and Xiuwen Liu. Tools for application-driven linear dimension reduction. *Neurocomputing*, 67:136–160, 2005.
55. E. Stiefel. Richtungsfelder und Fernparallelismus in n-dimensionalen Mannigfaltigkeiten. *Comment. Math. Helv.*, 8(1):305–353, 1935.
56. Fabian J. Theis, Thomas P. Cason, and P.-A. Absil. Soft dimension reduction for ICA by joint diagonalization on the Stiefel manifold. In *Proc. ICA 2009*, volume 5441 of *LNCS*, pages 354–361, Paraty, Brazil, 2009. Springer.
57. Frank Tompkins and Patrick J. Wolfe. Bayesian filtering on the Stiefel manifold. In *2nd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMPSAP 2007)*, 2007.
58. Nickolay T. Trendafilov and Ross A. Lippert. The multimode Procrustes problem. *Linear Algebra Appl.*, 349:245–264, 2002.
59. Pavan Turaga, Ashok Veeraraghavan, and Rama Chellappa. Statistical analysis on Stiefel and Grassmann manifolds with applications in computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.
60. Bart Vandereycken, P.-A. Absil, and Stefan Vandewalle. Embedded geometry of the set of symmetric positive semidefinite matrices of fixed rank. In *Proceedings of the IEEE 15th Workshop on Statistical Signal Processing*, pages 389–392, 2009.
61. Bart Vandereycken and Stefan Vandewalle. A Riemannian optimization approach for computing low-rank solutions of Lyapunov equations. Submitted, 2009.
62. Frank W. Warner. *Foundations of differentiable manifolds and Lie groups*, volume 94 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1983. Corrected reprint of the 1971 edition.
63. Gongyun Zhao. Representing the space of linear programs as the Grassmann manifold. *Math. Program.*, 121(2, Ser. A):353–386, 2010.

On the Best Low Multilinear Rank Approximation of Higher-order Tensors*

Mariya Ishteva[1], P.-A. Absil[1], Sabine Van Huffel[2], and Lieven De Lathauwer[2,3]

[1] Department of Mathematical Engineering, Université catholique de Louvain,
Bâtiment Euler - P13, Av. Georges Lemaître 4, 1348 Louvain-la-Neuve, Belgium,
mariya.ishteva@uclouvain.be, http://www.inma.ucl.ac.be/~absil

[2] Department of Electrical Engineering - ESAT/SCD, K.U.Leuven,
Kasteelpark Arenberg 10/2446, 3001 Leuven, Belgium,
sabine.vanhuffel@esat.kuleuven.be

[3] Group Science, Engineering and Technology, K.U.Leuven Campus Kortrijk,
E. Sabbelaan 53, 8500 Kortrijk, Belgium,
lieven.delathauwer@kuleuven-kortrijk.be

Summary. This paper deals with the best low multilinear rank approximation of higher-order tensors. Given a tensor, we are looking for another tensor, as close as possible to the given one and with bounded multilinear rank. Higher-order tensors are used in higher-order statistics, signal processing, telecommunications and many other fields. In particular, the best low multilinear rank approximation is used as a tool for dimensionality reduction and signal subspace estimation.

Computing the best low multilinear rank approximation is a nontrivial task. Higher-order generalizations of the singular value decomposition lead to suboptimal solutions. The higher-order orthogonal iteration is a widely used linearly convergent algorithm for further refinement. We aim for conceptually faster algorithms. However, applying standard optimization algorithms directly is not a good idea since there are infinitely many equivalent solutions. Nice convergence properties are observed when the solutions are isolated. The present invariance can be removed by working on quotient manifolds. We discuss three algorithms, based on Newton's method, the trust-region scheme and conjugate gradients. We also comment on the local minima of the problem.

* Research supported by: (1) The Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, "Dynamical systems, control and optimization", 2007–2011), (2) Communauté française de Belgique - Actions de Recherche Concertées, (3) Research Council K.U.Leuven: GOA-AMBioRICS, GOA-MaNet, CoE EF/05/006 Optimization in Engineering (OPTEC), (4) F.W.O. project G.0321.06, "Numerical tensor methods for spectral analysis". (5) "Impulsfinanciering Campus Kortrijk (2007–2012)(CIF1)" and STRT1/08/023. Part of this research was carried out while M. Ishteva was with K.U.Leuven, supported by OE/06/25, OE/07/17, OE/08/007, OE/09/004. The scientific responsibility rests with the authors.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_13, © Springer-Verlag Berlin Heidelberg 2010

1 Introduction

Multilinear algebra deals with higher-order tensors, generalizations of vectors and matrices to higher-dimensional tables of numbers. Tensor algebra is more involved than matrix algebra but can model more complex processes. Higher-order tensors are used in many application fields so efficient and reliable algorithms for handling them are required.

Matrices are second-order tensors with well-studied properties. The matrix rank is a well-understood concept. In particular, the low-rank approximation of a matrix is essential for various results and algorithms. However, the matrix rank and its properties are not easily or uniquely generalizable to higher-order tensors. The rank, the row rank and the column rank of a matrix are equivalent whereas in multilinear algebra these are in general different.

Of main concern for this paper is the multilinear rank [40, 41] of a tensor, which is a generalization of column and row rank of a matrix. In particular, we discuss algorithms for the best low multilinear rank approximation of a higher-order tensor. The result is a higher-order tensor, as close as possible to the original one and having bounded multilinear rank. In the matrix case, the solution is given by the truncated singular value decomposition (SVD) [34, §2.5]. In multilinear algebra, the truncated higher-order SVD (HOSVD) [22] gives a suboptimal approximation, which can be refined by iterative algorithms. The traditional algorithm for this purpose is the higher-order orthogonal iteration (HOOI) [23, 52, 53]. In this paper, we discuss conceptually faster algorithms based on the Newton method, trust-region scheme and conjugate gradients. We also comment on the fact that numerical methods converge to local minimizers [44] of the function associated with the best low multilinear approximation.

It will be shown that the cost function has an invariance property by the action of the orthogonal group. Conceptually speaking, the solutions are not isolated, i.e., there are whole groups of infinitely many equivalent elements. This is a potential obstacle for algorithms since in practice, convergence to one particular point has to be achieved. Differential geometric techniques remove successfully the mentioned invariance. The working spaces are quotient manifolds. The elements of such spaces are sets containing points that are in some sense equivalent. For our particular problem, we work with matrices but in practice we are only interested in their column space. There are infinitely many matrices with the same column space that can be combined in one compound element of a quotient space. Another possibility is to first restrict the set of all considered matrices to the set of matrices with column-wise orthonormal columns and then combine all equivalent matrices from the selected ones in one element. This is justified by the fact that any subspace can be represented by the column space of a column-wise orthonormal matrix. We consider both options. We can summarize that in this paper, a multilinear algebra optimization problem is solved using optimization on manifolds.

This paper is an overview of recent publications and technical reports

[47, 46, 43, 44, 45] and the PhD thesis [42]. We present a digest of current research results, a survey of the literature on the best low multilinear rank approximation problem and other tensor approximations and discuss some applications. The paper is organized as follows. In Section 2, some definitions and properties of higher-order tensors are given. The main problem is formulated, HOSVD and HOOI are presented and we also mention some other related algorithms from the literature. Some applications are demonstrated in Section 3. Three differential-geometric algorithms are discussed in Section 4. In Section 5, we talk about local minima. Conclusions are drawn in Section 6.

In this paper we consider third-order tensors. The differences in the properties and algorithms for third-order tensors and for tensors of order higher than three are mainly technical, whereas the differences between the matrix case and the case of third-order tensors are fundamental.

2 Background material

2.1 Basic definitions

An Nth-order tensor is an element of the tensor product of N vector spaces. When the choice of basis is implicit, we think of a tensor as its representation as an N-way array [28]. Each "direction" of an Nth order tensor is called a mode. The vectors, obtained by varying the nth index, while keeping the other indices fixed are called mode-n vectors ($n = 1, 2, \ldots, N$). For a tensor $\mathcal{A} \in \mathbb{R}^{6\times 5\times 4}$ they are visualized in Fig. 1. The mode-n rank of a tensor $\mathcal{A}$, denoted by $\text{rank}_n(\mathcal{A})$, is defined as the number of linearly independent mode-n vectors. The multilinear rank of a tensor is then the n-tuple of the mode-n ranks. An essential difference with the matrix case is that the mode-n ranks are in general different from each other.

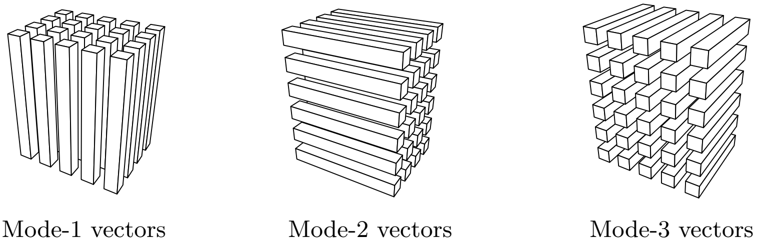


Fig. 1. Mode-n vectors of a $(6 \times 5 \times 4)$-tensor.

We use the following definition of mode-n products $\mathcal{A} \bullet_n \mathbf{M}^{(n)}$, $n = 1, 2, 3$ of a tensor $\mathcal{A} \in \mathbb{R}^{I_1\times I_2\times I_3}$ and matrices $\mathbf{M}^{(n)} \in \mathbb{R}^{J_n\times I_n}$:

$$\begin{aligned}(\mathcal{A} \bullet_1 \mathbf{M}^{(1)})_{j_1 i_2 i_3} &= \textstyle\sum_{i_1} a_{i_1 i_2 i_3} m^{(1)}_{j_1 i_1}\,,\\ (\mathcal{A} \bullet_2 \mathbf{M}^{(2)})_{i_1 j_2 i_3} &= \textstyle\sum_{i_2} a_{i_1 i_2 i_3} m^{(2)}_{j_2 i_2}\,,\\ (\mathcal{A} \bullet_3 \mathbf{M}^{(3)})_{i_1 i_2 j_3} &= \textstyle\sum_{i_3} a_{i_1 i_2 i_3} m^{(3)}_{j_3 i_3}\,,\end{aligned}$$

where $1 \leq i_n \leq I_n$, $1 \leq j_n \leq J_n$. In this notation, $\mathbf{A} = \mathbf{UMV}^T$ is presented as $\mathbf{A} = \mathbf{M} \bullet_1 \mathbf{U} \bullet_2 \mathbf{V}$. This is reasonable since the columns of $\mathbf{U}$ correspond to the column space of $\mathbf{A}$ in the same way as the columns of $\mathbf{V}$ correspond to the row space of $\mathbf{A}$. The mode-n product has the following properties

$$(\mathcal{A} \bullet_n \mathbf{U}) \bullet_m \mathbf{V} = (\mathcal{A} \bullet_m \mathbf{V}) \bullet_n \mathbf{U} = \mathcal{A} \bullet_n \mathbf{U} \bullet_m \mathbf{V}, \quad m \neq n,$$
$$(\mathcal{A} \bullet_n \mathbf{U}) \bullet_n \mathbf{V} = \mathcal{A} \bullet_n (\mathbf{V}\,\mathbf{U}).$$

It is often useful to represent a tensor in a matrix form, e.g., by putting all mode-n vectors one after the other in a specific order. For a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, the matrix representations $\mathbf{A}_{(n)}, n = 1, 2, 3$ that we use are

$$(\mathbf{A}_{(1)})_{i_1,(i_2-1)I_3+i_3} = (\mathbf{A}_{(2)})_{i_2,(i_3-1)I_1+i_1} = (\mathbf{A}_{(3)})_{i_3,(i_1-1)I_2+i_2} = a_{i_1 i_2 i_3}\,,$$

where $1 \leq i_n \leq I_n$. This definition is illustrated in Fig. 2 for $I_1 > I_2 > I_3$.

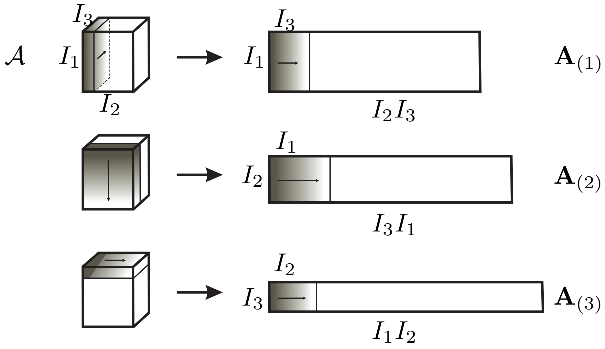


Fig. 2. Matrix representations of a tensor.

2.2 Best low multilinear rank approximation

Given $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, its best rank-$(R_1, R_2, R_3)$ approximation is a tensor $\hat{\mathcal{A}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, such that it *minimizes* the cost function $f : \mathbb{R}^{I_1 \times I_2 \times I_3} \rightarrow \mathbb{R}$,

$$f : \hat{\mathcal{A}} \mapsto \|\mathcal{A} - \hat{\mathcal{A}}\|^2 \tag{1}$$

under the constraints $\operatorname{rank}_1(\hat{\mathcal{A}}) \leq R_1$, $\operatorname{rank}_2(\hat{\mathcal{A}}) \leq R_2, \operatorname{rank}_3(\hat{\mathcal{A}}) \leq R_3$. This problem is equivalent [23, 52, 53] to the problem of *maximizing* the function

$$\begin{aligned} \overline{g} &: St(R_1, I_1) \times St(R_2, I_2) \times St(R_3, I_3) \rightarrow \mathbb{R}, \\ &(\mathbf{U}, \mathbf{V}, \mathbf{W}) \mapsto \|\mathcal{A} \bullet_1 \mathbf{U}^T \bullet_2 \mathbf{V}^T \bullet_3 \mathbf{W}^T\|^2 = \|\mathbf{U}^T \mathbf{A}_{(1)}(\mathbf{V} \otimes \mathbf{W})\|^2 \end{aligned} \tag{2}$$

over the matrices $\mathbf{U}, \mathbf{V}$ and $\mathbf{W}$ ($St(p, n)$ stands for the set of column-wise orthonormal $(n \times p)$-matrices, $\|\cdot\|$ is the Frobenius norm and $\otimes$ denotes the Kronecker product). This equivalence is a direct generalization of the matrix case where finding the best rank-R approximation $\hat{\mathbf{A}} = \mathbf{U}\,\mathbf{B}\,\mathbf{V}^T$ of $\mathbf{A} \in \mathbb{R}^{I_1 \times I_2}$, where $\mathbf{B} \in \mathbb{R}^{R \times R}$, $\mathbf{U} \in St(R, I_1)$, $\mathbf{V} \in St(R, I_2)$ and $\|\mathbf{A} - \hat{\mathbf{A}}\|$ is minimized, is equivalent to the maximization of $\|\mathbf{U}^T\,\mathbf{A}\,\mathbf{V}\| = \|\mathbf{A} \bullet_1 \mathbf{U}^T \bullet_2 \mathbf{V}^T\|$. Having estimated $\mathbf{U}, \mathbf{V}$ and $\mathbf{W}$ in (2), the solution of (1) is computed by

$$\hat{\mathcal{A}} = \mathcal{A} \bullet_1 \mathbf{U}\mathbf{U}^T \bullet_2 \mathbf{V}\mathbf{V}^T \bullet_3 \mathbf{W}\mathbf{W}^T.$$

Thus, in this paper, our goal is to solve the maximization problem (2). In practice, the function $-\overline{g}$ will be minimized.

2.3 Higher-order singular value decomposition

The SVD [34, §2.5] gives the best low-rank approximation of a matrix. In the sense of multilinear rank, a generalization of the SVD is the higher-order SVD (HOSVD) [22]. With possible variations it is also known as Tucker decomposition [72, 73]. HOSVD decomposes a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ as

$$\mathcal{A} = \mathcal{S} \bullet_1 \mathbf{U}^{(1)} \bullet_2 \mathbf{U}^{(2)} \bullet_3 \mathbf{U}^{(3)}\,,$$

where $\mathcal{S} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ and where $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times I_n}$, $n = 1, 2, 3$, are orthogonal, see Fig. 3. The matrices obtained from $\mathcal{S}$ by fixing any of the indices are orthogonal to each other and their norm is decreasing with increasing the fixed index. The mode-n singular values of $\mathcal{A}$ are the singular values of $\mathbf{A}_{(n)}$.

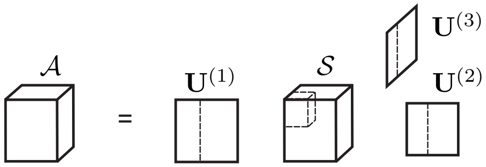


Fig. 3. Higher-order singular value decomposition.

For second-order tensors, i.e., matrices, HOSVD reduces to the well-known SVD. However, truncation of HOSVD results in a suboptimal solution of the best low multilinear rank approximation problem. This is due to the fact that in general, it is impossible to obtain a diagonal $\mathcal{S}$ tensor. The number of degrees of freedom in such a decomposition would be smaller than the number of the elements of the tensor that needs to be decomposed. However, the truncated HOSVD can serve as a good starting point for iterative algorithms.

Other generalizations of the matrix SVD have been discussed in the literature, focusing on different properties of the SVD. The tensor corresponding to $\mathcal{S}$ can be made as diagonal as possible (in a least squares sense) under orthogonal transformations [12, 24, 56, 10], or the original tensor can be decomposed in a minimal number of rank-1 terms (CANDECOMP/PARAFAC) [13, 37, 9, 25, 17], on which orthogonal [50] or symmetry [14] constraints can be imposed. A unifying framework for Tucker/HOSVD and CANDECOMP/PARAFAC is given by the block term decompositions [18, 19, 26].

2.4 Higher-order orthogonal iteration

The traditional iterative algorithm for maximizing (2) and thus minimizing (1) is the higher-order orthogonal iteration (HOOI) [23, 52, 53]. It is an alternating least-squares (ALS) algorithm. At each step the estimate of one of the matrices $\mathbf{U}, \mathbf{V}, \mathbf{W}$ is optimized, while the other two are kept constant. The function $\overline{g}$ from (2) is thought of as a quadratic expression in the components of the matrix that is being optimized. For fixed $\mathbf{V}$ and $\mathbf{W}$, since

$$\overline{g}(\mathbf{U}, \mathbf{V}, \mathbf{W}) = \|\mathcal{A} \bullet_1 \mathbf{U}^T \bullet_2 \mathbf{V}^T \bullet_3 \mathbf{W}^T\|^{\,2} = \|\mathbf{U}^T \left(\mathbf{A}_{(1)}(\mathbf{V} \otimes \mathbf{W})\right)\|^{\,2} ,$$

the columns of the optimal $\mathbf{U} \in \mathbb{R}^{I_1 \times R_1}$ build an orthonormal basis for the left R_1-dimensional dominant subspace of $\mathbf{A}_{(1)}(\mathbf{V} \otimes \mathbf{W})$. It can be obtained from the SVD of $\mathbf{A}_{(1)}(\mathbf{V} \otimes \mathbf{W})$. The optimization with respect to the other two unknown matrices is performed by analogy.

Initial matrices for HOOI are often taken from the truncated HOSVD. These matrices usually belong to the attraction region of (2) but there are exceptions. Moreover, convergence to the global maximum is not guaranteed.

HOOI is a simple concept and easy to implement. Therefore it is the most widely used algorithm at the moment [51]. If we assume for simplicity that $R_1 = R_2 = R_3 = R$ and $I_1 = I_2 = I_3 = I$, the total cost for one iteration of HOOI is then $O(I^3R + IR^4 + R^6)$ [32, 47]. However, the convergence speed of HOOI is at most linear.

2.5 Other methods in the literature

Recently, a Newton-type algorithm for the best low multilinear rank approximation of tensors has been proposed in [32]. It works on the so-called Grassmann manifold whereas the Newton-type algorithm considered in this paper is a generalization of the ideas behind the geometric Newton method for Oja's vector field [2]. Quasi-Newton methods have been suggested in [64].

We also mention other related methods. A Krylov method for large sparse tensors has been proposed in [63]. In [23, 75, 49], specific algorithms for the best rank-1 approximation have been discussed. Fast HOSVD algorithms for symmetric, Toeplitz and Hankel tensors have been proposed in [7]. For tensors with large dimensions, Tucker-type decompositions are developed in [59, 8, 54].

3 Some applications

The best low multilinear rank approximation of tensors is used for signal subspace estimation [60, 61, 52, 67, 51, 35] and as a dimensionality reduction tool for tensors with high dimensions [27, 4, 29, 30, 52, 67, 51], including simultaneous dimensionality reduction of a matrix and a tensor [27].

Independent component analysis (ICA) [27] extracts statistically independent sources from a linear mixture in fields like electroencephalography

(EEG), magnetoencephalography (MEG) and nuclear magnetic resonance (NMR). Sometimes only a few sources have significant contributions. A principal component analysis (PCA)-based prewhitening step for reducing the dimensionality is often used. This is beneficial if white Gaussian noise is present but is not applicable in case of colored Gaussian noise. In the latter case, low multilinear rank approximation of a higher-order cumulant tensor of the observation vector can be performed instead. The dimensionality of the problem is reduced from the number of observation channels to the number of sources.

A rank-1 tensor is an outer product of a number of vectors. The decomposition of higher-order tensors in rank-1 terms is called parallel factor decomposition (PARAFAC) [37] or canonical decomposition (CANDECOMP) [9]. It has applications in chemometrics [67], wireless communication [66, 21], and can also be used for epileptic seizure onset localization [30, 29, 4], since only one of the rank-1 terms is related to the seizure activity. The best low multilinear rank approximation of tensors is often used as a dimensionality reduction step preceding the actual computation of PARAFAC. Such a preprocessing step is implemented for example in the N-way toolbox for MATLAB [6].

Dimensionality reduction works as illustrated in Fig. 4. See also [16, Remark 6.2.2]. Let the rank-R decomposition of $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ be required. If $R < \max(I_1, I_2, I_3)$, then a reduction of $\mathcal{A}$ to a tensor $\mathcal{B} \in \mathbb{R}^{I'_1 \times I'_2 \times I'_3}$, $I'_n = \min(I_n, R), n = 1, 2, 3$ can be used for the actual computation of PARAFAC. This can be done as follows. Let $\hat{\mathcal{A}}$ be the best rank-(I'_1, I'_2, I'_3) approximation of $\mathcal{A}$. If $\mathbf{U}, \mathbf{V}, \mathbf{W}$ are the matrices as in (2), i.e., if

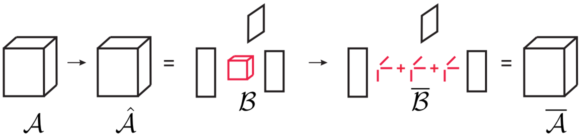


Fig. 4. Dimensionality reduction.

$$\hat{\mathcal{A}} = \mathcal{B} \bullet_1 \mathbf{U} \bullet_2 \mathbf{V} \bullet_3 \mathbf{W}$$

then a rank-R approximation $\overline{\mathcal{A}}$ of $\mathcal{A}$ is computed from the best rank-R approximation $\overline{\mathcal{B}}$ of $\mathcal{B}$ in the following way

$$\overline{\mathcal{A}} = \overline{\mathcal{B}} \bullet_1 \mathbf{U} \bullet_2 \mathbf{V} \bullet_3 \mathbf{W}.$$

Tensor $\mathcal{B}$ has smaller dimensions than $\mathcal{A}$ so that computing $\overline{\mathcal{B}}$ is much less expensive than directly computing $\overline{\mathcal{A}}$. In practice, due to numerical problems, in some applications $I'_n = \min(I_n, R+2), n = 1, 2, 3$ are used instead of the dimensions $I'_n = \min(I_n, R)$. In general, it is advisable to examine the mode-n singular values for gaps between them and use a corresponding low multilinear rank approximation. It might also be useful to perform a few additional PARAFAC steps on $\overline{\mathcal{A}}$ in order to find an even better approximation of $\mathcal{A}$.

In signal processing applications, a signal is often modeled as a sum of exponentially damped sinusoids (EDS). The parameters of the model have to be estimated given only samples of the signal. In the literature there are both matrix [31, 74] and tensor-based algorithms [60, 61]. The latter are based on the best rank-(R_1, R_2, R_3) approximation. In [48], the EDS model in the multi-channel case is considered in the case of closely spaced poles. This problem is more difficult than the case where the poles are well separated. A comparison of the performance of a matrix-based and a tensor-based method was performed. None of them always outperforms the other one. However, in the tensor-based algorithm, one can choose the mode-3 rank in such a way that the performance is optimal. Numerical experiments indicate that if ill-conditioning is present in the mode corresponding to the complex amplitudes, taking a lower value for the mode-3 rank than for the mode-1 and mode-2 ranks improves the performance of the tensor method to the extent that it outperforms the matrix method.

For more references and application areas, we refer to the books [67, 52, 11], to the overview papers [51, 20] and to the references therein.

4 Algorithms

In this section, we will review three classical optimization algorithms adapted for quotient matrix manifolds. We will then show how these algorithms can be applied on the best low multilinear rank approximation problem.

4.1 Geometric Newton algorithm

In order to apply Newton's method, the solutions of the optimization problem (2) have to be reformulated as zeros of a suitable function. The matrix $\mathbf{U} \in St(R_1, I_1)$ is optimal if and only if [38, Th. 3.17] its column space is the R_1-dimensional left dominant subspace of $\mathbf{A}_{(1)}(\mathbf{V} \otimes \mathbf{W})$. A necessary condition for this is that the column space of $\mathbf{U}$ is an invariant subspace of $\mathbf{A}_{(1)}(\mathbf{V} \otimes \mathbf{W})(\mathbf{V} \otimes \mathbf{W})^T \mathbf{A}_{(1)}^T$. Defining $\mathbf{X} = (\mathbf{U}, \mathbf{V}, \mathbf{W})$ and

$$\mathbf{R}_1(\mathbf{X}) = \mathbf{U}^T \mathbf{A}_{(1)}(\mathbf{V} \otimes \mathbf{W}),$$

this condition can be written as

$$F_1(\mathbf{X}) \equiv \mathbf{U}\,\mathbf{R}_1(\mathbf{X})\mathbf{R}_1(\mathbf{X})^T - \mathbf{A}_{(1)}(\mathbf{V} \otimes \mathbf{W})\mathbf{R}_1(\mathbf{X})^T = \mathbf{0}.$$

In the same way two more conditions are obtained for the matrices $\mathbf{V}$ and $\mathbf{W}$. The new function is then

$$\begin{aligned} F :\, & \mathbb{R}^{I_1 \times R_1} \times \mathbb{R}^{I_2 \times R_2} \times \mathbb{R}^{I_3 \times R_3} \to \mathbb{R}^{I_1 \times R_1} \times \mathbb{R}^{I_2 \times R_2} \times \mathbb{R}^{I_3 \times R_3}, \\ & \mathbf{X} \mapsto (F_1(\mathbf{X}),\, F_2(\mathbf{X}),\, F_3(\mathbf{X})). \end{aligned} \tag{3}$$

Newton's method can be applied for finding the zeros of F. However, F_1 has an invariance property

$$F_1(\mathbf{XQ}) = F_1(\mathbf{X})\,\mathbf{Q}_1, \tag{4}$$

where $\mathbf{XQ} = (\mathbf{UQ}_1,\ \mathbf{VQ}_2,\ \mathbf{WQ}_3)$ and $\mathbf{Q}_i \in O_{R_i}, i = 1, 2, 3$ are orthogonal matrices. The functions F_2 and F_3 have similar properties, i.e.,

$$F(\mathbf{X}) = \mathbf{0} \quad \Longleftrightarrow \quad F(\mathbf{XQ}) = \mathbf{0}.$$

Thus, the zeros of F are not isolated, which means that the plain Newton method is expected to have difficulties (see, for example, [3, Prop. 2.1.2], [2]).

A solution to this problem is to combine equivalent solutions in one element and work on the obtained quotient manifold (see [3] for the general theory on optimization on matrix manifolds). For information on differential-geometric version of Newton's method see also [5]. If we perform as little quotienting as possible in order to isolate the zeros, we obtain the quotient set

$$M = \mathbb{R}_*^{I_1 \times R_1}/O_{R_1} \times \mathbb{R}_*^{I_2 \times R_2}/O_{R_2} \times \mathbb{R}_*^{I_3 \times R_3}/O_{R_3}. \tag{5}$$

$\mathbb{R}_*^{n \times p}$ is the set of all full-rank $(n \times p)$-matrices, $n \geq p$ and each element $[\mathbf{U}]$ of $\mathbb{R}_*^{I_1 \times R_1}/O_{R_1}$ is a set of all matrices that can be obtained by multiplying $\mathbf{U}$ from the right by an orthogonal matrix. Any two sets $[\mathbf{U}_1]$ and $[\mathbf{U}_2]$ are either disjoint or coincide and the union of all such sets equals $\mathbb{R}_*^{n \times p}$. They are called equivalence classes. In each equivalence class all elements have the same column space.

For our problem (2), working on the manifold M removes the invariance and leads to a differential-geometric Newton algorithm [47]. The Newton algorithm has local quadratic convergence to the nondegenerate zeros of the vector field ξ on M (5) represented by the horizontal lift $P^h F$,

$$P^h_{\mathbf{U}}(\mathbf{Z}_{\mathbf{U}}) = \mathbf{Z}_{\mathbf{U}} - \mathbf{U}\,\mathrm{skew}((\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{Z}_{\mathbf{U}})\,,$$

where $\mathrm{skew}(\mathbf{B}) = (\mathbf{B} - \mathbf{B}^T)/2$. If $\mathbf{X}_*$ is a zero of F (3), then $[\mathbf{X}_*]$ is a zero of ξ. Numerical results indicate that that nondegeneracy holds under generic conditions.

Numerical examples also confirmed the fast quadratic convergence of the algorithm in the neighborhood of the solution. However, the cost per iteration of the geometric Newton algorithm $O(I^3R^3)$ is higher than the cost $O(I^3R + IR^4 + R^6)$ for one HOOI iteration. Another possible disadvantage of the proposed algorithm is that it does not necessarily converge to a local maximum of (2) since not all zeros of F correspond to local maxima of (2). In theory, Newton's method can even diverge. However, this was not observed in numerical experiments. To increase the chances of converging to a maximum of (2), one can first perform an HOSVD followed by a few iterations of HOOI and additionally check for the negative definiteness of the Hessian before starting the Newton algorithm.

4.2 Trust-region based algorithm

Another iterative method for minimizing a cost function is the trust-region method [15, 58]. At each step, instead of working with the original function, a quadratic model is obtained. This model is assumed to be accurate in a neighborhood (the trust-region) of the current iterate. The solution of the quadratic minimization problem is suggested as a solution of the original problem. The quality of the updated iterate is evaluated and is accepted or rejected. The trust-region radius is also adjusted.

On a Riemannian manifold, the trust-region subproblem at a point $\mathbf{x} \in M$ is moved to the tangent plane $T_{\mathbf{x}}M$. The tangent plane is a Euclidean space so the minimization problem can be solved with standard algorithms. The update vector $\xi \in T_{\mathbf{x}}M$ is a tangent vector, giving the direction in which the next iterate is to be found and the size of the step. However, the new iterate has to be on the manifold and not on the tangent plane. The correspondence between vectors on the tangent plane and points on the manifold is given by a retraction [65, 5], Fig. 5.

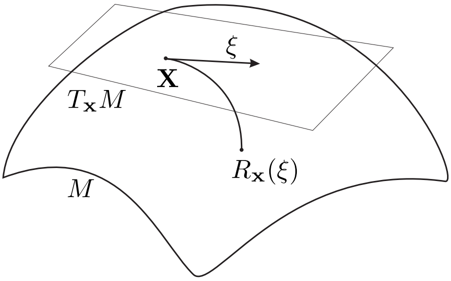


Fig. 5. Retraction.

The choice of retraction is important. The first obvious choice is the exponential map. However, depending on the manifold, this choice may be computationally inefficient [55]. A retraction can be thought of as a cheap approximation of the exponential map, without destroying the convergence behavior of the optimization methods.

As suggested in [70, 71], an approximate but sufficiently accurate solution to the trust-region subproblem (the minimization of the quadratic model) is given by the truncated conjugate gradient algorithm (tCG). An advantage here is that the Hessian matrix is not computed explicitly but only its application to a tangent vector is required. For other possible methods for (approximately) solving the trust-region subproblem see [57, 15].

Notice that $\overline{g}$ from (2) has the following invariance property

$$\overline{g}(\mathbf{U}, \mathbf{V}, \mathbf{W}) = \overline{g}(\mathbf{U}\mathbf{Q}_1, \mathbf{V}\mathbf{Q}_2, \mathbf{W}\mathbf{Q}_3)\,, \tag{6}$$

where $\mathbf{Q}_i \in O_{R_i}, i = 1, 2, 3$ are orthogonal matrices. This means that we are not interested in the exact elements of the matrices $\mathbf{U}, \mathbf{V}, \mathbf{W}$ but in the subspaces that their columns span. For the Newton algorithm in Section 4.1 we worked on the manifold defined in (5). Here we choose the Grassmann manifold which removes more unused information from the cost function. In (2) we optimize three matrices so we need the product manifold

$$M = St(R_1, I_1)/O_{R_1} \times St(R_2, I_2)/O_{R_2} \times St(R_3, I_3)/O_{R_3}, \tag{7}$$

which can be thought of as a product of three Grassmann manifolds. A natural choice of a retraction is [3, §4.1.2]

$$R_{\mathbf{X}O_p}(\mathbf{Z}) = \mathrm{qf}(\mathbf{X} + \overline{\mathbf{Z}})O_p\,, \tag{8}$$

where qf denotes the $\mathbf{Q}$ factor of the thin $\mathbf{QR}$ decomposition [34, §5.2] and $\mathbf{Z}$ is a tangent vector. This choice is also motivated by the fact that we are only interested in column spaces of the matrices $\mathbf{U}, \mathbf{V}$ and $\mathbf{W}$ from (2) and not in their actual values.

In order to apply the Riemannian trust-region scheme to the problem (2), we need to go through the "checklist" in [1, §5.1] and give closed-form expressions for all the necessary components. A summary of the first version of the trust-region algorithm has been proposed in [45]. The algorithm is described in detail in [46].

The trust-region method has superlinear convergence. On the other hand, the cost for one iteration $O(I^3R^3)$ is higher than the cost for one HOOI iteration $O(I^3R + IR^4 + R^6)$ [32, 47]. However, it should be taken into account that in applications, the multilinear rank is often much smaller than the dimensions of the tensor. Moreover, one can reduce the computational cost of the trust-region algorithm without losing its fast local convergence rate. This can be done by choosing a stopping criterion based on the gradient of the cost function for the inner iteration [1]. In this case, few inner tCG steps are taken when the current iterate is far away from the solution (when the gradient is large) and more inner tCG steps are taken close to the solution. Thus, the overall performance of the trust-region method is to be preferred to HOOI in many cases.

Newton-type methods (see [47, 32, 64] and Section 4.1) also have local quadratic convergence rate and their computational cost per iteration is of the same order as the one of the trust-region method. However, they are not globally convergent and strongly depend on the initialization point. Although the truncated HOSVD often gives good initial values, sometimes these values are not good enough. These methods might even diverge in practice. On the other hand, the trust-region method converges globally (i.e., for all initial points) to stationary points [1] except for very special examples that are artificially constructed. Moreover, since the trust-region method is decreasing the cost function at each step, convergence to saddle points or local maxima is not observed in practice. Newton methods do not distinguish between minima, maxima and saddle points. Thus, if the stationary points are close to each other, even if a relatively good starting point is chosen, these algorithms might converge to a maximum or to a saddle point instead of to a minimum.

4.3 Conjugate gradient based algorithm

The linear conjugate gradient (CG) method [39] is used for solving large systems of linear equations having a symmetric positive definite matrix. One can also regard CG as a method to minimize a convex quadratic cost function. The initial search direction is taken equal to the steepest descent direction. Every subsequent search direction is required to be conjugate to all previously generated search directions. The step length is chosen as the exact minimizer in the search direction and indicates where to take the next iterate. The optimal solution is found in n steps, where n is the dimension of the problem.

Nonlinear CG methods [33, 62] use the same idea as linear CG but apply it to general nonlinear functions. A few adjustments are necessary. The step size is obtained by a line search algorithm. The computation of the next search direction is not uniquely defined as in the linear CG. The main approaches are those provided by Fletcher-Reeves [33] and Polak-Ribière [62], both having advantages and disadvantages. The nonlinear CG methods reduce to the linear CG if the function is convex quadratic and if the step size is the exact minimizer along the search direction. However, since the cost function is in general not convex quadratic, convergence is obtained after more than n iterations. Some convergence results can be found in [58, §5] and the references therein.

In order to generalize the nonlinear CG from functions in $\mathbb{R}^n$ to functions defined on Riemannian manifolds, the expressions for the step length and search direction have to be adjusted. Exact line search for the step length could be extremely expensive. In that case, the step size could be computed using a backtracking procedure, searching for an Armijo point [3, §4.2].

When computing the new search direction η_{k+1}, another obstacle appears. The formula for η_{k+1} involves the gradient at the new point $\mathbf{x}_{k+1}$ and the previous search direction η_k, which are two vectors in two different tangent spaces. A solution for this problem is to carry η_k over to the tangent space of $\mathbf{x}_{k+1}$. Nonlinear CG on Riemannian manifolds was first proposed in [68, 69]. This algorithm makes use of the exponential map and parallel translation, which might be inefficient. The algorithm proposed in [3] works with the more general concepts of retraction and vector transport. The vector transport is a mapping that transports a tangent vector from one tangent plane to another. The vector transport has a different purpose than a retraction but is a similar concept in the sense that it is a cheap version of parallel translation, being just as useful as the parallel translation at the same time. We refer to [3, Def. 8.1.1] for the precise formulation. The concept is illustrated in Fig. 6. The vector ξ is transported to the tangent plane of $R_{\mathbf{x}}(\eta)$ and the result is $\mathcal{T}_\eta \xi$.

As in the trust-region algorithm, here, for solving (2) we work again on the Grassmann manifold. A simple vector transport in this case is

$$\overline{(\mathcal{T}_{\eta_{\mathbf{X}}}\xi_{\mathbf{X}})}_{\mathrm{qf}(\mathbf{X}+\overline{\eta}_{\mathbf{X}})} = P^h_{\mathrm{qf}(\mathbf{X}+\overline{\eta}_{\mathbf{X}})}\overline{\xi}_{\mathbf{X}}\,, \tag{9}$$

where $\eta_{\mathbf{X}}$ and $\xi_{\mathbf{X}}$ are two tangent vectors at point $[\mathbf{X}]$ and $\overline{\xi}_{\mathbf{X}}$ and $\overline{\eta}_{\mathbf{X}}$ are the horizontal lifts [3, §3.5.8] at $\mathbf{X}$ of $\xi_{\mathbf{X}}$ and $\eta_{\mathbf{X}}$ respectively. $P^h_{\mathbf{Y}}$ is the orthogonal

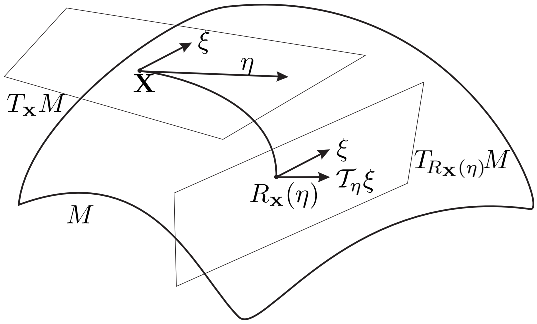


Fig. 6. Vector transport.

projection

$$P^h_{\mathbf{Y}}(\mathbf{Z}) = (\mathbf{I} - \mathbf{Y}\mathbf{Y}^T)\mathbf{Z}$$

onto the horizontal space of the point $\mathbf{Y}$. Note that $[\mathrm{qf}(\mathbf{X} + \overline{\eta}_{\mathbf{X}})] = R_{[\mathbf{X}]}\eta_x$.

Some remarks are in order. Since the step size is not the optimal one along η_k, it is possible that the new direction is not a descent direction. If this is the case, we set the new direction to be the steepest descent direction. A generalization of the computation of the search directions based on the Fletcher-Reeves and Polak-Ribière formulas is given in [3, §8.3]. The precision of CG was discussed in [3, 36]. When the distance between the current iterate and the local minimum is close to the square root of the machine precision, the Armijo condition within the line-search procedure can never be satisfied. This results in CG having maximum precision equal to the square root of the machine precision. To overcome this problem, an approximation of the Armijo condition was proposed in [36]. Finally, we mention that for better convergence results, it is advisable to "restart" the CG algorithm, i.e., to take as a search direction the steepest descent direction. This should be done at every n steps, where n is the number of unknown parameters, in order to erase unnecessary old information. The convergence of CG in $\mathbb{R}^n$ is then n-step quadratic. However, n is often too large in the sense that the algorithm already converges in less than n iterations.

The convergence properties of nonlinear CG methods are difficult to analyze. Under mild assumptions on the cost function, nonlinear CG converges to stationary points. Descent directions are guaranteed if we take the steepest descent direction when the proposed direction is not a descent direction itself. Thus, CG converges to local minima unless very special initial values are started from. The advantage of the nonlinear CG methods is their low computational cost and the fact that they do not require a lot of storage space. At each iteration, the cost function and the gradient are evaluated but the computation of the Hessian is not required, as it was the case for the trust-region algorithm from Section 4.2.

It is expected that the proposed geometric CG algorithm [43] has properties similar to those of nonlinear CG although theoretical results are difficult to prove. Numerical experiments indicate that the performance of CG strongly depends on the problem. If the tensor has a well-determined part with low multilinear rank, CG performs well. The difficulty of the problem is related to the distribution of the multilinear singular values of the original tensor. As far as the computational time is concerned, CG seems to be competitive with HOOI and the trust-region algorithm for examples that are not too easy and not too difficult, such as tensors with elements taken from a normal distribution with zero mean and unit standard deviation.

In our study of algorithms for the low multilinear rank approximation of tensors, it was important to investigate a CG-based algorithm. The convergence speed of the algorithm is not favorable but this is compensated by the fact that the iterations are extremely fast.

4.4 Remarks

HOOI is a simple algorithm with cheap iterations but linear convergence rate. This suggests to use it if the precision or the computational time are not critical. On the other hand, the Newton based algorithm has local quadratic convergence rate but has expensive iterations and convergence issues. Thus, this algorithm can be used if a *good* starting point is available. The trust-region based algorithm has also fast (up to quadratic) convergence rate and cost per iteration smaller or equal to the one of the Newton based algorithm. Its computational time per iteration is competitive with the one of HOOI for approximations with small multilinear rank. Finally, the CG based algorithm converges after a large amount of cheap iterations. The cost for one iteration is similar to the cost of one HOOI iteration. Numerical experiments suggest that the CG algorithm has best performance for easy problems, i.e., for approximations where the original tensor is close to a tensor with low multilinear rank. We summarize the most important features of the algorithms in Table 1. Some numerical examples can be found in [42].

	HOOI	Newton	TR	CG
global/local convergence	(global)	local	global	(global)
convergence to	min, (saddle point), ((max))	stationary point	min, (saddle point), ((max))	min, (saddle point), ((max))
local convergence speed	linear	quadratic	superlinear up to quadratic	$\left(\begin{array}{c} n\text{-step} \\ \text{quadratic} \end{array}\right)$
cost/iteration	$O(I^3R+IR^4+R^6)$	$O(I^3R^3)$	$\leq O(I^3R^3)$	$(\sim O(I^3R))$
monotonically decreasing?	yes	no	yes	yes

Table 1. Summary of the main features of HOOI, the Newton's algorithm, the trust-region algorithm and the conjugate gradient algorithm.

The low multilinear rank approximation problem (1) may have many local minima. Searching for distinct minima, all available algorithms could be run with a number of initial points. Because of the different functioning of the algorithms, they often find different solutions even if initialized in the same way.

5 Local minima

The best low multilinear rank approximation problem (1) has local minima [16, 23, 44, 42]. This is a key observation since the best low-rank approximation of a matrix has a unique minimum.

For tensors with low multilinear rank perturbed by a small amount of additive noise, algorithms converge to a small number of local minima. After increasing the noise level, the tensors become less structured and more local minima are found [44]. This behavior is related to the distribution of the mode-n singular values. In the first case, there is a large gap between the singular values. If the gap is small or nonexistent, the best low multilinear rank approximation is a difficult problem since we are looking for a structure that is not present. In this case, there are many equally good, or equally bad, solutions.

The values of the cost function at different local minima seem to be similar [44]. Thus, in applications where the multilinear rank approximation is merely used as a compression tool for memory savings, taking a nonglobal local minimum is not too different from working with the global minimum itself.

On the other hand, the column spaces of the matrices $\mathbf{U}_1$ and $\mathbf{U}_2$ corresponding to two different local minima are very different and the same holds for $\mathbf{V}$ and $\mathbf{W}$ [44]. In applications where these subspaces are important, local minima may be an issue. This concerns in particular the dimensionality reduction prior to computing a PARAFAC decomposition. One should inspect the gap between the mode-n singular values in each mode in order to choose meaningful values for the multilinear rank of the approximation.

An additional problem appears when the subspaces are important but the global minimum is not the desired one. This could happen when a tensor with low multilinear rank is affected by noise. The subspaces corresponding to the global minimum of (1) are not necessarily the closest to the subspaces corresponding to the original noise-free tensor, especially for high noise levels. This further stresses that solutions of the approximation problem have to be interpreted with care. It may even be impossible to obtain a meaningful solution.

It is usually a good idea to start from the truncated HOSVD. However, convergence to the global optimum is not guaranteed [16, 23, 44]. In some examples, a better (in the sense of yielding a smaller cost function value) local minimum is obtained from another initial point. Considering different algorithms with different initial values could improve the change to find the global minimum.

Finally, we describe a procedure for dimensionality reduction of large-scale problems. As an initial step, the HOSVD of the original tensor can be truncated so that the mode-n singular values close to zero be discarded. In this way, the dimensions of the original tensor are reduced without losing much precision. As a second step prior to computing e.g., a PARAFAC decomposition, an essential dimensionality reduction via low multilinear rank approximation on an already smaller scale can be performed. The latter needs to take into account gaps between mode-n singular values.

6 Conclusions

This paper combines several topics. The main problem, the best low multilinear rank approximation of higher-order tensors, is a key problem in multilinear algebra having various applications. We considered solutions based on optimization on manifolds. The fact that the cost function is invariant under right multiplication of the matrices $\mathbf{U}, \mathbf{V}$ and $\mathbf{W}$ by orthogonal matrices prohibits potential algorithms from converging to a particular solution. Working on quotient manifolds isolates the solutions and makes the work of "standard" optimization algorithms easier.

The optimization methods on which the discussed methods are based are Newton's method, trust-region and conjugate gradients. There are also other methods in the literature. It is difficult to say which algorithm is the best. All algorithms have their advantages and disadvantages. Depending on the application, the dimensions of the tensor, the required precision and the time restrictions, one of the algorithms can be the method of choice. The Newton algorithm has local quadratic convergence rate but might diverge or converge to a saddle point or a maximum instead of a minimum. Moreover, it needs a good starting point. A well-chosen stopping criterion for the inner iteration of the trust-region algorithm leads to an algorithm with local quadratic convergence. The computational cost per iteration is competitive with the one of HOOI, which has only linear local convergence. Moreover, convergence of the trust-region algorithm to a minimum is (almost always) guaranteed. On the other hand, the conjugate gradient based algorithm has much cheaper iterations but lacks solid theoretical proofs.

It can make sense to apply several algorithms to the same problem. For example, if one wishes to inspect several local minima, one strategy would be to run all available algorithms, starting from enough initial points and in this way to obtain a more complete set of solutions. Due to the different character of the algorithms, they often find different solutions even when starting from the same initial values.

We also discussed the issue of local minima of the low multilinear rank approximation problem. It concerns the problem itself and does not depend on the actual algorithm. There are important consequences for whole classes of applications. One should be very careful when deciding whether or not it is meaningful to use such an approximation. The higher-order singular values may provide relevant information in this respect.

References

1. P.-A. Absil, C. G. Baker, K. A. Gallivan. Trust-region methods on Riemannian manifolds. *Found. Comput. Math.*, 7(3):303–330, 2007.
2. P.-A. Absil, M. Ishteva, L. De Lathauwer, S. Van Huffel. A geometric Newton method for Oja's vector field. *Neural Comput.*, 21(5):1415–1433, 2009.
3. P.-A. Absil, R. Mahony, R. Sepulchre. *Optimization Algorithms on Matrix Manifolds.* Princeton University Press, Princeton, NJ, 2008.
4. E. Acar, C. A. Bingol, H. Bingol, R. Bro, B. Yener. Multiway analysis of epilepsy tensors. *ISMB 2007 Conference Proc., Bioinformatics*, 23(13):i10–i18, 2007.
5. R. L. Adler, J.-P. Dedieu, J. Y. Margulies, M. Martens, M. Shub. Newton's method on Riemannian manifolds and a geometric model for the human spine. *IMA J. Numer. Anal.*, 22(3):359–390, 2002.
6. C. A. Andersson, R. Bro. The N-way toolbox for MATLAB. *Chemometrics and Intelligent Laboratory Systems*, 52(1):1–4, 2000. See also http://www.models.kvl.dk/source/nwaytoolbox/.
7. R. Badeau, R. Boyer. Fast multilinear singular value decomposition for structured tensors. *SIAM J. Matrix Anal. Appl.*, 30(3):1008–1021, 2008.
8. C. Caiafa, A. Cichocki. Reconstructing matrices and tensors from few rows and columns. In *Proc. of 2009 International Symposium on Nonlinear Theory and its Applications*, 2009. In press.
9. J. Carroll, J. Chang. Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika*, 35(3):283–319, 1970.
10. J. Chen, Y. Saad. On the tensor SVD and the optimal low rank orthogonal approximation of tensors. *SIAM J. Matrix Anal. Appl.*, 30(4):1709–1734, 2009.
11. A. Cichocki, R. Zdunek, A. H. Phan, and S.-I. Amari. *Nonnegative Matrix and Tensor Factorizations.* Wiley, 2009.
12. P. Comon. Independent component analysis, a new concept? *Signal Processing*, 36(3):287–314, 1994.
13. P. Comon. Tensor decompositions. In J. G. McWhirter, I. K. Proudler (eds), *Mathematics in Signal Processing V*, pp. 1–24. Clarendon Press, Oxford, 2002.
14. P. Comon, G. Golub, L.-H. Lim, B. Mourrain. Symmetric tensors and symmetric tensor rank. *SIAM J. Matrix Anal. Appl.*, 30(3):1254–1279, 2008.
15. A. R. Conn, N. I. M. Gould, P. L. Toint. *Trust-Region Methods.* MPS-SIAM Series on Optimization. SIAM, Philadelphia, PA, 2000.
16. L. De Lathauwer. *Signal Processing Based on Multilinear Algebra.* PhD thesis, Dept. of Electrical Engineering, Katholieke Universiteit Leuven, 1997.
17. L. De Lathauwer. A link between the canonical decomposition in multilinear algebra and simultaneous matrix diagonalization. *SIAM J. Matrix Anal. Appl.*, 28(3):642–666, 2006.
18. L. De Lathauwer. Decompositions of a higher-order tensor in block terms — Part I: Lemmas for partitioned matrices. *SIAM J. Matrix Anal. Appl.*, 30(3): 1022–1032, 2008.
19. L. De Lathauwer. Decompositions of a higher-order tensor in block terms — Part II: Definitions and uniqueness. *SIAM J. Matrix Anal. Appl.*, 30(3):1033–1066, 2008.

20. L. De Lathauwer. A survey of tensor methods. In *Proc. of the 2009 IEEE International Symposium on Circuits and Systems (ISCAS 2009)*, pp. 2773–2776, Taipei, Taiwan, 2009.
21. L. De Lathauwer, J. Castaing. Tensor-based techniques for the blind separation of DS-CDMA signals. *Signal Processing*, 87(2):322–336, 2007.
22. L. De Lathauwer, B. De Moor, J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21(4):1253–1278, 2000.
23. L. De Lathauwer, B. De Moor, J. Vandewalle. On the best rank-1 and rank-$(R_1, R_2, \ldots, R_N)$ approximation of higher-order tensors. *SIAM J. Matrix Anal. Appl.*, 21(4):1324–1342, 2000.
24. L. De Lathauwer, B. De Moor, J. Vandewalle. Independent component analysis and (simultaneous) third-order tensor diagonalization. *IEEE Trans. Signal Process.*, 49(10):2262–2271, 2001.
25. L. De Lathauwer, B. De Moor, J. Vandewalle. Computation of the canonical decomposition by means of a simultaneous generalized Schur decomposition. *SIAM J. Matrix Anal. Appl.*, 26(2):295–327, 2004.
26. L. De Lathauwer, D. Nion. Decompositions of a higher-order tensor in block terms — Part III: Alternating least squares algorithms. *SIAM J. Matrix Anal. Appl.*, 30(3):1067–1083, 2008.
27. L. De Lathauwer, J. Vandewalle. Dimensionality reduction in higher-order signal processing and rank-$(R_1, R_2, \ldots, R_N)$ reduction in multilinear algebra. *Linear Algebra Appl.*, 391:31–55, 2004.
28. V. de Silva, L.-H. Lim. Tensor rank and the ill-posedness of the best low-rank approximation problem. *SIAM J. Matrix Anal. Appl.*, 30(3):1084–1127, 2008.
29. M. De Vos, L. De Lathauwer, B. Vanrumste, S. Van Huffel, W. Van Paesschen. Canonical decomposition of ictal scalp EEG and accurate source localization: Principles and simulation study. *Journal of Computational Intelligence and Neuroscience*, 2007(Article ID 58253):1–10, 2007.
30. M. De Vos, A. Vergult, L. De Lathauwer, W. De Clercq, S. Van Huffel, P. Dupont, A. Palmini, W. Van Paesschen. Canonical decomposition of ictal scalp EEG reliably detects the seizure onset zone. *NeuroImage*, 37(3):844–854, 2007.
31. M. Elad, P. Milanfar, G. H. Golub. Shape from moments — an estimation theory perspective. *IEEE Trans. on Signal Processing*, 52(7):1814–1829, 2004.
32. L. Eldén, B. Savas. A Newton–Grassmann method for computing the best multi-linear rank-(r_1, r_2, r_3) approximation of a tensor. *SIAM J. Matrix Anal. Appl.*, 31(2):248–271, 2009.
33. R. Fletcher, C. M. Reeves. Function minimization by conjugate gradients. *Comput. J.*, 7:149–154, 1964.
34. G. H. Golub, C. F. Van Loan. *Matrix Computations.* Johns Hopkins University Press, Baltimore, Maryland, 3rd edition, 1996.
35. M. Haardt, F. Roemer, G. Del Galdo. Higher-order SVD-based subspace estimation to improve the parameter estimation accuracy in multidimensional harmonic retrieval problems. *IEEE Trans. on Signal Processing*, 56(7):3198–3213, 2008.
36. W. W. Hager, H. Zhang. A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM Journal on Optimization*, 16(1):170–192, 2005.

37. R. A. Harshman. Foundations of the PARAFAC procedure: Model and conditions for an “explanatory” multi-mode factor analysis. *UCLA Working Papers in Phonetics*, 16(1):1–84, 1970.
38. U. Helmke, J. B. Moore. *Optimization and Dynamical Systems.* Springer-Verlag, 1993.
39. M. R. Hestenes, E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Research Nat. Bur. Standards*, 49:409–436 (1953), 1952.
40. F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematical Physics*, 6(1):164–189, 1927.
41. F. L. Hitchcock. Multiple invariants and generalized rank of a p-way matrix or tensor. *Journal of Mathematical Physics*, 7(1):39–79, 1927.
42. M. Ishteva. *Numerical methods for the best low multilinear rank approximation of higher-order tensors.* PhD thesis, Dept. of Electrical Engineering, Katholieke Universiteit Leuven, 2009.
43. M. Ishteva, P.-A. Absil, S. Van Huffel, L. De Lathauwer. Best low multilinear rank approximation with conjugate gradients. Tech. Rep. 09-246, ESAT-SISTA, K.U.Leuven, Belgium, 2009.
44. M. Ishteva, P.-A. Absil, S. Van Huffel, L. De Lathauwer. Tucker compression and local optima. Tech. Rep. UCL-INMA-2010.012, Université catholique de Louvain and 09-247, ESAT-SISTA, K.U.Leuven, Belgium, 2010.
45. M. Ishteva, L. De Lathauwer, P.-A. Absil, S. Van Huffel. Dimensionality reduction for higher-order tensors: algorithms and applications. *International Journal of Pure and Applied Mathematics*, 42(3):337–343, 2008.
46. M. Ishteva, L. De Lathauwer, P.-A. Absil, S. Van Huffel. Best low multilinear rank approximation of higher-order tensors, based on the Riemannian trust-region scheme. Tech. Rep. 09-142, ESAT-SISTA, K.U.Leuven, Belgium, 2009.
47. M. Ishteva, L. De Lathauwer, P.-A. Absil, S. Van Huffel. Differential-geometric Newton method for the best rank-(R_1, R_2, R_3) approximation of tensors. *Numerical Algorithms*, 51(2):179–194, 2009.
48. M. Ishteva, L. De Lathauwer, S. Van Huffel. Comparison of the performance of matrix and tensor based multi-channel harmonic analysis. In *7th International Conf. on Mathematics in Signal Processing, Cirencester, UK*, pp. 77–80, 2006.
49. E. Kofidis, P. A. Regalia. On the best rank-1 approximation of higher-order supersymmetric tensors. *SIAM J. Matrix Anal. Appl*, 23(3):863–884, 2002.
50. T. Kolda. Orthogonal tensor decompositions. *SIAM J. Matrix Anal. Appl.*, 23:243–255, 2001.
51. T. G. Kolda, B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.
52. P. M. Kroonenberg. *Applied Multiway Data Analysis.* Wiley, 2008.
53. P. M. Kroonenberg, J. de Leeuw. Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 45(1):69–97, 1980.
54. M. W. Mahoney, M. Maggioni, P. Drineas. Tensor-CUR decompositions for tensor-based data. *SIAM J. Matrix Anal. Appl.*, 30(3):957–987, 2008.
55. J. H. Manton. Optimization algorithms exploiting unitary constraints. *IEEE Trans. Signal Process.*, 50(3):635–650, 2002.
56. C. D. Moravitz Martin, C. F. Van Loan. A Jacobi-type method for computing orthogonal tensor decompositions. *SIAM J. Matrix Anal. Appl.*, 30(3):1219–1232, 2008.

57. J. J. Moré, D. C. Sorensen. Computing a trust region step. *SIAM J. Sci. Statist. Comput.*, 4(3):553–572, 1983.
58. J. Nocedal, S. J. Wright. *Numerical Optimization.* Springer Verlag, New York, 2nd edition, 2006. Springer Series in Operations Research.
59. I. V. Oseledets, D. V. Savostianov, E. E. Tyrtyshnikov. Tucker dimensionality reduction of three-dimensional arrays in linear time. *SIAM J. Matrix Anal. Appl.*, 30(3):939–956, 2008.
60. J.-M. Papy, L. De Lathauwer, S. Van Huffel. Exponential data fitting using multilinear algebra: The single-channel and the multichannel case. *Numer. Linear Algebra Appl.*, 12(8):809–826, 2005.
61. J.-M. Papy, L. De Lathauwer, S. Van Huffel. Exponential data fitting using multilinear algebra: The decimative case. *J. Chemometrics*, 23(7–8):341–351, 2009.
62. E. Polak, G. Ribière. Note sur la convergence de méthodes de directions conjuguées. *Rev. Française Informat. Recherche Opérationnelle*, 3(16):35–43, 1969.
63. B. Savas, L. Eldén. Krylov subspace methods for tensor computations. Tech. Rep. LITH-MAT-R-2009-02-SE, Dept. of Mathematics, Linköping University, 2009.
64. B. Savas, L.-H. Lim. Best multilinear rank approximation of tensors with quasi-Newton methods on Grassmannians. Tech. Rep. LITH-MAT-R-2008-01-SE, Dept. of Mathematics, Linköping University, 2008.
65. M. Shub. Some remarks on dynamical systems and numerical analysis. In L. Lara-Carrero, J. Lewowicz (eds), *Proc. VII ELAM.*, pp. 69–92. Equinoccio, U. Simón Bolívar, Caracas, 1986.
66. N. Sidiropoulos, R. Bro, G. Giannakis. Parallel factor analysis in sensor array processing. *IEEE Trans. Signal Process.*, 48:2377–2388, 2000.
67. A. Smilde, R. Bro, P. Geladi. *Multi-way Analysis. Applications in the Chemical Sciences.* John Wiley and Sons, Chichester, U.K., 2004.
68. S. T. Smith. *Geometric Optimization Methods for Adaptive Filtering.* PhD thesis, Division of Applied Sciences, Harvard University, Cambridge, MA, 1993.
69. S. T. Smith. Optimization techniques on Riemannian manifolds. In A. Bloch (ed), *Hamiltonian and gradient flows, algorithms and control*, volume 3 of *Fields Inst. Commun.*, pp. 113–136. Amer. Math. Soc., Providence, RI, 1994.
70. T. Steihaug. The conjugate gradient method and trust regions in large scale optimization. *SIAM J. Numer. Anal.*, 20(3):626–637, 1983.
71. P. L. Toint. Towards an efficient sparsity exploiting Newton method for minimization. In I. S. Duff (ed), *Sparse Matrices and Their Uses*, pp. 57–88. Academic Press, London, 1981.
72. L. R. Tucker. The extension of factor analysis to three-dimensional matrices. In H. Gulliksen, N. Frederiksen (eds), *Contributions to mathematical psychology*, pp. 109–127. Holt, Rinehart & Winston, NY, 1964.
73. L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.
74. L. Vanhamme, S. Van Huffel. Multichannel quantification of biomedical magnetic resonance spectroscopy signals. In F. Luk (ed), *Proc. of SPIE, Advanced Signal Processing Algorithms, Architectures, and Implementations VIII*, volume 3461, pp. 237–248, San Diego, California, 1998.
75. T. Zhang, G. H. Golub. Rank-one approximation to high order tensors. *SIAM J. Matrix Anal. Appl.*, 23:534–550, 2001.

Refining Sparse Principal Components

M. Journée[1], F. Bach[2], P.-A. Absil[3], and R. Sepulchre[1]

[1] Department of Electrical Engineering and Computer Science, University of Liège, Belgium, [M.Journee, R.Sepulchre]@ulg.ac.be

[2] INRIA - Willow project, Dèpartement d'Informatique, Ecole Normale Supérieure, Paris, France, Francis.Bach@mines.org

[3] Université catholique de Louvain, 1348 Louvain-la-Neuve, Belgium, absil@inma.ucl.ac.be

Summary. In this paper, we discuss methods to refine locally optimal solutions of sparse PCA. Starting from a local solution obtained by existing algorithms, these methods take advantage of convex relaxations of the sparse PCA problem to propose a refined solution that is still locally optimal but with a higher objective value.

1 Introduction

Principal component analysis (PCA) is a well-established tool for making sense of high dimensional data by reducing it to a smaller dimension. Its extension to *sparse principal component analysis*principal component analysis!sparce, which provides a sparse low-dimensional representation of the data, has attracted alot of interest in recent years (see, e.g., [1, 2, 3, 5, 6, 7, 8, 9]). In many applications, it is in fact worth to sacrifice some of the explained variance to obtain components composed only from a small number of the original variables, and which are therefore more easily interpretable.

Although PCA is, from a computational point of view, equivalent to a singular value decomposition, sparse PCA is a much more difficult problem of NP-hard complexity [8]. Given a data matrix $A \in \mathbf{R}^{m \times n}$ encoding m samples of n variables, most algorithms for sparse PCA compute a unit-norm loading vector $z \in \mathbf{R}^n$ that is only *locally* optimal for an optimization problem aiming at maximizing explained variance penalized for the number of non-zero loadings. This is in particular the case of the SCoTLASS [7], the SPCA [10], the rSVD [9] and the GPower [5] algorithms.

Convex relaxationsconvex relaxation have been proposed in parallel for some of these formulations [2, 1]. To this end, the unit-norm loading vector $z \in \mathbf{R}^n$ is lifted into a symmetric, positive semidefinite, rank-one matrix $Z = zz^T$ with unit trace. The relaxation consists of removing the rank-one constraintrank-one!constraint and accepting any matrix of the *spectahedron*spectahedron

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_14, © Springer-Verlag Berlin Heidelberg 2010

$$\mathcal{S} = \{Z \in \mathbf{S}^m \mid Z \succeq 0, \mathrm{Tr}(Z) = 1\},$$

which is a convex set. The solution of these convex problems has usually a rank larger than one. Hence, some post-processing is needed to round this solution to rank-onerank-one!matrices in order to reconstruct a unit-norm vector z.

The aim of this paper is to discuss a way to refine locally optimal solutions of sparse PCA by taking advantage of these convex relaxations. A well-known formulation of sparse PCA is first reviewed and relaxed into a convex program in Section 2. A method that uses both the initial formulation and the relaxation is then discussed in Section 3 in order to improve the quality of the components. Its efficiency is evaluated in Section 4.

2 Formulation and convex relaxation of sparse PCA

Under the assumption that the columns of the data matrix $A \in \mathbf{R}^{m\times n}$ are centered, PCA consists in computing the dominant eigenvectors of the scaled sample covariance matrix $\varSigma = A^T A$. The problem of computing the first principal component can thus be written in the form

$$\max_{\substack{z\in\mathbf{R}^n \\ z^T z=1}} \; z^T \varSigma z. \tag{1}$$

Several formulations of sparse PCA can be derived from (1) (see, e.g., [5]). A possible one is provided by the optimization problem

$$z^* = \arg \max_{\substack{z\in\mathbf{R}^n \\ z^T z=1}} \; z^T \varSigma z - \rho \|z\|_0, \tag{2}$$

with $\rho \geq 0$ and where the ℓ_0 "norm" is the number of nonzero coefficients (or *cardinality*) of z. The formulation (2) is essentially the problem of finding an optimal pattern of zeros and nonzeros for the vector z, which is of combinatorial complexity.

Interestingly, as shown in [2, 5], problem (2) can be equivalently rewritten as the maximization of a convex function on the unit Euclidean sphere,

$$x^* = \arg \max_{\substack{x\in\mathbf{R}^m \\ x^T x=1}} \sum_{i=1}^{n} ((a_i^T x)^2 - \rho)_+, \tag{3}$$

where a_i is the ith column of A and $x_+ = \max(0, x)$. The solution z^* of (2) is reconstructed from the solution x^* of (3) as follows,

$$z^* = \frac{[\mathrm{sign}((A^T x^*) \circ (A^T x^*) - \rho)]_+ \circ A^T x^*}{\|[\mathrm{sign}((A^T x^*) \circ (A^T x^*) - \rho)]_+ \circ A^T x^* \|_2},$$

where $\circ$ denotes the matrix element-wise product. The ith component of z^* is thus active (i.e., not constrained to zero) if the condition $(a_i^T x^*)^2 - \rho \geq 0$ holds.

For the purpose of relaxing (2) into a convex program, the unit-norm vector x is lifted into a matrix $X = xx^T$. The formulation (3) is so rewritten in terms of a matrix variable X as follows,

$$\begin{array}{ll} \max\limits_{X \in \mathbf{S}^m} & \sum_{i=1}^n (a_i^T X a_i - \rho)_+ \\ \text{s.t.} & \operatorname{Tr}(X) = 1, \\ & X \succeq 0, \\ & \operatorname{rank}(X) = 1, \end{array} \tag{4}$$

where $\mathbf{S}^m$ denotes the set of symmetric matrices in $\mathbf{R}^{m \times m}$.The problem (4) is relaxed into a convex program in two steps. First, the nonconvex rank constraint is removed. Then, the convex objective function

$$f_{cvx}(X) = \sum_{i=1}^n (a_i^T X a_i - \rho)_+$$

is replaced by the concave function

$$f_{ccv}(X) = \sum_{i=1}^n \operatorname{Tr}(X^{\frac{1}{2}}(a_i a_i^T - \rho I) X^{\frac{1}{2}})_+,$$

where $\operatorname{Tr}(X)_+$ denotes the sum of the positive eigenvalues of X. Observe that maximizing a concave function over a convex set is indeed a convex program. Since the values $f_{cvx}(X)$ and $f_{ccv}(X)$ are equal for matrices X that are rank one, the convex relaxation of the sparse PCA formulation (2),

$$\begin{array}{ll} \max\limits_{X \in \mathbf{S}^m} & \sum_{i=1}^n \operatorname{Tr}(X^{\frac{1}{2}}(a_i a_i^T - \rho I) X^{\frac{1}{2}})_+ \\ \text{s.t.} & \operatorname{Tr}(X) = 1, \\ & X \succeq 0, \end{array} \tag{5}$$

is tight for solutions of rank one. We refer to [1] for more details on the derivation of (5).

3 A procedure to refine the components

Several methods have been proposed to compute locally optimal solutions of the NP-hard formulation (2) of sparse PCA. For instance, the greedy algorithm of [2] sequentially increments the cardinality of the solution with the component of z that maximizes the objective function in (2). The GPower algorithm of [5] exploits the convexity of the objective function to generalize the well-known power method in the present context.

In parallel, a method for solving the convex relaxation (5) in an efficient manner is discussed in the recent paper [4]. This method parameterizes the positive semidefinite matrix variable X as the product $X = YY^T$ where the number of independent columns of $Y \in \mathbf{R}^{m \times p}$ fixes the rank of X. The parameter p enables to interpolate between the initial combinatorial problem (i.e., $p = 1$) and the convex relaxation (i.e., $p = n$). In practice, the dimension p is incremented until a sufficient condition is satisfied for Y to provide a solution YY^T of (5). Since this often holds for $p \ll n$, the reduction of per-iteration numerical complexity for solving (5) can be significant: from $\mathcal{O}(n^2)$ for traditional convex optimization tools to $\mathcal{O}(np)$ for the algorithm of [4].

Starting from a locally optimal solution of the sparse PCA formulation (2), the proposed method for improving the quality of this solution works in two steps. First, solve the convex relaxation (5) with the algorithm of [4] that increases the rank of the variable X from one until a sufficiently accurate solution is found. Then, in order to recover a rank-one matrix from this solution of rank $p \geq 1$, solve the optimization problem,

$$\begin{array}{ll} \max\limits_{X \in \mathbf{S}^m} & \mu f_{cvx}(X) + (1-\mu) f_{ccv}(X) \\ \text{s.t.} & \operatorname{Tr}(X) = 1, \\ & X \succeq 0, \end{array} \tag{6}$$

for the parameter μ that is gradually increased from 0 to 1. In the case $\mu = 0$, (6) is the convex relaxation (5). In the other limit case $\mu = 1$, problem (6) amounts to maximize a convex function on a convex set, which has local solutions at all the extreme points of this set. Solving a sequence of problems of the form of (6) for an increasing value of μ from zero to one converges to the extreme points of the spectahedron that are all rank-one matrices. Hence, this process reduces the rank of the solution of the convex relaxation (5) from $p \geq 1$ to one. This rank-one solution is hoped to have a larger objective value than the rank-one matrix chosen to initialize the resolution of (5). The algorithm of [4] can be used to solve (6) for any value of μ.

Figure 1 illustrates the proposed procedure in the case of a random Gaussian matrix $A \in \mathbb{R}^{150 \times 50}$. Because any matrix of the spectahedron has non-negative eigenvalues with the sum being one, the maximum eigenvalue can be used to monitor the rank: a matrix of the spectahedron is rank one if and only if its maximum eigenvalue is one. The homotopy methodHomotopy method (i.e., solving (6) for an increasing value of μ) is compared against the best rank-one least squares approximation of the solution of (5), i.e., the matrix $\tilde{X} = xx^T$ where x is the unit-norm dominant eigenvector of X. Let $f_{EVD}(X)$ denote the function

$$f_{EVD}(X) = f_{ccv}(\tilde{X}) = f_{cvx}(\tilde{X}).$$

The continuous plots of Figure 1 display the evolution of both functions $f_{ccv}(X)$ and $f_{EVD}(X)$ during the resolution of the convex program (5), i.e., $\mu = 0$ in (6). Point A represents a rank-one solution that is locally optimal for the sparse PCA formulation (2) and obtained, for instance, with the GPower algorithm [4]. When solving the convex relaxation (5), the rank of the matrix X is gradually incremented until a solution is identified (point B/B'). The dashed plots illustrate the resolution of (6) for a parameter μ that is gradually increased from 0 to 1 (by steps of 0.05). For a sufficiently large value of μ, problem (6) has a rank-one solution (point C). The objective value in C is clearly larger than that of the initialization A as well as than that of the best rank-one least-squares approximation B'. This improvement results most probably from the fact that the homotopy method takes the objective function into account whereas the least-squares approximation does not.

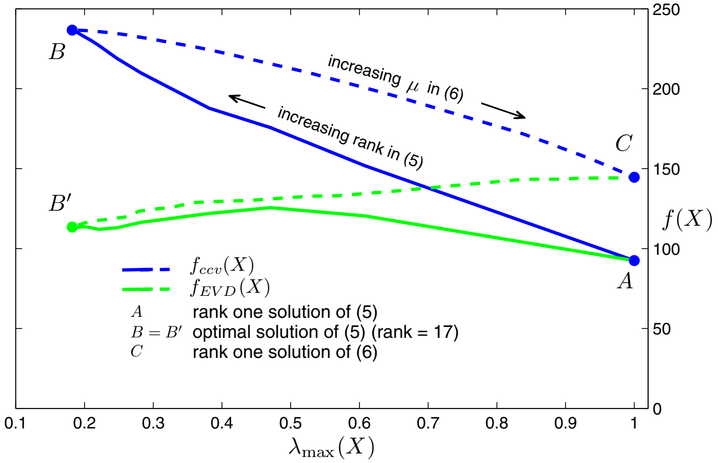


Fig. 1. Evolution of the functions $f_{ccv}(X)$ and $f_{EVD}(X)$ in two situations. Continuous plots: resolution of the convex program (5) ($\mu = 0$ in (6)). Dashed plots: projection of the solution of (5) on a rank-one matrix by gradual increase of μ in (6).

4 Numerical experiments

In Table 1, we compare the objective value obtained by the GPower algorithm which computes a locally optimal solution of the sparse PCA problem (3), the objective value of the best rank-one approximation of the solution of the convex relaxation (5) and finally the objective value of the proposed homotopy method, i.e., we compare the objective values at the points A, B' and C in Figure 1. Each value in Table 1 is an average on 20 instances for each problem dimension. The data is systematically generated according to a Gaussian distribution of zero mean and unit variance. The proposed homotopy method is shown to improve the objective value by several percents. Such an improvement might be significant for applications for which it is crucial to identify the best solution of sparse PCA. Compressed sensing is such an application [1].

Dimension	f_A	$f_{B'}$	$(f_{B'} - f_A)/f_A$	f_C	$(f_C - f_A)/f_A$
50 × 25	3.9757	4.0806	+ 2.64 %	4.1216	+ 3.67 %
100 × 50	3.6065	3.7038	+ 2.70 %	3.8276	+ 6.13 %
200 × 100	2.9963	2.8711	- 4.18 %	3.1904	+6.48 %
400 × 200	3.9549	4.1089	+3.89 %	4.2451	+ 7.34 %
800 × 200	5.6032	5.6131	+0.18 %	5.8754	+ 4.86 %
800 × 400	3.0541	3.0688	+ 0.48 %	3.4014	+11.37 %

Table 1. Average objective values at the points A, B' and C of Figure (1) for Gaussian data matrices of various size. The GPower algorithm of [5] is used to compute the rank-one solution A.

5 Acknowledgements

Michel Journée is a research fellow of the Belgian National Fund for Scientific Research (FNRS). This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors.

References

1. A. d' Aspremont, F. R. Bach, and L. El Ghaoui. Optimal solutions for sparse principal component analysis. *Journal of Machine Learning Research*, 9:1269–1294, 2008.
2. A. d' Aspremont, L. El Ghaoui, M. I. Jordan, and G. R. G. Lanckriet. A direct formulation for sparse PCA using semidefinite programming. *SIAM Review*, 49:434–448, 2007.
3. J. Cadima and I. T. Jolliffe. Loadings and correlations in the interpretation of principal components. *Journal of Applied Statistics*, 22:203–214, 1995.
4. M. Journée, F. Bach, P.-A. Absil, and R. Sepulchre. Low-rank optimization for semidefinite convex problems. *Submitted to SIAM Journal on Optimization (preprint available on ArXiv)*, 2008.
5. M. Journée, Y. Nesterov, P. Richtárik, and R. Sepulchre. Generalized power method for sparse principal component analysis. *Accepted to Journal of Machine Learning Research (preprint available on ArXiv)*, 2008.
6. I. T. Jolliffe. Rotation of principal components: choice of normalization constraints. *Journal of Applied Statistics*, 22:29–35, 1995.
7. I. T. Jolliffe, N. T. Trendafilov, and M. Uddin. A modified principal component technique based on the LASSO. *Journal of Computational and Graphical Statistics*, 12(3):531–547, 2003.

8. B. Moghaddam, Y. Weiss, and S. Avidan. Spectral bounds for sparse PCA: Exact and greedy algorithms. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 915–922. MIT Press, Cambridge, MA, 2006.
9. H. Shen and J. Z. Huang. Sparse principal component analysis via regularized low rank matrix approximation. *Journal of Multivariate Analysis*, 99(6):1015–1034, 2008.
10. H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286, 2006.

Optimal Data Fitting on Lie Groups: a Coset Approach

C. Lageman[1] and R. Sepulchre[2]

[1] Institut für Mathematik
Universität Würzburg
Am Hubland
97074 Würzburg, Germany
christian.lageman@mathematik.uni-wuerzburg.de

[2] Department of Electrical Engineering and Computer Science, B28
Université de Liège
B-4000 Liège Sart-Tilman, Belgium
r.sepulchre@ulg.ac.be

Summary. This work considers the problem of fitting data on a Lie group by a coset of a compact subgroup. This problem can be seen as an extension of the problem of fitting affine subspaces in $\mathbb{R}^n$ to data which can be solved using principal component analysis. We show how the fitting problem can be reduced for biinvariant distances to a generalized mean calculation on an homogeneous space. For biinvariant Riemannian distances we provide an algorithm based on the Karcher mean gradient algorithm. We illustrate our approach by some examples on $SO(n)$.

1 Introduction

In this paper we consider the problem of fitting a submanifold to data points on a Lie group. Such fitting problems are relevant for dimension reduction and statistical analysis of data on Lie groups. In Euclidean space it is well-known that the best fitting k-dimensional linear subspace can be computed via principal component analysis (PCA) and this tool is widely used in applications in natural sciences, statistics and engineering.

However, in some applications the data naturally arises as points on an embedded or abstract manifold, e.g. points on spheres [2] or manifolds of shape representations [4, 5]. This raises the question of extending subspace fitting and dimension reduction methods like PCA to nonlinear spaces like Riemannian manifolds and Lie groups. In the recent years some approaches have been proposed to construct local extensions of PCA [4, 5] on Riemannian manifolds or to consider fitting by single geodesics and interpolation problems on manifolds [7, 8]. Here, we focus on compact Lie groups and propose the different

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_15, © Springer-Verlag Berlin Heidelberg 2010

approach to fit a coset to the data. Our approach overcomes some limitations of the local approaches and leads to potentially efficient computational algorithms.

In Section 2 we recall basic facts on PCA. Section 3 discusses principal geodesic analysis from [4, 5]. Section 4 introduces our fitting of cosets approach and shows how it leads to a reduced optimization problem on a homogeneous space. For Riemannian distances we derive an algorithm based on known Karcher mean algorithms. Section 5 provides examples for fitting on $SO(n)$.

Notation

In this paper G will always denote a compact, connected Lie group. For more background on differential geometry, Lie groups etc. we refer to [1]. Recall that given a closed subgroup $H \subset G$ the quotient space G/H carries naturally a manifold structure. A Riemannian metric on G is called left- resp. right-invariant if it is invariant under the action of G on itself by left- resp. right-multiplication, i.e. for all $p, q \in G$, $v, w \in T_pG$ we have $\langle T_pL_qv, T_pL_qw\rangle = \langle v, w\rangle$ with L_q the left multiplication map $L_q(p) = qp$, analogously for the right-invariant case. A Riemannian metric is called biinvariant if it is left- and right-invariant. It can be shown that on any compact Lie group a biinvariant Riemannian metric exists. This is not the case for non-compact groups.

Furthermore, we recall the definition of a Karcher mean on a Riemannian manifold. Let M be a Riemannian manifold with Riemannian distance dist_R. The *Karcher mean* of points $q_1, \ldots, q_k$ on M is defined [10] as a minimum of the function $f(x) = \sum_{i=1}^k \mathrm{dist}_R(q_i, x)^2$. Note that a Karcher mean does not have to be unique.

2 Principal Component Analysis

In Euclidean spaces the most common method for dimension reduction of data is principal component analysis (PCA). We recall some basic facts on PCA, for a detailed account see the numerous literature on this topic, e.g. [3].

Given k data points $q_1, \ldots, q_k \in \mathbb{R}^n$, the problem is to determine an affine subspace $p + V$ of dimension m such that the sum of squared Euclidean distances

$$\textstyle\sum_{i=1}^k \min_{v \in p+V} \|q_i - v\|^2 = \sum_{i=1}^k \mathrm{dist}_E(q_i, p + V)^2 \qquad (1)$$

is minimized, dist_E denoting the Euclidean distance to a closed subset.

This problem can be solved by computing an eigenvalue decomposition UDU^T, $D = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, $\lambda_1 \geq \ldots \geq \lambda_n$ of the symmetric, positive semidefinite matrix $\sum_{i=1}^k (q_i - \overline{q})(q_i - \overline{q})^T$ with $\overline{q} = \frac{1}{k}\sum_{i=1}^k q_i$ the mean of the data points. The best fitting affine subspace is given by $(p + V)_{\mathrm{opt}} = \overline{q} + \mathrm{span}\{u_1, \ldots, u_m\}$ with $u_1, \ldots, u_m$ denoting the first m columns of U. The u_i are called the *principal components* of the q_i.

The orthogonal projection of the data points onto $(p+V)_{\text{opt}}$ in the basis $u_1,\dots,u_m$ of $(p+V)_{\text{opt}}$ is given by $\big(u_1\dots u_m\big)^T(q_i-\overline{q})$. This reduces the n-dimensional data points to m-dimensional data points.

In this paper we concentrate on generalizing the fitting of a subspace to the data (1) to Lie groups. This is justified by the statistical information hold by the $(p+V)_{\text{opt}}$ in the Euclidean case, cf. [3].

3 Principal geodesic analysis

Fletcher *et al.* propose *principal geodesic analysis (PGA)* — a local approach which lifts the data to a tangent space and performs PCA there — as a generalization of PCA to manifolds, [4, 5]. They consider data points $q_1,\dots,q_k$ on a Riemannian manifold M and a Karcher mean $\overline{q}$. Let $\exp_{\overline{q}}$ denote the Riemannian exponential map. They define *principal geodesic submanifolds* recursively as submanifolds $N_1 := \exp_{\overline{q}}(V_1),\dots,N_n := \exp_{\overline{q}}(V_{n-1})$, $V_1=\operatorname{span}\{v_1\},\dots,V_{n-1}=\operatorname{span}\{v_1,\dots,v_{n-1}\}$ minimizing the squared distance to the data; we refer to [5] for details. To calculate the submanifolds the data points are first lifted to $T_{\overline{q}}M$ by computing $p_i=\exp_{\overline{q}}^{-1}(q_i)$. Since $T_{\overline{q}}M$ is a finite dimensional Hilbert space with the scalar product given by the Riemannian metric, one can choose an orthonormal basis of $T_{\overline{q}}M$ and perform PCA on the p_i as points in an Euclidean space. The principal components $u_i\in T_{\overline{q}}M$ yield an approximation $\tilde{V}_m=\operatorname{span}\{u_1,\dots,u_m\}\subset T_{\overline{q}}M$ of the V_m and therefore an approximation of the fitting problem

$$\text{Minimize } \textstyle\sum_{i=1}^{k}\operatorname{dist}_R(q_i,\exp_{\overline{q}}(V))^2 \qquad (2)$$

over the set of m-dimensional subspaces V of $T_{\overline{q}}M$ with dist_R the Riemannian distance. Note that for $M=\mathbb{R}^n$ with the Euclidean metric this yields precisely $(p+V)_{\text{opt}}$ of (1) since $(p+V)_{\text{opt}}=\exp_{\overline{q}}(\tilde{V})=\overline{q}+\tilde{V}$.

For a sufficiently small neighborhood U of 0 in $T_{\overline{q}}M$ the set $\exp_{\overline{q}}(\tilde{V}\cap U)$ is an embedded submanifold and it is 'close' to the optimal $\exp_{\overline{q}}(V)$ of (2). Therefore PGA is suitable if the data are clustered around a unique Karcher mean. However, if the data are not clustered around a point, one has to take into account that the Karcher mean is not unique, that $\exp_{\overline{q}}(\tilde{V})$ is not necessarily an embedded manifold, and that $\exp_{\overline{q}}(\tilde{V})$ is not an exact solution of the fitting problem (2). In such cases PGA is not well-suited to compute a best fitting submanifold and a global approach might be more desirable as a generalization of (1).

4 Fitting cosets

We propose here a global approach to generalize (1) to compact Lie groups. It is based on an alternative interpretation of the Euclidean fitting problem.

Recall that the special Euclidean group $SE(n) = \{(R,p) \mid R \in SO(n), p \in \mathbb{R}^n\}$ acts on $\mathbb{R}^n$ transitively by $\phi\colon (x,(R,p)) \mapsto Rx + p$. Thus $\mathbb{R}^n$ can be thought of as the homogeneous space $\mathbb{R}^n \cong SE(n)/SO(n)$. For the Euclidean distance dist_E we have $\mathrm{dist}_E(x,y) = \mathrm{dist}_E(\phi(x,(R,p)),\phi(y,(R,p))$ for all (R,p), i.e. dist_E is invariant under the action of $SE(n)$ on $\mathbb{R}^n$. In general, a distance dist on a homogeneous space M is called *G-invariant* if for all $x,y \in M$, $s \in G$ $\mathrm{dist}(s\cdot x, s\cdot y) = \mathrm{dist}(x,y)$, $s\cdot x$ denoting the action of G on M. Note further that given a fixed subspace $\tilde{V} \subset \mathbb{R}^n$, $\dim \tilde{V} = m$, any m-dimensional affine subspace can be written as $R\tilde{V} + p$ with $(R,p) \in SE(n)$. Thus minimizing (1) over the set of affine subspaces is equivalent to $\min_{(R,p)\in SE(n)} \sum_{i=1}^k \mathrm{dist}_E(q_i, R\tilde{V} + p)^2$.

This motivates to consider the following fitting problem for invariant distances on homogeneous spaces as a generalization of (1).

Problem 1. Let M a homogeneous space with Lie group $\tilde{G}$ acting transitively on M via $\phi\colon \tilde{G} \times M \to M$, N a submanifold on M and dist an invariant distance. Solve the optimization problem

$$\min_{g\in\tilde{G}} \sum_{i=1}^k \mathrm{dist}(q_i, \phi(g,N))^2. \tag{3}$$

We have seen that (1) is a special case of (3) for $M = \mathbb{R}^n$, $\tilde{G} = SE(n)$, $N = \tilde{V} \cong \mathbb{R}^m$, $\mathrm{dist} = \mathrm{dist}_E$ and $\phi(x,(R,p)) = Rx + p$.

To use (3) for data on the Lie group G, we have to turn G into an homogeneous space, i.e. find another Lie group acting transitively on G. A naïve choice would be G with its action on itself by left- and right-multiplication. However, if e.g. N is a subgroup this would turn G into a fiber bundle, providing not enough degrees of freedom for a sensible fitting of the data by submanifolds diffeomorphic to N. The action ψ of $\tilde{G} = G \times G$ on G with $\psi\colon (x,(p,q)) \mapsto pxq^{-1}$ will be more suitable for our task: it will generate for subgroups N a larger class of submanifolds in G. The distances dist on G, invariant under the action ψ, are called *biinvariant* since for all $q,p,s \in G$ one has $\mathrm{dist}(sq,sp) = \mathrm{dist}(q,p) = \mathrm{dist}(qs,ps)$.

Examples of biinvariant distances include the following:

(a) Let $\langle\cdot,\cdot\rangle$ be a biinvariant Riemannian metric on G. Then the Riemannian distance on G is biinvariant.

(b) Let $\rho\colon G \to \mathbb{C}^{m\times m}$ be a faithful, unitary representation of G, i.e. a homomorpism onto the group of unitary transformations of a finite dimensional Hilbert space with $\ker\rho = \{e\}$. Then $\mathrm{dist}(q,p) = \|\rho(q) - \rho(p)\|_F$, $\|A\|_F = \mathrm{tr}(A^\dagger A)^{1/2}$ the Frobenius norm, $A^\dagger$ the Hermitian conjugate, is a biinvariant distance on G. In particular, for the special orthogonal and the unitary group, the Frobenius norm of the difference of two matrices $\|Q-P\|_F$ yields a biinvariant distance.

We have to choose the class of submanifolds which we use for fitting the data. For PCA in Euclidean space the fitting submanifolds are affine subspaces, i.e. totally geodesic submanifolds of $\mathbb{R}^n$. This suggests the use of totally geodesic submanifolds at least for biinvariant Riemannian distances/metrics, too. However, since we want to exploit the group structure to obtain a reduced optimization problem, we restrict ourselves to closed, i.e. in this case compact, subgroups of G. Indeed subgroups of G are totally geodesic for any biinvariant metric.

Considering G as a homogeneous space with $G \times G$ acting on it by ψ, the fitting problem (3) for N a compact subgroup $H \subset G$ has the form

$$\min_{(p,q) \in G \times G} \textstyle\sum_{i=1}^{k} \operatorname{dist}(q_i, \psi((p,q),H))^2 = \sum_{i=1}^{k} \operatorname{dist}(q_i, pHq^{-1})^2$$

with dist a ψ-invariant, i.e. biinvariant, distance on G. This gives the following fitting problem as a special case of (3) and a global generalization of (1) to Lie groups.

Problem 2. Let $H \subset G$ be a fixed, compact subgroup, dist: $G \times G \to \mathbb{R}$ a biinvariant distance function and $q_1, \ldots, q_k \in G$ data points. Solve the optimization problem

$$\min_{p,q \in G} \textstyle\sum_{i=1}^{k} \operatorname{dist}(q_i, pHq^{-1})^2. \tag{4}$$

Any of the pHq^{-1} can be written as $\tilde{p}qHq^{-1}$, i.e. it is a coset of a subgroup of G conjugate to H. Therefore our approach consists of *optimally fitting to the data a coset of a subgroup conjugate to H.*

4.1 Reduction to a homogeneous space

Note that $G \times G$ is, especially for large subgroups H, a vast overparameterization of the family of submanifolds pHq^{-1}. Fortunately, this problem can be reduced to an optimization problem on the homogeneous space $G/H \times G/H$. The key insight is that the biinvariant distance on G induces a G-invariant distance on G/H.

Proposition 1. *Let* dist_G *be a biinvariant distance on G and $H \subset G$ a compact subgroup. Then* dist_G *induces a G-invariant distance* $\operatorname{dist}_{G/H}$ *on G/H, such that* $\operatorname{dist}_{G/H}(qH, pH) = \operatorname{dist}_G(q, pH)$.

Proof. Since dist_G is right-invariant we have for all $k \in H$

$$\operatorname{dist}_G(q, pH) = \min_{h \in H} \operatorname{dist}_G(q, ph) = \min_{h \in H} \operatorname{dist}_G(qk, ph) = \operatorname{dist}_G(qk, pH).$$

Thus $\operatorname{dist}_G(q, pH)$ induces a distance $\operatorname{dist}_{G/H}$ on G/H. The G-invariance of $\operatorname{dist}_{G/H}$ follows directly from the left-invariance of dist_G.

Induced distances on G/H include the following examples:

(a) Let $\langle \cdot, \cdot \rangle_G$ be a biinvariant Riemannian metric on G. We can define on G the distribution $N(p) := (T_p pH)^{\perp}$, $W^{\perp}$ the orthogonal complement with respect to the Riemannian metric. Let $\pi\colon G \to G/H$ be the canonical projection $\pi(p) := pH$. Then, the formula

$$\langle v, w\rangle_{G/H} := \langle v^N, w^N\rangle \text{ for } v, w \in T_{pH}G/H$$

defines an G-invariant Riemannian metric on G/H with v^N, w^N uniquely defined by $v^N, w^N \in N(p)$, $T_p\pi v^N = v$, $T_p\pi w^N = w$. This Riemannian metric is called the *normal metric* [9]. The distance on G/H induced by the Riemannian metric on G is the Riemannian distance of the normal metric.

(b) Let ρ be again a faithful, finite dimensional, unitary representation of G and $H = \operatorname{stab}(v) = \{p \in G \mid \rho(p)v = v\}$ for a $v \in \mathbb{C}^m$. We can identify the orbit $O(v) = \{\rho(p)v \mid p \in G\}$ with G/H via $pH \mapsto \rho(p)v$. Then the distance $\operatorname{dist}(p,q) = \|\rho(p) - \rho(q)\|_F$ induces the the Euclidean distance $\operatorname{dist}(p,q) = \|\rho(p)(v) - \rho(q)(v)\|$ on $O(v) = G/H$.

Problem (4) thus leads to the following reduced optimization problem on $G/H \times G/H$.

Proposition 2. *Assume that* dist *is a biinvariant distance on* G. *Then* $(p,q) \in G \times G$ *is a solution of Problem* (2) *if and only if* (qH, pH) *is a minimum of* $g\colon G/H \times G/H \to \mathbb{R}$,

$$g(x,y) = \textstyle\sum_{i=1}^k \operatorname{dist}_{G/H}(q_i \cdot x, y)^2 \tag{5}$$

with $q \cdot x$ *denoting the canonical action of* G *on* G/H.

Proof. By the invariance of dist and Proposition 1 we have

$$\textstyle\sum_{i=1}^k \operatorname{dist}(q_i, pHq^{-1})^2 = \sum_{i=1}^k \operatorname{dist}(q_i q, pH)^2 = \sum_{i=1}^k \operatorname{dist}_{G/H}(q_i qH, pH)^2$$

Thus (p,q) solves (4) if and only if (qH, pH) is a minimum of g.

4.2 An algorithm for Riemannian fitting

If the distance on G is the Riemannian distance of a biinvariant Riemannian metric, we can derive a general gradient algorithm to find a minimum of (5). As discussed in the examples above the induced distance on G/H from the biinvariant metric on G is the Riemannian distance with respect to the normal metric on G/H. Thus we assume that G/H carries this normal metric in the remainder of this section. Note that

$$g(x,y) = \textstyle\sum_{i=1}^k \operatorname{dist}_{G/H}(q_i \cdot x, y)^2 = \sum_{i=1}^k \operatorname{dist}_{G/H}(x, q_i^{-1} \cdot y)^2 \tag{6}$$

is in each variable the Karcher mean cost function for points $q_i \cdot x$ resp. $q_i^{-1} \cdot y$ on G/H. It is well-known that the gradient of the Karcher mean cost $c(x) = \sum_{i=1}^k \operatorname{dist}(x, x_i)^2$ is given by $\operatorname{grad} c(x) = \frac{1}{k}\sum_{i=1}^k \exp_x^{-1}(x_i)$, see [4, 7, 11]. Thus the gradient of g with respect to the product metric on $G/H \times G/H$ is

$$\operatorname{grad} g(x,y) = \left(\tfrac{1}{k} \sum_{i=1}^{k} \exp_x^{-1}(q_i^{-1} \cdot y), \tfrac{1}{k} \sum_{i=1}^{k} \exp_y^{-1}(q_i \cdot x) \right).$$

The form (6) of the cost suggests the following gradient descent algorithm to minimize g as an adaption of the Karcher mean algorithm [4, 7, 11].

Riemannian fitting algorithm

```
1. Initialize $x_0, y_0 \in G/H$ and choose a $\varepsilon > 0$
2. $x_{j+1} = \exp_{x_j}\left(\frac{1}{k}\sum_{i=1}^{k} \exp_x^{-1}(q_i^{-1} \cdot y_j)\right)$
3. $y_{j+1} = \exp_{y_j}\left(\frac{1}{k}\sum_{i=1}^{k} \exp_y^{-1}(q_i \cdot x_j)\right)$
4. go to step 2 until $\operatorname{dist}(x_j, x_{j+1}) < \varepsilon$ and $\operatorname{dist}(y_j, y_{j+1}) < \varepsilon$
5. Let $x_j = qH$, $y_j = rH$.
6. Output: $(r, q)$ as an approximation of the minimum of $f$
```

This algorithm requires that the $q_i^{-1} \cdot y_j$ resp. $q_i \cdot x_j$ are in the domain of $\exp_{x_j}^{-1}$ resp. $\exp_{y_j}^{-1}$ and is not necessarily globally defined. However, since these are exponential maps on G/H the algorithm will work for data clustered near a coset pHq^{-1} even if there is a continuum of Karcher means on G. An alternative would be to globalize the algorithm using non-smooth optimization methods, but this is beyond the scope of the present paper.

5 Example: Fitting on SO(n)

We illustrate the proposed approach on the special orthogonal group $SO(n)$.

The distances discussed in the examples (a), (b) above yield two choices for distances on $SO(n)$: (a) the Riemannian distance of a biinvariant metric and (b) the Frobenius norm distance on the matrix representation of $SO(n)$.

(a) In the Riemannian case the induced distance on $SO(n)/H$ is the normal Riemannian metric and the algorithm from Section 4.2 can be applied to compute the optimal coset on $SO(n)$. As a special case consider the problem of fitting data with a coset of a conjugate of a subgroup $\mathcal{H} \cong SO(n-1)$. The quotient space $SO(n)/\mathcal{H}$ can be identified with S^{n-1} via the diffeomorphism $Q\mathcal{H} \mapsto Qv$ for $v \in S^{n-1}$ such that $\operatorname{stab}(v) = \mathcal{H}$. Any biivariant Riemannian metric on $SO(n)$ has the form $\langle X\Omega, X\Theta \rangle = C \operatorname{tr}(\Omega^T \Theta)$ with $C > 0$; w.l.o.g. assume $C = \frac{1}{2}$. Then the normal metric on S^{n-1} coincides with the Riemannian metric on S^{n-1}. Thus the exponential map on the sphere is given by $\exp_x(v) := \cos(\|v\|)x + \frac{\sin(\|v\|)}{\|v\|}v$ and its inverse by $\exp_x^{-1}(y) := \frac{s}{\sin(s)}(y - \cos(s)x)$ with $s = \arccos(y^T x)$. Using this information, it is straightforward to implement the algorithm from Section 4.2.

(b) As an example for the Frobenius norm distance on a matrix representation of $SO(n)$, consider the representation $\rho(U)\colon \mathbb{C}^{n \times p} \to \mathbb{C}^{n \times p}$ with $\rho(U)(A) = UA$. We treat $\mathbb{C}^{n \times p}$ as the vector space $\mathbb{C}^{np}$. Then $\rho(U) = (I_p \otimes U)$ and the Frobenius norm distance is given by

$$\mathrm{dist}_F(U,V) = \|\rho(U) - \rho(V)\|_F = \|(I_p \otimes U) - (I_p \otimes V)\|_F = p\|U - V\|_F.$$

Let $A = \begin{pmatrix} I_p & 0 \end{pmatrix}^T \in \mathbb{C}^{n\times p}$. Assume that we want to fit a coset of a subgroup conjugate to $\mathcal{H} = \mathrm{stab}(A) \cong SO(n-p)$ to the data. The orbit $O(A)$ is the compact Stiefel manifold $\mathrm{St}(n,p)$ and we can identify $SO(n)/\mathcal{H}$ with $\mathrm{St}(n,p)$ by $U\mathcal{H} \mapsto \rho(U)A$. By Section 4.1, Example (b), the induced distance on $SO(n)/\mathcal{H}$ is the Euclidean distance on $\mathrm{St}(n,p)$, i.e.

$$\mathrm{dist}_{SO(n)/\mathcal{H}}(UA, VA) = \|UA - VA\|_F.$$

Thus to find the best fitting coset $P\mathcal{H}Q^{-1}$, $P, Q \in SO(n)$, to data points $Q_1, \ldots, Q_k$ in $SO(n)$ one must minimize the cost

$$g(X,Y) = \textstyle\sum_{i=1}^k \|X - Q_iY\|_F^2$$

on $\mathrm{St}(n,p) \times \mathrm{St}(n,p)$. Here, we use the gradient descent with retractions from [6] on the product of the Stiefel manifold. To compute a gradient we use the Riemannian metric on the Stiefel manifold induced by the Euclidean one on $\mathbb{R}^{n\times p}$ and equip $\mathrm{St}(n,p) \times \mathrm{St}(n,p)$ with the product metric. The gradient with respect to this induced Riemannian metric is given by the orthogonal projection of the Euclidean gradient of an extension of g to $\mathbb{R}^{n\times p} \times \mathbb{R}^{n\times p}$ onto the tangent space $T_{(X,Y)}\left(\mathrm{St}(n,p) \times \mathrm{St}(n,p)\right)$. Since the Euclidean gradient of g is given by $\mathrm{grad}_E\, g(X,Y) = \Big(\sum_{i=1}^k (X - Q_i^TY), \sum_{i=1}^k (Y - Q_iX)\Big)$ and the projection $\pi_X \colon \mathbb{R}^{n\times p} \to T_X\,\mathrm{St}(n,p)$ by $\pi_X(V) = V - \frac{1}{2}X(X^TV + V^TX)$, cf. [6], we obtain $\mathrm{grad}\, g(X,Y) = \Big(\left(\frac{1}{2}XX^T - I_n\right)\sum_{i=1}^k Q_i^TY\ + \frac{1}{2}XY^T\sum_{i=1}^k Q_iX,$ $\left(\frac{1}{2}YY^T - I_n\right)\sum_{i=1}^k Q_iX + \frac{1}{2}YX^T\sum_{i=1}^k Q_i^TY\Big)$. A descent algorithm on a manifold needs suitable local charts R_X which map lines in the tangent space onto curves in the manifold. Here, we choose for the Stiefel manifold the polar decomposition retractions from [6], i.e. $R_X \colon T_X\,\mathrm{St}(n,p) \to \mathrm{St}(n,p)$, $R_X(V) = (X+V)(I_p + V^TV)^{-1/2}$. Since we have to optimize over the product of two Stiefel manifolds, we use this retraction in each component. The step length of the gradient descent is determined by an Armijo line search. This yields the following algorithm:

```
1. Initialize $X_0, Y_0 \in \mathrm{St}(n,p)$ and choose a $\varepsilon > 0$, $\sigma \in (0,1)$
2. Calculate $S_{X,j} = \sum_{i=1}^k Q_iX_j$ and $S_{Y,j} = \sum_{i=1}^k Q_i^TY_j$.
3. Set $\eta_j := (\frac{1}{2}X_jX_j^T - I_n)S_{Y,j} + \frac{1}{2}X_jY_j^TS_{X,j}$, $\zeta_j := (\frac{1}{2}Y_jY_j^T - I_n)S_{X,j} + \frac{1}{2}Y_jX_j^TS_{Y,j}$
4. Choose the smallest $\alpha \in \mathbb{N}$ such that

   $g(X_j,Y_j) - g\left(R_{X_j}(-2^{-\alpha}\eta_j), R_{Y_j}(-2^{-\alpha}\zeta_j)\right) \geq \sigma 2^{-\alpha}\left(\|\eta_j\|_F^2 + \|\zeta_j\|_F^2\right)$

5. Set $X_{j+1} := (X_j - 2^{-\alpha}\eta_j)\left(I_p + 2^{-2\alpha}\eta_j^T\eta_j\right)^{-1/2}$,
   $Y_{j+1} := (Y_j - 2^{-\alpha}\zeta_j)\left(I_p + 2^{-2\alpha}\zeta_j^T\zeta_j\right)^{-1/2}$
```

```
6. If $\|\eta_j\| > \varepsilon$ or $\|\zeta_j\| > \varepsilon$ then $j := j{+}1$ and go to step 2, otherwise
   go to step 7.
7. Find $Q, R \in SO(n)$ such that $X_j = QA$, $Y_j = RA$ and output
   $(R, Q)$ as an approximation of the minimum of $f$.
```

Figure 1 shows the behavior of the algorithm for the Riemannian distance and the $\mathcal{H} \cong SO(n-1)$ with 30 data points in $SO(10)$. The data points for the left graph are constructed by choosing random points on a coset $\cong SO(9)$, while for the right graph randomly chosen data points on the coset were perturbed by multiplication with i.d.d. random rotations $R = \exp(N)$ with N the skew-symmetric parts of i.d.d. random matrices $M \sim N(0, \sqrt{0.1})$. For the unperturbed case the algorithm shows linear convergence as it is to be expected for a gradient method. In the perturbed case the algorithm converges quickly to a cost function value larger than 0 since an exact fitting is not possible anymore.

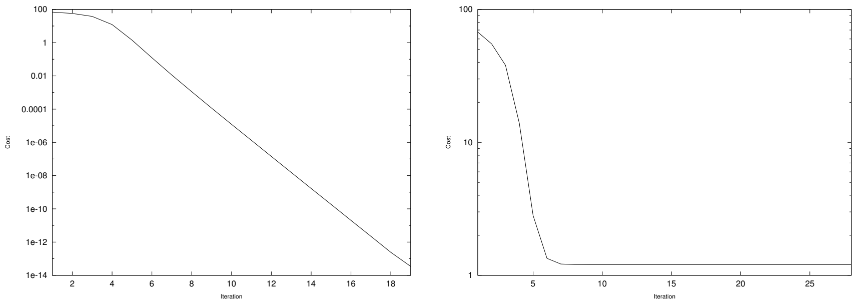


Fig. 1. Evolution of the cost for the first example in Section 5 with $n = 10$ and $k = 30$. The left figure shows the unperturbed case while the right the case of data points perturbed by random rotations.

Figure 2 illustrates the behavior of the algorithm for the Frobenius norm distance and the $\mathcal{H} = \mathrm{stab}((I_p 0)^T) \cong SO(n-p)$ with $n = 10$, $p = 8$ and $k = 30$. The left graph shows the case of data points randomly chosen on a fixed coset, while the right graph shows the case of random points on the coset perturbed by a random rotations $R = \exp(N)$ with N the skew-symmetric part of random $M \sim N(0, \sqrt{0.1})$.

Acknowledgments

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors.

The research was initiated during the postdoctoral stay of the first author at the University of Liège.

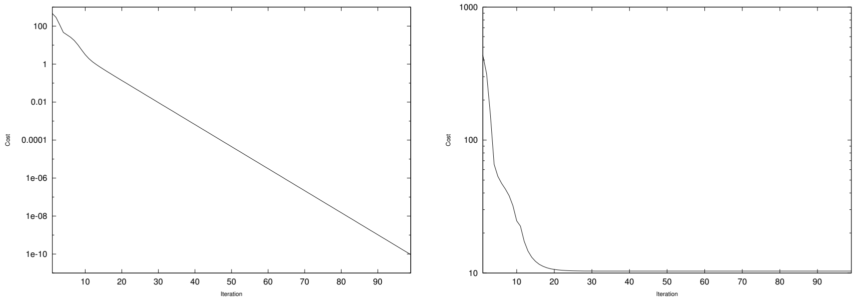


Fig. 2. Evolution of the cost for the second example in Section 5 with $n = 10$, $p = 8$ and $k = 30$. The left figure shows the case of unperturbed data on a coset while in the right one the data points have been perturbed by random rotations.

References

1. S. Helgason. (1994). Geometric analysis on symmetric spaces. American Math. Soc., Providence, RI.
2. K. V. Mardia, P. E. Jupp. (2000). Directional Statistics. Wiley, Chichester.
3. I. T. Jolliffe. (1986). Principal Component Analysis. Springer-Verlag, New York.
4. P.T. Fletcher, C. Lu, S. Joshi. (2003). Statistics of Shape via Principal Geodesic Analysis on Lie Groups. In: Proc. 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR03) p. I-95 – I-101
5. P.T. Fletcher, C. Lu, S.M. Pizer, S. Joshi. (2004). Principal Geodesic Analysis for the Study of Nonlinear Statistics of Shape. IEEE Trans. Medical Imagining 23(8):995–1005
6. P.-A. Absil, R. Mahony, R. Sepulchre. (2008). Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton
7. L. Machado (2006) Least Squares Problems on Riemannian Manifolds. Ph.D. Thesis, University of Coimbra, Coimbra
8. L. Machado, F. Silva Leite (2006). Fitting Smooth Paths on Riemannian Manifolds. Int. J. Appl. Math. Stat. 4(J06):25–53
9. J. Cheeger, D. G. Ebin (1975). Comparison theorems in Riemannian geometry. North-Holland, Amsterdam
10. H. Karcher (1977). Riemannian center of mass and mollifier smoothing. Comm. Pure Appl. Math. 30:509–541
11. J. H. Manton. (2004). A Globally Convergent Numerical Algorithm for Computing the Centre of Mass on Compact Lie Groups. Eighth Internat. Conf. on Control, Automation, Robotics and Vision, December, Kunming, China. p. 2211–2216
12. M. Moakher. (2002). Means and averaging in the group of rotations. SIAM Journal on Matrix Analysis and Applications 24(1):1–16
13. J. H. Manton. (2006). A centroid (Karcher mean) approach to the joint approximate diagonalisation problem: The real symmetric case. Digital Signal Processing 16:468–478

Riemannian BFGS Algorithm with Applications

Chunhong Qi[1], Kyle A. Gallivan[1], and P.-A. Absil[2]

[1] Department of Mathematics, Florida State University, Tallahassee, FL, 32306, USA, {cqi, gallivan}@math.fsu.edu

[2] Département d'ingénierie mathématique, Université catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium, absil@inma.ucl.ac.be

Summary. We present an algorithm model, called Riemannian BFGS (RBFGS), that subsumes the classical BFGS method in $\mathbb{R}^n$ as well as previously proposed Riemannian extensions of that method. Of particular interest is the choice of transport used to move information between tangent spaces and the different ways of implementing the RBFGS algorithm.

1 Introduction

Optimization on manifolds, or *Riemannian optimization*, concerns finding an optimum (global or local) of a real-valued function defined over a smooth manifold. A brief introduction to the area can be found in [1] in this volume, and we refer to [3] and the many references therein for more details. Optimization on manifolds finds applications in two broad classes of situations: classical equality-constrained optimization problems where the constraints specify a submanifold of $\mathbb{R}^n$; and problems where the objective function has continuous invariance properties that we want to eliminate for various reasons, e.g., efficiency, consistency, applicability of certain convergence results, and avoiding failure of certain algorithms due to degeneracy. As a result, the generalization to manifolds of algorithms for unconstrained optimization in $\mathbb{R}^n$ can yield useful and efficient numerical methods; see, e.g., recent work on Riemannian trust-region methods [2] and other methods mentioned in [3]. Since BFGS is one of the classical methods for unconstrained optimization (see [7, 10]), it is natural that its generalization is a topic of interest.

Some work has been done on BFGS for manifolds. Gabay [9, §4.5] discussed a version using parallel transport. Brace and Manton [6] have a version on the Grassmann manifold for the problem of weighted low-rank approximations. Savas and Lim [11] apply a version on a product of Grassmann manifolds to the problem of best multilinear low-rank approximation of tensors.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_16, © Springer-Verlag Berlin Heidelberg 2010

Gabay's Riemannian BFGS [9, §4.5] differs from the classical BFGS method in $\mathbb{R}^n$ (see, e.g., [10, Alg. 6.1]) in five key aspects: (i) The search space, to which the iterates x_k belong, is a Riemannian submanifold M of $\mathbb{R}^n$ specified by equality constraints; (ii) The search direction at x_k is a tangent vector to M at x_k; (iii) The update along the search direction is performed along the geodesic determined by the search direction; (iv) The usual quantities s_k and y_k that appear in the secant equation are tangent vectors to M at x_{k+1}, obtained using the Riemannian parallel transport (i.e., the parallel transport induced by the Levi-Civita connection) along the geodesic. (v) The Hessian approximation $\mathcal{B}_k$ is a linear transformation of the tangent space $T_{x_k}M$ that gets updated using a generalized version of the BFGS update formula. This generalized formula specifies recursively how $\mathcal{B}_k$ applies to elements of $T_{x_k}M$.

In this paper, we present an algorithm model (or meta-algorithm), dubbed RBFGS, that subsumes Gabay's Riemannian BFGS method. Whereas Gabay's method is fully specified by the Riemannian manifold, the cost function, and the initial iterate, our RBFGS algorithm offers additional freedom in the choice of a retraction and a vector transport (see Section 2 for a brief review of these two concepts). This additional freedom affects points (iii) and (iv) above. For (iii), the curves along which the update is performed are specified by the retraction. For (iv), the Levi-Civita parallel transport is replaced by the more general concept of vector transport. If the retraction is selected as the Riemannian exponential and the vector transport is chosen to be the Levi-Civita parallel transport, then the RBFGS algorithm reduces to Gabay's algorithm (barring variations of minor importance, e.g., in the line-search procedure used).

The impact of the greater freedom offered by the RBFGS algorithm varies according to the manifold of interest. On the sphere, for example, the computational cost of the Riemannian exponential and the Levi-Civita parallel transport is reasonable, and there is not much to be gained by choosing computationally cheaper alternatives. In contrast, as we will show in numerical experiments, when the manifold is the Stiefel manifold, $\mathrm{St}(p, n)$, of orthonormal p-frames in $\mathbb{R}^n$, the improvement in computational time can be much more significant.

This paper also improves on Gabay's work by discussing the practical implementation of the algorithm. When the manifold M is a submanifold of $\mathbb{R}^n$, we offer the alternatives of either representing the tangent vectors and the approximate Hessian using a basis in the tangent spaces, or relying on the canonical inclusion of M in $\mathbb{R}^n$. The latter leads to representations of tangent vectors as n-tuples of real numbers and of the approximate Hessian as an $n \times n$ matrix. This approach may offer a strong advantage when the co-dimension of M is sufficiently small.

Another feature of RBFGS is that it does not assume that M is a submanifold of a Euclidean space. As such, it can be applied to quotient manifolds as well. However, in this paper, we concentrate the practical implementation discussion on the submanifold case.

This paper is a first glimpse at ongoing work that aims at a systematic analysis and evaluation of the Riemannian versions of the BFGS algorithm. It is organized as follows. The general RBFGS algorithm is given in Section 3. The two implementation approaches and the particular implementation on certain manifolds are given in Section 4. In Section 5, we summarize the results of our numerical experiments for two application problems: the Rayleigh quotient problem on the sphere S^{n-1} and a matrix Procrustes problem on the compact Stiefel manifold.

2 Mathematical preliminaries

The notion of retraction on a manifold, due to Adler *et al.* [4], encompasses all first-order approximations to the Riemannian exponential. Here we recall the definition as given in [3].

Definition 1. *A* retraction *on a manifold M is a mapping R from the tangent bundle TM onto M with the following properties. Let R_x denote the restriction of R to T_xM.*

1. *R is continuously differentiable.*
2. *$R_x(0_x) = x$, where 0_x denotes the zero element of T_xM.*
3. *With the canonical identification $T_{0_x}T_xM \simeq T_xM$, R_x satisfies $\mathrm{D}R_x(0_x) = id_{T_xM}$, where D denotes the derivative and id_{T_xM} denotes the identity mapping on T_xM.*

The retraction is used as a way to take a step in the direction of a tangent vector. Choosing a good retraction amounts to finding an approximation of the exponential mapping that can be computed with low computational cost while not adversely affecting the behavior of the optimization algorithm.

Next we recall the concept of vector transport, which specifies how to move a tangent vector from one tangent space to another. This is also used to move a linear operator from one tangent space to another, e.g., the approximate Hessian in (4). The notion of vector transport was introduced in [3] for reasons similar to those that motivated the introduction of retractions, namely, to provide a framework for using computationally less expensive approximations of the Levi-Civita parallel translation. The definition below, illustrated in Figure 1, invokes the Whitney sum $TM \oplus TM$, which stands for the set of all ordered pairs of tangent vectors with same foot.

Definition 2. *A* vector transport *on a manifold M is a smooth mapping: $TM \oplus TM \to TM$, $(\eta_x, \xi_x) \mapsto \mathcal{T}_{\eta_x}(\xi_x) \in TM$ satisfying the following properties for all $x \in M$.*

1. *(Associated retraction) There exists a retraction R, called the* retraction associated with *$\mathcal{T}$, such that, for all η_x, ξ_x, it holds that $\mathcal{T}_{\eta_x}\xi_x \in T_{R_x(\eta_x)}M$.*
2. *(Consistency) $\mathcal{T}_{0_x}\xi_x = \xi_x$ for all $\xi_x \in T_xM$;*

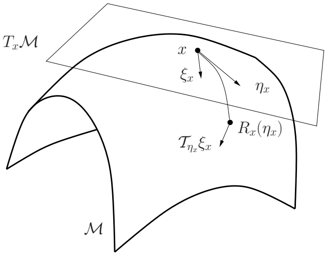


Fig. 1. Vector transport.

3. (Linearity) The mapping $\mathcal{T}_{\eta_x} : T_xM \to T_{R(\eta_x)}M$, $\xi_x \mapsto \mathcal{T}_{\eta_x}(\xi_x)$ is linear.

Note that, in general, vector transports are not isometries; in fact, the definition of a vector transport does not even assume an underlying Riemannian metric. When M is a Riemannian manifold and the vector transport is selected to be the Levi-Civita parallel translation, then it is an isometry. When it exists, the inverse of the linear map $\mathcal{T}_{\eta_x}$ is denoted by $(\mathcal{T}_{\eta_x})^{-1}$. Observe that $(\mathcal{T}_{\eta_x})^{-1}(\xi_{R_x(\eta_x)})$ belongs to T_xM. If M is an embedded submanifold of a Euclidean space and M is endowed with a retraction R, then a particular choice of vector transport is given by

$$\mathcal{T}_{\eta_x}\xi_x := \mathrm{P}_{R_x(\eta_x)}\xi_x, \tag{1}$$

where P_x denotes the orthogonal projector onto T_xM. Depending on the manifold, this vector transport may be much less expensive to compute than the Levi-Civita parallel transport. Other choices may also be used to achieve computational savings. It may happen that the chosen vector transport and its inverse are not defined everywhere, but then the set of problematic points is usually of measure zero, and no difficulty is observed in numerical experiments.

3 The RBFGS Algorithm

The structure of the RBFGS algorithm is given in Algorithm 0.1. Recall that, given a smooth scalar field f on a Riemannian manifold M with Riemannian metric g, the gradient of f at x, denoted by $\operatorname{grad} f(x)$, is defined as the unique element of T_xM that satisfies:

$$g_x(\operatorname{grad} f(x), \xi) = \mathrm{D}f(x)[\xi], \forall \xi \in T_xM. \tag{2}$$

The line-search procedure in Step 4 of RBFGS uses Armijo's condition.

The RBFGS algorithm can also be reformulated to work with the inverse Hessian approximation $\mathcal{H}_k = {\mathcal{B}_k}^{-1}$ rather than with the Hessian approximation B_k. In this case, Step 6 of RBFGS is replaced by

Algorithm 0.1 RBFGS

```
Given: Riemannian manifold $M$ with Riemannian metric $g$; vector transport $\mathcal{T}$ on $M$ with associated retraction $R$; smooth real-valued function $f$ on $M$; initial iterate $\mathbf{x}_0 \in M$; initial Hessian approximation $\mathcal{B}_0$.
for k = 0, 1, 2, ... do
    Obtain $\eta_k \in T_{\mathbf{x}_k}M$ by solving $\mathcal{B}_k \eta_k = -\operatorname{grad} f(\mathbf{x}_k)$.
    Set step size $\alpha = 1$, $c = g(\operatorname{grad} f(\mathbf{x}_k), \eta_k)$. While $f(R_{\mathbf{x}_k}(2\alpha\eta_k)) - f(\mathbf{x}_k) < \alpha c$, set $\alpha := 2\alpha$. While $f(R_{\mathbf{x}_k}(\alpha\eta_k)) - f(\mathbf{x}_k) \geq 0.5\alpha c$, set $\alpha := 0.5\alpha$. Set $\mathbf{x}_{k+1} = R_{\mathbf{x}_k}(\alpha\eta_k)$.
    Define $s_k = \mathcal{T}_{\alpha\eta_k}(\alpha\eta_k)$ and $y_k = \operatorname{grad} f(\mathbf{x}_{k+1}) - \mathcal{T}_{\alpha\eta_k}(\operatorname{grad} f(\mathbf{x}_k))$.
    Define the linear operator $\mathcal{B}_{k+1} : T_{\mathbf{x}_{k+1}}M \to T_{\mathbf{x}_{k+1}}M$ by
        $$\mathcal{B}_{k+1}p = \tilde{\mathcal{B}}_k p - \frac{g(s_k, \tilde{\mathcal{B}}_k p)}{g(s_k, \tilde{\mathcal{B}}_k s_k)}\tilde{\mathcal{B}}_k s_k + \frac{g(y_k, p)}{g(y_k, s_k)} y_k \quad \text{for all } p \in T_{\mathbf{x}_{k+1}}M, \qquad (3)$$
    with
        $$\tilde{\mathcal{B}}_k = \mathcal{T}_{\alpha\eta_k} \circ \mathcal{B}_k \circ (\mathcal{T}_{\alpha\eta_k})^{-1}. \qquad (4)$$
end for
```

$$\mathcal{H}_{k+1}p = \tilde{\mathcal{H}}_k p - \frac{g(y_k, \tilde{\mathcal{H}}_k p)}{g(y_k, s_k)} s_k - \frac{g(s_k, p_k)}{g(y_k, s_k)} \tilde{\mathcal{H}}_k y_k + \frac{g(s_k, p) g(y_k, \tilde{\mathcal{H}}_k y_k)}{g(y_k, s_k)^2} s_k + \frac{g(s_k, s_k)}{g(y_k, s_k)} p \qquad (5)$$

with

$$\tilde{\mathcal{H}}_k = \mathcal{T}_{\eta_k} \circ \mathcal{H}_k \circ (\mathcal{T}_{\eta_k})^{-1}. \qquad (6)$$

This yields a mathematically equivalent algorithm. It is useful because it makes it possible to cheaply compute an approximation of the inverse of the Hessian. This may make RBFGS advantageous even in the case where we have a cheap exact formula for the Hessian but not for its inverse.

4 Practical Implementation of RBFGS

4.1 Two Approaches

A practical implementation of RBFGS requires the following ingredients: (i) an efficient numerical representation for points x on M, tangent spaces T_xM and the inner products $g_x(\xi_1, \xi_2)$ on T_xM; (ii) an implementation of the chosen retraction $R_x : T_xM \to M$; (iii) efficient formulas for $f(x)$ and $\operatorname{grad} f(x)$; (iv) an implementation of the chosen vector transport $\mathcal{T}_{\eta_x}$ and its inverse $(\mathcal{T}_{\eta_x})^{-1}$; (v) a method for solving

$$\mathcal{B}_k \eta_k = -\operatorname{grad} f(\mathbf{x}_k), \qquad (7)$$

where $\mathcal{B}_k$ is defined recursively through (3), or alternatively, a method for computing $\eta_k = -\mathcal{H}_k \operatorname{grad} f(\mathbf{x}_k)$ where $\mathcal{H}_k$ is defined recursively by (5). Point (v) is the main difficulty. In this paper, we restrict to the case where M is a submanifold of $\mathbb{R}^n$, and we construct explicitly a matrix representation of $\mathcal{B}_k$. We discuss two implementation approaches.

Approach 1 realizes $\mathcal{B}_k$ as an $n \times n$ matrix $B_k^{(n)}$. Since M is a submanifold of $\mathbb{R}^n$, tangent spaces T_xM are naturally identified with subspaces of $\mathbb{R}^n$ (see [3, §3.5.7] for details), and it is very common to use the same notation for a tangent vector and its corresponding element of $\mathbb{R}^n$. However, to explain Approach 1, it is useful to distinguish the two objects. To this end, let ι_x denote the natural inclusion of T_xM in $\mathbb{R}^n$, $\iota_x : T_xM \to \mathbb{R}^n, \ \ \xi_x \mapsto \iota_x(\xi_x)$.

To represent $\mathcal{B}_k$, we pick $B_k^{(n)} \in \mathbb{R}^{n\times n}$ such that, for all $\xi_{x_k} \in T_{x_k}M$,

$$B_k^{(n)} \iota_{x_k}(\xi_{x_k}) = \iota_{x_k}(\mathcal{B}_k \xi_{x_k}). \tag{8}$$

Note that condition (8) does not uniquely specify $B_k^{(n)}$; its action on the normal space is irrelevant. Solving the linear system (7) then amounts to finding $\iota_{x_k}(\eta_k)$ in $\iota_{x_k}(T_{x_k}M)$ that satisfies

$$B_k^{(n)} \iota_{x_k}(\eta_k) = -\iota_{x_k}(\operatorname{grad} f(x_k)). \tag{9}$$

It remains to give an expression for the update formula (3). To this end, let $T_{\alpha\eta_k}^{(n)}$ be the $n \times n$ matrix that satisfies $T_{\alpha\eta_k}^{(n)} \iota_{x_k}(\xi_{x_k}) = \iota_{x_{k+1}}(\mathcal{T}_{\alpha\eta_k}\xi_{x_k})$ for all $\xi_{x_k} \in T_{x_k}M$ and $T_{\alpha\eta_k}^{(n)}\zeta_k = 0$ for all $\zeta_k \perp \iota_{x_k}(T_{x_k}M)$. Since M is an embedded submanifold of $\mathbb{R}^n$, the Riemannian metric is given by $g(\xi_x, \eta_x) = \iota_x(\xi_x)^T \iota_x(\eta_x)$ and the update equation (3) is then

$$B_{k+1}^{(n)} = \tilde{B}_k^{(n)} - \frac{\tilde{B}_k^{(n)} \iota_{x_{k+1}}(s_k) \iota_{x_{k+1}}(s_k)^T \tilde{B}_k^{(n)}}{\iota_{x_{k+1}}(s_k)^T \tilde{B}_k^{(n)} \iota_{x_{k+1}}(s_k)} + \frac{\iota_{x_{k+1}}(y_k) \iota_{x_{k+1}}(y_k)^T}{\iota_{x_{k+1}}(y_k)^T \iota_{x_{k+1}}(s_k)},$$

where $\tilde{B}_k^{(n)} = T_{\alpha\eta_k}^{(n)} B_k^{(n)} \left((T_{\alpha\eta_k})^{(n)}\right)^\dagger$ and † denotes the pseudoinverse.

Approach 2 realizes $\mathcal{B}_k$ by a $d \times d$ matrix $B_k^{(d)}$ using bases, where d denotes the dimension of M. Given a basis $(E_{k,1}, \ldots, E_{k,d})$ of $T_{x_k}M$, if $\hat{G}_k \in \mathbb{R}^d$ is the vector of coefficients of $\operatorname{grad} f(x_k)$ in the basis and $B_k^{(d)}$ is the $d \times d$ matrix representation of $\mathcal{B}_k$ in the basis, then we must solve $B_k^{(d)} \hat{\eta}_k = -\hat{G}_k$ for $\hat{\eta}_k \in \mathbb{R}^d$, and the solution η_k of (7) is given by $\eta_k = \sum_{i=1}^d E_{k,i}(\hat{\eta}_k)_i$.

4.2 Implementation on the Unit Sphere

We view the unit sphere $S^{n-1} = \{x \in \mathbb{R}^n : x^T x = 1\}$ as a Riemannian submanifold of the Euclidean space $\mathbb{R}^n$. In the rest of the paper, we abuse the notation by ignoring the inclusions to simplify the formulas.

The tangent space at x, orthogonal projection onto the tangent space at x, and the retraction chosen are given by

$$T_x S^{n-1} = \{\xi \in \mathbb{R}^n \;:\; x^T\xi = 0\}$$
$$\mathrm{P}_x \xi_x = \xi - xx^T\xi_x$$
$$R_x(\eta_x) = (x + \eta_x)/\|(x + \eta_x)\|,$$

where $\|\cdot\|$ denotes the Euclidean norm.

Vector transport (1) on S^{n-1} is given by

$$\mathcal{T}_{\eta_x}\xi_x = \left(I - \frac{(x+\eta_x)(x+\eta_x)^T}{\|x+\eta_x\|^2}\right)\xi_x \tag{10}$$

which takes a vector ξ_x that belongs to the orthogonal complement of x (because it is in the tangent space to the sphere at x) and projects it along $(x+\eta_x)$ into the orthogonal complement of $(x+\eta_x)$. To invert (10), we start from a vector in the orthogonal complement of $(x+\eta_x)$ and project it along $(x+\eta_x)$ into the orthogonal complement of x. The result is an oblique projection

$$(\mathcal{T}_{\eta_x})^{-1}(\xi_{R_x(\eta_x)}) = \left(I - \frac{(x+\eta_x)x^T}{x^T(x+\eta_x)}\right)\xi_{R_x(\eta_x)} \tag{11}$$

For the unit sphere, the Levi-Civita parallel transport of $\xi \in T_x S^{n-1}$ along the geodesic, γ, from x in direction $\eta \in T_x S^{n-1}$ is [5]

$$P_\gamma^{t\leftarrow 0}\xi = \left(I_n + (\cos(\|\eta\|t) - 1)\frac{\eta\eta^T}{\|\eta\|^2} - \sin(\|\eta\|t)\frac{x\eta^T}{\|\eta\|}\right)\xi.$$

This parallel transport and its inverse have computational costs comparable to the chosen vector transport and its inverse.

4.3 Implementation on the Compact Stiefel Manifold $\mathrm{St}(p,n)$

We view the compact Stiefel manifold $\mathrm{St}(p,n) = \{X \in \mathbb{R}^{n\times p} : X^TX = I_p\}$ as a Riemannian submanifold of the Euclidean space $\mathbb{R}^{n\times p}$ endowed with the canonical Riemannian metric $g(\xi,\eta) = \mathrm{tr}(\xi^T\eta)$. The tangent space at X and the associated orthogonal projection are given by

$$\begin{aligned} T_X\mathrm{St}(p,n) &= \{Z \in \mathbb{R}^{n\times p} : X^TZ + Z^TX = 0\} \\ &= \{X\Omega + X^\perp K : \Omega^T = -\Omega, K \in \mathbb{R}^{(n-p)\times p}\} \\ \mathrm{P}_X\xi_X &= (I - XX^T)\xi_X + X\mathrm{skew}(X^T\xi_X) \end{aligned}$$

We use the retraction given by $R_X(\eta_X) = \mathrm{qf}(X + \eta_X)$, where $\mathrm{qf}(A)$ denotes the Q factor of decomposition of $A \in \mathbb{R}_*^{n\times p}$ as $A = QR$, where $\mathbb{R}_*^{n\times p}$ denotes the set of all nonsingular $n \times p$ matrices, $Q \in \mathrm{St}(p,n)$, and R is an upper triangular $n \times p$ matrix with strictly positive diagonal elements.

Vector transport (1) and its inverse on $\mathrm{St}(p,n)$ are given by

$$\mathcal{T}_{\eta_X}\xi_X = (I - YY^T)\xi_X + Y\mathrm{skew}(Y^T\xi_X)$$
$$(\mathcal{T}_{\eta_X})^{-1}\xi_Y = \xi_Y + \zeta,$$

where $Y := R_X(\eta_X)$, ζ is in the normal space at Y which implies $\zeta = YS$ where S is a symmetric matrix, and $(\xi_Y + YS) \in T_x\mathrm{St}(p,n)$ which implies $X^T(\xi_Y + YS)$ is skew symmetric. We therefore have

$$X^TYS + SY^TX + X^T\xi_Y + \xi_Y^TX = 0.$$

Therefore, S can be found by solving a Lyapunov equation.

For $\mathrm{St}(p,n)$, the parallel transport of $\xi \neq H$ along the geodesic $\gamma(t)$ from Y in direction H, denoted by $w(t) = P_\gamma^{t\leftarrow 0}\xi$, satisfies [8, §2.2.3]:

$$w'(t) = -\frac{1}{2}\gamma(t)(\gamma'(t)^Tw(t) + w(t)^T\gamma'(t)), \quad w(0) = \xi. \tag{12}$$

In practice, the differential equation is solved numerically and the computational cost of parallel transport may be significantly higher than that of vector transport.

5 Applications and numerical experiment results

We have experimented extensively with the versions of RBFGS described above. Here we present the results of two problems that provide leading evidence supporting the value of using retraction and vector transport in RBFGS and its limits. We obtained similar iteration counts using different x_0.

For a symmetric matrix A, the unit-norm eigenvector, v, corresponding to the smallest eigenvalue, defines the two global minima, $\pm v$, of the Rayleigh quotient $f : S^{n-1} \to \mathbb{R}$, $x \mapsto x^TAx$. The gradient of f is given by

$$\mathrm{grad}\, f(x) = 2\mathrm{P}_x(Ax) = 2(Ax - xx^TAx).$$

We show results of the minimization of the Rayleigh quotient to illustrate the performance of RBFGS on S^{n-1}.

On $\mathrm{St}(p,n)$ we consider a matrix Procrustes problem that minimizes the cost function $f : \mathrm{St}(p,n) \to \mathbb{R}$, $X \to \|AX - XB\|_F$ given $n \times n$ and $p \times p$ matrices A and B respectively. The gradient of f on the submanifold of $\mathbb{R}^{n\times p}$ used to represent $\mathrm{St}(p,n)$ is

$$\mathrm{grad}\, f(X) = \mathrm{P}_X\mathrm{grad}\, \bar{f}(X) = Q - X\mathrm{sym}(X^TQ),$$
$$Q := A^TAX - A^TXB - AXB^T + XBB^T.$$

The versions of RBFGS that update B and B^{-1} perform similarly for these problems so we report data from the B^{-1} version. Approach 1 and Approach 2 display similar convergence behavior and on these manifolds Approach 2 has a higher computational complexity so we report data from Approach 1.

Table 1. Vector transport vs. Parallel transport

	Rayleigh $n = 300$		Procrustes $(n, p) = (12, 7)$	
	Vector	Parallel	Vector	Parallel
Time (sec.)	4.0	4.2	24.0	304.0
Iteration	97	95	83	175

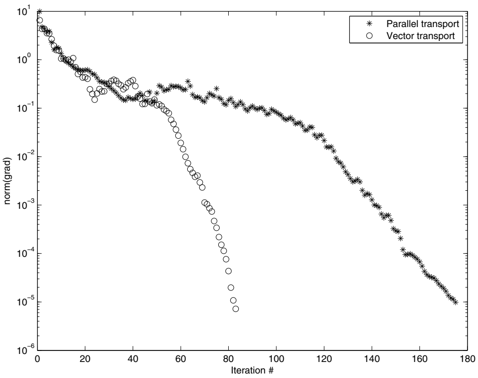


Fig. 2. Update of B^{-1}, Parallel and Vector Transport for Procrustes. n=12, p=7.

Since parallel transport and vector transport by projection have similar computational costs on S^{n-1}, the corresponding RBFGS versions have a similar computational cost per iteration. Therefore, we would expect any performance difference measured by time to reflect differences in rates of convergence. Columns 2 and 3 of Table 1 show that vector transport produces a convergence rate very close to parallel transport and the times are close as expected. This is encouraging from the point of view that the more flexible vector transport did not significantly degrade the convergence rate of RBFGS.

Given that vector transport by projection is significantly less expensive computationally than parallel transport on $\mathrm{St}(p, n)$, we would expect a significant improvement in performance as measured by time if the vector transport version manages to achieve a convergence rate similar to parallel transport. The times in columns 4 and 5 of Table 1 show an advantage to the vector transport version larger than the computational complexity predicts. The iteration counts provide an explanation. Encouragingly, the use of vector transport actually improves convergence compared to parallel transport. We note that the parallel transport version performs the required numerical integration of a differential equation with a stepsize sufficiently small so that decreasing it does not improve the convergence rate of RBFGS but no smaller to avoid unnecessary computations. Figure 2 illustrates in more detail the significant improvement in convergence rate achieved for vector transport. It provides strong evidence that a careful consideration of the choice of vector transport may have significant beneficial effects on both cost per step and overall convergence. More detailed consideration of this observation and the convergence theory for RBFGS will be presented in a future paper.

References

1. P.-A. Absil, R. Mahony, and R. Sepulchre (2010) Optimization on manifolds: methods and applications. In: Diehl M., Glineur F., Michiels W. (eds) Recent Trends in Optimization and its Applications in Engineering.
2. P.-A. Absil, C. G. Baker, and K. A. Gallivan (2007) Trust-region methods on Riemannian manifolds. Found. Comput. Math., 7(3):303–330
3. P.-A. Absil, R. Mahony, and R. Sepulchre (2008) Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton, NJ
4. Roy L. Adler, Jean-Pierre Dedieu, Joseph Y. Margulies, Marco Martens, and Mike Shub (2002) Newton's method on Riemannian manifolds and a geometric model for the human spine. IMA J. Numer. Anal., 22(3):359–390
5. N. Del Buono and C. Elia (2003) Computation of few Lyapunov exponents by geodesic based algorithms. Future Generation Computer systems, 19: 425-430
6. Ian Brace and Jonathan H. Manton (2006) An improved BFGS-on-manifold algorithm for computing weighted low rank approximations. In Proceedings of the 17h International Symposium on Mathematical Theory of Networks and Systems, pages 1735–1738
7. John E. Dennis, Jr. and Robert B. Schnabel (1983) Numerical methods for unconstrained optimization and nonlinear equations. Prentice Hall Series in Computational Mathematics, Prentice Hall Inc., Englewood Cliffs, NJ
8. Alan Edelman, Tomás A. Arias, and Steven T. Smith (1998) The geometry of algorithms with orthogonality constraints. SIAM J. Matrix Anal. Appl., 20(2):303–353
9. D. Gabay (1982) Minimizing a differentiable function over a differential manifold. J. Optim. Theory Appl., 37(2):177–219
10. Jorge Nocedal and Stephen J. Wright (2006) Numerical optimization. Springer Series in Operations Research and Financial Engineering, Springer, New York, second edition
11. Berkant Savas and Lek-Heng Lim (2008) Best multilinear rank approximation of tensors with quasi-Newton methods on Grassmannians. Technical Report LITH-MAT-R-2008-01-SE, Department of Mathematics, Linköpings University

Acknowledgements

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors. This work was supported in part by "Communauté française de Belgique - Actions de Recherche Concertées".

Identification Method for Time-Varying ARX Models

Quentin Rentmeesters, P.-A. Absil, and Paul Van Dooren

Département d'ingénierie mathématique
Université catholique de Louvain
B-1348 Louvain-la-Neuve, Belgium
{Quentin.Rentmeesters,PA.Absil,Paul.Vandooren}@uclouvain.be

Summary. This paper presents a new approach to identify time-varying ARX models by imposing a penalty on the coefficient variation. Two different coefficient normalizations are compared and a method to solve the two corresponding optimization problems is proposed.

1 Introduction

Time-varying processes appear in many applications such as speech processing, time-varying behavior detection (fault detection or wear detection) or more generally when some parameters of a linear system vary over time. In this paper, we are interested in time-varying systems identification using an ARX model of order $N-1$:

$$\sum_{i=0}^{N-1} y(t-i)\alpha_i(t) = \sum_{i=0}^{N-1} u(t-i)\beta_i(t) \tag{1}$$

where y is the output of the time-varying system, u is the input and $\alpha_i(t)$ and $\beta_i(t)$ are the coefficients of the model at time t.

Several approaches have been adopted to deal with time-varying modeling problems. One of the most popular ones is to use an adaptive algorithm that computes iteratively the coefficients of the model; see, e.g., [1]. This approach works quite well under the assumption that the time variations are slow.

Another approach is to expand the coefficients of the model in a finite set of basis functions [2]. The problem then becomes time-invariant with respect to the parameters in the expansion and is hence reduced to a least squares problem. The two main issues which are encountered when this approach is applied to general time-varying systems, are how to choose a family of basis functions, and how to select finitely many significant ones.

Here, we consider a method which identifies the time-varying coefficients in a fixed time window. This method is not recursive and does not assume

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_17, © Springer-Verlag Berlin Heidelberg 2010

strong hypotheses on the evolution of the coefficients. Moreover, at each time step, a value for the coefficients of the model is identified. Thus, it is not necessary to find a basis to expand the coefficients which is an important practical advantage. It will still be possible to choose a basis of functions to expand the coefficients after the identification to reduce the space complexity of the identified model. Our approach is based on a trade-off between the minimization of the prediction error and the minimization of the variation of the coefficients. The penalization of the variation of the coefficients enables the reduction of high frequency noises and the use of classical techniques to find the order of the model.

The paper is organized as follows. Section 2 introduces our approach and describes a method to solve efficiently the least squares problem that arises. Section 3 presents another normalization of the cost function introduced in section 2 that leads to an optimization problem on the Cartesian product of spheres. Numerical experiments and some ways to find the parameters of the method are presented in section 4.

2 Our approach

On the one hand, the coefficients must be allowed to vary sufficiently to deal with possibly large coefficient variations and to fit the data points. But, on the other hand, the variation of the coefficients must be penalized to reduce the influence of high frequency noises or outliers. To achieve this trade-off, the following cost function is considered:

$$\min_{X(0),\dots,X(T-1)} \sum_{t=1}^{T-1} \|X(t) - X(t-1)\|_2^2 + \mu \sum_{t=0}^{T-1} \|\phi^\top(t)X(t)\|_2^2, \qquad (2)$$

where T is the size of the time window where the identification is performed, $X(t) = \left[\alpha_0(t), \beta_0(t), \dots, \alpha_{N-1}(t), \beta_{N-1}(t)\right]^\top$ is the coefficient vector and $\phi(t) = \left[y(t), -u(t), \dots, y(t-N+1), -u(t-N+1)\right]^\top$ is the data vector. It is also possible to identify the model structure (1) where some of the coefficients are set to zero: it suffices to delete the coefficients in $X(t)$ and the corresponding inputs or outputs in $\phi(t)$.

The first term imposes that the coefficients do not vary too fast and the second term corresponds to the square of prediction error. The parameter $\mu > 0$ can be chosen to find a compromise between fitting the data and preventing the coefficients from varying too quickly.

This problem admits the trivial solution: $X(t) = 0$ for all t. Consequently, we must normalize the coefficient vector. Two kinds of normalizations are considered: fixing one coefficient at 1 for all t, and imposing $\|X(t)\| = 1$ for all t. The first one yields a least squares problem. The second one yields an optimization problem on the Cartesian product of spheres and is the subject of the next section.

The rest of this section explains how to solve the problem efficiently when the normalization: $\alpha_0(t) = 1$ for all t is chosen. In this case, the problem (2) can be rewritten as the following least squares problem:

$$\min_{X_2} \left\| \underbrace{\begin{bmatrix} I_{2N-1} & -I_{2N-1} & & & & \\ & I_{2N-1} & -I_{2N-1} & & & \\ & & \ddots & \ddots & & \\ & & & \ddots & -I_{2N-1} & \\ & & & & I_{2N-1} & -I_{2N-1} \\ \sqrt{\mu}\phi_2^\top(0) & & & & & \\ & \sqrt{\mu}\phi_2^\top(1) & & & & \\ & & \ddots & & & \\ & & & \ddots & & \\ & & & & \sqrt{\mu}\phi_2^\top(T-2) & \\ & & & & & \sqrt{\mu}\phi_2^\top(T-1) \end{bmatrix}}_{A_2} \underbrace{\begin{bmatrix} X_2(0) \\ \vdots \\ \vdots \\ X_2(T-1) \end{bmatrix}}_{X_2} - \underbrace{\begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ -\sqrt{\mu}y(0) \\ -\sqrt{\mu}y(1) \\ \vdots \\ \vdots \\ -\sqrt{\mu}y(T-2) \\ -\sqrt{\mu}y(T-1) \end{bmatrix}}_{b} \right\|_2^2$$

where $X_2(t) = [\beta_0(t), \ldots, \alpha_{N-1}(t), \beta_{N-1}(t)]^\top$ and $\phi_2(t) = [-u(t), y(t-1), \ldots]^\top$. To preserve the structure, a method based on the normal equations ($A_2{}^\top A_2 X_2 = A_2{}^\top b$) is proposed to solve the problem. The $A_2{}^\top A_2$ matrix is:

$$\underbrace{\begin{bmatrix} I & -I & & & \\ -I & 2I & -I & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -I \\ & & & \ddots & 2I & -I \\ & & & & -I & I \end{bmatrix}}_{M} + \mu \underbrace{\begin{bmatrix} \phi_2(0)\phi_2^\top(0) & & & & \\ & \phi_2(1)\phi_2^\top(1) & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & \phi_2(T-1)\phi_2^\top(T-1) \end{bmatrix}}_{\Phi} \tag{3}$$

where I is the identity matrix of size $2N-1$.

The matrix $A_2{}^\top A_2$ is block tri-diagonal and is the sum of two positive semi-definite matrices M and Φ. Hence, $A_2{}^\top A_2$ is invertible if the kernel of M has no intersection with the kernel of Φ. The eigenvalues λ_k and the corresponding eigenspaces v_k of M are (see [3]):

$$v_k = \left[\cos((0+\tfrac{1}{2})\tfrac{k\pi}{T})I \cdots \cos((j+\tfrac{1}{2})\tfrac{k\pi}{T})I \cdots \cos(((T-1)+\tfrac{1}{2})\tfrac{k\pi}{T})I\right]$$

$$\lambda_k = 2 - 2\cos(\frac{k\pi}{T}) \qquad 0 \le k \le T-1.$$

The eigenspace relative to $\lambda_0 = 0$ is: $v_0 = \left[I \ldots I\right]^\top$. Consequently, in order to get a unique solution, the following condition is required:

$$v_0^\top A_2{}^\top A_2 v_0 = \mu v_0^\top \Phi v_0 = \mu \sum_{i=0}^{T-1} \phi_2(i)\phi_2(i)^\top \succ 0.$$

This is true if $\lambda_{\min}\left(\sum_{i=0}^{T-1}\phi_2(i)\phi_2(i)^\top\right) > 0$ which means that the data vector $\phi_2(t)$ must span a space of dimension $2N-1$ on the whole time horizon of size T. This condition will be easily satisfied if the input is sufficiently exciting and if the order of the model is not overestimated. Notice that this tells no information about the reliability of the identified coefficients. To be able to recover the true coefficients of a model, the data should be unperturbed and as exciting as possible. If $\lambda_{\min}\left(\sum_{i=k}^{k+2N-2}\phi_2(i)\phi_2(i)^\top\right) > 0 \quad \forall k$, the data are very informative, and this will provide a more reliable approximation of the coefficients.

The system of normal equations can be efficiently solved by performing a block tri-diagonal LU factorization of the ${A_2}^\top A_2$ matrix (3), see [4] for more details. This decomposition has a complexity of $\mathrm{O}((T-1)(2N-1)^3)$ operations which is linear in T.

Using the same technique, it is also possible to normalize another coefficient than α_0 and to take into account already known coefficients by fixing them at their value. Unfortunately, the solution of the problem will depend on the coefficient which is normalized, that is why another normalization is proposed in the next section.

3 Normalization of the coefficient vector

In this section, we explain why it can be interesting to normalize the coefficient vector, i.e., fixing $\|X(t)\| = 1$ for all t and we describe the method used to solve the corresponding optimization problem.

The main idea behind this normalization is the following. The ARX relation (1) can be rewritten as:

$$X(t)^\top \phi(t) = 0$$

and is unchanged if it is multiplied by a scalar $\gamma(t) \neq 0$ which means that $\gamma(t)X(t)$ corresponds to the same ARX model as $X(t)$. Consequently, an ARX model at time t is not represented by a particular coefficient vector but by a direction in $\mathbb{R}^{2N}$. Hence, a good notion of distance between two ARX models is the relative angle. In fact, this notion of distance does not depend on the particular choice of vector in $\mathbb{R}^{2N}$ used to represent an ARX model. When $\|X(t)\| = 1$ for all t, the first term of (2) becomes:

$$\sum_{t=1}^{T-1} 4\sin^2\left(\frac{\angle X(t)X(t-1)}{2}\right)$$

and only depends on the angle $\angle X(t)X(t-1)$ between two coefficient vectors representing two ARX models at consecutive time steps.

This is also a more neutral normalization because the cost on the variation of the coefficients is uniformly distributed over all coefficients, as opposed to the normalization of the α_0 coefficient. In fact, when the α_0 coefficient is normalized, the distance between two ARX models represented by $\|\frac{X(t)}{\alpha_0(t)} - \frac{X(t-1)}{\alpha_0(t-1)}\|_2^2$ will be larger if the model at time t is well represented by a model whose α_0 coefficient gets close to 0 and lower if the model at time t is well represented by a model whose α_0 coefficient is large. This is shown in the following example. At time $t = 150$, the α_0 coefficient of the following system:

$$\begin{aligned}
\alpha_0(t) &= 0.5 + 0.45 \sin\left(\tfrac{t2\pi}{200}\right) \quad 1 \leq t \leq 200 \\
\beta_0(t) &= 5 \\
\alpha_1(t) &= 0.01 \\
\beta_1(t) &= -4
\end{aligned}$$

gets close to zero. Fig. 1. shows the identified β_0 coefficient using the two normalizations. If the coefficient α_0 is normalized, the true coefficient is not recovered in the neighborhood of $t = 150$ because a coefficient variation is highly penalized in this neighborhood. This is avoided when the coefficient vector is normalized since the cost on the variation of the coefficients depends only on the angle.

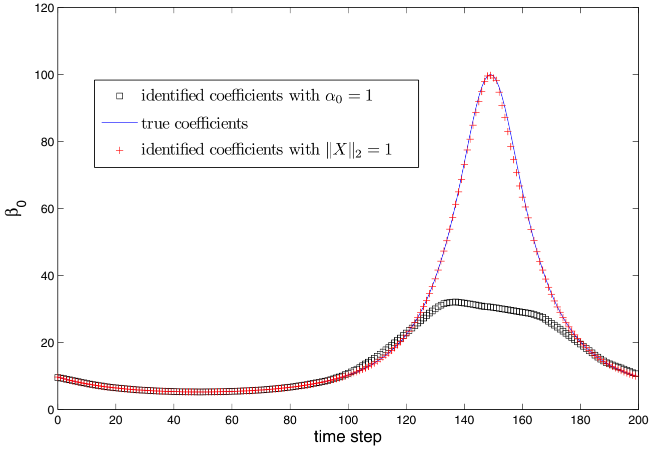


Fig. 1. true and identified coefficient β_0 when $\|X(t)\|_2 = 1$ for all t

With this constraint, the optimization problem (2) is no longer a least squares problem and an optimization technique on manifolds is proposed. We will only describe the main points of this method. For more details, see [5].

By introducing the following notation:

$$Y = \begin{bmatrix} X(0) \\ \vdots \\ X(T-1) \end{bmatrix} = \begin{bmatrix} X_0 \\ \vdots \\ X_{T-1} \end{bmatrix} \in \mathbb{R}^{2NT},$$

the constraint $\|X(t)\| = 1$ for all t can be also rewritten as: $Y \in (\mathbb{S}^{2N-1})^T$ where $(\mathbb{S}^{2N-1})^T$ stands for the Cartesian product of T unit spheres in $\mathbb{R}^{2N}$:

$$(\mathbb{S}^{2N-1})^T = \underbrace{S^{2N-1} \times \cdots \times S^{2N-1}}_{T} \subset \mathbb{R}^{2NT}$$

where $S^{2N-1} = \{x \in \mathbb{R}^{2N} | x^\top x = 1\}$ is the unit sphere in $\mathbb{R}^{2N}$. This is a submanifold of $\mathbb{R}^{2NT}$ and its tangent space at Y is:

$$T_Y(\mathbb{S}^{2N-1})^T = \{Z = \begin{bmatrix} Z_0 \dots Z_{T-1} \end{bmatrix}^\top \in \mathbb{R}^{2NT} | X_i^\top Z_i = 0 \quad 0 \leq i \leq T-1\}.$$

The orthogonal projection on this tangent space at the point Y is given by:

$$P_Y(Z) = \begin{bmatrix} P_{X_0}(Z_0) \\ \vdots \\ P_{X_{T-1}}(Z_{T-1}) \end{bmatrix} = \begin{bmatrix} (I_{2N} - X_0 X_0^\top) Z_0 \\ \vdots \\ (I_{2N} - X_{T-1} X_{T-1}^\top) Z_{T-1} \end{bmatrix}.$$

Then, the problem (2) becomes the following optimization problem on $(\mathbb{S}^{2N-1})^T$:

$$\boxed{\begin{array}{c} \min\limits_{Y} \quad \overline{f} : \mathbb{R}^{2NT} \longrightarrow \mathbb{R}, Y \longrightarrow Y^\top A^\top A Y \\ \text{s.t.} \quad Y \in (\mathbb{S}^{2N-1})^T \end{array}}$$

where the $A^\top A$ matrix is given by:

$$\begin{bmatrix} I + \mu\Phi(0) & -I & & & & \\ -I & 2I + \mu\Phi(1) & -I & & & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & -I & \\ & & & \ddots & 2I + \mu\Phi(T-2) & -I \\ & & & & -I & I + \mu\Phi(T-1) \end{bmatrix} \tag{4}$$

with $\Phi(t) = \phi(t)\phi^\top(t)$ and I is the identity matrix of size $2N$. The restriction of $\overline{f}$ to $(\mathbb{S}^{2N-1})^T$ is denoted by f.

A Newton method on the Cartesian product of spheres has been chosen to solve this problem because our numerical experiments have shown that the solution of the least squares problem (when α_0 is normalized) belongs to the attraction basin of the Newton method. So, the solution of the least squares problem can be used as a starting value for the local optimization problem on the Cartesian product of spheres. The Newton equation is given by:

$$\nabla_Z \mathrm{grad}\, f = -\mathrm{grad}\, f(Y), \quad Z \in T_Y(\mathbb{S}^{2N-1})^T \tag{5}$$

where grad $f(Y)$ represents the gradient at the current iterate Y and $\nabla_Z \mathrm{grad}\, f$ stands for the Riemannian covariant derivative of the vector field grad $f(Y)$ in the direction Z where Z will be the next Newton direction.

To implement this method, an expression for the gradient and for the Riemannian connection ∇ is required. The gradient with respect to the induced metric is the unique element grad $f(Y)$ of $T_Y(\mathbb{S}^{2N-1})^T$ which satisfies:

$$\mathrm{grad}\, f(X)^\top Z = DF(Y)[Z] \quad \forall Z \in T_Y(\mathbb{S}^{2N-1})^T$$

where $DF(Y)[Z]$ stands for the differential at Y in the direction Z. In our case, this gives:

$$\mathrm{grad}\, f(Y) = P_Y(2A^\top AY).$$

Since $(\mathbb{S}^{2N-1})^T$ is an $\mathbb{R}^n$ submanifold of the Euclidean space $\mathbb{R}^{2NT}$, the $\mathbb{R}^n$ connection is equivalent to the classical directional derivative in $\mathbb{R}^{2NT}$ followed by a projection on the tangent space at Y: $\nabla_Z \mathrm{grad}\, f = P_Y(D\mathrm{grad}\, f(Y)[Z])$. Since

$$(D\mathrm{grad}\, f(Y)[Z])_i = 2((-X_i Z_i^\top - Z_i X_i^\top)B_i Y + P_{X_i}(B_i Z)),$$

the Newton equation (5) becomes:

$$2\begin{bmatrix} P_{X_0}(B_0 Z) - Z_0 X_0^\top B_0 Y \\ \vdots \\ P_{X_{T-1}}(B_{T-1}Z) - Z_{T-1}X_{T-1}^\top B_{T-1}Y \end{bmatrix} = -\mathrm{grad}\, f(Y) \tag{6}$$

$$Z \in T_Y(\mathbb{S}^{2N-1})^T \tag{7}$$

where B_i is the block matrix composed of the rows $i2N+1$ up to $(i+1)2N$ and all the columns of $A^\top A$ in (4). By introducing the following change of variables,

$$\omega_i = X_i^{\perp\top} Z_i \text{ where } [X_i|X_i^\perp]^\top [X_i|X_i^\perp] = I_{2N}$$

the condition (7) is trivially satisfied and (6) becomes:

$$\begin{aligned} K_0\omega_0 - X_0^{\perp\top} X_1^\perp \omega_1 &= -X_0^{\perp\top} B_0 Y \\ -X_i^{\perp\top} X_{i-1}^\perp \omega_{i-1} + K_i\omega_i - X_i^{\perp\top} X_{i+1}^\perp \omega_{i+1} &= -X_i^{\perp\top} B_i Y \text{ for } 1 \le i \le T-2 \\ -X_{T-1}^{\perp\top} X_{T-2}^\perp \omega_{T-2} + K_{T-1}\omega_{T-1} &= -X_{T-1}^{\perp\top} B_{T-1} Y \end{aligned}$$

where $K_i = X_i^{\perp\top} \mu\Phi(i) X_i^\perp - I X_i^\top B_i Y$. This system is block tri-diagonal and can be easily solved using a block LU factorization which requires $O((T-1)(2N-1)^3)$ operations. Consequently from a computational complexity point of view, one iteration of this Newton method is equivalent to the least squares method presented in the previous section. Once the Newton step Z has been computed, the following retraction:

$$R_Y(Z) = \begin{bmatrix} \frac{X_0+Z_0}{\|X_0+Z_0\|} \\ \vdots \\ \frac{X_{T-1}+Z_{T-1}}{\|X_{T-1}+Z_{T-1}\|} \end{bmatrix}$$

can be used to compute the update $Y_+ = R_Y(Z)$.

4 Choice of μ and the order

In this section, some numerical experiments and methods to select or gain some insight in the μ parameter value and the order of the system are presented. Let us consider the system defined by the following coefficient vector:

$$X(t) = \begin{bmatrix} \alpha_0(t) \\ \beta_0(t) \\ \alpha_1(t) \\ \beta_1(t) \\ \alpha_2(t) \\ \beta_2(t) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 - 0.2e^{-\left(\frac{t-100}{50}\right)^2} \\ -0.8 \\ -0.2 \\ 0.6 \\ -0.6 \end{bmatrix}.$$

This system was simulated with a white noise of unitary variance as input. The output was perturbed in the following way: $y(t) \leftarrow y(t) + \Delta|y(t)|U(t)$ where $U(t)$ is a random variable distributed uniformly on $[-1, 1]$. Fig. 2. shows the error on the coefficients in function of μ for different levels of perturbation. For an unperturbed model ($\Delta = 0$), the error on the coefficients is smaller for a large value of μ because the bias introduced by the first term of our cost function is reduced. For a perturbed system, it is not optimal to trust too much the data, and there exists an optimal value of μ that minimizes the error on the coefficients. To get an insight of this optimal value of μ in practice, we can look at the identified coefficient β_0 shown in Fig. 3. For a small value of μ, we get an almost constant coefficient and for a large value of μ we identify a coefficient that oscillates around the true coefficient. This means that we are identifying the noise. So it is possible to get an idea of the best value of μ that makes a desired trade-off between the slow coefficient variation or equivalently the risk of bias and the rejection of the perturbations.

The notion of order for a time invariant system somehow represents the complexity of the model. If this complexity is increased, the model will better fit the data. So, a common criterion to find the order of a time-invariant system consists in measuring the fitting error (the prediction error in our case) and selecting the order that corresponds to a drop on the fit level. This idea does not directly extend to time-varying models. In fact, even with a time-varying model of order 0, it is easy to make the fitting error go to 0. But by imposing a cost on the variation of the coefficients, the same idea can be applied as shown in the following experiment. A time-varing ARX system of order 4 was identified using different models (different values of the order) and different

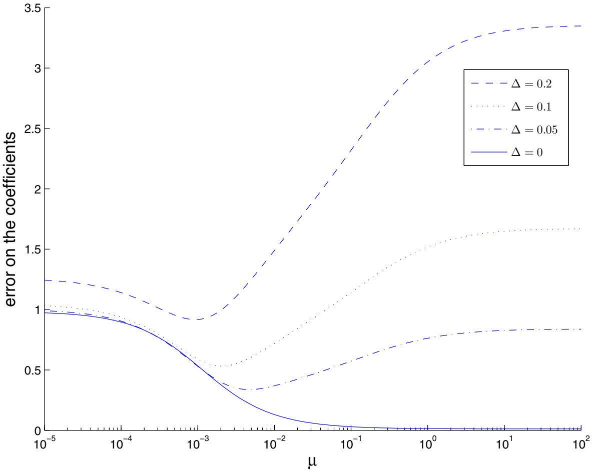


Fig. 2. difference between the true X_2 and the identified coefficients $\tilde{X}_2$: $\|X_2 - \tilde{X}_2\|_2$ in function of μ for different levels of perturbation Δ

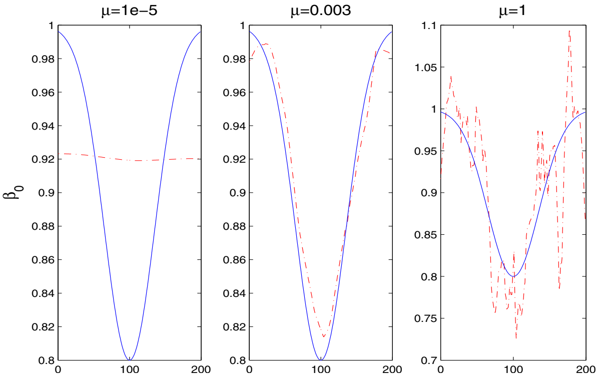


Fig. 3. identified (.-) and true (-) coefficients β_0 for different values of μ when $\Delta = 0.1$

values of μ, see Fig. 4. When we go from a model of order 3 to a model of order 4, the error drops and remains rather constant if the order is further increased. This drop indicates that the model order is probably 4 and it is interesting to notice that this conclusion does not depend on the value of μ.

5 Conclusions

We have presented a method to identify a time-varying ARX model by penalizing the variation of the coefficients. By doing so, we can choose the order

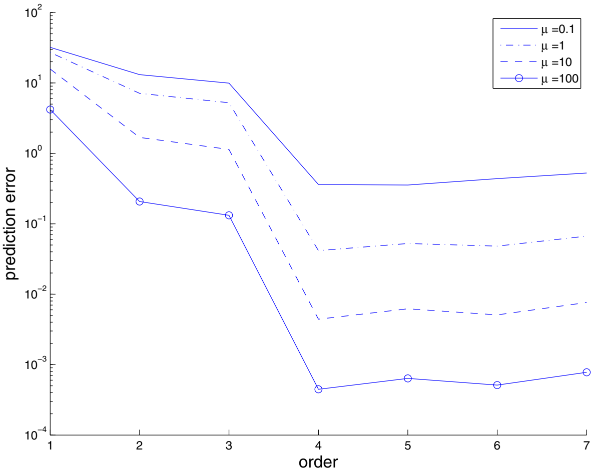


Fig. 4. prediction error ($\sum_{t=0}^{T-1} \|\phi^\top(t)\tilde{X}(t)\|_2$ where $\tilde{X}(t)$ stands for the identified coefficient vector) as a function of the order, for different values of μ

using classical techniques and the influence of the perturbations can be reduced. A more neutral normalization of the coefficient vector has also been proposed. This normalization leads to better results on models whose α_0 coefficient gets close to 0. In later work, we will extend these methods to MIMO systems. When the coefficient matrix is normalized, this yields an optimization problem on the Cartesian product of Grassmann manifolds.

References

1. L. Guo and L. Ljung, Performance analysis of general tracking algorithms, IEEE Trans. Automat. Control, volume 40, pages 1388–1402, 1995.
2. H.L. Wei and S.A. Billings, Identification of time-varying systems using multiresolution wavelet models, International Journal of Systems Science, 2002.
3. G. Strang, The discrete cosine transform, SIAM Review, volume 41, pages 135–147 (electronic), 1999.
4. G.H. Golub, and C.F. Van Loan, Matrix Computations second edition, Johns Hopkins University Press, 1989.
5. P.-A. Absil, R. Mahony and R. Sepulchre, Optimization algorithms on matrix manifolds, Princeton University Press, 2008.

Acknowledgement. This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its author(s). This work was also supported by “Communauté française de Belgique - Actions de Recherche Concertées”. The authors are grateful to Rik Pintelon and John Lataire for fruitful discussions.

Part IV

Optimal Control

On Some Riemannian Aspects of Two and Three-Body Controlled Problems*

J.-B. Caillau[1], B. Daoud[1], and J. Gergaud[2]

[1] Math. Institute, Bourgogne Univ. & CNRS
{jean-baptiste.caillau,bilel.daoud}@u-bourgogne.fr
[2] ENSEEIHT-IRIT, Toulouse Univ. & CNRS gergaud@enseeiht.fr

Dedicated to Joseph Noailles

Summary. The flow of the Kepler problem (motion of two mutually attracting bodies) is known to be geodesic after the work of Moser [21], extended by Belbruno and Osipov [2, 22]: Trajectories are reparameterizations of minimum length curves for some Riemannian metric. This is not true anymore in the case of the three-body problem, and there are topological obstructions as observed by McCord *et al.* [20]. The controlled formulations of these two problems are considered so as to model the motion of a spacecraft within the influence of one or two planets. The averaged flow of the (energy minimum) controlled Kepler problem with two controls is shown to remain geodesic. The same holds true in the case of only one control provided one allows singularities in the metric. Some numerical insight into the control of the circular restricted three-body problem is also given.

Key words: Two and three-body problems, geodesic flow, optimal control, cut and conjugate loci

MSC classification. 49K15, 53C20, 70Q05

1 Introduction

The circular restricted three-body problem is defined as follows [26].

> *Two bodies describe circular orbits around their center of mass under the influence of their mutual gravitational attraction, and a third one (attracted by the previous two but not influencing their motion) evolves*

* The first author is supported by Conseil Régional de Bourgogne (contract no. 2009-160E-160-CE-160T) and ANR GCM. The second author is supported by the French Ministry for Higher Education & Research (grant no. 31716-2008).

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_18, © Springer-Verlag Berlin Heidelberg 2010

> *in the plane defined by the two rotating ones. The restricted problem is to describe the motion of this third body.*

We investigate the optimal control of this problem. The two primaries are planets, typically Earth and Moon, the third body is a spacecraft. The control is the thrust of this spacecraft. A recent example of this problem is the SMART-1 mission [4, 23] of the European Space Agency in the Earth-Moon system. This case has three important features: (i) Provided we neglect non-coplanar effects, the circular restricted model is germane to the problem as the eccentricity of the Moon orbit is about 0.0549; (ii) The mass m_2 of the second primary (the Moon) is much smaller than the mass of the first (the Earth), m_1, so that $\mu = m_2/(m_1 + m_2) \simeq 0.0121$ is a small parameter of the model; (iii) The thrust of the engine is very low since solar-electric propulsion is used (around 0.07 Newtons for a 350 Kilogram vehicle), so the magnitude of the control is another small parameter.

In a rotating frame, the dynamics is normalized to the second order mechanical system

$$\ddot{q} + \nabla V_\mu(q) + 2iq = \varepsilon u, \quad |u| = \sqrt{u_1^2 + u_2^2} \le 1.$$

Coordinate $q \in \mathbf{C} \simeq \mathbf{R}^2$ is the position vector while u is the control (the normalized acceleration, here). In this moving frame, the circular restricted three-body potential is

$$V_\mu(q) = -\frac{q^2}{2} - \frac{1-\mu}{r_1} - \frac{\mu}{r_2},$$

$$r_1^2 = (q_1 + \mu)^2 + q_2^2, \quad r_2^2 = (q_1 - 1 + \mu)^2 + q_2^2.$$

Parameter μ is the ratio $m_2/(m_1 + m_2)$ of the masses of the two primaries, and ε is the bound on the acceleration. When μ vanishes, we have a controlled two-body problem. The uncontrolled equations of motion can also be written in Hamiltonian form using *Jacobi* first integral (total energy),

$$J_\mu(q, \dot{q}) = \frac{|\dot{q}|^2}{2} + V_\mu(q).$$

In complex notation, let $p = \dot{q} + iq$. Then

$$J_\mu(q, p) = \frac{|p|^2}{2} + p_2 q_1 - p_1 q_2 - \frac{1-\mu}{r_1} - \frac{\mu}{r_2}.$$

The controlled system with Hamiltonian drift is so

$$\dot{q} = \frac{\partial J_\mu}{\partial p}, \quad \dot{p} = -\frac{\partial J_\mu}{\partial q} + \varepsilon u, \quad |u| \le 1. \tag{1}$$

In the case of two bodies ($\mu = 0$) and no control ($\varepsilon = 0$), the equations of motion in a fixed frame are

$$\ddot{q} + \frac{q}{|q|^3} = 0, \quad q \in \mathbf{R}^2 - \{0\}. \tag{2}$$

In Hamiltonian form,

$$\dot{q} = \frac{\partial J_0}{\partial p}, \quad \dot{p} = -\frac{\partial J_0}{\partial q},$$

with energy $J_0 = |\dot{q}|^2/2 - 1/|q| = |p|^2/2 - 1/|q|$, as $p = \dot{q}$ in the fixed frame. It was proven in [21] that, for negative values of the energy, the Hamiltonian flow of the system is a reparameterization of the geodesic flow on the punctured two-sphere, $\hat{\mathbf{S}}^2 = \mathbf{S}^2 - \{N\}$ (North pole removed). We sketch the construction in dimension $n \geq 2$ where the result holds unchanged. (Take $q \in \mathbf{R}^n - \{0\}$ in (2).) One first identifies the tangent bundle of the punctured n-sphere with the set of vectors $\xi = (\xi_0, \dots, \xi_n)$, $\eta = (\eta_0, \dots, \eta_n)$ of $\mathbf{R}^{n+1}$ such that

$$|\xi| = 1, \quad (\xi|\eta) = 0.$$

The puncture is obtained by removing $\xi_0 = 1$. Then, the transformation from the tangent bundle to $\mathbf{R}^{2n}$ is

$$q_i = (1-\xi_0)\eta_i + \eta_0\xi_i, \quad p_i = -\frac{\xi_i}{1-\xi_0}, \quad i = 1, \dots, n.$$

Provided time is changed according to

$$\mathrm{d}t = |q|\mathrm{d}s, \tag{3}$$

the Hamiltonian flow on $J_0 = -1/2$ is mapped into the Hamiltonian flow on $\widetilde{J}_0 = 1/2 \subset T\hat{\mathbf{S}}^n$ where

$$\widetilde{J}_0(\xi, \eta) = \frac{1}{2}|\xi|^2|\eta|^2.$$

This level set is the *unit* or *spherical* tangent bundle of $\hat{\mathbf{S}}^2$ since $|\eta| = 1$. There,

$$\xi' = \eta, \quad \eta' = -\xi,$$

so $\xi'' + \xi = 0$ and one actually gets geodesics parameterized by arc length. The Levi-Civita change in time (3) regularizes the collision and the dynamics is extended on the whole n-sphere. The result of [21] was further generalized to cover the case of zero or positive energy levels by [2] and [22].

Trajectories in optimal control are projections of Hamiltonian flows, in general with singularities described by Pontryagin maximization condition. Riemannian problems being the simplest instance of control problems, one may ask whether a given smooth Hamiltonian flow is the reparameterization of some Riemannian flow as in the two-body case. This question is addressed in [20], noting the following fact. Given a flow on an odd $2n-1$-dimensional manifold M, a necessary condition for the flow to be geodesic is that the manifold be the unit tangent bundle of some other manifold of dimension n. This puts topological restrictions on M. These conditions are expressed in terms of the homology of M and applied to the (general) three-body problem. We state the results and recall some basic facts on homology [17].

On a topological space X, a *singular p-simplex* is a continuous map $\sigma_p : \varDelta_p \to X$. Here, $\varDelta_p$ is the standard p-simplex, that is the set of $(t_0, \dots, t_p) \in \mathbf{R}^{p+1}$ such that

$$t_0 + \cdots + t_p = 1, \quad t_i \geq 0, \quad i = 0, \dots, p.$$

Let $v_0, \dots, v_p$ be its vertices. The set $C_p(X)$ of *p-chains* is the free abelian group generated by singular p-simplices. The *boundary* operator $\partial_p : C_p(X) \to C_{p-1}(X)$ is

$$\partial_p(\sigma_p) = \sum_{i=0}^{p} (-1)^p \sigma_p | \varDelta(v_0, \dots, \hat{v}_i, \dots, v_p)$$

where the restriction is on the $(p-1)$-simplex $\varDelta(v_0, \dots, \hat{v}_i, \dots, v_p)$ with vertex v_i removed, implicitly identified with $\varDelta_{p-1}$. Images of $(p+1)$-chains by ∂_{p+1} are *p-boundaries*, and p-chains in the kernel of ∂_p are *p-cycles.* As $\delta_p \delta_{p+1} = 0$, boundaries are cycles while, conversely, one defines the *p-th homology group* $H_p(X)$ as the quotient

$$H_p(X) = \operatorname{Ker} \partial_p / \operatorname{Im} \partial_{p+1}.$$

The rank of the $\mathbf{Z}$-module $H_p(X)$ is β_p, the *p-th Betti number*, and the *Euler-Poincaré* characteristic of M is

$$\chi(M) = \sum_{p=0}^{n} \beta_p.$$

Proposition 1 ([20]). *If M is a non-compact connected orientable manifold of dimension $2n-1$, a necessary condition for it to be the unit tangent bundle of some orientable n-manifold is $\beta_{n-1} \neq 0$.*

Applying this condition to the three-body problem, one gets the following negative result.

Theorem 1 ([20]). *The flow of the planar three-body problem on a negative level of energy is not geodesic.*

In the case of controlled two and three-body problems, there is not much hope to retrieve Riemannian flows, unless one uses some approximation process. The paper is organized as follows. In section 2, we examine the case of two bodies and two controls. Using averaging on a relaxed problem, we show that the flow is Riemannian when the L^2-norm of the control is minimized. Its properties are essentially captured by those of a metric on the two-sphere. The same holds true for the case of two bodies and one control (directed by velocity) provided one allows singularities in the metric. This is addressed in section 3. A preliminary discussion of the restricted three-body and two-control case is made in section 4. The problem is control-affine, with a drift. One can still define the exponential mapping associated with minimum time extremals and compute conjugate points to ensure, as in the Riemannian case, local optimality of trajectories.

2 Two bodies, two controls

We consider an L^2-relaxation of the controlled two-body problem. The bound ε on the control is dropped,

$$\ddot{q} + \frac{q}{|q|^3} = u, \quad u \in \mathbf{R}^2, \tag{4}$$

while the final time, t_f, is a fixed parameter of the criterion:

$$\int_0^{t_f} |u|^2 \mathrm{d}t \to \min .$$

In the sub-Riemannian case, L^2-minimization parameterized by final time can be recast as a minimum time problem with a bound on the control. Both problems coincide, so t_f and ε play dual roles in this sense. The situation is more complicated here because of the Kepler drift in the motion. In order to identify a new small parameter of the problem and perform averaging, we notice that the negative energy level $J_0 < 0$ has a trivial fiber structure. This is apparent in suited geometric coordinates.

The set X of oriented ellipses has moduli space the product manifold $\mathbf{R}_+^* \times \mathbf{S}^2$: Each ellipse is defined by its semi-major axis $a > 0$ (we exclude trivial orbits, $a = 0$), and to any point on $\mathbf{S}^2$, (θ, φ) in standard spherical coordinates, is uniquely associated an eccentricity, $e = \sin\varphi$, an *argument of perigee* (angle of the semi-major axis with a fixed reference axis), θ, and an orientation. The orientation of the ellipse changes when the point goes from one hemisphere to the other. Collisions orbits correspond to the equator $\varphi = \pi/2$ and are included in the model.

Remark 1. Ellipses associated with the poles or the equator have richer symmetries (automorphisms) than others. The moduli space is then said to be *coarse*. It remains finer that the moduli space of conformal ellipses where homothety and rotation moduli (a, θ) would be dropped.

Position on the orbit is defined by the polar angle in the plane or *longitude*, $l \in \mathbf{S}^1$. The state space is hence $\mathbf{S}^1 \times X$, and we have a trivial fiber space whose fiber is the moduli space. To each uncontrolled trajectory on $J_0 < 0$ corresponds a unique point in the fiber, so the drift in (4) has the form

$$F_0(l, x) = \omega(l, x) \frac{\partial}{\partial l}, \quad (l, x) \in \mathbf{S}^1 \times X.$$

(See (5) hereafter for the definition of ω.) Keeping the same notation, let then l be the *cumulated longitude*, associated with the covering

$$\mathbf{R} \ni l \mapsto e^{il} \in \mathbf{S}^1.$$

Choosing l as new time, we recast the problem as control-affine problem on X without drift but with non-autonomous vector fields depending periodically on l,

$$\frac{\mathrm{d}x}{\mathrm{d}l} = u_1 F_1(l,x) + u_2 F_2(l,x), \quad u \in \mathbf{R}^2,$$

$$\int_0^{l_f} |u|^2 \frac{\mathrm{d}l}{\omega(l,x)} \to \min \quad (\text{fixed } l_f).$$

The two vector fields F_1, F_2 on X are periodic in the parameter l. Introducing *mean motion*, $n = a^{-3/2}$, and *true anomaly*, $\tau = l - \theta$, one gets

$$\begin{aligned} F_1(l,x) &= \frac{P^2}{W^2}\left(-\frac{3ne\sin\tau}{1-e^2}\frac{\partial}{\partial n} + \sin\tau\frac{\partial}{\partial e} - \cos\tau\frac{1}{e}\frac{\partial}{\partial\theta}\right),\\ F_2(l,x) &= \frac{P^2}{W^2}\left(-\frac{3nW}{1-e^2}\frac{\partial}{\partial n} + (\cos\tau + \frac{e+\cos\tau}{W})\frac{\partial}{\partial e}\right.\\ &\quad \left. + (\sin\tau + \frac{\sin\tau}{W})\frac{1}{e}\frac{\partial}{\partial\theta}\right), \end{aligned}$$

with $W = 1 + e\cos\tau$. The pulsation is

$$\omega(l,x) = \frac{nW^2}{(1-e^2)^{3/2}}. \tag{5}$$

Averaging on the base space eliminates l, that is the drift in the equation.

The normal maximized Hamiltonian on $\mathbf{S}^1 \times T^*X$ is

$$H(l,x,p) = \frac{\omega}{2}(H_1^2 + H_2^2)(l,x,p),$$

where $H_i = \langle p, F_i(l,x)\rangle$, $i = 1, 2$, are the Hamiltonian lifts of the vector fields. Let

$$\overline{H}(x,p) = \frac{1}{2\pi}\int_0^{2\pi} H(l,x,p)\mathrm{d}l \tag{6}$$

be the averaged Hamiltonian. As $1/l_f$, the new small parameter, tends to zero, the flow of $\overline{H}$ converges uniformly towards the flow of H on $[0, l_f]$. (See [16].) It turns out that the averaged flow is the flow of some Riemannian metric on X, a result which can be traced back to Edelbaum [14]. We refer to [5, 9] for details.

Proposition 2. *The averaged Hamiltonian is*

$$H(x,p) = \frac{p_r^2}{2} + \frac{c^2}{2r^2}\left(\frac{1-\lambda\sin^2\varphi}{\sin^2\varphi}p_\theta^2 + p_\varphi^2\right)$$

with $r = (2/5)n^{5/6}$, $c = \sqrt{2/5}$ *and* $\lambda = 4/5$.

The metric is

$$\mathrm{d}r^2 + \frac{r^2}{c^2}\left(\frac{\sin^2\varphi}{1-\lambda\sin^2\varphi}\mathrm{d}\theta^2 + \mathrm{d}\varphi^2\right).$$

It is Liouville integrable. The integration and the analysis of optimality can be made on the restriction to $\mathbf{S}^2$ by reparameterizing time according to $\mathrm{d}s = c^2\mathrm{d}l/r^2$. (See [6].) This amounts to restricting to a coarser moduli space where homothetic ellipses are identified. The restricted metric is

$$XR(\lambda X)\mathrm{d}\theta^2 + \mathrm{d}\varphi^2 \quad \text{with} \quad R = \frac{1}{1-X} \quad \text{and} \quad X = \sin^2\varphi. \tag{7}$$

As $\chi(\mathbf{S}^2) = 2$, the two vector fields $(XR(X))^{-1/2}\partial\theta$, $\partial/\partial\varphi$ cannot form a global frame on the sphere. They have polar singularities that do not define genuine singularities of the metric.

Remark 2. Coordinates (θ,φ) are associated with the covering of the sphere with two punctures at North and South poles,

$$\mathbf{R}\times(0,\pi)\ni(\theta,\varphi)\mapsto(\sin\varphi\cos\theta,\sin\varphi\sin\theta,\cos\varphi)\in\mathbf{R}^3.$$

One retrieves the standard covering $\exp:\mathbf{C}\to\mathbf{C}^*\simeq\mathbf{S}^2-\{N,S\}$ by putting $(\theta,\varphi)\mapsto\tan(\varphi/2)\exp(i\theta)$.

The Hamiltonian on $\mathbf{S}^2$ is $H_2 = (1/2)[(XR(\lambda X))^{-1}p_\theta^2 + p_\varphi^2]$. On the level $H_2 = 1/2$, time is arc length and we get the quadrature

$$Y^2 = 4(1-X)[X - p_\theta^2(1-\lambda X)], \quad Y = \dot{X}.$$

Since θ is cyclic, p_θ is constant (Clairaut first integral of a surface of revolution). The complex curve is of genus zero and admits a rational parameterization. We get

$$\sin z = \frac{1}{\delta^2 - p_\theta^2}[2\delta^2 X - (\delta^2 + p_\theta^2)], \quad \mathrm{d}t = \frac{\mathrm{d}z}{2\delta},$$

for $z\in\mathbf{R}$ and $\delta^2 = 1 + \lambda p_\theta^2$. We set $\theta_0 = 0$ by symmetry of revolution. We also assume $\varphi_0 = \pi/2$ without loss of generality since time translations generate any extremal on $H_2 = 1/2$ with arbitrary initial condition. The squared adjoint p_θ^2 is bounded by $1/(1-\lambda)$.

Proposition 3. *The system for two bodies and two controls can be integrated using harmonic functions. One has*

$$\sin^2\varphi = \frac{1}{2\delta^2}[(\delta^2 - p_\theta^2)\cos(2\delta t) + (\delta^2 + p_\theta^2)], \quad \delta^2 = 1 + \lambda p_\theta^2, \quad \lambda = 4/5,$$

$$\theta = \mathrm{sign}(p_\theta)\left[\mathrm{atan}\,\frac{(\delta^2 - p_\theta^2) + (\delta^2 + p_\theta^2)\tan(\delta t + \pi/4)}{2\delta p_\theta}\right]_0^t - \lambda p_\theta t.$$

Proof. The quadrature on θ is

$$\frac{\mathrm{d}\theta}{\mathrm{d}z} = \frac{p_\theta}{2\delta}\left(\frac{1}{X} - \lambda\right),$$

whence the result. □

Coordinate φ (*resp.* θ) is periodic (*resp.* quasi-periodic) with period $T = 2\pi/\delta = 2\pi/\sqrt{1+\lambda p_\theta^2}$. (The period of φ is twice the period of $X = \sin^2\varphi$.) The increment of θ over one period is important for the optimality analysis concluding the section. One has

$$\Delta\theta = 2\pi\left(1 - \frac{\lambda p_\theta}{\sqrt{1+\lambda p_\theta^2}}\right). \tag{8}$$

Fix y_0 on $\mathbf{S}^2$. The exponential mapping is defined for $t \in \mathbf{R}$ and $p_0 \in H_2(y_0,\cdot)^{-1}(1/2) \subset T_{y_0}^*\mathbf{S}^2$ by

$$\exp_{y_0} : (t, p_0) \mapsto \Pi \circ \exp t\overrightarrow{H}_2(y_0, p_0) = y(t, y_0, p_0)$$

where $\Pi : T^*\mathbf{S}^2 \to \mathbf{S}^2$ is the canonical projection and $\overrightarrow{H}_2$ the symplectic gradient. A *conjugate point* is a critical value of the exponential mapping. The time associated with such a critical point is the *conjugate time*, and one can define the *first conjugate point* along the geodesic associated with a given p_0. The (first) *conjugate locus* is the set of all such points on geodesics emanating from y_0. Jacobi theorem [13] asserts that, up to the first conjugate point, a geodesic is locally minimizing with respect to neighbouring continuous broken curves with same endpoints.

Theorem 2. *In the two-body two-control case, the conjugate locus of any point on the sphere has four (possibly degenerate) cusps, two horizontal and two meridional.*

Proof. According to [10] result, a sufficient condition is that $\Delta\theta$ is strictly decreasing convex. The condition is valid for (8). □

Finally, define the *cut time* along the geodesic defined by p_0 as the supremum of times t such that the geodesic $s \mapsto \exp_{y_0}(s, p_0)$ is globally minimizing on $[0, t]$. (See [13].) The corresponding point, if any, is the *cut point*. The *cut locus* is the set of all such points on geodesics emanating from y_0. It is known since Poincaré that the cut locus of an analytic metric on the sphere is a finite tree whose extremities are singularities of the conjugate locus. In the case of a metric with more symmetries, the result can be specialized as follows.

Theorem 3 ([25]). *The cut locus of an analytic metric on the sphere of revolution with equatorial symmetry is an antipodal*[3] *subarc provided the Gauss curvature is nondecreasing from North pole to equator.*

[3] Symmetric with respect to the center of the sphere.

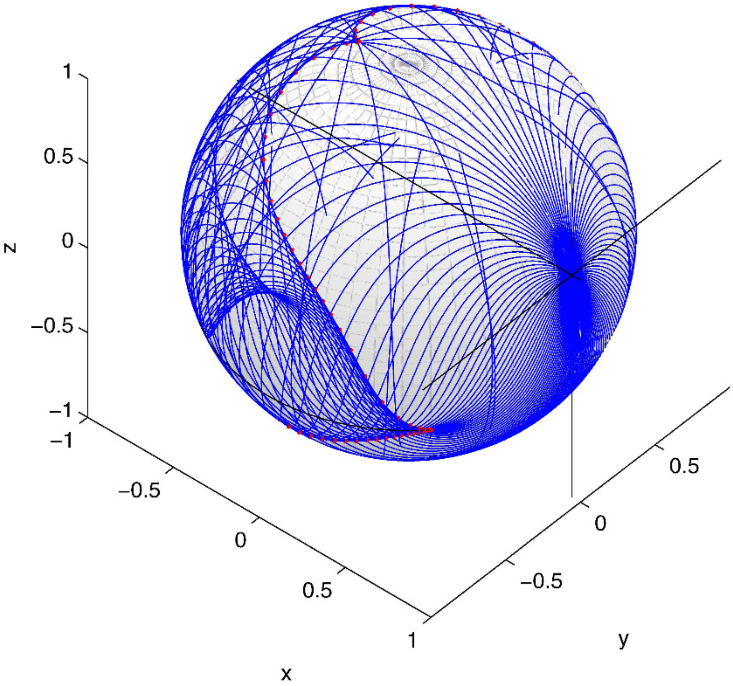


Fig. 1. Conjugate locus, two bodies and two controls. The astroid-shaped locus (in red) is the envelope of geodesics (in blue) emanating from the initial point. It has four (degenerate for initial condition on the poles) cusps, two horizontal and two meridional. The cut locus is a closed antipodal subarc (in black) whose extremities are horizontal cusps of the conjugate locus.

Though metric (7) has the required symmetries, the monotonicity condition on the curvature does not hold as

$$K = \frac{1-\lambda(3-2X)}{(1-\lambda X)^2}$$

is not decreasing when $X \in [0,1]$ (remember that $X = \sin^2\varphi$) for $\lambda = 4/5$. A refined result relying on $\Delta\theta$ being strictly decreasing still gives the result [10].

Theorem 4. *In the two-body two-control case, the cut locus of any point on the sphere is a closed antipodal subarc.*

Varying λ from zero to one in the definition of the metric (7), one connects the canonical metric on the sphere to a metric with an equatorial singularity,

$$\frac{\sin^2\varphi}{1-\sin^2\varphi}\mathrm{d}\theta^2+\mathrm{d}\varphi^2.$$

The original metric is conformal to the standard metric on an oblate ellipsoid of revolution with semi-minor axis $\sqrt{1-\lambda}$ since

$$XR(\lambda X)\mathrm{d}\theta^2+\mathrm{d}\varphi^2=\frac{1}{1-\lambda\sin^2\varphi}[\sin^2\varphi\,\mathrm{d}\theta^2+(1-\lambda\sin^2\varphi)\mathrm{d}\varphi^2].$$

Making λ tend to one can be interpretated as letting the semi-minor axis tend to zero, thus collapsing the sphere on a two-face disk [7]. Such a singularity is intrinsic in the case of only one control as explained in next section.

3 Two bodies, one control

Consider the L^2-minimization of the two-body problem with only one control acting tangentially [8],

$$\ddot{q}+\frac{q}{|q|^3}=u\frac{\dot{q}}{|\dot{q}|},\quad u\in\mathbf{R},\quad \int_0^{t_f}|u|^2\mathrm{d}t\to\min.$$

The state space is as before the trivial fiber space $\mathbf{S}^1\times X$, $X=\mathbf{R}_+^*\times\mathbf{S}^2$, but we correct the relation between φ and the eccentricity,

$$e=\sin\varphi\sqrt{1+\cos^2\varphi}.$$

Changing again time to cumulated longitude,

$$\frac{\mathrm{d}x}{\mathrm{d}l}=uF_1(l,x),\quad \int_0^{l_f}|u|^2\frac{\mathrm{d}l}{\omega(l,x)}\to\min\quad(\text{fixed } l_f).$$

In (n,e,θ) coordinates,

$$F_1=-\frac{3(1-e^2)w}{n^{1/3}(1+e\cos\tau)^2}\frac{\partial}{\partial n}+\frac{2(1-e^2)^2}{n^{4/3}(1+e\cos\tau)^2w}\left[(e+\cos\tau)\frac{\partial}{\partial e}+\frac{\sin\tau}{e}\frac{\partial}{\partial\theta}\right]$$

with true anomaly $\tau=l-\theta$ and $w=\sqrt{1+2e\cos\tau+e^2}$. Since the drift is unchanged, the pulsation is the same (compare (5)),

$$\omega(l,x)=\frac{n(1+e\cos\tau)^2}{(1-e^2)^{3/2}}\cdot$$

The normal maximized Hamiltonian on $\mathbf{S}^1\times T^*X$ is

$$H(l,x,p)=\frac{\omega}{2}H_1^2(l,x,p),$$

where $H_1=\langle p,F_1(l,x)\rangle$. Define the averaged Hamiltonian as in (6). It is remarkable that the averaged flow remains Riemannian.

Proposition 4. *The averaged Hamiltonian is*

$$H(x,p) = \frac{p_r^2}{2} + \frac{c^2}{2r^2}\left[\frac{(1-\sin^2\varphi)^2}{\sin^2\varphi(2-\sin^2\varphi)^2}p_\theta^2 + p_\varphi^2\right]$$

with $r = (2/5)n^{5/6}$ *and* $c = 2/5$.

As in the case of two controls, the flow is Liouville integrable and the whole analysis can be restricted to $\mathbf{S}^2$. The metric induced on the sphere is

$$XR(X)\mathrm{d}\theta^2 + \mathrm{d}\varphi^2, \quad R(X) = \frac{1}{4}\left[1 + \frac{2}{1-X} + \frac{1}{(1-X)^2}\right], \quad X = \sin^2\varphi. \tag{9}$$

There is now an equatorial singularity at $\varphi = \pi/2$. It is an order two pole at $X = 1$ of the rational fraction R. (Compare with $R = 1/(1-X)$ in the previous section.)

Let $H_2 = (1/2)[(XR(X))^{-1}p_\theta^2 + p_\varphi^2]$. On the level $H_2 = 1/2$, the quadrature on φ is

$$Y^2 = 4(1-X)[X(2-X)^2 - 4p_\theta^2(1-X)^2], \quad Y = \dot{X}. \tag{10}$$

The underlying curve is of genus one.[4] It is parameterized by a doubly periodic Weierstraß function,

$$X = 1 - \frac{1}{\wp(z) - 1/3}, \quad \frac{\mathrm{d}t}{\mathrm{d}z} = 1 + \frac{1}{\wp(z)-1/3}, \tag{11}$$

whose invariants reflect the dependence on p_θ,

$$g_2 = \frac{16}{3} + 16p_\theta^2, \quad g_3 = \frac{64}{27} - \frac{16}{3}p_\theta^2. \tag{12}$$

Without loss of generality, we restrict again the computation to $\theta_0 = 0$ and $\varphi_0 = \pi/2$. With the initial condition at singularity, p_θ^2 is unbounded in contrast to the two-control case. Analyzing roots of the degree three polynomial $4\xi^3 - g_2\xi - g_3$ associated with Weierstraß function, one sees that the parameterization has to be restricted to the unbounded component of the cubic to ensure $X \in [0,1]$. Hence z belongs to $\mathbf{R}$.

Proposition 5. *The transcendence for two bodies and one (tangential) control is elliptic. One has*

$$\sin^2\varphi = \frac{\wp(z) - 4/3}{\wp(z) - 1/3}, \quad z \in \mathbf{R},$$

$$t = \frac{1}{\wp'(a)}\left[\ln\frac{\sigma(z-a)}{\sigma(z+a)}\right]_0^z + \left(1 + \frac{2\zeta(a)}{\wp'(a)}\right)z,$$

$$\theta = 2p_\theta\left[\frac{1}{\wp'(b)}\ln\frac{\sigma(z-b)}{\sigma(z+b)} - \frac{1}{\wp'(c)}\ln\frac{\sigma(z-c)}{\sigma(z+c)}\right]_0^z + 4p_\theta\left(\frac{\zeta(b)}{\wp'(b)} - \frac{\zeta(c)}{\wp'(c)}\right)z,$$

with $\wp(a) = 1/3$, $\wp(b) = 4/3$, $\wp(c) = -2/3$, *and invariants (12).*

[4] Excluding the degenerate case $p_\theta = 0$ associated with meridians.

Proof. The quadrature on θ is

$$\frac{\mathrm{d}\theta}{\mathrm{d}z} = 2p_\theta \left(\frac{1}{\wp(z) - 4/3} - \frac{1}{\wp(z) + 2/3} \right).$$

It is similar to the quadrature (11) on t. Introducing Weierstraß ζ and σ functions, $\wp = -\zeta'$, $\zeta = \sigma'/\sigma$, one has

$$\int \frac{\wp'(a)\mathrm{d}z}{\wp(z) - \wp(a)} = 2\zeta(a)z + \ln \frac{\sigma(z-a)}{\sigma(z+a)},$$

whence the result. □

The family of genus one complex curves (10) are all homeomorphic to the torus. The topological classification of extremals is then trivial. We recall standard facts on the moduli space of elliptic curves [18] so as to refine the classification up to conformal equivalence.

Let L be a lattice in the complex plane with basis (l_1, l_2) (complex numbers linearly independent over $\mathbf{R}^2$). A pair (l_1', l_2') defines another basis if only if

$$l_1' = al_1 + bl_2,$$
$$l_2' = cl_1 + dl_2,$$

for some matrix

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \in \mathrm{SL}(2, \mathbf{Z}).$$

Two tori $\mathbf{C}/L$, $\mathbf{C}/L'$ are conformal if and only if there is some $\mu \in \mathbf{C}^*$ such that $L' = \mu L$. Let (l_1, l_2) and (l_1', l_2') be bases of L and L', respectively. We can assume that $\tau = l_2/l_1$ and $\tau' = l_2'/l_1'$ belong to Poincaré upper half-plane, $\mathbf{H}$. From the previous remarks, L and L' are conformal if and only if there is a quadruple (a, b, c, d) such that

$$\tau' = \frac{a\tau + b}{c\tau + d}, \quad a, b, c, d \in \mathbf{Z}, \quad ad - bc = 1. \tag{13}$$

Such particular *Möbius transformations* are automorphisms of $\mathbf{H}$. The induced group morphism between $\mathrm{SL}(2, \mathbf{Z})$ and $\mathrm{Aut}(\mathbf{H})$ has kernel ±id. Transformations (13) are then identified with the Fuchsian *modular group* $\mathrm{PSL}(2, \mathbf{Z}) = \mathrm{SL}(2, \mathbf{Z})/\pm\,\mathrm{id}$. Then $\mathbf{H}/\mathrm{PSL}(2, \mathbf{Z})$ is the moduli space for congruences of conformal tori. One eventually defines the *modular function* [18]

$$j(\tau) = \frac{g_2^3}{\Delta}.$$

It is a bijection from $\mathbf{H}/\mathrm{PSL}(2, \mathbf{Z})$ onto $\mathbf{C}$.

In our case, to each p_θ is associated a real rectangular lattice. Using (12), one can define

$$j(p_\theta) = \frac{16(1 + 3p_\theta^2)^3}{27p_\theta^2(8 + 13p_\theta^2 + 16p_\theta^4)} \tag{14}$$

and obtain the following classification of extremals.

Proposition 6. *There are not more than three conformal φ-curves.*

Proof. Function (14) has exactly two critical points, so $j(p_\theta) =$ constant has at most three distinct solutions (not taking into account symmetric solutions, $\pm p_\theta$). □

To estimate the conjugate locus at singularity, we use the following local model. Set $x = \pi/2 - \varphi$, $y = \theta$. The metric (9) is locally approximated by

$$\mathrm{d}x^2 + \frac{\mathrm{d}y^2}{x^{2p}}\cdot \tag{15}$$

In the case of one (tangential) control, $p = 2$.

Proposition 7 ([7]). *The conjugate locus at the origin of (15) is $y = \pm C_p x^{p+1}$ minus the origin itself. As $p \to \infty$, $C_p \sim 8/(3p+1)$.*

As a result, the conjugate locus of the metric on $\mathbf{S}^2$ has an order 3 contact with the equatorial singularity. Because of the symmetry $p_\theta \to -p_\theta$, this defines two meridional cusps of the conjugate locus at $\varphi_0 = \pi/2$. (See Fig. 2.) The result of section 2 can be extended to this singular setting.

Theorem 5 ([7]). *If $\varDelta\theta$ is strictly decreasing convex, the conjugate locus has four (possibly degenerate) cusps, all meridional for equatorial points, two horizontal and two meridional otherwise.*

The verification on $\varDelta\theta$ is intricate but can again be made. The following estimates are computed in [7],

$$\varDelta\theta \sim_0 2\pi\Big(1 - \frac{3\sqrt{2}}{4}p_\theta + \frac{35\sqrt{2}}{128}p_\theta^3\Big), \quad \varDelta\theta \sim_\infty \frac{4}{3}(2-\sqrt{2})K(3-2\sqrt{2})p_\theta^{-3/2},$$

where $K(k)$ is the complete Jacobi integral of first kind and modulus k. The previous structure result on the cut locus is also extended to include the two-body one-control case.

Theorem 6 ([7]). *If $\varDelta\theta$ is strictly decreasing, the cut locus of a point on the sphere is the equator minus the point itself for equatorial points, a closed antipodal subarc otherwise.*

4 Three bodies, two controls

In contrast with sections 2 and 3, we keep the original constraint on the control, and consider the final time minimization of

$$\ddot{q} + \nabla V_\mu(q) + 2iq = \varepsilon u, \quad |u| \le 1.$$

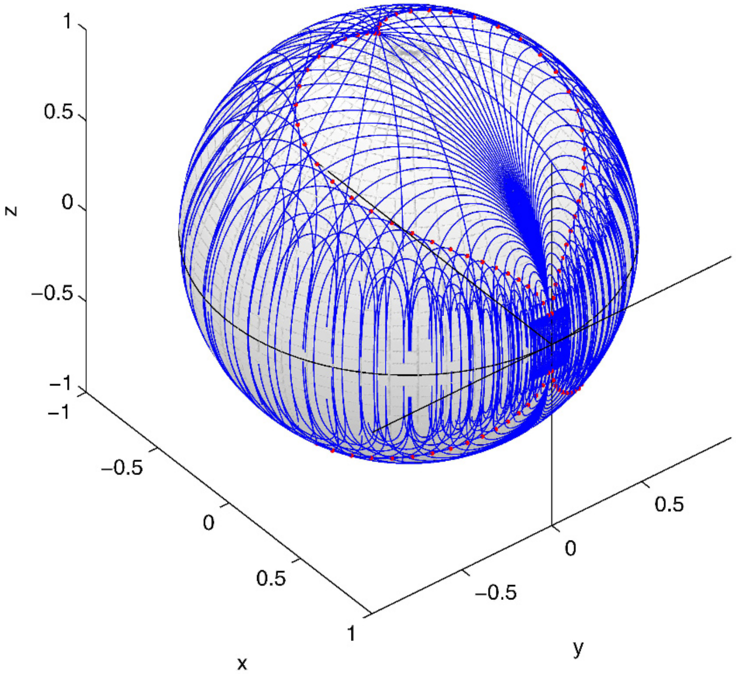


Fig. 2. Conjugate locus, two bodies and one (tangential) control. The double-heart locus (in red) is the envelope of geodesics (in blue) emanating from the initial point. It has four meridional cusps (two of them generated by order 3 contacts at origin). The cut locus (in black) is the whole equator minus the origin.

See [9] for preliminary computations on the L^2-relaxation of the problem. Available results on controlled three-body problems are mostly numerical. They usually deal with refined models taking into account three-dimensional effects, perturbations, and rely on direct optimization methods. (See, *e.g.*, [3].)

The position vector q belongs to the complex plane with two punctures at $-\mu$ and $1-\mu$, denoted Q_μ. The state space X_μ is the tangent space TQ_μ in (rotating) cartesian coordinates $(q, \dot{q})$. It is the cotangent space T^*Q_μ in (q, p) variables, see (1). In both cases, $X_\mu \simeq Q_\mu \times \mathbf{R}^2$ is a trivial bundle. In cartesian coordinates,

$$\dot{x} = F_0(x) + \varepsilon(u_1 F_1(x) + u_2 F_2(x)), \quad |u| \leq 1.$$

There,

$$F_0(x) = \dot{q}\frac{\partial}{\partial q} - (\nabla V_\mu(q) + 2iq)\frac{\partial}{\partial \dot{q}}, \quad F_1(x) = \frac{\partial}{\partial \dot{q}_1}, \quad F_2(x) = \frac{\partial}{\partial \dot{q}_2}\cdot$$

The maximized normal Hamiltonian[5] is

$$H = -1 + H_0 + \varepsilon\sqrt{H_1^2 + H_2^2}, \quad H_i = \langle p, F_i(x)\rangle, \quad i = 0, \ldots, 2.$$

Extremals are classified according to the order of their contact with the switching surface $\Sigma = \{H_1 = H_2 = 0\}$. (See [12].)

Proposition 8. *Contacts with Σ are of order one and define isolated π-singularities.*[6]

Proof. The distribution $\{F_1, F_2\}$ being involutive, the switching function $\psi = (H_1, H_2)$ is $\mathscr{C}^1$,

$$\dot{\psi}_1 = \{H_0, H_1\} - u_1\{H_1, H_2\}, \quad \dot{\psi}_2 = \{H_0, H_2\} + u_2\{H_1, H_2\}.$$

The bracket $\{H_1, H_2\}$ vanishes on Σ. The drift comes from a second order mechanical system, so $\{F_1, F_2, [F_0, F_1], [F_0, F_2]\}$ has full rank. Then $\dot{\psi} \neq 0$ on Σ and contacts are of order one. By Pontryagin maximization condition, $u = \psi/|\psi|$, so u is changed to $-u$ whenever ψ vanishes. □

As Σ is of codimension two in T^*X_μ, we can neglect these finitely many π-singularities for the numerical computation and restrict to smooth extremals not crossing the switching surface.

For the minimum time problem, the exponential mapping associated with order zero extremals is defined on a neighbourhood of the origin in $\mathbf{R} \times H(x_0, \cdot)^{-1}(0)$,

$$\exp_{x_0} : (t, p_0) \mapsto \Pi \circ \exp t\overrightarrow{H}(x_0, p_0) = x(t, x_0, p_0), \quad \Pi : T^*X_\mu \to X_\mu.$$

Given a target x_f, the problem is to find a zero of the shooting equation

$$\exp_{x_0}(t_f, p_0) = x_f.$$

The two-body problem is embedded into the three-body one thanks to parameter μ. This paves the way for using continuation methods between two and three-body control problems. (See also [15] for such an approach in the two-body case.) Rather than information on the adjoint, the knowledge of the Kepler minimum time from [12] turns out to be critical to initialize the continuation. Our target for numerical computation is an equilibrium point of the uncontrolled problem or *Lagrange point* [26]. Such points where the influences of the two primaries compensate each other are appropriate targets for the continuation. Here we use the L_2 Lagrange point. It is equal to

[5] From now on, p denotes the adjoint to x.

[6] Instantaneous rotations of angle π of the control.

the second primary when $\mu = 0$. Lagrange points are extensively studied in celestial mechanics and mission design [19]. A second continuation on ε is also used to reach low bounds on the control, see results in Fig. 3. Hill regions are projections on the q-space of level sets of the Jacobi integral. In the controlled case, they vary dynamically along the trajectory (see Fig. 4),

$$R_\mu(t) = \{\xi \in Q_\mu \mid J_\mu(q(t), \dot{q}(t)) - V_\mu(\xi) \geq 0\}.$$

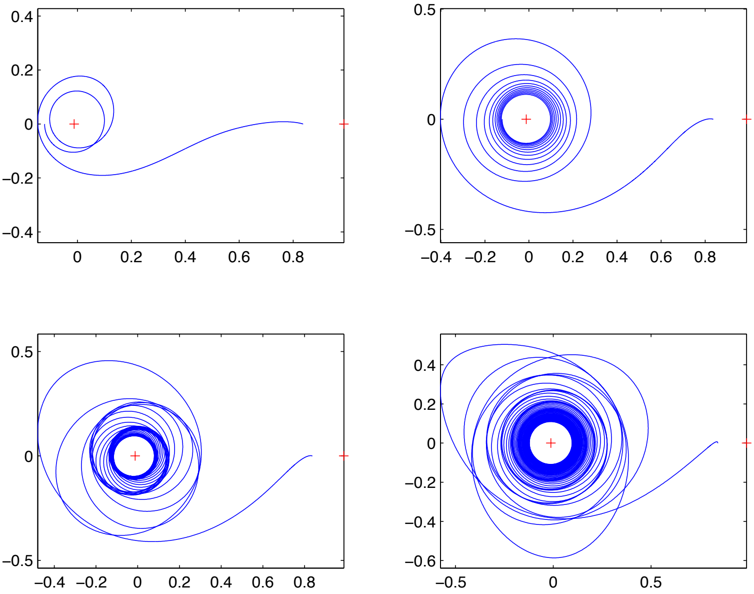


Fig. 3. Three bodies, two controls. Minimum time trajectories from the geostationary orbit to Lagrange L_2 point in the Earth-Moon system ($\mu \simeq 0.0121$). Successively, $\varepsilon = 2.440$, 0.2440, 0.1220 and 0.04148.

A normal extremal is *regular* if it verifies the strong Legendre condition that there exists some positive α such that, everywhere on $[0, t_f]$, $\partial^2 H/\partial u^2 \leq -\alpha I$ along the extremal.

Lemma 1. *Order zero extremals of the minimum time three-body problem with two controls are regular.*

Proof. Along an order zero extremal, $u \in \mathbf{S}^1$. In any chart,

$$\frac{\partial^2 H}{\partial u^2} = -\varepsilon\sqrt{H_1^2 + H_2^2} = -\varepsilon|\psi|.$$

The absence of π-singularity implies the strong Legendre condition as $|\psi|$ is then smooth and bounded below by some positive constant. □

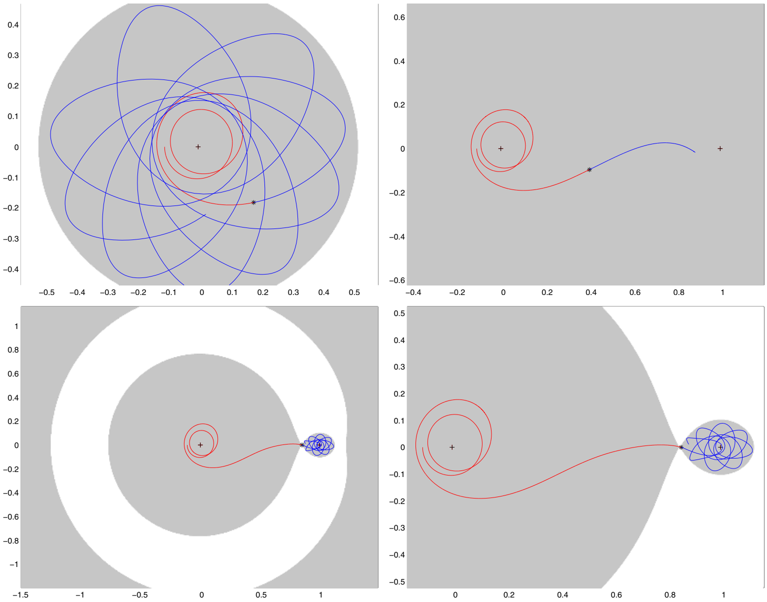


Fig. 4. Dynamics of the Hill regions, $\varepsilon = 2.440$. The controlled trajectory (in red) is plotted up to three different times and prolongated by the osculating uncontrolled trajectory (in blue). During a first phase, energy J_μ is increased so as to include the L_2 target. The second phase is close to the motion of the system towards projection in the q-space of the Lagrange point. The last two graphs are identical (the rightmost one has a finer scale) and illustrate instability of the L_2 point after the target is reached.

As in the Riemannian case (without singularities), regular extremals are locally time minimizing for short times [1]. To investigate further local optimality, one generalizes Jacobi theory to the optimal control setting. Define again conjugate points as critical values of the exponential mapping. The following technical condition is sufficient to avoid degenerate situations on the kernel of the second variation of the problem (see [24]). Let

$$E_{x_0} : (t_f, u) \mapsto x(t_f, x_0, u)$$

be the *endpoint mapping*. It is defined on a neighbourhood of the reference pair (t_f, u) in $\mathbf{R} \times \mathrm{L}^\infty([0, t_f], \mathbf{S}^1)$. We assume that, for any subinterval $[t_1, t_2]$ of $[0, t_f]$, the partial derivative $\partial E_{x(t_1)}/\partial u(t_2 - t_1, u|[t_1, t_2])$ has corank one.

Theorem 7 ([1, 24]). *Under the corank one assumption, the trajectory associated with a regular extremal is* $\mathscr{C}^0$*-locally time minimizing up to the first conjugate point. Past this point, the control is not even* L^∞*-locally minimizing.*

Local optimality of every extremal is verified by a conjugate point test. (See Fig. 5). The practical computation of conjugate points is done by rank evaluation on Jacobi fields [11]. The concept of conjugate point is extended by the notion of *focal point* [Ibid.] to encompass the case of submanifold targets. Such an example for a lunar orbit target is provided Fig. 6.

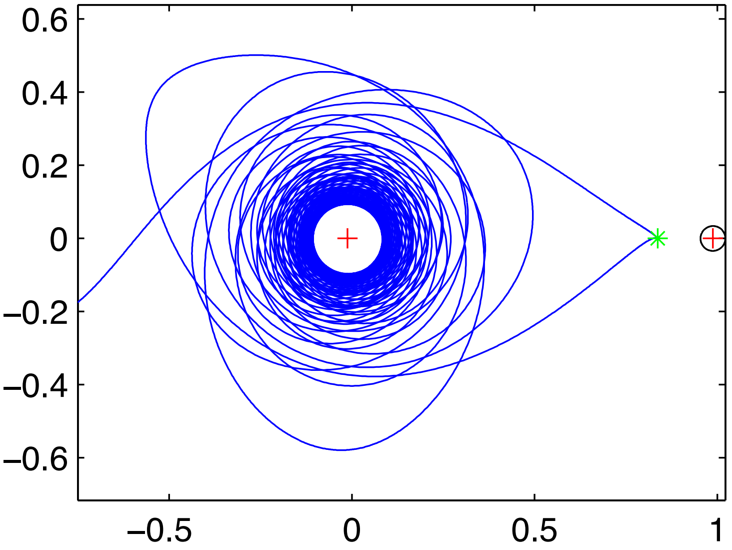


Fig. 5. Conjugate point computation, $\varepsilon = 0.04148$. The reference trajectory is prolongated up to the first conjugate point, beyond the L_2 target. Local optimality up to the target is guaranteed. The cuspidal point of first kind observed is generically due to the condition $\dot{q}_f = 0$.

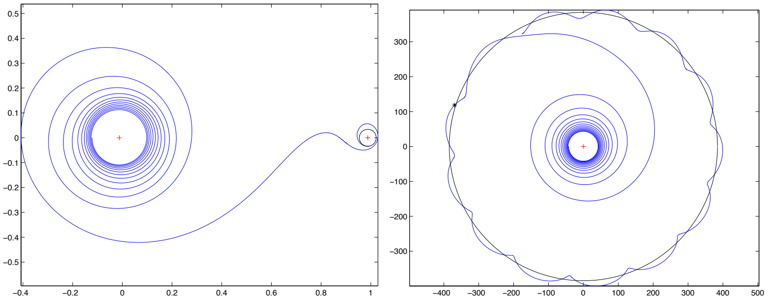


Fig. 6. Focal point computation, $\varepsilon = 0.2440$. The target is a lunar orbit, and the focal point test ensures local optimality of the trajectory. The leftmost frame is the rotating frame, the rightmost one is fixed.

Whatever the target, the value function $\varepsilon \mapsto t_f(\varepsilon)$ of the minimum time problem is decreasing: The smaller ε, the larger the transfer time. This is contradicted by results portrayed Fig. 7. We infer that the first extremal is locally but not globally minimizing. When decreasing the bound on the control ε from 0.2221 to 0.2196, one revolution around the first primary has to be added before escape towards the second body is obtained. There lies global analysis of the problem, in the interplay between the two small parameters μ, ε. This leaves open the question of global optimality.

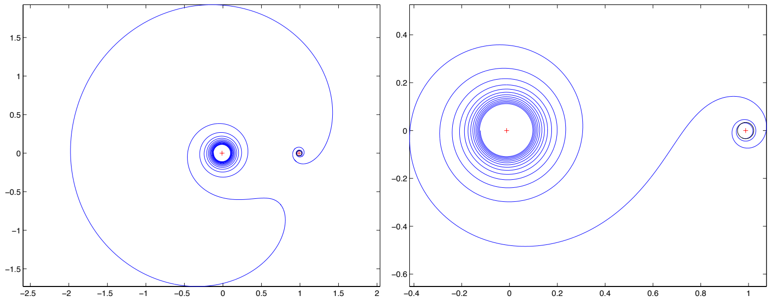


Fig. 7. Lunar target, $\varepsilon = 0.2221$ and 0.2196. Focal point tests ensure local optimality in both cases. However, $t_f \simeq 17.8$ versus $t_f \simeq 10.8$ in the second one. The first extremal is a local but not a global minimizer. The difference in strategies is apparent as one extra revolution around the Earth is added in the second case before reaching the lunar orbit target.

References

1. Agrachev, A. A.; Sachkov, Y. L. *Control Theory from the Geometric Viewpoint.* Springer, 2004.
2. Belbruno, E. A. Two-body motion under the inverse square central force and equivalent geodesic flows. *Celest. Mech.* **15** (1977), no. 4, 467-476.
3. Betts, J. T.; Erb, S. O. Optimal Low Thrust Trajectories to the Moon. *SIAM J. Appl. Dyn. Syst.* **2** (2003), no. 2, 144–170.
4. Bombrun, A.; Chetboun, J.; Pomet, J.-B. Transfert Terre-Lune en poussée faible par contrôle feedback. La mission SMART-1. *INRIA Research report* (2006), no. 5955, 1–27.
5. Bonnard, B.; Caillau, J.-B. Riemannian metric of the averaged energy minimization problem in orbital transfer with low thrust. *Ann. Inst. H. Poincaré Anal. Non Linéaire* **24** (2007), no. 3, 395–411.
6. Bonnard, B.; Caillau, J.-B. Geodesic flow of the averaged controlled Kepler equation. *Forum math.* **21** (2009), no. 5, 797–814.
7. Bonnard, B; Caillau, J.-B. Singular metrics on the two-sphere in space mechanics. *HAL preprint* (2008), no. 00319299, 1–25.
8. Bonnard, B.; Caillau, J.-B.; Dujol, R. Energy minimization of single-input orbit transfer by averaging and continuation. *Bull. Sci. Math.* **130** (2006), no. 8, 707–719.

9. Bonnard, B.; Caillau, J.-B.; Picot, G. Geometric and numerical techniques in optimal control of the two and three-body problems. *HAL preprint* (2010), no. 00432631, 1–39.
10. Bonnard, B.; Caillau, J.-B.; Sinclair, R.; Tanaka, M. Conjugate and cut loci of a two-sphere of revolution with application to optimal control. *Ann. Inst. H. Poincaré Anal. Non Linéaire* **26** (2009), no. 4, 1081–1098.
11. Bonnard, B.; Caillau, J.-B.; Trélat, E. Second order optimality conditions in the smooth case and applications in optimal control. *ESAIM Control Optim. and Calc. Var.* **13** (2007), no. 2, 207–236. (`apo.enseeiht.fr/cotcot`.)
12. Caillau, J.-B.; Noailles, J. Coplanar control of a satellite around the Earth. *ESAIM Control Optim. and Calc. Var.* **6** (2001), 239–258.
13. Do Carmo, M. P. *Riemannian geometry.* Birkhuser, Boston, 1992.
14. Edelbaum, T. N. Optimal low-thrust rendez-vous and station keeping. *AIAA J.* **2** (1964), no. 7, 1196–1201.
15. Gergaud, J.; Haberkorn, T. Homotopy method for minimum consumption orbit transfer problem. *ESAIM Control Optim. Calc. Var.* **12** (2006), no. 2, 294–310.
16. Guckenheimer, J.; Holmes, P. *Nonlinear oscillations, dynamical systems and bifurcations of vector fields.* Springer, 1993.
17. Hatcher, A. *Algebraic topology.* Cambridge University Press, 2002.
18. Jones, G. A.; Singerman, D. *Complex Functions. An Algebraic and Geometric Viewpoint.* Cambridge University Press, 1987.
19. Marsden, J. E.; Ross, S. D. New methods in celestial mechanics and mission design. *Bull. Amer. Math. Soc. (N.S.)* **43** (2006), no. 1, 43–73.
20. McCord, C.; Meyer, K. R.; Offin, D. Are Hamiltonian Flows Geodesic Flows? *Trans. Amer. Math. Soc.* **355** (2003), no. 3, 1237-1250.
21. Moser, J. K. Regularization of Kepler's problem and the averaging method on a manifold. *Comm. Pure Appl. Math.* **23** (1970), 609-635.
22. Osipov, Y. The Kepler problem and geodesic flows in spaces of constant curvature. *Celestial Mech.* **16** (1977), no. 2, 191-208.
23. Racca, G.; *et al.* SMART-1 mission description and development status. *Planetary and space science* **50** (2002), 1323–1337.
24. Sarychev, V. A. The index of second variation of a control system. *Mat. Sb.* **41** (1982), 338–401.
25. Sinclair, R.; Tanaka, M. The cut locus of a two-sphere of revolution and Toponogov's comparison theorem. *Tohoku Math. J.* **59** (2007), no. 2, 379–399.
26. Szebehely, V. *Theory of orbits: The restricted problem of three bodies.* Academic Press, 1967.

Optimization techniques for the computation of the effective Hamiltonian

Maurizio Falcone and Marco Rorro

[1] Maurizio Falcone, Dipartimento di Matematica, SAPIENZA - Università di Roma, falcone@mat.uniroma1.it

[2] Marco Rorro, CASPUR, Roma, rorro@caspur.it

Summary. In this paper we examine the numerical efficiency and effectiveness of some algorithms proposed for the computation of the effective Hamiltonian, a classical problem arising e.g. in weak KAM theory and homogenization. In particular, we will focus our attention on the performances of an algorithm of direct constrained minimization based on the SPG (Spectral Projected Gradient) algorithm proposed in [3, 4]. We will apply this method to the minimization of a functional proposed by C. Evans in [9] and we will compare the results with other methods.

1 Introduction

The approximation of the effective Hamiltonian is a challenging problem with a strong impact on many applications e.g. to the study of dynamical systems, weak KAM theory, homogenization, optimal mass transfer problems. For example, in homogenization theory one has to study the properties of a composite material with a periodic structure depending on a parameter ε in order to guess the physical properties of the material obtained in the limit for ε going to 0. In order to give a hint on the problem, let us consider the following initial value problem:

$$\begin{cases} u_t^\varepsilon + H(\frac{x}{\varepsilon}, Du^\varepsilon) = 0 & \text{in } \mathbb{T}^N \times]0, +\infty[\\ u(x,0) = u_0 & \text{in } \mathbb{T}^N \times \{0\} \end{cases} \tag{1}$$

where $\mathbb{T}^N$ is the unit flat torus, $u : \mathbb{T}^N \times (0,T) \to \mathbb{R}$, $Du : \mathbb{R}^N \to \mathbb{R}^N$ is its gradient and the Hamiltonian $H : \mathbb{T}^N \times \mathbb{R}^N \to \mathbb{R}$ satisfies the assumptions :

$$H \text{ is Lipschitz continous on } \mathbb{T}^N \times B(0,R), \tag{2}$$

$$\lim_{|p|\to+\infty} H(x,p) = +\infty \tag{3}$$

$$|H(x,p) - H(y,p)| \leq C\left[|x-y|(1+|p|)\right]. \tag{4}$$

We are interested in the limiting behaviour of the solution $u^\varepsilon(x,t)$ as ε goes to 0. It is known that,

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_19, © Springer-Verlag Berlin Heidelberg 2010

$$\lim_{\varepsilon \to 0} u^\varepsilon(x,t) = \overline{u}(x,t) \tag{5}$$

uniformly on compact sets where $\overline{u}$ is the solution of a new evolutive problem

$$\begin{cases} u_t + \overline{H}(Du) = 0 & \text{in } \mathbb{T}^N \times]0,+\infty[\\ u(x,0) = u_0(x) & \text{in } \mathbb{T}^N \times \{0\} \end{cases} \tag{6}$$

In order to know the limiting behavior of u^ε one could solve (6), but this can not be done without knowing the *effective Hamiltonian* $\overline{H}$ which just depends on Du. As we will see in the sequel, this computation is a very hard task even in low dimension and our goal here is to analyze and compare the performances of three methods on some typical benchmarks. Several methods have been proposed in the literature: some of them are based on the solution of nonlinear Hamilton-Jacobi equations, others use a variational formulation or discretize directly a representation formula for the solution based on a min-max operator.

In some previous papers [17, 18, 1] the computation of the effective Hamiltonian in low dimension has been obtained solving the so called *cell problem*

$$H(x, Du + P) = \lambda \text{ on } \mathbb{T}^N \tag{7}$$

where $P \in \mathbb{R}^N$ is a fixed vector and the unknown is the pair (u, λ), where λ is a scalar representing the value of the effective Hamiltonian at P. Then, in order to solve (7), one has to compute the solution u and the value $\lambda(P) = \overline{H}(P)$ for every vector P. Note that the problem was introduced circa in 1988 in [16] but the corresponding numerical methods were proposed only in the last decade due to the increase of computer power. Let us assume that H satisfies (2)–(4) and that it is convex in the second variable. It is well known (see [16], [8] for details) that for each fixed $P \in \mathbb{R}^N$ there exists a unique real number λ such that (7) has a periodic Lipschitz continuous viscosity solution. Since the effective Hamiltonian verifies the following identity

$$\overline{H}(P) = \inf_{u \in C^1(\mathbb{T}^N)} \sup_{x \in \mathbb{T}^N} H(x, Du + P) \tag{8}$$

usually indicated as the *min-max formula* (see [6] and [14]) one can think that the above characterization can lead to an algorithm. In fact, a direct discretization of (8) has been proposed in [14] and some examples have been computed using that formula. The main idea in that approach is to discretize $C^1(\mathbb{T}^N)$ by piecewise linear functions (the P_1 approximation of finite elements) and then apply a min-max search to the discrete formula using MATLAB. Here we try to improve the performances of the above method using the `FFSQP` library [19] which has been conceived to solve nonlinear min-max problems. Although this code has a better performance with respect to the MATLAB `minimax` function used in [14], the method is still too expensive in terms of *CPU* time and seems to be inadequate to compute $\overline{H}(P)$ with a reasonable accuracy (see the last section for details). This experience has motivated further efforts to find new ways to compute the effective Hamiltonian. We note that one of the main difficulties in both problems (7) and (8) is that, even if the value of $\overline{H}(P)$ is unique for each fixed P, the solution of (7) or the minimizer of (8) are in general not unique. In [18] and [17] different regularizations of (7) are considered (see [5] for some a priori estimates).

In this paper we follow a variational approach based on a regularization of the min-max formula (8) that was proposed in [9]. This approach, summarized in Section 2, is in principle less accurate than the direct discretization of the min-max formula since we replaced the original problem by a regularized problem introducing an additional error. However, as we will see in the sequel, this approach can be simpler and more efficient from a computational point of view. In [13] we have solved this problem deriving the Euler-Lagrange equation and finding a finite difference approximation of it. Here we try to solve it by a direct minimization via the SPG methods proposed by [3, 4]. To this end we will need to find an appropriate choice of the various parameters appearing in the algorithm. In the last section we solve some typical benchmarks where the exact solutions are known so that we can compare the effectiveness of the results obtained by different approaches.

2 A variational approximation

As we said in the introduction, our starting point is the approximation of the effective Hamiltonian proposed by Evans in [9]. This is defined by

$$\overline{H}_k(P) \equiv \frac{1}{k} \log \left(\int_{\mathbb{T}^n} e^{kH(x,Du_k+P)} \mathrm{d}x \right), \tag{9}$$

where $k \in \mathbb{N}$ and $u_k \in C^1(\mathbb{T}^N)$ is a minimizer of the functional

$$I_k[u_k] = \int_{\mathbb{T}^n} e^{kH(x,Du_k+P)} \mathrm{d}x \tag{10}$$

satisfying the normalization constraint

$$\int_{\mathbb{T}^N} u_k \mathrm{d}x = 0. \tag{11}$$

(this constraint is added in order to select a unique solution up to a constant). This approach is effective due to the following result.

Theorem 1 ([9]). *Assume that $H(x,p)$ is strictly convex in p. Then,*

$$\overline{H}(P) = \lim_{k\to+\infty} \overline{H}_k(P). \tag{12}$$

Moreover, the above approximation leads to the following estimates:

$$\overline{H}_k(P) \leq \overline{H}(P) \leq \overline{H}_k(P) + C\frac{\log k}{k} \tag{13}$$

for any $k \in \mathbb{N}$.

The Euler-Lagrange approximation

In [13] we have solved that problem via the Euler–Lagrange equation

$$\operatorname{div}\left(e^{kH(x,Dv_k)} D_p H(x, Dv_k)\right) = 0. \tag{14}$$

We first compute its solution $v_k = u_k + Px$ via a finite difference scheme and then we derive u_k from that expression.
For simplicity, let us fix the dimension to $N = 1$ and assume that the grid G is a standard lattice $G \equiv \{x_i : x_i = i\Delta x, i = 1, \ldots, n\}$. Using a standard second order finite difference approximation for v_x and a central approximation for v_{xx} we end up with a sparse nonlinear system of n equations in the n unknown $v_1, \ldots, v_n$. Since the term v_i is contained only in the discretization of the second derivative, it is easier to solve the i-th equation with respect to v_i, $v_i = F_i(v_{i+1}, v_{i-1})$, and obtain the numerical solution by the iterative scheme

$$v_i^{m+1} = F_i(v_{i+1}^m, v_{i-1}^m) \quad \text{for } i = 1, \ldots, n. \tag{15}$$

with boundary conditions $v_{n+1} = v_1 + P$ which correspond to the fact that u has to be periodic. Once a minimizer is obtained, we compute $\overline{H}_k(P)$ renormalizing formula (9) by adding and subtracting the constant $\overline{C} \equiv \max_{x \in \mathbb{T}^N} H(x, Dv_k)$ to obtain

$$\overline{H}_k(P) = \overline{C} + \frac{1}{k} \log \left(\int_{\mathbb{T}^N} e^{k\left(H(x,Dv_k) - \max\limits_{x \in \mathbb{T}^N} H(x,Dv_k)\right)} dx \right). \tag{16}$$

The SPG method

We want to construct a direct discretization of the functional I_k on the space of the piecewise linear function. Let us observe that the functional can be minimized with respect to the derivatives $c_i = \partial u(x_i)/\partial x_i$ instead of the values of u_i using standard finite difference approximation. Note that by the derivatives c_i we can get back to the values u_i by integration. Moreover, on a standard lattice G in dimension 1, the periodicity constraint has a simple translation in the new variables which correspond to a piecewise linear approximation,

$$\sum_{i=0}^{n} c_i = 0. \tag{17}$$

Since the constraint is an hyperplane in dimension n, we can apply a Projected Gradient (PG) method to solve it. Although the standard PG method is simple to code, it is rather slow due the fact that in order to maintain feasibility of the iterates it is necessary to project several times and recompute the optimal step. The projection is in general the most expensive part of the method even when the projection on the convex set of constraints is rather simple.

New methods have been proposed to overcome this difficulty and to define efficient techniques for the control of the step-size, see e.g. the survey by Dunn [7]. For our problem, we have used the Spectral Projected Gradient (SPG) method proposed by Birgin, Martinez and Rayan [3], see also [4]. This method combines two techniques to improve the performances of the PG method. The first is the non-monotone line search scheme developed by Grippo, Lampariello and Lucidi [15] for Newton's method. The second is based on the spectral step-length proposed by Barzilai and Borwein [2].
Let us sketch the basic steps of the SPG method for the minimization of a function $f : \mathbb{R}^n \to \mathbb{R}$ over a closed convex set $\Omega \subset \mathbb{R}^n$. We will assume that $f \in C^1(\mathbb{R}^n)$ and we will denote by $P(z)$ the orthogonal projection on Ω of a point z. The algorithm starts with a point x_0 and uses an integer $M \geq 1$, two real parameters α_{min} and α_{max} which allow to control the step-length, a sufficient decrease parameter $\gamma \in (0, 1)$, and two additional safeguarding parameters $0 < \sigma_1 < \sigma_2 < 1$. The initial choice of $\alpha_0 \in [\alpha_{min}, \alpha_{max}]$ is arbitrary and the algorithm below describes how to compute the sequences $\{x_k\}$ and $\{\alpha_k\}$ and when to stop. The notations $\|\cdot\|$ and $\langle\cdot,\cdot\rangle$ indicate respectively the Euclidean norm and the scalar product in $\mathbb{R}^n$.

The SPG Algorithm
Step 1. Detect if the current point x_k is stationary
if $||P(x_k - \nabla f(x_k)) - x_k|| = 0$, STOP and declare that the solution has been found.
Step 2 *Backtracking*
Step 2.1 Compute $d_k = P(x_k - \alpha_k \nabla f(x_k)) - x_k$. Set $\lambda \leftarrow 1$.
Step 2.2 Set $x_+ = x_k + \lambda d_k$.
Step 2.3 If

$$f(x_+) \leq \max_{0 \leq j \leq \min\{k, M-1\}} f(x_k - j) + \gamma\lambda\langle d_k, \nabla f(x_k)\rangle, \tag{18}$$

then define $\lambda_k = \lambda$, $x_{k+1} = x_+$, $s_k = x_{k+1} - x_k$, $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$, and go to Step 3. If (18) does not hold, define

$$\lambda_{new} \in [\sigma_1\lambda, \sigma_2\lambda], \tag{19}$$

set $\lambda \leftarrow \lambda_{new}$ and go to Step 2.2.
Step 3
Compute $b_k = \langle s_k, y_k\rangle$.
If $b_k \leq 0$, set $\alpha_{k+1} = \alpha_{max}$;
else, compute $a_k = \langle s_k, s_k\rangle$and set $\alpha_{k+1} = \min\{\alpha_{max}, \max\{\alpha_{min}, a_k/b_k\}\}$.

Note that Step 2 is the main modification with respect to the standard *PG* method. In the SPG method the condition (18) is computed along a set of trial points of the form $x_k + \lambda_k d_k$ which, in general are not aligned. The path connecting the trial points is a curvilinear path. It has been proved that this path is piecewise linear if the set of constraints Ω is a polyhedral set. A convergence result which only requires the regularity for f and the convexity of Ω can be found in [3]. The interested reader will also find there more details on the method and a detailed comparison with other constrained optimization methods.

The min-max discrete approximation

Let us briefly recall the results related to this approach. We denote by

$$\overline{H}^{\Delta x}(P) = \inf_{u\in W^1} \operatorname*{ess\,sup}_{x\in\mathbb{T}^N} H(x, Du + P)$$

where $W^1 \equiv \{w : \mathbb{T}^N \to \mathbb{R} : w \in C(\mathbb{T}^N) \text{ and } Dw(x) = c_j, \forall\, x \in T_j, \forall\, j\}$, T_j is a family of simplices such that $\mathbb{T}^N = \bigcup_j T_j$ and $\Delta x \equiv \max_j \operatorname{diam}(T_j)$.

Proposition 1 ([14]). *Assume that $H(x,p)$ is convex in p. Then $\overline{H}^{\Delta x}(P)$ is convex,*

$$\overline{H}(P) = \lim_{\Delta x\to 0} \overline{H}^{\Delta x}(P) \tag{20}$$

and

$$\overline{H}(P) \le \overline{H}^{\Delta x}(P) \tag{21}$$

It is interesting to note that some a priori error estimates are also available for this approximation. More precisely, when u is Lipschitz continuous (which is the case when $H(x,p)$ is strictly convex in p), we have

$$\overline{H}^{\Delta x}(P) \le \overline{H}(P) + O(\Delta x^{1/2}). \tag{22}$$

It is natural to discretize the spatial variable by computing the supremum only on the nodes of the triangulation x_i, $i = 1, \ldots, n$. So the fully discrete min-max problem is

$$\min_{u\in W^1} \max_{x_i} H\left(x_i, Du(x_i) + P\right). \tag{23}$$

The spatial approximation introduces an additional error of $O(\Delta x)$, which is proportional to the Lipschitz constant (in the x variable) of H. In our implementation, the min-max problem (23) has been written as a minimum problem for $F(Du)$, where the map $F : \mathbb{R}^{Nn} \to \mathbb{R}^{Nn}$ (recall that N is the dimension and n is the number of nodes) is defined by componentwise as $F_i(Du) = \max_i H(x_i, Du + P)$, for $i = 1, \ldots, n$. We note that the map F is still convex as long as H is convex. In order to simplify the notations, let us consider the case $N = 1$ with a uniform space step Δx. In [14] a discretization of Du by finite difference is used setting $Du(x_i) = (u_{i+1} - u_i)/\Delta x$ and then the problem is solved by the SQP (Sequential Quadratic Programming) MATLAB routine. The periodicity of u is automatically verified imposing $u_{n+1} = u_1$. Instead of introducing a discretization of $Du(x_i)$, we consider it as an independent variable, c_i, and so we consider the non linear constrained optimization problem

$$\min_{c_i} \max_{x_i} H(x_i, c_i + P) \quad \text{subject to} \sum_i c_i = 0. \tag{24}$$

As we said before, the linear constraint in (24) is equivalent to impose the periodicity of u. Although the linear constraint makes the problem harder, it improves the accuracy of the solution as $H \notin C^1$. In [14], the `fminimax` function, contained in the MATLAB Optimization Toolbox, is used to solve the problem. Here we use the optimization routine `ffsqp` [19]. Both the algorithms are based on SQP. It also provides two kinds of line search (monotone and non-monotone). We use the non-monotone line search, which forces a decrease of the objective function within at most four iterations. In fact, the monotone line search (of Armijo type) did not work in our experiments when H is not strictly convex, e.g. when $H(x,p) = |p| + V(x)$. We use `ffsqp` providing the gradient of the linear constraint and let it compute the gradient of the objective function. It uses a forward finite differences approximation.

3 Numerical results

The tests considered in this section are representative of a class of Hamiltonians of the kind $H = |p|^2/2 + V(x)$, i.e. the Hamiltonian is made by the sum of a kinetic plus a potential energy. In this case (see [16, 14]) an implicit formula for $\overline{H}$ is available. This allows for the precise computation of the numerical errors which is essential to compare the methods. Note also that the particular choice of the potential energy V is due to the fact that in this case we can also obtain the exact solution u of the cell problem. We present the numerical results for the three methods presented in the previous section comparing their accuracy and their cost in term of CPU time. The tests have been executed on a double-processor AMD opteron quad-core at 2.1 GHz without exploiting any parallelization option (so that the CPU time has to be intended for a serial run).
Let us consider the following one dimensional cell problem

$$\frac{1}{2}|Du + P|^2 = \frac{1}{2}\left(x - \frac{1}{2}\right)^2 + \overline{H}(P) \qquad x \in \mathbb{T}^N. \tag{25}$$

This implies $\overline{H}(P) \geq 0$. If $\overline{H}(P) > 0$,

$$|P| = \int_0^1 \sqrt{2\overline{H}(P) + \left(x - \frac{1}{2}\right)^2} \mathrm{d}x. \tag{26}$$

It easy to check that this equation has a solution $\overline{H}(P)$ whenever

$$|P| \geq \int_0^1 |x - \frac{1}{2}| \mathrm{d}x = \frac{1}{4}. \tag{27}$$

In this case $|P| = F(\overline{H}(P))$ with

$$F(z) = \frac{1}{4}\sqrt{8z+1} + z\left(\ln\left(2+2\sqrt{8z+1}\right) - \ln\left(-2+2\sqrt{8z+1}\right)\right). \qquad (28)$$

For $|P| \leq 1/4$, we have $\overline{H}(P) = 0$. Then, for every node P_j, $j = 1, \ldots, m$ in the P space, we define the error corresponding to an approximation based on n nodes in the x space,

$$e(P_j; n) = \begin{cases} \overline{H}^{\Delta}(P_j) & \text{for } |P| \leq 1/4 \\ F(\overline{H}^{\Delta}(P_j)) - |P_j| & \text{elsewhere} \end{cases}$$

In the following Tables we show the L^1 and L^∞ norm of the error vector e

$$\|e(n)\|_1 = \sum_{j=1}^{m} |e(P_j)|/m \qquad \|e(n)\|_\infty = \max_{j\in\{1,\ldots,m\}} |e(P_j)|. \qquad (29)$$

The numerical order of convergence is obtained by the standard formula

$$\frac{\log(\|e(n_1)\| \,/\, \|e(n_2)\|)}{\log(n_2/n_1)}$$

where $n_2 > n_1$ and $\|\cdot\|$ represent either the L^1 or the L^∞ norm. Note that in the tests $\overline{H}_1^{\Delta}$ has been computed over a grid on $[-0.5, 0.5]$ with $\Delta P = 0.01$.

The min-max method
As shown in Table 1, the order of convergence of the min-max method is greater than what we expected theoretically.

Table 1. L^1 and L^∞ norm of e and order for the min-max approximation

n	L^1	L^∞	L^1-order	L^∞-order
40	4.56×10^{-5}	1.04×10^{-4}		
80	1.14×10^{-5}	2.59×10^{-5}	2	2
160	2.85×10^{-6}	6.47×10^{-6}	2	2

The Euler-Lagrange method
We set $\varepsilon = n^{-3}$ and $k = n$, where k is that in formula (9). Note that the choice of k is conservative and that better results of that reported in Table 2 can be obtained increasing k at almost the same computational cost. Nevertheless for high values of k, the approximation of `argmin` of the functional is not stable when P belongs to the flat region of $\overline{H}$, i.e. for $|P| < 1/4$.

The SPG method
We set $\varepsilon = n^{-3}$ and $k = n$ as for the Euler–Lagrange method. We apply

Table 2. L^1 and L^∞ norm of e and order for the Euler–Lagrange approximation

n	L^1	L^∞	L^1-order	L^∞-order
40	1.10×10^{-2}	2.32×10^{-2}		
80	7.67×10^{-3}	1.59×10^{-2}	0.52	0.54
160	5.03×10^{-3}	1.01×10^{-2}	0.6	0.65

a normalization of the functional (10) like for the computation of the effective Hamiltonian in the Euler–Lagrange method to avoid computational overflow. In this case, we multiply (10) by the exponential of $-k\widetilde{H}$ where $\widetilde{H} = max_x H(x, 0 + P)$ and 0 is the starting point vector of the method. The value of $\overline{H}$ is then obtained by adding $\widetilde{H}$ to formula (9). Note that the errors reported in Table 3 are almost the same that for the Euler–Lagrange method.

Table 3. L^1 and L^∞ norm of e and order for the SPG approximation

n	L^1	L^∞	L^1-order	L^∞-order
40	1.05×10^{-2}	2.32×10^{-2}		
80	7.50×10^{-3}	1.59×10^{-2}	0.48	0.54
160	5.03×10^{-3}	1.01×10^{-2}	0.58	0.65

Two dimensional numerical results

The natural extension of the previous Hamiltonian to the two dimensional case is

$$\frac{1}{2}|Du + P|^2 = \frac{1}{2}\left(x - \frac{1}{2}\right)^2 + \frac{1}{2}\left(y - \frac{1}{2}\right)^2 + \overline{H}(P), \qquad x \in \mathbb{T}^N \tag{30}$$

so that $\overline{H}(P) = \overline{H}_1(P_1) + \overline{H}_1(P_2)$ where $P = (P_1, P_2)$ and $\overline{H}_1$ is the effective Hamiltonian in one dimension of the previous test. In this case to compute the error approximation we use an accurate approximation of $\overline{H}_1$. The computations have been made on $[-0.5, 0.5]^2$ with $\Delta P = 0.125$. This implies that we have to solve $m = 81$ optimization problems to compute $\overline{H}$.

The min-max method

Table 4 shows that just for $n = 20$ the computational cost is too high. Note also that when the argmin$H(x, 0)$ does not belong to the grid, as for $n = 11$, the method is less accurate.

The Euler–Lagrange method

We set $\varepsilon = n^{-3}$ and $k = n$ as in the one dimensional case. In Table 5 we do not report the order of convergence which varies between 0.4 and 0.6.

Table 4. L^1 and L^∞ norm of e and CPU time for the min-max approximation

n	L^1	L^∞	CPU time
10	5.11×10^{-4}	1.30×10^{-3}	500s
11	1.20×10^{-3}	2.10×10^{-3}	300s
20	1.27×10^{-4}	3.14×10^{-4}	$\simeq$ 16h

Table 5. L^1 and L^∞ norm of e and CPU time for the Euler–Lagrange approximation

n	L^1	L^∞	CPU time
40	1.55×10^{-2}	4.63×10^{-2}	1s
80	1.13×10^{-2}	3.18×10^{-2}	22s
160	7.42×10^{-3}	2.02×10^{-2}	7m52s

The SPG method

To obtain errors comparable with that of the Euler–Lagrange method we set $\varepsilon = n^{-3}/2$ and $k = n$. This choice is motivated by the fact that for $\varepsilon = n^{-3}$ the method, for $n = 160$, was less accurate with respect to the Euler–Lagrange method.

As we can see comparing Table 5 and 6 with the above choice we get almost the same accuracy. Increasing ε we get more accurate results but the computational cost increases. Note that the variables of the optimization problem are $2n^2$ whereas the number of constraints is $2n$.

Table 6. L^1 and L^∞ norm of e and CPU time for the SPG approximation

n	L^1	L^∞	CPU time
40	1.62×10^{-2}	4.64×10^{-2}	2s
80	1.16×10^{-2}	3.18×10^{-2}	23s
160	9.00×10^{-3}	2.02×10^{-2}	1m33s

In conclusion, we can say that the approximation of the functional (10) by SPG algorithm has comparable performance with respect to the Euler-Lagrange approach with the advantage to avoid to compute and discretize the Euler-Lagrange equation. Also the accuracy of the two methods is comparable and as expected from theoretical results.

Acknowledgement. The authors wish to thank also the CASPUR consortium for the technical support to the numerical tests.

References

1. Y. Achdou, F. Camilli, I. Capuzzo Dolcetta, *Homogenization of Hamilton–Jacobi equations: numerical methods*, Mathematical Models and Methods Applied Sciences, **18** (2008), 1115-1143.
2. J. Barzilai and J.M. Borwein, *Two point step size gradient methods*, IMA J. Numer. Anal., **8**, 141-148.
3. E. G. Birgin, J. M. Martinez and M. Raydan, *Nonmonotone spectral projected gradient methods on convex sets*, SIAM Journal on Optimization **10** (2000), 1196-1211.
4. E. G. Birgin, J. M. Martinez and M. Raydan, *Algorithm 813: SPG - software for convex-constrained optimization*, ACM Transactions on Mathematical Software **27** (2001), 340-349.
5. F. Camilli, I. Capuzzo-Dolcetta, and D. Gomes, *Error estimates for the approximation of the effective Hamiltonian*, Appl. Math. Optim. **57** (2008), 30–57.
6. G. Contreras, R. Iturriaga, G. P. Paternain, and M. Paternain, *Lagrangian graphs, minimizing measures and Mañé's critical values*, Geom. Funct. Anal. **8** (1998), no. 5, 788–809.
7. J.C. Dunn, *Gradient-related constrained minimization algorithms in function spaces: convergence properties and computational implications*, in Large Scale Optimization: State of the Art, W.W. Hager, D.W. Hearn and P.M. Pardalos (eds.), Kluwer, Dordrecht, 1994.
8. L. C. Evans, *Periodic homogenisation of certain fully nonlinear partial differential equations*, Proc. Roy. Soc. Edinburgh Sect. A **120** (1992), no. 3-4, 245–265.
9. L. C. Evans, *Some new PDE methods for weak KAM theory*, Calculus of Variations and Partial Differential Equations, **17** (2003), 159–177.
10. L. C. Evans. *Partial differential equations*, 19, *Graduate Studies in Mathematics*, American Mathematical Society, Providence, RI, 1998.
11. L. C. Evans and D. Gomes, *Effective Hamiltonians and Averaging for Hamiltonian Dynamics*, Archive for Rational Mechanics and Analysis, **157** 2001, 1–33.
12. M. Falcone and P. Lanucara and M. Rorro, *HJPACK Version 1.9 User's Guide*, 2006, `http://www.caspur.it/hjpack/user_guide1.9.pdf`
13. M. Falcone, M. Rorro, *On a variational approximation of the effective Hamiltonian*, in K. Kunisch, G. Of, O. Steinbach (eds.), Numerical Mathematics and Advanced Applications (Proceedings of ENUMATH 2007, Graz, Austria, September 10-14, 2007), Springer Berlin Heidelberg, 2008, 719-726.
14. D. Gomes and A. Oberman, *Computing the effective Hamiltonian using a variational approach*, SIAM J. Control Optim., **43** (2004), 792–812.
15. L. Grippo, F. Lampariello and S. Lucidi, *A non monotone line search technique for Newton's method*, SIAM J. Numer. Anal., **23** (1986), 707-716.
16. P. L. Lions and G. Papanicolau and S. Varadhan, *Homogenization of Hamilton–Jacobi equations*, unpublished.
17. J. Qian, *Two approximations for effective Hamiltonians arising from homogenization of Hamilton-Jacobi equations*, UCLA, Math Dept., preprint, 2003.
18. M. Rorro, *An approximation scheme for the effective Hamiltonian and applications*, Appl. Numer. Math., **56** (2006), 1238–1254.
19. J. L. Zhou and A. L. Tits and C. T. Lawrence, User's Guide for FFSQP Version 3.7: A FORTRAN Code for Solving Constrained Nonlinear Minimax Optimization Problems, Generating

Iterates Satisfying All Inequality and Linear Constraints, 1997, http://www.aemdesign.com/download-ffsqp/ffsqp-manual.pdf

Hybrid Solution Methods for Bilevel Optimal Control Problems with Time Dependent Coupling

Matthias Knauer[1] and Christof Büskens[2]

[1] Center for Industrial Mathematics, Universität Bremen, Bibliothekstraße 1, 28359 Bremen, knauer@math.uni-bremen.de
[2] Center for Industrial Mathematics, Universität Bremen, Bibliothekstraße 1, 28359 Bremen, bueskens@math.uni-bremen.de

Summary. To operate crane systems in high rack warehouses, reference trajectories have to ensure that the swinging of the crane is under control during the fast movement and disappears at the final point. These trajectories can be obtained solving optimal control problems.

For security reasons the optimal control problem of a main trajectory is augmented by additional constraints depending on the optimal solution of several safety stop trajectories leading to a bilevel optimal control problem.

1 Introduction

Bilevel programming as an extension to linear and nonlinear programming describes a static system of two decision makers or players, where the leader knows exactly, how the follower will react to his decision. It is well explored theoretically and various practical solution methods exist, cf. Bard [2] and Dempe [5]. Bilevel optimal control problems as a combination of two classical dynamic optimal control problems were introduced by Chen and Cruz [4] and are described in more detail by Ye [11], e.g.

While working at an industrial project, a closer look was necessary at the way, in which both levels of a bilevel optimal control problem interact in a dynamic system, leading to the formulation of time dependent coupling [7].

2 Path Planning for Container Cranes

The conventional solution to load and unload goods in a high rack warehouse is a floor driven shelf access equipment, similar to a fork lift truck. The same tasks can be fulfilled by using a ceiling driven container crane. The crane system consists of two parts, which have to be treated as a multibody system:

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_20, © Springer-Verlag Berlin Heidelberg 2010

Positioning A trolley moves along rails on the top of the warehouse rack. Via cable ropes it also controls the height by lifting or lowering a load-carrying equipment.

Loading/Unloading If the load-carrying equipment is positioned by the trolley, it can put or pick the payload from the rack with a fork-like construction.

This construction implies some crucial disadvantages. As soon as the trolley induces a lateral movement on the system, the load-carrying equipment starts to oscillate, as it is only attached to wire ropes. If oscillation still occurs at the final position, loading processes are impossible. This task of trajectory planning can be formulated as an optimal control problem.

As an industrial device, the crane system has to fulfil severe safety requirements: An **emergency stop** avoids critical behaviour in case of a system failure. At a user requested **safety stop**, the system should skip the current trajectory and invoke a controlled braking, so that the whole system comes to rest without any oscillation at an admissible final point within a given time. To allow a fast change between the main and the alternative trajectories, all data has to be online on the control unit before the main trajectory starts.

Obviously, the calculation of the alternative trajectories depends on the main trajectory. But the alternative trajectories can also influence the main trajectory. Coupling these two levels forms a bilevel optimal control problem.

3 Bilevel Optimization

The solution of a nonlinear programming problem is a variable $x^\star \in \mathbb{R}^n$, which minimizes an objective function under a set of (in-)equality constraints

$$\begin{aligned} \min_x \ & \bar{f}(x) \\ \text{s.t.} \ & \bar{g}_i(x) = 0,\ i = 1,\dots,m_e, \\ & \bar{g}_j(x) \le 0,\ j = m_e+1,\dots,m, \end{aligned} \tag{1}$$

where $\bar{f} : \mathbb{R}^n \to \mathbb{R}$ and $\bar{g} : \mathbb{R}^n \to \mathbb{R}^m$. Under regularity assumptions on $x^\star$ and differentiability assumptions on $\bar{f}$ and $\bar{g}$, first order necessary optimality conditions for (1) are given by the KKT conditions

$$\begin{aligned} \nabla_x L(x^\star, \lambda) &= 0, \\ \lambda_i &\ge 0, \quad \text{for } i \in \mathcal{I}(x^\star), \\ \lambda^T \bar{g}(x^\star) &= 0, \end{aligned} \tag{2}$$

with the Lagrangian $L(x,\lambda) = \bar{f}(x) + \lambda^T \bar{g}(x)$, the vector of Lagrange multipliers $\lambda \in \mathbb{R}^m$ and $\mathcal{I}(x^\star) = \{i \in \{m_e+1,\dots,m\} : \bar{g}_i(x^\star) = 0\}$ as the set of active inequality constraints, see [6].

In bilevel programming, two problems of type (1) are combined hierarchically. The first player, the leader, tries to find the best solution for his decision variable $x \in \mathbb{R}^N$ of the upper level problem[3]

$$\begin{aligned} \min_{x}\ & \bar{F}(x,y) \\ \text{s.t.}\ & \bar{G}(x,y) \leq 0, \\ & y \in \Psi(x). \end{aligned} \tag{3}$$

Problems of type (3) are known as mathematical programs with equilibrium constraints (MPEC), see Outrata et al. [8]. In bilevel programming, the equilibrium constraint is given by the solution set of another optimization problem: the leader's decision depends on the possible reactions $\Psi(x)$ of the other player, the follower. With a fixed variable x of the leader, the follower is looking for the best solution for his decision variable $y \in \mathbb{R}^n$ of the lower level problem

$$\begin{aligned} \min_{y}\ & \bar{f}(x,y) \\ \text{s.t.}\ & \bar{g}(x,y) \leq 0. \end{aligned} \tag{4}$$

Generally, the objective functions $\bar{F} : \mathbb{R}^N \times \mathbb{R}^n \to \mathbb{R}$, $\bar{f} : \mathbb{R}^N \times \mathbb{R}^n \to \mathbb{R}$ as well as the functions of the constraints $\bar{G} : \mathbb{R}^N \times \mathbb{R}^n \to \mathbb{R}^M$, $\bar{g} : \mathbb{R}^N \times \mathbb{R}^n \to \mathbb{R}^m$ depend on both decision variables on both levels.

If (4) is considered as a parametric problem depending on x, its feasible set $S(x)$ and its set of solutions $\Psi(x)$ are

$$\begin{aligned} S &: \mathbb{R}^N \to \{0,1\}^{\mathbb{R}^n}, \quad x \mapsto \{y \in \mathbb{R}^n : \bar{g}(x,y) \leq 0\}, \\ \Psi &: \mathbb{R}^N \to \{0,1\}^{\mathbb{R}^n}, \quad x \mapsto \arg\min_{y} \{\bar{f}(x,y) : \bar{g}(x,y) \leq 0\}. \end{aligned}$$

Problems (3) and (4) together form a Stackelberg game, where the leader knows the set $\Psi(x)$ of all possible reactions of the follower and is in advantage. However, he doesn't know which $y \in \Psi(x)$ will be actually chosen by the follower. By replacing the secondary problem of the bilevel problem (3, 4) by its KKT conditions (2), it can be reduced to a problem in standard formulation (1). Clearly, the solution set of the KKT conditions $KKT(x)$ holds the implication $\Psi(x) \subseteq KKT(x) \subseteq S(x)$. If the objective function and the feasible set in (4) are convex, the reduction leads to an equivalent problem, as $\Psi(x) = KKT(x)$:

$$\begin{aligned} \min_{x,y,\lambda}\quad & \bar{F}(x,y) \\ \text{s.t.}\quad & \bar{G}(x) \leq 0, \\ & \nabla_y L(x,y,\lambda) = 0, \\ & \bar{g}(x,y) \leq 0, \\ & \lambda \geq 0, \\ & \lambda^T \bar{g}(x,y) = 0, \end{aligned}$$

with $L(x,y,\lambda) = \bar{f}(x,y) + \lambda^T \bar{g}(x,y)$.

However, if the KKT conditions are used as complementary constraints, the standard constraint qualifications are violated at all feasible points, see

[3] For simplicity we omit the notation of possible equality constraints.

Scheel and Scholtes [10]. The concept of bilevel programming is used just as a motivation. Note that our analog approach for bilevel optimal control problems in the next section won't induce this problem.

4 Bilevel Optimal Control

The solution of an optimal control problem is a control vector $u^\star(t) \in \mathbb{R}^m$ and a set of free variables, as a free final time t_f for example, minimizing a given objective function, so that a boundary value problem for a state vector $x(t) \in \mathbb{R}^n$ holds under some additional control and state constraints

$$\begin{aligned} \min_{u,t_f}\ & \phi(x(t_f)) + \int_0^{t_f} f_0(x(t),u(t))dt \\ \text{s.t.}\ & \qquad \dot{x}(t) = f(x(t),u(t)), \\ & \omega(x(0),x(t_f)) = 0, \\ & g(x(t),u(t)) \le 0, \quad t \in [0;t_f], \end{aligned} \tag{5}$$

where $\phi : \mathbb{R}^n \to \mathbb{R}$, $f_0 : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$, $\omega : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^k$. If solved with indirect methods, (5) is converted into a boundary value problem using the necessary conditions from the minimum principle of Pontryagin [9]. The formulation for the special case, where only control constraints $u(t) \in U \subset \mathbb{R}^m$ are used instead of $g(x(t),u(t)) \le 0$ is needed at the end of this section. Under regularity assumption on the Hamiltonian $H(x,u,\lambda_0,\lambda) = \lambda_0 f_0(x,u) + \lambda^T f(x,u)$, there exist $\lambda_0 \in \mathbb{R}_0^+$, $\rho \in \mathbb{R}^r$, $\lambda \in C_p^1([0;t_f],\mathbb{R}^n)$ not all vanishing, so that

$$\begin{aligned} u^\star(x,\lambda) &= \operatorname*{arg\,min}_{u \in U} H(x,u,\lambda_0,\lambda), \\ \dot{\lambda}(t) &= -\nabla_x H(x^\star(t),u^\star(t),\lambda_0,\lambda(t)), \quad \text{for almost all } t \in [0;t_f], \\ \lambda(0)^T &= -\nabla_{x(0)} \left(\rho^T \omega(x^\star(0),x^\star(t_f))\right), \\ \lambda(t_f)^T &= -\nabla_{x(t_f)} \left(\lambda_0 \phi(x^\star(t_f)) + \rho^T \omega(x^\star(0),x^\star(t_f))\right). \end{aligned} \tag{6}$$

With direct methods, on the other hand, a numerical solution of (5) is found by transcribing it into an NLP problem of type (1). The continuous optimal control problem with $t \in [0;t_f]$ is reduced to a discretized version where only $t \in \{0 = t_1 \le t_2 \le \cdots \le t_l = t_f\}$, $l \in \mathbb{N}$ are considered. From the continuous control function $u(t)$, only a vector of discrete values $u = (u_1, \dots u^l)^T$ with $u^i \approx u(t_i)$ remains. The state function $x(t)$ is replaced by evaluations $x^i \approx x(t_i)$, which don't have to be stored, as they can be gathered depending on u:

$$\begin{aligned} \min \quad & \phi(x^l(u)) + \sum_{i=1}^{l-1} (t_{i+1} - t_i) f_0(x^i(u),u^i) \\ \text{s.t.} \quad & x^{i+1}(u) = x^i(u) + (t_{i+1} - t_i) f(x^i(u),u^i), \quad i = 1,\dots,l-1 \\ & \omega(x^1(u),x^l(u)) = 0 \\ & g(x^i(u),u^i) \le 0, \quad i = 1,\dots,l \end{aligned}$$

The software library NUDOCCCS by Büskens [3] generates an NLP problem of this type automatically and solves it with an SQP solver.

Similarly to bilevel programming, the optimal control problems for one leader and one follower can be combined, so that for the optimization of the control variable u of the leader's problem an additional constraint has to be considered, which depends on the optimal solution $v \in \Psi(u)$ of the follower's problem:

$$\begin{array}{l} \begin{array}{ll} \min\limits_{u} & \int\limits_0^{t_f} F_0(x(t), y(t), u(t), v(t))\, dt \\ \text{s.t.} & \dot{x}(t) = F(x(t), y(t), u(t), v(t)) \\ & \Omega(x(0), x(t_f)) = 0 \\ & G(x(t), u(t)) \leq 0 \\ & (y(t), v(t)) \in \Psi(u) \end{array} \\ \text{with } \Psi(u) \text{ set of solutions } y(t) = y(t;u) \text{ and } v(t) = v(t;u) \text{ of} \\ \begin{array}{ll} \min\limits_{v} & \int\limits_0^{t_f} f_0(x(t), y(t), u(t), v(t))\, dt \\ \text{s.t.} & \dot{y}(t) = f(x(t), y(t), u(t), v(t)) \\ & \omega(y(0), y(t_f)) = 0 \\ & v_{\min} \leq v(t) \leq v_{\max} \end{array} \end{array} \tag{7}$$

The formulation of bilevel optimal control problems in Ye [11] considers only a system of differential equations in the lower level. If two optimal control problems are coupled together as in (7), two distinct systems are used in the upper and the lower level. If F does not depend on $y(t)$ and $v(t)$, as in the following, these two formulations coincide, see [7].

Hence, if the lower level problem of (7) is replaced by its first order necessary conditions (6), a single level optimal control problem remains, being equivalent for a convex lower level problem:

$$\begin{array}{lrl} \min\limits_{u} & \multicolumn{2}{l}{\int\limits_0^{t_f} F_0(x(t), y(t), u(t), v(t))\, dt} \\ \text{s.t.} & \dot{x}(t) &= F(x(t), u(t)) \\ & \dot{y}(t) &= f(x(t), y(t), u(t), v(t)) \\ & \dot{\lambda}(t) &= -\nabla_y h(x(t), y(t), v(t), \lambda(t)) \\ & \Omega(x(0), x(t_f)) &= 0 \\ & \omega(y(0), y(t_f)) &= 0 \\ & \lambda(0)^T &= -\nabla_{x(0)} \left(\rho^T \omega(x(0), x(t_f)) \right) \\ & \lambda(t_f)^T &= -\nabla_{x(t_f)} \left(\lambda_0 \phi(x(t_f)) + \rho^T \omega(x(0), x(t_f)) \right) \\ & G(x(t), u(t)) &\leq 0 \\ & v(t) &= v(t; x, y, \lambda, u) \end{array} \tag{8}$$

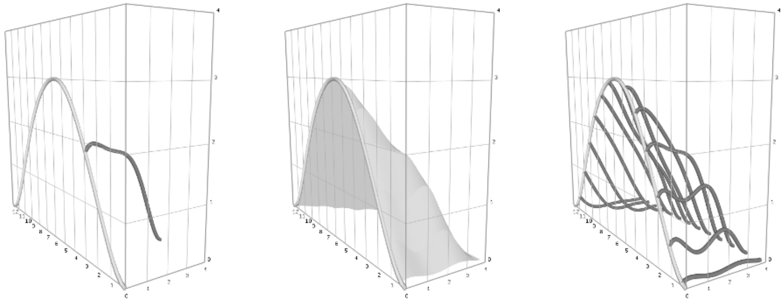


Fig. 1. Single, continuous and discrete time dependent coupling between primary and secondary problem.

Note that in (7), only control constraints are considered in the lower level problem. For a given system of ODEs $\dot{y} = f$, the control v can be expressed as a function of $v(x, y, \lambda, u)$ in the unconstrained case, and clipped to the control constraints in the constrained case.

Problem (8) as an ordinary optimal control problem can be transcribed into an NLP problem of type (1). Due to the structure of the Hessian and the Jacobian, sparse SQP solvers minimize calculation times, see Knauer [7].

5 Time Dependent Coupling

In section 4 the follower's problem was solved depending on the leader's decision variable u. Alternatively, for a given time ϑ the state $\xi = x(\vartheta)$ of the leader's system can be passed to the follower, who creates a solution set $\Psi(\xi)$:

$$\begin{aligned}
&\min_{u} \int_{0}^{t_f} F_0(x(t), u(t))\, dt \\
&\text{s.t. } \dot{x}(t) = F(x(t), u(t)) \\
&\qquad x(0) = x_0 \\
&\qquad (y, v) \in \Psi(\xi) \\
&\text{with } \xi = x(\vartheta), \vartheta \in [0; t_f] \text{ fixed} \\
&\text{and } \Psi(\xi) \text{ set of solutions } y(\tau) = y(\tau; \xi) \text{ and } v(\tau) = v(\tau, \xi) \text{ of} \\
&\min_{v} \int_{\tau_0}^{\tau_f} f_0(y(\tau), v(\tau))\, d\tau \\
&\text{s.t. } \dot{y}(\tau) = f(y(\tau), v(\tau)) \\
&\qquad y(0) = y_0 \\
&\qquad y(\tau_0) = \xi
\end{aligned} \tag{9}$$

Here, upper and lower level are treated as two systems with different timescales $t \in [0; t_f]$ and $\tau \in [\tau_0; \tau_f]$, which are only coupled at one time point. The lower level can be interpreted as an alternative ending of the upper level system, as shown in Fig. 1.

In order to calculate alternative trajectories for a safety stop as proposed in section 2, not just one state $\xi = x(\vartheta)$ at a single time point ϑ has to be considered for the set $\Psi(\xi)$ as in (9), but all states $\xi_\vartheta = x(\vartheta)$ at $\vartheta \in [0; t_f]$, leading to an infinite number of followers.

Due to differentiability of the solution of the lower level problem with respect to parameters ξ, a good approximation for the infinite number of followers can be found by only considering a finite number of followers for states $\xi_j = x(\vartheta_j)$ at selected time points $\vartheta_j \in [0; t_f]$, $j = 1, \dots, l$:

$$\begin{aligned} \min_u \ & \int_0^{t_f} F_0(x(t), u(t))\, dt \\ \text{s.t. } & \dot{x}(t) = F(x(t), u(t)) \\ & x(0) = x_0 \\ & (y_{\cdot,j}, v_{\cdot,j}) \in \Psi(\xi_j), \text{ for all } j = 1, \dots, k \end{aligned}$$

with $\xi_j = x(\vartheta_j)$, for $\vartheta_j \in [0; t_f]$ fixed
and $\Psi(\xi_j)$ set of solutions $y_{\cdot,j}(\tau) = y_{\cdot,j}(\tau, \xi_j)$ and $v_{\cdot,j}(\tau) = v_{\cdot,j}(\tau, \xi_j)$ of

$$\begin{aligned} \min_{v_{\cdot,j}} \ & \int_{\tau_{0,j}}^{\tau_{f,j}} f_0(y_{\cdot,j}(\tau), v_{\cdot,j}(\tau))\, d\tau \\ \text{s.t. } & \dot{y}_{\cdot,j}(\tau) = f(y_{\cdot,j}(\tau), v_{\cdot,j}(\tau)) \\ & y_{\cdot,j}(\tau_0) = \xi_j \end{aligned}$$

This problem formulation was already introduced by Abel and Marquardt [1]. They refer to the set of lower level problems for $\vartheta \in [0; t_f]$ as a scenario, and allow the consideration of different scenarios. They suggest to replace the bilevel problem either by a weighted formulation (SIOP3) or by a relaxed formulation (SIOP6), where the optimality in the lower level is neglected, see Knauer [7].

If the lower level problems are replaced by their first order necessary conditions one large optimal control problem is gained. Due to the special type of coupling, the subsystems of differential equations x and $y_{\cdot,j}, \lambda_{\cdot,j}$, $j = 1, \dots k$ can be integrated efficiently since they are independent from each other.

6 Main and Alternative Trajectories

The bilevel path planning problem can be outlined as "move from (S_0, L_0) to (S_f, L_f) and stop for $t = \tau_{0,j}$, $j = 1, \dots, k$", where for the system dynamics of the crane a jerk based model allows the necessary usage of acceleration values as boundary values ($g = 9.81 \left[\frac{m}{s^2}\right]$):

$$\dot{x} = f_{LK}(x,u) = \begin{pmatrix} x_2 \\ x_5 \\ x_4 \\ x_5 - (g - x_8)\frac{x_3}{x_6} \\ u_1 \\ x_7 \\ x_8 \\ u_2 \end{pmatrix} \quad \begin{array}{rl} x_1: & \text{position trolley} \\ x_2: & \text{velocity trolley} \\ x_3: & \text{rel. displacement payload} \\ x_4: & \text{rel. velocity payload} \\ x_5: & \text{acceleration trolley} \\ x_6: & \text{length of rope} \\ x_7: & \text{velocity rope} \\ x_8: & \text{acceleration rope} \end{array}$$

This leads to this bilevel optimal control problem:

$$\begin{aligned} \min_{u,t_f} \; & t_f + \int_0^{t_f} u_1^2(t) + u_2^2(t)\, dt \\ \text{s.t. } & \dot{x}(t) = f_{LK}(x(t), u(t)) \\ & x(0) = (S_0 \; 0 \; 0 \; 0 \; 0 \; L_0 \; 0 \; 0)^T, \quad x(t_f) = (S_f \; 0 \; 0 \; 0 \; 0 \; L_f \; 0 \; 0)^T \\ & u_i(t) \in [u_{i,\min}; u_{i,\max}], \quad i = 1, 2 \\ & x_i(t) \in [x_{i,\min}; x_{i,\max}], \quad i \in \mathcal{I}_c \\ & (y_{\cdot,j}, v_{\cdot,j}, \; j = 1, \ldots, k) \in \Psi(u) \end{aligned}$$

with $\Psi(u)$ set of solutions $y_{\cdot,j}(\tau) = y_{\cdot,j}(\tau; u)$ and $v_{\cdot,j}(\tau) = v_{\cdot,j}(\tau; u)$ of

$$\begin{aligned} \min_{v_{\cdot,j}} \; & \int_{\tau_{0,j}}^{\tau_{f,j}} v_{1,j}^2(\tau) + v_{2,j}^2(\tau)\, d\tau \\ \text{s.t. } & \dot{y}_{\cdot,j}(\tau) = f_{LK}(y_{\cdot,j}(\tau), v_{\cdot,j}(\tau)) \\ & y_{\cdot,j}(\tau_{0,j}) = x(\tau_{0,j}), \quad y_{\cdot,j}(\tau_{f,j}) = (\text{free } 0 \; 0 \; 0 \; 0 \text{ free } 0 \; 0)^T \\ & v_{i,j}(\tau) \in [v_{i,\min}; v_{i,\max}], \quad i = 1, 2 \end{aligned}$$

The main trajectory is calculated with respect to time and energy optimality. The alternative trajectories should stop within a fixed time minimizing energy consumption.

Each lower level problem for an alternative trajectory can be replaced by a boundary value problem using the necessary conditions. For energy optimality $\int v_{1,j}^2 + v_{2,j}^2 \, d\tau$ the system of differential equations

$$\dot{y}_{\cdot,j} = f_{LK}(y_{\cdot,j}, v_{\cdot,j}), \; y(\tau_{0,j}) = x(\tau_{0,j}), \; y(\tau_{f,j}) = (\text{free } 0 \; 0 \; 0 \; 0 \text{ free } 0 \; 0)^T$$

is augmented by a system of adjoint differential equations:

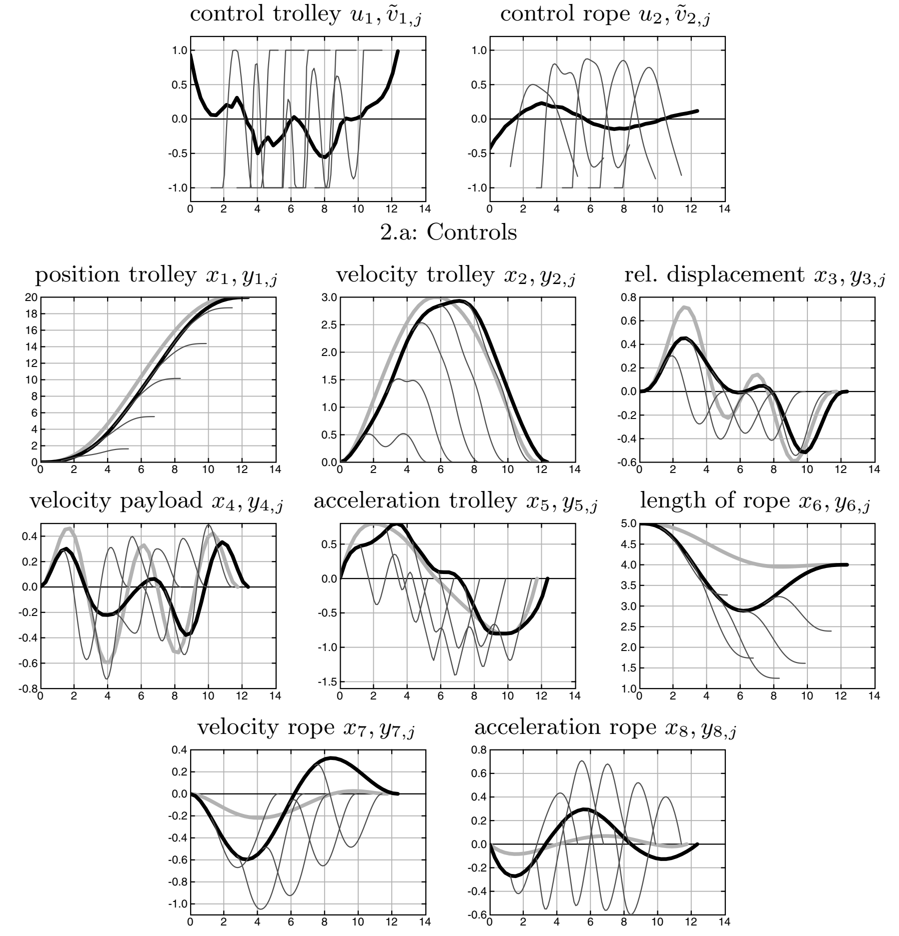
$$\dot{\lambda}_{\cdot,j} = \begin{pmatrix} 0 \\ -\lambda_{1,j} \\ \frac{\lambda_{4,j}(g - y_{8,j})}{y_{6,j}} \\ -\lambda_{3,j} \\ -\lambda_{2,j} - \lambda_{4,j} \\ -\frac{\lambda_{4,j} y_{3,j} (g - y_{8,j})}{y_{6,j}^2} \\ -\lambda_{6,j} \\ -\frac{\lambda_{4,j} y_{3,j}}{y_{6,j}} - \lambda_{7,j} \end{pmatrix}, \lambda_{\cdot,j}(\tau_{0,j}) = \begin{pmatrix} \text{free} \\ \text{free} \\ \text{free} \\ \text{free} \\ \text{free} \\ \text{free} \\ \text{free} \\ \text{free} \end{pmatrix}, \lambda_{\cdot,j}(\tau_{f,j}) = \begin{pmatrix} 0 \\ \text{free} \\ \text{free} \\ \text{free} \\ \text{free} \\ 0 \\ \text{free} \\ \text{free} \end{pmatrix}$$

Following the minimum principle (6), the controls can be calculated as

$$v_{1,j}(\tau) = -\frac{1}{2}\lambda_{5,j}(\tau), \quad v_{2,j}(\tau) = -\frac{1}{2}\lambda_{8,j}(\tau)$$

in the unconstrained case. Considering box constraints $v_{i,j}(t) \in [v_{i,\min}; v_{i,\max}]$, $i = 1, 2$ for the controls, the optimal solution is found using $\tilde{v}_{1,j}$ and $\tilde{v}_{2,j}$:

$$\tilde{v}_{i,j}(\tau) = \begin{cases} v_{i,\min} \text{ for } \lambda_{z,j}(\tau) > -2v_{i,\min} \\ -\frac{\lambda_{z,j}}{2} \text{ for } -2v_{i,\max} \leq \lambda_{z,j}(\tau) \leq -2v_{i,\min} \\ v_{i,\max} \text{ for } \lambda_{z,j}(\tau) < -2v_{i,\max} \end{cases} \quad (i, z) \in \{(1,5), (2,8)\}$$



2.a: Controls

2.b: States

Fig. 2. Optimal controls and state vectors for main trajectory (thick black) and 5 alternative trajectories (thin black) for constrained lower level control in comparison to single level problem (thick grey)

The numerical solution of this bilevel problem is shown in Fig. 2 for a main trajectory from $(S_0, L_0) = (0, 5)$ to $(S_f, L_f) = (20, 4)$ in $[m]$ with an equidistant discretization of $l = 41$ points. At the discretization points $l_j = 5 \cdot j$ of the main trajectory, alternative trajectories with $\tau_{f,j} - \tau_{0,j} = 4\,[s]$, $j = 1, \ldots, 5$ are considered with 11 discretization points. The state constraints in the upper level stay inactive. The controls for both levels are constrained by $(u_{i,\min}, u_{i,\max}) = (v_{i,\min}, v_{i,\max}) = (-1, 1)$ in $\left[\frac{m}{s^3}\right]$, $i = 1, 2$. The constraints in the lower level have a direct influence on the states of the main trajectory, where additionally the optimal solution without considering a safety stop is drawn. The oscillating behaviour, shown in the relative displacement of the payload, is reduced slightly at the cost of taking a longer time.

The continuous time dependent coupling led to a bilevel optimal control problem with an infinite number of followers, which was reduced to a finite selection, to find numerical solutions. Using necessary conditions, a classical optimal control problem was obtained, which was solved by direct methods, leading to a hybrid solution method.

References

1. Abel O, Marquardt W (2000) Scenario-integrated modeling and optimization of dynamic systems. AIChE J. 46(4):803–823
2. Bard J (1998) Practical Bilevel Optimization: Algorithms and Applications. Kluwer Academic Publishers, Dordrecht
3. Büskens C (1998) Optimierungsmethoden und Sensitivitätsanalyse für optimale Steuerprozesse mit Steuer- und Zustandsbeschränkungen. PhD Thesis, Universität Münster, Münster
4. Chen C, Cruz J (1972) Stackelberg Solution for Two-Person Games with Biased Information Patterns. IEEE Trans. Automat. Control 17(6):791–798
5. Dempe S (2002) Foundations of Bilevel Programming. Kluwer Academic Publishers, Dordrecht
6. Fletcher R (1987) Practical Methods of Optimization. Wiley, Chichester New York
7. Knauer M (2009) Bilevel-Optimalsteuerung mittels hybrider Lösungsmethoden am Beispiel eines deckengeführten Regalbediengerätes in einem Hochregallager. PhD Thesis, Universität Bremen, Bremen
8. Outrata J, Kočvara M, Zowe J (1998) Nonsmooth Approach to Optimization Problems with Equilibrium Constraints. Kluwer Academic Publishers, Dordrecht
9. Pontrjagin LS, Boltjansjkij VG, Gamkrelidze, RV, Miscenkzo EF (1967) Mathematische Theorie optimaler Prozesse. R. Oldenbourg, München Wien
10. Scheel H, Scholtes S (2000) Mathematical programs with equilibrium constraints: stationarity, optimality, and sensitivity. Math. Oper. Res. 25(1):1–22
11. Ye JJ (1997) Optimal Strategies for Bilevel Dynamic Problems. SIAM J Control Optim. 35(2):512–531

Consistent Control Procedures in the Monotone Structural Evolution. Part 1: Theory

Adam Korytowski and Maciej Szymkat

Institute of Automatic Control, AGH University of Science and Technology
Al. Mickiewicza 30, 30-059 Kraków, Poland {akor,msz}@ia.agh.edu.pl

Summary. The concept of consistent control procedures is introduced in optimal control computations. The stock of such procedures of the MSE, a direct method of dynamic optimization, is extended to handle state-constrained and interior arcs. Thus equipped, the MSE can automatically identify optimal control structures and yield arbitrarily exact approximations of optimal solutions by adjusting a bounded number of parameters.

1 Introduction

The method of Monotone Structural Evolution (MSE) is a direct computational method for dynamic optimization, see [6, 7, 8] and references therein. Its fundamental feature is that the decision space of the induced optimization problem undergoes gradual evolution, driven by discrepancy from the Maximum Principle conditions. The induced problems are solved by gradient methods with the cost derivatives evaluated by means of adjoint techniques, also used to trigger discrete changes of the decision space (*structural changes*). Since the control is not directly affected by structural changes, the cost monotonously decreases due to gradient optimization. Special rules prevent convergence to chattering modes.

We consider a control system described by a state equation

$$\dot{x}(t) = f(x(t), u(t)), \quad t \in [0, T], \quad x(0) = x_0, \tag{1}$$

with the state $x(t) \in \mathbf{R}^n$ and piecewise continuous controls u, $u(t) \in \mathbf{R}$. The optimal control problem is to find a control minimizing the cost functional

$$Q(u) = q(x(T)), \tag{2}$$

possibly subject to control bounds $u_{\min} \leq u(t) \leq u_{\max}$ and a scalar state constraint $g(x(t)) \leq 0$, $t \in [0, T]$. The initial state x_0 and the horizon T are fixed. The functions f, q and g are sufficiently smooth ($\mathcal{C}^1$, at least).

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_21, © Springer-Verlag Berlin Heidelberg 2010

Our aim is to introduce the concept of consistency in dynamic optimization, and propose consistent control procedures which handle state-constrained and interior arcs in the MSE algorithm. The MSE thus becomes more complete, with the stock of available control procedures being able to produce practically all types of arcs which may appear in an optimal solution. In consequence, the MSE can automatically identify optimal control structures and yield an arbitrarily exact approximation of the optimal solution by adjusting a bounded number of scalar parameters. To the authors' knowledge, this ability to find optimal structures automatically is unique among direct algorithms, although similar basic ideas (such as structural evolution, monotonicity, and spike generations, called 'mode insertions') are encountered in recent works on optimization of switched systems [1, 4].

The state-constrained problems are treated by penalty techniques. For index-1 constraints, a method with the exact fulfillment of the state constraint is described. The necessary jumps of the adjoint variables are calculated. A new technique of prototype adjoints, with some resemblance to the direct shooting [2], is constructed for consistent parameterization of interior arcs. For singular problems, we present a variant with partial elimination of adjoint variables, which extends earlier results [5] beyond pure state-feedback consistent representations of control. Numerical examples and a discussion of computational aspects are given in [9].

2 Basics of the MSE

We begin with the MSE elements relevant to this work. The general algorithm is described in [9], see also [6, 7, 8]. In the MSE approach to an optimal control problem, we first define a finite set $\mathbf{\Pi}$ (the *stock*) of appropriately regular *control procedures* $P : \mathbf{R} \times \mathbf{R}^{\mu(P)} \to \mathbf{R}$, where $\mu(P)$ is the number of scalar parameters of P. The functions P may be suggested by the Maximum Principle conditions and by general numerical techniques. Given *structural nodes* $\tau_0, \tau_1, ..., \tau_N$, $0 = \tau_0 \leq \tau_1 \leq ... \leq \tau_N = T$, and a *control structure* $S = (S_i)_{i=1}^N \in \mathbf{\Pi}^N$, the control is determined by

$$u(t) = S_i(t, p_i), \quad t \in [\tau_{i-1}, \tau_i[, \quad i = 1, ..., N. \tag{3}$$

Here $p_i \in \mathbf{R}^{\mu(S_i)}$. The restriction of u to an interval $[\tau_{i-1}, \tau_i[$ is called a *control arc*. The control structure, its parameters, and the nodes $\tau_1, ..., \tau_{N-1}$ are the decision variables of the MSE. Let $U_S : \mathbf{D}_a(S) \to \mathcal{U}$ be the mapping defined by (3), from the admissible set $\mathbf{D}_a(S)$ in the respective decision space $\mathbf{D}(S)$ into the functional control space $\mathcal{U}$. For a given control structure, the induced cost is given by $\Sigma = Q(U_S(\,\cdot\,)) : \mathbf{D}_a(S) \to \mathbf{R}$.

Recall that an admissible control is *extremal*, if it satisfies the Maximum Principle optimality conditions. A control procedure $P \in \mathbf{\Pi}$ is called *consistent*, if there are reals a and b, $0 \leq a < b \leq T$, a parameter $p^* \in \mathbf{R}^{\mu(P)}$ and an extremal control u^*, such that

$$u^*(t) = P(t, p^*), \quad t \in [a, b\,[. \tag{4}$$

The stock $\mathbf{\Pi}$ may also contain *approximative* control procedures. Observe that once an optimal control structure has been found, the concatenation of the corresponding control arcs becomes an optimal control when the parameters and nodes take optimal values, which is in contrast to typical approximation methods where approaching an exact optimal solution requires more and more parameters, with no finite limit in general. It is therefore important to equip the MSE with a sufficiently rich set of consistent control procedures, if we wish the method to identify optimal control structures.

The MSE algorithm uses cost gradients, computed by solving adjoint equations constructed for the induced problems. This construction is straightforward for control procedures of type A, which include all procedures considered in this paper, except those introduced in Section 4. Given $\mathbf{X} \subset \mathbf{R}^n$, $[a, b\,[\subset [0, T]$, and $\hat{\mathbf{D}} \subset \mathbf{R}^{\hat{\mu}}$, we say that a procedure $P \in \mathbf{\Pi}$ is *of type A* in $\mathbf{X} \times [a, b\,[\times \hat{\mathbf{D}}$, if there is a function $\hat{P} : \mathbf{R}^n \times \mathbf{R} \times \mathbf{R}^{\hat{\mu}} \to \mathbf{R}$ (also called a control procedure) such that $P(t, p) = \hat{P}(x(t, x_a), t, \hat{p})$ for all $t \in [a, b\,[$, $x_a \in \mathbf{X}$, $\hat{p} \in \hat{\mathbf{D}}$. Here $p = (x_a, \hat{p})$, and $x(t, x_a)$ is a solution of the state equation with $x(a, x_a) = x_a$. If S_i is of type A, that is, $S_i(t, p^i) = \hat{S}_i(x(t), t, \hat{p}_i)$, $t \in [\tau_{i-1}, \tau_i[$, the state equation in $[\tau_{i-1}, \tau_i[$ takes the form $\dot{x}(t) = \hat{f}(x(t), t)$, with $\hat{f}(\xi, t) = f(\xi, \hat{S}_i(\xi, t, \hat{p}_i))$. Define the Hamiltonian for the induced problem

$$\hat{H}(\hat{\psi}(t), x(t), t) = \hat{\psi}(t)^\top \hat{f}(x(t), t), \quad t \in [\tau_{i-1}, \tau_i[. \tag{5}$$

The adjoint function $\hat{\psi}$ is piecewise continuous, and its only possible discontinuities are described in Section 3. For a.a. $t \in [\tau_{i-1}, \tau_i[$ it satisfies

$$\dot{\hat{\psi}}(t) = -\nabla_x \hat{H}(\hat{\psi}(t), x(t), t). \tag{6}$$

For $i < N$, $\hat{\psi}(\tau_i-)$ is determined by the continuity or jump conditions, and

$$\hat{\psi}(T) = -\nabla q(x(T)). \tag{7}$$

3 State-constrained arcs

There are several ways of consistent, or asymptotically consistent representation of state-constrained arcs. All of them use a penalty approach with a family of auxiliary optimization problems parameterized by a (vector) penalty coefficient. An additional state equation is introduced to this end, $\dot{r} = \phi(x, \rho)$, $r(0) = 0$, where ϕ is an exterior penalty function, $\mathcal{C}^1$ in the first argument, and $\rho > 0$ is the penalty coefficient. As $\rho \to \infty$, it is required that $\phi(x, \rho) \to 0 \;\; \forall x : g(x) \leq 0$ and $\phi(x, \rho) \to \infty \;\; \forall x : g(x) > 0$. The respective auxiliary cost equals $Q_\rho(u) = q(x(T)) + r(T)$. Any differentiable exterior penalty function can be employed, but we find the exponential penalty especially useful. We then put $\phi(x, \rho) = \rho^{-1} \exp(\rho g(x))$.

To proceed, we need two definitions. A control procedure P is *candidate singular*, if it is consistent and additionally, the control (4) is singular on $[a, b\,[$. Similarly, a control procedure P is *candidate state-constrained*, if it is consistent and the state trajectory x^* produced by the control (4) satisfies $g(x^*(t)) = 0$, $t \in [a, b\,[$.

In the basic variant of the method, no special new elements are introduced into $\mathbf{\Pi}$. It should, however, include the candidate singular procedures of the auxiliary problems. The candidate singular arcs of the auxiliary problems can evolve, as $\rho \to \infty$, both into the candidate state-constrained and candidate singular arcs of the original problem. Since they are consistent in the auxiliary problems, for the original problem they are asymptotically consistent. The candidate singular arcs for a fixed ρ can be treated as in [5, 6], or by prototype adjoints as in Section 4.

In another variant, explicit candidate state-constrained procedures are used. Assume that the state constraint is of index k, that is, k differentiations of the identity $g(x(t)) = 0$ along system trajectories yield a control procedure of type A, $u(t) = P_{\mathrm{con}}(x(t))$. We construct a sequence of auxiliary optimization problems as in the basic variant, but adding P_{con} to $\mathbf{\Pi}$ and excluding from $\mathbf{\Pi}$ that candidate singular procedure which asymptotically represents state-constrained arcs of the original problem. Notice that P_{con} is consistent in the original state-constrained problem, but not in the auxiliary problems. In consequence, if we want to avoid complicated control structures in the intermediate solutions, we have to strengthen the conditions for generations on the candidate state-constrained arcs (see [6]).

In a third variant of the penalty method $P_{\mathrm{con}} \notin \mathbf{\Pi}$, and the elements of $\mathbf{\Pi}$ are modified. For a control procedure $P \in \mathbf{\Pi}$ assigned to a structural interval $[t_1, t_2[$, the entry time $t_{\mathrm{e}} \in\,]\, t_1, t_2[$ is defined by the relationships $g(x(t_{\mathrm{e}})) = 0$, and $g(x(t)) < 0$ for some $s > 0$ and every t in $[t_{\mathrm{e}} - s,\ t_{\mathrm{e}}[$. We put $t_{\mathrm{e}} = t_1$, if $g(x(t_1)) = 0$ and $\nabla g(x(t_1))^{\top} \hat{f}(x(t_1), t_1+) \geq 0$. The control u produced by the procedure P is modified for $t \in [t_{\mathrm{e}}, t_2[$ as follows

$$u(t) := \begin{cases} u_{\min}, & \text{if } P_{\mathrm{con}}(x(t)) \leq u_{\min} \\ P_{\mathrm{con}}(x(t)), & \text{if } P_{\mathrm{con}}(x(t)) \in [u_{\min}, u_{\max}] \\ u_{\max}, & \text{if } P_{\mathrm{con}}(x(t)) \geq u_{\max} \end{cases}$$

As a result, the adjoint variable defined by (6), (7) has a jump at t_{e}

$$\hat{\psi}(t_{\mathrm{e}}-) = Z\hat{\psi}(t_{\mathrm{e}}+), \tag{8}$$

where

$$Z = I - \frac{\nabla g(x(t_{\mathrm{e}}))\,\left(\hat{f}(x(t_{\mathrm{e}}), t_{\mathrm{e}}-) - \hat{f}(x(t_{\mathrm{e}}), t_{\mathrm{e}}+)\right)^{\top}}{\nabla g(x(t_{\mathrm{e}}))^{\top} \hat{f}(x(t_{\mathrm{e}}), t_{\mathrm{e}}-)}.$$

We give a proof in Appendix together with detailed assumptions. For the case of index 1 constraint, formula (8) coincides with a result in [3].

In some cases an additional penalization may improve convergence. Let t_1 denote t_{e} in the third variant and besides, the initial structural node of a control arc which asymptotically represents a state-constrained arc of the optimal solution. The additional penalty term in the auxiliary cost has the form

$$q_\sigma(x(t_1)) = \tfrac{1}{2} \sum_{i=0}^{k-1} \sigma_i \, g^{(i)}(x(t_1))^2, \quad \sigma_i \geq 0, \quad i = 0, ..., k-1, \tag{9}$$

where $g^{(i)}$ denotes the ith time derivative of $g(x(\,\cdot\,))$ along state trajectories. In the first two variants of the penalty method the resulting discontinuity of the adjoint variable at t_1 is given by $\hat{\psi}(t_1-) = \hat{\psi}(t_1+) - \nabla q_\sigma(x(t_1))$. In the third variant, this jump has to be added to that described by (8), and we put $\sigma_0 = 0$ in (9).

4 Interior arcs

The possibility of control arc parameterization in the MSE using adjoints was mentioned in [7]. While it was rightly estimated as promising in application to singular arcs in bang-singular optimal controls, in the general case it was then dismissed because of poor convergence, and in particular, the small area of convergence. Later it was found out that these difficulties can be overcome due to a proper generation policy and a freezing technique, which suppress the expansion of control arcs with parameter values far from optimal (see [9]).

4.1 Full parameterization with prototype adjoints

Define the Hamiltonian and the adjoint equation for the system (1) and cost (2)

$$H(\psi, x, u) = \psi^\top f(x, u)$$

$$\dot{\psi} = -\nabla_x H(\psi, x, u), \quad \psi(T) = -\nabla q(x(T)).$$

For ease of presentation, assume that there are no control or state constraints. Suppose that for every t the control maximizing the Hamiltonian can be obtained from the equation $\nabla_{u(t)} H(\psi(t), x(t), u(t)) = 0$ in the form $u(t) = P_B(x(t), \psi(t))$, with $P_B \in \mathcal{C}^1$. Define the *augmented system of state equations* (formally identical with the canonical system) in a structural time interval $[t_1, t_2[$

$$\dot{x} = F_1(x, y), \quad \dot{y} = F_2(x, y), \tag{10}$$

where $F_1(x, y) = f(x, P_B(x, y))$ and $F_2(x, y) = -\nabla_z H(y, z, P_B(x, y))|_{z=x}$. The variable x is continuous at t_1, and satisfies $x(t_1) = x_0$ if $t_1 = 0$. The variable y, called the *prototype adjoint*, satisfies $y(t_1) = p$. The parameter $p \in \mathbf{R}^n$ is a decision variable of the MSE.

We can now define a control procedure assigned to $[t_1, t_2[$

$$P'_B(t,p') = P_B(x(t\,;t_1,x(t_1),p), y(t\,;t_1,x(t_1),p)),$$

where $x(t\,;t_1,x(t_1),p)$ and $y(t\,;t_1,x(t_1),p)$ are the solution of (10) taking the value $p' = \mathrm{col}(x(t_1),p)$ at t_1. It directly follows from the construction that P'_B is consistent and is not of type A. Both P'_B and P_B will be called *control procedures of type B.* The *augmented Hamiltonian* is defined by

$$\hat{H}(\hat{\psi},\omega,x,y) = \hat{\psi}^\top F_1(x,y) + \omega^\top F_2(x,y),$$

where the *augmented adjoint* $\mathrm{col}(\hat{\psi},\omega)$ satisfies the *augmented adjoint system of equations*

$$\dot{\hat{\psi}} = -\nabla_x \hat{H}(\hat{\psi},\omega,x,y)$$

$$\dot{\omega} = -\nabla_y \hat{H}(\hat{\psi},\omega,x,y).$$

The variable $\hat{\psi}$ is continuous at t_2, and satisfies (7) if $t_2 = T$. The variable ω satisfies $\omega(t_2-) = 0$.

4.2 Derivatives of cost

Suppose that the control procedure S_i is of type B, for some $i \in \{1,...,N\}$. S_i is valid in the time interval $[\tau_{i-1},\tau_i[$. Denote the corresponding initial value of the prototype adjoint by p_i, $y(\tau_{i-1}) = p_i$. We will prove that the cost derivative w.r.t. p_i is given by

$$\nabla_{p_i} \Sigma = -\omega(\tau_{i-1}). \tag{11}$$

To this end, consider a variation δp_i of the parameter p_i and the resulting variation of the augmented state $\delta x(t)$, $\delta y(t)$ for $t \in [\tau_{i-1},\tau_i[$. By virtue of the well known property of the adjoints, $\hat{\psi}(t)^\top \delta x(t) + \omega(t)^\top \delta y(t) = \mathrm{const}$, $t \in [\tau_{i-1},\tau_i[$, and so

$$\hat{\psi}(\tau_i-)^\top \delta x(\tau_i-) + \omega(\tau_i-)^\top \delta y(\tau_i-) = \hat{\psi}(\tau_{i-1})^\top \delta x(\tau_{i-1}) + \omega(\tau_{i-1})^\top \delta y(\tau_{i-1}).$$

As $\delta x(\tau_{i-1}) = 0$, $\delta y(\tau_{i-1}) = \delta p_i$, $\omega(\tau_i-) = 0$, and $\hat{\psi}$ and δx are continuous at τ_i, we have $\hat{\psi}(\tau_i)^\top \delta x(\tau_i) = \omega(\tau_{i-1})^\top \delta p_i$. If $\tau_i = T$, the terminal condition (7) gives

$$-\nabla q(x(T))^\top \delta x(T) = \omega(\tau_{i-1})^\top \delta p_i, \tag{12}$$

whence (11) follows. Suppose now that $\tau_i < T$. If S_{i+1} is also of type B, we use a similar reasoning as above with $\delta y(\tau_i) = \delta p_{i+1} = 0$ and $\omega(\tau_{i+1}-) = 0$ to obtain

$$\hat{\psi}(\tau_{i+1})^\top \delta x(\tau_{i+1}) = \hat{\psi}(\tau_i)^\top \delta x(\tau_i) = \omega(\tau_{i-1})^\top \delta p_i. \tag{13}$$

If S_{i+1} is of type A, we immediately arrive at (13) using the equality $\hat{\psi}(t)^\top \delta x(t) = \mathrm{const}$, $t \in [\tau_i,\tau_{i+1}[$. Substituting $i := i+1$ and repeating this argument until $i = N$, we finally get (12) and (11).

We will now prove that the derivative of cost w.r.t. a structural node τ_i is given by

$$\nabla_{\tau_i} \Sigma = \hat{H}\,|_{\tau_i+} - \hat{H}\,|_{\tau_i-}, \quad \text{for } \ i \in \{1, ..., N-1\}. \tag{14}$$

Assume first that S_i and S_{i+1} are both of type B. A variation $\delta\tau_i$ of τ_i results in a variation of $x(\tau_i)$

$$\delta x(\tau_i) = (F_1(x(\tau_i), y(\tau_i-)) - F_1(x(\tau_i), y(\tau_i)))\,\delta\tau_i.$$

By assumption, $y(\tau_i) = y(\tau_i + \delta\tau_i) + \delta y(\tau_i + \delta\tau_i) + o(\delta\tau_i) = p_{i+1}$. Hence

$$\delta y(\tau_i) = -F_2(x(\tau_i), y(\tau_i))\,\delta\tau_i.$$

The variation of cost equals $\delta\Sigma = -\hat{\psi}(\tau_i)^\top \delta x(\tau_i) - \omega(\tau_i)^\top \delta y(\tau_i)$, and so

$$\nabla_{\tau_i} \Sigma = \hat{\psi}(\tau_i)^\top (F_1(x(\tau_i), y(\tau_i)) - F_1(x(\tau_i), y(\tau_i-))) + \omega(\tau_i)^\top F_2(x(\tau_i), y(\tau_i)). \tag{15}$$

As $\omega(\tau_i-) = 0$, we can rewrite this equality in the more convenient form (14) where $\hat{H}$ stands for the augmented Hamiltonian. It is easy to see that this formula remains valid if one of the procedures S_i and S_{i+1} is of type A. If S_i is of type A, the variation of $x(\tau_i)$ takes the form

$$\delta x(\tau_i) = (\hat{f}(x(\tau_i), \tau_i-) - F_1(x(\tau_i), y(\tau_i)))\,\delta\tau_i$$

and again we obtain (14), with $\hat{H}\,|_{\tau_i-}$ determined by (5). If S_{i+1} is of type A, we have

$$\delta x(\tau_i) = (F_1(x(\tau_i), y(\tau_i)) - \hat{f}(x(\tau_i), \tau_i+))\,\delta\tau_i$$

and $\delta\Sigma = -\hat{\psi}(\tau_i)^\top \delta x(\tau_i)$. As $\omega(\tau_i-) = 0$, we arrive at (14) with $\hat{H}\,|_{\tau_i+}$ determined by (5).

4.3 Spike and flat generations of type B procedures

Consider the situation immediately after a spike generation of procedure P_B as the ith element of a control structure S. Thus, S_i is of type B and $\tau_{i-1} = \tau_i$. Assume that the function $t \mapsto f(x(t), u(t))$ is continuous at τ_i, with u given by (3). Let first $\tau_i < T$ and S_{i+1} be of type A. The right-hand derivative of the cost w.r.t. τ_i can be written in the form

$$\nabla^+_{\tau_i} \Sigma = \hat{\psi}(\tau_i)^\top \hat{f}(x(\tau_i), \tau_i) - \hat{\psi}(\tau_i)^\top f(x(\tau_i), P_B(x(\tau_i), p_i)).$$

For $\tau_i > 0$ and S_{i-1} of type A, the left-hand derivative equals

$$\nabla^-_{\tau_{i-1}} \Sigma = \hat{\psi}(\tau_i)^\top f(x(\tau_i), P_B(x(\tau_i), p_i)) - \hat{\psi}(\tau_i)^\top \hat{f}(x(\tau_i), \tau_i).$$

Let now $\tau_i < T$ and S_{i+1} be of type B. From (15)

$$\begin{aligned}\nabla^+_{\tau_i} \Sigma = \hat{\psi}(\tau_i)^\top f(x(\tau_i), P_B(x(\tau_i), y(\tau_i))) - \hat{\psi}(\tau_i)^\top f(x(\tau_i), P_B(x(\tau_i), p_i))\\ + \omega(\tau_i)^\top F_2(x(\tau_i), y(\tau_i)).\end{aligned}$$

For $\tau_i > 0$ and S_{i-1} of type B

$$\nabla^{-}_{\tau_{i-1}} \Sigma = \hat{\psi}(\tau_i)^\top f(x(\tau_i), P_B(x(\tau_i), p_i)) - \hat{\psi}(\tau_i)^\top f(x(\tau_i), P_B(x(\tau_i), y(\tau_i-))) \\ -\omega(\tau_i)^\top F_2(x(\tau_i), y(\tau_i-)).$$

If $0 < \tau_i < T$, we have $\nabla^{-}_{\tau_{i-1}} \Sigma = -\nabla^{+}_{\tau_i} \Sigma$. As P_B maximizes the Hamiltonian, these formulae show that $\nabla^{+}_{\tau_i} \Sigma$ attains a minimum at $p_i = \hat{\psi}(\tau_i)$, and this minimum is nonpositive. The equality $\nabla^{+}_{\tau_i} \Sigma = 0$ may only occur if the necessary condition of the Maximum Principle is fulfilled at τ_i, but in that case the MSE algorithm does not allow generations. Similarly, $\nabla^{-}_{\tau_{i-1}} \Sigma$ attains a nonnegative maximum at $p_i = \hat{\psi}(\tau_i)$.

Consider now a flat generation of type B, which consists in inserting a new structural node τ_i, $0 < \tau_i < T$, inside a structural interval of type B. The procedures S_i and S_{i+1} are then of type B, and $p_{i+1} = y(\tau_i) = y(\tau_i-)$. The formulae (11) and (15) give the values of $\nabla_{p_i} \Sigma$, $\nabla_{p_{i+1}} \Sigma$, $\nabla_{\tau_{i-1}} \Sigma$ and $\nabla_{\tau_i} \Sigma$. The remaining components of the cost gradient in the decision space are not changed, except for the obvious renumeration.

4.4 Partial parameterization with prototype adjoints

Consider a problem (1), (2) with the Hamiltonian affine in control

$$H(\psi, x, u) = H_0(\psi, x) + H_1(\psi, x)u.$$

A control u is candidate singular on $[t_1, t_2[$, if $H_1(\psi(t), x(t)) \equiv 0$ in $[t_1, t_2[$. Assume that for some even $k > 0$, k successive differentiations of this identity along system trajectories yield

$$H_1^{(i)}(\psi(t), x(t)) = 0, \quad i = 0, ..., k-1 \tag{16}$$

$$H_{10}^{(k)}(\psi(t), x(t)) + H_{11}^{(k)}(\psi(t), x(t))\, u(t) = 0, \quad H_{11}^{(k)}(\psi(t), x(t)) \neq 0. \tag{17}$$

This set of $k+1$ equations is linear in $\psi(t)$. By virtue of (17), it can be solved w.r.t. $u(t)$ in the form $u(t) = P_B(x(t), \psi(t))$ and the theory of Section 4.1 may be applied. However, it is advantageous both from the numerical and analytical point of view to use the equations (16) to eliminate some components of $\psi(t)$ from the expression for $u(t)$. If the vector $\psi(t)$ is entirely eliminated, we obtain a candidate singular control in a pure state feedback form; that case was discussed in [5, 6]. In the general case assume that $n - \bar{n}$ components of ψ are eliminated and the remaining components constitute a vector $\bar{\psi} \in \mathbf{R}^{\bar{n}}$. Let the function $\chi : \mathbf{R}^{\bar{n}} \times \mathbf{R}^n \to \mathbf{R}^n$ assign the respective values of $\psi \in \mathbf{R}^n$ to every $\bar{\psi} \in \mathbf{R}^{\bar{n}}$ and $x \in \mathbf{R}^n$, which means that $\psi(t) = \chi(\bar{\psi}(t), x(t))$. We then define a procedure of type B

$$\bar{P}_B(x(t), \bar{\psi}(t)) = P_B(x(t), \chi(\bar{\psi}(t), x(t))).$$

The parameterization technique proposed in 4.1 can now be employed. Of course, the number of eliminated adjoints is to an extent a matter of choice, but the computational experience shows that eliminating more components of ψ usually improves convergence.

5 Conclusions

It has been shown that general interior and state-constrained arcs of optimal control may be produced in the MSE by means of consistent, or asymptotically consistent control procedures, with state constraints of index 1 exactly satisfied.

In consequence, typical optimal controls may be entirely approximated by consistent procedures, and so an arbitrarily accurate approximation can be fully characterized by a bounded, relatively small number of parameters, which are decision variables in the induced optimization problems. This is in contrast to most approximation methods where increasing accuracy requires more and more parameters, without a finite limit. An important property characteristic of indirect methods has thus been attained, but without the well-known drawbacks of those methods, such as small areas of convergence or discontinuous state trajectories in intermediate solutions.

Several consistent procedures have been proposed and characterized. It should be stressed that the construction of efficient computational algorithms based on the presented theory requires that the general scheme of the MSE be completed with some additional rules and specific techniques. These issues are discussed in Part 2 of this paper (see [9]), together with illustrative numerical examples.

References

1. Axelsson H, Wardi Y, Egerstedt M, Verriest E (2008), A gradient descent approach to optimal mode scheduling in hybrid dynamical systems. JOTA 136, 2:167–186
2. Diehl M, Leineweber D B, Schäfer A (2001), MUSCOD-II Users' Manual. University of Heidelberg, IWR-Preprint 2001-25
3. Fiorini P, Shiller Z (1997), Time optimal trajectory planning in dynamic environments. Appl Math Comp Sci 7, 2:101–126
4. Gonzalez H, Vasudevan R, Kamgarpour M, Sastry S S, Bajcsy R, Tomlin C J (2010), A descent algorithm for the optimal control of constrained nonlinear switched dynamical systems. 13th HSCC, Stockholm
5. Korytowski A, Szymkat M, Maurer H, Vossen G (2008), Optimal control of a fedbatch fermentation process: numerical methods, sufficient conditions and sensitivity analysis. 47th IEEE CDC, Cancun, 1551–1556
6. Szymkat M, Korytowski A (2003), Method of monotone structural evolution for control and state constrained optimal control problems. ECC, Cambridge
7. Szymkat M, Korytowski A (2007), Evolution of structure for direct control optimization. Discussiones Mathematicae DICO, 27:165–193
8. Szymkat M, Korytowski A (2008), The method of monotone structural evolution for dynamic optimization of switched systems. 47th IEEE CDC, Cancun, 1543–1550
9. Szymkat M, Korytowski A (2010), Consistent control procedures in the monotone structural evolution. Part 2: Examples and computational aspects. This volume

Appendix

Assume that $f_1 : \mathbf{R}^n \times \mathbf{R} \to \mathbf{R}^n$ is $\mathcal{C}^1$ in the first argument and continuous in the second, $\theta_0 \subset [0, T]$ is an open interval, P_{con} is the control procedure defined in Section 3, and $f_2(\xi) = f(\xi, P_{\text{con}}(\xi))$, $f_2 \in \mathcal{C}^1$. Consider a state equation

$$\dot{z}(t) = \left\{ \begin{array}{ll} f_1(z(t), t), & \text{if } g(z(t)) < 0 \\ f_2(z(t)), & \text{if } g(z(t)) \geq 0 \end{array} \right\}, \quad t \in \theta_0 \; . \tag{A1}$$

Let x be a solution of (A1), $s, t_\text{e} \in \theta_0$, $s < t_\text{e}$, $f_1(x(t_\text{e}), t_\text{e})^\top \nabla g(x(t_\text{e})) > 0$, $g(x(t)) < 0$ if $\theta_0 \ni t < t_\text{e}$, and $g(x(t)) \geq 0$ if $\theta_0 \ni t \geq t_\text{e}$. Due to the rule of the MSE algorithm which enforces saturation generations before every gradient computation (see [9]) we may assume, without a loss of generality, that $u_{\min} < P_{\text{con}}(x(t)) < u_{\max}$ if $t \in \theta_0$. Denote by $z(t, \xi)$ the solution of (A1) which satisfies $z(s, \xi) = \xi$. By a continuity argument and by the definition of f_2, there exist an open neighborhood X of $x(s)$ and an open interval $\theta \subset \theta_0$ containing s and t_e, with the following properties

$$\forall \xi \in X \quad \exists \eta(\xi) \in \theta : \quad f_1(z_\xi, \eta(\xi))^\top \nabla g(z_\xi) > 0,$$

$$g(z(t, \xi)) < 0 \;\text{ if } s \leq t < \eta(\xi), \; g(z(t, \xi)) \geq 0 \;\text{ if } t \geq \eta(\xi), t \in \theta,$$

where $z_\xi = z(\eta(\xi), \xi)$. It follows from the Implicit Function Theorem that the function η, $X \ni \xi \mapsto \eta(\xi) \in \theta$ is of class $\mathcal{C}^1$, and

$$\nabla \eta(\xi) = - \frac{\nabla_2 z(\eta(\xi)-, \xi) \, \nabla g(z_\xi)}{f_1(z_\xi, \eta(\xi))^\top \, \nabla g(z_\xi)} . \tag{A2}$$

Consider ξ_1, $\xi_2 \in X$ and denote $\eta_i = \eta(\xi_i)$, $z_i(t) = z(t, \xi_i)$, $i = 1, 2$. Let $\eta_2 \geq \eta_1$. From (A1),

$$z_2(\eta_2) - z_1(\eta_2) = z_2(\eta_1) - z_1(\eta_1) + (f_1(z_2(\eta_1), \eta_1) - f_2(z_1(\eta_1)))(\eta_2 - \eta_1) + o(\eta_2 - \eta_1).$$

Substituting here $z_2(\eta_1) - z_1(\eta_1) = \nabla_2 z(\eta_1-, \xi_1)^\top (\xi_2 - \xi_1) + o(\xi_2 - \xi_1)$, $z_2(\eta_2) - z_1(\eta_2) = \nabla_2 z(\eta_2, \xi_1)^\top (\xi_2 - \xi_1) + o(\xi_2 - \xi_1)$ and $\eta_2 - \eta_1 = \nabla \eta(\xi_1)^\top (\xi_2 - \xi_1) + o(\xi_2 - \xi_1)$, obtain

$$\begin{aligned} \nabla_2 z(\eta_2, \xi_1)^\top (\xi_2 - \xi_1) &= \nabla_2 z(\eta_1-, \xi_1)^\top (\xi_2 - \xi_1) \\ &+ (f_1(z_2(\eta_1), \eta_1) - f_2(z_1(\eta_1))) \; \nabla \eta(\xi_1)^\top (\xi_2 - \xi_1) + o(\xi_2 - \xi_1). \end{aligned} \tag{A3}$$

Let $\xi_1 \to x(s)$, $\xi_2 \to x(s)$, so that $\eta_1 \to t_\text{e}-$, $\eta_2 \to t_\text{e}+$. Then $\nabla_2 z(\eta_1-, \xi_1) \to \nabla_2 z(t_\text{e}-, x(s))$, $\nabla_2 z(\eta_2, \xi_1) \to \nabla_2 z(t_\text{e}+, x(s))$. Finally, from (A3) and (A2)

$$\nabla_2 z(t_\text{e}+, x(s)) = \nabla_2 z(t_\text{e}-, x(s)) \, Z$$

$$Z = I - \frac{\nabla g(x(t_\text{e}))}{\nabla g(x(t_\text{e}))^\top f_1(x(t_\text{e}), t_\text{e})} \; (f_1(x(t_\text{e}), t_\text{e}) - f_2(x(t_\text{e})))^\top \; .$$

We require $\nabla_2 z(t_\text{e}-, x(s)) \, \psi(t_\text{e}-) = \nabla_2 z(t_\text{e}+, x(s)) \, \psi(t_\text{e}+)$. As $\nabla_2 z(t_\text{e}-, x(s))$ is nonsingular, we obtain $\psi(t_\text{e}-) = Z \psi(t_\text{e}+)$.

Consistent Control Procedures in the Monotone Structural Evolution. Part 2: Examples and Computational Aspects

Maciej Szymkat and Adam Korytowski

Institute of Automatic Control, AGH University of Science and Technology
Al. Mickiewicza 30, 30-059 Kraków, Poland {msz,akor}@ia.agh.edu.pl

Summary. The consistent control procedures for state-constrained and interior arcs are implemented in the MSE, and their performance demonstrated on numerical examples. For state constrained problems with index 1, a two-phase technique is proposed which ensures the exact fulfillment of the state constraint. To enhance efficiency of the method of prototype adjoints applied to consistent representation of interior control arcs, a new 'freezing' technique is used.

1 Introduction

Our aim is to present an implementation of the consistent control procedures of Part 1 [5] in the MSE. To make it efficient, the MSE has been equipped with special techniques essential for the rate of convergence and accuracy of results.

We first recall the basic features of the MSE algorithm. Then, a consistent approach to state constrained problems with index 1 is demonstrated using two-phase optimization with shifted penalty. It ensures the exact fulfillment of the state constraint. We next show the method of prototype adjoints with full parameterization and a 'freezing' technique, applied to interior arcs. The variant with partial parameterization is presented for the singular case. The explanations are illustrated with three numerical examples. We also give an account of the computational environment and numerical procedures of the MSE. The notations and definitions introduced in [5] are used throughout the paper.

2 Algorithm of MSE

In the MSE, the decision space evolves in a series of *structural changes*, separated by periods of gradient optimization in a constant space. Let $\mathbf{\Pi}$ denote

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_22, © Springer-Verlag Berlin Heidelberg 2010

the stock of available control procedures and $\mathbf{D}_a(S)$, the set of admissible decision vectors for a given control structure S. Each structural change consists in replacing the current control structure $S \in \mathbf{\Pi}^N$ and decision vector $d \in \mathbf{D}_a(S)$ by a new structure $\bar{S} \in \mathbf{\Pi}^{\bar{N}}$ and decision vector $\bar{d} \in \mathbf{D}_a(\bar{S})$. It has to satisfy the *condition of control preservation* $U_{\bar{S}}(\bar{d}) = U_S(d)$ where $U_\sigma(\delta)$ stands for the control produced by $\delta \in \mathbf{D}_a(\sigma)$. The control as an element of the functional control space $\mathcal{U}$ is not immediately affected, and in consequence, the cost monotonously decreases. To define the *efficiency* E of a structural change, denote $\bar{\Sigma} = Q(U_{\bar{S}}(\cdot))$. If the antigradient $\gamma = -\nabla\Sigma(d)$ points to $\text{int}\mathbf{D}_a(S)$ and $\bar{\gamma} = -\nabla\bar{\Sigma}(\bar{d})$ to $\text{int}\mathbf{D}_a(\bar{S})$, $E = \,||\,\bar{\gamma}||^2 - \,||\,\gamma\,||^2$. In the general case the antigradients are replaced by their orthogonal projections onto the local conical approximations of the admissible sets.

Two kinds of structural changes are typical for the MSE: the number of decision variables increases in *generations*, and is diminished in *reductions*. The aim of *driving* generations is to speed up optimization when it is approaching a stationary point in the current decision space. Such a generation usually consists of adding one or more procedures to the current control structure, or reparameterization of some procedures. Typically, it has the form of inserting a *spike* of a new control arc of zero length. The driving generation occurs when its efficiency exceeds a given threshold, $E > \varepsilon(||\,\gamma\,||)$ where ε is a continuous strictly increasing function vanishing at 0. The new structure $\bar{S}$ and point $\bar{d}$ are chosen so as to maximize the efficiency subject to some additional rules, such as limiting the number of new decision variables or the number of affected procedures. One of the most essential is the *rule of minimum positive efficiency* (see [9] and references therein) used for choosing new consistent procedures to be inserted into S. It prevents the MSE algorithm from convergence to chattering modes. The MSE also admits *saturation generations*, enforced by the requirement that at the moment of gradient computation each control arc has to be either purely boundary or purely interior. Typical reductions consist of eliminating arcs of zero length when they are not promising, or unification of two adjacent arcs described by identical procedures.

The MSE algorithm begins with the selection of an initial control structure S_0 and a starting point in $\mathbf{D}_a(S_0)$. An iteration of the algorithm, in its basic form, contains the following steps.

1^0 Termination, if MP optimality conditions in $\mathcal{U}$ are satisfied.

2^0 Generation, if it is sufficiently efficient or needed.

3^0 Iteration of gradient optimization in current decision space.

4^0 Reduction, if necessary.

The iteration involves a solution of the adjoint equations and an evaluation of the cost gradient. In step 1^0 it is verified if the condition of Hamiltonian maximization is fulfilled with sufficient accuracy. This can be also formulated as a condition of existence of appropriately efficient generations. Step 2^0 is distinctive for the MSE and crucial for its convergence.

3 State-constrained arcs

Here we present a consistent approach to problems with state constraint of index 1, which ensures the exact fulfillment of the constraint due to the modification of control procedures described in Section 3 of [5]. This modification will enforce the state trajectory to slide along the boundary of the admissible region until the end of the structural interval. To avoid the Zeno effect and traps of conflicting constraints, known as ‘blocking behavior’, we employ a preliminary phase of computations where the penalty method of the basic variant (see Section 3 of [5]) is used with only bang control procedures and a strengthened state constraint, that is, the boundary of the set of admissible states shifted inward. The preliminary phase is continued until a solution is obtained which is sufficiently close to optimal and satisfies the original state constraint. The proper, penalty-free phase is then started, with the original state constraint, original cost, and the modified control procedures. In this phase, generations are suppressed on the time intervals where the state constraint is active. Note that the control is preserved at the switch of phases, as it is admissible.

We illustrate this method with an example of the pendulum on a cart [8]. The state equations are as follows

$$\dot{x}_1 = x_3, \quad \dot{x}_2 = x_4$$

$$\dot{x}_3 = \frac{u - x_4^2 \sin x_2 + \sin x_2 \cos x_2}{2 - \cos^2 x_2}, \quad \dot{x}_4 = \frac{(u - x_4^2 \sin x_2)\cos x_2 + 2 \sin x_2}{2 - \cos^2 x_2}.$$

The control is bounded, $-u_{\max} \le u(t) \le u_{\max}$, and a pathwise constraint is imposed on the cart velocity $x_3(t) \le x_{3\,\max}$, $t \in [0, T]$. The initial state $x(0) = \mathrm{col}(0, \pi, 0, 0)$ and the horizon T are fixed. The cost is given by

$$Q(u) = \tfrac{1}{2}(x_1(T)^2 + x_2(T)^2 + x_3(T)^2 + x_4(T)^2).$$

At the initial moment of time the cart is at rest at zero, with the pendulum in the down stable position. The control task is to steer the system as close as possible to another, unstable equilibrium where the cart again is stopped at zero, but the pendulum is at rest in the upward position. For calculations we take $T = 2.5$, $u_{\max} = 4$, $x_{3\,\max} = 1.8$. On a state-constrained arc $x_3 = x_{3\,\max}$, $\dot{x}_3 = 0$, whence $u = P_{\mathrm{con}}(x) = (x_4^2 - \cos x_2)\sin x_2$, and $\dot{x}_1 = x_{3\,\max}$, $\dot{x}_4 = \sin x_2$.

In the preliminary phase of computations we use the penalty coefficient $\rho = 8$ and the state boundary shifted inward, $x_{3\,\max} = 1.4$. Only two control procedures are allowed, $P_{1,2} = \pm u_{\max}$. Fig. 1a shows the generation of two spikes of bang control at iteration 4. The corresponding state trajectories are depicted in Fig. 1b. The preliminary phase is stopped after 12 iterations, when the solution is sufficiently close to optimal and satisfies the original state constraint, see Figs. 1c (control) and 1d (states). In the next, proper phase, the original cost Q is minimized and the original state constraint with $x_{3\,\max} = 1.8$

is strictly observed. The same two control procedures are used, but modified as in Section 3 of [5]. One state-constrained arc is created as a result of constraint activation, with the entry time enforced by the trajectory and the exit time being a decision variable. The optimal solution, shown in Figs. 1e and 1f is obtained in 24 iterations. The respective adjoints are presented in Fig. 2b (note the discontinuity at the entry time).

The switching function defined as $\hat{\psi}^\top \nabla_u f(x,u)$ and plotted in Fig. 1e in a normalized form, indicates that the Maximum Principle necessary conditions of optimality are satisfied. Note that this function is decreasing on the state-constrained arc and vanishes at its end. Simple calculations show that the conditions of Theorem 5.2 in [4] are thus fulfilled, which is a consequence of the fact that the adjoint variable $\hat{\psi}$ of the MSE coincides with the adjoint variable in the 'indirect adjoining' approach, and the switching function multiplied by a weakly varying positive function $2 - \cos^2 x_2(t)$ is equal to the appropriate Lagrange multiplier on the state-constrained arc.

Fig. 2a shows the evolution of control structure during the optimization. The iteration numbers are on the horizontal axis, and the control time $t \in [0,T]$ on the vertical axis. The black color represents $u(t) = u_{\max}$, white $u(t) = -u_{\max}$, and grey $u(t) = P_{\mathrm{con}}(x(t))$. The vertical dashed line marks the switch of phases (also in Figs. 2c and 2d). The values of $\max_{t\in[0,T]} g(x(t)) = \max_{t\in[0,T]}(x_3(t) - 1.8)$ in successive iterations are presented in Fig. 2c. Note that the preliminary phase (to the left of the vertical dashed line) ends when the state trajectory becomes admissible, $\max_{t\in[0,T]} g(x(t)) < 0$. As can be seen in Fig. 2a, a state-constrained control time-interval appears in the next, 13th iteration. Fig. 2d shows the values of norm of gradient (dots) and of the difference 'cost − optimal cost' (circles) in successive iterations. The scale on the vertical axis is decimal logarithmic.

4 Full parameterization with prototype adjoints

We again consider the pendulum on a cart described by the state equations of Section 3. No control or state constraints are assumed. The initial state $x(0) = \mathrm{col}(0, \pi, 0, 0)$ and the horizon T are fixed. The cost function has an integral form

$$Q(u) = \tfrac{1}{2} \int\limits_0^T (\beta_1 x_1^2 + \beta_2 x_2^2 + x_3^2 + x_4^2 + \alpha u^2)\mathrm{d}t,$$

where $\beta_1 = \beta_2 = 4$, $\alpha = 0.1$, $T = 2.5$. The adjoint equations read

$$\dot{\psi}_1 = \beta_1 x_1, \quad \dot{\psi}_2 = -A_{32}\psi_3 - A_{42}\psi_4 + \beta_2 x_2$$

$$\dot{\psi}_3 = -\psi_1 + x_3, \quad \dot{\psi}_4 = -\psi_2 - A_{34}\psi_3 - A_{44}\psi_4 + x_4,$$

where

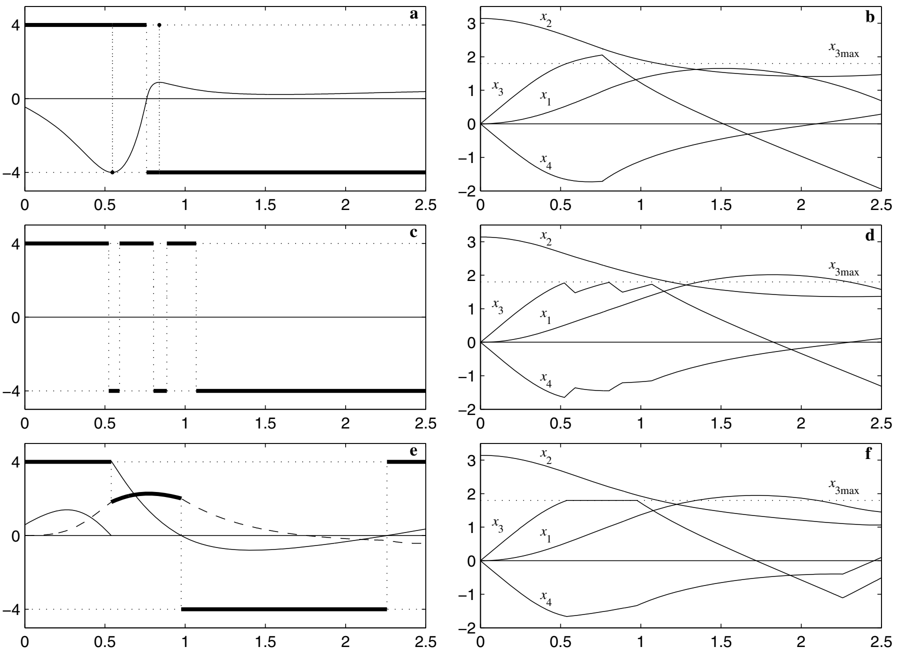


Fig. 1. Control (bold) vs. time in **a**, **c**, **e**; states vs. time in **b**, **d**, **f**; solid line in **a** and **e** represents the normalized switching function; dashed line in **e** denotes $P_{\text{con}}(x)$

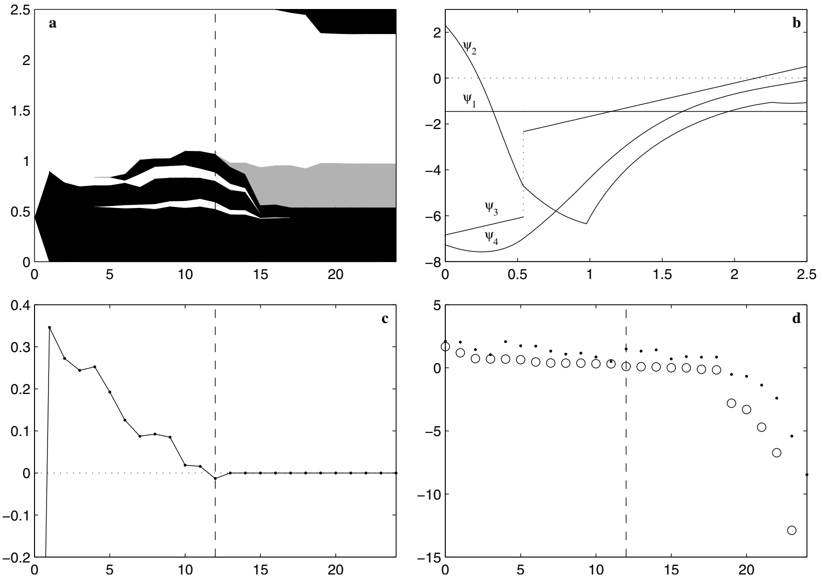


Fig. 2. Evolution of control structure (**a**); optimal adjoints (**b**); state constraint violation (**c**); norm of gradient and cost (**d**) (explanations in text)

$$A_{32} = \frac{\cos 2x_2 - x_4^2 \cos x_2 - f_3 \sin 2x_2}{2 - \cos^2 x_2}, \quad A_{34} = \frac{-2x_4 \sin x_2}{2 - \cos^2 x_2}$$

$$A_{42} = \frac{2\cos x_2 - u \sin x_2 - x_4^2 \cos 2x_2 - f_4 \sin 2x_2}{2 - \cos^2 x_2}, \quad A_{44} = A_{34} \cos x_2$$

$$f_3 = \frac{u - x_4^2 \sin x_2 + \sin x_2 \cos x_2}{2 - \cos^2 x_2}, \quad f_4 = \frac{(u - x_4^2 \sin x_2)\cos x_2 + 2\sin x_2}{2 - \cos^2 x_2}.$$

To determine the control procedure P_B, we proceed as in Section 4.1 of [5]. Note that no consistent alternatives to type B procedures are available as there are no control or state constraints, or singular arcs. We first augment the set of state equations given in Section 3 with the prototype adjoint equations

$$\dot{y}_1 = \beta_1 x_1, \quad \dot{y}_2 = -A_{32}y_3 - A_{42}y_4 + \beta_2 x_2$$

$$\dot{y}_3 = -y_1 + x_3, \quad \dot{y}_4 = -y_2 - A_{34}y_3 - A_{44}y_4 + x_4.$$

Next, we find the control maximizing the Hamiltonian H as a function of x and ψ, and substitute the prototype adjoint y for the original adjoint ψ. Hence

$$P_B(x, y) = \frac{y_3 + y_4 \cos x_2}{\alpha(2 - \cos^2 x_2)}.$$

In the cases where the evolution of control structure leads to more than one control arcs of type B, it has been observed that the convergence significantly slows down in the final phase of optimization. The reason for this effect might be that the optimization path then zigzags along a curved, narrow valley with steep slopes. A practical remedy is offered by the 'freezing' technique, which reduces the dimension of the decision space. When it is noticed that the optimization becomes slower, the best fitted arc of type B is selected for further optimization, and all the other arcs of type B are 'frozen', that is, their parameters and structural nodes (as far as possible) are kept constant. If the optimization potential of that arc is exhausted before reaching the optimum, e.g., a stationary point is achieved or the arc gets reduced, we return to the original algorithm, continue it for some time and then try again.

The freezing technique is illustrated by Fig. 3. Figures 3a and 3b show the evolution of control structure during the optimization. As in Fig. 1, the iteration numbers are on the horizontal axis, and the control time on the vertical axis. The white color represents $u(t) = 0$, and the grey colors, control arcs of type B. We start from a zero control. In the first iteration, a spike of type B is generated at $t \approx 0.3$ (see Fig. 3a). This arc (dark grey) evolves due to BFGS optimization in a constant decision space until iteration 17, when another spike of type B appears at $t \approx 1.1$ (light grey). They then evolve together until iteration 20, when the 'freezing' decision is taken. Surprisingly, the second, less developed arc proves better fitted and so the first arc of type B is frozen. The control at iteration 20 is shown in Fig. 3c (bold line) together with the function $P_B(x(t), \hat{\psi}(t))$ (dashed line). The corresponding state trajectories are plotted in Fig. 3d. The optimization is then continued for the next 200 iterations (Fig. 3b), until a satisfactory approximation of the optimal solution is obtained. Fig. 3e depicts the optimal control and Fig. 3f, the optimal state trajectories. The final control structure consists of only one arc of type B. As can be seen from the history of optimization in Fig. 3b, the second arc of type B has eventually ‘eaten up’ all other arcs.

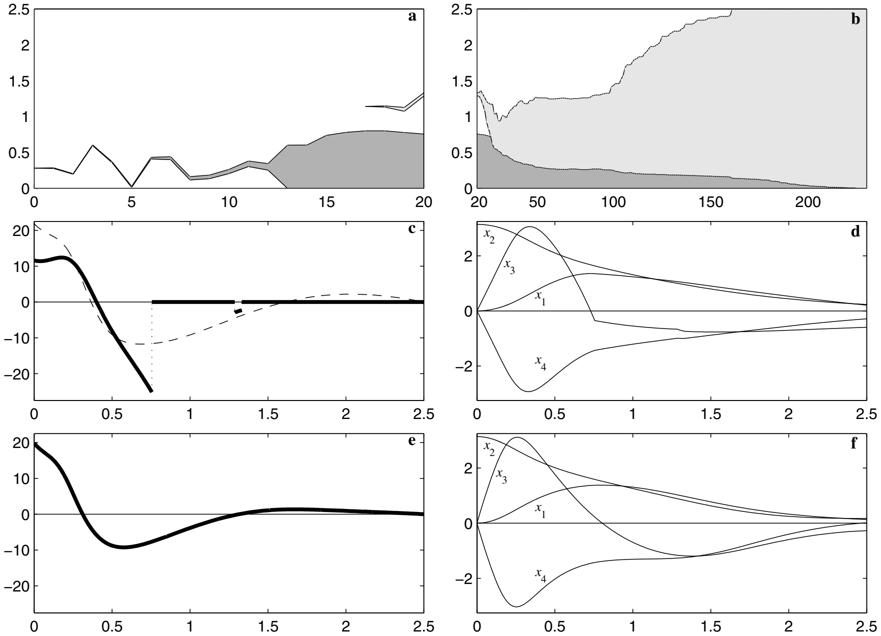


Fig. 3. Evolution of control structure (**a** and **b**); control (**c**) and states (**d**) at iteration 20; optimal control (**e**) and optimal state trajectories (**f**) (explanations in text)

5 Partial parameterization with prototype adjoints

Consider a bilinear control system

$$\dot{x} = u\,A_1 x + (1-u)\,A_2 x$$

$$A_1 = \begin{bmatrix} -1 & 1 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 2 \end{bmatrix}, \quad A_2 = \begin{bmatrix} -2 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & -1 & 1 \end{bmatrix}, \quad x(0) = \begin{bmatrix} -81 \\ -27 \\ -3 \end{bmatrix},$$

with bounded controls $0 \leq u \leq 1$, and a cost functional

$$Q(u) = \tfrac{1}{2} \int\limits_0^T (x_1^2 + x_2^2 + x_3^2)\, \mathrm{d}t, \quad T = 1.$$

We write the adjoint equations

$$\dot{\psi}_1 = (2 - u)\psi_1 + u\psi_2 + x_1, \quad \psi_1(T) = 0$$

$$\dot{\psi}_2 = -u\psi_1 + (2u - 1)\psi_2 + (1 - u)\psi_3 + x_2, \quad \psi_2(T) = 0$$

$$\dot{\psi}_3 = (1 - u)\psi_2 - (1 + u)\psi_3 + x_3, \quad \psi_3(T) = 0.$$

Equating the switching function to zero, we obtain the condition of singularity $\psi^\top (A_1 - A_2)\, x = 0$. As the singularity is of order one, this condition yields three equations for ψ and u. We solve them w.r.t. ψ_2 and u (ψ_1 is unnecessary in the sequel)

$$\psi_2 = \chi_2(x, \psi_3) = \frac{a_1(x) + a_2(x)\psi_3}{a_3(x)}, \quad u = P_B(x, \psi_3) = \frac{a_4(x) + a_5(x)\psi_3}{a_6(x) + a_7(x)\psi_3}\ .$$

The functions $a_1, \ldots, a_7$ are low-degree homogeneous polynomials. The augmented state equations, valid in control intervals of type B, read

$$\dot{x} = P_B(x, y)\, A_1 x + (1 - P_B(x, y))\, A_2 x$$

$$\dot{y} = (1 - P_B(x, y))\, \chi_2(x, y) - (1 + P_B(x, y))\, y + x_3.$$

In the MSE algorithm we define three control procedures, $P_{\min} = 0$, $P_{\max} = 1$, and $P_B(x, y)$. The computations are started from a control identically equal to one. The optimal control (see Fig. 4b) is obtained in 38 iterations. The evolution of control structure is shown in Fig. 4a. The black color represents $u(t) = 1$, white $u(t) = 0$, and grey $u(t) = P_B(x(t), y(t))$. Fig. 4c allows a comparison of the adjoint $\hat{\psi}$ and prototype adjoint y at t_{ini}, the initial time of the arc of type B. The values of gradient norm and cost are plotted in Fig. 4d (same conventions are used as in Fig. 2d).

6 MSE implementation

The MSE software is run within the MATLAB environment. The problem description is required from the user in a file form. It should use predefined objects and structures. The first version required C language procedures for faster ODE integration. In the current version, all computations are solely performed in MATLAB. Optimization in variable decision spaces relies on the computation of adjoint variables, for gradient and efficiency evaluations.

The implemented solvers include partitioned RK methods using formally adjoint pairs [3]: explicit RK4/RK4 or RK38/RK38 (in the fixed step mode), or a fully implicit 5th order, 3 stage Radau IIa/Radau Ia pair (in the variable step mode for stiff cases). This choice proves particularly advantageous

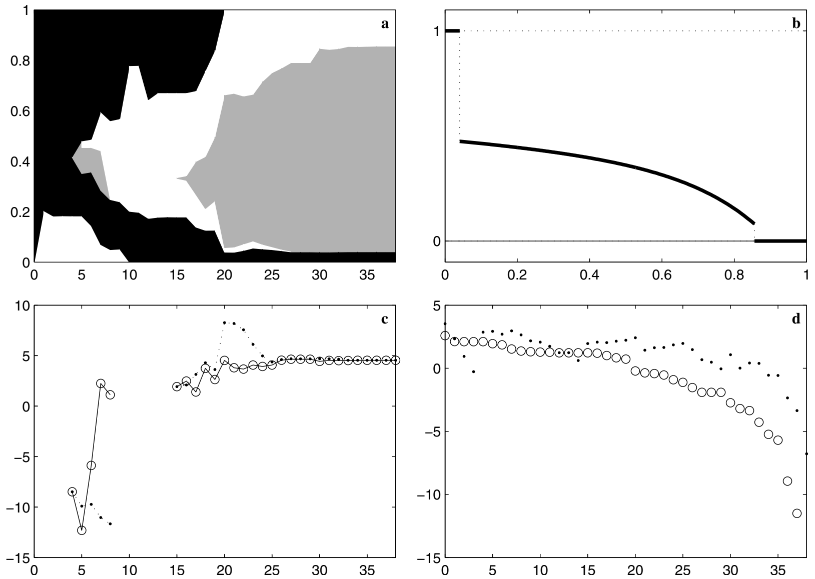


Fig. 4. Evolution of control structure (**a**); optimal control (**b**); $y(t_{\mathrm{ini}})$ (dots) and $\hat{\psi}_3(t_{\mathrm{ini}})$ (circles) in successive iterations (**c**); norm of gradient and cost (**d**)

both from the numerical and optimal control point of view [2]. In all cases interpolation is used to obtain a dense representation of trajectories. Similar concepts are employed in KPP v. 2.2 [6]. In the MSE, the mesh generation process always guarantees a proper treatment of discontinuities. The available optimizers include the BFGS method and the Newton method with curvilinear search, if Hessians are supplied. In both cases the constraints on structural nodes are always preserved and a simple active set strategy is employed. The current version of the MSE includes various types of approximative control procedures based on piecewise polynomial interpolation with fixed or moving nodes.

The forthcoming version of the MSE will have a more intuitive and friendly interface, and an explicit representation of discontinuous functions in data structures (including states and adjoints), with automatic mesh indexing and contiguous storage. As a further improvement we consider automatic differentiation for the generation of procedures computing adjoint variables, gradients and Hessians. We also plan a stand-alone version, dependent only on the freely distributed MATLAB run-time component.

The MSE as described here is a direct sequential method. Recently, a new variant has been initially tested [7], which goes along the ideas of simultaneous collocation [1]. It has been proved that for certain problems, especially with terminal constraints, the area of convergence of the new variant is larger.

7 Conclusions

The MSE equipped with a sufficiently rich stock of consistent control procedures can provide optimal solutions in a particularly attractive form, fully characterized by a relatively small number of parameters, and giving precise information on the optimal control structure. These features and the automatic handling of structural changes may prove advantageous in the design of NMPC control schemes.

It is often helpful to use approximative procedures in conjunction with the consistent ones. It may speed up the convergence, especially if the adjacent approximative and consistent control arcs are distinctly different. If they are similar, a ‘substitution effect’ sometimes occurs. As a result, the optimization becomes slower and nonconsistent procedures may persist in the final approximation of optimal solution. To prevent this effect, we have applied the freezing technique to approximative arcs.

An apparent drawback of the current version of the MSE, shared to some extent with typical indirect methods is the burden of analytical work needed in the preparatory stage of computations (mainly, analytical differentiation). However, the use of computer packages for symbolic calculations which become increasingly effective and popular, is an obvious remedy. Another direction for reducing this burden is in DAE formulations.

References

1. Biegler L T (2009), Efficient nonlinear programming algorithms for chemical process control and operations. In: Korytowski A, Malanowski K, Mitkowski W, Szymkat M (eds), System Modeling and Optimization, IFIP Adv Inf Comm Tech, 312, Springer Berlin Heidelberg, 21–35
2. Hager W (2000), Runge Kutta methods in optimal control and the transformed adjoint system. Numerische Mathematik, 87(2):247–282
3. Hairer E, Lubich Ch, Wanner G (2006), Geometric Numerical Integration. Comp Math, 31, 2nd ed, Springer Berlin Heidelberg
4. Hartl R F, Sethi S P, Vickson R G (1995), A survey of the maximum principles for optimal control problems with state constraints, SIAM Review, 17:181–218
5. Korytowski A, Szymkat M (2010), Consistent control procedures in the monotone structural evolution. Part 1: Theory. This volume
6. Miehe P, Sandu A (2006), Forward, tangent linear, and adjoint Runge Kutta methods in KPP-2.2. ICCS 2006, III, Alexandrov V N, Dimov I T, Karaivanova A, Tan C J K (eds), LNCS 3993, Springer Berlin Heidelberg, 120–127
7. Miller J (2009), Application of the collocation method to the Monotone Structural Evolution algorithm for bang-bang optimal control problems. 7th Conf Comp Meth Sys CMS09, Kraków, Poland
8. Szymkat M, Korytowski A, Turnau A (2000), Variable control parameterization for time-optimal problems. 8th IFAC Symp CACSD, Salford, UK
9. Szymkat M, Korytowski A (2008), The method of monotone structural evolution for dynamic optimization of switched systems. 47th IEEE CDC, Cancun, 1543–1550

Minimizing Tumor Volume for a Mathematical Model of Anti-Angiogenesis with Linear Pharmacokinetics

Urszula Ledzewicz[1], Helmut Maurer[2], and Heinz Schättler[3]

[1] Dept. of Mathematics and Statistics, Southern Illinois University Edwardsville, Edwardsville, Il, 62026-1653, USA, uledzew@siue.edu
[2] Institut für Numerische und Angewandte Mathematik, Rheinisch Westfälische Wilhelms Universität Münster, D-48149 Münster, Germany, maurer@math.uni-muenster.de
[3] Dept. of Electrical and Systems Engineering, Washington University, St. Louis, Mo, 63130-4899, USA, hms@wustl.edu

Summary. Optimal and suboptimal protocols are given for a mathematical model for tumor anti-angiogenesis. If a linear model for the pharmacokinetics of the anti-angiogenic agent is included in the modeling, optimal controls have chattering arcs, but excellent suboptimal approximations can be given.

1 Introduction

Tumor anti-angiogenesis is a rather novel cancer treatment approach that limits a tumor's growth by inhibiting it from developing the vascular network it needs for its further supply with nutrients and oxygen. Ideally, deprived of its sustenance, the tumor regresses. As with any novel approach, the underlying biological mechanisms are not fully understood and several important questions such as how to best schedule these therapies over time still need to be answered. Various anti-angiogenic agents have been and still are tested in clinical trials (e.g., [6, 11]). Naturally, the scope of these experiments is limited to simple structured protocols. Mathematical modeling and analysis can give valuable insights here into the structure of both optimal and suboptimal protocols and can thus become an important tool towards the overall aim of establishing robust and effective treatment protocols (e.g., [1, 9]).

Mathematically, the various protocols can be considered as control functions defined over time and the tools and techniques from optimal control theory are uniquely suited to address these difficult scheduling problems. In previous research, for various formulations of the dynamics underlying anti-angiogenic treatments that were based on a biologically validated model developed at Harvard Medical School [10] and one of its modifications formulated

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_23, © Springer-Verlag Berlin Heidelberg 2010

at the Cancer Research Institute at NIH [7], Ledzewicz et al. have considered the optimal control problem of how to schedule an a priori given amount of anti-angiogenic agents in order to minimize the tumor volume. Complete solutions in form of a regular synthesis of optimal controlled trajectories [2] (which specifies the optimal controls and their corresponding trajectories for arbitrary initial conditions) have been given for the two main models in [13, 17].

Because of the great complexity of the underlying biological processes, in these papers the dosages and concentrations of the anti-angiogenic agents have been identified, a commonly made first modeling simplification. In reality these clearly are different relations studied as *pharmacokinetics (PK)* in the medical and pharmaceutical literature. The standard and most commonly used model for PK is a simple model of exponential growth and decay given by

$$\dot{c} = -\rho c + u, \qquad c(0) = 0, \tag{1}$$

where u denotes the dosage of the agent and c its concentration. The coefficient ρ is the clearance rate and is related to the half-life of the agents. The important question is to what extent optimal controls will be altered under the addition of pharmacokinetic equations, both qualitatively and quantitatively. In models for chemotherapy which we had considered earlier optimal controls were bang-bang and this structure was retained if a linear pharmacokinetic model of the form (1) was added [14, 16]. Thus in this case no qualitative changes and in fact also only minor quantitative changes arose. But the solutions to the mathematical models for tumor anti-angiogenesis are characterized by optimal singular arcs which are defined by highly nonlinear relations (see below, [13, 17]) and now significant qualitative changes in the concatenation structure of optimal controls occur. They lead to the presence of optimal chattering arcs once a pharmacokinetic model (1) is included. In this paper we describe these changes for the mathematical model proposed by Ergun et al. [7] and give numerical results that show that the minimal tumor values can very accurately be approximated by reasonably simple, piecewise constant controls.

2 A Mathematical Model for Tumor Anti-Angiogenesis

We consider a mathematical model for tumor anti-angiogenesis that was formulated by Ergun, Camphausen and Wein in [7] and is a modification of the model by Hahnfeldt et al. from [10]. In both models the spatial aspects of the underlying consumption-diffusion processes that stimulate and inhibit angiogenesis are incorporated into a non-spatial 2-compartment model with the primary tumor volume p and its carrying capacity q as variables. The carrying capacity is mostly determined by the volume of endothelial cells that form the lining of the newly developing blood vessels and capillaries and we also call it the endothelial support. The tumor dynamics is modeled by a Gompertzian function,

$$\dot{p} = -\xi p \ln\left(\frac{p}{q}\right) \tag{2}$$

with ξ denoting a tumor growth parameter. The carrying capacity q is variable and in [7] its dynamics is modeled as

$$\dot{q} = bq^{\frac{2}{3}} - dq^{\frac{4}{3}} - \mu q - \gamma u q, \tag{3}$$

where b (birth) and d (death), respectively, are endogeneous stimulation and inhibition parameters for the endothelial support; the term μq represents natural death terms and $\gamma u q$ stands for additional exogenous inhibition. The variable u represents the control in the system and corresponds to the angiogenic dose rate while γ is a constant that represents the anti-angiogenic killing parameter. The particular inhibition and stimulation terms chosen in this model, $I(q) = dq^{\frac{4}{3}}$ and $S(q) = bq^{\frac{2}{3}}$, are a modification of the original terms in [10] in the sense that the tumor's stimulation of the carrying capacity now becomes proportional to the tumor radius, no longer its surface area. Also, compared with [10] the dynamics of the vascular support has been slowed down leading to an overall improved balance in the substitution of stimulation and inhibition (see [7]).

Anti-angiogenic agents are "biological drugs" that need to be grown in a lab and are very expensive and limited. From a practical point of view, it is therefore of importance how given amounts of these agents,

$$\int_0^T u(t)dt \leq y_{\max}, \tag{4}$$

can be administered to have "optimal" effect. Taking as objective to maximize the possible tumor reduction and adding an extra variable y that keeps track of the total amounts of agent that have been given, this problem takes the following form:

[C] for a free terminal time T, minimize the objective $J(u) = p(T)$ subject to the dynamics

$$\dot{p} = -\xi p \ln\left(\tfrac{p}{q}\right), \qquad p(0) = p_0, \tag{5}$$

$$\dot{q} = bq^{\frac{2}{3}} - dq^{\frac{4}{3}} - \mu q - \gamma u q, \qquad q(0) = q_0, \tag{6}$$

$$\dot{y} = u, \qquad y(0) = 0 \tag{7}$$

over all Lebesgue measurable functions $u : [0, T] \to [0, u_{\max}]$ for which the corresponding trajectory satisfies the end-point constraints $y(T) \leq y_{\max}$.

It is easily seen that for any admissible control u and arbitrary positive initial conditions p_0 and q_0 the solution (p, q, y) to the corresponding differential equation exists for all times $t > 0$ and both p and q remain positive [15]. Hence no state space constraints need to be imposed.

Necessary conditions for optimality are given by the Pontryagin Maximum Principle (e.g., see [3, 4]) and these conditions identify the constant controls $u = 0$ (no dose) and $u = u_{\max}$ (maximum dose), the so-called *bang controls*, as well as a time-varying feedback control, a so-called *singular control*, as candidates for optimality. Using Lie-algebraic calculations, analytic formulas for the singular control and the corresponding trajectory can be given.

Proposition 1. [15, 13] *The singular control is given in feedback form as*

$$u_{\sin}(q) = \psi(q) = \frac{1}{\gamma}\left(\frac{b - dq^{\frac{2}{3}}}{q^{\frac{1}{3}}} + 3\xi\frac{b + dq^{\frac{2}{3}}}{b - dq^{\frac{2}{3}}} - \mu\right). \tag{8}$$

This control is locally optimal if the state of the system lies on the corresponding singular arc S defined in (p, q)-space by

$$p_{\sin} = p_{\sin}(q) = q\exp\left(3\frac{b - dq^{\frac{2}{3}} - \mu q^{\frac{1}{3}}}{b + dq^{\frac{2}{3}}}\right). \tag{9}$$

This curve is an admissible trajectory (i.e., the singular control takes values between 0 *and* $u_{\max}$*) for $q_\ell^* \leq q \leq q_u^*$ where q_ℓ^* and q_u^* are the unique solutions to the equation $\psi(q) = a$ in $(0, \sqrt{\left(\frac{b}{d}\right)^3})$.*

Fig. 1 gives the graph of the singular control (8) on the left and the corresponding singular arc defined by (9) is shown on the right.

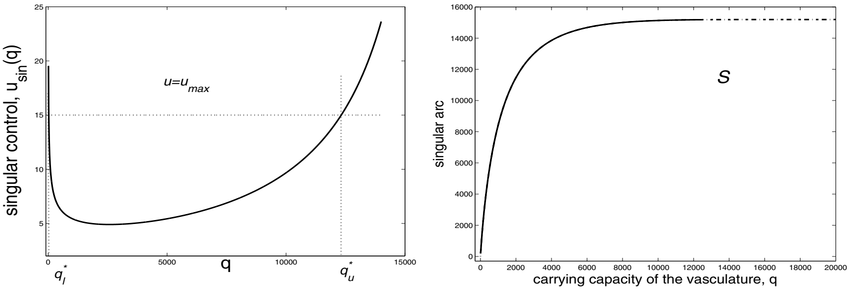


Fig. 1. Singular control (left) and singular arc, the corresponding trajectory (right)

Overall, optimal controls then need to be synthesized from bang and singular controls. This requires to analyze concatenations of these structures. Based on the formulas above a full synthesis of optimal controlled trajectories was given in [13]. Such a synthesis provides a complete "road map" of how optimal protocols look like depending on the initial condition in the problem, both qualitatively and quantitatively. Examples of projections of optimal trajectories into the (p, q)-space are given in Fig. 2. The admissible singular arc is shown as a solid curve which becomes dotted after the saturation point. Trajectories corresponding to $u \equiv u_{\max}$ are marked as thinner solid curves whereas trajectories corresponding to $u \equiv 0$ are marked as dashed-dotted curves. The dotted line on the graph is the diagonal, $p = q$. We highlighted with bold one specific, characteristic example of the synthesis. Initially the optimal control is given by a full dose $u = u_{\max}$ segment until the corresponding trajectory hits the singular arc. At that time the optimal control becomes singular following the singular arc until all inhibitors become exhausted. Since the singular arc lies in the region $p > q$, the tumor volume still shrinks along $u = 0$ until the trajectory reaches the diagonal $p = q$ at the final time T when the minimum tumor volume is realized. This structure $\mathbf{u}_{\max} - \mathbf{s} - \mathbf{0}$ is the most typical form of optimal controls. (For a more precise description of the synthesis, see [13])

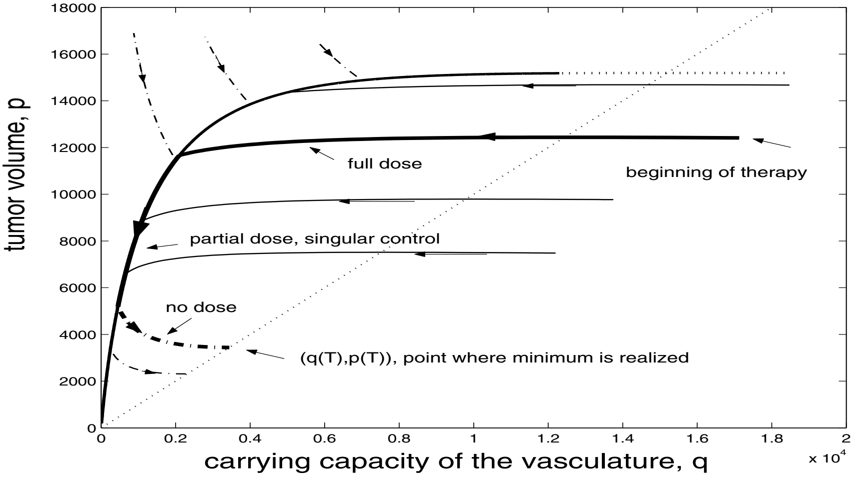


Fig. 2. Synthesis of optimal trajectories (vertical axis p, horizontal axis q)

3 Addition of a Pharmacokinetic Model

We now add the standard linear pharmacokinetic model (1) to the mathematical model and replace the control u in (6) by the concentration c, but otherwise preserve the same formulation. Thus the optimal control problem now becomes to minimize $p(T)$ subject to

$$\dot{p} = -\xi p \ln\left(\frac{p}{q}\right), \qquad p(0) = p_0, \tag{10}$$

$$\dot{q} = bq^{\frac{2}{3}} - dq^{\frac{4}{3}} - \mu q - \gamma c q, \qquad q(0) = q_0, \tag{11}$$

$$\dot{c} = -\rho c + u, \qquad c(0) = 0, \tag{12}$$

$$\dot{y} = u, \qquad y(0) = 0. \tag{13}$$

Once more the conditions of the Maximum Principle identify bang and singular controls as candidates. In [18] it is shown for a more general control-linear nonlinear system of a form that includes problem $[C]$ that the optimality status of a singular arc is preserved under the addition of a linear pharmacokinetic model. In fact, the very same equations that define the singular control and arc in the model without PK remain valid verbatim, albeit with a different interpretation. The singular curve (9) is preserved as a vertical surface in (p, q, c)-space and the singular arc is now defined as the intersection of this cylindrical surface with the graph of the function $c = \psi(q)$ defined by (8), see Fig. 3.

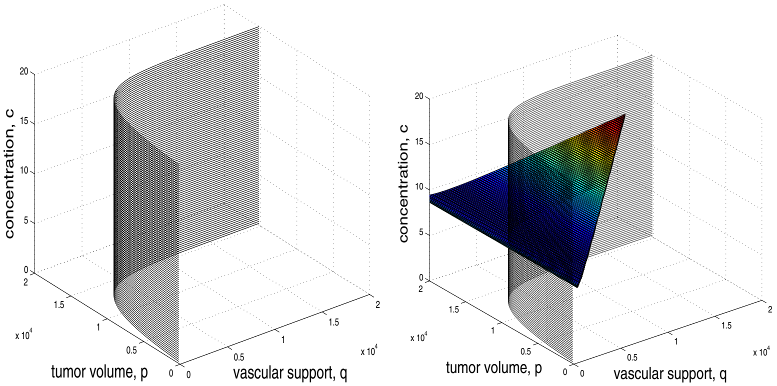


Fig. 3. Vertical singular surface in (p, q, c)-space (left) and intersection with the concentration $c = \psi(q)$ (right)

However, under the addition of pharmacokinetic equations of the form (1), the so-called order of the singular arc increases from 1 to 2. It is well-known that a smooth singular control with values in the interior of the control set for which the Kelley condition (a high order necessary condition for optimality of singular controls of order 2, [21]) is satisfied, cannot be concatenated optimally with either of the constant bang controls $u = 0$ or $u = u_{\max}$ [3, 20, 21]. These concatenations are now accomplished by means of *chattering controls*. This fact is also known as the Fuller phenomenon in the optimal control literature [8]. The structure of optimal controlled trajectories therefore clearly changes. The construction of an optimal synthesis that contains chattering arcs is quite a challenging task [21] and has not yet been completed for this model. However, practically the relevant question is what effect these changes actually have on the value of the objective. Chattering controls are not practical and thus the question about realizable suboptimal structures arises.

4 Suboptimal Approximations

In this section we give numerical results which show that it is possible to give simple suboptimal controls that achieve a tumor volume which gives an excellent approximation of the optimal value. In our calculations we use the following parameter values taken from [10] that are based on biologically validated data for the anti-angiogenic agent *angiostatin*: $\xi = 0.192$ *per day*, $b = 5.85$ *per day*, $d = 0.00873$ *per* mm^2 *per day*, $\gamma = 0.15$ *kg per mg of dose per day* with concentration in *mg of dose per kg*. For illustrative purposes we also chose a small positive value for μ, $\mu = 0.02$ *per day*. The upper limit of the dosage was taken as $u_{\max} = 15$ *mg per kg* and $y_{\max} = 60$ *mg per kg*. The half-life k of the agent is $k = 0.38$ *per day* [10]. The variables p and q are volumes measured in mm^3 and the initial conditions for our numerical calculations are $p_0 = 8,000\ mm^3$ and $q_0 = 10,000\ mm^3$.

The optimal control package NUDOCCCS due to Büskens [5] is implemented to compute a solution of the discretized control problem using nonlinear programming methods. We chose a time grid with $N = 400$ points and a high order Runge-Kutta integration method. Fig. 4 shows the graph of a numerically computed 'optimal' chattering control on the left and gives the corresponding concentration c on the right. The highly irregular structure of the control is caused by the fact that the theoretically optimal control chatters and has a singular middle segment. Due to numerical inaccuracies, as the intervals shrink to 0, the actual control values no longer are at their upper and lower values ±1. But the value of the objective is within the desired error tolerance. The final time is $T = 11.6406$ and the minimum tumor volume is given by $p(T) = 78.5326$.

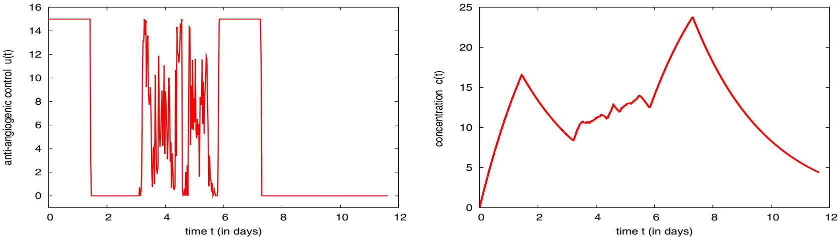


Fig. 4. A numerically computed 'optimal' chattering control (left) with corresponding concentration c (right)

For the same parameter values, Fig. 5 gives an example of a suboptimal control that is computed taking a control of the following simple bang-bang structure:

$$u(t) = \begin{cases} u_{\max} & \text{for } 0 \le t < t_1 \\ 0 & \text{for } t_1 \le t < t_2 \\ u_{\max} & \text{for } t_3 \le t < t_3 \\ 0 & \text{for } t_4 \le t \le T \end{cases} .$$

Thus both the chattering and singular arcs are completely eliminated at the expense of two adjacent bang-bang arcs that become larger. The switching times t_1, t_2, t_3 and the final time T are the free optimization variables. Using the arc-parametrization method developed in [19] and the code NUDOCCCS [5], we obtain the optimal switching times $t_1 = 1.78835$, $t_2 = 4.60461$, $t_3 = 6.86696$ and the final time $T = 11.3101$. This suboptimal control approximation gives the minimal tumor volume $p(T) = 78.8853$. It is surprising that this rather crude approximation of the chattering control gives a final tumor volume that is very close to the minimal tumor volume $p(T) = 78.5326$ for the "chattering control" in Fig. 4. On the right of Fig. 5 the corresponding concentration is shown.

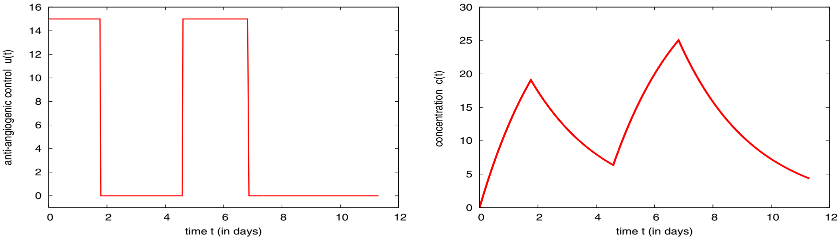


Fig. 5. A simple suboptimal bang-bang control with four arcs (left) and corresponding concentration c (right)

Tumor volumes that are close to identical with those realized by the optimal control can be achieved with a slightly more refined control structure given by

$$u(t) = \begin{cases} u_{\max} & \text{for } 0 \le t < t_1 \\ 0 & \text{for } t_1 \le t < t_2 \\ v & \text{for } t_2 \le t < t_3 \\ u_{\max} & \text{for } t_3 \le t < t_4 \\ 0 & \text{for } t_4 \le t \le T \end{cases} .$$

Again both chattering arcs are approximated by a simple bang-bang control that switches once from $u_{\max}$ to 0 and the singular segment is approximated by a constant control v over the full singular interval. This particular choice is probably the simplest reasonable approximation to the control structure that the theory predicts as optimal: a chattering control followed by a singular control and one more chattering control. Here the switching times t_i, $i = 1, \ldots, t_4$, the final time T, and the value v of the control are free optimization variables. Again, we use the arc-parametrization method [19] and the code NUDOCCCS [5] to compute the optimal switching times $t_1 = 1.46665$, $t_2 = 3.08056, t_3 = 5.98530$, $t_4 = 7.35795$, the final time $T = 11.6388$ and the constant control v is given by $v = 6.24784$. This gives the minimal tumor volume $p(T) = 78.5329$ for the suboptimal approximation which, for practical purposes, is identical with the minimal tumor volume $p(T) = 78.5326$ for the "chattering control" in Fig. 4. The computations also show that second order sufficient conditions for the underlying optimization problem are satisfied and hence we have found a strict (local) minimum. On the right of Fig. 6 the corresponding concentration is shown. Overall the behavior is very similar as in case of the chattering control, but the system has a much smoother and thus for many aspects preferable response. Like in the case of the problem when PK is not modeled [12], it appears that the differences in the minimum tumor volumes that can be achieved on the level of suboptimal controls are negligible.

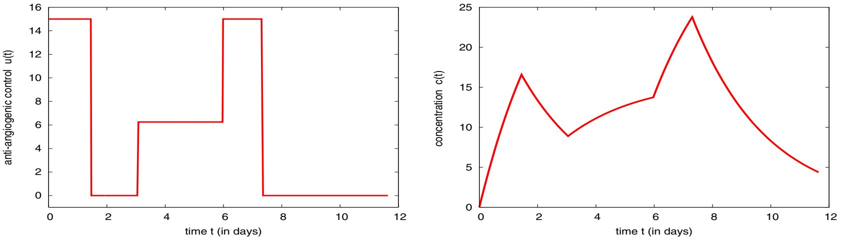


Fig. 6. A suboptimal piecewise constant control (left) and corresponding concentration c (right)

5 Conclusion

For a model of tumor anti-angiogenesis we have shown that, based on the structure of optimal controls, excellent simple suboptimal protocols can be constructed that realize tumor volumes close to the optimal ones. This holds both for the model without and with a linear pharmacokinetic model (1). Obviously, our numerical results are only for one special case, but we expect similar good approximations to be valid for a broad range of parameters. The significance of knowing the *optimal* solutions is bifold: it sets the benchmark to which suboptimal ones will be compared and it suggests the simpler structures for the approximating controls.

References

1. E. Afenya, Using mathematical modelling as resource in clinical trials, *Mathematical Biosciences and Engineering (MBE)*, **2**, no. 3, (2005), pp. 421-436
2. V.G. Boltyansky, Sufficient conditions for optimality and the justification of the dynamic programming method, *SIAM J. Control*, **4**, no. 2, 1966, pp. 326-361
3. B. Bonnard and M. Chyba, *Singular Trajectories and their Role in Control Theory*, Mathématiques & Applications, vol. 40, Springer Verlag, Paris, 2003

4. A. Bressan and B. Piccoli, *Introduction to the Mathematical Theory of Control*, American Institute of Mathematical Sciences, 2007
5. C. Büskens, Optimierungsmethoden und Sensitivitätsanalyse für optimale Steuerprozesse mit Steuer- und Zustands-Beschränkungen, Dissertation, Institut für Numerische Mathematik, Universität Münster, Germany, 1998
6. T.A. Drixler et al., Continuous administration of angiostatin inhibits accelerated growth of colorectal liver metastasies after partial hepatectomy, *Cancer Research*, **60**, (2000), pp. 1761-1765
7. A. Ergun, K. Camphausen and L.M. Wein, Optimal scheduling of radiotherapy and angiogenic inhibitors, *Bulletin of Mathematical Biology*, **65**, (2003), pp. 407-424
8. A.T. Fuller, Study of an optimum non-linear system, *J. Electronics Control*, **15**, (1963), pp. 63-71
9. A. Friedman, Cancer models and their mathematical anaysis, in: Tutorials in Mathematical Biosciences III, LN in Mathematics, vol. 1872, (2006), pp. 223-246
10. P. Hahnfeldt, D. Panigrahy, J. Folkman and L. Hlatky, Tumor development under angiogenic signalling: a dynamical theory of tumor growth, treatment response and postvascular dormancy, *Cancer Research*, **59**, (1999), pp. 4770-4775
11. O. Kisker et al., Continuous administration of endostatin by intraperitoneally implanted osmotic pump improves the efficacy and potency of therapy in a mouse xenograft tumor model, *Cancer Research*, **61**, (2001), pp. 7669-7674
12. U. Ledzewicz, J. Marriott, H. Maurer and H. Schättler, Realizable protocols for optimal administration of drugs in mathematical models for anti-angiogenic treatment, *Mathematical Medicine and Biology,* to appear
13. U. Ledzewicz, J. Munden, and H. Schättler, Scheduling of angiogenic inhibitors for Gompertzian and logistic tumor growth Models, *Discrete and Continuous Dynamical Systems, Series B*, **12**, (2009), pp. 415-438
14. U. Ledzewicz and H. Schättler, The influence of PK/PD on the structure of optimal control in cancer chemotherapy models, *Mathematical Biosciences and Engineering (MBE)*, **2**, no. 3, (2005), pp. 561-578
15. U. Ledzewicz and H. Schättler, A synthesis of optimal controls for a model of tumor growth under angiogenic inhibitors, *Proc. 44th IEEE Conference on Decision and Control*, Sevilla, Spain, (2005), pp. 945-950
16. U. Ledzewicz and H. Schättler, Optimal controls for a model with pharmacokinetics maximizing bone marrow in cancer chemotherapy, *Mathematical Biosciences*, **206**, (2007), pp. 320-342.
17. U. Ledzewicz and H. Schättler, Anti-Angiogenic therapy in cancer treatment as an optimal control problem, *SIAM J. Contr. Optim.*, **46**, (2007), pp. 1052-1079
18. U. Ledzewicz and H. Schaettler, Singular controls and chattering arcs in optimal control problems arising in biomedicine, *Control and Cybernetics*, **38** (4), (2009),
19. H. Maurer, C. Büskens, J.-H.R. Kim and C.Y. Kaya, Optimization methods for the verification of second order sufficient conditions for bang-bang controls, *Optimal Control Appliations and Methods*, **26**, (2005), pp. 129–156
20. J.P. McDanell and W.F. Powers, Necessary conditions for joining optimal singular and non-singular subarcs, *SIAM J. Control*, **9**, (1971), pp. 161–173
21. M.I. Zelikin and V.F. Borisov, *Theory of Chattering Control with Applications to Astronautics, Robotics, Economics and Engineering*, Birkhäuser, 1994

On Infinite Horizon Optimal Control of a Lotka-Voltera-System

Sabine Pickenhain

Brandenburg University of Technology Cottbus sabine@math.tu-cottbus.de

Summary. We describe a prey-predator model by a nonlinear optimal control problem with infinite horizon. This problem is non convex. Therefore we apply a duality theory developed in [17] with quadratic statements for the dual variables S. The essential idea is to use weighted Sobolev spaces as spaces for the states and to formulate the dual problem in topological dual spaces. We verify second order sufficient optimality condition to prove local optimality of the steady state in $[T, \infty)$.

1 Introduction

Control problems with infinite horizon have been investigated since the 1970s and became very important with regards to applications in economics, where an infinite horizon seems to be a very natural phenomenon, [2], [3],[4],[5], [6],[8],[11], [14], [15],[21],[22]. “The infinite horizon is an idealization of the fundamental point that the consequences of an investment are very long-lived; any short horizon requires some methods of evaluating end-of-period capital stock, and the only proper evaluation is their value in use the subsequent future”, (Arrow and Kurz (1970),[1]). Infinite horizon optimal control problems naturally arises not only in economics but also in natural sciences, like Biology. This can be explained by the fact that it is often unrealistic to assume the time is fixed or free. It is much more realistic to assume, that the termination time T of an admissible process is a random variable. Suppose, it is Poisson–distributed the conditional probability $P(T < t + \triangle t | T \geq t)$ satisfies the equation

$$P(T < t + \triangle t | T \geq t) = \rho \triangle t + o(\triangle t), \tag{1}$$

where $\rho > 0$ and $\frac{o(\triangle t)}{\triangle t} \to 0$ for $\triangle t \to 0$. By the definition of the conditional probability we have

$$\begin{aligned} P(T < t + \nabla t) &= P(T < t) + P(T < t + \triangle t | T \geq t) P(T \geq t) \\ &= P(T < t) + P(T < t + \triangle t | T \geq t)(1 - P(T < t)). \end{aligned}$$

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*, DOI 10.1007/978-3-642-12598-0_24, © Springer-Verlag Berlin Heidelberg 2010

Therefore, by (1) we obtain for the distribution function $\Phi(t) := P(T < t)$:

$$\Phi(t + \triangle t) = \Phi(t) + \rho(1 - \Phi(t))\triangle t + o(\triangle t) \tag{2}$$

and the function Φ satisfies the initial value problem

$$\dot{\Phi}(t) = \rho(1 - \Phi(t)), \quad \Phi(0) = 0. \tag{3}$$

Solving (3), we find the distribution function

$$\Phi(t) = 1 - e^{-\rho t}$$

with the density function $\varphi(t) := \rho e^{-\rho t}$. The objective in our optimal control problem is now to maximize (or minimize) the mathematical expectation of the random variable

$$J_T(x, u) = \int\limits_0^T f(t, x(t), u(t))\, dt, \tag{4}$$

$$J_\infty(x, u) = \rho \int\limits_0^\infty \left(\int\limits_0^T f(t, x(t), u(t))\, dt \right) e^{-\rho T}\, dT. \tag{5}$$

If we assume that

$$\lim_{T\to\infty} e^{-\rho T} \int\limits_0^T f(t, x(t), u(t))\, dt = 0, \tag{6}$$

then upon integrating (5) by parts, we obtain

$$J_\infty(x, u) = \int\limits_0^\infty e^{-\rho t} f(t, x(t), u(t))\, dt. \tag{7}$$

The purpose of this contribution is to illustrate the use of optimal control theory for infinite horizon problems, to obtain an optimal strategy for the control of a prey-predator system.

Following the paper [10], we use as a control the rate of release of predators or preys, which are bred in the laboratories. In the cited paper [10] the performance index is (4), where the *time T is specified or unspecified*, see ([10], p. 266). The final state is the steady state of the system.

In our consideration, the time is a Poisson-distributed random variable. We prove that the steady state is a local optimal solution, if the system achieves this point at any time T_0.

2 Problem formulation

We deal with problems of the following type $(P)_\infty$: Minimize the functional

$$J_\infty(x,u) = \int\limits_0^\infty f(t,x(t),u(t))\tilde{\nu}(t)\,dt \tag{8}$$

with respect to all

$$[\,x\,,\,u\,] \in W^{1,n}_{p,\nu}(0,\infty) \times L^r_\infty(0,\infty) \tag{9}$$

fulfilling the

$$\begin{aligned}
&\textit{State equations} && x'(t) = g(t,x(t),u(t)) \text{ a.e. on } (0,\infty), && (10)\\
&\textit{Control restrictions} && u(t) \in U \subseteq \mathrm{Comp}(\mathbb{R}^r) \quad \text{a.e. on } (0,\infty), && (11)\\
&\textit{State constraints} && x(t) \in G(t) \qquad \text{on } (0,\infty), && (12)\\
&\textit{Initial conditions} && x(0) = x_0. && (13)
\end{aligned}$$

The spaces $W^{1,n}_{p,\nu}(0,\infty)$ are weighted Sobolev spaces, see [13]. There application in control theory is shown in [20].

Let us note that in this paper all appearing integrals are understood as Lebesgue integrals and $\mathcal{A}_L$ consists of all processes (x,u), which make the Lebesgue integral in (8) convergent and satisfy the constraints (9) – (13).

Throughout the paper we assume that the data satisfy the following conditions:

1. The functions f,g are continuously differentiable in all arguments.
2. The control set U is assumed to be compact.
3. The functions ν and $\tilde{\nu}$ are weight functions in the sense of [13] explained below.
4. For all $(x,u) \in \mathcal{A}_L$ let

$$\lim_{T\to\infty} \tilde{\nu}(T) \int\limits_0^T f(t,x(t),u(t))\,dt = 0. \tag{14}$$

For the prey-predator model we have the following setting in this problem $(P)_\infty$: x_1 is the population of preys, x_2 is the population of predators. The state equations form a Lotka-Volterra-System ,

$$\dot{x_1}(t) = x_1(t)(\alpha_1 - \beta_1 x_2(t)), \tag{15}$$

$$\dot{x_2}(t) = x_2(t)(\beta_2 x_1(t) - \alpha_2 - b u_2(t)), \tag{16}$$

where the control variable u_2 is an insecticide which kills the predator only,

$$0 \leq u_2(t) \leq u_{max}.$$

In the mathematical model we normalize the constants,

$$\alpha_1 = \alpha_2 = \beta_1 = \beta_2 = u_{max} = 1.$$

The only steady state of the uncontrolled system is then $x_1 = x_2 = 1$, see [10]. Starting in this steady state $x_1(T_0) = x_2(T_0) = 1$, we ask for the optimal solution of the control problem $(P)_\infty$ with the performance index

$$J_\infty(x_1, x_2, u_2) : = \int\limits_{T_0}^{\infty} e^{-\rho t}(x_1(t) - cu_2(t))dt. \tag{17}$$

By the time shift $t := t - T_0$ we obtain the optimal control problem $(P)_\infty$,

$$J_\infty(x_1, x_2, u_2) := \int\limits_0^\infty e^{-\rho t}(x_1(t) - cu_2(t))dt,\ (0 < \rho < 1) \tag{18}$$

with respect to all

$$[\, x\, ,\ u\,] \in W^{1,2}_{p,\nu}(0, \infty) \times L_\infty(0, \infty) \tag{19}$$

fulfilling a.e. on $(0, \infty)$ the

$$\begin{aligned}
&\textit{State equations} & \dot{x_1}(t) &= x_1(t)(1 - x_2(t)), & (20)\\
& & \dot{x_2}(t) &= x_2(t)(x_1(t) - 1 - bu_2(t)), & (21)\\
&\textit{Control restrictions} & u(t) &\in U = [0, 1], & (22)\\
&\textit{State constraints} & x(t) &\in G(t) \qquad \text{on } (0, \infty), & (23)\\
&\textit{Initial conditions} & x_1(0) &= x_1(0) = 1. & (24)
\end{aligned}$$

3 Optimality Criteria

In the case of infinite horizon optimal control problems we can find several optimality criteria, which are adopted either to problems with Riemann integrals or to problems $(P)_\infty$ with Lebesgue integrals, see [6],[20].

Our considerations are focused on *global* and *strong local* optimality in the following sense:

Definition 1. *Let a process $(x, u) \in \mathcal{A}_L$ be given. We define*

$$\Delta_L(T) := L\text{-}\int_0^T f(t, x(t), u(t))\tilde{\nu}(t)\, dt - L\text{-}\int_0^T f(t, x^*(t), u^*(t))\tilde{\nu}(t)\, dt. \tag{25}$$

Then the pair $(x^*, u^*) \in \mathcal{A}_L$ is called global optimal for $(P)_\infty$ in the sense of

criterion L1, *if for any pair* $(x,u) \in \mathcal{A}_L$ *we have* $\lim\limits_{T\to\infty} \Delta_L(T) \geq 0$.

The pair $(x^*,u^*) \in \mathcal{A}_L$ *is called strong locally optimal for* $(P)_\infty$ *in the sense of*

criterion L1$_{sl}$, *if for an* $\epsilon > 0$ *any pair* $(x,u) \in \mathcal{A}_L$, *with* $\|x^* - x\|_{C(0,\infty)} < \epsilon$, *we have* $\lim\limits_{T\to\infty} \Delta_L(T) \geq 0$.

4 Duality in Weighted Sobolev Spaces

We consider Weighted Sobolev Spaces $W^{1,n}_{p,\nu}(\Omega)$ as subspaces of weighted $L^n_{p,\nu}(\Omega)$ spaces of those absolutely continuous functions x for which both x and its derivative $\dot{x}$ lie in $L^n_{p,\nu}(\Omega)$, see [13].

Let $\Omega = [0,\infty)$ and let $\mathcal{M}^n = \mathcal{M}(\Omega;\mathbb{R}^n)$ denote the space of Lebesgue measurable functions defined on Ω with values in $\mathbb{R}^n$. Let a *weight function* ν be given, i.e. ν is a function continuous on Ω, $0 < \nu(t) < \infty$, then we define the space $L^n_{p,\nu}(\Omega)$ with $p \geq 2$ by

$$L^n_{p,\nu}(\Omega) = \{x \in \mathcal{M}^n | \, \|x\|_p^p := \int\limits_\Omega |x(t)|^p \nu(t)\, dt < \infty\}, \tag{26}$$

for $p = \infty$

$$L^n_{\infty,\nu}(\Omega) = \{x \in \mathcal{M}^n | \, \|x\|_\infty := \operatorname*{ess\,sup}_{t\in\Omega} |x(t)|\nu(t) < \infty\} \tag{27}$$

and the weighted Sobolev space by

$$W^{1,n}_{p,\nu}(\Omega) = \{x \in \mathcal{M}^n | \, x \in L^n_{p,\nu}(\Omega), \dot{x} \in L^n_{p,\nu}(\Omega)\}. \tag{28}$$

Here $\dot{x}$ is the distributional derivative of x in the sense of [23], [p. 49]. This space, equipped with the norm

$$\|x\|^p_{W^{1,n}_{p,\nu}(\Omega)} = \int\limits_\Omega \{|x(t)| + |\dot{x}(t)|\}^p \nu(t) dt, \tag{29}$$

is a Banach space.

The following lemma, proved in [17], provides basic properties of functions in Weighted Sobolev spaces:

Lemma 1. *Let* $x^* \in W^{1,n}_{p,\nu}(\Omega)$ *with* $x^*(0) = x_0$ *and* $S : \Omega \times \mathbb{R}^n \to \mathbb{R}$ *be a function of the form*

$$S(t,\xi) = a(t) + \langle y(t), \xi - x^*(t)\rangle + \frac{1}{2}\langle Q(t)(\xi - x^*(t)), (\xi - x^*(t))\rangle, \tag{30}$$

having $a \in W^1_1(\Omega)$; $y \in W^{1,n}_{q,\nu^{1-q}}(\Omega)$ *and* $Q \in W^{1,n\times n}_{\infty,\nu^{-1}}(\Omega)$ *symmetric. Then, for any* $x \in W^{1,n}_{p,\nu}(\Omega)$ *with* $x(0) = x_0$, *it holds:*

$$\lim_{T\to\infty} S(T,x(T)) = 0, \tag{31}$$

$$\int\limits_0^\infty \frac{d}{dt} S(t,x(t))dt = -S(0,x_0). \tag{32}$$

We introduce the Hamiltonian as

$$\mathcal{H}(t,\xi,\eta) = \sup_{v\in U} H(t,\xi,v,\eta) \tag{33}$$

with

$$H(t,\xi,v,\eta) = -f(t,\xi,v) + \frac{1}{\tilde{\nu}(t)} < \eta, g(t,\xi,v) >, \tag{34}$$

where H represents the Pontrjagin function. Let

$$X := \{(t,\xi)\,|t \in (0,\infty),\, \xi \in G(t)\,\} \tag{35}$$

and

$$Y = \left\{ S : X \to \mathbb{R} \;\middle|\; \begin{array}{c} S(t,\xi) = a(t) + \langle y(t), \xi - x^*(t)\rangle \\ +\frac{1}{2}\left\langle Q(t)(\xi - x^*(t)), (\xi - x^*(t))\right\rangle \\ \\ a \in W_1^1(\overline{\Omega}),\ y \in W^{1,n}_{q,\nu^{1-q}}(\overline{\Omega}), \\ Q \in W^{1,n\times n}_{\infty,\nu^{-1}}(\Omega) \\ \\ \frac{1}{\tilde{\nu}(t)}\partial_t S(t,\xi) + \mathcal{H}(t,\xi,\partial_\xi S(t,\xi)) \le 0 \\ \forall (t,\xi) \in X \end{array} \right\}. \tag{36}$$

Using the scheme described in [12] we construct a dual problem $(D)_\infty$ and prove

Theorem 1. (Weak Duality) *Let a problem $(P)_\infty$ be given. Then the problem $(D)_\infty$:*

$$g_\infty(S) :=\ -S(0,x_0) \to \sup! \tag{37}$$

$$\text{with respect to S} \in Y, \tag{38}$$

is a dual problem to $(P)_\infty$, i.e. the weak duality relation

$$\inf(P)_\infty \ge \sup(D)_\infty \tag{39}$$

holds.

For the proof see [17]. The next two corollaries provide sufficiency conditions for optimality in the sense of criterion **L1** and criterion $\mathbf{L1}_{sl}$, respectively.

Corollary 1. (Sufficient optimality conditions, criterion L1):

Let $G(t) = \mathbb{R}^n$ (no state constraints). An admissible pair (x^*, u^*) is a global minimizer of $(P)^L_\infty$ (in the sense of criterion **L1***), if there exists an admissible S for $(D)_\infty$, $S \in Y$, such that the following conditions are fulfilled for almost all $t > 0$:*

$$(\mathbf{M})\ \mathcal{H}(t, x^*(t), \partial_\xi S(t, x^*(t))) = H(t, x^*(t), u^*(t), \partial_\xi S(t, x^*(t)))\ , \quad (40)$$

$$(\mathbf{HJ}) \quad \tfrac{1}{\nu(t)} \partial_t S(t, x^*(t)) + \mathcal{H}(t, x^*(t), \partial_\xi S(t, x^*(t))) = 0, \quad (41)$$

Proof: This follows immediately from the weak duality relation (39), the proof is given in [17],[20].

Conclusion 1. The boundary condition

$$(\mathbf{B}_\infty) \qquad \lim_{T\to\infty} S(T, x^*(T)) = 0 \quad (42)$$

is automatically satisfied due to Lemma 1.

Conclusion 2. Let now $G(t) = \mathcal{K}_\epsilon(x^*(t))$, $\epsilon > 0$,

$$\mathcal{K}_\epsilon(x^*(t)) := \{\xi \in \mathbb{R}^n \mid |\xi - x^*(t)| < \epsilon\}. \quad (43)$$

The corresponding sets X and Y from (35) and (36) are now denoted by X_ϵ and Y_ϵ.

Corollary 2. (Sufficient optimality conditions, criterion L1$_{sl}$)*:*

An admissible pair (x^*, u^*) is a strong local minimizer of $(P)^L_\infty$ (in the sense of criterion **L1**$_{sl}$*), if there exists an admissible S for $(D)^L_\infty$, $S \in Y_\epsilon$, such that the conditions* **(M)** *and* **(HJ)** *are fulfilled for almost all $t > 0$.*

5 Application to the Prey-Predator model

We want to prove that the steady-state of the uncontrolled system

$$(x_1^*, x_2^*, u_2^*) = (1, 1, 0)$$

is a strong local minimizer of $(P)_\infty$ in the sense of **criterion L1**$_{sl}$.

1. Die Pontrjagin-function of this problem is:

$$H(t, \xi, v, \eta) = (-\xi_1 - cv) + e^{\rho t}\left(\eta_1 \xi_1 (1 - \xi_2) - \eta_2 \xi_2 (1 - \xi_1 + bv)\right). \quad (44)$$

2. Assuming $u_2^* = 0$, the Hamiltonian $\mathcal{H}$ is twice continuously differentiable and is calculated by the condition **(M)**:

$$\mathcal{H}(t, \xi, v, \eta) = (-\xi_1) + e^{\rho t}\left(\eta_1 \xi_1 (1 - \xi_2) - \eta_2 \xi_2 (1 - \xi_1)\right). \quad (45)$$

3. Admissibility of S means that the Hamilton-Jacobi-Inequality

$$\Lambda(t,\xi) := \frac{1}{\tilde{\nu}(t)} S_t(t,\xi) + \mathcal{H}(t,\xi,y(t)) \leq 0 \tag{46}$$

has to be satisfied for all $\xi \in K_\epsilon(x^*(t))$, $t \in \Omega$.
4. The condition **(HJ)** means

$$\Lambda(t,x^*(t)) = 0 \quad \forall\, t \in \Omega. \tag{47}$$

5.(46) and (47) are satisfied, iff $x^*(t)$ solves the optimization problem

$$\Lambda(t,\xi) \longrightarrow \max! \quad \text{with respect to}\, \xi \in\ K_\epsilon(x^*(t)). \tag{48}$$

(48) is a parametric optimization problem. The following second order optimality conditions are necessary and sufficient for local optimality of $x^*(t)$.

$$\Lambda_\xi(t,x^*(t)) = 0. \tag{49}$$

$$\Lambda_{\xi\xi}(t,x^*(t)) \prec 0. \tag{50}$$

In [9] is shown, that

$$S(t,\xi) = a(t) + \langle y(t), \xi - x^*(t)\rangle + \frac{1}{2}\left\langle Q(t)(\xi - x^*(t)), (\xi - x^*(t))\right\rangle, \tag{51}$$

with

$$a(t) = 0, \qquad y_1(t) = -\frac{\rho}{1+\rho^2}e^{-\rho t}, \qquad y_2(t) = \frac{1}{1+\rho^2}e^{-\rho t}. \tag{52}$$

and the quadratic $2 \times 2 -$ matrix Q,

$$Q_{11}(t) = Q_{22} = Ae^{-\rho t}, Q_{12}(t) = Q_{21} = 0,\ A > \frac{1}{\rho(1-\rho)}$$

is admissible, $S \in Y\epsilon$, (with $\epsilon > 0$, independent of t) and solves (48).

Particularly the tuple $(x_1^*, x_2^*) = (1,1)$ belongs to the weighted Sobolev space,

$$x^* \in W^{1,2}_{2,\nu}(\Omega), \tilde{\nu}(t) = \nu(t) = e^{-\rho t},$$

and (a,y,Q) belong to corresponding dual spaces,

$$a \in W^1_1(\Omega),\ y \in W^{1,2}_{2,\nu^{-1}}(\Omega),\ Q \in W^{1,2\times 2}_{\infty,\nu^{-1}}(\Omega).$$

Finally, conditions **(M)** and **(HJ)** are fulfilled and Corollary 2 can be applied. Summarizing we have shown that $(x_1^*, x_2^*, u_2^*) = (1,1,0)$ is a strong local minimizer of $(P)_\infty$ in the sense of criterion $\mathbf{L1}_{sl}$.

6 Summary and Conclusions

The considered prey-predator model was described by a nonlinear optimal control problem with infinite horizon. The problem is non convex. Therefore it was necessary to apply the duality theory with quadratic statement for the dual variables S. The essential idea is to use weighted Sobolev spaces as spaces for the states and to formulate the dual problem in topological dual spaces. We verified second order sufficient optimality condition to prove local optimality of the steady state in $[T, \infty)$.

Pontryagins Maximum Principle was used to find candidates for the optimal solution which transfers the system from an arbitrary starting point into the steady state, [10]. Up to now it was not possible to verify sufficient optimality conditions as in [16] for this problem. Since this problem can not be solved analytically, numerical methods should be used to solve the dual problem.

References

1. Arrow, K.J.; Kurz, M., *Public Investment, the rate of Return, and the Optimal Fiscal Policy*, J. Hopkins Univ. Press, Baltimore, MD (1970).
2. Aseev, S.M., Kryazhimskii, A.V., Tarasyev, A. M., *The Pontryagin Maximum Principle and Transversality Conditions for a Class of Optimal Control Problems with Infinite Time Horizons* ,Proc. Steklov Inst. Math. ,233, 64-80(2001).
3. Aubin, J.P., Clarke, F.H., Shadow Prices and Duality for a Class of Optimal Control Problems, SIAM J. Conrol and Optimization, Vol.17, No.5, (1979).
4. Benveniste, L.M.; Scheinkman, J.A. (1982). Duality theory for dynamic optimization models of economics: the continuous time case. *Journal of Economic Theory* **27**, 1–19.
5. Blot, J.; Cartigny, P. Bounded solutions and oscillations of concave Lagrangian systems in presence of a discount rate. *Journal for Analysis and its Applications***14**, pp.731–750(1995).
6. Carlson, D.A., Haurie, A.B., Leizarowitz, A., *Infinite Horizon Optimal Control*, Springer-Verlag, New York, Berlin, Heidelberg (1991).
7. Dunford N., Schwartz, J. T., *Linear Operators. Part I: General Theory.* Wiley-Interscience; New York, etc.(1988).
8. Feichtinger, G.; Hartl, R. F, *Optimale Kontrolle ökonomischer Prozesse.* de Gruyter; Berlin - New York, (1986).
9. Golinko, A., *Optimale Steuerung eines Lotka-Volterra Systems mit unendlichem Zeithorizont*, Diplomarbeit, BTU Cottbus, (2009).
10. Goh, B. S.; Leitmann, G.; Vincent; T. L., *Optimal Control of a Prey -Predator System*, Mathematical Biosciences, 19, 263 – 286, (1974).
11. Halkin, H., Necessary conditions for optimal control problems with infinite horizons, Econometrica, **42**, 267 – 272 (1979).
12. Klötzler, R., On a general conception of duality in optimal control, In: *Equadiff IV. Proceedings of the Czechoslovak Conference on Differential Equations and*

their Applications held in Prague, August 22 – 26, 1977. (Fábera, J. (Ed.)), Lecture Notes in Mathematics **703**, 189 – 196, Springer, Berlin (1979).

13. Kufner, A. *Weighted Sobolev Spaces.* John Wiley & Sons; Chichester, etc.(1985).
14. Magill, M. J. P., Pricing infinite horizon programs. *J. Math. Anal. Appl.* **88**, 398 – 421(1982).
15. Michel, P., On the Transversality Condition in Infinite Horizon Optimal Problems, Econometrica, Vol.50, No.4, July, 1982.
16. Pickenhain, S., Tammer, K., Sufficient conditions for local optimality in multidimensional control problems with state restrictions, *Z. Anal. Anw.* **10** , 397 – 405 (1991).
17. Pickenhain, S., Lykina, V., Sufficiency conditions for infinite horizon optimal control problems. In: *Recent Advances in Optimization.* (Seeger, A. (Ed.)), (*Lecture Notes in Economics and Mathematical Systems* **563**), 217 – 232, Springer, Berlin, etc.,(2006).
18. Pickenhain, S.; Lykina, V.; Wagner, M. Lebesgue and improper Riemann integrals in infinite horizon optimal control problems. Control and Cybernet. 37, 451 – 468(2006).
19. Pickenhain, S.; Lykina, V.; Wagner, M., *On the lower semi continuity of functionals involving Lebesgue or improper Riemann integrals in infinite horizon optimal control problems.* Control and Cybernet. 37, 451 – 468, (2008).
20. Pickenhain, S., *On adequate transversality conditions for infinite horizon optimal control problems – a famous example of Halkin.*, Dynamic Systems, Economic Growth, and the Environment, Ed.: Cuaresma, J.C.; Tarasyev, A.; Palokangas, T., Springer, Heidelberg Dordrecht London New York, 3 – 21, (2009).
21. Rockafellar, R.T., Convex Processes and Hamilton Dynamical Systems, Convex Analysis and Mathematical Economics, 1978.
22. Sethi, S. P.; Thompson, G. L.,*Optimal Control Theory. Applications to Management Science and Economics.* Kluwer; Boston - Dordrecht - London, 2nd ed. (1985).
23. Yosida,K., Functional Analysis, Springer-Verlag, New York (1974).

Part V

Model Predictive Control

Performance of NMPC Schemes without Stabilizing Terminal Constraints

Nils Altmüller, Lars Grüne, and Karl Worthmann

Mathematical Institute, University of Bayreuth, 95440 Bayreuth, Germany
nils.altmueller, lars.gruene, karl.worthmann@uni-bayreuth.de

Summary. In this paper we investigate the performance of unconstrained nonlinear model predictive control (NMPC) schemes, i.e., schemes in which no additional terminal constraints or terminal costs are added to the finite horizon problem in order to enforce stability properties. The contribution of this paper is twofold: on the one hand in Section 3 we give a concise summary of recent results from [7, 3, 4] in a simplified setting. On the other hand, in Section 4 we present a numerical case study for a control system governed by a semilinear parabolic PDE which illustrates how our theoretical results can be used in order to explain the differences in the performance of NMPC schemes for distributed and boundary control.

1 Introduction

Model predictive control (MPC) is a well established method for approximating the optimal control of linear and nonlinear systems [1, 8, 9]. MPC approximates the optimal solutions of in general computationally intractable infinite horizon optimal control problems by the iterative solution of finite horizon problems, the so called receding horizon strategy. This interpretation of MPC immediately leads to the question of how good the performance of the MPC scheme is compared to the original infinite horizon optimization criterion. Since infinite horizon problems are often formulated in order to obtain stabilizing feedback laws, another important question is whether the resulting MPC feedback law will still stabilize the system.

In this paper we investigate these issues for so called unconstrained nonlinear MPC (NMPC) schemes. Here *unconstrained* refers to those terminal constraints or terminal costs which are added to the finite horizon problem in order to enforce stability properties; other constraints like, e.g., state and control constraints motivated by physical considerations can easily be included in our analysis although for simplicity of exposition we do not elaborate on this aspect in this paper and refer to, e.g., [9] for an extensive treatment of feasibility issues. Such unconstrained schemes are appealing in many ways, cf. the discussion at the end of the introductory Section 2.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_25, © Springer-Verlag Berlin Heidelberg 2010

The contribution of this paper is twofold: on the one hand in Section 3 we give a concise summary of recent results from [3, 4, 7] in a simplified setting, restricting the reasoning to the special case of exponential controllability and classical NMPC feedback laws. For an extended setting including networked control systems, finite time controllability and additional weights in the cost functional we refer to [3, 4] and [5]. On the other hand, in Section 4 we present a numerical case study for a control system governed by a semilinear parabolic PDE. This case study illustrates how our theoretical results can be used in order to explain the differences in the performance of NMPC schemes for distributed and boundary control.

2 Setup and Preliminaries

We consider a nonlinear discrete time control system given by

$$x(n+1) = f(x(n), u(n)), \quad x(0) = x_0 \tag{1}$$

with $x(n) \in X$ and $u(n) \in U$ for $n \in \mathbb{N}_0$. Here the state space X and the control value space U are arbitrary metric spaces with metrics denoted by $d(\cdot,\cdot)$. We denote the space of control sequences $u : \mathbb{N}_0 \to U$ by $\mathcal{U}$ and the solution trajectory for given $u \in \mathcal{U}$ by $x_u(\cdot)$. State and control constraints can be incorporated by replacing X and U by appropriate subsets of the respective spaces, however, for brevity of exposition we will not address feasibility issues in this paper.

A typical class of such discrete time systems are sampled- data systems induced by a controlled — finite or infinite dimensional — differential equation with sampling period $T > 0$ where the discrete time control value $u(n)$ corresponds to the constant control value $u_c(t)$ applied in the sampling interval $[nT, (n+1)T)$.

Our goal is to minimize the infinite horizon cost functional $J_\infty(x_0, u) = \sum_{n=0}^{\infty} \ell(x_u(n), u(n))$ with running cost $\ell : X \times U \to \mathbb{R}_0^+$ by a static state feedback control law $\mu : X \to U$ which is applied according to the rule

$$x_\mu(0) = x_0, \quad x_\mu(n+1) = f(x_\mu(n), \mu(x_\mu(n))). \tag{2}$$

We denote the optimal value function for this problem by $V_\infty(x_0) := \inf_{u \in \mathcal{U}} J_\infty(x_0, u)$. The motivation for this problem stems from stabilizing the system (1) at a fixed point, i.e., at a point $x^\star \in X$ for which there exists a control value $u^\star \in U$ with $f(x^\star, u^\star) = x^\star$ and $\ell(x^\star, u^\star) = 0$. Under mild conditions on ℓ it is known that the optimal feedback for J_∞ indeed asymptotically stabilizes the system with V_∞ as a Lyapunov function, see, e.g., [6].

Since infinite horizon optimal control problems are in general computationally infeasible, we use a receding horizon NMPC method in order to compute an approximately optimal feedback law. To this end, we consider the finite horizon functional

$$J_N(x_0,u) = \sum_{n=0}^{N-1} \ell(x_u(n), u(n)) \tag{3}$$

with *optimization horizon* $N \in \mathbb{N}_{\geq 2}$ and optimal value function $V_N(x_0) := \inf_{u \in \mathcal{U}} J_N(x_0,u)$. By minimizing (3) over $u \in \mathcal{U}$ we obtain an optimal control sequence[1] $u^\star(0), u^\star(1), \ldots, u^\star(N-1)$ depending on the initial value x_0. Implementing the first element of this sequence, i.e., $u^\star(0)$, yields a new state $x_{u^\star}(1,x_0)$ for which we redo the procedure, i.e., at the next time instant we minimize (3) for $x_0 := x_{u^\star}(1,x_0)$. Iterative application of this procedure provides a control sequence on the infinite time interval. A corresponding closed loop representation of the type (2) is obtained as follows.

Definition 1. *For $N \geq 2$ we define the MPC feedback law $\mu_N(x_0) := u^\star(0)$, where $u^\star$ is a minimizing control for* (3) *with initial value x_0.*

In many papers in the (N)MPC literature additional stabilizing terminal constraints or terminal costs are added to the optimization objective (3) in order to ensure asymptotic stability of the NMPC closed loop despite the truncation of the horizon (see, e.g., the monograph [9] for a recent account of this theory). In contrast to this approach, here we investigate (3) without any changes. This is motivated by the fact that this "plain" NMPC scheme is the most easy one to implement and appears to be predominant in practical applications, cf. [8]. Another reason appears when looking at the infinite horizon performance of the NMPC feedback law μ_N given by $J_\infty(x_0,\mu_N) := \sum_{n=0}^{\infty} l(x_{\mu_N}(n), \mu_N(x_{\mu_N}(n)))$. As we will see, under a suitable controllability condition for NMPC without stabilizing constraints we can establish an upper bound for this value in terms of the optimal value function $V_\infty(x_0)$, which is in general not possible for schemes with stabilizing constraints.

3 Performance and stability analysis

In this section we summarize the main steps of the stability and suboptimality analysis of unconstrained NMPC schemes from [3, 4, 7] in a simplified setting. The cornerstone of our analysis is the following proposition which uses ideas from relaxed dynamic programming.

Proposition 1. *Assume there exists $\alpha \in (0,1]$ such that for all $x \in X$ the inequality*

$$V_N(x) \geq V_N(f(x,\mu_N(x))) + \alpha \ell(x,\mu_N(x)) \tag{4}$$

holds. Then for all $x \in X$ the estimate

[1] For simplicity of exposition we assume that a minimizing control sequence $u^\star$ exists for (3). However, given that in this abstract formulation $\mathcal{U}$ may be infinite dimensional we do not assume uniqueness of $u^\star$.

$$\alpha V_\infty(x) \leq \alpha J_\infty(x, \mu_N) \leq V_N(x) \leq V_\infty(x) \tag{5}$$

holds. If, in addition, there exist $x^\star \in X$ and $\mathcal{K}_\infty$-functions[2] α_1, α_2 *such that the inequalities*

$$\ell^\star(x) := \min_{u \in U} \ell(x, u) \geq \alpha_1(d(x, x^\star)) \quad \textit{and} \quad V_N(x) \leq \alpha_2(d(x, x^\star)) \tag{6}$$

hold for all $x \in X$, then $x^\star$ is a globally asymptotically stable equilibrium for (2) *with $\mu = \mu_N$ with Lyapunov function V_N.*

Proof. See [7, Prop. 2.2] or [3, Prop. 2.4] and [3, Theorem 5.2]. □

In order to compute α in (4) we use the following controllability property: we call the system (1) *exponentially controllable* with respect to the running cost ℓ if there exist constants $C \geq 1$ (overshoot bound) and $\sigma \in [0, 1)$ (decay rate) such that

$$\begin{aligned} &\text{for each } x \in X \text{ there exists } u_x \in \mathcal{U} \text{ with} \\ &\ell(x_{u_x}(n, x), u_x(n)) \leq C\sigma^n \ell^\star(x) \ \text{ for all } \ n \in \mathbb{N}_0. \end{aligned} \tag{7}$$

This condition implies

$$V_N(x) \leq J_N(x, u_x) \leq \sum_{n=0}^{N-1} C\sigma^n \ell^\star(x) = C\frac{1 - \sigma^N}{1 - \sigma}\ell^\star(x) =: B_N(\ell^\star(x)). \tag{8}$$

Hence, in particular (6) follows for $\alpha_2 = B_N \circ \alpha_3$ if the inequality

$$\alpha_1(d(x, x^\star)) \leq \ell^\star(x) \leq \alpha_3(d(x, x^\star)) \tag{9}$$

holds for some $\alpha_1, \alpha_3 \in \mathcal{K}_\infty$ and all $x \in X$. Now consider an arbitrary $x \in X$ and let $u^\star \in \mathcal{U}$ be an optimal control for $J_N(x, u)$, i.e., $J_N(x, u^\star) = V_N(x)$. Note that by definition of μ_N the identity $x_{u^\star}(1, x) = f(x, \mu_N(x))$ follows.

For the following lemma we abbreviate

$$\lambda_n = \ell(x_{u^\star}(n, x), u^\star(n)), \ \ n = 0, \ldots, N-1 \quad \text{and} \quad \nu = V_N(x_{u^\star}(1, x)). \tag{10}$$

Lemma 1. *Assume* (7) *holds. Then the inequalities*

$$\sum_{n=k}^{N-1} \lambda_n \leq B_{N-k}(\lambda_k) \quad \textit{and} \quad \nu \leq \sum_{n=0}^{j-1} \lambda_{n+1} + B_{N-j}(\lambda_{j+1}) \tag{11}$$

hold for $k = 0, \ldots, N-2$ and $j = 0, \ldots, N-2$.

Proof. See [3, Section 3 and Proposition 4.1]. □

[2] A continuous function $\alpha : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ is said to be of class $\mathcal{K}_\infty$ if it is strictly increasing and unbounded with $\alpha(0) = 0$.

The inequalities from Lemma 1 now lead to the following theorem.

Theorem 1. *Assume that the system* (1) *and ℓ satisfy the controllability condition* (7)*. Then inequality* (4) *holds for all $x \in X$ with*

$$\alpha = \min_{\lambda_0,\ldots,\lambda_{N-1},\nu} \sum_{n=0}^{N-1} \lambda_n - \nu \tag{12}$$

subject to the constraints (11)*, $\lambda_0 = 1$ and $\lambda_1,\ldots,\lambda_{N-1},\nu \geq 0$.*

Proof. See [3, Section 4]. □

The consequence of this theorem for the performance of the NMPC closed loop, i.e., (2) with $\mu = \mu_N$, is as follows: if (1) and ℓ satisfy (7) and (9), then global asymptotic stability and the suboptimality estimate (5) are guaranteed whenever α from (12) is positive. In fact, regarding stability we can show more: the construction of an explicit example yields that whenever α from (12) is negative, then there is a system (1) and an ℓ which satisfy (7) and (9) but for which (2) with $\mu = \mu_N$ is not asymptotically stable, cf. [3, Theorem 5.3].

The key observation for computing an explicit expression for α in (4) is that the linear program in Theorem 1 can be solved explicitly.

Theorem 2. *Under the assumptions of Theorem 1 the value α from* (12) *is given by*

$$\alpha = 1 - \frac{(\gamma_N - 1)\prod_{i=2}^{N}(\gamma_i - 1)}{\prod_{i=2}^{N}\gamma_i - \prod_{i=2}^{N}(\gamma_i - 1)} \quad \textit{with} \quad \gamma_i = C\frac{1-\sigma^i}{1-\sigma}. \tag{13}$$

Proof. See [4, Theorem 5.3]. □

The explicit formula thus derived for α allows us to visualize the impact of the parameters C, σ in (7) on the value of α in (4). As an example, Figure 1 shows the regions in the C,σ-plane for which $\alpha > 0$ and thus asymptotic stability holds for optimization horizons $N = 2, 4, 8$, and 16. Note that since α is increasing in N the stability region for N is always contained in the stability region for all $\widetilde{N} > N$.

Figure 1 clearly shows the different roles of the parameters C and σ in (7): While for fixed C the minimal stabilizing N for varying σ is usually larger than 2, for fixed σ it is always possible to achieve stability with $N = 2$ by reducing C. Thus, the overshoot bound C plays a decisive role for the stability and performance of NMPC schemes.

An important observation in this context is that C and σ do not only depend on the control system but also on the running cost ℓ. Hence, ℓ can be used as a design parameter in order to "tune" C and σ with the goal to obtain good closed loop performance with small control horizons N by reducing C as much as possible. For examples see, e.g., [3] and [2] and the following section in which we will illustrate and explain this procedure for a semilinear parabolic PDE control system.

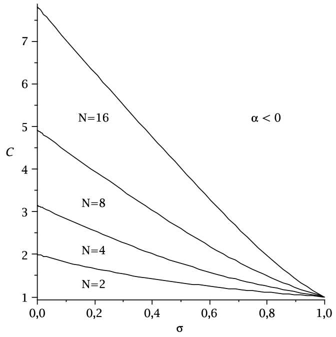


Fig. 1. Stability regions for various optimization horizons N depending on C and σ from (7)

4 A numerical case study

In practice, for many complex control systems and associated running cost functions ℓ it is difficult if not impossible to exactly determine the constants C and σ. However, by means of a controlled semilinear parabolic PDE, in this section we demonstrate that an exact computation of these constants is not necessarily needed in order to understand differences in the NMPC closed loop behavior for different running costs ℓ.

The first model we are considering is the semilinear parabolic PDE

$$y_t(t,x) = \nu y_{xx}(t,x) - y_x(t,x) + \mu\left(y(t,x) - y(t,x)^3\right) + u(t,x) \qquad (14)$$

with distributed control $u \in L^\infty(\mathbb{R}\times\Omega,\mathbb{R})$ and $\Omega = (0,1)$ and real parameters $\nu = 0.1$, $\mu = 10$. Here y_t and y_x denote the partial derivatives with respect to t and x, respectively and y_{xx} denotes the second partial derivative with respect to x.

The solution y of (14) is supposed to be continuous in $\overline{\Omega}$ and to satisfy the boundary and initial conditions

$$y(t,0) = 0,\ y(t,1) = 0 \text{ for all } t \geq 0 \text{ and } y(0,x) = y_0(x) \text{ for all } x \in \Omega \qquad (15)$$

for some given continuous function $y_0 : \overline{\Omega} \to \mathbb{R}$ with $y_0(0) = y_0(1) = 0$.

Observe that we have changed notation here in order to be consistent with the usual PDE notation: $x \in \Omega$ is the independent space variable while the unknown function $y(t,\cdot) : \Omega \to \mathbb{R}$ in (14) is the state now. Hence, the state is now denoted by y (instead of x) and the state space of this PDE control system is a function space, more precisely the Sobolev space $H_0^1(\Omega)$, although the specific form of this space is not crucial for the subsequent reasoning.

Figure 2 shows the solution of the uncontrolled system (14), (15), i.e., with $u \equiv 0$. For growing t the solution approaches an asymptotically stable steady state $y^{**} \neq 0$. The figure (as well as all other figures in this section) was computed numerically using a finite difference scheme with 50 equidistant nodes on $(0,1)$ (finer resolutions did not yield significantly different results) and initial value y_0 with $y_0(0) = y_0(1) = 0$, $y_0|_{[0.02,0.3]} \equiv -0.1$, $y_0|_{[0.32,0.98]} \equiv 0.1$ and linear interpolation in between.

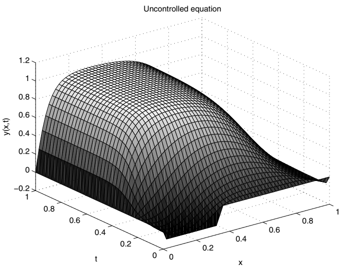


Fig. 2. Solution $y(t,x)$ of (14), (15) with $u \equiv 0$.

By symmetry of (14) the function $-y^{**}$ must be an asymptotically stable steady state, too. Furthermore, from (14) it is obvious that $y^* \equiv 0$ is another steady state, which is, however, unstable. Our goal is now to use NMPC in order to stabilize the unstable equilibrium $y^* \equiv 0$.

To this end we consider the sampled-data system corresponding to (14) with sampling period $T = 0.025$ and denote the state of the sampled-data system at the n-th sampling instant, i.e., at time nT by $y(n,\cdot)$. For penalizing the distance of the state $y(n,\cdot)$ to $y^* \equiv 0$ a popular choice in the literature is the L^2-functional

$$\ell(y(n,\cdot),u(n,\cdot)) = \|y(n,\cdot)\|^2_{L^2(\Omega)} + \lambda\|u(n,\cdot)\|^2_{L^2(\Omega)} \tag{16}$$

with $\lambda = 0.1$ which penalizes the mean squared distance from $y(n,\cdot)$ to $y^* \equiv 0$.

Another possible choice of measuring the distance to $y^* \equiv 0$ is obtained by using the H^1 norm for $y(n,\cdot)$ in ℓ, i.e,

$$\ell(y(n,\cdot),u(n,\cdot)) = \|y(n,\cdot)\|^2_{L^2(\Omega)} + \|y_x(n,\cdot)\|^2_{L^2(\Omega)} + \lambda\|u(n,\cdot)\|^2_{L^2(\Omega)}, \tag{17}$$

which in addition to the L^2 distance (16) also penalizes the mean squared distance from $y_x(n,\cdot)$ to $y^*_x \equiv 0$. Figures 3 and 4 show the respective NMPC closed loop solutions with optimization horizons $N = 3$ and $N = 11$.[3]

Figure 3 indicates that for $N = 3$ the NMPC scheme with ℓ from (16) does not stabilize the system at $y^* \equiv 0$ while for ℓ from (17) it does. For (16) we need an optimization horizon of at least $N = 11$ in order to obtain a stable

[3] The computations were performed with PCC, `http://www.nonlinearmpc.com/`

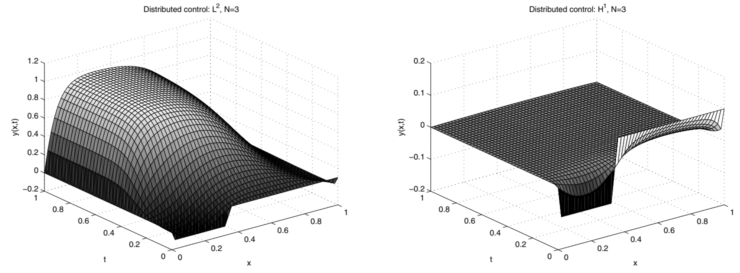


Fig. 3. NMPC closed loop for (14) with $N = 3$ and ℓ from (16)(left) and (17)(right)

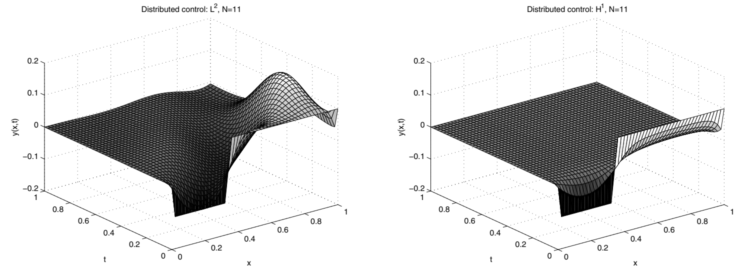


Fig. 4. NMPC closed loop for (14) with $N = 11$ and ℓ from (16)(left) and (17)(right)

closed loop solution, cf. Figure 4. For ℓ from (17) the right images in Figure 3 and 4 show that enlarging the horizon does not improve the solution.

Using our theoretical results we can explain why ℓ from (17) performs much better for small horizons N. For this example our controllability condition (7) reads

$$\ell(y(n,\cdot), u(n,\cdot)) \leq C\sigma^n \ell^\star(y(0,\cdot)). \tag{18}$$

For ℓ from (16) this becomes

$$\|y(n,\cdot)\|^2_{L^2(\Omega)} + \lambda \|u(n,\cdot)\|^2_{L^2(\Omega)} \leq C\sigma^n \|y(0,\cdot)\|^2_{L^2(\Omega)}. \tag{19}$$

Now in order to control the system to $y^* \equiv 0$, in (14) the control needs to compensate for y_x and $\mu\left(y(t,x) - y(t,x)^3\right)$, i.e., any control steering $y(n,\cdot)$ to 0 must satisfy

$$\|u(n,\cdot)\|^2_{L^2(\Omega)} \approx \|y_x(n,\cdot)\|^2_{L^2(\Omega)} + \|\mu\left(y(n,\cdot) - y(n,\cdot)^3\right)\|^2_{L^2(\Omega)}. \tag{20}$$

This implies — regardless of the value of σ — that the overshoot bound C in (19) is large if $\|y_x(n,\cdot)\|^2_{L^2(\Omega)} \gg \|y(0,\cdot)\|^2_{L^2(\Omega)}$ holds, which is the case in our example.

For ℓ from (17) inequality (18) becomes

$$\begin{aligned} &\|y(n,\cdot)\|^2_{L^2(\Omega)} + \|y_x(n,\cdot)\|^2_{L^2(\Omega)} + \lambda\|u(n,\cdot)\|^2_{L^2(\Omega)} \\ &\le \; C\sigma^n \left(\|y(0,\cdot)\|^2_{L^2(\Omega)} + \|y_x(0,\cdot)\|^2_{L^2(\Omega)}\right). \end{aligned} \tag{21}$$

Due to the fact that $\|y_x(0,\cdot)\|^2_{L^2(\Omega)} \gg \|y(0,\cdot)\|^2_{L^2(\Omega)}$ holds in our example, the approximate equation (20) does not imply large C in (21), which explains the considerable better performance for ℓ from (17).

The fact that the H^1-norm penalizes the distance to $y^* \equiv 0$ in a "stronger" way might lead to the conjecture that the better performance for this norm is intuitive. Our second example shows that this is not necessarily the case. This example is similar to the equation (14), (15), except that the distributed control is changed to Dirichlet boundary control. Thus, (14) becomes

$$y_t(t,x) = \nu y_{xx}(t,x) - y_x(t,x) + \mu\left(y(t,x) - y(t,x)^3\right), \tag{22}$$

again with $\nu = 0.1$ and $\mu = 10$, and (15) changes to

$$y(t,0) = u_0(t),\ y(t,1) = u_1(t) \text{ for all } t \ge 0,\ y(0,x) = y_0(x) \text{ for all } x \in \Omega$$

with $u_0, u_1 \in L^\infty(\mathbb{R},\mathbb{R})$. The cost functions (16) and (17) change to

$$\ell(y(n,\cdot),u(n,\cdot)) = \|y(n,\cdot)\|^2_{L^2(\Omega)} + \lambda(u_0(n)^2 + u_1(n)^2) \tag{23}$$

and

$$\ell(y(n,\cdot),u(n,\cdot)) = \|y(n,\cdot)\|^2_{L^2(\Omega)} + \|y_x(n,\cdot)\|^2_{L^2(\Omega)} + \lambda(u_0(n)^2 + u_1(n)^2), \tag{24}$$

respectively, again with $\lambda = 0.1$.

Due to the more limited possibilities to control the equation the problem obviously becomes more difficult, hence we expect to need larger optimization horizons for stability of the NMPC closed loop. However, what is surprising at the first glance is that ℓ from (23) stabilizes the system for smaller horizons than ℓ from (24), as the numerical results in Figure 5 confirm.

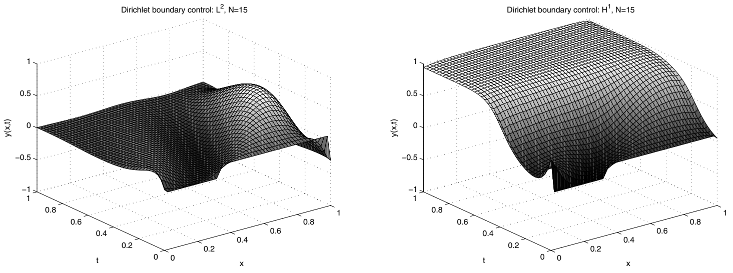


Fig. 5. NMPC closed loop for (22) with $N = 15$ and ℓ from (16)(left) and (17)(right)

A closer look at the dynamics reveals that we can again explain this behaviour with our theoretical results. In fact, steering the chosen initial solution to $y^* = 0$ requires u_1 to be such that a rather large gradient appears close to 1. Thus, during the transient phase $\|y_x(n,\cdot)\|^2_{L^2(\Omega)}$ becomes large which in turn causes ℓ from (24) to become large and thus causes a large overshoot bound C in (18). In ℓ from (23), on the other hand, these large gradients are not “visible” which is why the overshoot in (18) is smaller and thus allows for stabilization with smaller N.

5 Conclusions

In this paper we have shown how performance of NMPC schemes can be analyzed on basis of a controllability condition involving both the system dynamics and the cost function used in the optimization. The example of a semilinear parabolic PDE with distributed and boundary control illustrates how our theoretical results can be used for analyzing concrete systems.

References

1. Allgöwer F, Zheng A, eds. (2000), Nonlinear model predictive control, Birkhäuser, Basel
2. Altmüller N, Grüne L, Worthmann K (2010), Instantaneous control of the linear wave equation, Proceedings of MTNS 2010, Budapest, Hungary, to appear
3. Grüne L (2009) Analysis and design of unconstrained nonlinear MPC schemes for finite and infinite dimensional systems, SIAM J. Control Optim., 48, pp. 1206–1228
4. Grüne L, Pannek J, Seehafer M, Worthmann K (2009), Analysis of unconstrained nonlinear MPC schemes with time varying control horizon, Preprint, Universität Bayreuth; submitted
5. Grüne L, Pannek J, Worthmann K (2009), A networked unconstrained nonlinear MPC scheme, Proceedings of ECC 2009, Budapest, Hungary, pp. 371–376
6. Grüne L, Nešić D (2003), Optimization based stabilization of sampled–data nonlinear systems via their approximate discrete-time models, SIAM J. Control Optim., 42, pp. 98–122
7. Grüne L, Rantzer A (2008), On the infinite horizon performance of receding horizon controllers, IEEE Trans. Automat. Control, 53, pp. 2100–2111
8. Qin S, Badgwell T (2003), A survey of industrial model predictive control technology, Control Engineering Practice, 11, pp. 733–764
9. Rawlings JB, Mayne DQ (2009), Model Predictive Control: Theory and Design, Nob Hill Publishing, Madison

Nonlinear Model Predictive Control for an Artificial β-cell

Dimitri Boiroux, Daniel A. Finan, John B. Jørgensen, Niels K. Poulsen, and Henrik Madsen

DTU Informatics, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark, {dibo, dafi, jbj, nkp, hm}@imm.dtu.dk

Summary. In this contribution we apply receding horizon constrained nonlinear optimal control to the computation of insulin administration for people with type 1 diabetes. The central features include a multiple shooting algorithm based on sequential quadratic programming (SQP) for optimization and an explicit Dormand-Prince Runge-Kutta method (DOPRI54) for numerical integration and sensitivity computation. The study is based on a physiological model describing a virtual subject with type 1 diabetes. We compute the optimal insulin administration in the cases with and without announcement of the meals (the major disturbances). These calculations provide practical upper bounds on the quality of glycemic control attainable by an artificial β-cell.

1 Introduction

The World Health Organization estimates that more than 220 million people worldwide have diabetes, and this number is growing quickly [13]. The number of people with diabetes is projected to double between 2005 and 2030. In addition to the obvious physical and personal effects of diabetes, the disease also has a detrimental economic impact. In the USA, for example, the budget for diabetes care represents 10% of the health care budget, or more than \$130 billion (\$132 billion in 2002).

In people without diabetes, the pancreas regulates the blood glucose concentration tightly near 90 mg/dL (~5 mmol/L). Type 1 diabetes is a chronic disease characterized by the autoimmune destruction of the insulin-producing β-cells in the pancreas. Consequently, without insulin—a hormone whose key physiological role is to facilitate the uptake of glucose from the blood into the cells where it is metabolized—elevated concentrations of blood glucose, or *hyperglycemia*, occur. Prolonged hyperglycemia is known to cause a litany of complications: eye, nerve, and kidney disease, to name a few. Thus, exogenous insulin must be injected to lower the blood glucose. This treatment must be done carefully, however, because overinsulinization results in low blood glucose

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_26, © Springer-Verlag Berlin Heidelberg 2010

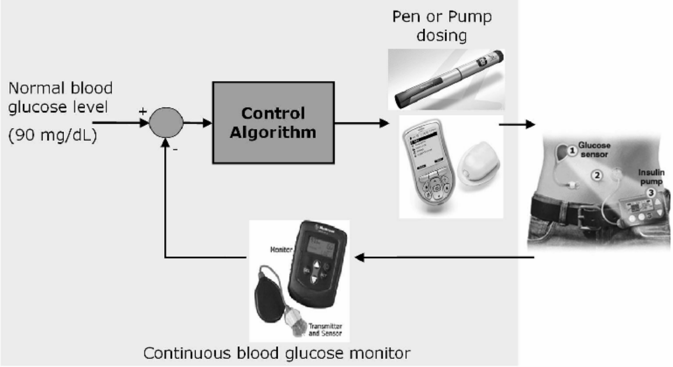


Fig. 1. Closed-loop glucose control for an artificial β-cell. Glucose is measured subcutaneously using a continuous glucose monitor (CGM). Insulin is dosed either continuously (using a pump) or in discrete instances (using a pen), based on the control algorithm.

concentrations, or *hypoglycemia*, which can pose immediate and severe health threats. Ideally, the blood glucose concentration should be kept within the *normoglycemic* range of approximately 70–140 mg/dL (or 3.9–7.8 mmol/L).

By today's standards, treatment consists of administration of exogenous insulin either continuously using an insulin pump or in discrete instances using an insulin pen (or syringe). In any case, the insulin is infused or injected into the subcutaneous tissue of the user, and thus must absorb into the intravenous system before being dispersed throughout the body. A critical component of this insulin therapy is the delivery of boluses (i.e., rapid injections) to offset the effects of carbohydrate (CHO) meals. The size of the bolus is based on a measurement of the current blood glucose and the (estimated) size of the meal, i.e., the amount of CHO in the meal.

Unfortunately, estimating the size of a meal can be a difficult task. Furthermore, having measurements only at meal times does not provide enough information about blood glucose. Hypoglycemic and hyperglycemic events can be missed due to these infrequent blood glucose measurements. In addition, such a measurement process does not provide any information about the dynamic trends of the blood glucose. Consequently, people with diabetes often tolerate frequent hyperglycemia in order to avoid hypoglycemia and its drastic effects.

An artificial β-cell is a biomedical device which would provide automatic regulation of blood glucose (in the case of a pump-based system), or at least optimal treatment suggestions (in the case of a pen-based system), based on a robust control algorithm [4]. A vital element to the success of such a device is the continuous glucose monitor (CGM), which will be used as the sensor in the closed-loop controller. A schematic of the artificial β-cell algorithm is shown in Fig. 1.

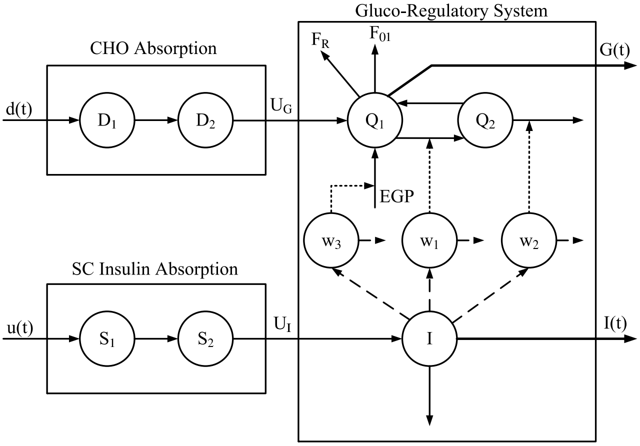


Fig. 2. Diagram of the physiological Hovorka model [8].

From a control perspective, insulin administration via an insulin pump offers a key advantage over insulin pens. Since an insulin pump is permanently attached to the patient, it is suitable for truly automatic, user-free control. That is, a pump-based system has the ability to adjust the manipulated variable, insulin infusion rate, at any time, independent of the patient. In contrast, a pen-based system ultimately relies on the patient physically delivering the insulin dose. There is, of course, an associated tradeoff: insulin pens are less invasive and cheaper for patients with type 1 diabetes.

2 Model Description

A prominent physiological model of the glucose-insulin dynamics in type 1 diabetes developed by Hovorka and colleagues [8] is depicted in Fig. 2. We use this Hovorka model to simulate a virtual subject with type 1 diabetes. In brief, it is a nonlinear model describing the effect of exogenous insulin, $u(t)$, on plasma insulin concentration, $I(t)$, and ultimately on blood glucose concentration, $G(t)$. In addition, the model accounts for the appearance of glucose in the blood due to CHO meals, $d(t)$, and endogenous insulin production, EGP, and removal due to insulin-independent cellular uptake, F_{01}, and renal excretion, F_R.

The model includes descriptions of subcutaneous (SC)-to-intravenous insulin absorption and CHO absorption from a meal, which are both represented as two-compartment (i.e., second order) submodels with time constants of $\tau_S = 55$ min and $\tau_D = 40$ min, respectively. The "slower" appearance of insulin in the blood, relative to meal-related glucose, has important and limiting control implications. These implications are elucidated through one of our key results, which is discussed in Optimization Results.

The nonlinearity in the Hovorka model is due primarily to the time-varying actions of insulin on glucose processes (namely, glucose transport, disposal, and endogenous production), denoted by w_1–w_3 in Fig. 2. Two other sources of nonlinearity are the insulin-independent glucose consumption F_{01} and the renal excretion of glucose F_R, which are both (modeled as) piecewise affine functions of the glucose concentration.

3 Problem Formulation

In this section, we state and discuss the continuous-time optimal control problem that is the basis for computing the insulin injection profiles for people with type 1 diabetes. We also discuss a numerically tractable discrete-time approximation to the continuous-time optimal control problem. The optimal insulin administration is formulated as the bound-constrained continuous-time Bolza problem

$$\min_{[x(t),u(t)]_{t_0}^{t_f}} \quad \phi = \int_{t_0}^{t_f} g(x(t),u(t))dt + h(x(t_f)) \tag{1a}$$

$$\text{s.t.} \quad x(t_0) = x_0 \tag{1b}$$

$$\dot{x}(t) = f(x(t),u(t),d(t)) \qquad t \in [t_0, t_f] \tag{1c}$$

$$u_{\min} \leq u(t) \leq u_{\max} \qquad t \in [t_0, t_f] \tag{1d}$$

in which $x(t) \in \mathbf{R}^{n_x}$ is the state vector, $u(t) \in \mathbf{R}^{n_u}$ is the vector of manipulated inputs, and $d(t) \in \mathbf{R}^{n_d}$ is a vector of known disturbances. $\dot{x}(t) = f(x(t),u(t),d(t))$ represents the model equations. The initial time, t_0, and the final time, t_f, are specified parameters. The initial state, x_0, is a known parameter in (1). The inputs are bound-constrained and must be in the interval $[u_{\min}, u_{\max}]$.

The objective function is stated generally with a stage cost term, $g(x(t),u(t))$, and a cost-to-go term, $h(x(t_f))$. The numerical algorithms for the problem are based on this general structure of the objective function.

3.1 Discrete-time Approximation

The continuous-time bound-constrained Bolza problem (1) is approximated by a numerically tractable discrete-time bound-constrained Bolza problem using the zero-order-hold input parameterization of the manipulated variables, $u(t)$, as well as the known disturbance variables, $d(t)$. We divide the time interval, $[t_0, t_f]$, into N intervals, each of length T_s. Let $\mathcal{N} = \{0, 1, ..., N-1\}$ and $t_k = t_0 + kT_s$ for $k \in \mathcal{N}$. The zero-order-hold restriction on the input variables, $u(t)$ and $d(t)$, implies

$$u(t) = u_k \qquad t_k \le t < t_{k+1} \qquad k \in \mathcal{N} \tag{2a}$$

$$d(t) = d_k \qquad t_k \le t < t_{k+1} \qquad k \in \mathcal{N} \tag{2b}$$

Using this zero-order-hold restriction on the inputs, the bound constrained continuous-time Bolza problem (1) may be approximated by

$$\min_{\{x_{k+1},u_k\}_{k=0}^{N-1}} \quad \phi = \sum_{k=0}^{N-1} G_k(x_k, u_k, d_k) + h(x_N) \tag{3a}$$

$$\text{s.t.} \quad b_k := F_k(x_k, u_k, d_k) - x_{k+1} = 0 \quad k \in \mathcal{N} \tag{3b}$$

$$u_{\min} \le u_k \le u_{\max} \qquad k \in \mathcal{N} \tag{3c}$$

The discrete-time state transition function is

$$F_k(x_k, u_k, d_k) = \{x(t_{k+1}) : \dot{x}(t) = f(x(t), u_k, d_k),\, x(t_k) = x_k\} \tag{4}$$

and the discrete time stage cost is

$$G_k(x_k, u_k, d_k) = \{\int_{t_k}^{t_{k+1}} g(x(t), u_k)dt : \dot{x}(t) = f(x(t), u_k, d_k),\, x(t_k) = x_k\} \tag{5}$$

4 Numerical Optimization Algorithm

In this section, we implement a multiple-shooting based SQP algorithm for the numerical solution of (1) [1, 5, 10]. The SQP algorithm is based on line search and structured high rank BFGS updates of the Hessian matrix [1, 10]. The structures of the quadratic subproblems are utilized and they are solved by a primal-dual interior-point algorithm using Riccati iterations [9, 11]. DOPRI54 is used for numerical solution of the differential equation model and sensitivities [3, 6, 7].

4.1 SQP Algorithm

We define the parameter vector, p, as $p = \begin{bmatrix} u_0' \; x_1' \; u_1' \; x_2' \; \ldots \; x_{N-1}' \; u_{N-1}' \; x_N' \end{bmatrix}'$, and the disturbance vector, d, as $d = \begin{bmatrix} d_0' \; d_1' \; \ldots \; d_{N-1}' \end{bmatrix}'$.

Then the bound constrained discrete-time Bolza problem (3) may be expressed as a constrained optimization problem in standard form

$$\min_{p} \quad \phi = \phi(p) \tag{6a}$$

$$\text{s.t.} \quad b(p) = 0 \tag{6b}$$

$$c(p) \ge 0 \tag{6c}$$

The concise formulation (6) is useful for presentation of the numerical optimization algorithm used for solving the bound constrained continuous-time Bolza problem (1).

The steps for solution of (6) by an SQP algorithm with line search are listed in Algorithm 1.

Algorithm 0.1 1 SQP Algorithm for (6)

```
Require: Initial guess: $(p^0, y^0, z^0)$ with $z^0 \geq 0$.
  Compute: $\phi(p^0)$, $\nabla_p \phi(p^0)$, $b(p^0)$, $\nabla_p b(p^0)$, $c(p^0)$, $\nabla_p c(p^0)$
  Set $\lambda = 0$, $\mu = 0$, $W^0 = I$
  while NOT stop do
    Compute $(\Delta p^k, \tilde{y}^{k+1}, \tilde{z}^{k+1})$ by solution of:

      $\min_{\Delta p} \quad \frac{1}{2} \Delta p' W^k \Delta p + \nabla_p \phi'(p^k) \Delta p$    (7a)
      $\text{s.t.} \quad \left[\nabla_p b(p^k)\right]' \Delta p = -b(p^k)$    (7b)
      $\left[\nabla_p c(p^k)\right]' \Delta p \geq -c(p^k)$    (7c)

    Compute $\Delta y^k = \tilde{y}^{k+1} - y^k$ and $\Delta z^k = \tilde{z}^{k+1} - z^k$
    Update the penalty parameter:
    $\mu \leftarrow \max\{|z|, \frac{1}{2}(\mu + |z|)\}$ and $\lambda \leftarrow \max\{|y|, \frac{1}{2}(\lambda + |y|)\}$
    Compute $\alpha$ using soft line search and Powell's $\ell_1$ merit function.
    $p^{k+1} = p^k + \alpha \Delta p^k$, $y^{k+1} = y^k + \alpha \Delta y^k$, $z^{k+1} = z^k + \alpha \Delta z^k$
    Compute $\phi(p^{k+1}), \nabla_p \phi(p^{k+1})$, $c(p^{k+1})$, $\nabla_p c(p^{k+1})$, $b(p^{k+1})$ and $\nabla_p b(p^{k+1})$
    Compute $W^{k+1}$ by Powell's modified BFGS update. $k \leftarrow k + 1$.
  end while
```

4.2 Gradient Computation

The most demanding computations in Algorithm 1 are those of the objective function $\phi(p)$, the derivatives of the objective function $\nabla_p \phi(p)$, the dynamics $b(p)$, and the sensitivities, $\nabla_p b(p)$, associated with the dynamics. $b(p)$ and $\phi(p)$ are computed by evaluation of (4) and (5), respectively. Consequently

$$b_k = b_k(x_k, x_{k+1}, u_k, d_k) = F_k(x_k, u_k, d_k) - x_{k+1} \tag{8a}$$

$$\nabla_{x_k} b_k = \nabla_{x_k} F_k(x_k, u_k, d_k) \tag{8b}$$

$$\nabla_{u_k} b_k = \nabla_{u_k} F_k(x_k, u_k, d_k) \tag{8c}$$

$$\nabla_{x_{k+1}} b_k = -I \tag{8d}$$

The gradients $\nabla_{x_k} F_k(x_k, u_k, d_k)$ $\nabla_{u_k} F_k(x_k, u_k, d_k)$ are computed by numerical integration of the sensitivity equations [2].

In the evaluation of the functions and derivatives needed in the SQP algorithm, i.e., $\phi(p)$, $\nabla_p \phi(p)$, $b(p)$, and $\nabla_p b(p)$, the major computational task is solving the sensitivity equations and evaluating the associated quadrature equations. The Hovorka model is a non-stiff system of differential equations. Therefore, we use an embedded Dormand-Prince explicit Runge-Kutta scheme (DOPRI54) for solving the differential equations and integrating the quadrature equations. A special DOPRI54 method has been implemented [2] in which we use the internal stages already computed by solving $\dot{x}(t) = f(x(t), u_k, d_k)$ in the evaluation of the quadrature equation. The implementation uses an adaptive time step based on PI-control [7].

5 Application to an Artificial β-cell

In this section we state and discuss the objective function and the scenarios used in the simulations. We also state the strategy for the nonlinear model predictive controller.

5.1 Nonlinear Model Predictive Control (NMPC)

NMPC is a receding horizon control technology that repeatedly solves open-loop nonlinear optimal control problems and implements the computed optimal control associated to the current time period [12]. In this contribution, we use a receding horizon strategy to compute the ideal insulin administration profile for people with type 1 diabetes. In order to obtain the ideal insulin profile, the NMPC uses state feedback and relative long prediction horizons.

5.2 Objective Function with Soft Output Constraints

The objective of the insulin administration is to compensate for glucose excursions caused by meals and by variations in endogenous glucose production and utilization. We use a penalty function defined as

$$\begin{aligned}\rho(G(t)) = \frac{\kappa_1}{2}|\max\{0, G(t) - \bar{G}\}|^2 + \frac{\kappa_2}{2}|\max\{0, \bar{G} - G(t)\}|^2 + \\ \frac{\kappa_3}{2}|\max\{0, G(t) - G_U\}|^2 + \frac{\kappa_4}{2}|\max\{0, G_L - G(t)\}|^2\end{aligned} \tag{9}$$

where $G(t)$ is the blood glucose concentration, $\bar{G} = 5$ mmol/L is the target value for the blood glucose concentration, $G_L = 4$ mmol/L is a lower acceptable limit, and $G_U = 8$ mmol/L is an upper acceptable limit. The weights κ_1–κ_4 are used to balance the desirability of different deviations from the target. As hypoglycemia is considered a more immediate risk than hyperglycemia, $\kappa_1 < \kappa_2$ and $\kappa_3 < \kappa_4$. The penalty function used in the simulations is illustrated in Fig. 3. Even though the penalty function (9) is not twice differentiable, we use the standard BFGS update procedure. $G(t)$ is a function of the state, $x(t)$, in the Hovorka model. Therefore, the penalty function (9) may be expressed as a stage cost in the form $g(x(t), u(t))$. The objective function used in the simulations is

$$\phi = \int_{t_0}^{t_f} g(x(t), u(t))dt + \frac{\eta}{2}\sum_{k=0}^{N-1} \|\Delta u_k\|_2^2 \tag{10}$$

where $u(t)$ represents the rate of insulin injection at any time and $\Delta u_k = u_{k+1} - u_k$. Given an initial state, x_0, and a CHO intake rate profile, $[d(t)]_{t_0}^{t_f}$, the continuous-time Bolza problem (1) computes the optimal insulin injection rate profile, $[u(t)]_{t_0}^{t_f}$, as well as the optimal state trajectory, $[x(t)]_{t_0}^{t_f}$. This

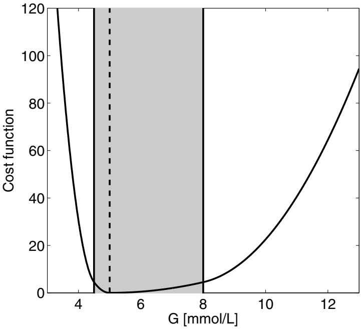


Fig. 3. Penalty as a function of the blood glucose concentration. The shaded region is the interval of acceptable glucose concentrations. The target glucose concentration is 5 mmol/L. Blood glucose concentrations less than 3 mmol/L are very undesirable as severe hypoglycemia can result in immediate dangers for the patient.

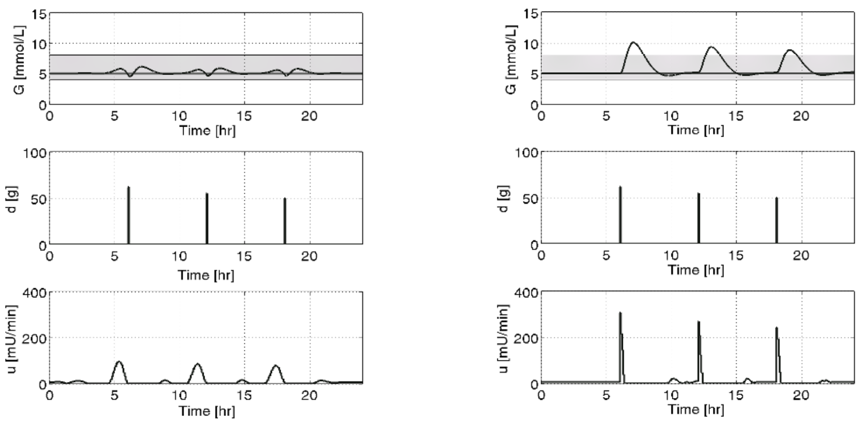
objective function has no cost-to-go function, i.e., $h(x(t_f)) = 0$, and can be brought into the standard form (3a) using state augmentation [12].

We use $u_{\min} = 0$ and a large $u_{\max}$ such that the upper bound is never active. (The former bound is self-evident, and the latter is consistent with realistic insulin pump and pen specifications.) We perform the optimization over a 24-hour window, i.e., $t_0 = 0$ min and $t_f = 24 \cdot 60 = 1440$ min, using a sampling time of $T_s = 5$ min (consistent with realistic CGM and insulin pump specifications). In the scenario considered, the simulated 70-kg subject has a 62-g CHO meal at 6:00, a 55-g CHO meal at 12:00, and a 50-g CHO meal at 18:00. To ensure an optimal blood glucose profile, a prediction horizon of six hours, i.e., $N = 6 \cdot 12 = 72$ samples, is employed in the receding horizon strategy.

6 Optimization Results

In this section, we use the Hovorka model and the developed multiple shooting SQP algorithm for (1) to compute insulin administration profiles for a virtual patient with type 1 diabetes.

Fig. 4(a) depicts the optimal insulin administration profile for the scenario in which the controller knows the size and time of all meals in advance. It illustrates the absolutely best insulin dosage and the corresponding glucose profile. This profile is obtained by solving the discrete-time constrained optimal control problem (3) given the disturbance vector d. It is evident from Fig. 4(a) that, due to the slower absorption of insulin relative to meal-related glucose (see Model Description), the optimal glucose concentration is achieved by administering the insulin in advance of the meal. Knowing the meal times and sizes allows the controller to deliver this anticipatory insulin to preempt



(a) Optimal insulin administration for the case with meal announcement in advance of the meal. Most insulin is taken before the meals.

(b) Optimal insulin administration with meal announcement at mealtime. Most insulin is taken in bolus like form at meal time.

Fig. 4. Optimal insulin administration and blood glucose profiles.

postprandial hyperglycemia. However, the assumption that the patient would know in advance—and with accuracy—the meal times and sizes is not practical. Safety considerations would preclude significant amounts of insulin from being delivered prior to mealtime.

Fig. 4(b) shows the simulation results for the more practical case in which the meals are announced to the MPC only at mealtime. Thus, the controller can deliver no anticipatory insulin prior to meals. The limitations for this case force the subject into (mild) hyperglycemia, but hypoglycemia is avoided. The insulin delivery profile for this case looks qualitatively similar to bolus delivery of insulin by a pen; most of the meal-related insulin is delivered in bolus form within the few samples after the meals are taken (and announced). Simulated optimal bolus treatment with a pen provides glucose profiles comparable to the glucose profile in Fig. 4(b) (results not shown).

These results demonstrate that for realistic cases, e.g., cases for which meal information is unknown until mealtime, acceptable control can still be obtained.

7 Conclusion

In this paper, we described a multiple shooting SQP algorithm for the solution of a bound-constrained discrete-time Bolza problem. Based on the Hovorka model for people with type 1 diabetes, we use an optimal control algorithm to compute insulin administration profiles for the cases with and without meal announcement in advance. The blood glucose profiles provide information about the best achievable performance in the case where anticipatory insulin administration is allowed, and in the case where insulin is delivered at mealtimes. The insulin profile for the realistic case with announcement of meals at mealtime is reminiscent of a bolus-based treatment regimen. This suggests that, for certain situations, insulin treatment based on pen systems may be nearly as effective as insulin treatment based on pump systems.

References

1. H.G. Bock and K.J. Plitt (1984) A multiple shooting method for direct solution of optimal control problems. Proc. of the IFAC 9th World Congress, pp. 242-247. Budapest, Hungary
2. D. Boiroux (2009) Nonlinear Model Predictive Control for an Artificial Pancreas. MSc Thesis, DTU Informatics, Technical University of Denmark
3. J. C. Butcher (2003) Numerical Methods for Ordinary Differential Equations. Wiley, Chichester, England
4. C. Cobelli, C. Dalla Man, G. Sparacino, L. Magni, G. De Nicolao and B. P. Kovatchev (2009) Diabetes: Models, Signals, and Control. IEEE Reviews in Biomedical Engineering, vol. 2, pp. 54-96
5. M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy and F. Allgöwer (2002) Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. Journal of Process Control, vol. 12, pp. 577-585
6. J. R. Dormand and P. J. Prince (1980) A family of embedded Runge-Kutta formulae. Journal of Computational and Applied Mathematics, vol. 6, no. 1, pp. 19-26
7. K. Gustafsson (1992) Control of Error and Convergence in ODE Solvers. PhD Thesis, Department of Automatic Control, Lund Institute of Technology
8. R. Hovorka, V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. Orsini Federici, T. R. Pieber, H. C. Schaller, L. Schaupp, T. Vering and M. E. Wilinska (2004) Nonlinear Model Predictive Control of Glucose Concentration in Subjects with Type 1 Diabetes. Physiological Measurement, vol. 25, pp. 905-920
9. J. B. Jørgensen (2005) Moving Horizon Estimation and Control. PhD Thesis, Department of Chemical Engineering, Technical University of Denmark
10. D. B. Leineweber, I. Bauer, H. G. Bock, J. P. Schlöder (2003) An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization. Part 1: theoretical aspects. Computers and Chemical Engineering, vol. 27, pp. 157-166
11. C. V. Rao, S.J. Wright and J. B. Rawlings (1998) Application of Interior-Point Methods to Model Predictive Control. Journal of Optimization Theory and Applications, vol. 99, nr. 3, pp. 723-757.
12. J. B. Rawlings and D. Q. Mayne (2009) Model Predictive Control: Theory and Design. Nob Hill Publishing. Madison, Wisconsin, USA
13. World Health Organization (2009) Diabetes (fact sheet no. 312). Website: http://www.who.int/mediacentre/factsheets/fs312/en/.

An Optimized Linear Model Predictive Control Solver

Dimitar Dimitrov[1], Pierre-Brice Wieber[2], Olivier Stasse[3], Hans Joachim Ferreau[4], and Holger Diedam[5]

[1] Örebro University - Sweden, mitko@roboresearch.net
[2] INRIA Grenoble - France pierre-brice.wieber@inria.fr
[3] JRL - Japan olivier.stasse@aist.go.jp
[4] KU Leuven - Belgium joachim.ferreau@esat.kuleuven.be
[5] Heidelberg University - Germany hdiedam@ix.urz.uni-heidelberg.de

Summary. This article addresses the fast on-line solution of a sequence of quadratic programs underlying a linear model predictive control scheme. We introduce an algorithm which is tailored to efficiently handle small to medium sized problems with relatively small number of active constraints. Different aspects of the algorithm are examined and its computational complexity is presented. Finally, we discuss a modification of the presented algorithm that produces "good" approximate solutions faster.

1 Introduction

Model Predictive Control (MPC) is an advanced control tool that originates in the late seventies. Due to its simplicity, it quickly became the preferred control tool in many industrial applications [1]. Some of the fields where MPC is already considered to be a mature technique involve linear and rather slow systems like the ones usually encountered in the chemical process industry. However, the application of MPC to more complex systems, involving nonlinear, hybrid, or very fast processes is still in its infancy.

MPC does not designate a specific control strategy but rather an ample range of control methods which use a model of a process to obtain control actions by minimizing an objective function, possibly subject to given constraints. The various algorithms in the MPC family can be distinguished mainly by: (i) the model used to represent the process; (ii) the objective function to be minimized; (iii) the type of constraints. The most popular scheme applied in practice involves a linear time-invariant process model, linear constraints and quadratic objective function [2]. In general, it is referred to as linear MPC (LMPC). The computational burden associated with the application of LMPC is mainly due to forming and solving a Quadratic Program

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_27, © Springer-Verlag Berlin Heidelberg 2010

(QP) at each sampling interval. This imposes restrictions on the application of LMPC to systems that require short sampling times.

In practice, the solution of the underlying sequence of QPs is left to state of the art QP solvers [3]. Even though such solvers implement very efficient algorithms, in most cases they do not make use of the properties of each particular problem, which could speed up computations considerably.

In this article we present an algorithm for the fast on-line solution of a sequence of QPs in the context of LMPC. When the control sampling times become so small, that classical methods fail to reach a solution (within a given sampling interval), algorithms that can exploit the particular structure of LMPC problems become attractive. The proposed algorithm is tailored to efficiently utilize data that can be precomputed off-line, leading to smaller on-line computational burden. We assume that the problem to be solved is small to medium sized, with relatively small[6] number of active constraints.

The presented algorithm can be classified as a *primal active set method* with *range space linear algebra*. We motivate our choice by analyzing the requirements of our problem. Different aspects of the algorithm are examined, and its computational complexity is presented. We discuss details related to efficient update methods for the solution of the underlying systems of linear equations. Finally, we present a strategy for altering the *working set* resulting in a "good" approximate solutions that can be computed faster.

2 Linear Model Predictive Control

There is a great variety of models commonly used in the context of MPC. In general, they can be divided into two groups: (i) *first principles models* and (ii) *identified models*. The former are based on physical or chemical laws of nature, whereas the latter are built as a result of empirical measurements of the real process. Here, we assume that regardless of the way the model is obtained, it is represented in the following form

$$x_{k+1} = A\,x_k + B\,u_k, \tag{1a}$$

$$y_k = C\,x_k \tag{1b}$$

where, $x_k \in R^{n_x}$ represents the state of the system, $u_k \in R^{n_u}$ is control input, $y_k \in R^{n_y}$ is a vector of measured outputs which are to be controlled (to satisfy given constraints and when possible to follow certain reference profile), and A, B, C are constant matrices with appropriate dimensions.

In order to express the behavior of system (1) for N discrete steps in the future as a function of x_k and $n = Nn_u$ control actions, equation (1a) is iterated N times (combined with N versions of (1b)) as follows

[6] The term "relatively small" will be properly defined in Section 3.

$$y_{k+\tau} = CA^{\tau}x_k + C\sum_{\rho=0}^{\tau-1} A^{(\tau-\rho-1)}Bu_{k+\rho}\,, \quad (\tau = 1, ..., N). \tag{2}$$

Using the notation

$$Y_{k+1} = \begin{bmatrix} y_{k+1} \\ \vdots \\ y_{k+N} \end{bmatrix} \in \mathbb{R}^{Nn_y}, \qquad U = \begin{bmatrix} u_k \\ \vdots \\ u_{k+N-1} \end{bmatrix} \in \mathbb{R}^{n}$$

recursion (2) can be expressed in the following compact way

$$Y_{k+1} = P_x\,x_k + P_u\,U \tag{3}$$

where, $P_x \in \mathbb{R}^{Nn_y \times n_x}$ and $P_u \in \mathbb{R}^{Nn_y \times n}$ are constant matrices (independent of k).

MPC uses a process model in order to predict the process behavior starting at a given discrete time k, over a future *prediction horizon* $k+N$. Assuming that information about disturbances and state measurement noise is not available, the predicted behavior depends on the current state x_k and the assumed control input trajectory U that is to be applied over the prediction horizon. The idea is in step (i) to select U which leads to the “best” predicted behavior (according to a given objective function). Once U is obtained, in step (ii) only the first control action (u_k) is applied to the system until the next sampling instant. Then in step (iii) the new state x_{k+1} is measured (or estimated), and the process is repeated again from step (i). Hence, MPC is a feedback control strategy. In a standard LMPC scheme, the n future control actions U are computed to minimize given quadratic cost function [2]

$$\underset{U}{\text{minimize}}\ \frac{1}{2}U^{T}QU + U^{T}p_k \tag{4}$$

where, $Q \in R^{n\times n}$ is a symmetric and positive-definite constant Hessian matrix, and $p_k \in R^{n}$ is a gradient vector.

Furthermore, the profile of the outputs Y_{k+1} are possibly constrained to satisfy a set of m linear constraints of the form

$$D_{k+1}Y_{k+1} \leq b^{'}_{k+1}, \tag{5}$$

for some matrix $D_{k+1} \in R^{m\times Nn_y}$ and vector $b^{'}_{k+1} \in R^{m}$. If the i^{th} row of D_{k+1} ($i = 1,\ldots,m$) imposes constraints only on y_{k+i} (which is very common in practice), D_{k+1} will be extremely sparse and well structured matrix with at most n_y nonzero entries in each row. Hereafter, we assume that D_{k+1} has such structure. Introducing (3) in (5) leads to

$$G_{k+1}U \leq b_{k+1} \tag{6}$$

where, $G_{k+1} = D_{k+1}P_u$, and $b_{k+1} = b^{'}_{k+1} - D_{k+1}P_x x_k$. Additional constraints accounting for actuator limits etc. could be imposed.

The objective function (4) in combination with the constraints (6) define a canonical optimization problem known as quadratic program. Its solution is required for the application of a LMPC scheme.

3 General design choices for a QP solver

The choice of algorithm that can efficiently solve a sequence of quadratic programs defined by (4) and (6) is not unique. In general, the choice depends mostly on: (i) the number m_a of active constraints (constraints that hold as equalities at the optimal point) and the dimension N; (ii) whether a "warm start" is available; (iii) whether there is a cheap way to determine an initial feasible point that satisfies the constraints in (6). The following short overview aims at outlining some of the considerations that need to be made when choosing a QP solver.

3.1 Interior point vs. active set methods

Fast and reliable solvers for solving QPs are generally available, usually based on *interior point* or *active set* methods, and there has been a great deal of research related to the application of both approaches in the context of MPC [4].

Finding the solution of a QP in the case when the set of active constraints at the optimum is known, amounts to solving a linear system of equations that has a unique solution [5]. Active set methods are iterative processes that exploit the above property and try to guess at each iteration which are the active constraints at the optimal point. They usually consider active constraints one at a time, inducing a computation time directly related to m_a. On the contrary, the computation time of interior point methods is relatively constant, regardless of the number of active constraints. However, this constant computation time can be large enough to compare unfavorably with active set methods in cases where m_a is relatively small.

It should be noted that what we have to solve is not a singe QP but a series of QPs, which appear to be sequentially related. It is possible then to use information about the solution computed at sampling time k to accelerate the computation of the solution at sampling time $k+1$. Such information is usually referred to as "warm starting", and *active set* methods typically gain more from it [4]. Hence, they are preferred when dealing with small to medium sized QPs where m_a is kept small.

3.2 Primal vs. dual strategies

There exist mainly two classes of active set methods, *primal* and *dual* strategies. Primal strategies ensure that all the constraints (6) are satisfied at every iteration. An important implication of this feature is that if there is a limit on computation time (a real-time bound), e.g. because of the sampling period of the control law, the iterative process can be interrupted and still produce at any moment a feasible motion. Obviously, this comes at the cost of obtaining a sub-optimal solution.

One limitation of primal strategies is that they require an initial value for the variables U which already satisfy all the constraints. For a general QP, computing such an initial value can take as much time as solving the QP afterwards, which is a strong disadvantage. This is why dual methods are usually preferred: they satisfy all the constraints (6) only at the last iteration, but they do not require such an initial value.

3.3 Null space vs. range space algebra

There exist mainly two ways of making computations with the linear constraints (6), either considering the *null space* of the matrix G_{k+1}, orthogonal to the constraints, or the *range space* of this matrix, parallel to the constraints. The first choice leads to working with matrices of size $(n - m_a) \times (n - m_a)$, while the second choice leads to working with matrices of size $m_a \times m_a$. Hence, the most efficient of those two options depends on whether $m_a < n/2$ or not. It should be noted that, when dealing with ill-conditioned matrices, range space algebras can behave poorly.

3.4 Problem structure

As it was already pointed out, in practice the solution of the QP underlying a LMPC scheme is left to state of the art QP solvers [3]. Even though such solvers implement very efficient algorithms, in most cases they do not exploit the properties of each particular problem. One such property is that the matrix G_{k+1} of the constraints (6) can be expressed as a product of $D_{k+1}P_u$, where P_u is constant. In many applications [6] D_{k+1} is well structured and extremely sparse (a property that is lost after G_{k+1} is formed explicitly). The primal algorithm in [7] and dual algorithm in [8] are probably the ones that are considered as first choices when dealing with small to medium sized problems, however, they are not able to take advantage of the sparsity pattern of D_{k+1} and the fact that P_u is constant, leading to a requirement for new algorithms that account for this structure.

3.5 Our choice

- If the system in (1) is output controllable, P_u will have full row rank and by computing its (generalized) inverse off-line, a feasible U can be obtained at a low cost [9], [10]. Furthermore, if the solution of a QP can not be obtained within a predefined sampling time, we want to be able to interrupt the process and still obtain a feasible motion of our system. These considerations led to the development of a *primal solver*.

- Due to our assumption, that the number of active constraints is relatively small *i.e.* $m_a < n/2$, we chose to use a solver with *range space linear algebra.*

4 An optimized QP solver

4.1 Off-line change of variable

Typically, the first action of an active set method is to make a Cholesky decomposition of the matrix $Q = L_Q L_Q^T$. When range space algebra is used, at each iteration a change of variable involving L_Q is performed twice [7]. First, when adding a constraint to the so called *working set*, and second, when the search direction is evaluated. This results in using n^2 *flops* at each iteration[7]. In this way, the QP defined by (4) and (6) simplifies to a Least Distance Problem (LDP) [11]

$$\underset{V}{\text{minimize}} \quad \frac{1}{2}V^T V + V^T g_k \tag{7a}$$

$$\text{subject to} \quad \underbrace{D_{k+1} P_u}_{G_{k+1}} L_Q^{-T} V \leq b_{k+1}, \tag{7b}$$

where, $V = L_Q^T U$ and $g_k = L_Q^{-1} p_k$. In a general setting, representing (4) and (6) in the form of (7) using one change of variable before solving the QP is not performed, because the matrix-matrix product $G_{k+1} L_Q^{-T}$ has to be evaluated (which is computationally expensive if both matrices are dense).

For the problem treated in this article, however, the matrices L_Q and P_u are constant and the product $P_u L_Q^{-T}$ can be precomputed off-line. Furthermore, due to the assumption that D_{k+1} is sparse (with at most n_y nonzero entries in each row), forming $D_{k+1} P_u L_Q^{-T}$ requires mnn_y flops, which is computationally cheaper than using n^2 flops during each step of the solution. Note that in many applications, large parts of D_{k+1} can remain unchanged from one sampling time to the next. Due to the above considerations, we perform a change of variable and solve on-line the LDP (7).

4.2 The iterative process

Active set methods are iterative processes that try to guess at each iteration which are the active constraints, the inequalities in (7b) which hold as equalities at the minimum V^*. Indeed, once these equalities, denoted by

$$EV = q$$

[7] We measure computational complexity in number of floating-point operations, flops. We define a flop as one multiplication/division together with an addition. Hence, a dot product $a^T b$ of two vectors a, $b \in \mathbb{R}^n$ requires n flops.

are identified, the minimum of the LDP is [5]

$$V^* = -g_k + E^T \lambda \tag{8}$$

with Lagrange multipliers λ solving

$$EE^T \lambda = q + Eg_k. \tag{9}$$

In the case of a primal strategy, the iterations consist in solving these equations with a guess of what the active set should be, and if the corresponding solution violates some of the remaining constraints, include (usually) one of them (using a give criterion) in our guess (*working set*) and try again. Once the solution does not violate any other constraint, it remains to check that all the constraints we have included in our guess should actually hold as equalities. That is done by checking the sign of the Lagrange multipliers. A whole new series of iterations could begin then which alternate removing or adding constraints to our guess. All necessary details can be found in [5], [11].

4.3 Efficient update method

At each iteration we need to solve equations (8) and (9) with a new guess of the active set (here, we assume that the constraints in our guess are linearly independent, *i.e.* EE^T is full rank). The only thing that changes from one iteration to the next is that a single constraint is added or removed to/from the *working set*, *i.e.* only one line is either added or removed to/from the matrix E. Due to this structure, there exist efficient ways to compute the solution of (8) and (9) at each iteration by updating the solution obtained at the previous iteration without requiring the computation of the whole solution from scratch.

Probably the most efficient way to do so in the general case is the method described in [7]. There, a Gram-Schmidt decomposition of the matrix E is updated at each iteration at a cost of $2nm_a$ flops. Consequently, the Gram-Schmidt decomposition is used in a "clever way", allowing to update the solution of (8) and (9) at a negligible cost. In this way, the only computational cost when adding a constraint is the $2nm_a$ flops of the Gram-Schmidt update.

In our specific case, we can propose a slightly better option, based on the Cholesky decomposition of the matrix $EE^T = L_E L_E^T$. Below we describe the update procedure when a new row e is added to the matrix E. In such case, we need the decomposition of the new matrix

$$\begin{bmatrix} E \\ e \end{bmatrix} \begin{bmatrix} E^T & e^T \end{bmatrix} = \begin{bmatrix} EE^T & Ee^T \\ eE^T & ee^T \end{bmatrix}. \tag{10}$$

First note that, since the matrix $P_u L_Q^{-T}$ is constant, we can form off-line the Gramian matrix $\mathcal{G} = P_u L_Q^{-T} L_Q^{-1} P_u^T$, which is the matrix containing the dot products of each row of $P_u L_Q^{-T}$ with all others. Noting that, the rows of matrix E and the (row) vector e are taken from the constraints (7b), the dot products Ee^T and ee^T can be obtained at a negligible cost from the entries of $\mathcal{G}$ under the action of the varying but extremely sparse and well structured matrix D_{k+1}. After matrix (10) is formed, classical methods for updating its Cholesky decomposition (once $EE^T = L_E L_E^T$ is known) require $m_a^2/2$ flops.

Using Cholesky decomposition, equation (9) can be solved in three very efficient steps:

$$w_1 = q + Eg_k, \tag{11a}$$

$$L_E w_2 = w_1, \tag{11b}$$

$$L_E^T \lambda = w_2. \tag{11c}$$

When one constraint is added to the matrix E, updating the value of w_1 requires only one dot product to compute its last element. Since only the last element of w_1 changes and only one new line is added to L_E, only the last element of w_2 needs to be computed to update its value, at the cost of a dot product. Only the third step requires more serious computations: since the matrix L_E is lower triangular of size m_a, solving this system requires $m_a^2/2$ flops.

Once equation (9) is solved for the Lagrange multipliers λ, the computation of V in (8) requires a nm_a matrix-vector product. In total, the above update requires $nm_a + m_a^2$ flops, which is slightly better than the $2nm_a$ found in [7], which is possible in our case due to the precomputation of the matrix $\mathcal{G}$ off-line. Even though V (computed from (8)) satisfies the equality constraints $EV = q$, it is not guaranteed to satisfy all the inequality constraint not included in the *working set*. In order to produce feasible iterates, at each step, the scheme presented in [5] (pp. 468-469), [9] is used.

The case when a constraint is removed from E is handled in a classical way (see [12]), and is not presented here.

4.4 Approximate solution & warm start

Depending on the sampling time of the control, obtaining the solution of each QP might not be possible. Because of this, here we present a modification of a standard primal algorithm that computes a “good” approximate solution faster. As observed in [9], [14], a “good” approximate solution does not result in a significant decrease in the quality of the MPC control law.

As already mentioned in Section 4.2, once a solution V (for some guess of the active set) that does not violate any of the constraints (7b) is found, it can be certified to be the optimal point if $\lambda_i > 0$ ($i = 1, \ldots, m_a$). If this test fails, (usually) one constraint is dropped from the *working set*, resulting in a new series of iterations. In order to speed-up the on-line computation, we propose to terminate the solution of each QP once a solution V of (8) that satisfies all constraints in (7b) is found, regardless of signs of the Lagrange multipliers. Accounting for the negative entries of λ is then performed when formulating the warm start for the next QP. Under the assumption that the active set of the QP solved at sampling time k closely resembles the one of the QP that needs to be solved at sampling time $k+1$, we use as an initial guess for the *working set* all active constraints from the previous QP except the ones that correspond to negative Lagrange multipliers. In that way, the modification of the *working set* is no longer treated separately at a local level (for each separate QP), but rather considered as a shared resource among the whole sequence.

The reasoning for starting with a nonempty *working set* can be motivated by noting that, if only adding constraints to our guess for the active set is considered, each iteration of the presented algorithm requires $nm + m_a^2$ flops. If the solution of each QP starts with an empty *working set*, the complexity of adding m_a constraints (one at a time) is approximately $nmm_a + m_a^3/3 + m_a^2/2$ flops[8]. In contrast, if matrix E from the previous QP is used (with some rows removed), the only necessary computation required for realizing the warm start is finding the Cholesky decomposition of the modified EE^T. This can be done by updating the already available factorization $L_E L_E^T$ from the previous QP, which (depending on which constraints are removed) requires at most $m_a^3/3$ flops, which is a tremendous improvement over the $nmm_a + m_a^3/3 + m_a^2/2$ flops that would have been necessary to reach the same active set through the whole set of iterations.

In [9], we already applied the above idea using the LMPC scheme for walking motion generation for a humanoid robot proposed in [13], and the active set when doing so is in most cases correct or includes only one, and in rare cases two unnecessarily activated constraints. This leads to slightly sub-optimal solutions, which nevertheless are feasible. We have observed that this does not affect the stability of the scheme: the difference in the generated walking motions is negligible, however, the computation time is considerably smaller (see [9] for results from a numerical comparison with a state of the art QP solver).

When a nonempty initial active set is specified, the initial point needs to lie on the constraints in this set. If the system in (1) is output controllable, such point can be generated by using a procedure similar to the one presented in [10]. In the general case, however, a general feasibility problem has to be solved.

5 Conclusion

In this article we presented an optimized algorithm for the fast solution of a quadratic program in the context of model predictive control. We discussed

[8] To this count one should add $nmm_a - nm_a^2/2 - nm_a/2$ flops, which is the complexity of checking whether V (computed from (8)) violates any of the inequality constraints not included in the active set. This check is common for all active set algorithms, and is not discussed in this article.

alternative solution methods, and analyzed their properties for different problem structures. The presented algorithm was designed with the intention of using as much as possible data structures which can be precomputed off-line. In such a way, we are able to decrease the on-line computational complexity. A strategy for producing "good" approximate solutions in the presence of a real-time bound on the computation time was presented.

References

1. S. J. Qin, and T. A. Badgwell, "An overview of industrial model predictive control technology. In chemical process control: Assessment and new directions for research," in *AIChE Symposium Series 316, 93, Jeffrey C. Kantor, Carlos E. Garcie and Brice Carnahan Eds.,* 232-256, 1997.
2. J. Maciejowski, "Predictive Control with Constraints," in *Prentice Hall,* 2001.
3. K. Schittkowski, "QL: A Fortran code for convex quadratic programming - User's guide," *University of Bayreuth*, Report, Version 2.11, 2005.
4. S. Wright, "Applying new optimization algorithms to model predictive control," in *Proc. of CPC-V*, 1996.
5. J. Nocedal, and S. J. Wright, "Numerical optimization," *Springer Series in Operations Research, 2nd edition*, 2000.
6. H. Diedam, D. Dimitrov, P.-B. Wieber, M. Katja, and M. Diehl, "Online walking gait generation with adaptive foot positioning through linear model predictive control," in *Proc. of the IEEE/RSJ IROS,* pp. 1121-1126, 2008.
7. P.E. Gill, N.I. Gould, W. Murray, M.A. Saunders, and M.H. Wright "A weighted gram-schmidt method for convex quadratic programming," *Mathematical Programming,* Vol.30, No.2, pp.176-195, 1984
8. D. Goldfarb, and A. Idnani, "A numerically stable dual method for solving strictly convex quadratic programs," *Mathematical Programming,* 27:1-33, 1983.
9. D. Dimitrov, P.-B. Wieber, O. Stasse, J. Ferreau, and H. Diedam, "An optimized linear model predictive control solver for online walking motion generation," in *Proc. of the IEEE Int. Conf. on Robot. & Automat.,* pp. 1171-1176, 2009.
10. D. Dimitrov, J. Ferreau, P.-B. Wieber, and M. Diehl, "On the implementation of model predictive control for on-line walking pattern generation," in *Proc. of the IEEE Int. Conf. on Robot. & Automat.,* pp. 2685-2690, 2008.
11. R. Fletcher, "Practical Methods of Optimization," *John Wiley & Sons,* 1981.
12. H.J. Ferreau, "An online active set strategy for fast solution of parametric quadratic programs with applications to predictive engine control," *University of Heidelberg,* 2006.
13. P.-B. Wieber, "Trajectory free linear model predictive control for stable walking in the presence of strong perturbations," in *Proc. of IEEE-RAS Int. Conf. on Humanoid Robots,* pp.137-142, 2006.
14. Y. Wang, and S. Boyd, "Fast model predictive control using online optimization," in *Proc. of 17th IFAC World Congress on Automatic Control,* pp. 6974-6979, 2008.

A Linear-Quadratic Model-Predictive Controller for Control and State Constrained Nonlinear Control Problems

Matthias Gerdts[1] and Björn Hüpping[1]

Institut für Mathematik, Universität Würzburg, Am Hubland, 97074 Würzburg, Germany gerdts@mathematik.uni-wuerzburg.de, bjoern.huepping@mathematik.uni-wuerzburg.de

The authors are supported by DFG grant GE 1163/5-1

We consider nonlinear control problems subject to control and state constraints and develop a model-predictive controller which aims at tracking a given reference solution. Instead of solving the nonlinear problem, we suggest solving a local linear-quadratic approximation in each step of the algorithm. Application of the virtual control concept introduced in [1, 4] ensures that the occuring control-state constrained linear-quadratic problems are solvable and accessible to fast function space methods like semi-smooth Newton methods. Numerical examples support this approach and illustrate the idea.

1 LQR control with constraints

The goal of this work is to propose a fast and reliable numerical method for controlling control-state constrained control systems. The underlying system in the time interval $[0, t_f]$ with fixed $t_f > 0$ is described by ordinary differential equations

$$\dot{x}(t) = f(t, x(t), u(t)), \qquad x(0) = x_0, \tag{1}$$

subject to control and state constraints

$$u(t) \in \mathcal{U}(t) \tag{2}$$

$$s(t, x(t)) \leq 0, \tag{3}$$

where f and s are sufficiently smooth functions and $\mathcal{U}(t)$ is a convex set with non-empty interior. The aim is to track a given reference state trajectory $x_{ref} \in W^{1,\infty}([0, t_f], \mathbb{R}^{n_x})$ and a given reference control trajectory $u_{ref} \in L^{\infty}([0, t_f], \mathbb{R}^{n_u})$ in the time interval $[0, t_f]$. Herein, $W^{1,\infty}([0, t_f], \mathbb{R}^{n_x})$ and $L^{\infty}([0, t_f], \mathbb{R}^{n_u})$ denote the space of absolutely continuous functions with

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_28, © Springer-Verlag Berlin Heidelberg 2010

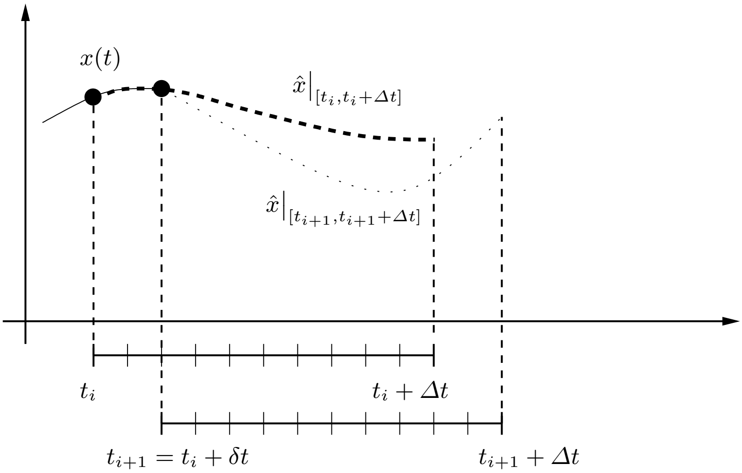

$x(t)$
$\hat{x}\big|_{[t_i,t_i+\Delta t]}$
$\hat{x}\big|_{[t_{i+1},t_{i+1}+\Delta t]}$
t_i
$t_i+\Delta t$
$t_{i+1}=t_i+\delta t$
$t_{i+1}+\Delta t$

Fig. 1. MPC Concept

essentially bounded first derivative and the space of essentially bounded functions, respectively.

The reference functions may result from an optimal control problem or an equilibrium solution. For a given initial state $x_0 \in \mathbb{R}^{n_x}$, which may deviate from the reference state trajectory, a controller aims at solving the following

Problem 1 (Tracking problem). Find a state $x \in W^{1,\infty}([0,t_f],\mathbb{R}^{n_x})$ and a control $u \in L^\infty([0,t_f],\mathbb{R}^{n_u})$ with (1), (2), and (3), that minimizes the functional

$$F(x,u) := \frac{1}{2}\int_0^{t_f} \begin{pmatrix} \Delta x(t) & \Delta u(t) \end{pmatrix} \begin{pmatrix} Q(t) & R(t) \\ R^\top(t) & S(t) \end{pmatrix} \begin{pmatrix} \Delta x(t) \\ \Delta u(t) \end{pmatrix} dt,$$

for some positive semidefinite symmetric time dependent weighting matrix $\begin{pmatrix} Q & R \\ R^\top & S \end{pmatrix}$, where

$$\Delta x(t) := x(t) - x_{ref}(t), \qquad \Delta u(t) := u(t) - u_{ref}(t).$$

Solving the fully nonlinear Problem 1 in real-time is often not possible owing to high computational costs. Hence, we aim at approximately solving the problem using a model predictive control (MPC) approach in combination with a linear-quadratic regulator (LQR) approximation. Related approaches using nonlinear MPC and efficient implementations with realtime ability have been established in [5, 6, 7].

The idea of model predictive control is illustrated in Figure 1. The algorithm depends on a local time horizon $\Delta t > 0$ and sampling times $t_{i+1} = t_i + \delta t$, $i = 0, 1, 2, \ldots$. On each local time horizon $[t_i, t_i + \Delta t]$, a local tracking problem similar to Problem 1 with initial state $x(t_i) = x_i$ has to be solved. Then the resulting optimal control is applied on the interval $[t_i, t_i + \delta t]$. In the next step, the computation is started anew in the period $[t_{i+1}, t_{i+1} + \Delta t]$ with $t_{i+1} := t_i + \delta t$ and new initial state $x_{i+1} = x(t_{i+1})$, cf. Figure 1.

In order to accelerate the computation, linear-quadratic approximations of Problem 1 are being solved on each local time horizon in spite of the linearization error that unavoidably will occur. These problems take the form

Problem 2. Let Δx_i denote the deviation of the actual state from the reference state trajectory at t_i. Find a control correction $\Delta u \in L^\infty([t_i, t_i+\Delta t], \mathbb{R}^{n_u})$ and a state correction $\Delta x \in W^{1,\infty}([t_i, t_i + \Delta t], \mathbb{R}^{n_x})$ that minimize

$$F(\Delta u, \Delta x) := \frac{1}{2} \int\limits_{t_i}^{t_i+\Delta t} \begin{pmatrix} \Delta x(t) & \Delta u(t) \end{pmatrix} \begin{pmatrix} Q(t) & R(t) \\ R^\top(t) & S(t) \end{pmatrix} \begin{pmatrix} \Delta x(t) \\ \Delta u(t) \end{pmatrix} dt$$

and satisfy the constraints

$$\begin{aligned} \Delta \dot{x}(t) &= A(t)\Delta x(t) + B(t)\Delta u(t), \qquad \Delta x(t_i) = \Delta x_i, \\ C(t)\Delta x(t) &\leq d(t), \\ \Delta u(t) &\in U(t) - \{u_{ref}(t)\}. \end{aligned}$$

Herein, A, B, C, and d are given by

$$\begin{aligned} A(t) &= f'_x(t, x_{ref}(t), u_{ref}(t)), & B(t) &= f'_u(t, x_{ref}(t), u_{ref}(t)) \\ C(t) &= s'_x(t, x_{ref}(t)), & d(t) &= -s(t, x_{ref}(t)). \end{aligned}$$

Summarizing, we obtain

Algorithm: (Linear-quadratic MPC algorithm)

1. Let $i = 0$, $\Delta x_0 = x_0 - x_{ref}(0)$.
2. Compute the solution $(\Delta u, \Delta x)$ of Problem 2 on $[t_i, t_i + \Delta t]$.
3. Apply the control $u|_{[t_i,t_i+\delta t)} := u_{ref}|_{[t_i,t_i+\delta t)} + \Delta u$ in $[t_i, t_i+\delta t)$ and predict the state trajectory by solving in $[t_i, t_i + \delta t]$ the initial value problem
$$\dot{x}(t) = f(t, x(t), u(t)), \qquad x(t_i) = x_i.$$
4. Let $x_{i+1} = x(t_i + \delta t)$, $\Delta x_{i+1} := x_{i+1} - x_{ref}(t_i + \delta t)$, $i := i + 1$ and goto 2.

It should be mentioned that this algorithm will not be able to exactly satisfy the state constraints (3). But in contrast to classical LQR controllers it will take these constraints into account in a weak sense that often is sufficiently accurate for practical purposes. The algorithm can be extended using a robust optimal control setting, where robust solutions w.r.t. to perturbations, e.g. linearization errors, are sought. The robust optimal control setting will help to further reduce the constraint violation in the MPC algorithm.

2 Virtual control regularization

Problem 2 in Section 1 is a linear quadratic control problem with control and state constraints, so directly applying an appropriate optimal control algorithm seems natural. However, owing to the linearization of the dynamics (1) and the state constraint (3) along to the reference trajectory, Problem 2 may become infeasible, especially if the initial state violates the state constraint. The virtual control approach used in this paper regularizes inconsistent problems in a way that helps to decreases the violation of the state constraints.

This technique also makes the problem accessible for a fast and reliable function space Newton method, see [2]. As a side product, the regularization method suggested below can be used within a function space SQP method to regularize inconsistent quadratic subproblems.

Various ways for regularization of state constrained problems have been suggested. Lavrientiev regularization uses the control u itself to approximate the pure state constraint by a mixed control-state constraint of type

$$s(x(t)) - \alpha \|u(t)\|^2 e \leq 0,$$

where e is a vector of all ones of appropriate dimension and $\alpha > 0$ is a regularization parameter that has to be driven to zero. However, if the state constraints cannot be met, this technique enforces potentially large controls $\|u(t)\|^2 \geq \alpha^{-1} s_i(x(t))$. $i = 1, \ldots, n_s$. This condition often is in conflict with the set constraints $u(t) \in \mathcal{U}(t)$ as often $\mathcal{U}(t) \cap \{u \mid s(x(t)) - \alpha\|u\|^2 \leq 0\}$ turns out to be empty.

For this reason, we prefer to introduce an additional so-called virtual control v to regularize the pure state constraint. This virtual control approach was suggested in [1, 4] in the context of optimal control of a state constrained elliptic PDE optimal control problem. While this additional control increases the dimension of the problem, it has the advantage that it does not directly interfere with the original control u and the resulting regularized problems are always feasible. The virtual control approach can be applied to a slightly more general problem class than Problem 2 with general linear boundary conditions of type $E_0 x(0) + E_1 x(1) = g$ instead of just initial conditions as in Problem 2 and we present it for the following problem (LQR):

$$\begin{aligned}
\text{Minimize } & \frac{1}{2} \int_0^1 x^\top Q x + 2x^\top R u + u^\top S u dt \\
\text{s.t. } & x' = Ax + Bu \text{ a.e. in } [0,1], \\
& E_0 x(0) + E_1 x(1) = g, \\
& Cx \leq d \text{ in } [0,1], \\
& u \in \mathcal{U} \text{ a.e. in } [0,1].
\end{aligned}$$

For notational convenience, we omit the explicit dependence on time and note that $Q(\cdot) \in \mathbb{R}^{n_x \times n_x}, R(\cdot) \in \mathbb{R}^{n_x \times n_u}, S(\cdot) \in \mathbb{R}^{n_u \times n_u}, A(\cdot) \in \mathbb{R}^{n_x \times n_x}, B(\cdot) \in$

$\mathbb{R}^{n_x \times n_u}, C(\cdot) \in \mathbb{R}^{n_s \times n_x}$, and $d(\cdot) \in \mathbb{R}^{n_s}$ are time dependent functions. Moreover, the matrices $E_0, E_1 \in \mathbb{R}^{n_r \times n_x}$ and the vector $g \in \mathbb{R}^{n_r}$ are given. For a regularization parameter $\alpha > 0$, LQR is embedded into a family of perturbed problems (LQR$_\alpha$) with mixed-control state constraints using the virtual control $v(\cdot) \in \mathbb{R}^{n_s}$:

$$\begin{aligned}
&\text{Minimize } \frac{1}{2}\int_0^1 x^\top Qx + 2x^\top Ru + u^\top Su\,dt + \frac{\phi(\alpha)}{2}\int_0^1 \|v\|^2 dt \\
&\text{s.t.} \qquad x' = Ax + Bu - \kappa(\alpha)\sum_{i=1}^{n_s} v_i e \text{ a.e. in } [0,1], \\
&\qquad\quad\;\; E_0 x(0) + E_1 x(1) = g, \\
&\qquad\quad\;\; Cx - \gamma(\alpha) v \le d \text{ in } [0,1], \\
&\qquad\quad\;\; u \in \mathcal{U} \text{ a.e. in } [0,1].
\end{aligned}$$

Herein, $\phi(\alpha), \kappa(\alpha)$, and $\gamma(\alpha)$ are functions to be defined later. For each $\alpha > 0$ problem LQR$_\alpha$ contains only mixed control-state constraints and can be solved by the semi-smooth Newton method in [2] provided that first-order necessary optimality conditions hold, which we will assume throughout this paper. A sufficient condition for first-order necessary optimality conditions to hold is controllability and a Slater condition. The optimality conditions for a minimizer $(\hat{x}, \hat{u}) \in L^\infty([0,1], \mathbb{R}^{n_u}) \times W^{1,\infty}([0,1], \mathbb{R}^{n_x})$ of LQR read as follows: There exist multipliers $\hat{\lambda} \in BV([0,1], \mathbb{R}^{n_x})$, $\hat{\mu} \in NBV([0,1], \mathbb{R}^{n_s})$ and $\hat{\sigma} \in \mathbb{R}^{n_r}$ such that

$$\hat{\lambda}(t) = \hat{\lambda}(0) - \int_0^t Q\hat{x} + R\hat{u} + A^\top \hat{\lambda} d\tau - \int_0^t C^\top d\hat{\mu} \tag{4}$$

$$\hat{\lambda}(0) = -E_0^\top \hat{\sigma}, \quad \hat{\lambda}(1) = E_1^\top \hat{\sigma}, \tag{5}$$

$$0 \le \left(\hat{x}^\top R + \hat{u}^\top S + \hat{\lambda}^\top B\right)(u - \hat{u}) \quad \forall u \in U, \tag{6}$$

$$0 = \int_0^1 (d - C\hat{x})^\top d\hat{\mu} \tag{7}$$

$$0 \le \int_0^1 z^\top d\mu \quad \forall z \in \{C([0,1], R^{n_s}) \mid z(\cdot) \ge 0\}. \tag{8}$$

The respective conditions for a minimizer $(\hat{x}_\alpha, \hat{u}_\alpha) \in L^\infty([0,1], \mathbb{R}^{n_u}) \times W^{1,\infty}([0,1], \mathbb{R}^{n_x})$ of LQR$_\alpha$ read as follows: There exist multipliers $\hat{\lambda}_\alpha \in W^{1,\infty}([0,1], \mathbb{R}^{n_x})$, $\hat{\eta}_\alpha \in NBV([0,1], \mathbb{R}^{n_s})$ and $\hat{\sigma}_\alpha \in \mathbb{R}^{n_r}$ such that

$$\hat{\lambda}'_\alpha = -\left(Q\hat{x}_\alpha + R\hat{u}_\alpha + A^\top \hat{\lambda}_\alpha + C^\top \hat{\eta}_\alpha\right), \tag{9}$$

$$\hat{\lambda}_\alpha(0) = -E_0^\top \hat{\sigma}_\alpha, \quad \hat{\lambda}_\alpha(1) = E_1^\top \hat{\sigma}_\alpha, \tag{10}$$

$$0 \le \left(\hat{x}_\alpha^\top R + \hat{u}_\alpha^\top S + \hat{\lambda}_\alpha^\top B\right)(u - \hat{u}_\alpha) \quad \forall u \in \mathcal{U}, \tag{11}$$

$$\hat{\eta}_\alpha \ge 0 \ , \ \hat{\eta}_\alpha(Cx - d) = 0, \tag{12}$$

$$0 = \phi(\alpha)\hat{v}_{i,\alpha} - \kappa(\alpha)\hat{\lambda}_\alpha^\top e - \gamma(\alpha)\hat{\eta}_{i,\alpha}. \tag{13}$$

The following theorem establishes a convergence result and shows how the functions ϕ, κ, γ have to be chosen. A proof can be found in the recent report [3].

Theorem 1. *Let $(\hat{x}, \hat{u}, \hat{\lambda}, \hat{\eta}, \hat{\sigma})$ and $(\hat{x}_\alpha, \hat{u}_\alpha, \hat{\lambda}_\alpha, \hat{\eta}_\alpha, \hat{\sigma}_\alpha)$ be solutions of the conditions (4)-(8) and (9)-(13), respectively. Let there be a constant $\delta > 0$, such that a.e. in $[0,1]$,*

$$\begin{pmatrix} x & u \end{pmatrix} \begin{pmatrix} Q(t) & R(t) \\ R(t)^\top & S(t) \end{pmatrix} \begin{pmatrix} x \\ u \end{pmatrix} \geq \delta \|(x,u)\|^2 \qquad \forall (x,u) \in \mathbb{R}^{n_x+n_u}.$$

Let one of the following conditions be fulfilled:

(a) $\hat{\mu} \in W^{1,2}([0,1], \mathbb{R}^{n_s})$, i.e. $\hat{\mu}, \hat{\mu}' \in L^2([0,1], \mathbb{R}^{n_s})$.
(b) $\|\hat{v}_\alpha\|_\infty \leq C_v$ for some constant C_v independent of α.

Let $\phi, \kappa, \gamma : \mathbb{R}^+ \to \mathbb{R}^+$ be such that $\phi(\alpha) \geq \tilde{\delta}$ for all $\alpha > 0$ and

$$\lim_{\alpha \to 0} \frac{\gamma(\alpha)}{\phi(\alpha)} = 0, \quad \lim_{\alpha \to 0} \frac{\kappa(\alpha)}{\phi(\alpha)} = 0, \quad \lim_{\alpha \to 0} \gamma(\alpha) = 0, \quad \lim_{\alpha \to 0} \kappa(\alpha) = 0.$$

Then,

$$\lim_{\alpha \to 0} \|(\hat{x}_\alpha - \hat{x}, \hat{u}_\alpha - \hat{u})\|_2 = 0 \qquad \text{and} \qquad \lim_{\alpha \to 0} \|\hat{v}_\alpha\|_2 = 0.$$

Remark 1. A similar result holds if the matrix $\begin{pmatrix} Q & R \\ R^\top & S \end{pmatrix}$ is just positive semidefinite and S is uniformly positive definite. In this case, $\hat{u}_\alpha$ converges to $\hat{u}$ in the L^2 norm sense.

3 Examples

We present two examples for which the following approaches are compared:

(A1) linear-quadratic MPC algorithm with a state constraint and weighting matrices Q, R, and S,
(A2) linear-quadratic MPC algorithm without state constraint and Q, R, S as in (A1),
(A3) linear-quadratic MPC algorithm without state constraint and R and S as in (A1). Q will be adapted by increasing the weight for the constrained state in order to better track the constrained reference state, which implicitly aims at reducing constraint violations.

Hence, while (A1) is the algorithm proposed in this paper, (A2) is the well-known standard LQR control, and (A3) is an LQR variation aimed at state constrained problems. The following examples have been calculated for $\kappa(\alpha) = 0$, $\phi(\alpha) = 1$, $\gamma(\alpha) = \alpha$. The algorithms have been implemented in SCILAB. The computations were done on a laptop (Dell XPS M1530) running at 2 GHz.

3.1 Inverse Pendulum

The Inverse Pendulum example is a simple representation of an inverse pendulum mounted on a cart that can be accelerated.

The non-linear dynamics on the left are linearized in the unstable equilibrium state $x_{ref} \equiv (0,0,0,0)^\top$ and $u_{ref} \equiv 0$ and lead to the linearized equations on the right:

$$\begin{aligned}
\dot{x}_1 &= x_2 & \qquad \Delta\dot{x}_1 &= \Delta x_2 \\
\dot{x}_2 &= g \sin x_1 - k x_2 + u \cos x_1 & \qquad \Delta\dot{x}_2 &= g \Delta x_1 - k \Delta x_2 + \Delta u \\
\dot{x}_3 &= x_4 & \qquad \Delta\dot{x}_3 &= \Delta x_4 \\
\dot{x}_4 &= u & \qquad \Delta\dot{x}_4 &= \Delta u
\end{aligned}$$

Here, $g = 9.81$ [m/s^2] denotes the gravitational acceleration, and $k = 1$ models the friction. In practice, the space in which the wagon can be moved is not unlimited. We account for this fact by inducing the state constraint $-0.3 \leq x_3(t) \leq 0.3$ on the system. Linearization in x_{ref} leads to

$$-0.3 \leq \Delta x_3(t) \leq 0.3. \tag{14}$$

For (A1) and (A2) we used $Q \in \mathbb{R}^{4\times 4}$ with $Q_{11} = 1$, $Q_{ij} = 0$, $(i,j) \neq (1,1)$, $S = 0.01$, $R = 0$. Note that these weights do not 'encourage' the system to move back to the center position $x_3 = 0$. Although such a behavior

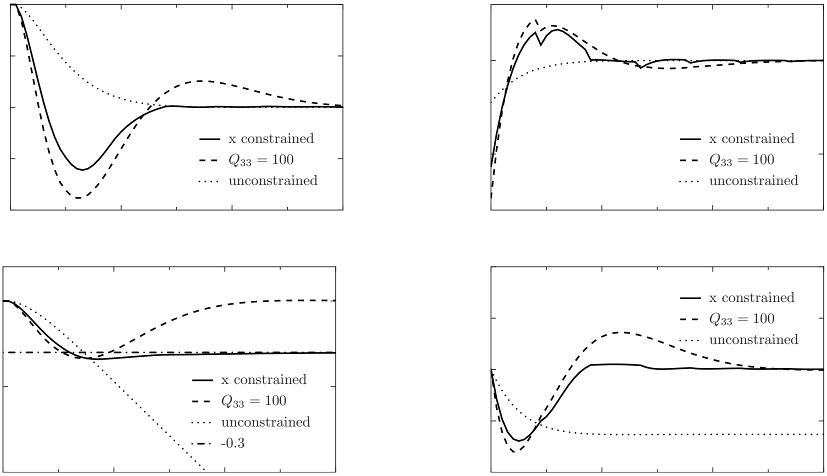


Fig. 2. Inverse Pendulum: Constrained and unconstrained algorithm

might be desirable in practice, the purpose of this example is to illustrate that satisfying the constraints can be encouraged with no further influence on the tracking goal. For (A3) we increase the weight Q_{33} to $Q_{33} = 100$. This weight 'encourages' the system to move back to the center position $x_3 = 0$.

For the simulations in Figure 2, we used $\Delta t = 1.8$ [s] (with step size $h = 0.05$ [s]), $\delta t = 0.45$ [s], and $\alpha = 0.1$.

While the unconstrained algorithm (A2) tracks the given equilibrium with the lowest control effort, the state constraint will not be obeyed. Approach (A2) works smoothly and almost satisfies the state constraint, but its performance in tracking the remaining states suffers from the altered weight matrix as the weight concentrates on tracking state x_3. The linear-quadratic MPC algorithm (A1) satisfies the state constraint even better and it also tracks the remaining states very well. The CPU time for (A1) is 0.75 [s] (at most 0.064 [s] per step) and 0.44 [s] (at most 0.04 [s] per step) for (A2) and (A3). In this example at most two Newton steps were performed, which turned out to be sufficient. The inverse pendulum was controlled for 6 [s] (the pictures only show the first 3 [s] as afterwards an equilibrium was reached).

3.2 Trolley

This is a model of a trolley carrying a freight on a rope (with a length of $l = 0.73$ [m]). The acceleration $\dot{x}_3$ of the trolley can be controlled by u. Here we make use of a reference trajectory resulting from a suitably defined optimal control problem. The task was to move the trolley (and the freight) over a total distance of one meter ($x_1(t_f) = 1$ [m] at final time t_f) in a way that is time efficient but also prevents the rope from swinging too much. The system is described by the following dynamic equations:

$$
\begin{aligned}
\dot{x}_1 &= x_3,\\
\dot{x}_2 &= x_4,\\
\dot{x}_3 &= \frac{(m_2^2 l^3 x_4^2 + m_2 I_{y_2} l x_4^2 + m_2^2 l^2 g \cos(x_2)) \sin(x_2) - (m_2 l^2 + I_{y_2}) u}{-m_1 m_2 l^2 - m_1 I_{y_2} - m_2^2 l^2 - m_2 I_{y_2} + m_2^2 l^2 \cos(x_2)^2},\\
\dot{x}_4 &= \frac{m_2 l (m_2 l \cos(x_2)^2 x_4^2 \sin(x_2) + g \sin(x_2)(m_1 + m_2) - \cos(x_2) u)}{-m_1 m_2 l^2 - m_1 I_{y_2} - m_2^2 l^2 - m_2 I_{y_2} + m_2^2 l^2 \cos(x_2)^2}.
\end{aligned}
$$

Here, x_1 and x_3 denote the distance and velocity, respectively, of the trolley, while x_2 and x_4 model the angle and the angle velocity, respectively, of the freight. We impose the state constraint $x_3 \leq 0.25$ [m/s]. The parameters m_1 and m_2 describe the masses of the trolley (0.6 [kg]) and the freight (0.62 [kg]), respectively. $I_{y_2} = 0.00248$ [kg·m^2] denotes the moment of inertia of the freight, and $g = 9.81$ [m/s^2] is the gravitational acceleration. We start the control problem with a deviation $\Delta x = (-0.05, 0, 0, 0)^\top$ of the reference trajectory. For all calculations we used a discretization with 201 data points. The LQR methods predict the model behavior over 21 points, before the next control is applied in the next six time steps.

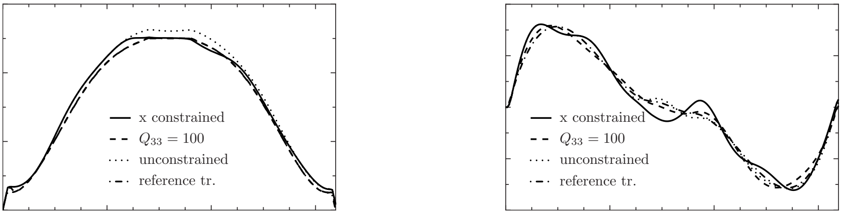


Fig. 3. Trolley: States x_3 and x_2.

For (A1) and (A2) we used the weight matrices $Q = I$, $S = I$, and $R = 0$. For (A3) we set $Q_{33} = 100$ instead. Figure 3 shows that (A1) and (A3) obey the state constraint, which is active in the middle of the time interval, while (A2) does not. The tracking behavior of the three approaches with respect to the reference angle x_2 of the freight is satisfactory for (A1)-(A3). In order to investigate the behavior of the three approaches more clearly, Figure 4 shows the displacements $\Delta x = x - x_{ref}$.

While (A3) owing to the high weight Q_{33} tracks the reference trajectory component $x_{3,ref}$ very well, the deviation in $x_{1,ref}$ stays rather large, i.e. this approach makes the trolley travel at the same speed as the reference trolley, but with a displacement in the position. Hence, the terminal condition of traveling one meter will not be met by (A3). In contrast, the linear-quadratic MPC approach (A1) obeys the state constraint, manages to decrease the deviation from the reference $x_{1,ref}$, and almost meets the terminal condition

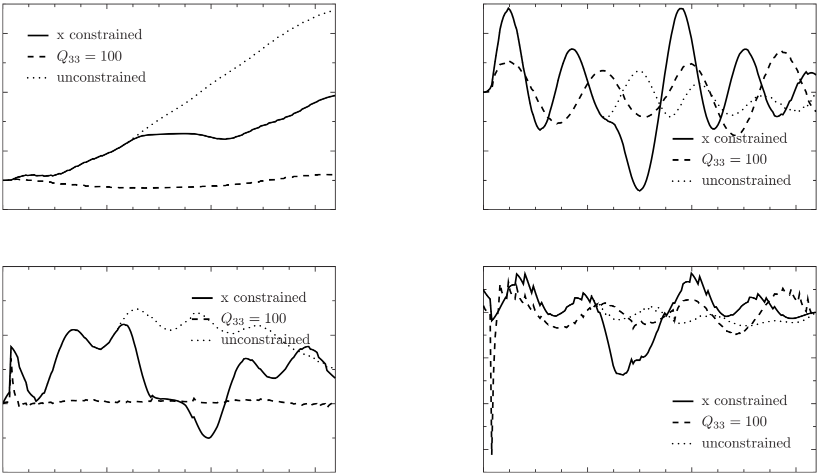


Fig. 4. Trolley: Displacements for states and control.

$x_{1,ref}(t_f) = 1$ [m]. At the same time, the deviation in $x_{2,ref}$ and u_{ref} are moderate.

The CPU time for (A1) is 1.82 [s] (at most 0.048 [s] per step) and 0.824 [s] (at most 0.022 [s] per step) for (A2) and (A3). In this example at most four Newton steps were performed and turned out to be sufficiently accurate.

References

1. Cherednichenko S, Krumbiegel K, Rösch A (2008) Error estimates for the Lavrientiev regularization of elliptic optimal control problems. Inverse Problems, 24:1–21
2. Gerdts M (2008) Global convergence of a nonsmooth Newton's method for control-state constrained optimal control problems. SIAM Jouornal on Optimization, 19(1):326–350. A corrected version can be found on http://www.mathematik.uni-wuerzburg.de/~gerdts/erratum_SIAM_19_1_2008_326-350_full.pdf.
3. Gerdts M, Hüpping B (2010) Virtual control regularization of state constrained linear quadratic optimal control problems. Technical report, University of Würzburg.
4. Krumbiegel K, Rösch A (2008) On the regularization error of state constrained Neumann control problems. Control and Cybernetics, 37(2):369–392
5. Diehl M (2001) Real-time optimization for large scale nonlinear processes. PhD thesis, Universität Heidelberg, Naturwissenschaftlich-Mathematische Gesamtfakultät, Heidelberg
6. Diehl M, Bock HG, Schlöder, JP (2003) Newton-type methods for the approximate solution of nonlinear programming problems in real-time. In: Di Pillo G (ed.) et al. High performance algorithms and software for nonlinear optimization. Selected lectures presented at the workshop, Erice, Italy, June 30–July 8, 2001. Boston, Kluwer Academic Publishers, Appl. Optim. 82:177–200
7. Diehl M, Bock HG, Schlöder, JP (2005) A real-time iteration scheme for nonlinear optimization in optimal feedback control. SIAM J. Control Optimization 43(5):1714–1736

NMPC Suboptimality Estimates for Sampled–Data Continuous Systems

Lars Grüne, Marcus von Lossow, and Karl Worthmann

Mathematical Institute, University of Bayreuth, 95440 Bayreuth, Germany
lars.gruene, marcus.vonlossow, karl.worthmann@uni-bayreuth.de

Summary. In this paper we consider unconstrained model predictive control (MPC) schemes and investigate known stability and performance estimates with respect to their applicability in the context of sampled–data systems. To this end, we show that these estimates become rather conservative for sampling periods tending to zero which is, however, typically required for sampled–data systems in order to inherit the stability behavior of their continuous–time counterparts. We introduce a growth condition which allows for incorporating continuity properties in the MPC performance analysis and illustrate its impact – especially for fast sampling.

1 Introduction

In order to deal with optimal control problems on an infinite horizon we use model predictive control (MPC). This method relies on an iterative online solution of finite horizon optimal control problems. To this end, a performance criterion is optimized over the predicted trajectories of the system. The stability and performance analysis of linear and nonlinear MPC schemes has attracted considerable attention during the last years, cf. [2, 9]. Here we consider unconstrained nonlinear MPC (NMPC) schemes which are frequently used in industrial applications, cf. [8]. These incorporate neither additional terminal constraints nor terminal costs in the finite horizon problems in order to enforce stability properties. Nevertheless, a stability analysis – based on a controllability assumption – is possible and given in [3, 5].

In the present paper we focus on sampled–data continuous systems. Typically, these require sufficiently fast sampling in order to preserve their stability properties, cf. [7]. However, the direct application of [3, 5] leads to very pessimistic performance bounds, cf. Section 4. In order to compensate for this drawback we incorporate a growth condition which reflects properties of the considered sampled–data systems in the ensuing section. Finally, we investigate qualitative and quantitative effects related to the proposed condition.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_29, © Springer-Verlag Berlin Heidelberg 2010

2 Setup and Preliminaries

We consider a nonlinear discrete time control system given by

$$x(n+1) = f(x(n), u(n)), \quad x(0) = x_0 \tag{1}$$

with $x(n) \in X$ and $u(n) \in U$ for $n \in \mathbb{N}_0$. Here the state space X and the control value space U are arbitrary metric spaces. We denote the space of control sequences $u : \mathbb{N}_0 \to U$ by $\mathcal{U}$ and the solution trajectory for given $u \in \mathcal{U}$ by $x_u(\cdot)$. A typical class of such discrete time systems are sampled–data systems induced by a controlled — finite or infinite dimensional — differential equation with sampling period $T > 0$, see Section 4 for details.

Our goal consists of minimizing the infinite horizon cost $J_\infty(x_0, u) = \sum_{n=0}^{\infty} l(x_u(n), u(n))$ with running cost $l : X \times U \to \mathbb{R}_0^+$ by a static state feedback control law $\mu : X \to U$ which is applied according to the rule $x_\mu(0) = x_0$,

$$x_\mu(n+1) = f(x_\mu(n), \mu(x_\mu(n))). \tag{2}$$

We denote the optimal value function for this problem by $V_\infty(x_0) := \inf_{u \in \mathcal{U}} J_\infty(x_0, u)$. Since infinite horizon optimal control problems are in general computationally intractable, we use a receding horizon approach in order to compute an approximately optimal controller. To this end, we consider the finite horizon functional

$$J_N(x_0, u) = \sum_{n=0}^{N-1} l(x_u(n), u(n)) \tag{3}$$

with *optimization horizon* $N \in \mathbb{N}_{\geq 2}$ inducing the optimal value function

$$V_N(x_0) = \inf_{u \in \mathcal{U}} J_N(x_0, u). \tag{4}$$

By solving this finite horizon optimal control problem we obtain N control values $u^*(0), u^*(1), \ldots, u^*(N-1)$ which depend on the state x_0. Implementing the first element of this sequence, i.e., $u^*(0)$, yields a new state $x(1)$. Iterative application of this construction provides a control sequence on the infinite time interval. We obtain a closed loop representation by applying the map $\mu_N : X \to U$ which is given in Definition 1 as a static state feedback law.

Definition 1. *For $N \in \mathbb{N}_{\geq 2}$ we define the MPC feedback law $\mu_N(x_0) := u^\star(0)$, where $u^\star$ is a minimizing control for (4) with initial value x_0.*

Remark 1. For simplicity of exposition we assume that the infimum in (4) is a minimum, i.e., that a minimizing control sequence u^* exists.

In this paper we consider the conceptually simplest MPC approach imposing neither terminal costs nor terminal constraints. In order to measure the suboptimality degree of the MPC feedback for the infinite horizon problem we define

$$V_\infty^\mu(x_0) := \sum_{n=0}^{\infty} l(x_\mu(n), \mu(x_\mu(n))).$$

3 Controllability and performance bounds

In this section we introduce an exponential controllability assumption and deduce several consequences for our optimal control problem. In order to facilitate this relation we will formulate our basic controllability assumption not in terms of the trajectory but in terms of the running cost l along a trajectory. To this end, we define $l^\star(x) := \min_{u\in U} l(x,u)$.

Property 1. Assume *exponential controllability* with overshoot bound $C \geq 1$ and decay rate $\sigma \in (0,1)$, i.e., for each $x_0 \in X$ there exists a control function $u_{x_0} \in \mathcal{U}$ satisfying the estimate

$$l(x_{u_{x_0}}(n), u_{x_0}(n)) \leq C\sigma^n l^\star(x_0) \qquad \text{for all } n \in \mathbb{N}_0. \tag{5}$$

Based on Property 1 and Bellman's optimality principle an optimization problem is derived in [3] whose solution, which depends on the optimization horizon N, coincides with the parameter α_N in the relaxed Lyapunov inequality $V_N(f(x,\mu_N(x))) \leq V_N(x) - \alpha_N l(x,\mu_N(x))$. As a consequence the estimate

$$\alpha_N V_\infty(x) \leq \alpha_N V_\infty^{\mu_N}(x) \leq V_N(x) \tag{6}$$

holds for all $x \in X$. Hence, α_N specifies a suboptimality degree. For details we refer to [1]. Since we focus on the stability behavior of systems satisfying (5), i.e. exponential controllability, it is possible to calculate this performance index α_N explicitly, cf. [5, section 5].

Theorem 1. *Assume Property 1 and let the optimization horizon N be given. Then we obtain for the suboptimality degree α_N from* (6) *the formula*

$$\alpha_N = 1 - \frac{(\gamma_N - 1)\prod_{i=2}^{N}(\gamma_i - 1)}{\prod_{i=2}^{N}\gamma_i - \prod_{i=2}^{N}(\gamma_i - 1)} \qquad \textit{with} \quad \gamma_i := C\sum_{n=0}^{i-1}\sigma^n = C\frac{1-\sigma^i}{1-\sigma}. \tag{7}$$

Remark 2. Theorem 1 is generalizable to functionals including an additional weight on the final term. This may enhance the stability behavior of the underlying system significantly. Moreover, it remains valid for more general controllability assumptions, for instance, *finite time controllability* with linear overshoot, cf. [5, Sections 5 and 8] for details.

Remark 3. Theorem 1 is also applicable in the context of networked control systems which require the implementation of more than only the first element of the obtained sequence of control values, cf. [6] for details.

4 Sampled–data systems and arbitrary fast sampling

Given a continuous time control system governed by the differential equation $\dot{\varphi} = g(\varphi(t), \tilde{u}(t))$, we assume exponential controllabilty, i.e., that for each $x_0 \in X$ there exists a control function $\tilde{u}_{x_0}(\cdot)$ such that

$$l(\varphi(t; x_0, \tilde{u}_{x_0}), \tilde{u}_{x_0}(t)) \leq Ce^{-\lambda t} l^*(x_0) \tag{8}$$

holds almost everywhere for given overshoot $C \geq 1$ and decay rate $\lambda > 0$. Here $\varphi(t; x_0, \tilde{u})$ denotes the solution of the respective control system. In order to analyze the stability behavior, we define the discrete time system (1) by $f(x, u) := \varphi(T; x, \tilde{u})$ with discretization parameter $T > 0$. Consequently, the assumed exponential controllability of the continuous time system implies (5) in the discrete time setting, i.e., Property 1 with $\sigma = e^{-\lambda T}$ for an appropriately chosen control value space. Moreover, we fix the continuous time optimization interval $[0, t_F)$ which corresponds to an optimization horizon of length $N = t_F/T$ in the discrete time setting.

A typical representative of this class are sampled–data systems with sampling period $T_0 := T$ and piecewise constant control, i.e., $\tilde{u}(t) = u$ for all $t \in [0, T_0)$. However, sampled–data systems require sufficiently fast sampling in order to inherit the stability behavior from (8), cf. [7]. Consequently, it may be necessary to increase the sampling rate, i.e., using smaller sampling periods. In this section we focus on effects caused by this adjustment. Thus, we reduce the discretization parameter of the discrete time model along with the sampling rate of the sampled–data system in consideration.

In order to investigate this issue systematically, we consider the sequence of sampling periods $T_0, T_0/2, T_0/4, \ldots$, i.e., $T_k = 2^{-k}T_0$. This determines the optimization horizons $N_0, 2N_0, 4N_0, \ldots$, i.e. $N_k = 2^k N_0$, for the discrete time system because we have fixed the optimization interval $[0, t_F)$ and coupled the discretization parameter with the sampling period. The corresponding decay rate from (8) is $\sigma_k = e^{-\lambda T_k}$, cf. Figure 1 on the left. Hence, we consider the sequence

$$(T_k,\, N_k,\, \sigma_k)_{k \in \mathbb{N}_0} = (2^{-k}T_0,\, 2^k N_0,\, e^{-\lambda T_k})_{k \in \mathbb{N}_0} \tag{9}$$

of parameter combinations consisting of sampling period, optimization horizon, and decay rate. Note that the interval $[0, T_k)$ on which the first element of the calculated control value sequence is applied scales down as well.

We require the following definition.

Definition 2. *Let $C \geq 1$, $\lambda > 0$, and $T_0 > 0$ be given. In addition, we set $\sigma_k := e^{-\lambda(2^{-k}T_0)}$. Then we define*

$$\gamma_i^k := \sum_{n=0}^{i-1} Ce^{-\lambda(2^{-k}T_0)n} = C \sum_{n=0}^{i-1} \sigma_k^n = \frac{C(1 - \sigma_k^i)}{1 - \sigma_k}. \tag{10}$$

Remark 4. Note that we use an additional index in order to indicate the dependence of γ_i^k on σ_k. For $k = 0$ we obtain exactly γ_i from (7). Moreover, the relation $\sigma_k = \sqrt{\sigma_{k-1}} = \sigma_0^{2^{-k}}$ holds.

Theorem 2 shows that the suboptimality estimates from Theorem 1 become arbitrarily bad for sampling periods tending to zero, cf. Figure 1. In order to compensate this drawback for sampled-data continuous systems we introduce an appropriate condition in the ensuing section.

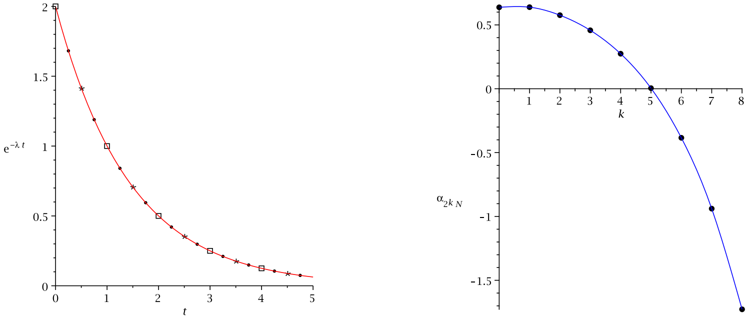


Fig. 1. Visualization of the bounds induced by our controllability assumption for $(2^{-k}T_0,\, 2^k N_0,\, e^{-\lambda(2^{-k}T_0)})_{k\in\mathbb{N}_0}$ with $T_0 = 1$, $N_0 = 8$, $\lambda = -\ln(1/2)$, and $C = 2$ for $k = 0, 1, 2$ ($\square$, $*$, $\cdot$) on the left. On the right we depict the suboptimality estimates $\alpha^k_{N_k}$, $k = 0, 1, 2, \ldots, 8$, from Theorem 2 for this sequence.

Theorem 2. *Assume* (8) *and let* $N_0 := N \in \mathbb{N}_{\geq 2}$, $T_0 > 0$ *be given. Then the suboptimality bounds corresponding to the sequence* $(T_k,\, N_k,\, \sigma_k)_{k\in\mathbb{N}_0} = (2^{-k}T_0,\, 2^k N_0,\, e^{-\lambda(2^{-k}T_0)})_{k\in\mathbb{N}_0}$ *diverge to* $-\infty$, *i.e.,*

$$\alpha^k_{N_k} = 1 - \frac{(\gamma^k_{N_k} - 1)\prod_{i=2}^{N_k}(\gamma^k_i - 1)}{\prod_{i=2}^{N_k}\gamma^k_i - \prod_{i=2}^{N_k}(\gamma^k_i - 1)} \longrightarrow -\infty \qquad \textit{for} \qquad k \to \infty \tag{11}$$

with γ^k_i *from Definition 2.*

Proof. Since $\prod_{i=2}^{2^k N}\gamma^k_i \geq \prod_{i=2}^{2^k N}(\gamma^k_i - 1) \geq 0$ proving the assertion follows from

$$0 \leq \frac{1}{\gamma^k_{2^k N} - 1} \cdot \prod_{i=2}^{2^k N} \frac{\gamma^k_i}{\gamma^k_i - 1} \overset{k\to\infty}{\longrightarrow} 0. \tag{12}$$

In order to estimate (12) we establish the inequalities

$$\frac{1}{\gamma^k_{2^k N} - 1} \leq \frac{1 - \sigma_k}{C_1} \qquad \text{and} \qquad \prod_{i=2}^{2^k N} \frac{\gamma^k_i}{\gamma^k_i - 1} \leq C_0 (2^{1/C})^k \tag{13}$$

with $C_0 := \sigma_0^{-N/C} \prod_{i=2}^{N} \frac{iC}{iC-1}$ and $C_1 := C(1-\sigma_0^N)-1+\sigma_0$. Note that C_0 and C_1 do not depend on k. The first inequality is directly implied by Definition 2. In order to show the second we prove the inequality

$$\frac{\gamma_i^k}{\gamma_i^k - 1} = \frac{C}{C-1+\sigma_k} \frac{(1-\sigma_k^i)(C-1+\sigma_k)}{C-1+\sigma_k - C\sigma_k^i} \leq \frac{C}{C-1+\sigma_k} \cdot \frac{iC}{iC-1}$$

which is equivalent to $i\sigma_k^i C(1-\sigma_k) \leq (C-1+\sigma_k)(1-\sigma_k^i)$, $k \in \mathbb{N}_0$ and $i \in \mathbb{N}_{\geq 1}$. Since $C\sigma_k/(C-1+\sigma_k) \leq 1$ this is shown by $i\sigma_k^{i-1} \leq \sum_{n=0}^{i-1} \sigma_k^n = (1-\sigma_k^i)/(1-\sigma_k)$. Moreover, we require the inequality

$$\left(\frac{C}{C-1+\sigma_k}\right)^{2^k N} \leq \sigma_0^{-N/C} \tag{14}$$

which is – in consideration of Definition 2 – equivalent to $f(\sigma_k) := C - C\sigma_k^{1/C} - 1 + \sigma_k \geq 0$. However, since $f(0) = C-1 \geq 0$ and $f(1) = 0$ the inequality $f'(\sigma_k) = 1 - \sigma_k^{-(C-1)/C} \leq 0$ implies (14).

Hence, taking into account that the factor $C/(C-1+\sigma_k)$ is independent of the control variable i and applying the two deduced estimates leads to

$$\prod_{i=2}^{2^k N} \frac{\gamma_i^k}{\gamma_i^k - 1} < \sigma_0^{-N/C} \cdot \prod_{i=2}^{2^k N} \frac{iC}{iC-1} = C_0 \prod_{j=0}^{k-1} \left(\prod_{i=2^j N+1}^{2^{j+1} N} \frac{iC}{iC-1} \right) \tag{15}$$

for $k \in \mathbb{N}_0$. Thus, it suffices to estimate the expression in brackets uniformly from above by $2^{1/C}$ for $j \in \mathbb{N}_{\geq 0}$ in order to show (13).

In the following, we use the functional equation, i.e., $\Gamma(x+1) = x\,\Gamma(x)$ and $\Gamma(1) = 1$, for the gamma function $\Gamma(\cdot)$ which is connected to the beta function $B(\cdot,\cdot)$ via the formula

$$B(x,y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}, \tag{16}$$

cf. [11, p.442]. Moreover, we require the equation

$$B(p, p+s) = \frac{B(p,p)}{2^s} \left(1 + \frac{s(s-1)}{2(2p+1)} + \frac{s(s-1)(s-2)(s-3)}{2 \cdot 4 \cdot (2p+1) \cdot (2p+3)} + \ldots \right) \tag{17}$$

which holds for $p > 0$, $p + s > 0$ according to [10, p.262] in order to estimate the term in brackets from (15) as follows

$$\begin{aligned}
\prod_{i=2^k N+1}^{2^{k+1} N} \frac{iC}{iC-1} &= \prod_{i=2^k N+1}^{2^{k+1} N} \frac{i}{i - \frac{1}{C}} = \frac{(2^{k+1}N)!}{(2^k N)!} \left(\prod_{i=2^k N+1}^{2^{k+1} N} i - \frac{1}{C} \right)^{-1} \\
&= \frac{\Gamma(2^{k+1}N+1)}{\Gamma(2^k N+1)} \cdot \frac{\Gamma(2^k N + 1 - \frac{1}{C})}{\Gamma(2^{k+1} N + 1 - \frac{1}{C})} \\
&\overset{(16)}{=} \frac{B(2^k N, 2^k N + \frac{C-1}{C})}{B(2^k N, 2^k N + 1)} \\
&\overset{(17)}{=} 2^{1/C} \left(1 + \frac{s(s-1)}{2(2p+1)} + \frac{s(s-1)(s-2)(s-3)}{2 \cdot 4 \cdot (2p+1) \cdot (2p+3)} + \ldots \right)
\end{aligned}$$

with $s = (C-1)/C \in [0,1)$ and $p = 2^k N$. Since $s \in [0,1)$ the term in brackets is less or equal to one. Hence, we obtain the desired estimate (13).

Thus, it suffices to show $(2^{1/C})^k(1-\sigma_k) \to 0$ as k approaches infinity in order to complete the proof. To this aim, we define $a_k := (2^{1/C})^k(1-\sigma_k)$ and show that the quotient a_{k+1}/a_k converges to $2^{1/C}/2$ for $k \to \infty$:

$$\frac{a_{k+1}}{a_k} = \frac{1-\sigma_{k+1}}{1-\sigma_k} 2^{1/C} = \frac{(1-\sigma_{k+1})2^{1/C}}{(1-\sigma_{k+1})(1+\sigma_{k+1})} = \frac{2^{1/C}}{1+\sigma_0^{2^{-(k+1)}}} \overset{k\to\infty}{\longrightarrow} 2^{1/C}/2.$$

Thus, there exists k^* such that the considered quotient a_{k+1}/a_k is less or equal $\theta := (2+2^{1/C})/4 < 1$ for all $k \geq k^*$. This implies the convergence of $a_k = 2^{1/C}(1-\sigma_k)$ to zero for k approaching infinity.

5 Growth condition and analytic formula

Although the estimate stated in Theorem 1 is strict for the whole class of systems satisfying the assumed controllability condition, cf. [3, Theorem 5.3], it may be conservative for subsets of this class. For instance, for sampled–data continuous time systems the difference between $x(n+1)$ and $x(n)$ is usually of order $\mathcal{O}(T)$, a property which is not reflected in the optimization problem on which Theorem 1 is based on. Neglecting this leads to very pessimistic estimates if the sampling period T tends to 0 and the continuous time optimization horizon $H = [0, t_F)$ is fixed, cf. Section 4.

In order to compensate for this drawback, we incorporate a growth condition in our suboptimality estimate.

Property 2. For each $x_0 \in X$ there exists a control function $\tilde{u}_{x_0}(\cdot) \in \mathcal{U}$ such that

$$l(\varphi(t; x_0, \tilde{u}_{x_0}), \tilde{u}_{x_0}(t)) \leq e^{L_c t} l^*(x_0) \qquad \text{for all } t \geq 0 \tag{18}$$

with constant $L_c > 0$ which is independent of the chosen state x_0. Let a discretization parameter $T > 0$ be given and define the discrete time system by $f(x, \tilde{u}) = \varphi(T; x, \tilde{u})$ with an appropriately chosen control value space U. Then – as a consequence from (18) – the inequality

$$J_{N-k}(x_{\tilde{u}_{x_0}}(k), \tilde{u}_{x_0}(k+\cdot)) \leq l^*(x_{\tilde{u}_{x_0}}(k)) \sum_{n=0}^{N-k-1} L^n$$

holds with $L = e^{L_c T} > 1$ for $k = 0, \ldots, N-1$.

In combination with our controllability property (8) this leads to the definition

$$\gamma_i := \min\left\{ C \sum_{n=0}^{i-1} \sigma^n,\ \sum_{n=0}^{i-1} L^n \right\} \tag{19}$$

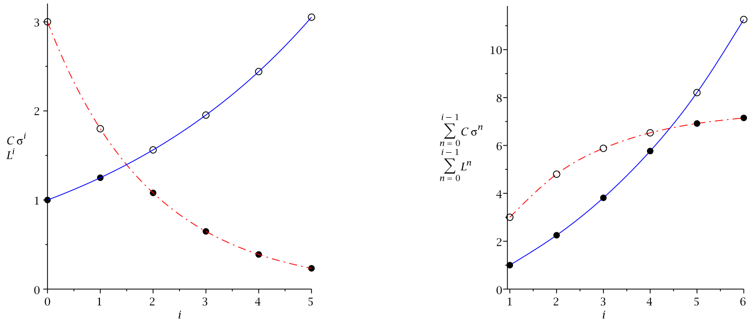


Fig. 2. Visualization of the bounds induced by our controllability assumption (dashed-dotted line) and our growth condition (solid line) for $C = 3$, $\sigma = 3/5$, and $L = 5/4$. Each time the minimum is marked with solid circles. The solid circles on the right coincide with γ_i from (19)

with $\sigma := e^{-\lambda T}$ and L from Property 2. Thus, we obtain tighter bounds with respect to the stage costs where the introduced growth condition is applicable in contrast to γ_i from (7), cp. Figure 2.

Theorem 1 remains valid if we substitute the definition of γ_i in (7) by (19).

Theorem 3. *Assume exponential controllability and our growth condition, i.e., Properties 1 and 2, with parameters $\sigma \in (0,1)$, $C \geq 1$, and $L \geq 1$ then we obtain for given optimization horizon N Formula* (7) *with γ_i from* (19)*.*

Proof. Sifting through the proof of Theorem 1 one notices that changing the definition of γ_i to (19) does not affect the part of the proof in which (7) is established as the solution of the relaxed optimization problem, cf. [5, Problem 5.3]. However, we have to show the inequality

$$(\gamma_2 - 1) \prod_{i=3}^{N-j+1} (\gamma_i - 1) \geq (\gamma_{N-j+1} - \gamma_{N-j}) \prod_{i=2}^{N-j} \gamma_i, \qquad j = 1, \ldots, N-2,$$

which implies [5, Inequality (5.8)] for $m = 1$, $\omega = 1$ and – as a consequence – ensures that Formula (7) provides the solution of the respective optimization problem.

Moreover, note that there exists exactly one index $i^\star \in \mathbb{N}_{\geq 1}$ such that $\gamma_{i^\star} = \sum_{n=0}^{i^\star - 1} L^n$ and $\gamma_{i^\star + 1} < \sum_{n=0}^{i^\star} L^n$. $n^\star \geq N - j + 1$ corresponds to $C := L \geq 1$ and $\sigma := 1$. However, since [5] shows the desired inequality for arbitrary $\sigma \in (0,1)$ this situation is covered. $n^\star = N - j$ is also trivial, since we may estimate $\gamma_{N-j+1} \leq \sum_{n=0}^{N-j} L^n$. Thus, $\gamma_{N-j+1} = \gamma_{N-j} + C\sigma^{N-j} = C \sum_{n=0}^{N-j} \sigma^n$ holds. We rewrite the above inequality as

$$(C-1) \prod_{i=2}^{N-j} (\gamma_i - 1) + C \prod_{i=2}^{N-j} (\gamma_i - 1) \sum_{n=1}^{N-j} \sigma^n \geq C\sigma^{N-j} \prod_{i=2}^{N-j} \gamma_i.$$

Consequently, it suffices to show $\prod_{i=2}^{N-j}(\gamma_i - 1)\sum_{n=1}^{N-j}\sigma^n \geq \sigma^{N-j}\prod_{i=2}^{N-j}\gamma_i$ which can be done by induction. The induction start $j = N-2$ is $(\gamma_2 - 1)(\sigma+\sigma^2) \geq \sigma^2\gamma_2$ or equivalently $\sigma(\gamma_2 - (1+\sigma)) \geq 0$ which holds due to the definition of γ_2. The induction step from $j+1 \rightsquigarrow j$ holds since the desired inequality may be written as

$$\prod_{i=2}^{N-\bar{j}}(\gamma_i - 1)\left[\sigma\gamma_{N-j} - \sum_{n=1}^{N-j}\sigma^n\right] + \sigma\gamma_{N-j}\left[\prod_{i=2}^{N-\bar{j}}(\gamma_i - 1)\sum_{n=1}^{N-\bar{j}}\sigma^n - \sigma^{N-\bar{j}}\prod_{i=2}^{N-\bar{j}}\gamma_i\right] \geq 0.$$

with $\bar{j} := j+1$.

Remark 5. Conditions which guarantee Property 2 can be found in [4].

6 Numerical Examples

We have observed that sampling periods tending to zero cause serious problems in applying our estimates from Theorem 1, cf. Figure 1. In order to compensate for this drawback we introduced Property 2 for sampled–data continuous time systems and generalized our results to this setting, cf. Theorem 3. This justifies the application of Formula (7) in consideration of the imposed growth condition and enables us to analyze its impact.

Again, we fix the continuous time optimization interval $[0, t_F)$ and consider sequence (9). However, we assume – in addition to (8) – Property 2. As a consequence, we obtain the suboptimality bounds from Formula (7) with

$$\gamma_i^k := \min\left\{C\sum_{n=0}^{i-1}\sigma_k^n,\ \sum_{n=0}^{i-1}L_k^n\right\} \tag{20}$$

with $\sigma_k := e^{-\lambda T_k} = e^{-\lambda(2^{-k}T_0)}$ and $L_k := e^{L_c T_k} = e^{L_c(2^{-k}T_0)}$. As shown by Figure 3, our continuity condition counteracts occurring problems in connection with arbitrary fast sampling.

Next, we consider quantitative effects related to Property 2. Since the overshoot C has been proven to be the decisive parameter in order to establish stability, cf. [3, section 6], [5, section 6], we investigate its sensitivity to changes in the growth constant L. To this aim, we fix the decay rate $\sigma = 0.7$. Our goal consists of determining the maximal overshoot C which allows for guaranteeing stability for the whole class of systems, i.e., $\alpha_N \geq 0$, for a given optimization horizon N. Neglecting our growth condition yields the values 1.8189, 2.0216, 2.2208 for $N = 8$, 10, and 12 respectively. Whereas Figure 3 shows that Property 2 allows for significantly larger values for C. The impact of our growth condition remains basically the same for $\alpha_N > 0$, i.e., if we do not only aim at ensuring stability, but also set performance specification on our calculated NMPC-Feedback.

Hence, Property 2 allows for calculating tighter bounds, i.e. larger α_N values, and consequently a more accurate characterization of the system's behavior.

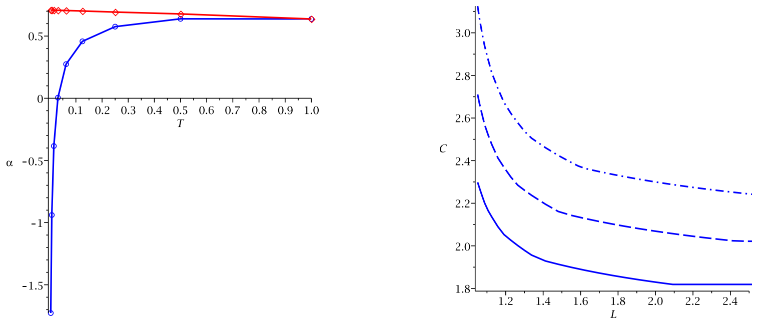


Fig. 3. On the left we depict the suboptimality estimates obtained from (7) with γ_i^k from (20) with ($\diamond$) and without ($\circ$) our growth condition (2) in dependence on the sampling period T. The parameters are the same as used for Figure 1. On the right the maximal feasible overshoot C in dependence of our growth constant L is presented for given overshoot σ and optimization horizon $N = 8$ (solid), 10 (dashed), and 12 (dash–dotted) respectively for which Theorem 3 guarantees stability, i.e., $\alpha_N \geq 0$.

References

1. Altmüller N., Grüne L., Worthmann K. (2009), Performance of NMPC schemes without stabilizing terminal constraints, submitted to BFG 09.
2. Allgöwer F., Zheng A., eds. (2000), Nonlinear model predictive control, Birkhäuser, Basel.
3. Grüne L. (2009) Analysis and design of unconstrained nonlinear MPC schemes for finite and infinite dimensional systems, SIAM J. Control Optim., 48, pp. 1206–1228.
4. Grüne L., von Lossow M., Pannek J., Worthmann K. (2010), MPC: implications of a growth condition on exponentially controllable systems, submitted.
5. Grüne L., Pannek J., Seehafer M., Worthmann K. (2009), Analysis of unconstrained nonlinear MPC schemes with time varying control horizon, submitted.
6. Grüne L., Pannek J., Worthmann K. (2009), A networked unconstrained nonlinear MPC scheme, Proceedings of the European Control Conference 2009, Budapest, Hungary, pp. 371–376.
7. Nešić D., Teel A.R. (2004), A framework for stabilization of nonlinear sampled-data systems based on their approximate discrete-time models, IEEE Trans. Automat. Control, 49 (7), pp. 1103–1122.
8. Qin S., Badgwell T. (2003), A survey of industrial model predictive control technology, Control Engineering Practice, 11, pp. 733-764.
9. Rawlings J.B., Mayne D.Q. (2009), Model Predictive Control: Theory and Design, Nob Hill Publishing, Madison.
10. Whittaker E.T., Watson G.N. (1929), A course of Modern Analysis, Cambridge University Press, New York, 4th edition.
11. Zorich V.A. (2004), Mathematical Analysis II, Springer, Berlin Heidelberg.

Efficient Numerics for Nonlinear Model Predictive Control

Christian Kirches[1], Leonard Wirsching[1], Sebastian Sager[1], and Hans Georg Bock[1]

Interdisciplinary Center for Scientific Computing (IWR)
Ruprecht–Karls–Universität Heidelberg,
Im Neuenheimer Feld 368, 69120 Heidelberg, Germany
{christian.kirches|leonard.wirsching|
sebastian.sager|bock}@iwr.uni-heidelberg.de

Summary. We review a closely connected family of algorithmic approaches for fast and real–time capable nonlinear model predictive control (NMPC) of dynamic processes described by ordinary differential equations or index-1 differential-algebraic equations. Focusing on active–set based algorithms, we present emerging ideas on adaptive updates of the local quadratic subproblems (QPs) in a multi–level scheme. Structure exploiting approaches for the solution of these QP subproblems are the workhorses of any fast active–set NMPC method. We present linear algebra tailored to the QP block structures that act both as a preprocessing and as block structured factorization methods.

1 Introduction

Nonlinear model predictive control has become an increasingly popular control approach, and is both theoretically and computationally well-established. However, its application to time-critical systems requiring fast feedback is still a major computational challenge. We review a closely connected family of algorithmic approaches for fast and real–time capable NMPC of dynamic processes described by ordinary differential equations (ODEs) or differential-algebraic equations (DAEs). We start with the discretization of the optimal control problems (OCPs), focus on active–set based algorithms for the solution of the resulting nonlinear programs (NLPs), and present emerging ideas on adaptive updates of the local quadratic subproblems in a multi–level scheme. Structure exploiting approaches for the solution of these QP subproblems are the workhorses of any fast active–set NMPC method. Here, we present linear algebra tailored to the QP block structures that act both as a preprocessing and as block structured factorization methods. An introduction to a new block structured active set QP method concludes our review.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_30, © Springer-Verlag Berlin Heidelberg 2010

1.1 Direct Optimal Control in Nonlinear Model Predictive Control

We consider the following class of optimal control problems which typically arise in nonlinear model predictive control.

$$\begin{aligned}
\min_{x(\cdot),u(\cdot)} \quad & J(x(t),u(t);p) = \int_{t_0}^{t_\mathrm{f}} L\left(x(t),u(t);p\right)\ \mathrm{d}t + E\left(x(t_\mathrm{f};p)\right) && \text{(1a)}\\
\text{s.t.} \quad & x(t_0) = x_0, && \text{(1b)}\\
& \dot{x}(t) = f\left(t,x(t),u(t);p\right), \qquad \forall t \in [t_0,t_\mathrm{f}] && \text{(1c)}\\
& 0 \le h_\mathrm{path}\left(x(t),u(t);p\right), \qquad \forall t \in [t_0,t_\mathrm{f}] && \text{(1d)}\\
& 0 \le h_\mathrm{end}(x(t_\mathrm{f});p). && \text{(1e)}
\end{aligned}$$

The OCPs are formulated on a fixed and finite time horizon $\mathcal{T} := [t_0, t_\mathrm{f}]$ which is called the *prediction horizon*. We denote by $x(t) \in \mathbb{R}^{n_\mathrm{x}}$ the state vector of the dynamic process, and by $u(t) \in \mathbb{R}^{n_\mathrm{u}}$ the vector of continuous controls influencing the dynamic process. In the following, we drop the explicit time dependency and write x and u as shorthands for $x(t)$ and $u(t)$.

The state trajectory is determined by the initial value problem (IVP) (1b)-(1c), where x_0 is the current state of the process and $f\left(t,x(t),u(t);p\right)$ describes the dynamic process model. In this paper we consider process models described by ordinary differential equations to keep the presentation clear. However, the approach can naturally be extended to models described by differential-algebraic equations (see [22]). States and controls may be subject to path constraints $h_\mathrm{path}\left(x(t),u(t);p\right)$ and the final state may be restricted by an end-point constraint $h_\mathrm{end}(x(t_\mathrm{f});p)$.

The objective function is of Bolza type with a Lagrange term $L\left(x,u;p\right)$ and a Mayer term $E\left(x(t_\mathrm{f});p\right)$. An important and frequently occurring choice for the Lagrange term are *least-squares* objective functions of the form

$$L\left(x,u;p\right) = \|l(x,u;p)\|_2^2, \tag{2}$$

where l is the least-squares residual vector. A typical example is the *tracking-type objective*

$$L\left(x,u;p\right) = \left(x-\bar{x}\right)^T Q(t)\left(x-\bar{x}\right) + \left(u-\bar{u}\right)^T R(t)\left(u-\bar{u}\right), \tag{3}$$

with $\bar{x}$ and $\bar{u}$ are given reference trajectories for x and u, and $Q(t)$ and $R(t)$ are suitable positive definite weighting matrices. A typical choice for the Mayer term is the quadratic cost

$$E\left(x(t_\mathrm{f});p\right) = \left(x(t_\mathrm{f}) - \bar{x}(t_\mathrm{f})\right)^T P\left(x(t_\mathrm{f}) - \bar{x}(t_\mathrm{f})\right), \tag{4}$$

with a suitable weighting matrix P. The Mayer term can be used — typically in conjunction with the end–point constraint h_end — to design feedback control schemes that guarantee stability of the closed-loop system. For a detailed discussion of nominal stability for NMPC see, e.g., [24].

The problem may also depend on time-independent model parameters $p \in \mathbb{R}^{n_{\mathrm{p}}}$, but they are not included as degrees of freedom for the optimization. In practice, it may happen that some of the parameters change their value during the runtime of the process. This gives rise to the important area of online state and parameter estimation (see [27, 11]). However, in this work we assume the parameters to be known and constant over time, and we will drop them in the following presentation.

1.2 The Principle of Model Predictive Control

Model predictive control schemes generate feedback by repetitively performing the following actions:

1. Obtain the process state x_0 at the current sampling time t_0.
2. Solve OCP (1) for the current x_0 to obtain optimal state and control trajectories $x^\star(\cdot; x_0)$ and $u^\star(\cdot; x_0)$.
3. Feed back the first part of $u^\star(\cdot; x_0)$ as feedback control to the process during the current sampling period $[t_0, t_0 + \delta]$.

Advantages of this approach are the possibility to use a sophisticated process model to predict the behavior of the process, the flexibility in the choice of an optimization criterion and a natural incorporation of the process constraints.

However, solving an OCP for each sampling time is computationally challenging. The fact that OCP (1) depends parametrically on x_0 has to be exploited by carefully using the results from the last problem to solve the current problem.

1.3 Direct Multiple Shooting Discretization

Approaches to solve OCP (1) divide up in *indirect* methods which first set up optimality conditions for the OCP and then discretize and solve these conditions (see [8]) and *direct* methods which first discretize the OCP and then setup und solve optimality conditions for the arising nonlinear program. In this work, we will consider the *Direct Multiple Shooting* method, first described by [26] and [7] and extended in a series of subsequent works (see, e.g., [23]). With the optimal control software package MUSCOD-II an efficient implementation of this method is available. For the use of other direct methods such as *Single Shooting* and *Collocation* in the context of online optimization we refer to the recent survey [10] and the references therein.

For a suitable partition of the horizon $[t_0, t_{\mathrm{f}}]$ into N subintervals $[t_i, t_{i+1}]$, $0 \leq i < N$, we set

$$u(t) = \varphi_i(t, q_i), \quad \text{for } t \in [t_i, t_{i+1}] \tag{5}$$

where φ_i are given basis functions parametrized by a finite dimensional parameter vector q_i. The functions φ_i may be for example vectors of polynomials; a common choice for NMPC are piecewise constant controls

$$\varphi_i(t, q_i) = q_i \quad \text{for } t \in [t_i, t_{i+1}]. \tag{6}$$

Note that for this particular choice of basis functions bounds on the control u transfer immediately to bounds on the parameter vectors q_i and vice versa.

Furthermore, we introduce additional variables s_i that serve as initial values for computing the state trajectories independently on the subintervals

$$\dot{x}_i(t) = f(t, x_i(t), \varphi_i(t, q_i)),\ x_i(t_i) = s_i,\ t \in [t_i, t_{i+1}],\ 0 \le i < N.$$

To ensure continuity of the optimal trajectory on the whole interval $[t_0, t_\mathrm{f}]$ we add matching conditions to the optimization problem

$$s_{i+1} = x_i(t_{i+1}; t_i, s_i, q_i),\ 0 \le i < N \tag{7}$$

where $x_i(t; t_i, s_i, q_i)$ denotes the solution of the IVP on $[t_i, t_{i+1}]$, depending on s_i and q_i. This method allows using state-of-the-art adaptive integrators for function and sensitivity evaluation, cf. [1, 25]. The path constraints (1d) are enforced in the shooting nodes t_i.

1.4 Sequential Quadratic Programming

From the multiple shooting discretization we obtain the NLP

$$\begin{aligned}
\min_{s,q} \quad & \sum_{i=0}^{N-1} L_i\left(s_i, q_i\right) + E\left(s_N\right) && && \text{(8a)}\\
\text{s.t.} \quad & 0 = s_0 - x_0, && && \text{(8b)}\\
& 0 = s_{i+1} - x_i(t_{i+1}; t_i, s_i, q_i), && 0 \le i < N, && \text{(8c)}\\
& 0 \le h_\mathrm{path}(s_i, \varphi_i(t_i, q_i)), && 0 \le i < N, && \text{(8d)}\\
& 0 \le h_\mathrm{end}(s_N), && && \text{(8e)}
\end{aligned}$$

where

$$L_i\left(s_i, q_i\right) = \int_{t_i}^{t_{i+1}} L(x(t), \varphi_i(t, q_i))\, \mathrm{d}t. \tag{9}$$

This NLP depends parametrically on x_0 and can be written in the generic form

$$\min_{w}\ \phi(w) \text{ s.t. } c(w) + \Lambda x_0 = 0,\ d(w) \ge 0, \tag{10}$$

where $\Lambda = (-I_{n_x}, 0, 0, \dots)$ and $w = (s_0, q_0, \dots, s_{N-1}, q_{N-1}, s_N)$ is the vector of all unknowns.

We choose to solve this NLP using a Newton–type framework. The various structural features such as the separable Lagrangian, the block diagonal Hessian, and the block structure of the Jacobians of the matching constraints (7) can be extensively exploited by tailored linear algebra. In particular using block–wise high–rank updates of the Hessian and a structure–exploiting algorithm for the solution of the arising QP subproblems as presented in section 4 improves convergence speed and computational efficiency.

Starting with an initial guess (w^0, λ^0, μ^0), a full step sequential quadratic programming (SQP) iteration is performed as follows

$$w^{k+1} = w^k + \Delta w^k, \quad \lambda^{k+1} = \lambda^k_{\mathrm{QP}}, \quad \mu^{k+1} = \mu^k_{\mathrm{QP}} \tag{11}$$

where $(\Delta w^k, \lambda^k_{\mathrm{QP}}, \mu^k_{\mathrm{QP}})$ is the solution of the QP subproblem

$$\min_{\Delta w} \quad \tfrac{1}{2}\Delta w^T B^k \Delta w + b^{kT} \Delta w \tag{12a}$$

$$\text{s.t.} \quad 0 = C^k \Delta w + c(w^k) + \Lambda x_0, \tag{12b}$$

$$0 \leq D^k \Delta w + d(w^k). \tag{12c}$$

Here, B^k denotes an approximation of the Hessian of the Lagrangian of (8), and b^k, C^k and D^k are the objective gradient and the Jacobians of the constraints c and d.

2 SQP based Model–Predictive Control

2.1 Initial Value Embedding and Tangential Predictors

The key to a performant numerical algorithm for NMPC is to reuse information from the last QP subproblem to initialize the new subproblem. This is due to the fact that subsequent problems differ only in the parameter x_0 of the linear embedding Λ. Given that the sampling periods are not too long and that the process does not behave too different from the prediction by the model, the solution information of the last problem can be expected to be a very good initial guess close to the solution of the new subproblem.

In [9] and related works it has been proposed to initialize the current problem with the full solution of the previous optimization run, i.e., control *and* state variables. Doing so, the value of s_0 will in general *not* be the value of the current state. By explicitly including the initial value constraint (8b) in the QP formulation, we can guarantee that the constraint is satisfied after the first full Newton–type step due to its linearity in x_0. This is called the *initial value embedding* technique.

On the other hand, by using the full solution of the last problem as initialization of the new problem, the first full Newton–type step already gives us a first order approximation of the solution of the new problem, even in the presence of an active set change. This motivates the idea of *real–time iterations*, which perform only one Newton–type iteration per sample, and is at the same time the main reason for our preference of active set methods over interior–point techniques. We refer to [10] for a detailed survey on the topic of initial value embeddings and the resulting first order tangential predictors.

2.2 Real–Time Iterations

Using the initial value embedding also has an important algorithmical advantage. We can evaluate all derivatives and all function values except the initial value constraint prior to knowing the current state x_0. Consequently, we can also presolve a major part of QP (12). This allows to separate each real–time iteration into the following three phases.

Preparation

All functions and derivatives that do not require knowledge of x_0 are evaluated using the iterate of the previous step (w^k, λ^k, μ^k). Due to its special structure, the variables $(\Delta s_1, \dots, \Delta s_N)$ can be eliminated from QP (12), cf. section 4.

Feedback

As soon as x_0 is available, Δs_0 can be eliminated as well and a small QP only in the variables $(\Delta q_0, \dots, \Delta q_{N-1})$ is solved. The variable $q_0^{k+1} = q_0^k + \Delta q_0^k$ is then given to the process, allowing to compute the feedback control $\varphi_0(t, q_0^{k+1})$. Thus, the actual feedback delay reduces to the solution time of the QP resulting from both eliminations. The affine-linear dependence of this QP on x_0 via Λ can further be exploited by parametric quadratic programming as described in section 2.3.

Transition

Finally, the eliminated variables are recovered and step (11) is performed to obtain the new set of NLP variables $(w^{k+1}, \lambda^{k+1}, \mu^{k+1})$.

2.3 Parametric Quadratic Programming

Both the structured NLP (8) and the QP subproblems (12) derived from it depend parametrically on x_0. This linear dependence on x_0 is favourably exploited by parametric active set methods for the solution of (12), cf. [4] and [12]. The idea here is to introduce a linear affine homotopy in a scalar parameter $\tau \in [0, 1] \subset \mathbb{R}$ from the QP that was solved in iteration $k - 1$ to the QP to be solved in iteration k:

$$\begin{aligned} \min_{\Delta w} \quad & \tfrac{1}{2}\Delta w^T B^k \Delta w + b^T(\tau)\Delta w && \text{(13a)} \\ \text{s.t.} \quad & 0 = C^k \Delta w + c(\tau) + \Lambda x_0(\tau), && \text{(13b)} \\ & 0 \le D^k \Delta w + d(\tau), && \text{(13c)} \end{aligned}$$

with initial values $x_0(0) = x_0^{k-1}$, $x_0(1) = x_0^k$. Linear affine gradient and constraint right hand sides on the homotopy path,

$$b(\tau) = (1-\tau)b(w^{k-1}) + \tau b(w^k), \tag{14a}$$

$$c(\tau) = (1-\tau)c(w^{k-1}) + \tau c(w^k), \tag{14b}$$

$$d(\tau) = (1-\tau)d(w^{k-1}) + \tau d(w^k), \tag{14c}$$

allow for an update of the QP's vectors in iteration k by one of the multi–level scheme's modes, cf. section 3. From the optimality conditions of QP (13) in $\tau = 0$ and $\tau = 1$ it is easily found that an update of the QP's matrices is possible as well, without having to introduce matrix–valued homotopies.

Using this approach to compute the SQP algorithm's steps has multiple advantages. First, a phase I for finding a feasible point of the QP is unnecessary, as we can start the homotopy in a trivial QP with zero vectors and known optimal solution. Second, we can monitor the process of solving the QP using the distance $1-\tau$ to the homotopy path's end. Intermediate iterates are physically meaningful and optimal for a known QP on the homotopy path. Thus, intermediate control feedback can be given during the ongoing solution process. Finally, premature termination of the QP solution process due to computing time constraints becomes possible, cf. [12].

3 The Multi–Level Iteration Scheme

A novel and promising algorithmic approach to SQP based nonlinear model predictive control is the multi–level iteration method, first proposed in [6, 5].

The multi–level iteration method aims at providing feedback very fast, while updating the data of the feedback-generating QP with information from the process on different levels. We distinguish four levels or modes, from which multi–level iteration schemes can be combined.

3.1 Mode A: Feedback Iterations

For Mode A, we assume that QP (12) is given with a Hessian approximation $\overline{B}$, objective gradient $\overline{b}$, constraint values $\overline{c}$, $\overline{d}$, and Jacobians $\overline{C}$, $\overline{D}$, and working on a reference solution $(\overline{w}, \overline{\lambda}, \overline{\mu})$. The aim of Mode A is to compute feedback by resolving the QP for new given current states x_0 and returning the control parameters $\overline{q}_0 + \Delta q_0^k$ to the process as quickly as possible. Mode A is essentially a linear model predictive controller (LMPC). In contrast to LMPC which uses linearizations of a steady state model, Mode A works on linearizations provided by higher modes of the multi–level scheme, which may include transient phases of the nonlinear process.

3.2 Mode B: Feasibility Improvement Iterations

In Mode B, we assume that we have a Hessian approximation $\overline{B}$, a reference objective gradient $\overline{b}$, Jacobians $\overline{C}$, $\overline{D}$ and a reference solution $(\overline{w}, \overline{\lambda}, \overline{\mu})$. Furthermore, Mode B holds its own variables w_{B}^k, which are initially set to $\overline{w}$.

To finish the setup of QP (12), we evaluate new function values $c(w^k)$ and $d(w^k)$ and approximate the QP gradient by $b(w^k) = \overline{b} + \overline{B}\left(w^k_{\mathrm{B}} - \overline{w}\right)$, so that we come up with the following QP

$$\min_{\varDelta w^k_{\mathrm{B}}} \quad \frac{1}{2}\,\varDelta w^{k\;T}_{\mathrm{B}}\;\overline{B}\;\varDelta w^k_{\mathrm{B}} + b(w^k)^T \varDelta w^k_{\mathrm{B}} \tag{15a}$$

$$\mathrm{s.\,t.} \quad \overline{C}\,\varDelta w^k_{\mathrm{B}} + c(w^k) + \varLambda x_0 = 0 \tag{15b}$$

$$\overline{D}\,\varDelta w^k_{\mathrm{B}} + d(w^k) \geq 0. \tag{15c}$$

Once we have solved the QP, we return the control parameters $q^k_{\mathrm{B},0} + \varDelta q^k_{\mathrm{B},0}$ to the process and iterate by setting $w^{k+1}_{\mathrm{B}} = w^k_{\mathrm{B}} + \varDelta w^k_{\mathrm{B}}$.

When performing Mode B iterations with a fixed x_0, one can show that w^k_{B} converges locally to a suboptimal but feasible point of NLP (8), thus Mode B iterations are also referred to as *feasibility improvement iterations*. Optimality is approximately treated by the gradient updates. In comparison to Mode A, the additional computational cost for a Mode B iteration are evaluations of the constraints c and d, and condensing of the constraint vectors and the approximated gradient. Since the QP matrices are fixed, no new matrix decompositions are required during QP solving.

3.3 Mode C: Optimality Improvement by Adjoint SQP Iterations

In Mode C, we assume that we have a Hessian approximation $\overline{B}$, Jacobians $\overline{C}$, $\overline{D}$ and a reference solution $(\overline{w}, \overline{\lambda}, \overline{\mu})$. Furthermore, Mode C holds its own variables $(w^k_{\mathrm{C}}, \lambda^k_{\mathrm{C}}, \mu^k_{\mathrm{C}})$, which are initially set to $(\overline{w}, \overline{\lambda}, \overline{\mu})$. To finish the setup of QP (12), we have to evaluate new function values $c(w^k)$ and $d(w^k)$, and we compute a modified gradient by

$$b(w^k) = \nabla\phi(w^k) + \left(\overline{C}^T - C^{k\;T}\right)\lambda^k + \left(\overline{D}^T - D^{k\;T}\right)\mu^k, \tag{16}$$

where C^k and D^k are the Jacobians of the constraints c and d at w^k. However, the Jacobians need not to be calculated completely, but rather the adjoint derivatives $C^{k\;T}\lambda^k$ and $D^{k\;T}\mu^k$. This can be done efficiently by the reverse mode of automatic differentiation, cf. [17]. After solving the following QP

$$\min_{\varDelta w^k_{\mathrm{C}}} \quad \frac{1}{2}\,\varDelta w^{k\;T}_{\mathrm{C}}\;\overline{B}\;\varDelta w^k_{\mathrm{C}} + b(w^k)^T \varDelta w^k_{\mathrm{C}} \tag{17a}$$

$$\mathrm{s.\,t.} \quad \overline{C}\,\varDelta w^k_{\mathrm{C}} + c(w^k) + \varLambda x_0 = 0 \tag{17b}$$

$$\overline{D}\,\varDelta w^k_{\mathrm{C}} + d(w^k) \geq 0, \tag{17c}$$

we return the control parameters $q^k_{\mathrm{C},0} + \varDelta q^k_{\mathrm{C},0}$ to the process and iterate by setting

$$w^{k+1}_{\mathrm{C}} = w^k_{\mathrm{C}} + \varDelta w^k_{\mathrm{C}}, \quad \lambda^{k+1}_{\mathrm{C}} = \lambda^k_{\mathrm{QP}}, \quad \mu^{k+1}_{\mathrm{C}} = \mu^k_{\mathrm{QP}}, \tag{18}$$

where λ^k_{QP} and μ^k_{QP} are the multipliers obtained from the QP solution.

When performing Mode C iterations with a fixed x_0, one can show local convergence of the sequence $(w^k_{\mathrm{C}}, \lambda^k_{\mathrm{C}}, \mu^k_{\mathrm{C}})$ to a KKT–point of NLP (8), cf. [31], thus Mode C iterations are also referred to as *optimality improvement* iterations. In comparison to Mode B, the additional computational cost for a Mode C iteration are evaluations of the adjoint derivatives $C^{k\ T}\lambda^k$ and $D^{k\ T}\mu^k$ which can be obtained at no more than five times the cost of the respective constraint evaluation [17]. Again, no new matrix decompositions are required during QP solving.

3.4 Mode D: Forward SQP Iterations

Mode D iterations are essentially standard real–time iterations, i.e. full SQP iterations. Mode D holds its own variables $(w^k_{\mathrm{D}}, \lambda^k_{\mathrm{D}}, \mu^k_{\mathrm{D}})$ and in each Mode D iteration, we evaluate the constraints $c(w^k)$ and $d(w^k)$, the objective gradient $b(w^k)$, and the constraint Jacobians $C(w^k)$ and $D(w^k)$, and build a new Hessian approximation $B(w^k)$. After solving QP (12) the control parameters $q^k_{\mathrm{D},0} + \Delta q^k_{\mathrm{D},0}$ are given to the process and we iterate by setting

$$w^{k+1}_{\mathrm{D}} = w^k_{\mathrm{D}} + \Delta w^k_{\mathrm{D}}, \quad \lambda^{k+1}_{\mathrm{D}} = \lambda^k_{\mathrm{QP}}, \quad \mu^{k+1}_{\mathrm{D}} = \mu^k_{\mathrm{QP}}, \tag{19}$$

where λ^k_{QP} and μ^k_{QP} are the multipliers obtained from the QP solution. In each Mode D iteration we have to evaluate the full constraint Jacobians, which amounts to the computational cost of the number of degrees of freedom times the cost for a constraint evaluation. Furthermore, a full initial decomposition has to be performed for the solution of the QP, cf. section 4, which depending on the chosen block structured QP method may have a computational complexity of up to $O(N^2n^3)$.

3.5 Assembling Multi-level Iteration Schemes

From the four modes described above, we can assemble multi-level iteration schemes in various ways. A sequential approach is outlined in the following:

```
Choose initial $\overline{B}$,$\overline{C}$, $\overline{D}$,$\overline{b}$,$\overline{c}$, $\overline{d}$ and $(\overline{w}, \overline{\lambda}, \overline{\mu})$ for all modes
while Process running do
  Determine mode
  case mode A: Perform calculations described in subsection 3.1
  case mode B: Perform calculations described in subsection 3.2
    Update $b$, $c$, $d$ in mode A with the new values from mode B
    Update $\overline{w}$ in mode A with $w_{\mathrm{B}}$
  case mode C: Perform calculations described in subsection 3.3
    Update $b$, $c$, $d$ in mode A and B with the new values from mode C
    Update $\overline{w}$ in mode A and $w_{\mathrm{B}}$ with $w_{\mathrm{C}}$
  case mode D: Perform calculations described in subsection 3.4
    Update $B$, $C$, $D$, $b$, $c$, $d$ in mode A, B and C with the new mode D values
    Update $\overline{w}$ in mode A and $w_{\mathrm{B}}$, $w_{\mathrm{C}}$ with $w_{\mathrm{D}}$ and $(\lambda_{\mathrm{c}}, \mu_{\mathrm{c}})$ with $(\lambda_{\mathrm{D}}, \mu_{\mathrm{D}})$
end while
```

However, a parallel implementation would be an even more natural choice, starting all modes at one time and then performing the updates described above whenever one of the modes has finished one calculation cycle. Ofcouse, one has to consider the issue of synchronization, so that the faster modes are updated only after finishing their current feedback calculation.

Multi-level iteration schemes do not need to employ all modes described above. An example application of a sequential multi-level iteration scheme using modes A and D to a vehicle model is presented in [1].

3.6 Euler Steps

In some cases the limiting factor for feedback generation is the sampling rate of the system states x_0, e.g., if the current states are obtained from a measurement procedure with limited throughput.

If it is still desired to update the feedback control with a higher frequency, a possible remedy is to use the model to predict the next x_0 by an Euler step

$$x_0^{\mathrm{new}} = x_0 + hf(x_0, \varphi_0(t_0, q_0^k)) \tag{20}$$

with a small stepsize $h = t_0^{\mathrm{new}} - t_0$ and use x_0^{new} to obtain a new feedback q_0^{k+1}. In addition, as the explicit Euler scheme generates a linear affine homotopy path for $x_0^{\mathrm{new}}(t)$ starting in t_0, it can be readily combined with the parametric QP strategy of section 2.3. This allows system state predictions to enter the QP solution even before the solution process has been completed.

3.7 Computing the Local Feedback Law

Phase A iterations can even be used to generate a local feedback law which maps differences $\Delta x_0 = x_0^{\mathrm{new}} - x_0$ to feedback updates and thus can be used as an explicit continuous feedback law betweeen two following QP solutions.

To see this, we consider the Karush-Kuhn-Tucker (KKT) system of the QP after a successful solution

$$\underbrace{\begin{pmatrix} B & -C^T & -D_{\mathbb{A}}^T \\ C & & \\ D_{\mathbb{A}} & & \end{pmatrix}}_{:=K} \begin{pmatrix} \Delta w \\ \Delta\lambda \\ \Delta\mu_{\mathbb{A}} \end{pmatrix} = - \begin{pmatrix} b \\ c + \Lambda x_0 \\ d_{\mathbb{A}} \end{pmatrix}, \tag{21}$$

where $\mathbb{A}$ is the optimal active set. Let $\mathbb{I}$ be the index set of Δq_0 within Δw. We can easily calculate the part of the inverse of K which gives us Δq_0 when applied to the right hand side by solving

$$K^T X_i = e_i, \quad i \in \mathbb{I}, \tag{22}$$

with e_i the i-th unity vector. Since a decomposition of K is available from the QP solver, this amounts to only n_{u} backsolves. Assuming that $\mathbb{A}$ keeps constant for small changes in x_0, we can determine an update for Δq_0 by building

$$X^T \begin{pmatrix} 0 \\ \Lambda\Delta x_0 \\ 0 \end{pmatrix}, \tag{23}$$

for which we actually need only a small part of the matrix X.

4 Structured Quadratic Programming

This final part of our survey is concerned with numerical methods for the efficient solution of the QPs that arise from a direct multiple shooting discretization of the model predictive control problem. The focus is put on methods that efficiently exploit the block structure of problem (24) by appropriate linear algebra. We present the condensing algorithm due to [26, 7] that works as a preprocessing step, mention Riccati recursion to exploit the block structure, and conclude with a block structured active set method.

4.1 The Block Structured Quadratic Subproblem

To gain insight into the direct multiple shooting structure of QP (12) we rewrite it to expose the individual block matrices. The matching conditions (7) are separated in (24b), and equality as well as inequality point constraints are collected in (24c):

$$\min_{\Delta w} \sum_{i=0}^{N} \left(\tfrac{1}{2} \Delta w_i^T B_i \Delta w_i + \Phi_i^T \Delta w \right) \tag{24a}$$

$$\text{s.t.} \quad 0 = X_i \Delta w_i - \Delta w_{i+1} - h_i \qquad 0 \le i < N \tag{24b}$$

$$0 \leqq R_i \Delta w_i + r_i \qquad 0 \le i \le N \tag{24c}$$

4.2 Condensing and Dense Active Set Methods

The purpose of the following condensing algorithm that is due to [26] and [7], cf. also [23], is to exploit the block sparse structure of QP (24) in a preprocessing or *condensing* step that transforms the QP into a smaller and densely populated one.

Reordering the Sparse Quadratic Problem

We start by reordering the constraint matrix of QP (24) to separate the multiple shooting state values $\Delta v_1 = (\Delta s_1, \dots, \Delta s_N)$ introduced in section 1.3 from the single shooting values $\Delta v_2 = (\Delta s_0, \Delta q_0, \dots, \Delta q_{N-1})$ as shown in (25). Therein, we use partitions $X_i = (X_i^{\mathrm{s}}\; X_i^{\mathrm{q}})$ and $R_i = (R_i^{\mathrm{s}}\; R_i^{\mathrm{q}})$ of the Jacobians X_i and R_i with respect to Δs and Δq.

$$\left(\begin{array}{ccccc|cccc}
X_0^{\mathrm{s}} & X_0^{\mathrm{q}} & & & & -I & & & \\
 & & X_1^{\mathrm{q}} & & & X_1^{\mathrm{s}} & -I & & \\
 & & & \ddots & & & \ddots & \ddots & \\
 & & & & X_{N-1}^{\mathrm{q}} & & & X_{N-1}^{\mathrm{s}} & -I \\
\hline
R_0^{\mathrm{s}} & R_0^{\mathrm{q}} & & & & & & & \\
 & & R_1^{\mathrm{q}} & & & R_1^{\mathrm{s}} & & & \\
 & & & \ddots & & & \ddots & & \\
 & & & & R_{N-1}^{\mathrm{q}} & & & R_{N-1}^{\mathrm{s}} & \\
 & & & & & & & & R_N^{\mathrm{s}}
\end{array}\right). \tag{25}$$

Elimination Using the Matching Conditions

We may now use the negative identity matrix blocks of the equality matching conditions as pivots to formally eliminate the state values $(\Delta s_0, \dots, \Delta s_N)$ from system (25), analogous to the usual Gaussian elimination method for triangular matrices. From this elimination procedure the dense constraint matrix

$$\begin{pmatrix} \overline{X} & -I \\ \overline{R} & 0 \end{pmatrix} := \left(\begin{array}{ccccc|cccc}
X_0^{\mathrm{s}} & X_0^{\mathrm{q}} & & & & -I & & & \\
X_1^{\mathrm{s}} X_0^{\mathrm{s}} & X_1^{\mathrm{s}} X_0^{\mathrm{q}} & X_1^{\mathrm{q}} & & & & -I & & \\
\vdots & \vdots & \vdots & \ddots & & & & \ddots & \\
\Pi_0^{N-1} & \Pi_1^{N-1} X_0^{\mathrm{q}} & \Pi_2^{N-1} X_1^{\mathrm{q}} & \cdots & X_{N-1}^{\mathrm{q}} & & & & -I \\
\hline
R_0^{\mathrm{s}} & R_0^{\mathrm{q}} & & & & & & & \\
R_1^{\mathrm{s}} X_0^{\mathrm{s}} & R_1^{\mathrm{s}} X_0^{\mathrm{q}} & R_1^{\mathrm{q}} & & & & & & \\
\vdots & \vdots & \vdots & \ddots & & & & & \\
R_N^{\mathrm{s}} \Pi_0^{N-1} & R_N^{\mathrm{s}} \Pi_1^{N-1} X_0^{\mathrm{q}} & R_N^{\mathrm{s}} \Pi_2^{N-1} X_1^{\mathrm{q}} & \cdots & R_N^{\mathrm{s}} X_{N-1}^{\mathrm{q}} & & & &
\end{array}\right)$$

is obtained, with sensitivity matrix products Π_j^k defined to be

$$\Pi_j^k := \prod_{l=j}^{k} X_l^{\mathrm{s}}, \quad 0 \le j \le k \le N-1. \tag{26}$$

From (4.2) we deduce that, after this elimination step, the transformed QP in terms of the two unknowns Δv_1 and Δv_2 reads

$$\min_{\Delta v} \quad \frac{1}{2}\begin{pmatrix}\Delta v_1\\ \Delta v_2\end{pmatrix}^T \begin{pmatrix}\overline{B}_{11} & \overline{B}_{12}\\ \overline{B}_{12}^T & \overline{B}_{22}\end{pmatrix}\begin{pmatrix}\Delta v_1\\ \Delta v_2\end{pmatrix} + \begin{pmatrix}\overline{\Phi}_1\\ \overline{\Phi}_2\end{pmatrix}^T\begin{pmatrix}\Delta v_1\\ \Delta v_2\end{pmatrix} \tag{27a}$$

$$\text{s.t.} \quad 0 = \overline{X}\Delta v_1 - \Delta v_2 - \overline{h} \tag{27b}$$

$$0 \leqq \overline{R}\Delta v_1 - \overline{r} \tag{27c}$$

wherein $\overline{B}$ and $\overline{\Phi}$ are reorderings of B and Φ, and $\overline{h}$ and $\overline{r}$ are appropriate right hand side vectors obtained by applying the Gaussian elimination steps to h and r.

Reduction to a Single Shooting Sized System

System (27) lends itself to the elimination of the unknown Δv_2. By this step we arrive at the final *condensed QP*

$$\min_{\Delta v_1} \quad \tfrac{1}{2}\Delta v_1^T \overline{\overline{B}} \Delta v_1 + \overline{\overline{\Phi}}^T \Delta v_1 \tag{28a}$$

$$\text{s.t.} \quad 0 \leq \overline{R}\Delta v_1 - \overline{r} \tag{28b}$$

with the following dense Hessian matrix and gradient obtained from substitution of Δv_2 in the objective (27a)

$$\overline{\overline{B}} = \overline{B}_{11} + \overline{B}_{12}\overline{X} + \overline{X}^T\overline{B}_{12}^T + \overline{X}^T\overline{B}_{22}\overline{X}, \tag{29a}$$

$$\overline{\overline{\Phi}} = \overline{\Phi}_1 + \overline{X}^T\overline{\Phi}_2 - \overline{B}_{12}^T\overline{h} - \overline{X}^T\overline{B}_{22}\overline{h}. \tag{29b}$$

The required matrix multiplications are easily laid out to exploit the block triangular structure of $\overline{X}$ and the block diagonal structure of B. In addition, from the elimination steps described in the previous two paragraphs one obtains relations that allow to recover $\Delta v_2 = (\Delta s_1, \ldots, \Delta s_N)$ from the solution $\Delta v_1 = (\Delta s_0, \Delta q_0, \ldots, \Delta q_{N-1})$ of the condensed QP (28).

Solving the Condensed Quadratic Problem

The resulting condensed QP (28) no longer has a multiple shooting specific structure. It may thus be solved using any standard dense active–set method, which is what condensing ultimately aims for. Popular codes are the null space method `QPSOL` and its successor `QPOPT` [15]. The code `BQPD` [13] is even able to exploit remaining sparsity to some extent. An efficient code for parametric quadratic programming is `qpOASES` [12].

Condensing in Model–Predictive Control

The run time complexity of the condensing preprocessing step is $O(N^2)$ due to the elimination in (4.2). As all controls remain in the condensed QP, from which all states additionally introduced in section 1.3 are eliminated, condensing is a computationally favourable approach for model predictive control problems with a large number n_{x} of system states, few control parameters, and a limited number N of discretization points of the prediction horizon. The majority of condensing can be carried out in the preparation phase, cf. section 2.2, as the initial value x_0^{new} need not be known in advance. This reduces the control feedback delay to essentially the run time of the QP solver on the condensed QP (28).

4.3 Riccati Recursion

While the condensing algorithm acts as a preprocessing step on the block structured QP data, an alternative approach is to exploit this structure inside the QP solver, i.e. to solve block structured KKT systems. Riccati recursion, based on the dynamic programming principle, is a popular concept here. Starting with the last shooting node's cost function

$$\phi_N(\Delta s_N) = \tfrac{1}{2}\Delta s_N^T B_N \Delta s_N + \Phi_N^T \Delta s_N \tag{30}$$

the cost–to–go function ϕ_{N-1} of the previous node is found from tabulation of the optimal control step Δq_{N-1} for each admissible state step Δs_{N-1}. This procedure is repeated until the backwards recursion arrives at the first node $i = 0$, at which point the sequence of optimal control steps Δq can simply be obtained from a table look–up using the estimated or measured initial value x_0^{new}.

The optimal control steps Δq_i of nodes $i = N-1, \dots, 0$ are found by solving the purely equality-constrained QP (31) using an appropriate factorization of the associated KKT system,

$$\phi_i(\Delta s_i) = \min_{\substack{\Delta s_{i+1}\\ \Delta q_i}} \frac{1}{2}\begin{pmatrix}\Delta s_i\\ \Delta q_i\end{pmatrix}^T \begin{pmatrix}B_i^{\mathrm{ss}} & B_i^{\mathrm{sq}}\\ B_i^{\mathrm{qs}} & B_i^{\mathrm{qq}}\end{pmatrix}\begin{pmatrix}\Delta s_i\\ \Delta q_i\end{pmatrix} + \begin{pmatrix}\Phi_i^{\mathrm{s}}\\ \Phi_i^{\mathrm{q}}\end{pmatrix}^T\begin{pmatrix}\Delta s_i\\ \Delta q_i\end{pmatrix} + \phi_{i+1}(\Delta s_{i+1}) \tag{31a}$$

$$\text{s.t.} \quad h_i = \begin{pmatrix}X_i^{\mathrm{s}} & X_i^{\mathrm{q}}\end{pmatrix}\begin{pmatrix}\Delta s_i\\ \Delta q_i\end{pmatrix} - \Delta s_{i+1}. \tag{31b}$$

Riccati Recursion in Model–Predictive Control

By observing that the optimal cost–to–go functions $\phi_i(\Delta s_i)$ remain quadratic functions,

$$\phi_i(\Delta s_i) = \Delta s_i^T P_i \Delta s_i + p_i^T \Delta s_i + \pi_i, \tag{32}$$

and by inserting (31b) into (31a), the unknown $\varDelta s_{i+1}$ can be eliminated. Problem (31) then becomes an unconstrained minimization problem,

$$\phi_i(\varDelta s_i) = \min_{\varDelta q_i} \frac{1}{2} \begin{pmatrix} \varDelta s_i \\ \varDelta q_i \end{pmatrix}^T \begin{pmatrix} B_i^{\mathrm{ss}} + {X_i^{\mathrm{s}}}^T P_{i+1} X_i^{\mathrm{s}} & B_i^{\mathrm{sq}} + {X_i^{\mathrm{s}}}^T P_{i+1} X_i^{\mathrm{q}} \\ B_i^{\mathrm{qs}} + {X_i^{\mathrm{q}}}^T P_{i+1} X_i^{\mathrm{s}} & B_i^{\mathrm{qq}} + {X_i^{\mathrm{q}}}^T P_{i+1} X_i^{\mathrm{q}} \end{pmatrix} \begin{pmatrix} \varDelta s_i \\ \varDelta q_i \end{pmatrix} \tag{33a}$$

$$+ \begin{pmatrix} \varPhi_i^{\mathrm{s}} - {X_i^{\mathrm{s}}}^T P_{i+1} h_i + {X_i^{\mathrm{s}}}^T p_{i+1} \\ \varPhi_i^{\mathrm{q}} - {X_i^{\mathrm{q}}}^T P_{i+1} h_i + {X_i^{\mathrm{q}}}^T p_{i+1} \end{pmatrix}^T \begin{pmatrix} \varDelta s_i \\ \varDelta q_i \end{pmatrix} \tag{33b}$$

$$+ h_i^T P_{i+1} h_i - p_{i+1}^T h_i + \pi_{i+1} \tag{33c}$$

From this, an explicit expression for the optimal $\varDelta q_i$ is easily obtained. Inserting it into the cost–to–go function finally allows for the direct computation of $\phi_i(\varDelta s_i)$. The backwards recursion can thus be started with $P_N = B_N$, $p_N = \varPhi_n$, $\pi_n = 0$, and carried out without knowledge of the true system state x_0^{new}. After the backward sweep has been completed, the feedback control step is available as

$$\varDelta q_0 = K_0(x_0 - x_0^{\mathrm{new}}) + k_0 \tag{34}$$

with an $n_{\mathrm{q}} \times n_{\mathrm{x}}$ matrix K_0 and an n_{q}-vector k_0 obtained from the backward sweep eliminations. The feedback delay is as small as the time required for a matrix–vector–multiplication with K_0. A forward recursion starting with the known initial value $\varDelta s_0 = x_0^{\mathrm{new}} - s_0$ is employed afterwards to recover the steps $\varDelta q_1, \ldots, \varDelta q_{N-1}$ and $\varDelta s$ which are not needed for the immediate control feedback.

Inequality Constraints

The applicability of Riccati recursion is restricted to purely equality constrained systems, i.e. KKT systems or QPs with only equality constraints. In order to treat inequality constraints, a Riccati recursion based KKT solver can be employed inside an active set method. The performance of such Riccati recursion based active set solvers suffers from the $O(Nn^3)$ runtime complexity of the KKT system solution, and the approach is thus more popular for interior point methods [18, 28, 30], where it has been successfully used in place of symmetric indefinite factorizations.

4.4 Block Structured Active Set Methods

A third possibility of solving QP (24) is to employ a block structured factorization of the KKT system inside an active set method. For efficiency, matrix updates for such factorizations should be available.

Block Structured Factorization

We present here a block structured factorization due to [19, 30] that is composed from step-wise reductions of the KKT matrix (35), in which all matrices and vectors are understood as restrictions onto the current active set,

$$\begin{pmatrix}
B_0 & R_0^T & X_0^T & & & & & & \\
R_0 & & & & & & & & \\
X_0 & & & P_1 & & & & & \\
 & & P_1^T & B_1 & R_1^T & X_1^T & & & \\
 & & & R_1 & & & & & \\
 & & & X_1 & & & & & \\
 & & & & & & \ddots & & \\
 & & & & & & & B_N & R_N^T \\
 & & & & & & & R_N &
\end{pmatrix} \tag{35}$$

Similar to Ricatti recursion, the idea is to factorize this KKT matrix and exploit the inherent block structure while avoiding any fill–in.

Matrix Updates

Contrary to Ricatti recursion, though, we desire to derive a factorization that opens up the possibility of applying *matrix update* techniques, cf. [14, 16]. These allow to recover the factorization of the KKT matrix after an active set exchange in $O(n^2)$ time, while computing a factorization anew usually requires $O(n^3)$ time. An example is the Schur complement based dual active set method presented in [29, 3]. A factorization tailored to the block structure (35) that is based on a hybrid null–space range–space is given in [19]. Suitable updates are derived in [20], based on techniques by [14].

Runtime Complexity

This approach has $O(N)$ runtime complexity compared to $O(N^2)$ for the condensing approach of section 4.2, and is therefore suited problems that require longer prediction horizons or finer discretizations of the prediction horizon. As the factorization eliminates all controls from the system in the first step, problems with limited state dimension but with a large number of controls, e.g. in mixed–integer predictive control [21] or in online optimal experimental design, will benefit from this approach. Compared to the $O(n^3)$ runtime complexity of Riccati recursion techniques, the availability of matrix updates reduces the runtime complexity for all but the first iteration of the active set loop to $O(n^2)$.

5 Summary

We reviewed a collection of state–of–the–art numerical methods for efficient NMPC of nonlinear dynamic processes in ODE and DAE systems under real–time conditions. Our presentation started with a presentation of the discussed problem class and a brief introduction to direct multiple shooting for the discretization of the optimal control problem. We focussed on a Newton–type framework for the solution of the resulting nonlinear problem, relying on active set based methods. In combination with initial value embedding, the real–time iteration scheme provides an efficient first order tangential predictor of the optimal feedback control. A multi–level scheme featuring at least four distinct modes that provide adaptive updates to selected components of the quadratic subproblem is presented, and we mentioned theoretical results as well as computational effort of the different modes. Connections to emerging ideas such as parametric quadratic programming, Euler feedback steps, and the computation of the local linear feedback law providing microsecond control feedback opportunities are shown. The efficient solution of the arising quadratic subproblems is the core of all active–set based NMPC algorithms. Here we introduced the block structure that is due to direct multiple shooting, and reviewed the condensing preprocessing step as well as a Riccati recursion scheme. Both exploit the exhibited block structure, but also left room for improvements. Our survey concluded with mentioning block structured active set methods. These require matrix updates tailored to the block structure, but are able to reduce the run time of an active set iteration to an unmatched complexity of $O(Nn^2)$.

References

1. J. Albersmeyer, D. Beigel, C. Kirches, L. Wirsching, H. Bock, and J. Schlöder. Fast nonlinear model predictive control with an application in automotive engineering. In L. Magni, D. Raimondo, and F. Allgöwer, editors, *Lecture Notes in Control and Information Sciences*, volume 384, pages 471–480. Springer Verlag Berlin Heidelberg, 2009.
2. J. Albersmeyer and H. Bock. Sensitivity Generation in an Adaptive BDF-Method. In H. G. Bock, E. Kostina, X. Phu, and R. Rannacher, editors, *Modeling, Simulation and Optimization of Complex Processes: Proceedings of the International Conference on High Performance Scientific Computing, March 6–10, 2006, Hanoi, Vietnam*, pages 15–24. Springer Verlag Berlin Heidelberg New York, 2008.
3. R. Bartlett and L. Biegler. QPSchur: A dual, active set, schur complement method for large-scale and structured convex quadratic programming algorithm. *Optimization and Engineering*, 7:5–32, 2006.
4. M. Best. *An Algorithm for the Solution of the Parametric Quadratic Programming Problem*, chapter 3, pages 57–76. Applied Mathematics and Parallel Computing. Physica-Verlag, Heidelberg, 1996.

5. H. Bock, M. Diehl, E. Kostina, and J. Schlöder. Constrained Optimal Feedback Control for DAE. In L. Biegler, O. Ghattas, M. Heinkenschloss, D. Keyes, and B. van Bloemen Waanders, editors, *Real-Time PDE-Constrained Optimization*, chapter 1, pages 3–24. SIAM, 2007.
6. H. Bock, M. Diehl, P. Kühl, E. Kostina, J. Schlëder, and L. Wirsching. Numerical methods for efficient and fast nonlinear model predictive control. In R. Findeisen, F. Allgöwer, and L. T. Biegler, editors, *Assessment and future directions of Nonlinear Model Predictive Control*, volume 358 of *Lecture Notes in Control and Information Sciences*, pages 163–179. Springer, 2005.
7. H. Bock and K. Plitt. A Multiple Shooting algorithm for direct solution of optimal control problems. In *Proceedings of the 9th IFAC World Congress*, pages 243–247, Budapest, 1984. Pergamon Press. Available at http://www.iwr.uni-heidelberg.de/groups/agbock/FILES/Bock1984.pdf.
8. A. Bryson and Y.-C. Ho. *Applied Optimal Control.* Wiley, New York, 1975.
9. M. Diehl, H. Bock, J. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer. Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *J. Proc. Contr.*, 12(4):577–585, 2002.
10. M. Diehl, H. Ferreau, and N. Haverbeke. Efficient numerical methods for nonlinear mpc and moving horizon estimation. In L. Magni, D. Raimondo, and F. Allgöwer, editors, *Nonlinear Model Predictive Control*, volume 384 of *Springer Lecture Notes in Control and Information Sciences*, pages 391–417. Springer-Verlag, Berlin, Heidelberg, New York, 2009.
11. M. Diehl, P. Kuehl, H. Bock, and J. Schlöder. Schnelle Algorithmen für die Zustands- und Parameterschätzung auf bewegten Horizonten. *Automatisierungstechnik*, 54(12):602–613, 2006.
12. H. Ferreau, H. Bock, and M. Diehl. An online active set strategy to overcome the limitations of explicit MPC. *International Journal of Robust and Nonlinear Control*, 18(8):816–830, 2008.
13. R. Fletcher. Resolving degeneracy in quadratic programming. Numerical Analysis Report NA/135, University of Dundee, Dundee, Scotland, 1991.
14. P. Gill, G. Golub, W. Murray, and M. A. Saunders. Methods for modifying matrix factorizations. *Mathematics of Computation*, 28(126):505–535, 1974.
15. P. Gill, W. Murray, and M. Saunders. *User's Guide For QPOPT 1.0: A Fortran Package For Quadratic Programming*, 1995.
16. G. Golub and C. van Loan. *Matrix Computations.* Johns Hopkins University Press, Baltimore, 3rd edition, 1996.
17. A. Griewank. *Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation.* Number 19 in Frontiers in Appl. Math. SIAM, Philadelphia, 2000.
18. N. Haverbeke, M. Diehl, and B. de Moor. A structure exploiting interior-point method for moving horizon estimation. In *Proceedings of the 48th IEEE Conference on Decision and Control (CDC09)*, pages 1–6, 2009.
19. C. Kirches, H. Bock, J. Schlöder, and S. Sager. Block structured quadratic programming for the direct multiple shooting method for optimal control. *Optimization Methods and Software*, 2010. DOI 10.1080/10556781003623891.
20. C. Kirches, H. Bock, J. Schlöder, and S. Sager. A factorization with update procedures for a KKT matrix arising in direct optimal control. *Mathematical Programming Computation*, 2010. (submitted). Available Online: http://www.optimization-online.org/DB_HTML/2009/11/2456.html.

21. C. Kirches, S. Sager, H. Bock, and J. Schlöder. Time-optimal control of automobile test drives with gear shifts. *Optimal Control Applications and Methods*, 31(2):137–153, 2010.
22. D. Leineweber. *Efficient reduced SQP methods for the optimization of chemical processes described by large sparse DAE models*, volume 613 of *Fortschritt-Berichte VDI Reihe 3, Verfahrenstechnik*. VDI Verlag, Düsseldorf, 1999.
23. D. Leineweber, I. Bauer, A. Schäfer, H. Bock, and J. Schlöder. An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization (Parts I and II). *Computers and Chemical Engineering*, 27:157–174, 2003.
24. D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: stability and optimality. *Automatica*, 26(6):789–814, 2000.
25. L. Petzold, S. Li, Y. Cao, and R. Serban. Sensitivity analysis of differential-algebraic equations and partial differential equations. *Computers and Chemical Engineering*, 30:1553–1559, 2006.
26. K. Plitt. Ein superlinear konvergentes Mehrzielverfahren zur direkten Berechnung beschränkter optimaler Steuerungen. Diploma thesis, Rheinische Friedrich–Wilhelms–Universität zu Bonn, 1981.
27. C. V. Rao, J. B. Rawlings, and D. Q. Mayne. Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. *IEEE Transactions on Automatic Control*, 48(2):246–258, 2003.
28. C. Rao, S. Wright, and J. Rawlings. Application of interior-point methods to model predictive control. *Journal of Optimization Theory and Applications*, 99:723–757, 1998.
29. C. Schmid and L. Biegler. Quadratic programming methods for tailored reduced Hessian SQP. *Computers & Chemical Engineering*, 18(9):817–832, September 1994.
30. M. Steinbach. Structured interior point SQP methods in optimal control. *Zeitschrift für Angewandte Mathematik und Mechanik*, 76(S3):59–62, 1996.
31. L. Wirsching. An SQP algorithm with inexact derivatives for a direct multiple shooting method for optimal control problems. Diploma thesis, Universität Heidelberg, 2006.

Part VI

PDE-Constrained Optimization

Optimal Control of Periodic Adsorption Processes: The Newton-Picard Inexact SQP Method

A. Potschka[1], A. Küpper[2], J.P. Schlöder[1], H.G. Bock[1], and S. Engell[2]

[1] Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University,
Im Neuenheimer Feld 368, 69123 Heidelberg, Germany,
potschka@iwr.uni-heidelberg.de

[2] Department of Biochemical and Chemical Engineering, Technical University
Dortmund, Emil-Figge-Str. 70, 44227 Dortmund, Germany

Summary. The Newton-Picard method for the computation of time-periodic solutions of Partial Differential Equations (PDE) is an established Newton-type method. We present an improvement of the contraction rate by an overrelaxation for the Picard iteration which comes with no additional cost. Theoretical convergence results are given. Further, we extend the idea of Newton-Picard to the solution of optimization problems with time-periodic Partial Differential Equations. We discuss the resulting inexact Sequential Quadratic Programming (SQP) method and present numerical results for the ModiCon variant of the Simulated Moving Bed process.

1 Introduction

Periodic adsorption processes play an important role in chemical engineering. E.g., in preparative chromatography, product purity and yield can be significantly increased by transition from batch to periodic operation. A production plant will usually be operated in a periodic steady state, also called "cyclic steady state" (CSS). To exploit the high potential of periodic operation, numerical optimization has been proven to be an indispensable tool (see, e.g., [12, 17, 4, 8]). The resulting optimization problems are subject to PDE constraints and additional boundary conditions for periodicity in time. The main focus of this article is on the employment of a fast solver for the determination of periodic solutions [11] simultaneously with an inexact SQP optimization method similar to [5].

2 Application: The Simulated Moving Bed Process

In a chromatographic column, different components that are dissolved in a liquid are separated due to different affinities to the adsorbent. As a result,

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_31, © Springer-Verlag Berlin Heidelberg 2010

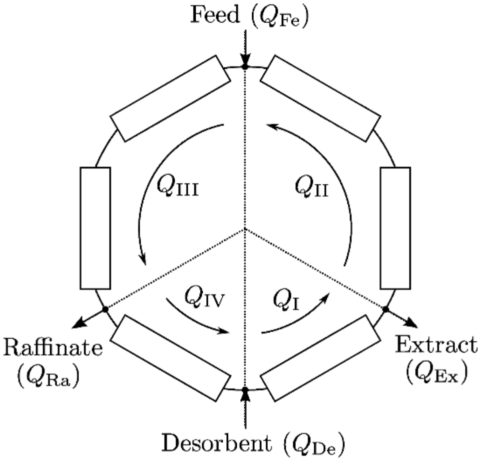


Fig. 1. SMB configuration with 6 columns and 4 zones.

the different components move with different velocities through the column, and hence, can be separated into nearly pure fractions at the outlet. The SMB process consists of several chromatographic columns which are interconnected in series to constitute a closed loop (see Figure 1). An effective counter-current movement of the stationary phase relative to the liquid phase is realized by periodic and simultaneous switching of the inlet and outlet ports by one column in the direction of the liquid flow. Compared to the batch operation of a single chromatographic column, the Simulated Moving Bed (SMB) process offers great improvements of process performance in terms of eluent consumption and utilization of the solid bed. In the basic SMB process, all flow rates are constant and the switching of the columns is simultaneous with a fixed switching period. By introducing more degrees of freedom, the efficiency of the separation can be increased further. E.g., the flow rates can be varied during the switching periods (PowerFeed), the feed concentration can be varied during the switching periods (ModiCon) or asynchronous switching of the ports can be introduced (VariCol) [14, 15].

Accurate dynamic models of such multi-column continuous chromatographic processes consist of the dynamic process models of the single chromatographic columns, the node balances which describe the connection of the columns, and the port switching. The behaviour of radially homogeneous chromatographic columns is described by the General Rate Model [13].

For both species $i = 1, 2$, the General Rate Model considers three phases, namely the instationary phase c_i which moves through the columns between the fixed bed particles, the liquid stationary phase $c_{\mathrm{p},i}$ inside the porous fixed bed particles, and the adsorbed stationary phase $q_{\mathrm{p},i}$ on the inner surface of the particles.

It is assumed that the columns are long and thin enough that radial concentration profiles can be neglected. The fixed bed particles are assumed to be spherical and the concentrations inside the particles are assumed to be rotationally symmetric. The governing equations in non-dimensional form are

$$\partial_t c_i = \mathrm{Pe}_i^{-1} \partial_z^2 c_i - \partial_z c_i - \mathrm{St}_i \left(c_i - c_{\mathrm{p},i}|_{r=1} \right), \quad (t,z) \in (0,T) \times (0,1), \quad \text{(1a)}$$
$$\partial_t \left((1-\epsilon_\mathrm{p}) q_{\mathrm{p},i} + \epsilon_\mathrm{p} c_{\mathrm{p},i} \right) = \eta_i \left(r^{-2} \partial_r \left(r^2 \partial_r c_{\mathrm{p},i} \right) \right), (t,r) \in (0,T) \times (0,1), \quad \text{(1b)}$$

together with the boundary conditions

$$\begin{aligned} &\partial_z c_i(t,0) = \mathrm{Pe}_i \left(c_i(t,0) - c^{\mathrm{in}}(t) \right), && \partial_z c_i(t,1) = 0, \\ &\partial_r c_{\mathrm{p},i}(t,0) = 0, && \partial_r c_{\mathrm{p},i}(t,1) = \mathrm{Bi}_i \left(c_i(t,z) - c_{\mathrm{p},i}(t,1) \right), \end{aligned}$$

with positive constants ϵ_p (porosity), η_i (nondimensional diffusion coefficient), Pe_i (Péclet number), St_i (Stanton number), and Bi_i (Biot number). The stationary phases are coupled by an algebraic condition, e.g. the nonlinear Bi-Langmuir isotherm equation

$$q_{\mathrm{p},i} = \frac{H_i^1 c_{\mathrm{p},i}}{1 + k_1^1 c_{\mathrm{p},1} + k_2^1 c_{\mathrm{p},2}} + \frac{H_i^2 c_{\mathrm{p},i}}{1 + k_1^2 c_{\mathrm{p},1} + k_2^2 c_{\mathrm{p},2}}, \quad \text{(2)}$$

with non-negative constants H_i^j (Henry coefficients) and k_i^j (isotherm parameters).

The model poses a number of difficulties:

1. The isotherm equations are algebraic constraints.
2. The time derivatives $\partial_t q_{\mathrm{p},i}$ and $\partial_t c_{\mathrm{p},i}$ are coupled on the left hand side of equation (1b).
3. For each point $z \in [0,1]$ in the axial direction a stationary phase equation (1b) is supposed to hold.
4. The stationary phase equation has a singularity for $r = 0$.

Points 1 and 2 are addressed by elimination of $q_{\mathrm{p},i}$ via substitution of the algebraic constraints (2) into equation (1b). After differentiation with respect to t, one obtains a system of the form

$$G(c_{\mathrm{p},1}, c_{\mathrm{p},2}) \begin{pmatrix} \partial_t c_{\mathrm{p},1} \\ \partial_t c_{\mathrm{p},2} \end{pmatrix} = \begin{pmatrix} \eta_1 \left(r^{-2} \partial_r \left(r^2 \partial_r c_{\mathrm{p},1} \right) \right) \\ \eta_2 \left(r^{-2} \partial_r \left(r^2 \partial_r c_{\mathrm{p},2} \right) \right) \end{pmatrix},$$

where the coupling 2-by-2 matrix G depends nonlinearly on $c_{\mathrm{p},i}$.

Regarding point 3, one should think of equation (1b) as living on the 2-dimensional (z,r) domain without any derivatives in the axial direction. The coupling occurs through the boundary conditions and equation (1a).

The model for the whole Simulated Moving Bed process consists of a fixed number N_{col} of columns described by the General Rate Model and mass balances at the ports between the columns. The concentrations of column j are denoted by a superscript j. In the ModiCon variant, the process is controlled by the time-independent flow rates Q_{De} (desorbent), Q_{Ex} (extract), Q_{Rec} (recycle) and the time-dependent feed concentration $c_{\mathrm{Fe}}(t)$. The feed flow rate Q_{Fe} is fixed. The remaining flow rates, which are the raffinate flow rate Q_{Ra} and the zone flow rates $Q_I, \ldots, Q_{IV}$, are fully determined by conservation of mass via

$$\begin{aligned} Q_{\mathrm{Ra}} &= Q_{\mathrm{De}} - Q_{\mathrm{Ex}} + Q_{\mathrm{Fe}}, \\ Q_{\mathrm{I}} &= Q_{\mathrm{De}} + Q_{\mathrm{Rec}}, \\ Q_{\mathrm{II}} &= Q_{\mathrm{I}} - Q_{\mathrm{Ex}}, \\ Q_{\mathrm{III}} &= Q_{\mathrm{II}} + Q_{\mathrm{Fe}}, \\ Q_{\mathrm{IV}} &= Q_{\mathrm{III}} - Q_{\mathrm{Ra}} = Q_{\mathrm{Rec}}. \end{aligned}$$

The inflow concentrations of the column after the feed and the desorbent ports can be calculated from the feed concentration $c_{\mathrm{Fe},i}$ and from the outflow concentrations $c_{\cdot,i}^{\mathrm{out}}$ of the previous column according to

$$\begin{aligned} c_{\mathrm{I},i}^{\mathrm{in}} Q_{\mathrm{I}} &= c_{\mathrm{IV},i}^{\mathrm{out}} Q_{\mathrm{IV}}, \\ c_{\mathrm{III},i}^{\mathrm{in}} Q_{\mathrm{III}} &= c_{\mathrm{II},i}^{\mathrm{out}} Q_{\mathrm{II}} + c_{\mathrm{Fe},i} Q_{\mathrm{Fe}}, \end{aligned}$$

for $i = 1, 2$, With the port concentrations and the flow rates the feed, extract, and raffinate masses, and the product purities can be calculated via

$$\begin{aligned} m_{\mathrm{Fe},i}(t) &= \int_0^t c_{\mathrm{Fe},i}(\tau) Q_{\mathrm{Fe}} \mathrm{d}\tau, \\ m_{\mathrm{Ex},i}(t) &= \int_0^t c_{\mathrm{I},i}^{\mathrm{out}}(\tau, 1) Q_{\mathrm{Ex}} \mathrm{d}\tau, \\ m_{\mathrm{Ra},i}(t) &= \int_0^t c_{\mathrm{III},i}^{\mathrm{out}}(\tau, 1) Q_{\mathrm{Ra}} \mathrm{d}\tau, \\ \mathrm{Pur}_{\mathrm{Ex}}(t) &= m_{\mathrm{Ex},1}(t) / (m_{\mathrm{Ex},1}(t) + m_{\mathrm{Ex},2}(t)), \\ \mathrm{Pur}_{\mathrm{Ra}}(t) &= m_{\mathrm{Ra},2}(t) / (m_{\mathrm{Ra},1}(t) + m_{\mathrm{Ra},2}(t)). \end{aligned}$$

3 Formulation of the Optimization Problem

We consider the optimization of an SMB process with variable feed concentration (ModiCon process) with respect to the purity of the two product streams $\mathrm{Pur}_{\min}$ for a constant feed flow Q_{Fe} but varying feed concentration $c_{\mathrm{Fe}}(t)$. Over one period T, the average feed concentration must be equal to the given feed concentration $c_{\mathrm{Fe}}^{\mathrm{SMB}}$ of a reference SMB process.

We discretize the spatial part of the PDE inside the particles with one polynomial (see [7]) and use an appropriate discontinuous Galerkin method for the axial direction of the columns. This leads to a system of ordinary differential equations. Let all the discretized concentrations $c_i^j, c_{\mathrm{p},i}^j$ be combined into a single vector x, all masses $m_{\mathrm{Fe},i}, m_{\mathrm{Ex},i}, m_{\mathrm{Ra},i}$ into m, and the time-independent quantities Q_{De}, Q_{Ex}, Q_{Rec}, T, $\mathrm{Pur}_{\min}$ into v.

The semi-discretized optimal control problem to be solved is then

$$\begin{aligned}
\underset{x,m,c_{\mathrm{Fe}},v}{\text{maximize}} \quad & \mathrm{Pur}_{\min} \\
\text{subject to} \quad & \dot{x}(t) = f(t, x(t), c_{\mathrm{Fe}}(t), v), \quad t \in [0,T], \\
& \dot{m}(t) = f^{\mathrm{m}}(x(t), c_{\mathrm{Fe}}(t), v), \quad t \in [0,T], \\
& x(0) = Px(T), \\
& m(0) = 0, \\
& m_{\mathrm{Fe}}(T) = c_{\mathrm{Fe}}^{\mathrm{SMB}} Q_{\mathrm{Fe}} T, \\
& \mathrm{Pur}_{\mathrm{Ex}}(T) \geq \mathrm{Pur}_{\min}, \\
& \mathrm{Pur}_{\mathrm{Ra}}(T) \geq \mathrm{Pur}_{\min}, \\
& c_{\mathrm{Fe,max}} \geq c_{\mathrm{Fe}}(t) \geq 0, \quad t \in [0,T], \\
& Q_{\mathrm{De,max}} \geq Q_{\mathrm{De}}, \\
& Q_{\max} \geq \begin{pmatrix} 0 & \mathrm{I} & 0 & 0 \\ 0 & 0 & 0 & \mathrm{I} \\ \mathrm{I} & -\mathrm{I} & \mathrm{I} & 0 \\ \mathrm{I} & 0 & 0 & \mathrm{I} \\ \mathrm{I} & -\mathrm{I} & 0 & \mathrm{I} \\ \mathrm{I} & -\mathrm{I} & \mathrm{I} & \mathrm{I} \end{pmatrix} \begin{pmatrix} Q_{\mathrm{De}} \\ Q_{\mathrm{Ex}} \\ Q_{\mathrm{Fe}} \\ Q_{\mathrm{Rec}} \end{pmatrix} \geq Q_{\min},
\end{aligned}$$

The constant permutation matrix P represents the switching of ports.

We discretize the control $c_{\mathrm{Fe}}(t)$ as piecewise constant on a control grid $0 = t^0 < t^1 < \cdots < t^N = T$ with values q^i in $(t^{i-1}, t^i]$. The states x are parametrized by their free initial value, denoted by s^0. A parametrization for the integration states m is not needed. The remaining degrees of freedom are just $w = (s^0, v, q^1, \ldots, q^N)$ with the dimensions $n_s + n_v + Nn_q$.

We denote intermediate state variables on the control discretization grid as s^i and m^i. They can be determined by integration of the differential equation. The optimization problem is (with obvious choices of the functions J, r and the vectors v_l, v_u, q_l, q_u)

$$\begin{aligned}
\underset{w}{\text{minimize}} \quad & J(v) && \text{(3a)} \\
\text{subject to} \quad & s^0 - Ps^N = 0, && \text{(3b)} \\
& r(m^N, v) \begin{Bmatrix} = \\ \geq \end{Bmatrix} 0, && \text{(3c)} \\
& v_u \geq v \geq v_l, && \text{(3d)} \\
& q_u \geq q \geq q_l. && \text{(3e)}
\end{aligned}$$

We denote the sensitivities of s^N and m^N with

$$X_s = \frac{\mathrm{d}s^N}{\mathrm{d}s^0}, \qquad X_v = \frac{\mathrm{d}s^N}{\mathrm{d}v}, \qquad X_q^i = \frac{\mathrm{d}s^N}{\mathrm{d}q^i},$$

$$Y_s = \frac{\mathrm{d}m^N}{\mathrm{d}s^0}, \qquad Y_v = \frac{\mathrm{d}m^N}{\mathrm{d}v}, \qquad Y_q^i = \frac{\mathrm{d}m^N}{\mathrm{d}q^i},$$

for $i = 1, \ldots, N$. Furthermore, we define $R_m = \frac{\partial r}{\partial m}$ and $R_v = \frac{\partial r}{\partial v}$.

4 Calculating Periodic Solutions

4.1 The Newton-Shift-Picard Method

In this section we want to review and extend existing techniques for the determination of the CSS when the controls and switching period are fixed. Thus, we are interested in solving equation (3b) alone. Lust et al. [11] have studied and analyzed Newton-Picard methods in the framework of approximate Newton methods. In a simplified form, they approximate the Newton system for the step Δs^0

$$(\mathrm{I} - M)\,\Delta s^0 = -(s^0 - Ps^N), \quad \text{with } M = PX_s, \tag{4}$$

by the projective approximation

$$\left(\mathrm{I} - MVV^{\mathrm{T}}\right)\Delta s^0 = -(s^0 - Ps^N), \tag{5}$$

where M is only calculated on the subspace spanned by the columns of the orthonormal n_s-by-p matrix V. We extend this approximation to the form

$$\left[1/(1+\sigma)\left(\mathrm{I} - VV^{\mathrm{T}}\right) + (\mathrm{I} - M)\,VV^{\mathrm{T}}\right]\Delta s^0 = -(s^0 - Ps^N),$$

with a scalar relaxation factor $\sigma \neq -1$. We call σ the "shift" factor for reasons which will become clear later in the convergence proof. For $\sigma = 0$, we exactly recover equation (5), and if additionally $p = n_s$, we recover equation (4). The shift leads to an overrelaxation for the Picard iteration of the fast modes which accelerates the overall convergence.

4.2 Asymptotic Convergence Rates of the Newton-Shift-Picard Method

Let $\lambda_i, i = 1, \ldots, n_s$, denote the eigenvalues of M ordered with decreasing modulus. We generally assume equation (4) to have a unique solution, requiring $\lambda_i \neq 1$ for all $i = 1, \ldots, n_s$.

In order to investigate the local linear convergence rate κ of the different approximations, we use the following lemma which is a version of the Local Contraction Theorem of Bock [3] adapted to our purposes.

Lemma 1. *Let $x \in \mathbf{R}^n$, and let $f : \mathbf{R}^n \to \mathbf{R}^n$ be twice continuously differentiable in an ε-neighborhood $U = B_\varepsilon(x)$ and such that $f(x) = 0$. Furthermore, let $\tilde{x} \in U$ be a sufficiently small perturbation of x and let J be an invertible n-by-n matrix. Define $x^+ \in \mathbf{R}^n$ according to*

$$x^+ = \tilde{x} - J^{-1} f(\tilde{x}).$$

Then it holds that

$$\|x^+ - x\| \leq \kappa \|\tilde{x} - x\| + O(\varepsilon^2), \text{ with } \kappa = \rho(J^{-1}(J - \nabla f(x)^{\mathrm{T}})).$$

Conversely, there exists a $\tilde{x} \in U$ such that

$$\|x^+ - x\| \geq \kappa \|\tilde{x} - x\| - O(\varepsilon^2).$$

Proof. By Taylor's Theorem we have

$$x^+ = \tilde{x} - J^{-1} f(\tilde{x}) = \tilde{x} - J^{-1} \nabla f(x)^{\mathrm{T}} (\tilde{x} - x) + O(\varepsilon^2),$$

and thus

$$x^+ - x = J^{-1}(J - \nabla f(x)^{\mathrm{T}})(\tilde{x} - x) + O(\varepsilon^2).$$

The first assertion then holds by the triangle inequality. For the second assertion, construct $\tilde{x} = x + (\varepsilon/2)\delta x$, where

$$\delta x = \underset{\|y\|=1}{\arg\max} \|J^{-1}(J - \nabla f(x)^{\mathrm{T}}) y\|.$$

The lower-bound version of the triangle equality completes the proof. ◇

To appreciate the convergence rate of the different approximations, we also need an explicit form of the inverse of the approximated Jacobian.

Lemma 2. *Let $S = 1/(1+\sigma)\left(\mathrm{I} - VV^{\mathrm{T}}\right) + (\mathrm{I} - M)\, VV^{\mathrm{T}}$ be as above and invertible. Define the p-by-p matrix $J := \mathrm{I} - V^{\mathrm{T}} M V$. Then the inverse S^{-1} can be explicitly calculated as*

$$S^{-1} = \left[V J^{-1} V^{\mathrm{T}} + (1+\sigma)\left(\mathrm{I} - VV^{\mathrm{T}}\right)\left(\mathrm{I} + MVJ^{-1}V^{\mathrm{T}}\right)\right].$$

Proof. Let the columns of $V_\perp$ be a basis completion for the full space such that $\begin{pmatrix} V & V_\perp \end{pmatrix}$ is unitary. Let y solve the system

$$Sy = b. \tag{6}$$

For each y there exists a unique decomposition

$$y = V\bar{p} + V_\perp \bar{q}. \tag{7}$$

Substitution of (7) into (6) and multiplication with the unitary matrix $Q = \begin{pmatrix} V_\perp & V \end{pmatrix}^{\mathrm{T}}$ from the left yields a 2-by-2 block system which can be reduced to

$$\begin{pmatrix} -V_\perp^{\mathrm{T}} MV & 1/(1+\sigma)\mathrm{I} \\ \mathrm{I} - V^{\mathrm{T}} MV & 0 \end{pmatrix} \begin{pmatrix} \bar{p} \\ \bar{q} \end{pmatrix} = \begin{pmatrix} V_\perp^{\mathrm{T}} b \\ V^{\mathrm{T}} b \end{pmatrix} \tag{8}$$

by the properties $V_\perp^{\mathrm{T}} V = 0, V^{\mathrm{T}} V_\perp = 0, V_\perp^{\mathrm{T}} V_\perp = \mathrm{I}, V^{\mathrm{T}} V = \mathrm{I}$.

The second row of (8) can now be used to calculate $\bar{p}$ by inversion of $J := \mathrm{I} - V^{\mathrm{T}} MV$. The first row then yields $\bar{q}$. Thus, y can be expressed as

$$\begin{aligned} y &= \left[VJ^{-1}V^{\mathrm{T}} + (1+\sigma)\left(V_\perp V_\perp^{\mathrm{T}} + V_\perp V_\perp^{\mathrm{T}} MVJ^{-1}V^{\mathrm{T}} \right) \right] b \\ &= \left[VJ^{-1}V^{\mathrm{T}} + (1+\sigma)\left(\mathrm{I} - VV^{\mathrm{T}} \right) \left(\mathrm{I} + MVJ^{-1}V^{\mathrm{T}} \right) \right] b, \end{aligned}$$

by virtue of $V_\perp V_\perp^{\mathrm{T}} = \mathrm{I} - VV^{\mathrm{T}}$. ◇

Remark 1. From an algorithmical point of view, only the action of M on V is needed, which can be evaluated by p directional forward derivatives of the solution of the differential equation.

Lemma 3. *Let* $\operatorname{span} V$ *be an invariant subspace of the monodromy matrix* M *in the solution of equation* (3b)*. Then, the asymptotic linear convergence rate of the Newton-Shift-Picard method is*

$$\kappa_{\mathrm{NSP}} = \rho((1+\sigma)V_\perp^{\mathrm{T}} MV_\perp - \sigma \mathrm{I}).$$

Proof. By Lemma 1 and unitary basis transformation with $Q = \begin{pmatrix} V_\perp & V \end{pmatrix}^{\mathrm{T}}$ we have

$$\begin{aligned} \kappa_{\mathrm{NSP}} &= \rho(S^{-1}(S - (\mathrm{I} - M))) \\ &= \rho(Q^{\mathrm{T}} S^{-1} Q Q^{\mathrm{T}} (S - (\mathrm{I} - M))Q). \end{aligned}$$

Using Lemma 2, we explicitly calculate the matrices

$$\begin{aligned} A &= Q^{\mathrm{T}} S^{-1} Q \\ &= \begin{pmatrix} V_\perp & V \end{pmatrix}^{\mathrm{T}} \left[VJ^{-1}V^{\mathrm{T}} + (1+\sigma)\left(\mathrm{I} - VV^{\mathrm{T}} \right)\left(\mathrm{I} + MVJ^{-1}V^{\mathrm{T}} \right) \right] \begin{pmatrix} V_\perp & V \end{pmatrix} \\ &= \begin{pmatrix} (1+\sigma)\mathrm{I} & (1+\sigma)V_\perp^{\mathrm{T}} MVJ^{-1} \\ 0 & J^{-1} \end{pmatrix} = \begin{pmatrix} (1+\sigma)\mathrm{I} & 0 \\ 0 & J^{-1} \end{pmatrix}, \end{aligned}$$

and

$$\begin{aligned} B &= Q^{\mathrm{T}} (S - (\mathrm{I} - M))Q \\ &= \begin{pmatrix} V_\perp & V \end{pmatrix}^{\mathrm{T}} \left[1/(1+\sigma)\left(\mathrm{I} - VV^{\mathrm{T}} \right) + (\mathrm{I} - M)VV^{\mathrm{T}} - (\mathrm{I} - M) \right] \begin{pmatrix} V_\perp & V \end{pmatrix} \\ &= \begin{pmatrix} V_\perp & V \end{pmatrix}^{\mathrm{T}} \left[1/(1+\sigma)\mathrm{I} - (\mathrm{I} - M) \right] \left(\mathrm{I} - VV^{\mathrm{T}} \right) \begin{pmatrix} V_\perp & V \end{pmatrix} \\ &= \begin{pmatrix} V_\perp^{\mathrm{T}}(M - \sigma/(1+\sigma)\mathrm{I})V_\perp & V_\perp^{\mathrm{T}}(M - \sigma/(1+\sigma)\mathrm{I})V \\ 0 & 0 \end{pmatrix} \\ &= \begin{pmatrix} V_\perp^{\mathrm{T}} MV_\perp - \sigma/(1+\sigma)\mathrm{I} & 0 \\ 0 & 0 \end{pmatrix}. \end{aligned}$$

Here we used that span V is invariant under M and thus $V_{\perp}^{\mathrm{T}}MV = 0$. Finally, we obtain

$$\kappa_{\mathrm{NSP}} = \rho(AB) = \rho((1+\sigma)V_{\perp}^{\mathrm{T}}MV_{\perp} - \sigma \mathrm{I}).$$

This completes the proof. ◇

Theorem 1. *Let V span the so called "dominant" invariant subspace corresponding to the eigenvalues $\lambda_1, \dots, \lambda_p$ of M, and let all eigenvalues $\lambda_{p+1}, \dots, \lambda_{n_s}$ be enclosed in the closed disc $\overline{B}_r(c) \subset B_1(0) \subset \mathbf{C}$ with $r, c \in \mathbf{R}$. Then, the Newton-Shift-Picard method with shift $\sigma = c/(1-c)$ is locally linearly convergent with asymptotic convergence rate of*

$$\kappa_{\mathrm{NSP}} \leq \frac{r}{1-c}.$$

Equality holds if there is an $i \in \{p+1, \dots, n_s\}$ such that $\lambda_i \in \partial B_r(c)$.

Proof. Let $C = (1+\sigma)V_{\perp}^{\mathrm{T}}MV_{\perp} - \sigma\mathrm{I}$. The eigenvectors of C are the eigenvectors of $V_{\perp}^{\mathrm{T}}MV_{\perp}$ which correspond to the eigenvalues $\lambda_{p+1}, \dots, \lambda_{n_s}$ of M. By Lemma 3 we get

$$\begin{aligned}\kappa_{\mathrm{NSP}} &= \rho(C) \\ &= \max_{i=p+1,\dots,n_s} |(1+\sigma)\lambda_i - \sigma| \\ &= \max_{i=p+1,\dots,n_s} |\lambda_i - c|/(1-c) \leq r/(1-c).\end{aligned}$$

Equality holds if $|\lambda_i - c| = r$ for one i. ◇

Figure 2 shows a typical spectrum of the monodromy matrix. These spectra are characterized by real and complex conjugated pairs of eigenvalues which cluster around the origin. Only few eigenvalues have modulus larger than 1/2, and one eigenvalue is close to the unit circle, resulting in a rather slow linear convergence rate for the pure Picard method of about 0.98.

In order to illustrate the importance of Theorem 1, we refer to Figure 3 for the example of $p = 4$ and $\sigma = 0.28$. First of all, by proper choice of p and V, one can eliminate a few "slow" modes. In the figure, these correspond to the four right-most +-marks. The remaining spectrum consists of eigenvalues which lie inside the circle with radius $r' = 0.50$ (dashed circle). Thus, a pure Newton-Picard without shift ($\sigma = 0$) has an asymptotic linear convergence rate of about 0.5. Using the shift, this convergence rate can be further improved with negligible additional numerical effort by exploiting the asymmetric distribution of eigenvalues in the sense that the smallest enclosing disc has a (obviously real) center (small solid black circle) with a non-zero distance to the origin. In the example, an optimal shift of $\sigma = 0.28$ improves the convergence rate to $r = 0.36$ (centered solid black circle). The ×-marks display the eigenvalues of the "shifted" matrix $(1+\sigma)M - \sigma\mathrm{I}$.

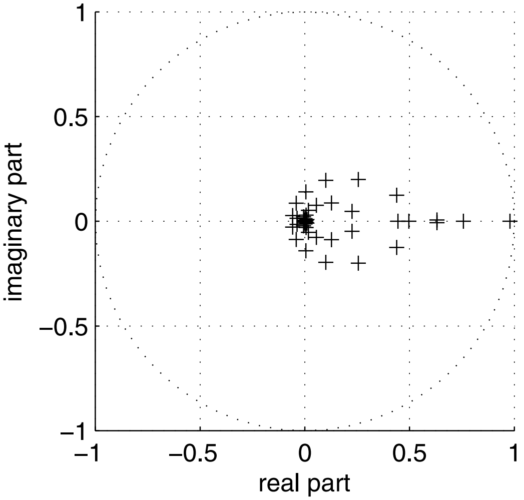


Fig. 2. Typical spectrum of the monodromy matrix M in the solution of problem (3).

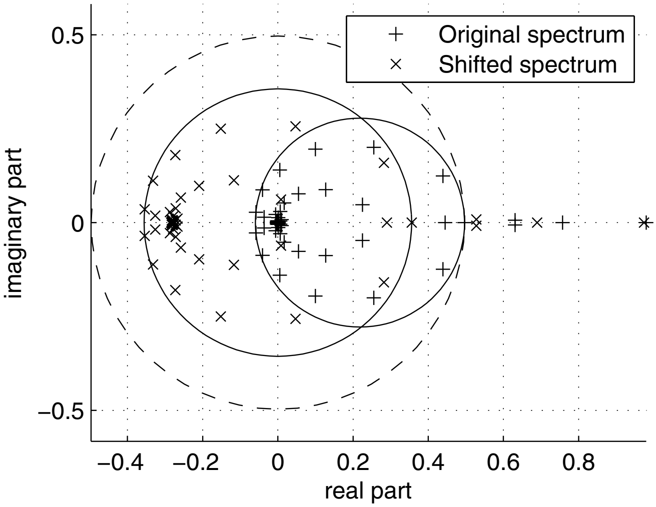


Fig. 3. Illustration of Theorem 1 for $p = 4$ and $\sigma = 0.28$ with contraction circles of radii $r' = 0.50$ and $r = 0.36$.

4.3 Numerical Effort of the Newton-Shift-Picard Method: Example SMB

We want to estimate the numerical effort of the Newton-Shift-Picard method for the SMB example. For simplicity, we assume that one forward simulation is as expensive as the evaluation of one directional forward derivative, which is needed to evaluate one matrix vector product Mv.

We calculate the number of iterations, which are needed for a reduction of the distance to the solution by a factor of 10^{-3} as the minimal $k \in \mathbf{N}$ such that

$$\kappa^k \leq 10^{-3},$$

with different values of κ coming from the previous theoretical investigations applied to the example.

To reduce the distance to the solution by a factor of 10^{-3}, a pure Picard method takes at most 335 iterations. One Picard iteration has the cost of one forward simulation. One step of a full Newton method costs already 384 forward derivatives plus one forward simulation and is thus not competitive with a pure Picard method. The Shift-Picard method with an optimal $\sigma = 0.85$ completes the reduction of 10^{-3} within at most 179 iterations. Because one Shift-Picard iterations also needs only one forward simulation, we get a reduction of effort to 53% of the pure Picard method. The effort for different choices p for the full Newton-Shift-Picard method with optimal shift is assembled in Table 1. We assume that one Newton-Shift-Picard iteration needs p additional matrix vector products. For a detailed description of the effort required to determine the subspaces, see Section 7.

p	1	2	4	6	10	20
σ	0.54	0.40	0.28	0.24	0.094	0.021
κ	0.63	0.48	0.36	0.31	0.16	0.086
Iters	15	10	7	6	4	3
#SIM	15	10	7	6	4	3
#MVP	15	20	28	36	40	60
Effort	30	30	35	42	44	63

Table 1. Comparison of numerical effort for the Newton-Shift-Picard method for different subspace sizes p. Legend: σ (optimal shift), κ (convergence rate), Iters (number of iterations), #SIM (number of forward simulations), #MVP (number of matrix vector products/directional forward derivatives), Effort (#SIM + #MVP)

5 The Newton-Picard Inexact SQP Method

We solve the optimization problem with an inexact Sequential Quadratic Programming method. As opposed to the well-known SQP methods where the

Hessians are approximated (for example by update methods) and the linearizations of the constraints in the subproblems are exact, we understand inexact SQP methods as those that also approximate the constraint Jacobians. Theoretical and numerical details can be found in Diehl et al. [5] and Wirsching [18]. Here, the approximation will be based on the Newton-Picard ideas.

5.1 Inexact SQP

First, we write problem (3) in the more suggestive form

$$\begin{aligned} \underset{w \in \mathbf{R}^n}{\text{minimize}} \quad & F(w) \\ \text{subject to} \quad & G_i(w) = 0, \quad i \in \mathcal{E}, \\ & G_j(w) \geq 0, \quad j \in \mathcal{I}. \end{aligned}$$

Starting from an initial guess w^0, a sequence of iterates

$$w^{k+1} = w^k + \Delta w^k, \quad \lambda^{k+1} = \lambda^k + \Delta\lambda^k$$

is generated from the primal-dual solution $(\Delta w^k, \Delta\lambda^k)$ of the quadratic subproblems

$$\begin{aligned} \underset{\Delta w^k \in \mathbf{R}^n}{\text{minimize}} \quad & \tfrac{1}{2}\Delta w^{k\mathrm{T}} B^k \Delta w^k + b^{k\mathrm{T}} \Delta w^k && \text{(9a)} \\ \text{subject to} \quad & G_i(w^k) + C_i^k \Delta w^k = 0, \quad i \in \mathcal{E}, && \text{(9b)} \\ & G_j(w^k) + C_j^k \Delta w^k \geq 0, \quad j \in \mathcal{I}, && \text{(9c)} \end{aligned}$$

with inexact Hessians B^k, exact gradient b^k of the Lagrangian

$$L(w^*, \lambda^*) = F(w^*) - \sum\nolimits_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i^* G_i(w^*),$$

and inexact rows C_i^k of the constraint Jacobians $\nabla G(w^k)^\mathrm{T}$. The exact gradient b^k is preferably evaluated by reverse mode of Algorithmic Differentiation [6]. In the view of Internal Numerical Differentiation, this is an adjoint solve of the differential equations, with the adjoint integration scheme added to the forward integrator [1].

5.2 QP Reduction

Let us assume that QP (9) can be written in the split form

$$\begin{aligned} \underset{(\Delta w_1^k, \Delta w_2^k) \in \mathbf{R}^{n_1+n_2}}{\text{minimize}} \quad & \frac{1}{2} \begin{pmatrix} \Delta w_1^k \\ \Delta w_2^k \end{pmatrix}^\mathrm{T} \begin{pmatrix} B_{11}^k & B_{12}^k \\ B_{21}^k & B_{22}^k \end{pmatrix} \begin{pmatrix} \Delta w_1^k \\ \Delta w_2^k \end{pmatrix} + \begin{pmatrix} b_1^k \\ b_2^k \end{pmatrix}^\mathrm{T} \begin{pmatrix} \Delta w_1^k \\ \Delta w_2^k \end{pmatrix} && \text{(10a)} \\ \text{subject to} \quad & G_i(w_1^k, w_2^k) + C_{1i}^k \Delta w_1^k + C_{2i}^k \Delta w_2^k = 0, \quad i \in \mathcal{E}_1, && \text{(10b)} \\ & G_i(w_1^k, w_2^k) + C_{1i}^k \Delta w_1^k + C_{2i}^k \Delta w_2^k = 0, \quad i \in \mathcal{E}_2, && \text{(10c)} \\ & G_j(w_1^k, w_2^k) + C_{1j}^k \Delta w_1^k + C_{2j}^k \Delta w_2^k \geq 0, \quad j \in \mathcal{I}, && \text{(10d)} \end{aligned}$$

such that the partial derivative $C_1 := (C_{1i}^k)_{i\in\mathcal{E}_1}$ is non-singular. Then, $\varDelta w_1^k$ is uniquely determined by the affine-linear dependence on $\varDelta w_2^k$ via equation (10b) and can be eliminated. The remaining QP only has the dimension n_2 instead of $n_1 + n_2$. For details, including techniques to recover the Lagrange multipliers of the eliminated constraint (10b) can be recovered from the primal-dual solution of the reduced QP by KKT transformation rules, see [10].

We shall describe the elimination procedure for problem (3) in the following. We drop the iteration index, and furthermore omit the bound constraints because they are not changed by the QP tranformations. We use the variable partition $\varDelta w_1 = \varDelta s^0$ and $\varDelta w_2 = (\varDelta v, \varDelta q)$ and eliminate the constraint (3b). The linearized constraint system in the subproblems is

$$\begin{bmatrix} \mathrm{I} - PX_s & -PX_v & -PX_q \\ R_m Y_s & R_m Y_v + R_v & R_m Y_q \end{bmatrix} \begin{bmatrix} \varDelta s^0 \\ \varDelta v \\ \varDelta q \end{bmatrix} + \begin{bmatrix} s^0 - Ps^0 \\ r \end{bmatrix} \begin{Bmatrix} = \\ \geq \end{Bmatrix} 0.$$

We approximate the upper left block with an invertible matrix S choice of which was discussed in Sections 4. The approximated system is premultiplied with the matrix

$$\begin{bmatrix} -S^{-1} & 0 \\ -R_m Y_s S^{-1} & \mathrm{I} \end{bmatrix}$$

to yield the new block-triangular approximated system

$$\begin{bmatrix} -\mathrm{I} & S^{-1}PX_v & S^{-1}PX_q \\ 0 & R_m(Y_v + Y_s S^{-1}PX_v) + R_v & R_m(Y_q + Y_s S^{-1}PX_q) \end{bmatrix} \begin{bmatrix} \varDelta s^0 \\ \varDelta v \\ \varDelta q \end{bmatrix} + \begin{bmatrix} -S^{-1}(s^0 - Ps^N) \\ r - R_m Y_s S^{-1}(s^0 - Ps^N) \end{bmatrix} \begin{Bmatrix} = \\ \geq \end{Bmatrix} 0. \quad (11)$$

Then, the first line shows the affine-linear dependence of $\varDelta s^0$ on $\varDelta v$ and $\varDelta q$. Thus, we arrive at a QP only in the space of parameters and control variables. Eventually, also the Lagrange multipliers of the eliminating constraint can be recovered from the primal QP solution $\varDelta s^0, \varDelta q, \varDelta v$ and the reduced dual QP solution $\varDelta\mu$ via

$$\varDelta\lambda = -S^{-\mathrm{T}} \left(-B_{11}\varDelta s^0 - B_{12} \begin{pmatrix} \varDelta v \\ \varDelta v \end{pmatrix} - b_1 + Y_s^{\mathrm{T}} R_m^{\mathrm{T}} \varDelta\mu \right).$$

The term $Y_s^{\mathrm{T}} R_m^{\mathrm{T}} \varDelta\mu$ is computed with one adjoint solve.

6 Simultaneous Approximation of Dominant Subspaces

The choice of the orthonormal matrices V^k has not been discussed so far. The key idea of Lust et al. [11] is to use an iterative method for the calculation of the dominant subspace of the monodromy matrix simultaneously with the iterations of the approximate Newton method. This extends very naturally to the SQP case, which is also a Newton-type method, once the active set of the solution is found.

We employ a variant of the Subspace Iteration which is used by Lust et al. [11] and was originally developed by Stewart [16] to calculate the dominant subspaces of a complex non-Hermitian n_s-by-n_s matrix M using only matrix-vector products Mv. The subspaces are represented as the span of an orthonormal n_s-by-p matrix V. The method is efficient for $p \ll n_s$.

In its simplest form, the Subspace Iteration comprises three steps: First, calculate MV, second, orthogonalize MV, and third, perform a Schur-Rayleigh-Ritz (SRR) approximation to update V. This procedure is repeated until convergence of span(V).

The Subspace Iteration is intertwined with the inexact SQP method such that per SQP iteration only a small number of Subspace Iterations has to be performed to update the basis of the approximation of the dominant subspace.

7 Numerical Effort

7.1 Effort per SQP iteration

The numerical effort of the proposed method can be best described by the number of forward simulations (FS), forward directional derivatives (FDD), and adjoint directional derivatives (ADD) of the differential equations, because typically more than 95% of the method is spent in these parts of the algorithm. Additionally, the effort for directional derivatives can be estimated by small constants times the effort for a forward simulation (for details see Albersmeyer [2]).

Per iteration, the method needs one FS of the differential equations in order to evaluate the constraint residuals and the objective functional value.

The number of ADD per SQP step is three. The first is needed to calculate the subproblem gradient $g^k = \nabla L(w^{k+1}, \lambda^{k+1}, \mu^{k+1})$, the second for a Hessian update from the Lagrangian gradients g^k and $\nabla L(w^k, \lambda^{k+1}, \mu^{k+1})$. The third is needed for the recovery of the Lagrange multipliers steps $\Delta\lambda^k$ of equation (10b).

The number of FDD is composed of the following contributions: For the evaluation of X_v, X_q, Y_v, Y_q, it is necessary to perform $n_v + ((N+1)/2)n_q$ FDD. Furthermore, the evaluation of MV takes another $p_{\min}$ FDD plus a small number p_{add} additional FDD for the Subspace Iterations (compare also the Section 7.2). The evaluation of the Jacobian terms $Y_s S^{-1} P(X_v \;\, X_q)$ and the residual term $Y_s S^{-1}(s^0 - Ps^N)$ in the eliminated QP constraint system (11) contribute with additional $n_v + Nn_q + 1$ FDD.

7.2 Additional Effort for Subspace Iterations

Finally, we shortly discuss the extra work of p_{add} FDD which is needed for the piggy-back Subspace Iteration: In the first iterations and far away from the solution, it may be necessary to perform more Subspace Iterations in order to guarantee a sufficiently accurate approximation of the dominant subspace. Close to the solution, however, the matrices M^k will eventually differ only slightly and the dominant subspaces will become stationary, such that from a certain iteration on, no extra directional forward derivatives of the differential equations have to be calculated, i.e., $p_{\text{add}} = 0$. We also want to remark that the numerical effort for the linear algebra necessary for the Subspace Iteration is negligible compared to the cost for the solution and differentiation of the differential equations because $p_{\max} \ll n_s$ can be assumed.

7.3 Comparison with Exact Jacobian SQP

An SQP method with exact Jacobians needs one FS, zero ADD, and $n_s + n_v + ((N+1)/2)n_q$ FDD. The grand total of FS, ADD, and FDD is shown in Table 2. The fundamental improvement of the inexact SQP method over the exact SQP method is the independence of the state discretization degrees of freedom n_s.

Method	FS	ADD	FDD
Inexact SQP	1	3	$2n_v + ((3N+1)/2)n_q + p_{\min} + p_{\text{add}} + 1$
Exact SQP	1	0	$n_s + n_v + ((N+1)/2)n_q$

Table 2. Comparison of the numerical effort of the inexact and exact SQP method per SQP step.

8 Numerical Results

The Newton-Picard Inexact SQP method was implemented in the software package MUSCOD-II [9] in a single-shooting version.

A self-convergence test for the ModiCon optimization scenario was conducted on an SMB configuration of $N_{\text{col}} = 6$ columns. Each column was discretized with four Discontinuous Galerkin elements of polynomial degree three, resulting in a total number of $n_s = 384$ periodic state variables. The feed concentration was discretized as piecewise constant on the grid $t_i = T\tau_i$, with $\tau_0 = 0, \tau_i = 0.6 + 0.05(i-1), i = 1, \ldots, 9$. Figure 4 depicts the optimal feed concentration profile. All the feed mass is inserted as late as possible in the switching period. In Figure 5 shows the distance to the reference solution in a weighted norm of primal and dual variables for varying values of

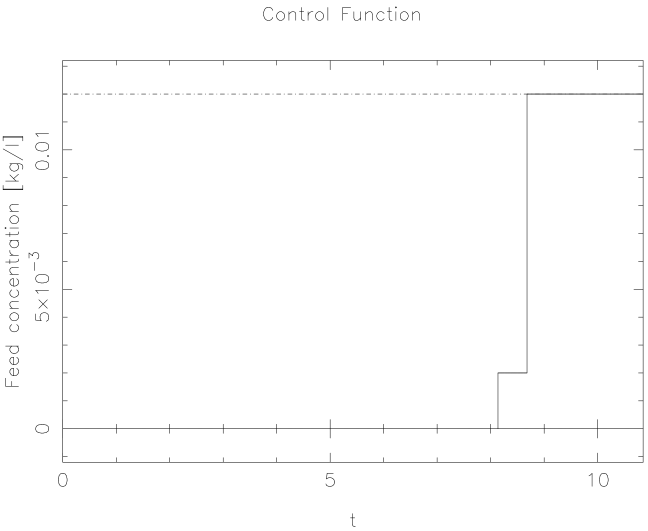


Fig. 4. Optimal feed concentration for the ModiCon process.

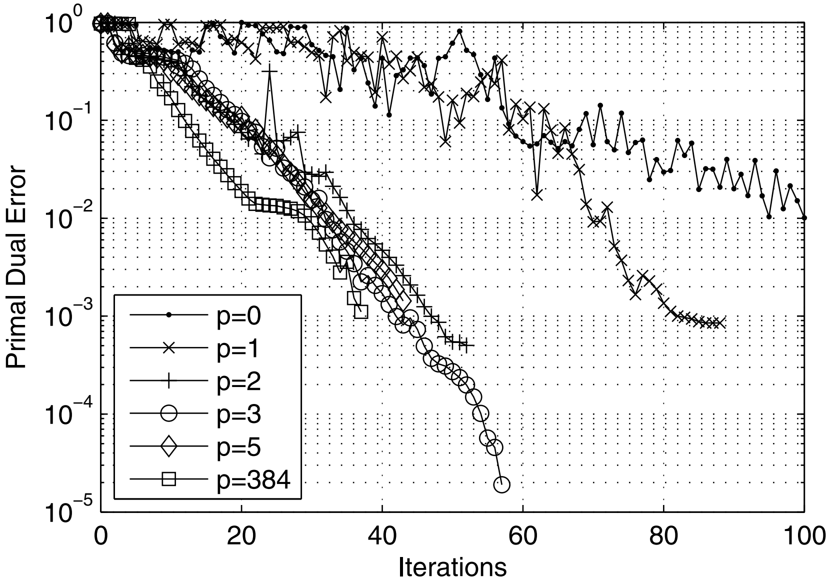


Fig. 5. Self-convergence of the Newton-Picard Inexact SQP method.

$p = 0, 1, 2, 3, 5, 384$. One can see that starting from $p = 2$, a linear convergence rate is achieved on the average which is almost as good as the SQP convergence rate with exact monodromy matrix ($p = 384$), especially close to the solution. The counterintuitive fact that we obtain faster convergence for $p = 3$ than for $p = 5$ is indeed also theoretically possible. This non-monotonic behavior shall be discussed in more detail in a future paper.

9 Conclusion

In the operating regime of the ModiCon variant of the SMB process, the forward problem of finding a cyclic steady state for fixed controls can be efficiently solved by the Newton-Picard method. Only few Newton directions ($p = 1$ or $p = 2$) are needed to achieve good contraction. The contraction rate can be further improved without additional numerical effort by introducing an overrelaxation for the fast modes. We have shown how the overrelaxation can be cast in a Newton-type framework, and have presented convergence results for the Newton-Shift-Picard method.

We have demonstrated how to use Jacobian approximations from the Newton-Shift-Picard method in an inexact SQP method. We have implemented the method and numerical results for the ModiCon SMB process show that already low subspace dimensions between 3 and 5 are sufficient to yield a fast linear convergence rate, while each inexact SQP iteration is less expensive than a conventional SQP iteration, due to less evaluations of forward derivatives. This leads to a reduction in the complexity of the algorithm with respect to the degrees of freedom n_s of the state discretization.

Acknowledgements

This work was supported by the German Research Foundation (DFG) within the priority program SPP1253 under grant BO864/12-1 and by the German Federal Ministry of Education and Research (BMBF) under grant 03BONCHD.

References

1. Albersmeyer J, Bock HG (2008) Sensitivity Generation in an Adaptive BDF-Method. In: Bock HG, Kostina E, Phu XH, Rannacher R (eds), Modeling, Simulation and Optimization of Complex Processes: Proceedings of the International Conference on High Performance Scientific Computing, March 6-10, 2006, Hanoi, Vietnam. Springer
2. Albersmeyer J and Bock HG (2009) Efficient sensitivity generation for large scale dynamic systems. Technical report, SPP 1253 Preprints, University of Erlangen

3. Bock HG (1987) Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen. Volume 183 of Bonner Mathematische Schriften, Universität Bonn, Bonn.
4. De la Torre V, Walther A, Biegler LT (2004) Optimization of periodic adsorption processes with a novel problem formulation and nonlinear programming algorithm. In: AD 2004 – Fourth International Workshop on Automatic Differentiation, July 19-23, 2004, Argonne National Laboratory, USA
5. Diehl M, Walther A, Bock HG, Kostina E (2010) An adjoint-based SQP algorithm with quasi-Newton Jacobian updates for inequality constrained optimization. Optim Method Softw 25:531–552
6. Griewank A (2000) Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation. Number 19 in Frontiers in Appl Math, SIAM, Philadelphia
7. Gu T (1995) Mathematical modelling and scale up of liquid chromatography. Springer, New York
8. Kawajiri Y, Biegler LT (2006) Large scale nonlinear optimization for asymmetric operation and design of Simulated Moving Beds. J Chromatogr A 1133:226–240
9. Leineweber DB (1996) The theory of MUSCOD in a nutshell. IWR-Preprint 96-19, Universität Heidelberg
10. Leineweber DB (1999). Efficient reduced SQP methods for the optimization of chemical processes described by large sparse DAE models. Volume 613 of Fortschritt-Berichte VDI Reihe 3, Verfahrenstechnik, VDI Verlag, Düsseldorf
11. Lust K, Roose D, Spence A, Champneys AR (1998) An adaptive Newton-Picard algorithm with subspace iteration for computing periodic solutions. SIAM J Sci Comput 19:1188–1209
12. Nilchan S, Pantelides C (1998) On the optimisation of periodic adsorption processes. Adsorption 4:113–147
13. Schmidt-Traub, ed (2005) Preparative Chromatography of Fine Chemicals and Pharmaceuticals. Harwell Report, Wiley-VCH
14. Schramm H, Kaspereit M, Kienle A, Seidel-Morgenstein A (2003) Simulated Moving Bed process with cyclic modulation of the feed concentration. J Chromatogr 1006:77–86
15. Schramm H, Kienle A, Kaspereit M, Seidel-Morgenstein A (2003) Improved operation of Simulated Moving Bed processes through cyclic modulation of feed flow and feed concentration. Chem Eng Sci 58:5217–5227
16. Stewart GW (1976) Simultaneous iteration for computing invariant subspaces of non-Hermitian matrices. Numer Math 25:123–136
17. Toumi A, Engell S, Diehl M, Bock HG, Schlöder JP (2007) Efficient optimization of Simulated Moving Bed Processes, Chem Eng Process 46:1067–1084
18. Wirsching L (2006) An SQP algorithm with inexact derivatives for a Direct Multiple Shooting method for optimal control problems. Diploma thesis, Universität Heidelberg

On the Optimization of Steady Bingham Flow in Pipes

Juan Carlos De Los Reyes[1,2]

[1] Department of Mathematics, Escuela Politécnica Nacional de Quito, Ecuador.
[2] Institute of Mathematics, Technical University of Berlin, Germany.
delosreyes@math.tu-berlin.de

Summary. Optimal control problems of Bingham fluid flow in pipes are considered. After introducing a family of regularized problems, convergence of the regularized solutions towards the orignal one is verified. An optimality condition for the original problem is obtained as limit of the regularized optimality systems. For the solution of each regularized system a semismooth Newton algorithm is proposed.

We consider a pipe with cross section $\Omega \subset \mathbb{R}^2$ convex and bounded. The tracking type optimal control problem for a Bingham flow passing through Ω is given by

$$\min_{(y,u)\in H_0^1(\Omega)\times L^2(\Omega)} \quad J(y,u) = \frac{1}{2}\int_\Omega |y-y_d|^2 \, dx + \frac{\alpha}{2}\int_\Omega |u|^2 \, dx \tag{1a}$$

subject to

$$\nu \int_\Omega (\nabla y, \nabla(v-y))_{\mathbb{R}^2} \, dx + g\int_\Omega |\nabla v| \, dx - g \int_\Omega |\nabla y| \, dx \\ \geq \int_\Omega (f+u)(v-y)\, dx, \text{ for all } v \in H_0^1(\Omega), \tag{1b}$$

where $\nu > 0$ denotes the viscosity coefficient of the fluid and $g > 0$ stands for the plasticity threshold of the material.

Bingham materials are characterized by the presence of a so-called yield stress: they behave like solids in regions where the stresses are small and like incompressible fluids where the stresses are larger than a plasticity threshold. The solid regions, in addition, are of two types:

$$\mathcal{I}_S = \{x \in \Omega : \nabla y(x) = 0, y(x) = 0\} \text{ and } \mathcal{I}_N = \{x \in \Omega : \nabla y(x) = 0, y(x) > 0\}.$$

The first one corresponds to the stagnation zones, while the second one is called nucleus and corresponds to the sector where the Bingham flow moves like a rigid solid. If Ω is strongly symmetric and simply connected, it is known

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*, DOI 10.1007/978-3-642-12598-0_32, © Springer-Verlag Berlin Heidelberg 2010

(see [10]) that the set $\mathcal{I}_N$ is also simply connected. Moreover, in such cases the nucleus is unique and its internal boundary is convex.

Existence and uniqueness of solutions to (1b) can be obtained by standard techniques (see [7]). Moreover, by using Fenchel duality theory, inequality (1b) can be equivalently written as:

$$a(y,v)+(q,\nabla v)=(f+u,v), \qquad \text{for all } v\in H_0^1(\Omega), \tag{2a}$$

$$(q(x),\nabla y(x))_{\mathbb{R}^2}=g|\nabla y(x)| \qquad \text{a.e. in } \Omega, \tag{2b}$$

$$|q(x)|\leq g \qquad \text{a.e. in } \Omega. \tag{2c}$$

where $q\in\mathbb{L}^2(\Omega):=\left(L^2(\Omega)\right)^2$ stands for the dual variable, $a(v,w):=\nu(\nabla v,\nabla w)$ for all $v,w\in H_0^1(\Omega)$, and $f\in L^2(\Omega)$. We denote by $\|\cdot\|_X$ the norm in a Banach space X and by $(\cdot,\cdot)_Y$ the scalar product of a Hilbert space Y. For the space $L^2(\Omega)$ no subindex is used.

Existence of an optimal solution for the control problem (1) can be obtained by standard arguments (see e.g. [1, pg.151]). Due to the non-differentiability of the control-to-state operator resulting from (1b), however, the derivation of a detailed necessary optimality condition turns out to be challenging. Moreover, in order to obtain a solution for (1) numerically, an appropriate approximation technique has to be considered.

In what follows we propose a unifying regularization approach, which enables us, on one hand, to derive an optimality system for (1) and, on the other hand, to approximate an optimal solution by using a Newton type algorithm.

1 Regularized problem

We start by considering the following regularized version of the primal-dual system:

$$a(y_\gamma,v)+(q_\gamma,\nabla v)=(f+u,v), \text{ for all } v\in H_0^1(\Omega) \tag{3a}$$

$$q_\gamma=\frac{g\gamma\nabla y_\gamma}{\max(g,\gamma|\nabla y_\gamma|)} \text{ a.e. in } \Omega. \tag{3b}$$

Such a system results from a Tikhonov regularization of the dual problem and has been previously considered for the numerical solution of some variational inequalities of the second kind by semismooth Newton methods (see [11, 9, 4]).

Based on the regularization of the governing variational inequality given by (3) and a local smoothing of the *max* function, a family of regularized optimal control problems is introduced and studied next. The additional smoothing enables us to obtain differentiability properties of the problem and is important in the construction of the approximation algorithm given in Section 3.

The local C^1-smoothing of the max function is given by

$$\max_c(0,x) = \begin{cases} x & \text{if } x \geq \frac{1}{2c} \\ \frac{c}{2}\left(x + \frac{1}{2c}\right)^2 & \text{if } |x| \leq \frac{1}{2c} \\ 0 & \text{if } x \leq -\frac{1}{2c} \end{cases} \tag{4}$$

and its derivative by

$$\max{}'_c(0,x) = \begin{cases} 1 & \text{if } x \geq \frac{1}{2c} \\ c\left(x + \frac{1}{2c}\right) & \text{if } |x| \leq \frac{1}{2c} \\ 0 & \text{if } x \leq -\frac{1}{2c}. \end{cases} \tag{5}$$

According to (3) and the proposed smoothing of the max function, we introduce, for each $\gamma > 0$, the following regularized optimal control problem:

$$\min_{(y,u)\in H_0^1(\Omega)\times L^2(\Omega)} J(y,u) = \frac{1}{2}\int_\Omega |y - y_d|^2\, dx + \frac{\alpha}{2}\int_\Omega |u|^2\, dx \tag{6a}$$

subject to

$$a(y,v) + \left(\frac{g\gamma\nabla y}{\max_\gamma(g,\gamma|\nabla y|)}, \nabla v\right) = (f+u,v), \text{ for all } v \in H_0^1(\Omega). \tag{6b}$$

Theorem 1. *Let $g > 0$, $\gamma > 0$ and $u_\gamma \in L^2(\Omega)$. There exist a unique solution $y_\gamma \in H_0^1(\Omega)$ to the equation*

$$a(y,v) + \left(\frac{g\gamma\nabla y}{\max_\gamma(g,\gamma|\nabla y|)}, \nabla v\right) = (f+u_\gamma,v), \text{ for all } v \in H_0^1(\Omega). \tag{7}$$

Moreover, if u_γ converges to u strongly in $L^2(\Omega)$ as $\gamma \to \infty$, then the corresponding sequence of solutions $\{y_\gamma\}$ converges to the solution y of (1b)*, with $f + u$ on the right hand side, strongly in $H_0^1(\Omega)$, as $\gamma \to \infty$.*

Proof. For the complete proof we refer to [5].

Additionally, it can be verified that there exists an optimal solution for problem (6). Moreover, the sequence $\{u_\gamma\}$ of solutions to (6) contains a weakly convergent subsequence and any weak accumulation point of $\{u_\gamma\}$ is an optimal solution for (1). Considering, in addition, the special structure of the cost functional, $u_\gamma \to \bar{u}$ strongly in U, where $\bar{u}$ stands for an optimal solution to (1).

Proposition 1. *Let $y_\gamma \in H_0^1(\Omega)$ and $h \in L^2(\Omega)$. There exists a unique solution $z \in H_0^1(\Omega)$ to the linearized equation*

$$a(z,v) + g\gamma \int\limits_\Omega \left(\frac{\nabla z}{\widetilde{\max}_\gamma}, \nabla v\right)_{\mathbb{R}^2} ds - g\gamma \int\limits_{\mathcal{A}_\gamma} \left(\frac{\nabla y_\gamma}{\widetilde{\max}_\gamma}\, \frac{\gamma(\nabla y_\gamma, \nabla z)_{\mathbb{R}^2}}{|\nabla y_\gamma|}, \frac{\nabla v}{\widetilde{\max}_\gamma}\right)_{\mathbb{R}^2} ds$$
$$- g\gamma \int\limits_{\mathcal{S}_\gamma} \gamma\left(\gamma|\nabla y_\gamma| - g + \frac{1}{2\gamma}\right)\left(\frac{\nabla y_\gamma}{\widetilde{\max}_\gamma}\, \frac{\gamma(\nabla y_\gamma, \nabla z)_{\mathbb{R}^2}}{|\nabla y_\gamma|}, \frac{\nabla v}{\widetilde{\max}_\gamma}\right)_{\mathbb{R}^2} ds$$
$$= (h,v), \textit{ for all } v \in H_0^1(\Omega), \tag{8}$$

where $\widetilde{\max}_\gamma := \max_\gamma(g, \gamma|\nabla y_\gamma|)$, $\mathcal{A}_\gamma = \{x \in \Omega : \gamma|\nabla y_\gamma(x)| - g \geq \frac{1}{2\gamma}\}$, $\mathcal{S}_\gamma = \{x \in \Omega : |\gamma|\nabla y_\gamma(x)| - g| \leq \frac{1}{2\gamma}\}$ *and* $\mathcal{I}_\gamma = \Omega \backslash (\mathcal{A}_\gamma \cup \mathcal{S}_\gamma)$.

Proof. Choosing $v = z$, the left hand side of the last equality takes the form

$$a(z,z) + g\gamma \int\limits_{\Omega} \frac{|\nabla z|^2}{\widetilde{\max}_\gamma}\, ds - g\gamma \int\limits_{\mathcal{A}_\gamma} \frac{\gamma(\nabla y_\gamma, \nabla z)^2_{\mathbb{R}^2}}{|\nabla y_\gamma| \widetilde{\max}^2_\gamma}\, ds$$
$$- g\gamma \int\limits_{\mathcal{S}_\gamma} \gamma \left(\gamma|\nabla y_\gamma| - g + \frac{1}{2\gamma}\right) \frac{\gamma(\nabla y_\gamma, \nabla z)^2_{\mathbb{R}^2}}{|\nabla y_\gamma| \widetilde{\max}^2_\gamma}\, ds. \quad (9)$$

Considering that $\widetilde{\max}_\gamma \geq \max(g, \gamma|\nabla y_\gamma|)$ a.e. on Ω, $\frac{\gamma|\nabla y_\gamma|}{\max(g,\gamma|\nabla y_\gamma|)} \leq 1$ a.e. on Ω and $\gamma(\gamma|\nabla y_\gamma| - g + \frac{1}{2\gamma}) \leq 1$ a.e. on $\mathcal{S}_\gamma$, and using Cauchy-Schwarz, it follows that

$$g\gamma \int_{\mathcal{S}_\gamma} \gamma(\gamma|\nabla y_\gamma| - g + \frac{1}{2\gamma})\, \frac{\gamma(\nabla y_\gamma, \nabla z)^2_{\mathbb{R}^2}}{|\nabla y_\gamma| \widetilde{\max}^2_\gamma}\, ds \leq g\gamma \int_{\mathcal{S}_\gamma} \frac{|\nabla z|^2}{\widetilde{\max}_\gamma}\, ds. \quad (10)$$

Similarly, we get that

$$g\gamma \int_{\mathcal{A}_\gamma} \frac{\gamma(\nabla y_\gamma, \nabla z)^2_{\mathbb{R}^2}}{|\nabla y_\gamma| \widetilde{\max}^2_\gamma}\, ds \leq g\gamma \int_{\mathcal{A}_\gamma} \frac{|\nabla z|^2}{\widetilde{\max}_\gamma}\, ds. \quad (11)$$

Altogether we obtain that

$$g\gamma \int_{\Omega} \frac{|\nabla z|^2}{\widetilde{\max}_\gamma}\, ds - g\gamma^2 \int_{\mathcal{A}_\gamma} \frac{(\nabla z, \nabla y_\gamma)^2_{\mathbb{R}^2}}{\widetilde{\max}^2_\gamma\, |\nabla y_\gamma|}\, ds$$
$$- g\gamma^2 \int_{\mathcal{S}_\gamma} \gamma(\gamma|\nabla y_\gamma| - g + \frac{1}{2\gamma})\, \frac{(\nabla z, \nabla y_\gamma)^2_{\mathbb{R}^2}}{\widetilde{\max}^2_\gamma\, |\nabla y_\gamma|}\, ds \geq g\gamma \int_{\mathcal{I}_\gamma} \frac{|\nabla z|^2}{\widetilde{\max}_\gamma}\, ds. \quad (12)$$

The result then follows from the Lax-Milgram theorem. □

Let us now introduce the control-to-state operator $G : L^2(\Omega) \to H^1_0(\Omega)$, which assigns to each control $u \in L^2(\Omega)$ the correspondent solution to equation (7). The governing equation in this case corresponds to a PDE of quasilinear type and it can be proved (see [3, Thm. 3.1]) that G is Gateaux differentiable. Moreover, its derivative $z = G'(u)v$ corresponds to the unique solution of equation (8).

Theorem 2. *Let* (y_γ, u_γ) *be an optimal solution of the regularized problem* (6)*. Then it satisfies the following optimality system:*

$$a(y_\gamma, v) + (q_\gamma, \nabla v) = (f + u_\gamma, v), \ \textit{for all } v \in H_0^1(\Omega), \tag{13a}$$

$$q_\gamma = \frac{g\gamma \nabla y_\gamma}{\widetilde{\max}_\gamma} \quad \textit{in } \mathbb{L}^2(\Omega), \tag{13b}$$

$$a(p_\gamma, v) + (\lambda, \nabla v) = -\int_\Omega (y_\gamma - y_d) v \, dx, \ \textit{for all } v \in H_0^1(\Omega), \tag{13c}$$

$$\begin{aligned} \lambda := g\gamma \frac{\nabla p_\gamma}{\widetilde{\max}_\gamma} - g\gamma^2 \chi_{\mathcal{A}_\gamma} \frac{(\nabla p_\gamma, \nabla y_\gamma)_{\mathbb{R}^2}}{\widetilde{\max}_\gamma^2} \frac{\nabla y_\gamma}{|\nabla y_\gamma|} \\ - g\gamma^3 \chi_{\mathcal{S}_\gamma} (\gamma |\nabla y_\gamma| - g + \frac{1}{2\gamma}) \frac{(\nabla p_\gamma, \nabla y_\gamma)_{\mathbb{R}^2}}{\widetilde{\max}_\gamma^2} \frac{\nabla y_\gamma}{|\nabla y_\gamma|}, \end{aligned} \tag{13d}$$

$$\alpha u_\gamma = p_\gamma, \tag{13e}$$

where χ_D denotes the indicator function of a set D.

Proof. Let $T : U \to \mathbb{R}$ be the reduced cost functional defined by $T(u) := J(G(u), u)$. From the structure of J and due to the differentiability of G we obtain that u_γ satisfies the equality $T'(u_\gamma)h = 0, \text{ for all } h \in L^2(\Omega)$.

Introducing p_γ as the unique solution to the adjoint equation (13c), where $\lambda \in \mathbb{L}^2(\Omega)$ is given by (13d), we obtain that

$$\begin{aligned} T'(u_\gamma)h &= (y_\gamma - y_d, z) + \alpha(u_\gamma, h)_U = -a(p_\gamma, z) - (\lambda, \nabla z) + \alpha(u_\gamma, h)_U \\ &= -a(z, p_\gamma) - g\gamma \int_\Omega \left(\frac{\nabla z}{\widetilde{\max}_\gamma}, \nabla p_\gamma \right) ds + g\gamma \int_{\mathcal{A}_\gamma} \left(\frac{\nabla y_\gamma}{\widetilde{\max}_\gamma} \frac{\gamma\,(\nabla y_\gamma, \nabla z)}{|\nabla y_\gamma|}, \frac{\nabla p_\gamma}{\widetilde{\max}_\gamma} \right) ds \\ &+ g\gamma \int_{\mathcal{S}_\gamma} \gamma \left(\gamma |\nabla y_\gamma| - g + \frac{1}{2\gamma} \right) \left(\frac{\nabla y_\gamma}{\widetilde{\max}_\gamma} \frac{\gamma(\nabla y_\gamma, \nabla z)}{|\nabla y_\gamma|}, \frac{\nabla p_\gamma}{\widetilde{\max}_\gamma} \right) ds + \alpha(u_\gamma, h)_U. \end{aligned}$$

From Proposition 1 we consequently obtain (13e). □

2 Optimality system

Next an optimality condition for the original optimal control problem (1) is obtained as limit of the regularized optimality systems (13).

Theorem 3. *Let $\bar{u}$ be an optimal solution for* (1) *and $\{u_\gamma\}$ a convergent subsequence of solutions to* (6) *such that $u_\gamma \to \bar{u}$ in $L^2(\Omega)$, as $\gamma \to \infty$. There exists a subsequence (denoted in the same way) and $p \in H_0^1(\Omega)$, $\lambda \in \mathbb{L}^2(\Omega)$ such that*

$$\begin{aligned} \nabla y_\gamma(x) &\to \nabla \bar{y}(x) && \textit{a.e. in } \Omega, \\ p_\gamma &\rightharpoonup p && \textit{weakly in } H_0^1(\Omega) \textit{ (strongly in } L^2(\Omega)), \\ -\Delta p_\gamma &\rightharpoonup -\Delta p && \textit{weakly in } H^{-1}(\Omega), \\ \lambda &\rightharpoonup \lambda && \textit{weakly in } \mathbb{L}^2(\Omega). \end{aligned}$$

where Δ denotes the Laplacian operator. Moreover, the multipliers (p, λ) satisfy together with the optimal solution of the original control problem $(\bar{y}, \bar{u})$ the following optimality system:

$$a(\bar{y}, v) + (\bar{q}, \nabla v) = (f + \bar{u}, v), \text{ for all } v \in H_0^1(\Omega) \tag{14a}$$

$$(\bar{q}, \nabla \bar{y}) = |\nabla \bar{y}| \text{ a.e. in } \Omega \tag{14b}$$

$$|\bar{q}| \leq g \text{ a.e. in } \Omega \tag{14c}$$

$$a(p, v) + (\lambda, \nabla v) = -\int_\Omega (\bar{y} - y_d) v \, dx, \text{ for all } v \in H_0^1(\Omega) \tag{14d}$$

$$\alpha \bar{u} = p \text{ a.e. in } \Omega \tag{14e}$$

$$\int_\Omega \lambda \cdot \nabla p \, dx \geq 0 \tag{14f}$$

$$\nabla p = 0 \text{ a.e. in } \mathcal{I} := \{x \in \Omega : \nabla \bar{y}(x) = 0\}, \tag{14g}$$

In addition, if $\mathcal{I}_N$ is Lipschitz and $\overline{\mathcal{I}}_N \subset \Omega$, then

$$\operatorname{div} \lambda = \bar{y} - y_d \text{ in } H^{-1}(\mathcal{I}_N). \tag{15}$$

Proof. Theorem [5, Thm.5.1] may be applied and system (14) is obtained.

Let us now consider test functions $\tilde{v} \in H_0^1(\Omega)$ of the following form

$$\tilde{v} = \begin{cases} v \text{ in } \mathcal{I}_N \\ 0 \text{ elsewhere,} \end{cases}$$

where $v \in H_0^1(\mathcal{I}_N)$. It then follows from the adjoint equation (14d) that

$$\nu \int_{\mathcal{I}_N} (\nabla p, \nabla v) \, dx + \int_{\mathcal{I}_N} (\lambda, \nabla v) \, dx = -\int_{\mathcal{I}_N} (\bar{y} - y_d) v \, dx, \text{ for all } v \in H_0^1(\mathcal{I}_N).$$

Since by (14g) $\nabla p = 0$ a.e. on $\mathcal{I}$ we obtain that

$$\int_{\mathcal{I}_N} (\lambda, \nabla v) \, dx = -\int_{\mathcal{I}_N} (\bar{y} - y_d) v \, dx, \text{ for all } v \in H_0^1(\mathcal{I}_N),$$

which, by applying integration by parts, yields (15).

Note that in order to obtain (15) we assumed some properties about the nucleus of the flow. Such properties hold in many cases (see [10] and the references therein). It is important to distinguish, however, between the two types of inactive sets ($\mathcal{I}_N$ and $\mathcal{I}_S$), since stagnation zones are usually attached to the boundary of the domain Ω.

From equation (14g) we also conclude that the adjoint variable p has a constant value on the sectors where the material behaves like a rigid solid.

3 Semi-smooth Newton algorithm and numerical tests

Based on the structure of the regularized optimality systems given by (13) we propose next a generalized Newton algorithm for its numerical approximation.

3.1 Algorithm

By introducing the operator $F : H_0^1(\Omega) \times H_0^1(\Omega) \to H^{-1}(\Omega) \times H^{-1}(\Omega)$ given by

$$F(y,p) = \begin{pmatrix} a(y,\cdot) + g\gamma \left(\frac{\nabla y}{\widetilde{\max}_\gamma}, \nabla\cdot \right) - (f + \frac{1}{\alpha}p, \cdot) \\ a(p,\cdot) + g\gamma \left(\frac{\nabla p}{\widetilde{\max}_\gamma}, \nabla\cdot \right) - g\gamma \left(\frac{(\nabla p, \nabla y)_{\mathbb{R}^2}}{\widetilde{\max}_\gamma^2}, \widetilde{\max}'_\gamma(\cdot) \right) + (y - y_d, \cdot) \end{pmatrix}$$

where

$$\widetilde{\max}'_\gamma(\delta_y) := \max'_\gamma(g, \gamma|\nabla y|)(\delta_y) = \begin{cases} \gamma \frac{(\nabla y, \nabla \delta_y)_{\mathbb{R}^2}}{|\nabla y|} & \text{in } \mathcal{A}_\gamma, \\ \gamma^2 \left(\gamma|\nabla y| - g + \frac{1}{2\gamma} \right) \frac{(\nabla y, \nabla \delta_y)_{\mathbb{R}^2}}{|\nabla y|} & \text{in } \mathcal{S}_\gamma, \\ 0 & \text{in } \mathcal{I}_\gamma, \end{cases}$$

each regularized optimality system may be written as:

$$F(y,p) = 0. \tag{16}$$

To apply a Newton type method for the solution of (16) a generalized Jacobian of F must be computed (see e.g. [8] for further details). From (5) a natural candidate for the generalized second derivative of the max_c function is given by

$$\max''_c(0,x) = \begin{cases} c & \text{if } |x| \leq \frac{1}{2c}, \\ 0 & \text{elsewhere.} \end{cases} \tag{17}$$

Taking the vector-valued infinite dimensional counterpart of this candidate, the components of the generalized derivative of F at (y,p), in direction (δ_y, δ_p), are given by

$$\begin{aligned} G_1 F(y,p)(\delta_y,\delta_p) = a(\delta_y,\cdot) + g\gamma \left(\frac{\nabla \delta_y}{\widetilde{\max}_\gamma}, \nabla\cdot \right) \\ - g\gamma \left(\frac{\widetilde{\max}'_\gamma(\delta_y)}{\widetilde{\max}_\gamma^2} \nabla y, \nabla\cdot \right) - \frac{1}{\alpha}(\delta_p, \cdot), \end{aligned} \tag{18}$$

$$\begin{aligned} G_2 F(y,p)(\delta_y,\delta_p) = a(\delta_p,\cdot) + g\gamma \left(\frac{\nabla \delta_p}{\widetilde{\max}_\gamma}, \nabla\cdot \right) - g\gamma \left(\frac{\widetilde{\max}'_\gamma(\delta_y)}{\widetilde{\max}_\gamma^2} \nabla p, \nabla\cdot \right) \\ - g\gamma \left(\frac{(\nabla \delta_p, \nabla y)_{\mathbb{R}^2}}{\widetilde{\max}_\gamma^2} + \frac{(\nabla p, \nabla \delta_y)_{\mathbb{R}^2}}{\widetilde{\max}_\gamma^2} - 2\frac{(\nabla p, \nabla y)_{\mathbb{R}^2}}{\widetilde{\max}_\gamma^3} \widetilde{\max}'_\gamma(\delta_y), \widetilde{\max}'_\gamma(\cdot) \right) \\ - g\gamma \left(\frac{(\nabla p, \nabla y)_{\mathbb{R}^2}}{\widetilde{\max}_\gamma^2}, \widetilde{\max}''_\gamma[\delta_y](\cdot) \right) + (\delta_y, \cdot), \end{aligned} \tag{19}$$

where, for $v \in H_0^1(\Omega)$,

$$\widetilde{\max}_\gamma''[\delta_y](v) = \begin{cases} \gamma \left[\frac{(\nabla y, \nabla v)_{\mathbb{R}^2}}{|\nabla y|} - \frac{(\nabla y, \nabla \delta_y)_{\mathbb{R}^2}}{|\nabla y|^3} (\nabla y, \nabla v)_{\mathbb{R}^2} \right] & \text{in } \mathcal{A}_\gamma \\ \gamma^2 \left(\gamma |\nabla y| - g + \frac{1}{2\gamma} \right) \left[\frac{(\nabla y, \nabla v)_{\mathbb{R}^2}}{|\nabla y|} - \frac{(\nabla y, \nabla \delta_y)_{\mathbb{R}^2}}{|\nabla y|^3} (\nabla y, \nabla v)_{\mathbb{R}^2} \right] & \\ \qquad + \gamma^3 \frac{(\nabla y, \nabla \delta_y)_{\mathbb{R}^2}}{|\nabla y|^2} (\nabla y, \nabla v)_{\mathbb{R}^2} & \text{in } \mathcal{S}_\gamma \\ 0 & \text{in } \mathcal{I}_\gamma. \end{cases}$$

A Newton type algorithm for solving each regularized system can therefore be given as follows:

Algorithm 1

1. Initialize $(y_0, p_0) \in H_0^1(\Omega) \times H_0^1(\Omega)$ and set $k = 0$.
2. Set $\mathcal{A}_k = \{x \in \Omega : \gamma |\nabla y_k(x)| - g \geq \frac{1}{2\gamma}\}$, $\mathcal{S}_k = \{x \in \Omega : |\gamma |\nabla y_k(x)| - g| \leq \frac{1}{2\gamma}\}$ and $\mathcal{I}_k = \Omega \backslash (\mathcal{A}_k \cup \mathcal{S}_k)$.
3. Solve the increment equation
$$GF(y_k, p_k)(\delta_y, \delta_p) = -F(y_k, p_k) \tag{20}$$
and update $y_{k+1} = y_k + \delta_y,\ p_{k+1} = p_k + \delta_p$.
4. Stop or set $k = k + 1$ and goto 2.

3.2 Example

Next we apply the proposed semi-smooth Newton algorithm for the optimal control of a Bingham flow with parameter values $\nu = 1$ and $g = 2$. We consider a homogeneous finite differences scheme, with centered differences for the approximation of the gradient and the divergence operators. For the discrete Laplacian the five point stencil is utilized. The algorithm starts with all variables equal to zero and terminates when the norm of the optimality system is smaller than $tol = 10^{-4}$.

The controlled state for the parameter values $\nu = 1,\ g = 2,\ \gamma = 100,\ \alpha = 0.1,\ h = 1/120,\ f \equiv 10$ and the desired state $z \equiv 1$ is plotted in Figure 1 jointly with the Euclidean norm of the dual variable.

The optimal control for the problem is plotted in Figure 2. Since in this case $\alpha \bar{u} = p$, the satisfaction of (14g) can be inferred from the plot.

The convergence of the algorithm is registered in Table 1. With $\varrho_k := \|F(y_k, p_k)\|$ and $\sigma_k := \frac{\|F(y_k, p_k)\|}{\|F(y_{k-1}, p_{k-1})\|}$ as indicators for the residuum and the convergence rate, local superlinear behavior of the algorithm can be experimentally verified.

References

1. V. Barbu (1993). *Analysis and Control of nonlinear infinite dimensional systems.* Academic Press, New York.

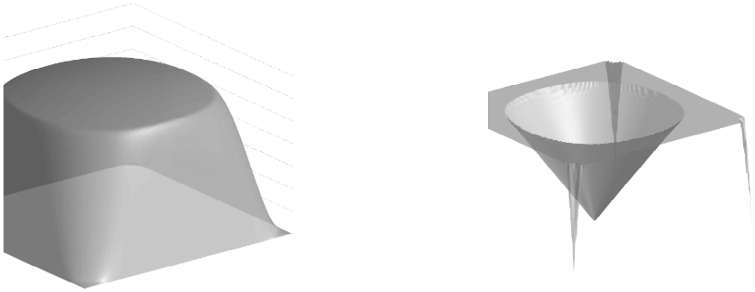


Fig. 1. Controlled state and Euclidean norm of the controlled dual variable: $\nu = 1, g = 2, \gamma = 100, \alpha = 0.1, h = 1/120$

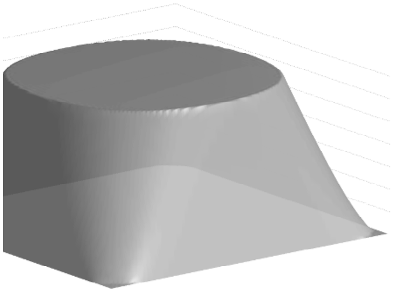


Fig. 2. Optimal control: $\nu = 1, g = 2, \gamma = 100, \alpha = 0.1, h = 1/120$

2. M. Bergounioux (1998). Optimal control of problems governed by abstract elliptic variational inequalities with state constraints. *SIAM Journal on Control and Optimization*, 36(1):273–289.
3. E. Casas and L. A. Fernández (1993). Distributed control of systems governed by a general class of quasilinear elliptic equations. *J. Differential Equations*, 104(1):20–47.
4. J. C. De Los Reyes and S. González (2009). Path following methods for steady laminar Bingham flow in cylindrical pipes. *ESAIM M2AN*, 43:81–117.
5. J. C. De Los Reyes (2009). Optimal control of a class of variational inequalities of the second kind. *Preprint 15-2009*, TU Berlin.
6. E. J. Dean, R. Glowinski, and G. Guidoboni (2007). On the numerical simulation of Bingham viscoplastic flow: old and new results. *J. Non-Newton. Fluid Mech.*, 142(1–3):36–62.
7. G. Duvaut and J.-L. Lions (1976). *Inequalities in mechanics and physics.* Springer-Verlag, Berlin.
8. M. Hintermüller, K. Ito, and K. Kunisch (2003). The primal-dual active set strategy as a semismooth Newton method. *SIAM J. Optim.*, 13(3):865–888.
9. M. Hintermüller and G. Stadler (2007). An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM J. Sci. Comput.*, 28(1):1–23.

Iteration	$\mid \mathcal{I}_k \mid$	ϱ_k	σ_k
1	14400	1177.9	0
2	9336	529.02	0.44914
3	8140	3528.4	6.6697
4	7088	1897.6	0.53779
5	6972	1494.7	0.78768
6	6832	871.63	0.58316
7	6796	1919.6	2.2023
8	6736	521.29	0.27156
9	6736	496.21	0.95188
10	6664	347.45	0.70021
11	6656	251.69	0.7244
12	6656	89.094	0.35399
13	6656	30.466	0.34196
14	6656	4.5033	0.14781
15	6656	0.27634	0.061364
16	6656	0.015882	0.057474
17	6656	0.0009369	0.05899
18	6656	5.5419e-5	0.059151

Table 1. $g = 2$, $\nu = 1$, $\gamma = 100$, $f = 10$ and $h = 1/120$.

10. P.P. Mosolov and V.P. Miasnikov (1965). Variational methods in the theory of the fluidity of a viscous plastic medium. *Journal of Applied Mathematics and Mechanics (PMM)*, 29:468–492.
11. G. Stadler (2004). Semismooth newton and augmented Lagrangian methods for a simplified friction problem. *SIAM Journal on Optimization*, 15(1):39–62.

Semismooth Newton Methods for an Optimal Boundary Control Problem of Wave Equations

Axel Kröner[1], Karl Kunisch[2], and Boris Vexler[3]

[1] Lehrstuhl für Mathematische Optimierung, Technische Universität München, Fakultät für Mathematik, Boltzmannstraße 3, 85748 Garching b. München, Germany kroener@ma.tum.de
[2] University of Graz, Institute for Mathematics and Scientific Computing, Heinrichstraße 36, A-8010 Graz, Austria karl.kunisch@uni-graz.at
[3] Lehrstuhl für Mathematische Optimierung, Technische Universität München, Fakultät für Mathematik, Boltzmannstraße 3, 85748 Garching b. München, Germany vexler@ma.tum.de

Summary. In this paper optimal Dirichlet boundary control problems governed by the wave equation and the strongly damped wave equation with control constraints are analyzed. For treating inequality constraints semismooth Newton methods are discussed and their convergence properties are investigated. For numerical realization a space-time finite element discretization is introduced. Numerical examples illustrate the results.

1 Introduction

In this paper we consider primal-dual active set methods (PDAS) applied to optimal Dirichlet boundary control problems governed by the wave equation and the strongly damped wave equation subject to pointwise control constraints. We interprete the PDAS-methods as semismooth Newton methods and analyze them with respect to superlinear convergence, cf. [10, 13, 27, 28, 17].

Let $\Omega \subset \mathrm{R}^n$, $n \geq 1$, be a bounded domain which has either a C^2-boundary or is polygonal and convex. For $T > 0$ we denote $I = (0, T)$, $Q = I \times \Omega$ and $\Sigma = I \times \partial\Omega$. Here and in what follows, we employ the usual notion of Lebesgue and Sobolev spaces.

Then the optimal control problem under consideration is formulated as follows:

$$\begin{cases} \text{Minimize} & J(y,u) = \mathcal{G}(y) + \frac{\alpha}{2}\|u\|^2_{L^2(\Sigma)}, \\ \text{subject to} & y = S(u), \\ & y \in L^2(Q),\ u \in U_{\text{ad}}, \end{cases} \tag{1}$$

for $\alpha > 0$ and where $S\colon L^2(\Sigma) \to L^2(Q)$ is given as the control-to-state operator of the following equation with $0 \leq \rho \leq \rho_0$, $\rho_0 \in \mathrm{R}^+$:

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_33, © Springer-Verlag Berlin Heidelberg 2010

$$\begin{cases} y_{tt} - \Delta y - \rho \Delta y_t = f & \text{in } Q, \\ y(0) = y_0 & \text{in } \Omega, \\ y_t(0) = y_1 & \text{in } \Omega, \\ y = u & \text{on } \Sigma. \end{cases} \tag{2}$$

The functional $\mathcal{G}\colon L^2(Q) \to \mathrm{R}$ is assumed to be quadratic with $\mathcal{G}'$ being an affine operator from $L^2(Q)$ to itself, and $\mathcal{G}''$ is assumed to be non-negative. The set of admissible controls U_{ad} is given by bilateral box constraints

$$U_{\text{ad}} = \{u \in L^2(\Sigma) | u_a \le u \le u_b\} \quad \text{with} \;\; u_a, u_b \in L^2(\Sigma).$$

If we set $\rho = 0$ in (2) we obtain the usual wave equation. For $\rho > 0$ we get the strongly damped wave equation which often appears in models with loss of energy, e.g., it arises in the modelling of longitudinal vibrations in a homogeneous bar, in which there are viscous effects, cf. [22]. The corresponding optimal control problem (with small $\rho > 0$) can also be regarded as regularization of the Dirichlet boundary control problem for the wave equation.

Optimal control problems governed by wave equations are considered in several publications, see [20, 21, 24, 25, 18, 8, 19, 9]. A survey about finite difference approximations in the context of control of the wave equation is presented in [29].

In this paper we summarize the results from [16] for the case of optimal Dirichlet boundary control. We analyze semismooth Newton methods applied to (1) with respect to superlinear convergence. Here, an important ingredient in proving superlinear convergence is a smoothing property of the operator mapping the control variable u to the trace of the normal derivative of the adjoint state p. For $\rho > 0$ we verify, that such a smoothing property is given. For $\rho = 0$ we will provide an example illustrating the fact that such a property can not hold in general. This is different to optimal distributed and Neumann boundary control of the wave equation, see [16], where this property is given. For the numerical realization of the arising infinite dimensional optimal control problems we use space-time finite element methods following [4, 23, 17].

The paper is organized as follows. In the next section we discuss the semismooth Newton method for an abstract optimal control problem. Section 3 is devoted to relevant existence, uniqueness and regularity results for the state equation. In Section 4 we check the assumptions for superlinear convergence of the semismooth Newton method. In Section 5 we describe the space-time finite element discretization and in Section 6 we present numerical examples illustrating our results.

2 Semismooth Newton methods and the primal-dual active set strategy

In this section we summarize known results for semismooth Newton methods, which are relevant for the analysis in this paper.

Let X and Z be Banach spaces and let $F\colon D \subset X \to Z$ be a nonlinear mapping with open domain D. Moreover, let $\mathcal{L}(X,Z)$ be the set of continuous, linear mappings from X to Z.

Definition 1. *The mapping $F\colon D \subset X \to Z$ is called Newton-differentiable in the open subset $U \subset D$ if there exists a family of generalized derivatives $G\colon U \to \mathcal{L}(X,Z)$ such that*

$$\lim_{h\to 0} \frac{1}{\|h\|_X} \|F(x+h) - F(x) - G(x+h)h\|_Z = 0,$$

for every $x \in U$.

Using this definition there holds the following proposition, see [10].

Proposition 1. *The mapping* $\max(0,\cdot)\colon L^q(\varSigma) \to L^p(\varSigma)$ *with* $1 \le p < q < \infty$ *is Newton-differentiable on $L^q(\varSigma)$.*

The following theorem provides a generic result on superlinear convergence for semismooth Newton methods, see [10].

Theorem 1. *Suppose, that $x^* \in D$ is a solution to $F(x) = 0$ and that F is Newton–differentiable with Newton-derivative G in an open neighborhood U containing x^* and that*

$$\{\|G(x)^{-1}\|_{\mathcal{L}(X,Z)} | x \in U\}$$

is bounded. Then for $x_0 \in D$ the Newton–iteration

$$x_{k+1} = x_k - G(x_k)^{-1}F(x_k), \quad k = 0,1,2,\ldots,$$

converges superlinearly to x^* provided that $\|x_0 - x^*\|_X$ is sufficiently small.

In the following we consider the linear quadratic optimal control problem (1). The operator S is affine-linear, thus it can be characterized in the following way

$$S(u) = Tu + \bar{y}, \quad T \in \mathcal{L}(L^2(\varSigma), L^2(Q)), \quad \bar{y} \in L^2(Q).$$

From standard subsequential limit arguments, see, e. g., [20], follows:

Proposition 2. *There exists a unique global solution of the optimal control problem under consideration.*

We define the reduced cost functional

$$j\colon U \to \mathrm{R}, \quad j(u) = \mathcal{G}(S(u)) + \frac{\alpha}{2}\|u\|^2_{L^2(\varSigma)}$$

and reformulate the optimal control problem under consideration as

$$\text{Minimize } \ j(u), \quad u \in U_{\text{ad}}.$$

The first (directional) derivative of j is given as

$$j'(u)(\delta u) = (\alpha u - q(u), \delta u)_{L^2(\Sigma)},$$

where the operator $q\colon L^2(\Sigma) \to L^2(\Sigma)$ is given by

$$q(u) = -T^* \mathcal{G}'(S(u)). \tag{3}$$

A short calculation proves the next proposition, cf. [12].

Proposition 3. *The necessary optimality condition for* (1) *can be formulated as*

$$\mathcal{F}(u) = 0, \tag{4}$$

with the operator $\mathcal{F}\colon L^2(\Sigma) \to L^2(\Sigma)$ *defined by*

$$\mathcal{F}(u) = \alpha(u - u_b) + \max(0,\, \alpha u_b - q(u)) + \min(0, q(u) - \alpha u_a).$$

The following assumption will insure the superlinear convergence of the semismooth Newton method applied to (4).

Assumption 1. *We assume, that the operator q defined in* (3) *is a continuous affine-linear operator* $q\colon L^2(\Sigma) \to L^r(\Sigma)$ *for some* $r > 2$.

In Section 4 we will check this assumption for the optimal control problem under consideration.

Lemma 1. *Let Assumption 1 be fulfilled and* $u_a, u_b \in L^r(\Sigma)$ *for some* $r > 2$. *Then the operator* $\mathcal{F}\colon L^2(\Sigma) \to L^2(\Sigma)$ *is Newton-differentiable and a generalized derivative* $G_{\mathcal{F}}(u) \in \mathcal{L}(L^2(\Sigma), L^2(\Sigma))$ *exists. Moreover,*

$$\|G_{\mathcal{F}}(u)^{-1}(w)\|_{L^2(\Sigma)} \leq C_G \, \|w\|_{L^2(\Sigma)} \quad \textit{for all}\, w \in L^2(\Sigma)$$

for a constant C_G *and each* $u \in L^2(\Sigma)$.

For a proof see [16].

After these considerations we can formulate the following theorem.

Theorem 2. *Let Assumption 1 be fulfilled and suppose that* $u^* \in L^2(\Sigma)$ *is the solution to* (1). *Then, for* $u_0 \in L^2(\Sigma)$ *with* $\|u_0 - u^*\|_{L^2(\Sigma)}$ *sufficiently small, the semismooth Newton method*

$$G_{\mathcal{F}}(u_k)(u_{k+1} - u_k) + \mathcal{F}(u_k) = 0, \quad k = 0, 1, 2, \ldots,$$

converges superlinearly.

Proof. This follows from Theorem 1 and Lemma 1.

Remark 1. This semismooth Newton method is known to be equivalent to a primal-dual active set strategy (PDAS), cf. [10, 13] which we apply for our numerical examples.

3 On the state equation

In this section we summarize some existence and regularity results for equation (2), cf. [16]. Here and in what follows, we use the following notations $(\cdot,\cdot)$, $\langle\cdot,\cdot\rangle$, $(\cdot,\cdot)_I$ and $\langle\cdot,\cdot\rangle_I$ for the inner products in the spaces $L^2(\Omega)$, $L^2(\partial\Omega)$, $L^2(L^2(\Omega))$ and $L^2(L^2(\Sigma))$, respectively.

Theorem 3. *Let $\rho = 0$, $u|_\Sigma = 0$ and $(f, y_0, y_1) \in L^2(L^2(\Omega)) \times H_0^1(\Omega) \times L^2(\Omega)$. Then equation (2) admits a unique solution $(y, y_t) \in C(H_0^1(\Omega)) \times C(L^2(\Omega))$ depending continuously on the data (f, y_0, y_1).*

Theorem 4. *Let $\rho = 0$, $(f,\, y_0,\, y_1,\, u) \in L^1((H_0^1(\Omega))^*) \times L^2(\Omega) \times (H_0^1(\Omega))^* \times L^2(\Sigma)$. Then equation (2) admits a unique solution $(y, y_t) \in C(L^2(\Omega)) \times C(H^{-1}(\Omega))$ depending continuously on the data (f, y_0, y_1, u). It satisfies*

$$(y, \zeta_{tt} - \Delta\zeta)_I = (f, \zeta)_I - (y_0, \zeta_t(0)) + \langle y_1, \zeta(0)\rangle_{(H^1(\Omega))^*, H^1(\Omega)} - \langle u, \partial_n\zeta\rangle_I$$

where ζ is the solution to

$$\begin{cases} \zeta_{tt} - \Delta\zeta = g, \\ \zeta(T) = 0, \quad \zeta_t(T) = 0, \quad \zeta|_\Sigma = 0 \end{cases}$$

for any $g \in L^1(L^2(\Omega))$.

Theorem 5. *Let $\rho > 0$, $u|_\Sigma = 0$ and $(f, y_0, y_1) \in L^2(L^2(\Omega)) \times H_0^1(\Omega) \cap H^2(\Omega) \times H_0^1(\Omega)$. Then equation (2) admits a unique solution*

$$y \in D = H^2(L^2(\Omega)) \cap C^1(H_0^1(\Omega)) \cap H^1(H^2(\Omega))$$

defined by the conditions: $y(0) = y_0$, $y_t(0) = y_1$ and

$$(y_{tt}(s), \phi) + (\nabla y(s), \nabla\phi) + \rho(\nabla y_t(s), \nabla\phi) = (f(s), \phi) \qquad \text{for all } \phi \in H_0^1(\Omega) \text{ a.e. in } (0, T).$$

Moreover, the a priori estimate

$$\|y\|_D \le C\left(\|f\|_{L^2(L^2(\Omega))} + \|\nabla y_0\|_{L^2(\Omega)} + \|\Delta y_0\|_{L^2(\Omega)} + \|\nabla y_1\|_{L^2(\Omega)}\right),$$

holds, where the constant $C = C(\rho)$ tends to infinity as ρ tends to zero.

Theorem 6. *Let $\rho > 0$ and $(f, y_0, y_1, u) \in L^2(L^2(\Omega)) \times H^1(\Omega) \times L^2(\Omega) \times L^2(\Sigma)$. Then equation (2) admits a unique very weak solution $y \in L^2(L^2(\Omega))$ defined by*

$$\begin{aligned}(v, y)_I = -(y_0, \zeta_t(0)) + (y_1, \zeta(0)) - \langle u, \partial_n\zeta\rangle_I + \rho\langle u, \partial_n\zeta_t\rangle_I - \rho(y_0, \Delta\zeta(0)) \\ + \rho\langle y_0, \partial_n\zeta(0)\rangle + (f, \zeta)_I \quad \text{for all } v \in L^2(L^2(\Omega)),\end{aligned}$$

where ζ is the solution of

$$\begin{cases} \zeta_{tt} - \Delta\zeta + \rho\Delta\zeta_t = v, \\ \zeta(T) = 0, \quad \zeta_t(T) = 0, \quad \zeta|_\Sigma = 0. \end{cases}$$

Furthermore, the following estimate

$$\|y\|_{L^2(L^2(\Omega))} \leq C\left(\|u\|_{L^2(\Sigma)} + \|f\|_{L^2(L^2(\Omega))} + \|y_0\|_{H^1(\Omega)} + \|y_1\|_{L^2(\Omega)}\right),$$

holds, where the constant $C = C(\rho)$ tends to infinity as ρ tends to zero.

4 Optimal control problem

In this section we check Assumption 1 for the control problem under consideration. Let $y_0 \in H_0^1(\Omega)$, $y_1 \in L^2(\Omega)$ and $f \in L^2(L^2(\Omega))$. Then we have the following optimality system

$$\begin{cases} y_{tt} - \Delta y - \rho\Delta y_t = f, \\ y(0) = y_0, \quad y_t(0) = y_1, \quad y|_\Sigma = u, \\ p_{tt} - \Delta p + \rho\Delta y_t = -\mathcal{G}'(y), \\ p(T) = 0, \quad p_t(T) = 0, \quad p|_\Sigma = 0, \\ \alpha u + \lambda = -\partial_n p|_\Sigma, \\ \lambda = \max(0, \lambda + c(u - u_b)) + \min(0, \lambda + c(u - u_a)) \end{cases}$$

for $c > 0$, $\lambda \in L^2(\Sigma)$ and the solution p of the adjoint equation.

The operator q defined in (3) turns out to be a continuous affine-linear operator $q\colon L^2(\Sigma) \to L^2(\Sigma)$ with $q(u) = -\partial_n p$.

However, Assumption 1 is not fulfilled for $\rho = 0$, see Example 1.

Example 1. We consider an one dimensional wave equation with Dirichlet boundary control

$$\begin{aligned} &y_{tt} - y_{xx} = 0 && \text{in } (0,1) \times (0,1), \\ &y(t,0) = u(t), \quad y(t,1) = 0 && \text{in } (0,1), \\ &y(0,x) = 0, \quad y_t(0,x) = 0 && \text{in } (0,1) \end{aligned}$$

with $u \in L^2(0,1)$. Here, for a general control $u \in L^2(0,1)$ it turns out that $q(u)(t) = -16(1-t)u(t)$ for $t \in (0,1)$, and therefore the image $q(u)$ does not have an improved regularity $q(u) \in L^r(0,1)$ for $r > 2$, see [16]. This lack of additional regularity is due to the nature of the wave equation. In the elliptic as well as in the parabolic cases the corresponding operator q possess the required regularity for Dirichlet boundary control, see [17].

For $\rho > 0$ Assumption 1 is true:

Theorem 7. *For $\rho > 0$, the operator q defined in* (3) *satisfies $q\colon L^2(\Sigma) \to L^r(\Sigma)$ with some $r > 2$.*

For a proof we refer to [16]. Therein we apply Theorem 5 to derive an improved regularity of $\partial_n p$.

5 Discretization

In this section we present a short overview about the discretization of the optimal control problem under consideration, for details we refer to [16]. Finite element discretizations of the wave equations are analyzed, e.g., in [1, 2, 3, 6, 11, 14, 15]. Here, we apply a cG(1)cG(1) discretization, which is known to be energy conserving.

For a precise definition of our discretization we consider a partition of the time interval $\bar{I} = [0, T]$ as $\bar{I} = \{0\} \cup I_1 \cup \dots \cup I_M$ with subintervals $I_m = (t_{m-1}, t_m]$ of size k_m and time points $0 = t_0 < t_1 < \dots < t_{M-1} < t_M = T$.

For spatial discretization we will consider two- or three-dimensional shape regular meshes $\mathcal{T}_h = \{K\}$, for details see [5].

Let $V = H^1(\Omega)$ and $V^0 = H_0^1(\Omega)$. On the mesh $\mathcal{T}_h$ we construct conforming finite element spaces $V_h \subset V$ and $V_h^0 \subset V^0$ in a standard way:

$$V_h = \{v \in V | v|_K \in \mathcal{Q}^1(K) \text{ for } K \in \mathcal{T}_h\},$$
$$V_h^0 = \{v \in V^0 | v|_K \in \mathcal{Q}^1(K) \text{ for } K \in \mathcal{T}_h\},$$

where $\mathcal{Q}^1(K)$ is a space of bi- or trilinear shape functions on the cell K.

We define the following space-time finite element spaces:

$$\begin{aligned}
X_{kh} &= \{v_{kh} \in C(\bar{I}, V_h) | v_{kh}|_{I_m} \in \mathcal{P}^1(I_m, V_h)\},\\
X_{kh}^0 &= \{v_{kh} \in C(\bar{I}, V_h^0) | v_{kh}|_{I_m} \in \mathcal{P}^1(I_m, V_h^0)\},\\
\widetilde{X}_{kh} &= \{v_{kh} \in L^2(I, V_h) | v_{kh}|_{I_m} \in \mathcal{P}^0(I_m, V_h) \text{ and } v_{kh}(0) \in V_h\},\\
\widetilde{X}_{kh}^0 &= \{v_{kh} \in L^2(I, V_h^0) | v_{kh}|_{I_m} \in \mathcal{P}^0(I_m, V_h^0) \text{ and } v_{kh}(0) \in V_h\},
\end{aligned}$$

where $\mathcal{P}^r(I_m, V_h)$ denotes the space of polynomials up to degree r on I_m with values in V_h.

For the definition of the discrete control space, we introduce the space of traces of functions in V_h:

$$W_h = \{w_h \in H^{\frac{1}{2}}(\partial\Omega) | w_h = \gamma(v_h), v_h \in V_h\},$$

where $\gamma \colon H^1(\Omega) \to H^{\frac{1}{2}}(\partial\Omega)$ denotes the trace operator. Thus, we can define

$$U_{kh} = \{v_{kh} \in C(\bar{I}, W_h) | v_{kh}|_{I_m} \in \mathcal{P}^1(I_m, W_h)\}.$$

For a function $u_{kh} \in U_{kh}$ we define an extension $\hat{u}_{kh} \in X_{kh}$ such that

$$\gamma(\hat{u}_{kh}(t, \cdot)) = u_{kh}(t, \cdot) \text{ and } \hat{u}_{kh}(t, x_i) = 0$$

on all interior nodes x_i of $\mathcal{T}_h$ and for all $t \in \bar{I}$.

Then the discrete optimization problem is formulated as follows:

$$\text{Minimize } J(y_{kh}^1, u_{kh})$$

for $u_{kh} \in U_{kh} \cap U_{\mathrm{ad}}$ and $y_{kh} = (y^1_{kh}, y^2_{kh}) \in (\hat{u}_{kh} + X^0_{kh}) \times X_{kh}$ subject to

$$a_\rho(y_{kh}, \xi_{kh}) = (f, \xi^1_{kh})_I + (y_1, \xi^1_{kh}(0)) - (y_0, \xi^2_{kh}(0))$$
$$\text{for all } \xi_{kh} = (\xi^1_{kh}, \xi^2_{kh}) \in \widetilde{X}^0_{kh} \times \widetilde{X}_{kh}, \quad (5)$$

where the bilinear form $a_\rho\colon X_{kh} \times X_{kh} \times \widetilde{X}_{kh} \times \widetilde{X}_{kh} \to \mathrm{R}$ is defined by

$$\begin{aligned} a_\rho(y, \xi) = a_\rho(y^1, y^2, \xi^1, \xi^2) &= (\partial_t y^2, \xi^1)_I + (\nabla y^1, \nabla \xi^1)_I + \rho(\nabla y^2, \nabla \xi^1)_I \\ &+ (\partial_t y^1, \xi^2)_I - (y^2, \xi^2)_I + (y^2(0), \xi^1(0)) - (y^1(0), \xi^2(0)), \end{aligned}$$

with $y = (y^1, y^2)$ and $\xi = (\xi^1, \xi^2)$ with a real parameter $\rho \geq 0$.

Remark 2. We approximate the time integrals in equation (5) piecewise by the trapezoidal rule, thus the time discretization results in a Crank-Nicolson scheme.

As on the continuous level equation (5) defines the corresponding discrete solution operator S_{kh} mapping a given control u_{kh} to the first component of the state y^1_{kh}. We introduce the discrete reduced cost functional

$$j_{kh}(u_{kh}) = J(S_{kh}(u_{kh}), u_{kh})$$

and reformulate the discrete optimization problem as

$$\text{Minimize } \; j_{kh}(u_{kh}) \quad \text{for } \; u_{kh} \in U_{kh} \cap U_{\mathrm{ad}}.$$

This optimization problem is solved using the semismooth Newton method (primal-dual active set method) as described in Section 2 for the continuous problem, see [16].

6 Numerical examples

In this section we present a numerical example illustrating our theoretical results for the optimal control problem under consideration. All computations are done using the optimization library RoDoBo [26] and the finite element toolkit Gascoigne [7].

We specify the functional $\mathcal{G}$ in the following way: For a given function $y_d \in L^2(L^2(\Omega))$ we define $\mathcal{G}(y) = \frac{1}{2}\|y - y_{\mathrm{d}}\|^2_{L^2(Q)}$.

Then we consider the control problem for the following data:

$$f(t,x) = \begin{cases} 1, & x_1 > 0.5, \\ x_1, & \text{else} \end{cases}, \quad u_a = -0.18, \quad u_b = 0.2, \quad T = 1,$$

$$y_{\mathrm{d}}(t,x) = \begin{cases} x_1 & x_1 > 0.5 \\ -x_1 & \text{else} \end{cases}, \quad y_0(x) = \sin(\pi x_1)\sin(\pi x_2), \quad y_1(x) = 0$$

Table 1. Numbers of PDAS-iterations on the sequence of uniformly refined meshes for different parameters α and ρ

			$\alpha = 10^{-4}$			$\alpha = 10^{-2}$		
Level	N	M	$\rho = 0$	$\rho = 0.1$	$\rho = 0.7$	$\rho = 0$	$\rho = 0.1$	$\rho = 0.7$
1	16	2	4	3	5	4	4	5
2	64	4	5	4	3	4	4	3
3	256	8	5	5	4	5	4	4
4	1024	16	6	6	6	5	7	5
5	4096	32	11	7	7	9	6	5
6	16384	64	13	9	7	10	8	5

for $t \in [0, T]$ and $x = (x_1, x_2) \in \Omega = (0, 1)^2$.

Table 1 illustrates the effect of damping introduced by the term $-\rho\Delta y_t$ on the number of PDAS steps. For $\alpha = 0.01$ and $\rho = 0$ we observe a mesh-dependence of the algorithm. Moreover, the number of PDAS steps declines for increasing value of ρ and stays nearly mesh independent for $\rho > 0$. Furthermore, we consider the effect of α on the number of PDAS steps. As expected the number of iterations declines also for increasing α.

Further numerical examples indicate that on a given mesh we have superlinear convergence only for $\rho > 0$, see [16].

Acknowledgment. The authors would like to thank Dominik Meidner for helpful discussions about the implementation.

References

1. Bales L, Lasiecka I (1994) Continuous finite elements in space and time for the nonhomogeneous wave equation. *Computers Math. Applic.*, 27(3):91–102.
2. Bales L, Lasiecka I (1995) Negative norm estimates for fully discrete finite element approximations to the wave equation with nonhomogeneous L_2 Dirichlet boundary data. *Math. Comp.*, 64(209):89–115.
3. Bangerth W, Rannacher R (2001) Adaptive finite element techniques for the acoustic wave equation. *J. Comput. Acoustics*, 9(2):575–591.
4. Becker R, Meidner D, Vexler B (2007) Efficient numerical solution of parabolic optimization problems by finite element methods. *Optim. Methods Softw.*, 22(5):813–833.
5. Braess D (2007) *Finite Elements: Theory, Fast Solvers and Applications in Solid Mechanics.* Cambridge, Cambridge University Press.
6. French D A, Peterson T E (1996) Continuous space-time finite elements method for the wave equation. *Math. Comp.*, 65(214):491–506.
7. Gascoigne: The finite element toolkit. `http://www.gascoigne.uni-hd.de`.
8. Gerdts M, Greif G, Pesch H J (2008) Numerical optimal control of the wave equation: optimal boundary control of a string to rest in finite time. *Math. Comput. Simulation*, 79(4):1020–1032.

9. Gugat M, Keimer A, Leugering G (2009) Optimal distributed control of the wave equation subject to state constraints. *ZAMM Z. Angew. Math. Mech.*, 89(6):420–444.
10. Hintermüller M, Ito K, Kunisch K (2003) The primal-dual active set strategy as a semismooth Newton method. *SIAM J. Optim.*, 13(3):865–888.
11. Hulbert G M, Hughes T J R (1990) Space-time finite element methods for second-order hyperbolic equations. *Comput. Methods Appl. Mech. Engrg.*, 84:327–348.
12. Ito K, Kunisch K (2008) *Lagrange Multiplier Approach to Variational Problems and Applications.* Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.
13. Ito K, Kunisch K (2003) Semi-smooth Newton methods for state-constrained optimal control problems. *Systems and Control Lett.*, 50:221–228.
14. Johnson C (1993) Discontinuous Galerkin finite element methods for second order hyperbolic problems. *Comput. Methods Appl. Mech. Engrg.*, 107:117–129.
15. Karakashian O, Makridakis C (2004) Convergence of a continuous Galerkin method with mesh modification for nonlinear wave equation. *Math. Comp.*, 47(249):85–102.
16. Kröner A, Kunisch K, Vexler B (2009) Semismooth Newton methods for optimal control of the wave equation with control constraints, *submitted.*
17. Kunisch K, Vexler B (2007) Constrained Dirichlet boundary control in L^2 for a class of evolution equations. *SIAM J. Control Optim.*, 46(5):1726–1753.
18. Lagnese J E, Leugering G (2000) Dynamic domain decomposition in approximate and exact boundary control in problems of transmission for wave equations. *SIAM J. Control Optim.*, 38(2):503–537.
19. Lasiecka I, Triggiani R (2000) *Control Theory for Partial Differential Equations: Continuous and Approximation Theories, Vol. 1 and Vol. 2.* Encyclopedia of mathematics and its applications. Cambridge University Press, Philadelphia.
20. Lions J L (1971) *Optimal Control of Systems Governed by Partial Differential Equations*, volume 170 of *Grundlehren Math. Wiss.* Springer-Verlag, Berlin.
21. Lions J L (1985) *Control of distributed singular systems.* Gauthier-Villars, Kent.
22. Massatt P (1983) Limiting behavior for strongly damped nonlinear wave equations. *J. Differential Equations*, 48:334–349.
23. Meidner D, Vexler B (2007) Adaptive space-time finite element methods for parabolic optimization problems. *SIAM J. Control Optim.*, 46(1):116–142.
24. Mordukhovich B S, Raymond J P (2004) Dirichlet boundary control of hyperbolic equations in the presence of state constraints. *Appl. Math. Optim.*, 49:145–157.
25. Mordukhovich B S, Raymond J P (2005). Neumann boundary control of hyperbolic equations with pointwise state constraints. *SIAM J. Control Optim.*, 43(4):1354–1372.
26. RoDoBo: A C++ library for optimization with stationary and nonstationary PDEs with interface to [7]. `http://www.rodobo.uni-hd.de`.
27. Ulbrich M (2002) Semismooth Newton methods for operator equations in function spaces. *SIAM J. Control Optim.*, 13(3):805–842.
28. Ulbrich M (2003) Constrained optimal control of Navier-Stokes flow by semismooth Newton methods. *Sys. Control Lett.*, 48:297–311.
29. Zuazua E (2005) Propagation, observation, and control of waves approximated by finite difference methods. *SIAM Rev.*, 47(2):197–243.

A Space Mapping Approach for the p-Laplace Equation

Oliver Lass[1] and Stefan Volkwein[2]

[1] Fachbereich Mathematik und Statistik, Universität Konstanz, Universitätsstraße 10, D-78457 Konstanz, Germany. oliver.lass@uni-konstanz.de

[2] Fachbereich Mathematik und Statistik, Universität Konstanz, Universitätsstraße 10, D-78457 Konstanz, Germany. stefan.volkwein@uni-konstanz.de

Summary. Motivated by car safety applications the goal is to deternmine a thickness coefficient in the nonlinear p-Laplace equation. The associated optimal problem is hard to solve numerically. Therefore, the computationally expensive, nonlinear p-Laplace equation is replaced by a simpler, linear model. The space mapping technique is utilized to link the linear and nonlinear equations and drives the optimization iteration of the time intensive nonlinear equation using the fast linear equation. For this reason an efficient realization of the space mapping is utilized. Numerical examples are presented to illustrate the advantage of the proposed approach.

1 Introduction

A main aspect in the design of passenger cars with respect to pedestrian safety is the energy absorption capability of the car parts. Besides that, the car parts have to fulfill several other requirements. The associated optimal problem is hard to solve numerically. That makes it necessary to develop easy and fast to solve prediction models with little loss in accuracy for optimization purpose. Current simulation tools combined with standard optimization software are not well suited to deal with the above mentioned needs [13].

We will show the application of mathematical methods on a simplified model to reduce the optimization effort. The goal of the structural optimization problem (see [7, 8]) is to determine a thickness parameter λ of a plate $\Omega \subset \mathbb{R}^2$ (representing a part of the vehicle) and an associated displacement u satisfying the nonlinear p-Laplace equation

$$-\operatorname{div}\left(2(1+n)\lambda(\mathbf{x})\,|\nabla u(\mathbf{x})|_2^{2n}\,\nabla u(\mathbf{x})\right) = g(\mathbf{x}) \quad \text{for all } \mathbf{x} \in \Omega \tag{1}$$

together with Dirichlet boundary conditions, where g represents a force acting on Ω, $n \in (0,1)$ is the Hollomon coefficient, and $|\cdot|_2$ stands for the Euclidean norm. We suppose that $0 < \lambda_a \leq \lambda(\mathbf{x}) \leq \lambda_b$ with positive scalars λ_a and λ_b. Our goal is to minimize the mass of the plate, i.e., to minimize the integral

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*, DOI 10.1007/978-3-642-12598-0_34, © Springer-Verlag Berlin Heidelberg 2010

$$J_1(\lambda) = \int_\Omega \lambda(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

but also to avoid that the displacement is larger than a given threshold $u_b > 0$. This issue is motivated by our pedestrian safety application. Thus we choose

$$J_2(u) = \beta \int_\Omega \min(u(\mathbf{x}) \;-\; u_b(\mathbf{x}), 0)^3\,\mathrm{d}\mathbf{x}$$

as the second part of our cost functional. Here $\beta > 0$ is a weighting parameter. Due to the nonlinear structure of the elliptic partial differential equation, the numerical solution of the optimization problem governed by the partial differential equation (PDE) constraint (1) is expensive, we consider an alternative constraint given by

$$-\mathrm{div}\,\big(2(1+n)\mu(\mathbf{x})\,\nabla v(\mathbf{x})\big) = g(\mathbf{x}) \quad \text{for all } \mathbf{x} \in \Omega, \tag{2}$$

which is a linear elliptic PDE. We will call (1) the fine model and (2) the coarse model. It turns out that the space mapping technique [9] provides an attractive framework to improve the use of the coarse model as a surrogate for the optimization of the fine model. The space mapping technique is utilized to link the linear and nonlinear equations and drives the optimization iteration of the time intensive nonlinear equation using the fast linear equation. For this reason an efficient realization of the space mapping is utilized.

The space mapping technique was first introduced in [2]. The idea of the space mapping has been developed along different directions and generalized to a number of contexts [14]. One of the problems lies in the information necessary to compute the Jacobian of the space mapping which involves expensive gradient information of (1). In [1] Broyden's method is utilized to construct an approximation of the Jacobian. This approach will be presented. In the context of PDEs, we refer to [6, 10]. Compared to [1, 2, 14], our modified approach is similar to [6], where a modified Broyden formula is used.

The paper is organized in the following manner. In Section 2 we introduce the infinite-dimensional optimization problem for the p-Laplace equation. The space mapping approach is described in Section 3, whereas in Section 4 the surrogate optimization problem is formulated. Section 5 is devoted to present numerical examples illustrating the advantage of the proposed approach.

2 Optimization of the complex model

In this section we formulate optimal control problem governed by the p-Laplace equation. By $W_0^{1,p}(\Omega)$, $p \in [1, \infty)$, we denote the Sobolev space of weakly differentiable functions, whose weak derivative belongs to $L^p(\Omega)$ and whose function values are zero on the boundary $\Gamma = \partial\Omega$. We set $p = 2n + 2$ for $n \in (0, 1)$. Let us define the Banach space $X = L^\infty(\Omega) \times W_0^{1,p}$ and the nonlinear operator $f : X \to W_0^{1,p}(\Omega)'$ (*fine model*) as

$$\langle f(x),\varphi\rangle_{(W_0^{1,p})',W_0^{1,p}} = \int_\Omega 2(1+n)\lambda(\mathbf{x})|\nabla u(\mathbf{x})|_2^{p-2}\nabla u(\mathbf{x})\cdot\nabla\varphi(\mathbf{x})-g(\mathbf{x})\varphi(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

for $x=(\lambda,u)\in X$ and $\varphi\in W_0^{1,p}(\Omega)$, where $\langle\cdot\,,\cdot\rangle_{(W_0^{1,p})',W_0^{1,p}}$ denotes the dual pairing between $W_0^{1,p}(\Omega)'$ and $W_0^{1,p}(\Omega)$. Now $f(x)=0$ in $W_0^{1,p}(\Omega)'$ for $x=(\lambda,u)\in X$ is equivalent with the fact that u is a weak solution to (1) for thickness parameter λ.

The goal is to determine an optimal thickness parameter λ and a corresponding optimal displacement u minimizing the cost functional $J_f : X\to\mathbb{R}$ given by

$$J_f(x)=\int_\Omega \lambda(\mathbf{x})+\frac{\eta}{2}\left|\lambda(\mathbf{x})-\lambda^\circ(\mathbf{x})\right|^2+\beta\min\big(u(\mathbf{x})-u_b(\mathbf{x}),0\big)^3\,\mathrm{d}\mathbf{x}$$

for $x=(\lambda,u)\in X$ subject to (s.t.) the equality constraints $f(x)=0$ in $W_0^{1,p}(\Omega)'$ and to the inequality constraints $\lambda_a\le\lambda(\mathbf{x})\le\lambda_b$ f.a.a. $\mathbf{x}\in\Omega$, where λ_a, λ_b are positive scalars with $\lambda_a\le\lambda_b$, $\eta\ge 0$ is a regularization parameter and $\lambda^\circ\in C^{0,1}(\overline{\Omega})$ is a nominal thickness parameter satisfying $\lambda_a\le\lambda^\circ(\mathbf{x})\le\lambda_b$ f.a.a. $\mathbf{x}\in\Omega$. Furthermore, $\beta\ge 0$ is a weighting parameter and $u_b\in L^\infty(\Omega)$ satisfies $u_b(\mathbf{x})>0$ f.a.a. $\mathbf{x}\in\Omega$. The last term of the cost functional J_f penalizes the situation if the displacement is larger than the given threshold u_b. We introduce the set of admissible thickness parameters by

$$\Lambda_{ad}=\big\{\lambda\in C^{0,1}(\overline{\Omega})\,\big|\,\lambda_a\le\lambda(\mathbf{x})\le\lambda_b \text{ f.a.a. } \mathbf{x}\in\Omega \text{ and } \|\lambda\|_{C^{0,1}(\overline{\Omega})}\le c_b\big\}$$

with $c_b=\|\lambda_b\|_{C^{0,1}(\overline{\Omega})}$ and define $X_{ad}=\Lambda_{ad}\times W_0^{1,p}(\Omega)$. Then, the infinite-dimensional, nonconvex minimization problem can be formulated abstractly as

$$\min J_f(x)\quad\text{s.t.}\quad x\in\mathcal{F}_f=\big\{x\in X_{ad}\,\big|\,f(x)=0\text{ in }W_0^{1,p}(\Omega)'\big\},\tag{3}$$

where $\mathcal{F}_f$ is the set of admissible solutions. Let us refer to [4, 5] for optimal solutions existence results for (3), where a Dirichlet and Neumann optimal control problem governed by the p-Laplace equation is considered.

Solving (1) numerically is a difficult task due to the quasilinear elliptic constraint $f(x)=0$ (fine model). In the next section we utilize instead of the accurate, but complex model (1) a linear elliptic PDE as a simpler model that is much easier to solve. Then we combine the simple and the complex model by applying a space mapping approach.

3 Space mapping

The space mapping is a mapping between the fine model space parameter or variables and the coarse model space. Then the optimization can be carried out for the coarse model, but information from the fine model is utilized to improve the accuracy of the optimization result with respect to the real application.

As introduced in Section 1 the goal is to replace the fine model (1) by the coarse model (2). Later this fine model will be used in the optimization problem. Existence and uniqueness of a weak solution to (2) were discussed in [3]. Let us now define the Banach space $Y = L^\infty(\Omega) \times H_0^1(\Omega)$ and introduce the bilinear operator $c : Y \to H^{-1}(\Omega)$ (*coarse model*) by

$$\langle c(y), \varphi \rangle_{H^{-1},H_0^1} = \int_\Omega 2(1+n)\mu(\mathbf{x})\nabla v(\mathbf{x})\,\mathrm{d}\mathbf{x} - \langle g, \varphi \rangle_{H^{-1},H_0^1}$$

for $y = (\mu, v) \in Y$ and $\varphi \in H_0^1(\Omega)$, where $\langle \cdot\,, \cdot \rangle_{H^{-1},H_0^1}$ stands for the dual pairing between $H_0^1(\Omega)$ and its dual space $H^{-1}(\Omega)$.

Let us now formulate the space mapping. Our fine model is the p-Laplace equation (1) with the model output u together with the thickness parameter λ. The coarse model is given by the linear ellipic PDE (2) with the model output v and the thickness parameter μ. The goal of the space mapping is to adjust the thickness parameter μ in the coarse model so that the model outputs u and v are similar. Furthermore we want to achieve that the thickness parameters μ and λ are not too distinct.

Concentrating on the region of interest (the subset of Ω, where the force g acts) we consider the space mapping on a subset $\mathcal{A} \subseteq \Omega$. We define the restriction operator $\mathcal{R}_\mathcal{A} : L^2(\Omega) \to L^2(\Omega)$ as $\mathcal{R}_\mathcal{A} v = v$ on $\mathcal{A}$ a.e. and $\mathcal{R}_\mathcal{A} v = 0$ otherwise. Further we introduce the set of admissible thickness parameters by

$$M_{ad} = \big\{\mu \in C^{0,1}(\overline{\Omega}) \,\big|\, \mu_a \le \mu(\mathbf{x}) \le \mu_b \text{ f.a.a. } \mathbf{x} \in \Omega \text{ and } \|\mu\|_{C^{0,1}(\overline{\Omega})} \le C_b\big\}$$

with $C_b = \|\mu_b\|_{C^{0,1}(\overline{\Omega})}$. For $\mu \in M_{ad}$ the solution to (2) belongs to $H^2(\Omega)$.

Now we introduce the *space mapping* $\mathcal{P} : \Lambda_{ad} \to M_{ad}$ as follows: for a given thickness parameter $\lambda \in \Lambda_{ad}$ the corresponding $\mu = \mathcal{P}(\lambda) \in M_{ad}$ is the thickness parameter so that $\mathcal{R}_\mathcal{A} v$ is as close as possible to $\mathcal{R}_\mathcal{A} u$. We formulate μ as the solution to a minimization problem. The goal is to determine an optimal thickness μ for a given λ minimizing the cost functional $J_{sp} : Y \to \mathbb{R}$ given by

$$J_{sp}(y) = \frac{\gamma}{2}\int_\mathcal{A} \big|v(\mathbf{x}) - u(\mathbf{x})\big|^2 \mathrm{d}\mathbf{x} + \frac{\kappa}{2}\int_\Omega \big|\mu(\mathbf{x}) - \lambda(\mathbf{x})\big|^2 \mathrm{d}\mathbf{x}$$

for $y = (\mu, v) \in Y$ subject to $\mu \in M_{ad}$ and the equality constraint $c(y) = 0$ in $H^{-1}(\Omega)$, where $\gamma > 0$ is a weighting and $\kappa \ge 0$ is a smoothing parameter.

Let us now formulate the minimization problem more abstractly. We define $Y_{ad} = M_{ad} \times H_0^1(\Omega)$, then the problem can then be written as follows

$$\min J_{sp}(y) \quad \text{s.t.} \quad y \in \mathcal{F}_{sp} = \big\{y \in Y_{ad} \,\big|\, c(y) = 0 \text{ in } H^{-1}(\Omega)\big\}, \qquad (\mathbf{P}_{sp})$$

where $\mathcal{F}_{sp}$ is the set of admissible solutions.

The following theorem ensures existence of optimal solutions to $(\mathbf{P}_{sp})$ and states the first-order necessary optimality conditions. The proof follows from [3] and [8].

Theorem 1. *The problem* $(\mathbf{P}_{sp})$ *has at least one optimal solution* $y^* = (\mu^*, v^*) \in Y_{ad}$*, which can be characterized by first-order necessary optimality conditions: There exists a unique associated Lagrange multiplier* $p^* \in V$ *together with* y^**satisfying the* adjoint equation

$$\begin{aligned} -\mathrm{div}\,\big(2(1+n)\mu^*(\mathbf{x})\nabla p^*(\mathbf{x})\big) &= -\gamma\big(\mathcal{R}_{\mathcal{A}}(v^*-u)\big)(\mathbf{x}) && \textit{f.a.a. } \mathbf{x} \in \Omega, \\ p^*(\mathbf{x}) &= 0 && \textit{f.a.a. } \mathbf{x} \in \Gamma. \end{aligned} \tag{4}$$

Moreover, the variational inequality

$$\int_\Omega \big(\kappa\big(\mu^*(\mathbf{x}) - \lambda(\mathbf{x})\big) + 2(1+n)\big(\nabla v^*(\mathbf{x}) \cdot \nabla p^*(\mathbf{x})\big)\big)\,\big(\mu_\delta(\mathbf{x}) - \mu^*(\mathbf{x})\big)\,\mathrm{d}\mathbf{x} \geq 0$$

holds for all $\mu_\delta \in M_{ad}$.

The optimal control problem given by $(\mathbf{P}_{sp})$ can be written in reduced form

$$\min \hat{J}_{sp}(\mu) \quad \text{s.t.} \quad \mu \in M_{ad}. \tag{$\hat{\mathbf{P}}_{sp}$}$$

The gradient of the reduced cost functional at a given point $\mu \in M_{ad}$ in a direction $\mu_\delta \in L^\infty(\Omega)$ is given by

$$\hat{J}'_{sp}(\mu)\mu_\delta = \int_\Omega \left(\kappa\left(\mu(\mathbf{x}) - \lambda(\mathbf{x})\right) + 2(1+n)\nabla v(\mathbf{x}) \cdot \nabla p(\mathbf{x})\right)\mu_\delta(\mathbf{x})\,\mathrm{d}\mathbf{x},$$

where v satisfies (2) and p solves (4).

In our numerical experiments we assume that $(\hat{\mathbf{P}}_{sp})$ has an inactive solution μ^*, i.e., $\mu_a < \mu^* < \mu_b$ f.a.a. $\mathbf{x} \in \Omega$ and $\|\mu^*\|_{C^{0,1}(\Omega)} < C_b$. We utilize a globalized Newton method with Armijo backtracking line search algorithm [12, p. 37] to solve $(\hat{\mathbf{P}}_{sp})$. In each level of the Newton method the linear system

$$\hat{J}''_{sp}(\mu^\ell)d^\ell = -\hat{J}'_{sp}(\mu^\ell) \tag{5}$$

is solved by the truncated conjugate gradient method [12, p. 169]. We find

$$\big(\hat{J}''_{sp}(\mu^\ell)\mu_\delta\big)(\mathbf{x}) = \kappa\mu_\delta(\mathbf{x}) + 2(1+n)\big(\nabla v_\delta(\mathbf{x}) \cdot \nabla p^\ell(\mathbf{x}) + \nabla v^\ell(\mathbf{x}) \cdot \nabla p_\delta(\mathbf{x})\big)$$

f.a.a. $\mathbf{x} \in \Omega$, where u^ℓ and p^ℓ satisfy (2) and (4) respectively and u_δ and p_δ satisfy linearized state and adjoint equations; see [8]. Another possibility to solve (5) is to utilize a quasi Newton approximation or the Hessian.

4 Surrogate optimization

In this subsection we turn to the surrogate optimization that is used to solve approximately (3). The main idea is to solve the optimization problem using the coarse model $c(y) = 0$, but to take the fine model $f(x) = 0$ into account by the space mapping technique introduced in Section 3.

Let us introduce the Banach space $Z = L^\infty(\Omega) \times H_0^1(\Omega)$ and the subset $Z_{ad} = \Lambda_{ad} \times H_0^1(\Omega)$. We define the cost functional $J_{so} : Z \to \mathbb{R}$ as

$$J_{so}(z) = \int_\Omega \lambda(\mathbf{x}) + \frac{\eta}{2} \left|\lambda - \lambda^\circ\right|^2 + \beta \min\big(v(\mathbf{x}) - u_b(\mathbf{x}), 0\big)^3 \,\mathrm{d}\mathbf{x}$$

for $z = (\lambda, v) \in Z$, where η, λ°, β, u_b are as in Section 2. We consider the optimization problem

$$\min J_{so}(z) \quad \text{s.t.} \quad z \in \mathcal{F}_{so} = \big\{z \in Z_{ad} \,\big|\, c(\mu, v) = 0 \text{ and } \mu = \mathcal{P}(\lambda)\big\}. \qquad (\mathbf{P}_{so})$$

Note that in the surrogate optimization the space mapping is used to link the coarse and the fine model and therefore informations of the fine model are taken into account in the optimization prozess. We suppose that $(\mathbf{P}_{so})$ has a local optimal solution $z^* = (\lambda^*, v^*) \in Z_{ad}$. In particular, we have $v^* = \mathcal{S}_c(\mathcal{P}(\lambda^*))$, where $\mathcal{S}_c$ denotes the solution operator for the coarse model. The corresponding reduced problem is given by

$$\min \hat{J}_{so}(\lambda) \quad \text{s.t.} \quad \lambda \in \Lambda_{ad}$$

with

$$\hat{J}_{so}(\lambda) = \int_\Omega \lambda(\mathbf{x}) + \frac{\eta}{2} \left|\lambda - \lambda^\circ\right|^2 + \beta \min\big(v(\mathbf{x}) - u_b(\mathbf{x}), 0\big)^3 \,\mathrm{d}\mathbf{x}, \quad \lambda \in \Lambda_{ad}.$$

with $v = \mathcal{S}_c(\mathcal{P}(\lambda))$. Next we state the first-order necessary optimality conditions for $(\mathbf{P}_{so})$; see [7].

Theorem 2. *Suppose that $z^* = (\lambda^*, v^*)$ is a local solution to $(\mathbf{P}_{so})$ and the space mapping $\mathcal{P}$ is Fréchet-differentiable. Then there exist unique associated Lagrange multipliers $p^* \in V$ and $\xi^* \in L^2(\Omega)$ together with z^* satisfying the* adjoint equation

$$\begin{aligned} -\mathrm{div}\big(2(1+n)\mu^*(\mathbf{x})\nabla p^*(\mathbf{x})\big) &= -3\beta \min\big(v^*(\mathbf{x}) - u_b(\mathbf{x}), 0\big)^2 && \textit{f.a.a. } \mathbf{x} \in \Omega, \\ p^*(\mathbf{x}) &= 0 && \textit{f.a.a. } \mathbf{x} \in \Gamma. \end{aligned}$$

Moreover, the variational inequality

$$\int_\Omega \big(1 + \eta\left(\lambda^*(\mathbf{x}) - \lambda^\circ(\mathbf{x})\right) + 2(1+n)\mathcal{P}'(\lambda^*)^\star\big(\nabla v^*(\mathbf{x}) \cdot \nabla p^*(\mathbf{x})\big)\big) \\ \left(\lambda_\delta(\mathbf{x}) - \lambda^*(\mathbf{x})\right) \mathrm{d}\mathbf{x} \geq 0$$

holds for all $\lambda_\delta \in \Lambda_{ad}$, where $\mathcal{P}'(\lambda^*)^\star$ denotes the adjoint operator to $\mathcal{P}'(\lambda^*)$.

It follows that the gradient $\hat{J}'_{so}$ of the reduced cost functional is given by

$$\hat{J}'_{so}(\lambda) = 1 + \eta(\lambda - \lambda^\circ) + \mathcal{P}'(\lambda)^\star 2(1+n)\nabla v(\cdot) \cdot \nabla p(\cdot) \quad \text{in } \Omega,$$

where the function v satisfies

$$\begin{aligned} -\mathrm{div}\left(2(1+n)\mu(\mathbf{x})\nabla v(\mathbf{x})\right) &= g(\mathbf{x}) && \text{f.a.a. } \mathbf{x} \in \Omega, \\ v(\mathbf{x}) &= 0 && \text{f.a.a. } \mathbf{x} \in \Gamma \end{aligned}$$

with $\mu = \mathcal{P}(\lambda)$ and p is the solution to

$$\begin{aligned} -\mathrm{div}\left(2(1+n)\mu(\mathbf{x})\nabla p(\mathbf{x})\right) &= -3\beta \min(v^*(\mathbf{x}) - u_b(\mathbf{x}), 0)^2 && \text{f.a.a. } \mathbf{x} \in \Omega, \\ p(\mathbf{x}) &= 0 && \text{f.a.a. } \mathbf{x} \in \Gamma. \end{aligned}$$

To avoid the computation of the operator $\mathcal{P}'(\lambda)$ we apply Broyden's updating formula providing a matrix B which can be used to replace $\mathcal{P}'(\lambda)$, but also $\mathcal{P}'(\lambda)^\star$. We use a modified Broyden's update formula introduced in [6]:

$$B_{\ell+1} = B_\ell + \frac{\widetilde{\mathcal{P}_\delta} - B_\ell \lambda_\delta}{\|\lambda_\delta\|^2_{L^2(\Omega)}} \langle \lambda_\delta, \cdot \rangle_{L^2(\Omega)}$$

with

$$\widetilde{\mathcal{P}_\delta} = \mathcal{P}_\delta + \sigma \, \frac{\hat{J}_\delta - \langle \hat{J}'_{sur}(\lambda^k), \mathcal{P}_\delta \rangle_{L^2(\Omega)}}{\|\lambda_\delta\|^2_{L^2(\Omega)}} \hat{J}'_{sur}(\lambda^\ell),$$

where $\hat{J}_\delta = \hat{J}'_{so}(\lambda^{\ell+1}) - \hat{J}'_{so}(\lambda^\ell)$, $\lambda_\delta = \lambda^{\ell+1} - \lambda^k$ and $\mathcal{P}_\delta = \mathcal{P}(\lambda^{\ell+1}) - \mathcal{P}(\lambda^\ell)$. Note that for $\sigma = 0$ we get the classical Broyden's update formula.

For the numerical solution we apply the gradient projection method using Broyden's updating to obtain an approximation of the sensitivity $\mathcal{P}'(\lambda)$.

5 Numerical results

In this section we present numerical results for the space mapping and the surrogate optimization. For our numerical example we consider a domain representing a simplified door, denoted by Ω. The gray line in Figure 2 (left plot) indicates the section of the boundary, where homogeneous Neuman boundary conditions of the form $\langle \nabla u(x), \overrightarrow{n} \rangle_2 = 0$ are applied, where $\overrightarrow{n}$ denotes an outer normal on the boundary and $\langle \cdot\,, \cdot \rangle_2$ the Euclidean inner product. We use the finite element discretization and solvers for (1) and (2) provided by the `Matlab Partial Differential Equation Toolbox`. The right-hand side $g(\mathbf{x})$ (force term) is given as follows:

$$g(\mathbf{x}) = \begin{cases} 47.71, \, \mathbf{x} \in \mathcal{B}_r(\mathbf{x}_{mid}) = \left\{ \mathbf{x} \in \Omega \,\middle|\, |\mathbf{x}_{mid} - \mathbf{x}|_2 < r \right\}, \\ \quad 0, \text{ otherwise,} \end{cases}$$

where $\mathbf{x}_{mid} = (0.5, 0.45)^T$ and $r = 0.1$. This force term is indicated as the gray circle in Figure 2 (left plot). Let us next state the parameters for our numerical example. The Hollomon coefficient is set to $n = 0.22$. For the space mapping we choose the weight parameter as $\gamma = (\int_\Omega |u(\mathbf{x})|^2 \, \mathrm{d}\mathbf{x})^{-1}$ and $\kappa = 10^{-3}\gamma$. Further we choose the region $\mathcal{A}$ to be a circle with radius 0.2 and midpoint $(0.5, 0.45)$, illustrated in Figure 2 (left plot) by a black circle. Next we have a look at the parameters for the surrogate optimization. We choose η, β and λ° to be 1.25, 25^5 and 1.7, respectively. The threshold u_b is set to 0.3 and the bounds for the thickness parameter are set to $\mu_a = \lambda_a = 0.05$ and $\mu_b = \lambda_b = 10$. As a stopping criteria we choose the norm of the reduced gradient to be smaller than 0.1 times the maximum diameter of the finite elements. We will report on numerical results for two different settings for the parameter σ.

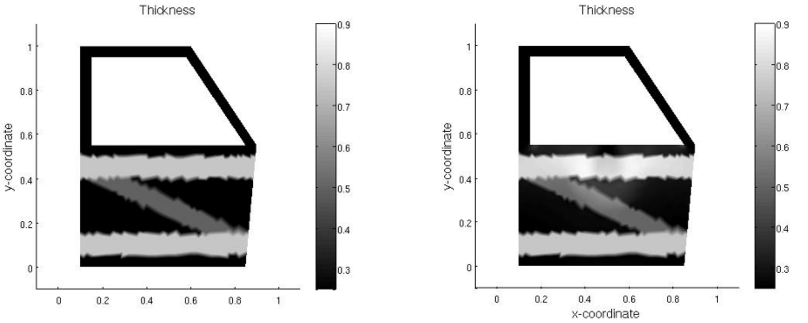


Fig. 1. Initial thickness parameter (left plot) and the optimal thickness parameter μ^* (right plot) for the space mapping using the Newton-CG method.

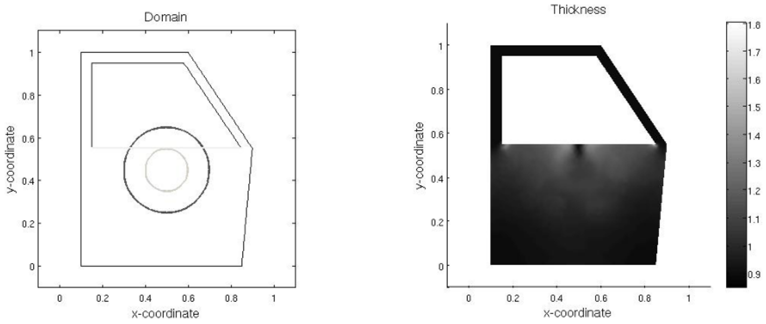


Fig. 2. Domain Ω with region $\mathcal{A}$ (black circle) and region $B_r(\mathbf{x}_{mid})$ (gray circle) (left plot) and the optimal thickness parameter λ^* (right plot) for the surrogate optimization.

Let us first present a numerical result for the space mapping. As an initial thickness for the space mapping we choose a structured initial thickness parameter, shown in Figure 1 (left plot). In the right plot of Figure 1 we present the corresponding thickness parameter μ^* computed by the space mapping. We observe that the thickness parameter is enlarged in the region $\mathcal{A}$. In Table 1 the numerical results and performace for the space mapping utilizing the

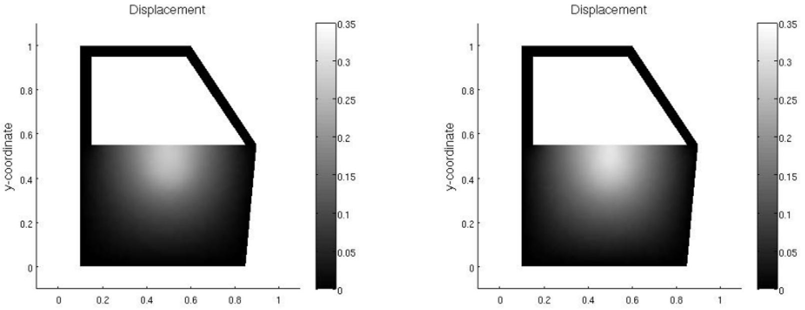


Fig. 3. Displacement v solving (2) for $\mu = \lambda^*$ (left plot) and solution u to (1) for $\lambda = \lambda^*$ (right plot).

Table 1. Summary of the results for the space mapping and the performance for two different methods.

	v	u	BFGS	Newton-CG
$\max_{\Omega}$	0.68961	0.59601	0.59541	0.59462
Iterations			9	4
Time (sec)			8.52	4.81

Table 2. Summary of the results for the surrogate optimization and the performance of the gradient projection method for two different Broyden's updates ($\sigma = 0$ and $\sigma = 0.2$).

σ	$\max_{\Omega} u$	$\max_{\Omega} v$	Volume	$\min_{\Omega} \lambda$	$\max_{\Omega} \lambda$	$\lVert u - v \rVert_{L^2(\Omega)}$	Iter	Time (sec)
0.0	0.31307	0.27650	0.48857	0.89759	1.77613	0.01198	10	82.72
0.2	0.31313	0.27606	0.48899	0.89555	1.67856	0.01204	7	57.65

Newton-CG and the BFGS algorithms are summarized. It turns out that for the thickness parameter shown in Figure 1 (left plot) the maximal displacements for v (solution to the linear model) and u (solution to the p-Laplacian) are quite different. Using the space mapping the optimal thickness parameter leads to a maximal displacement in the linear model that is very close to maximal one of u. Furthermore, we observe from Table 1 that the Newton-CG method performs significantly better then the BFGS method while giving nearly the same results measured in the maximum displacement.

Next we present the numerical results for the surrogate optimization. In Figure 2 (right plot) the optimal thickness parameter λ^* for the surrogate optimization is shown. The corresponding displacements for the coarse and fine model are shown in Figure 3 (left and right plot), respectively. Comparing the plots in Figure 3 we observe that the maximum displacement of the non-linear model is significantly larger than the maximal displacement for the linear model. Therefore, if we make the thickness parameter λ^* smaller, the maximal displacement for the non-linear model would be significantely larger than the threshold $u_b = 0.3$. The surrogate optimization takes this fact into account. In Table 2 we summarize the numerical results for the two different values for σ. Note that the modified Broyden's update gives a better performance than the classical Broyden's update with respect to the number of iterations and CPU time while giving nearly the same results. Further it is observed that for different initial guesses of λ^0 the algorithm converges to the same numerical solution.

References

1. M.H. Bakr, J.W. Bandler, K. Masden, and J. Søndergaard (2001) An introduction to the space mapping technique. Optimization and Engineering 2(4):369–384
2. J.W. Bandler, R.M. Biernacki, Shao Hua Chen, P.A. Grobelny, R.H. Hemmers (1994) Space mapping technique for electromagnetic optimization. IEEE Transactions on Microwave Theory and Techniques 42(12):2536–2544
3. E. Casas (1992) Optimal control in coefficients of elliptic equations with state constraints. Applied Mathematics and Optimization 26(1):21–37
4. E. Casas, L.A. Fernández (1993) Distributed control of systems governed by a general class of quasilinear elliptic equations. Journal of Differential Equations 104(1):20–47
5. E. Casas, L.A. Fernández (1995) Dealing with integral state constraints in boundary control problems or quasilinear elliptic equations. SIAM Journal on Control and Optimization 33(2):568–589
6. M. Hintermüller, L.N. Vicente (2005) Space mapping for optimal control of partial differential equations. SIAM Journal on Optimization 15(4):1002–1025
7. O. Lass, C. Posch, G. Scharrer, S. Volkwein (Submitted 2009) Space mapping techniques for the optimization of a thickness parameter in the p-Laplace equation.
8. O. Lass (2009) Efficient numerical space mapping techniques for the p-Laplace equation. Diploma thesis, Karl-Franzens-Universität, Graz
9. S.J. Leary, A. Bhaskar, A.J. Keane (2001) A constraint mapping approach to the structural optimization of an expensive model using surrogates. Optimization and Engineering 2(4):385–398
10. J. Marburger (2007) Space mapping and optimal shape design. Diploma Thesis, Technische Universität, Kaiserslautern
11. H. Maurer, J. Zowe (1979) First and second order necessary and sufficient optimality conditions for infinite-dimensional programming problems. Mathematical Programming 16(1):98–110
12. J. Nocedal, S.J. Wright (2006) Numerical optimization, 2. Edition. Springer Series in Operations Research, Springer-Verlag, New York
13. G. Scharrer, S. Volkwein, T. Heubrandtner (2010) Mathematical optimization of the plate volume under a p-laplace partial differential equation constraint by using standard software. To appear in International Journal of Multiphysics
14. L.N. Vicente (2003) Space mapping: models, sensitivities, and trust-region methods. Optimization and Engineering 4(3):159–175

Numerical Solutions for Optimal Control of Monodomain Equations in Cardiac Electrophysiology

Ch. Nagaiah[1], K. Kunisch[1], and G. Plank[2]

[1] Institute of Mathematics and Scientific Computing, University of Graz, Heinrichstr. 36, Graz, A-8010, Austria.
nagaiah.chamakuri@uni-graz.at,karl.kunisch@uni-graz.at

[2] Institute of Biophysics, Medical University of Graz, Harrachgasse 21, Graz, A-8010, Austria. gernot.plank@meduni-graz.at

Summary. In this article, we present computational techniques for optimal control of monodomain equations which are a well established model for describing wave propagation of the action potential in the heart. The model consists of a non-linear parabolic partial differential equation of reaction-diffusion type, where the reaction term is a set of ordinary differential equations which characterize the dynamics of cardiac cells.

Specifically, an optimal control formulation is presented for the monodomain equations with an extracellular current as the control variable which must be determined in such a way that wavefronts of transmembrane voltage are smoothed in an optimal manner. Numerical results are presented based on the optimize before discretize and discretize before optimize techniques. Moreover, the derivation of the optimality system is given for both techniques and numerical results are discussed for higher order methods to solve the optimality system. Finally, numerical results are reported which show superlinear convergence when using Newton's method.

1 Introduction

The bidomain equations are considered to be among the most accurate descriptions of cardiac electric activity at the tissue and organ level. They characterize cardiac tissue as a syncytial continuum, derived via a homogenization procedure, that consists of two interpenetrating domains, intracellular and extracellular, separated by a cellular membrane at any given point in space. The equations state that current leaving one domain, by traversing the cellular membrane, acts as source of current density in the other domain. Mathematically, this leads to a degenerate parabolic problem that can be recast as an elliptic partial differential equation (PDE) coupled to a parabolic PDE.The elliptic PDE expresses the extracellular potential distribution, Φ_e, as a function

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*, DOI 10.1007/978-3-642-12598-0_35, © Springer-Verlag Berlin Heidelberg 2010

of the transmembrane voltage distribution, V_m whereas the parabolic PDE models cellular activation and recovery processes (reaction term) and how they affect adjacent tissue by diffusion. We refer to [8, 2] for more detailed derivation of the bidomain model and further discussions. The numerical solution of the bidomain equations is computationally expensive owing to the high spatio-temporal resolution required to resolve the fast transients and steep gradients governing wavefront propagation in the heart. Assuming that the anisotropy ratios of the two spaces are equal leads to a reduced bidomain model, referred to as monodomain, which can be solved at a much cheaper cost by avoiding the time consuming solution of the elliptic PDE [7]. Under most circumstances of practical relevance the monodomain model can be set up to approximate the bidomain model fairly well [9, 6].

Under pathological conditions regular activation sequences may decay into complex and irregular patterns which impair the heart's capability to pump blood. If sufficiently fast and disorganized, such patterns, referred to as cardiac arrhythmias, may lead to death if not treated immediately. Electrical defibrillation, i.e. the delivery of a strong electrical shock to the heart, is the only known therapy to reliably restore a normal rhythm. During defibrillation shocks extracellular currents are injected via electrodes to establish an extracellular potential distribution which acts to reduce the complexity of the activity. This is achieved either by extinguishing all electrical activity, i.e. the entire tissue returns to its quiescent state, or gradients in V_m are smoothed out to drive the system to a less heterogeneous state which reduces the likelihood of triggering new wavefronts via "break" mechanisms when switching off the applied field. To optimally control cardiac arrhythmias, it is essential to determine the control response to an applied electric field as well as the optimal extracellular current density that acts to damp gradients of transmembrane voltage in the system. The present article is devoted to the development of efficient numerical techniques to solve this optimal control problem for the monodomain equations.

The finite element method is chosen for the spatial discretization and higher order linearly implicit Runge-Kutta time stepping methods for the temporal discretization. Numerical techniques for solving optimal control problems typically require combining a discretization technique with an optimization method. We will give a brief description of the optimize before discretize technique, that is write the continuous optimality system first before discretizing them, and discretize before optimize, that is first discretize the differential equations before discretizing the optimality system to solve the monodomain equations. To the authors knowledge this is the first attempt to combine the linearly implicit time stepping methods with the discretize before optimize technique to solve the optimality system. The optimal control approach is based on minimizing a properly chosen cost functional $J(V_m, I_e)$ depending on the extracellular current I_e as input and on the transmembrane potential V_m as one of the state variables.

The organization of this article is as follows: in the next section the governing equations for the action potential and the behavior of the ionic current variables using ionic models are described. In section 3 the control problem is posed for the monodomain equations and the optimality system is derived for the two discretization approaches. Numerical results are presented in section 4. Finally concluding remarks are given.

2 The monodomain equations

The monodomain model consists of the equations for the transmembrane potential and ionic current variables. We set $Q = \Omega \times [0, t_f]$ where $\Omega \subset \mathbf{R}^d$, $d = 2$, denotes the cardiac tissue sample domain.

$$\nabla \cdot \bar{\sigma_i} \nabla V_m = \frac{\partial V_m}{\partial t} + I_{ion}(V_m, w) - I_e \quad \text{in } Q \tag{1}$$

$$\frac{\partial w}{\partial t} = g(V_m, w) \quad \text{in } Q \tag{2}$$

where $V_m : Q \to \mathbf{R}$ is the transmembrane voltage, $w : Q \to \mathbf{R}^n$ represents the ionic current variables, $\bar{\sigma_i} : \Omega \to \mathbf{R}^{d \times d}$ is the intracellular conductivity tensor, I_e is an extracellular current density stimulus, and I_{ion} is the current density flowing through the ionic channels. The function $g(V_m, w)$ determines the evolution of the gating variables. Eq. (1) is a parabolic equation and Eq. (2) is a set of ordinary differential equations which can be solved independently for each node. Here the initial and boundary conditions are chosen as

$$\bar{\sigma_i} \nabla V_m \cdot \eta = 0 \quad \text{on } \partial Q = \partial \Omega \times [0, t_f] \tag{3}$$

$$w(0) = w_0 \quad \text{and } V_m(0) = V_0 \quad \text{in } \Omega\,. \tag{4}$$

Ionic model

The ionic activity is modeled by nonlinear ordinary differential equations. For the present paper we use the modified FitzHugh-Nagumo (FHN) model based on the work of Rogers and McCulloch [10] and the simulation parameters are taken from Colli Franzone et al. [1].

$$I_{ion}(V_m, w) = G V_m \left(1 - \frac{V_m}{v_{th}}\right)\left(1 - \frac{V_m}{v_p}\right) + \eta_1 V_m w\,. \tag{5}$$

$$g(V_m, w) = \eta_2 \left(\frac{V_m}{v_p} - \eta_3 w\right). \tag{6}$$

where $G, \eta_1, \eta_2, \eta_3$ are positive real coefficients, v_{th} is a threshold potential and v_p the peak potential.

3 Optimal control framework and numerical discretization

In this section we set forth the optimal control problem, for which the numerical experiments were carried out. We consider

$$\text{(P)} \qquad \begin{cases} \min J(V_m, I_e)\,, \\ \\ e(V_m, w, I_e) = 0 \quad \text{in} \quad Q\,, \end{cases} \tag{7}$$

where V_m and w are the state and I_e is the control variable. The coupled PDE and ODE constraints (1-2) for the monodomain equation together with initial and boundary conditions for V_m are expressed as $e(V_m, w, I_e) = 0$. The control variable I_e is chosen such that it is nontrivial only on the control domain Ω_{con} of Ω and I_e equals zero on $(\Omega \setminus \Omega_{con}) \times (0, T)$.

The cost functional which is used to optimize the potentials and currents is given by

$$J(V_m, I_e) = \frac{1}{2}\int_0^T \left(\int_{\Omega_{obs}} |V_m - Z|^2 \,\mathrm{d}\Omega_{obs} + \alpha \int_{\Omega_{con}} |I_e|^2 \,\mathrm{d}\Omega_{con} \right) dt, \tag{8}$$

where α is the weight of the cost of the control, Ω_{obs} is the observation domain and Ω_{con} is the control domain. If $Z = 0$ then the interpretation of the cost-functional J for the problems to be considered is such that by properly applying I_e excitation waves are suppressed in the region Ω_{obs}. The inclusion of the tracking type term Z in the cost functional serves code-validation purposes.

Due to their size and complexity PDE based optimization problems are generally challenging to solve in practice. The interplay of optimization and infinite dimensionality of the problem is a crucial one. There are essentially two approaches to deal with it. In the optimize before discretize (OBD) approach, first a minimization strategy is applied to the continuous optimal control problem, (this may consist of deriving the optimality system), and subsequently the resulting formalism is discretized. Alternatively, in the discretize before optimize (DBO) approach, first the differential equations as well as the cost J in (P) are discretized and subsequently the optimization procedure for solving the finite-dimensional minimization problem is fixed.

3.1 Optimize before discretize

In this subsection we follow an OBD technique to solve the monodomain model. More specifically for the problem under consideration the Lagrangian is defined by

$$\begin{aligned} \mathcal{L}\ \ (V_m, w, I_e, p, q) = J(V_m, I_e) + \int_0^T \int_\Omega \left(\frac{\partial w}{\partial t} - g(V_m, w) \right) q \,\mathrm{d}\Omega\,\mathrm{d}t \\ + \int_0^T \int_\Omega \left(\nabla \cdot \bar{\sigma}_i \nabla V_m - \frac{\partial V_m}{\partial t} + I_{ion}(V_m, w) - I_e \right) p \,\mathrm{d}\Omega\,\mathrm{d}t, \end{aligned} \tag{9}$$

where the initial conditions are kept as explicit constraints. The first order optimality system is obtained by formally setting the partial derivatives of $\mathcal{L}$ equal to 0. We find

$$\begin{aligned}\mathcal{L}_{V_m}: &\quad (V_m - Z)_{\Omega_{obs}} + \nabla\cdot\bar{\sigma_i}\nabla p + p_t - (I_{ion})_{V_m} p - g_{V_m} q = 0\,, &(10)\\ \mathcal{L}_w: &\quad -(I_{ion})_w p - q_t - g_w q = 0\,, &(11)\end{aligned}$$

where the subscripts V_m and w denote partial derivatives with respect to these variables. Further we obtain the

$$\begin{aligned}&\text{terminal conditions: } p(T) = 0, \quad q(T) = 0, &(12)\\ &\text{boundary conditions: } \bar{\sigma_i}\nabla p\cdot\eta = 0 \quad \text{on } \partial Q\,, &(13)\\ &\text{and the optimality condition: } \mathcal{L}_{I_e}: \quad \alpha I_e + p = 0\,, \quad \text{on } \Omega_{con}\,. &(14)\end{aligned}$$

To solve (P) numerically we need to solve the coupled system of primal equations (1-2), adjoint equations (10-11), together with initial conditions (4), boundary conditions (3,13), terminal conditions (12), and the optimality system (14). The optimality system serves as a gradient of the cost functional for our computations.

In this study, we have chosen the finite element method for the spatial- and higher order linearly implicit Runge-Kutta time stepping methods for the temporal discretizations, specifically a 2-stage Rosenbrock type method [3]. We now give a brief description of spatial and temporal discretizations for the primal and adjoint equations. For further details we refer to Nagaiah et al. [4].

Discretization of primal and adjoint problems

In computations, the primal problem is solved by decoupling the parabolic part from the ordinary differential equation. In a first step we use the Euler explicit time stepping method to solve the ODE part. In a second step, using the new solution of the gating variables w, we solve the parabolic part by employing a Rosenbrock time stepping method, refer to [4, 5] for more details. After the space and time discretization for the primal problem, the system of linear equations can be expressed as follows:

$$\begin{aligned}\mathbf{w}_n &= \mathbf{w}_{n-1} + \delta t\eta_2\left(\frac{\mathbf{v}_{n-1}}{v_p} - \eta_3\mathbf{w}_{n-1}\right)\\ \mathbf{J}_1\mathbf{k}_1^n &= -\mathbf{K}\mathbf{v}_{n-1} - \mathbf{M}\mathbf{I}_{ion}(\mathbf{v}_{n-1},\mathbf{w}_n) + \mathbf{M}\mathbf{I}_e\,,\\ \mathbf{J}_1\mathbf{k}_2^n &= -\mathbf{K}\left(\mathbf{v}_{n-1} + \alpha_{21}\mathbf{k}_1^n\right) - \mathbf{M}\mathbf{I}_{ion}(\mathbf{v}_{n-1} + \alpha_{21}\mathbf{k}_1^n, \mathbf{w}_n) + \mathbf{M}\mathbf{I}_e - \frac{c_{21}}{\delta t}\mathbf{M}\mathbf{k}_1^n\\ \mathbf{v}_n &= \mathbf{v}_{n-1} + m_1\mathbf{k}_1^n + m_2\mathbf{k}_2^n, \qquad \text{for } n = 1,\ldots,N_t, \qquad (15)\end{aligned}$$

where $\mathbf{K}$ is the stiffness matrix, $\mathbf{M}$ is the mass matrix, $\mathbf{J}_1 = (\frac{1}{\delta t\gamma}\mathbf{M} + \mathbf{K} + \mathbf{M}[\mathbf{I}_{ion}(\mathbf{v}_{n-1},\mathbf{w}_n)]_{\mathbf{v}})$, N_t is the maximum number of time steps, the coefficients $\gamma, \alpha_{ij}, c_{ij}$ are constants and the subscript $\mathbf{v}$ denotes the partial derivative with respect to this variable. For solving the linear system the BiCGSTAB method with ILU preconditioning is used. We use the same discretization techniques to solve the adjoint problem. After spatial discretization by FEM and time discretization by a 2-stage Rosenbrock type method for the adjoint problem the system can be written as follows:

$$\begin{aligned}
\mathbf{q}_n &= (1-\delta t\eta_2\eta_3)\mathbf{q}_{n+1} + \delta t\eta_1 \mathbf{v}_{n+1}\mathbf{p}_n \\
\mathbf{J}_2\mathbf{l}_1 &= \mathbf{K}\mathbf{p}_{n+1} + \mathbf{M}[\mathbf{I}_{ion}(\mathbf{v}_{n+1})]_{\mathbf{v}}\mathbf{p}_{n+1} + \frac{\eta_2}{v_p}\mathbf{M}\mathbf{q}_n - \mathbf{M}(\mathbf{v}_{n+1}-\mathbf{z}_{n+1})_{\Omega_{obs}}\,, \\
\mathbf{J}_2\mathbf{l}_2 &= \mathbf{K}\left(\mathbf{p}_{n+1}+\alpha_{21}\mathbf{l}_1\right) + \mathbf{M}[\mathbf{I}_{ion}(\mathbf{v}_{n+1})]_{\mathbf{v}}\left(\mathbf{p}_{n+1}+\alpha_{21}\mathbf{l}_1\right) + \frac{\eta_2}{v_p}\mathbf{M}\mathbf{q}_n \\
&\qquad -\mathbf{M}(\mathbf{v}_{n+1}-\mathbf{z}_{n+1})_{\Omega_{obs}} - \frac{c_{21}}{\tau}\mathbf{M}\mathbf{l}_1 \\
\mathbf{p}_n &= \mathbf{p}_{n+1} + m_1\mathbf{l}_1 + m_2\mathbf{l}_2\,, \text{ for } n=1,\ldots,N_t-1\,, \qquad (16) \\
&\quad \text{where } \mathbf{J}_2 = -\left(\frac{1}{\tau^n\gamma}\mathbf{M} - (\mathbf{K}+\mathbf{M}[\mathbf{I}_{ion}(\mathbf{v}_{n+1})]_{\mathbf{v}_{n+1}})\right)
\end{aligned}$$

3.2 Discretize before optimize

In this subsection we explain a discretize before optimize (DBO) technique to solve the monodomain model. This technique first transforms the original continuous problem into a finite dimensional optimization problem by discretizing in space and time. Then the fully discretized optimization problem is solved by existing optimization solvers. First, in this process the objective functional is written as follows

$$\begin{aligned}
J(\mathbf{v},\mathbf{I_e}) &= \frac{\delta t}{2}(\sum_{n=1}^{N_t-1}(\mathbf{v}_n-\mathbf{z}_n)^\top\mathbf{M}(\mathbf{v}_n-\mathbf{z}_n) + \alpha(\mathbf{I}_e^n)^\top\mathbf{M}\mathbf{I}_e^n) \\
+ &\quad \frac{\delta t}{4}\left[(\mathbf{v}_{N_t}-\mathbf{z}_{N_t})^\top\mathbf{M}(\mathbf{v}_{N_t}-\mathbf{z}_{N_t}) + \alpha(\mathbf{I}_e^{N_t})^\top\mathbf{M}\mathbf{I}_e^{N_t}\right] + \frac{\delta t}{4}\alpha(\mathbf{I}_e^0)^\top\mathbf{M}\mathbf{I}_e^0.
\end{aligned}$$

To solve the monodomain problem with the DBO approach we discretize the problem first in space and time. For the space discretization we used piecewise linear FEM, and for the temporal discretization a 2 stage Rosenbrock type method. The resulting algebraic system can be obtained as in Eq. (15). The corresponding Lagrangian is given by

$$\begin{aligned}
&\mathcal{L}(\mathbf{w},\mathbf{v},\mathbf{I}_e,\mathbf{k}_1,\mathbf{k}_2,\mathbf{p},\mathbf{q},\phi,\psi) \\
&= J(\mathbf{v},\mathbf{I}_e) + \sum_{n=1}^{N}\mathbf{q}_n^\top\left(\mathbf{M}\mathbf{w}_n - \mathbf{M}\mathbf{w}_{n-1} - \delta t\ \mathbf{M}g(\mathbf{v}_{n-1},\mathbf{w}^{n-1})\right) \\
&\quad + \sum_{n=1}^{N}\phi_n^\top\left(\mathbf{J}_1\mathbf{k}_1^n + \mathbf{K}\mathbf{v}_{n-1} + \mathbf{M}\mathbf{I}_{ion}(\mathbf{v}_{n-1},\mathbf{w}_n) - \mathbf{M}\mathbf{I}_e^n\right) \\
&\quad + \sum_{n=1}^{N}\psi_n^\top\Big(\mathbf{J}_1\mathbf{k}_2^n + \mathbf{K}\left(\mathbf{v}_{n-1}+\alpha_{21}\mathbf{k}_1^n\right) + \mathbf{M}\mathbf{I}_{ion}(\mathbf{v}_{n-1}+a_{21}\mathbf{k}_1^n,\mathbf{w}_n)
\end{aligned}$$

$$-\mathbf{M}\mathbf{I}_e^n + \mathbf{M}\frac{c_{21}}{\delta t}\mathbf{k}_1^n\Big) + \sum_{n=1}^{N} \mathbf{p}_n^\top (\mathbf{v}_n - \mathbf{v}_{n-1} - m_1\mathbf{k}_1^n - m_2\mathbf{k}_2^n)\,.$$

The first order optimality system is obtained by formally setting the partial derivatives of $\mathcal{L}$ equal to 0. We find

$$\begin{aligned}
\mathcal{L}_{\mathbf{w}^n}: &\quad \mathbf{q}_n^\top - \mathbf{q}_{n+1}^\top + \delta t\eta_2\eta_3\mathbf{q}_{n+1}^\top + \phi_n^\top\eta_1\mathbf{v}^{n+1} + \psi_n^\top\left(\mathbf{v}^{n+1} + a_{21}\mathbf{k}_1^n\right)\eta_1 = 0\\
\mathcal{L}_{\mathbf{k}_2^n}: &\quad \psi_n^\top\mathbf{J}_1 - m_2\mathbf{p}_n^\top = 0\\
\mathcal{L}_{\mathbf{k}_1^n}: &\quad \phi_n^\top\mathbf{J}_1 + \psi_n^\top\mathbf{K}a_{21} + \psi_n^\top\mathbf{M}(\mathbf{I}_{ion})_{\mathbf{k}_1} + \frac{c_{21}}{\delta t}\psi_n^\top\mathbf{M} - m_1\mathbf{p}_n^\top = 0\\
\mathcal{L}_{\mathbf{v}}: &\quad \delta t[\mathbf{M}(\mathbf{v}_n - \mathbf{z}_n)_{\Omega_{obs}} - \frac{\eta_2}{v_p}\mathbf{M}\mathbf{q}_{n+1}^\top] + \phi_{n+1}^\top\mathbf{K} + \phi_{n+1}^\top\mathbf{M}(\mathbf{I}_{ion}(\mathbf{v}))_v\\
&\quad +\psi_{n+1}^\top\mathbf{K} + \psi_{n+1}^\top\mathbf{M}\left(\mathbf{I}_{ion}(\mathbf{v}^{n+1} + a_{21}\mathbf{k}_1, \mathbf{w}^n)\right)_v + \mathbf{p}_n^\top - \mathbf{p}_{n+1}^\top = 0 \qquad (17)\\
\mathcal{L}_{\mathbf{v}_{Nt}}: &\quad \mathbf{p}_{Nt} = -\frac{\delta t}{2}\mathbf{M}(\mathbf{v}_{Nt} - \mathbf{z}_{Nt}) \qquad (18)\\
\mathcal{L}_{\mathbf{I}_e}: &\quad \delta t\alpha\mathbf{M}\mathbf{I}_e^n = \mathbf{M}(\phi_n + \psi_n)\,, \quad \text{where } n = N-1,\dots,1 \qquad (19)\\
\mathcal{L}_{\mathbf{I}_e^{Nt}}: &\quad \frac{\delta t}{2}\alpha\mathbf{M}\mathbf{I}_e^{Nt} = \mathbf{M}(\phi_{Nt} + \psi_{Nt}). \qquad (20)
\end{aligned}$$

In this case eqs. (19) and (20) serve as a gradient of the cost functional in computations.

3.3 Comparison of optimization methods

If we observe the first derivative of the cost functional, it involves the adjoint stage solutions ϕ_n and ψ_n of time stepping method in the DBO case and in the OBD case it involves the adjoint variable of the primal solution. The terminal solution to solve the adjoint problem is different in the DBO from the OBD case. Also, one needs to evaluate two extra matrix times vector products in the DBO case, see eq. (17), in comparison to algebraic system of the OBD. If one uses Newton's method to solve the optimality system, the DBO case requires more memory than the OBD case, because the stage solutions of primal problem are involved in the linearized primal and adjoint equations.

A nonlinear conjugate gradient (NCG) method and Newton's method are adopted to solve the optimality system. In both cases a line search is required. For this purpose we use the strong Wolfe conditions with a back tracking strategy. A more in-depth description will be found in [4, 5] to solve the current optimization problem.

4 Results

In this section numerical results are presented to demonstrate the capability of dampening an excitation wave of the transmembrane potential by properly applying an extracellular current stimulus. In this note the numerical results for the OBD and DBO approaches are compared for 1D examples, see [5] for 2D results. Also comparisons with respect to the NCG and Newton optimization algorithms are given. The computational domain is $\Omega = (0,1)$. The relevant subdomains are depicted in Figure 1. The observation domain is $\Omega_{obs} = \Omega \backslash (\Omega_{f1} \cup \Omega_{f2})$, the excitation domain is Ω_{exi} and the control domain is $\Omega_{con} = \Omega_{con1} \cup \Omega_{con2}$.

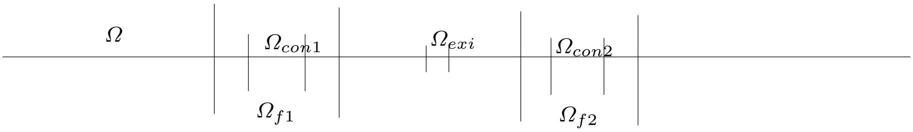


Fig. 1. Control and excitation region at the cardiac domain

The choice $Z = 0$ corresponds to the desire to dampen the wave in Ω_{obs}. For the computations the simulation time is set to 4 *msec*. A uniform spatial mesh consisting of 100 nodes, and 200 equidistant time steps are used. Also we assume that the initial wave excitation takes place on the excitation domain. In all simulations the weight of the cost of the control is fixed at $\alpha = 5 \cdot 10^{-3}$ and the optimization iterations were terminated when the following condition is satisfied: $\|\nabla J_k\|_\infty \leq 10^{-3}(1 + |J_k|)$ or difference of the cost functional between two successive optimization iterations is less than 10^{-3}. The code is implemented using MATLAB-7.4 version.

The continuous L^2 norm of the gradient and the minimum value of the cost functional with respect to the optimization iterations are depicted in Figure 2 for OBD and DBO, using the NCG and Newton optimization algorithms. The norm of the gradient and the minimal values of the cost functional decrease more rapidly for Newton's method. In this case both OBD and DBO take 7 optimization iterations to reach the stopping criterion. The DBO algorithm is bit faster and takes 13 sec of CPU time. The OBD algorithm takes 1.04 times of CPU time more than the DBO case. Indeed, there is no big difference between the OBD and DBO techniques for this particular problem. Also, similar behavior between the OBD and DBO is observed using the NCG algorithm. For all methods the cost functional value is approximately 102 at the optimal state solution. The optimal state solution of transmembrane voltage is shown in Figure 3 at time $t = 0.04$ msec and $t = 1.40$ msec and we can observe that excitation wave is completely dampened.

The line search algorithm takes small step lengths at the beginning of optimization iterations and full steps towards the end of the iterations. In Table 4 the optimization iterations, the norm of the gradient of the cost functional and the order of convergence for the OBD method using Newton's algorithm is presented. From this table we can conclude that the OBD technique based on the Newton method shows super linear convergence from iteration 3 to 6.

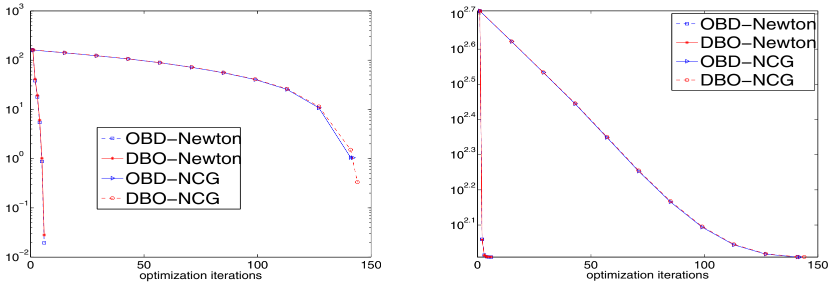


Fig. 2. The norm of the gradient and minimum value of the cost functional are shown on left and right respectively for $T = 4$ msec of simulation time.

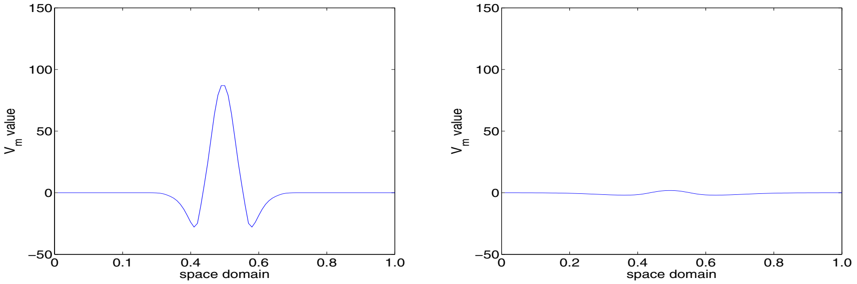


Fig. 3. The optimal state solution of V_m at time $t = 0.04$ msec and $t = 1.80$ msec for $T = 4$ msec of simulation time.

opt.iters	$\lvert\lvert\nabla J(V_m, I_e)\rvert\rvert$	$\frac{\lvert\lvert\nabla J(V_m,I_e)\rvert\rvert_{i+1}}{\lvert\lvert\nabla J(V_m,I_e)\rvert\rvert_i}$
1	160.4675668	
2	38.2739193	0.2385
3	17.7594672	0.4640
4	5.4176392	0.3051
5	0.4178937	0.0771
6	0.0064591	0.0155
7	0.0001882	0.0291

Table 1. Optimization iterations, norm of gradient of cost functional and order of convergence for the OBD technique with Newton's algorithm are presented.

5 Conclusions

In this note, two concrete realizations of the OBD and the DBO approaches for optimal control of the action potential in cardiac electrophysiology based on the monodomain equation were discussed and numerical results are presented for a one-D example. For the current problem there is no significant difference for these two techniques. However, there is a significant difference between the NCG and the Newton methods. Due to the strong nonlinearities in the model, it appears to be difficult to observe a second order convergence. In this respect we were more successful to achieve a superlinear convergence for both discretization methods. The results motivate us to continue our investigations for the bidomain model. The computational results, with extracellular control dampening the complete wave propagation of the transmembrane potential, suggest to also strive for more insight into longer time horizons, with complete simulations of several heart beats, and more realistic geometries and finer meshes.

Acknowledgement: The authors gratefully acknowledge the Austrian Science Foundation (FWF) for financial support under SFB 032, "Mathematical Optimization and Applications in Biomedical Sciences".

References

1. P. Colli Franzone, P. Deuflhard, B. Erdmann, J. Lang and L. F. Pavarino, Adaptivity in Space and Time for Reaction-Diffusion Systems in Electrocardiology, *SIAM Journal on Numerical Analysis*, **28(3)**, 942-962, 2006
2. C. S. Henriquez, Simulating the electrical behavior of cardiac tissue using the bidomain model, *Crit. Rev. Biomed. Eng.*, **21**, 1 77, 1993.
3. J. Lang, Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems, *Lecture Notes in Computational Science and Engineering*, **16**, 2001.
4. Ch. Nagaiah, K. Kunisch and G. Plank, Numerical solution for optimal control of the reaction-diffusion equations in cardiac electrophysiology, *to appear in Computational Optimization and Applications* doi:10.1007/s10589-009-9280-3.
5. Ch. Nagaiah and K. Kunisch, Higher order optimization and adaptive numerical solution for optimal control of monodomain equations in cardiac electrophysiology, *Applied Numerical Mathematics*, **accpeted** .
6. B. F. Nielsen, T. S. Ruud, G. T. Lines and A. Tveito, Optimal monodomain approximations of the bidomain equations, *Applied Mathematics and Computation*, **184(2)**, 276-290, 2007.
7. G. Plank, M. Liebmann, R. Weber dos Santos, EJ. Vigmond, G. Haase, Algebraic multigrid preconditioner for the cardiac bidomain model, *IEEE Trans Biomed Eng.*, **54(4)**, 585-596, 2007.
8. R. Plonsey, Bioelectric sources arising in excitable fibers (ALZA lecture), *Ann. Biomed. Eng.*, **16**, 519 546, 1988.
9. M. Potse, B. Dube, J. Richer, A. Vinet and R. M. Gulrajani, A Comparison of Monodomain and Bidomain Reaction-Diffusion Models for Action Potential Propagation in the Human Heart, *IEEE Transactions on Biomedical Engineering*, **53(12)**, 2425-2435, 2006.
10. J. M. Rogers and A. D. McCulloch, A collocation-Galerkin finite element model of cardiac action potential propagation, *IEEE Trans. Biomed. Eng.* , **41**,743-757, 1994.
11. E. J. Vigmond, R. Weber dos Santos, A. J. Prassl, M. Deo, and G. Plank, Solvers for the cardiac bidomain equations, *Prog Biophys Mol Biol*, **96(1-3)**, 3-18, 2008.

Barrier Methods for a Control Problem from Hyperthermia Treatment Planning

Anton Schiela[1] and Martin Weiser[2]

[1] Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany. schiela@zib.de
[2] Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany. weiser@zib.de

Summary. We consider an optimal control problem from hyperthermia treatment planning and its barrier regularization. We derive basic results, which lay the groundwork for the computation of optimal solutions via an interior point path-following method in function space. Further, we report on a numerical implementation of such a method and its performance at an example problem.

1 Hyperthermia Treatment Planning

Regional hyperthermia is a cancer therapy that aims at heating up deeply seated tumors in order to make them more susceptible to an accompanying chemo or radio therapy [12]. We consider a treatment modality where heat is induced by a phased array microwave ring-applicator containing 12 antennas. Each antenna emits a time-harmonic electromagnetic field the amplitude and phase of which can be controlled individually. The linearly superposed field acts as a heat source inside the tissue. We are interested in controlling the resulting stationary heat distribution, which is governed by a semi-linear elliptic partial differential equation, the bio-heat transfer equation (BHTE), see [7]. The aim is to heat up the tumor as much as possible, without damaging healthy tissue. We thus have to impose constraints on the temperature, and mathematically, we have to solve an optimization problem subject to a PDE as equality constraint and pointwise inequality constraints on the state.

We consider an interior point path-following algorithm that has been applied to this problem. In order to treat the state constraints, the inequality constraints are replaced by a sequence of barrier functionals, which turn the inequality constrained problem into a sequence of equality constrained problems. We will show existence of barrier minimizers and derive first and second order optimality conditions, as well as as local existence and differentiability of the path, and local convergence of Newtons method. Our work extends the results of [10], which covers the case of linear PDE constraints, to a problem with a non-linear control-to-state mapping, governed by a semi-linear PDE.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_36, © Springer-Verlag Berlin Heidelberg 2010

1.1 The Bio-Heat Transfer Equation

The stationary bio-heat transfer equation was first introduced in [7] to model the heat-distribution T in human tissue. This partial differential equation is a semi-linear equation of elliptic type, which can be written as $A(T) - B(u) = 0$, where $A(T)$ is a differential operator, applied to the temperature distribution, and $B(u)$ is a source term, which can be influenced by complex antenna parameters $u \in \mathbb{C}^{12}$.

More concretely, we set $v := (T, u)$ and consider the following equation in the weak form on a domain $\Omega \subset \mathbb{R}^3$, which is an individual model of a patient:

$$\begin{aligned}
\langle A(T), \varphi \rangle &:= \int_\Omega \langle \kappa \nabla T, \nabla \varphi \rangle_{\mathbb{R}^3} + w(T)(T - T_0)\varphi \, dx + \int_{\partial\Omega} h(T - T_{out})\varphi \, dS, \\
\langle B(u), \varphi \rangle &:= \int_\Omega \frac{\sigma}{2} \, |E(u)|^2_{\mathbb{C}^3} \, \varphi \, dx \\
\langle c(v), \varphi \rangle &:= \langle A(T) - B(u), \varphi \rangle = 0 \quad \forall \varphi \in C^\infty(\Omega),
\end{aligned}$$

where all coefficients may depend of the spacial variable x, and $E(u) = \sum_{k=1}^{12} E_k u_k$ is the superposition of complex time-harmonic electro-magnetic fields, and u_k are the complex coefficients of the control. Further, κ is the temperature diffusion coefficient, σ is the electric conductivity and $w(T)$ denotes the blood perfusion. By T_0, we denote the temperature of the unheated blood, e.g. 37°C. The domain Ω consists of a number of subdomains Ω_i, corresponding to various types of tissue. All coefficients may vary significantly from tissue type to tissue type. For a more detailed description of the parameters we refer to [2].

Assumption 4 *Assume that $\kappa, \sigma \in L_\infty(\Omega)$ are strictly positive on Ω. Similarly, let $h \in L_\infty(\partial\Omega)$ be strictly positive on $\partial\Omega$. Further, assume that $w(T, x)(T - T_0)$ is strictly monotone, bounded and measurable for bounded T, and twice continuously differentiable in T. Assume also that each electric field E_k is contained in $L_{q_E}(\Omega, \mathbb{C}^3)$ for some $q_E > 3$.*

Remark 1. Our assumptions are chosen in a way that that the temperature distribution inside the body is bounded and continuous, while still covering the case of jumping coefficients due to different tissue properties inside the patient models. Also the assumptions on the regularity of the fields $E_k \in L_{q_E}, q_E > 3$ are necessary for guaranteeing continuity of the temperature distribution (cf. e.g. [4, Thm. 6.6]). For the generic regularity $E_k \in L_2$ this cannot be guaranteed a-priori. In clinical practice, of course, pointwise unbounded temperature profiles do not occur. Overly large intensity peaks are avoided by construction of the applicator. However, it is observed that near tissue boundaries so called *hot spots* occur: small regions, where the temperature is significantly higher than in the surrounding tissue due to singularities in the electro-magnetic fields at tissue boundaries. One of the challenges of optimization is to eliminate these hot spots.

Under these assumption we can fix our functional analytic framework. As usual in state constrained optimal control, we have to impose an $\| \cdot \|_\infty$-topology on the space of temperature distributions. To this end, let q be in the range $q_E > q > 3$, and $q' = q/(q-1)$ its dual exponent. We define $V = C(\overline{\Omega}) \times \mathbb{C}^{12}$ and

$$c : (C(\overline{\Omega}) \supset D_q) \times \mathbb{C}^{12} \to (W^{1,q'})^*,$$

where D_q is the set of all T, such that $A(T) \in (W^{1,q'})^*$, i.e. $\langle A(T), \varphi \rangle \leq M \|\varphi\|_{W^{1,q'}}\ \forall \varphi \in C^\infty(\overline{\Omega})$. By suitable regularity assumptions $D_q = W^{1,q}(\Omega)$, a result, which we will, however, not need.

It is well known (cf. e.g. [11, 4]) that A has a continuous inverse $A^{-1} : (W^{1,q'})^* \to C(\overline{\Omega})$, and even $\|T\|_{C^\beta} \leq c\|A(T)\|_{(W^{1,q'})^*}$ for some $\beta > 0$ locally, where C^β is the space of Hölder continuous functions. Moreover, it is straightforward to show that D_q only depends on the main part of A, and is thus independent of T.

Lemma 1. *The mapping $c(v) : (C(\overline{\Omega}) \supset D_q) \times \mathbb{C}^{12} \to (W^{1,q'}(\Omega))^*$ is twice continuously Fréchet differentiable. Its derivatives are given by*

$$\begin{aligned}
&\langle c'(v)\delta v, \varphi \rangle = \langle A'(T)\delta T - B'(u)\delta u, \varphi \rangle \\
&\langle A'(T)\delta T, \varphi \rangle = \int_\Omega \langle \kappa \nabla \delta T, \nabla \varphi \rangle_{\mathbb{R}^3} + (w'(T)(T-T_0) + w(T))\delta T \varphi \, dx + \int_{\partial\Omega} h \delta T \varphi dS \\
&\langle B'(u)\delta u, \varphi \rangle = \int_\Omega \sigma \mathrm{Re} \left\langle \sum_{k=1}^{12} E_k u_k, \sum_{k=1}^{12} E_k \delta u_k \right\rangle_{\mathbb{C}^3} \varphi \, dx \\
&\langle c''(v)(\delta v)^2, \varphi \rangle = \langle A''(T)(\delta T)^2 - B''(u)(\delta u)^2, \varphi \rangle = \\
&= \int_\Omega (w''(T)(T-T_0) + 2w'(T))\delta T^2 \varphi - \sigma \mathrm{Re} \left\langle \sum_{k=1}^{12} E_k \delta u_k, \sum_{k=1}^{12} E_k \delta u_k \right\rangle_{\mathbb{C}^3} \varphi \, dx.
\end{aligned}$$

Proof. Since all other parts are linear in T, it suffices to show Fréchet differentiability of $T \to w(T,x)(T-T_0)$ and $u \to |E(u,x)|^2$. Since by assumption, $w(T,\cdot) \in C^1(\Omega)$, differentiability of $T \to w(T,x)(T-T_0) : C(\overline{\Omega}) \to L_t(\Omega)$ for every $t < \infty$ follows from standard results of Nemyckii operators (cf. e.g. [3, Prop. IV.1.1], applied to remainder terms). By the dual Sobolev embedding $L_t(\Omega) \hookrightarrow (W^{1,q'}(\Omega))^*$ for sufficiently large t, differentiability of $T \to w(T,x)(T-T_0) : C(\overline{\Omega}) \supset D_q \to (W^{1,q'}(\Omega))^*$ is shown.

Similarly, differentiability of the mapping $u \to |E(u,x)|^2 : \mathbb{C}^{12} \to L_s(\Omega)$ for some $s > 3/2$ follows by the chain rule from the linearity of the mapping $u \to E(u,x) : \mathbb{C}^{12} \to L_{q_E}(\Omega, \mathbb{C}^3)$ and the differentiability of the mapping $w \to |w|^2 : L_{q_E}(\Omega, \mathbb{C}^3) \to L_{q_E/2}(\Omega, \mathbb{C}^3)$ with $q_E/2 = s > 3/2$. Again, by the dual Sobolev embedding $L_s(\Omega) \hookrightarrow (W^{1,q'}(\Omega))^*$ we obtain the desired result.

Similarly, one can discuss the second derivatives. We note that $(|E(u,x)|^2)'$ is linear in u, and thus it coincides with its linearization.

Remark 2. Note that $A' : C(\overline{\Omega}) \supset D_q \to (W^{1,q'}(\Omega))^*$ is not a continuous linear operator, but since it has a continuous inverse, it is a closed operator. Moreover, since the main part of A is linear, $A'(T) - A'(\tilde{T})$ contains no differential operator. Hence $\|\tilde{T} - T\|_\infty \to 0$ implies $\|A'(T) - A'(\tilde{T})\|_{C(\overline{\Omega}) \to (W^{1,q'})^*} \to 0$. These facts allow us to apply results, such as the open mapping theorem and the inverse function theorem to A.

Lemma 2. *For each $v \in D_q \times \mathbb{C}^{12}$ the linearization*

$$c'(v) = A'(T) - B'(u) : D_q \times \mathbb{C}^{12} \to (W^{1,q'}(\Omega))^*$$

is surjective and has a finite dimensional kernel.

For each v with $c(v) = 0$ there is a neighborhood $U(v)$ and a local diffeomorphism

$$\psi_v : \ker c'(v) \leftrightarrow U(v) \cap \{v : c(v) = 0\},$$

satisfying $\psi_v'(0) = Id$ and $c'(v)\psi_v''(0) = -c''(v)$.

Proof. It follows from the results in [4] that $A'(T)$ has a continuous inverse $A'(T)^{-1} : (W^{1,q'}(\Omega))^* \to C(\overline{\Omega})$. Since A' is bijective, also $c'(v) = (A'(T), -B'(u))$ is surjective, and each element $\delta v = (\delta T, \delta u)$ of $\ker c'$ can be written in the form $(A'(T)^{-1}B'(u)\delta u, \delta u)$. Since $\delta u \in \mathbb{C}^{12}$, $\ker c'(v)$ is finite dimensional. Via the inverse function theorem we can now conclude local continuous invertibility of A, and also that A^{-1} is twice differentiable.

Let $(\delta T, \delta u) = \delta v \in \ker c'(v)$. Then we define

$$\psi_v(\delta v) := \begin{pmatrix} (A^{-1} \circ B)(u + \delta u) \\ u + \delta u \end{pmatrix}$$

and compute

$$\begin{aligned}
(A^{-1} \circ B)'(v)\delta u &= A'(T)^{-1}B'(u)\delta u = \delta T \\
(A^{-1} \circ B)''(v)(\delta u)^2 &= -A'(T)^{-1}A''(T)A'(T)^{-1}(B'(u)\delta u)^2 + A'(T)^{-1}B''(u)(\delta u)^2 \\
&= -A'(T)^{-1}\left(A''(T)(\delta T)^2 - B''(u)(\delta u)^2\right).
\end{aligned}$$

It follows

$$\begin{aligned}
\psi_v'(0)\delta v &= (\delta T, \delta u) = \delta v \\
c'(v)\psi_v''(0)(\delta v)^2 &= (A'(T), -B'(u))\psi_v''(0)(\delta v)^2 \\
&= -(A''(T)(\delta T)^2 - B''(u)(\delta u)^2) = -c''(v)(\delta v)^2.
\end{aligned}$$

1.2 Inequality constraints and objective

As for inequality constraints, we impose upper bounds on the amplitudes of the controls to model the limited power of the microwave applicator:

$$|u_k| \le u_{\max}, \quad k = 1 \ldots 12.$$

Moreover, crucially, we impose upper bounds on the temperature inside the healthy tissue. These are state constraints, which pose significant practical and theoretical difficulties. These constraints are necessary to avoid excessive heating of healthy tissue, which would result in injuries of the patient. We have

$$T \le T_{\max}(x),$$

where $T_{\max}$ is chosen as a piecewise constant function on each tissue type, depending on the sensitivity of the tissue with respect to heat.

Algorithmically, we treat the inequality constrained optimization problem in function space by a barrier approach (cf. [10]) and replace the inequality constraints by a sequence of barrier functionals, depending on a parameter μ (setting again $v = (T, u)$):

$$b(v;\mu) = \int_\Omega l(T_{\max} - T;\mu)\,dx - \mu \sum_{i=1}^{12} \ln(u_{\max} - |u_k|)$$

here l may be a sum of logarithmic and rational barrier functionals:

$$\begin{aligned} &l_k(\cdot;\mu) : \mathbb{R}_+ \to \overline{\mathbb{R}} := \mathbb{R} \cup \{+\infty\} \\ &\qquad l_1(t;\mu) := -\mu \ln(t), \qquad l_k(t;\mu) := \mu^k/((k-1)t^{k-1}) \ (k > 1) \end{aligned}$$

A straightforward computation shows that $b(v;\mu)$ is a convex function (as a composition of convex and convex, monotone functions), and it is also clear that for strictly feasible v, $b : C(\overline{\Omega}) \times \mathbb{C}^{12}$ is twice continuously differentiable near v, and thus locally Lipschitz continuous there. It has been shown in [10] that b is also lower semi-continuous.

Finally, we consider an objective functional $J : C(\overline{\Omega}) \times \mathbb{C}^{12} \to \mathbb{R}$, which we assume to be twice continuously differentiable, and thus locally Lipschitz continuous. For our numerical experiments, below, we will choose a simple objective of the form $J(v) = \|T - T_{des}\|_{L_2}^2$ (recall that the control is finite dimensional), but more sophisticated functionals are under consideration, which more directly model the damage caused in the tumor.

Summarizing, we can write down regularized optimal control problem:

$$\min_{v \in V} J_\mu(v) := J(v) + b(v;\mu) \text{ s.t. } c(v) = 0. \tag{1}$$

2 Barrier Minimizers and their Optimality Conditions

Next we study existence and basic properties of solutions of the barrier problems. For this purpose we impose the assumption that there is at least one strictly feasible solution. This is fulfilled, for example by $u = 0$, if the upper bounds $T_{\max}$ are chosen reasonably.

Theorem 1. *For every $\mu > 0$ the barrier problem* (1) *has an optimal solution, which is strictly feasible with respect to the inequality constraints.*

Proof. Since the set of feasible controls is finite dimensional, closed, and bounded and by our assumptions the control-to-space mapping $u \to T$ is continuous (cf. e.g. [4, Thm. 6.6] and the discussion after that theorem), the set of all feasible pairs (T, u) is compact in $C(\overline{\Omega}) \times \mathbb{C}^{12}$. By assumption, there is at least one strictly feasible solution, for which $J + b$ takes a finite value. Hence, existence of an optimal solution follows immediately from the Theorem of Weierstraß (its generalization for lower semi-continuous functions).

Since all solutions of our PDE are Hölder continuous, strict feasibility for sufficiently high order of the barrier functional follows from [10, Lemma 7.1].

Lemma 3. *If v_μ is a locally optimal solution of* (1)*, then $\delta v = 0$ is a minimizer of the following convex problem:*

$$\min_{\delta v} J'(v_\mu)\delta v + b(v_\mu + \delta v; \mu) \;\; s.t. \;\; c'(v_\mu)\delta v = 0 \tag{2}$$

Proof. For given, $\delta v \in \ker c'(v_\mu)$, and $t > 0$ let $\tilde{v} = v_\mu + t\delta v$. By Lemma 2 there are $\hat{v} = \psi_{v_\mu}(\delta v)$, such that $c(\hat{v}) = 0$ and $\hat{v} - \tilde{v} = o(t)$. Further, by strict feasibility of v_μ, $J + b$ is locally Lipschitz continuous near v_μ with Lipschitz constant L_{J+b}. We compute

$$\begin{aligned} J'(v_\mu)(t\delta v) + b'(v_\mu;\mu)(t\delta v) &= (J+b)(\tilde{v};\mu) - (J+b)(v_\mu;\mu) + o(t) \\ &= (J+b)(\hat{v};\mu) - (J+b)(v_\mu;\mu) + (J+b)(\tilde{v};\mu) - (J+b)(\hat{v};\mu) + o(t) \\ &\geq 0 + L_{J+b}o(t) + o(t). \end{aligned}$$

it follows $J'(v_\mu)\delta v + b'(v_\mu;\mu)\delta v \geq 0$, and by linearity $J'(v_\mu)\delta v + b'(v_\mu;\mu)\delta v = 0$. By convexity of b we have $b'(v_\mu;\mu)\delta v \leq b(v_\mu + \delta v;\mu) - b(v_\mu;\mu)$ and thus

$$J'(v_\mu)\delta v + b(v_\mu + \delta v;\mu) - b(v_\mu;\mu) \geq 0$$

which proofs our assertion.

Theorem 2. *If v_μ is a locally optimal solution of* (1)*, then there exists a unique $p \in H^1(\Omega)$, such that*

$$0 = F(v, p; \mu) := \begin{cases} J'_\mu(v_\mu) + c'(v_\mu)^* p, \\ c(v_\mu). \end{cases} \tag{3}$$

Proof. Clearly, the second row of (3) holds by feasibility of v_μ. By Lemma 3 $\delta v = 0$ is a minimizer of the convex program (2). Hence, we can apply [10, Thm. 5.4] to obtain first order optimality conditions for this barrier problem with $p \in W^{1,p'}(\Omega)$. Taking into account strict feasibility of v_μ with respect to the inequality constraints, all elements of subdifferentials in [10, Thm. 5.4] can be replaced by Fréchet derivatives, so (3) follows. In particular, p satisifies the adjoint equation $\partial_y J_\mu(v_\mu) + A'(T)^* p = 0$, which can be interpreted as a PDE in variational form with $\partial_y J_\mu(v_\mu) \in L_\infty(\Omega)$, and thus $p \in H^1(\Omega)$ follows.

Before we turn to second order conditions we perform a realification of the complex vector $u \in \mathbb{C}^{12}$. Since $|E(u,x)|$ only depends on the the relative phase shifts of the antenna parameters, optimal controls of our problem are non-unique. This difficulty can be overcome easily by fixing $\mathrm{Im}(u_1) = 0$. After that, realification $(x + iy \to (x, y))$ yields a new control vector $u \in \mathbb{R}^{23}$ (dropping the component that corresponds to $\mathrm{Im}(u_1)$), which we will use in the following. We define the Hessian of the Lagrangian $H(v;p)$ by

$$H(v,p)\delta v^2 = J''_\mu(v)\delta v^2 + \langle p, c''(v)\delta v^2\rangle$$

Theorem 3. *Let* (v_μ, p_μ) *be a solution of* (3). *Then,*

$$\frac{1}{2}H(v_\mu,p_\mu)\delta v^2 = J_\mu(\psi_{v_\mu}(\delta v)) - J_\mu(v_\mu) + o(\|\delta v\|^2). \tag{4}$$

(i) $H(v_\mu, p_\mu)$ *is positive semi-definite on* $\ker c'(v_\mu)$, *if* v_μ *is a local minimizer of* (1).

(ii)$H(v_\mu; p_\mu)$ *is positive definite on* $\ker c'(v_\mu)$, *if and only if* v_μ *is a local minimizer of* (1) *and* J_μ *satisfies a local quadratic growth condition.*

Then for each $(r_1, r_2) \in ((H^1(\Omega))^* \times \mathbb{R}^{23}) \times (W^{1,q'}(\Omega))^*$ *the linear system*

$$\begin{pmatrix} H(v_\mu,p_\mu) & c'(v_\mu)^* \\ c'(v_\mu) & 0 \end{pmatrix} \begin{pmatrix} \delta v \\ \delta p \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 \end{pmatrix} \tag{5}$$

has a unique solution $(\delta v, \delta p) \in V \times H^1(\Omega)$, *depending continuously on* (r_1, r_2).

Proof. Let $\delta v \in \ker c'(v_\mu)$, and ψ_{v_μ} be defined as in Lemma 2. We show (4) by Taylor expansion:

$$\begin{aligned} J_\mu(\psi_{v_\mu}(\delta v)) - J_\mu(v_\mu) =& J'_\mu(v_\mu)\psi'_{v_\mu}(0)\delta v \\ &+ 0.5\left(J''_\mu(v_\mu)(\psi'_{v_\mu}(0)\delta v)^2 + J'_\mu(v_\mu)\psi''_{v_\mu}(0)(\delta v)^2\right) + o(\|\delta v\|^2). \end{aligned} \tag{6}$$

Since $J'_\mu(v_\mu)\delta v = 0\,\forall \delta v \in \ker c'(v_\mu)$, $\psi'_{v_\mu}(0) = Id$, it follows $J'_\mu(v_\mu)\psi'_{v_\mu}(0)\delta v = 0$. Further, by $J'_\mu(v_\mu)\delta v + \langle p_\mu, c'(v_\mu)\delta v\rangle = 0\,\forall \delta v \in V$ and $c'(v_\mu)\psi''_{v_\mu}(0) = -c''(v_\mu)$ we deduce

$$J'_\mu(v_\mu)\psi''_{v_\mu}(0)(\delta v)^2 = -\langle p_\mu, c'(v_\mu)\psi''_{v_\mu}(0)(\delta v)^2\rangle = \langle p_\mu, c''(v_\mu)(\delta v)^2\rangle.$$

Inserting these two results into (6) yields (4).

All other assertions, except for solvability of (5) then follow directly, using the fact that $|\|\delta v\| - \|\psi_{v_\mu}(\delta v) - v_\mu\|| \le \|v_\mu + \delta v - \psi_{v_\mu}(\delta v)\| = o(\|\delta v\|)$.

Let us turn to (5). If $H(v_\mu; p_\mu)$ is positive definite on $\ker c'(v_\mu)$ (which is finite dimensional), then the minimization problem

$$\min_{c'(v_\mu)\delta v = r_2} -\langle r_1, \delta v\rangle + H(v_\mu; p_\mu)\delta v^2$$

is strictly convex and has a unique solution δv. The first order optimality conditions for this problem yield solvability of the system (5) at (v_μ, p_μ). Since we have assumed $r_1 \in (H^1)^* \times \mathbb{R}^{23}$ and $A'(T_\mu)^* : H^1 \to H^{-1}$ is an isomorphism, we obtain $\delta p \in H^1$. Thus, the matrix in (5) is surjective, and we may deduce its continuous invertibility by the open mapping theorem.

Corollary 1. *If $H(v_\mu, p_\mu)$ is positive definite on $\ker c'(v_\mu)$, then, locally, there is a differentiable path $\mu \to z_\mu$ of local minimizers of the barrier problems, defined in some open interval $]\overline{\mu}, \underline{\mu}[\supset \mu$. Further, Newton's method, applied to $F(v, p; \mu)$ converges locally superlinearly to (v_μ, p_μ).*

Proof. We note that $F(v, p; \mu)$ is differentiable w.r.t. μ, and w.r.t. (v, p). Since $F' = dF/d(v, p)$, given by (5) is continuously invertible, local existence and differentiability follows from the implicit function theorem. Since $F'(v, p; \mu)$ depends continuously on (v, p), we can use a standard local convergence result for Newton's method (cf. e.g. [6, Thm. 10.2.2]).

Remark 3. Since all these results depend on the positive definiteness of H, we cannot expect to obtain global convergence results for barrier homotopy paths. From a global point of view, several branches may exist, and if H is only positive semi-definite at a point of one such branch, it may cease to exist or bifurcate. As a consequence, a local Newton path-following scheme should be augmented by a globalization scheme. for non-convex optimization in the spirit of trust-region methods. This is subject to current reasearch.

3 Numerical results

For the optimization of the antenna parameters we use an interior point path-following method, applying Newton's method to the system (3). As barrier functional we use the sum of rational barrier functionals, and the reduction of the barrier parameter is chosen adaptively in the spirit of [1, Chapt. 5] by an affine covariant estimation of the non-linearity of the barrier subproblems. Further, Newton's method is augmented by a pointwise damping step. A more detailed description of this algorithm can be found in [9]. This algorithm can be applied safely in a neighborhood of the barrier homotopy path, as long as positive definiteness of $H(v_\mu, p_\mu)$ holds. In practice, this works well, as long as a reasonable starting guess is available for the antenna parameters. Just as predicted by the theory in the convex case (cf. [10]) the error in the function value decreases linearly with μ (cf. Figure 1, right).

The discretization of the Newton steps was performed via linear finite element spaces X_h for T and p (cf. [5]). Discretization and assembly were performed with the library Kaskade7. In view of Newton's method this gives rise to the following block matrix, which has to be factorized at each Newton step:

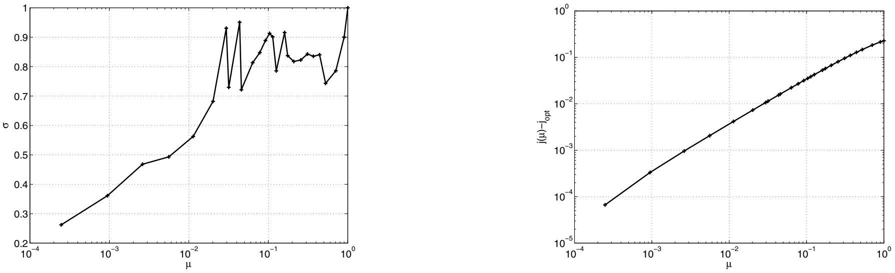


Fig. 1. Left: μ-reduction factors $\sigma_k = \mu_{k+1}/\mu_k$. Right: error in functional values.

$$F'(v, p; \mu) = \begin{pmatrix} H_1(T, p; \mu) & 0 & A'(T)^* \\ 0 & H_2(u, p; \mu) & B'(u)^* \\ A'(T) & B'(u) & 0 \end{pmatrix},$$

where

$$\begin{aligned} H_1(T, p; \mu)(v, w) &= J''(T)(v, w) + b''(T; \mu)(v, w) + \langle p, A''(T)(v, w)\rangle_{L_2(\Omega)} \\ H_2(u, p; \mu)(v, w) &= b''(u; \mu)(v, w) + \langle p, B''(u)(v, w)\rangle_{L_2(\Omega)}. \end{aligned}$$

Note that $H_2 : \mathbb{R}^{23} \to \mathbb{R}^{23}$, and $B' : \mathbb{R}^{23} \to X_h^*$ are dense matrices, while $A', H_1 : X_h \to X_h^*$ are sparse. The factorization of this matrix is performed via building a Schur complement for the $(2, 2)$-block, so that essentially only a sparse factorization of A' and a couple of back-solves have to be performed via a direct sparse solver. As an alternative one can use an iterative solver, preconditioned by incomplete factorizations as proposed in [8].

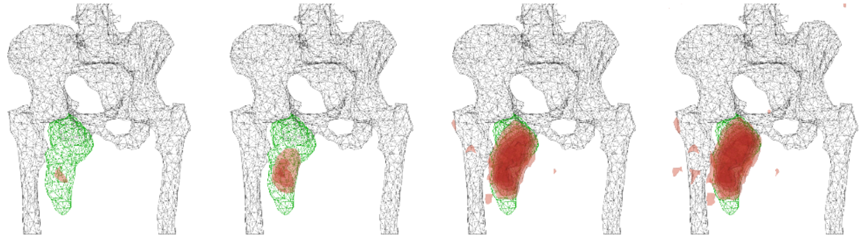


Fig. 2. Heat distribution inside body for $\mu = 1.0, 0.7, 0.1, 10^{-4}$ (left to right).

Let us consider the development of the stationary heat distribution during the algorithm in Figure 2. We observe the effect of the barrier regularization. The algorithm starts with a very conservative choice of antenna parameters, an tends to a more and more aggressive configuration, as μ decreases. This may be of practical value for clinicians. Further, it is interesting to observe that already at a relatively large value of $\mu = 0.1$, we are rather close to the optimal solution. This is reflected by the choice of steps (cf. Figure 1).

4 Conclusion and Outlook

In this work basic results in function space for barrier methods applied to a hyperthermia planning problem with state constraints were established. The theory extends known results from the convex case. While the set of assumptions is taylored for hyperthermia, it is clear that the theory also applies to a wider class of optimal control problems, as long as appropriate regularity results for the involved differential equation are at hand. Subject of current research is the extension of our algorithm by a globalization scheme in the spirit of non-linear programming, in order to increase its robustness in the presence of non-convexity.

Acknowledgment

This work was supported by the DFG Research Center MATHEON "Mathematics for key technologies".

References

1. P. Deuflhard. *Newton Methods for Nonlinear Problems. Affine Invariance and Adaptive Algorithms*, volume 35 of *Series Computational Mathematics*. Springer, 2004.
2. P. Deuflhard, M. Weiser, and M. Seebaß. A new nonlinear elliptic multilevel fem applied to regional hyperthermia. *Comput. Vis. Sci.*, 3(3):115–120, 2000.
3. I. Ekeland and R. Témam. *Convex Analysis and Variational Problems.* Number 28 in Classics in Applied Mathematics. SIAM, 1999.
4. R. Haller-Dintelmann, J. Rehberg, C. Meyer, and A. Schiela. Hölder continuity and optimal control for nonsmooth elliptic problems. *Appl. Math. and Optimization*, 60(3):397–428, 2009.
5. M. Hinze and A. Schiela. Discretization of interior point methods for state constrained elliptic optimal control problems: Optimal error estimates and parameter adjustment. *Comput. Optim. Appl.*, published online, 2009.
6. J.M. Ortega and W.C. Rheinboldt. *Iterative solution of nonlinear equations in several Variables.* Academic Press, 1970.
7. H.H. Pennes. Analysis of tissue and arterial blood temperatures in the resting human forearm. *J. Appl. Phys.*, 1:93–122, 1948.
8. O. Schenk, A. Wächter, and M. Weiser. Inertia revealing preconditioning for large-scale nonconvex constrained optimization. *SIAM J. Sci. Comput.*, 31(2):939–960, 2008.
9. A. Schiela. An interior point method in function space for the efficient solution of state constrained optimal control problems. ZIB Report 07-44, Zuse Institute Berlin, 2008.
10. A. Schiela. Barrier methods for optimal control problems with state constraints. *SIAM J. Optim.*, 20(2):1002–1031, 2009.
11. F. Tröltzsch. *Optimale Steuerung partieller Differentialgleichungen. Theorie, Verfahren und Anwendungen.* Vieweg, 2005.
12. A. Wust, B. Hildebrandt, G. Sreenivasa, B. Rau, J. Gellermann, H. Riess, R. Felix, and P.M. Schlag. Hyperthermia in a combined treatment of cancer. *The Lancet Oncology*, 3:487–497, 2002.

On a State-Constrained PDE Optimal Control Problem arising from ODE-PDE Optimal Control

S. Wendl, H. J. Pesch, and A. Rund

Chair of Mathematics in Engineering Sciences, University of Bayreuth, Germany
stefan.wendl@uni-bayreuth.de, hans-josef.pesch@uni-bayreuth.de,
armin.rund@uni-bayreuth.de

Summary. The subject of this paper is an optimal control problem with ODE as well as PDE constraints. As it was inspired, on the one hand, by a recently investigated flight path optimization problem of a hypersonic aircraft and, on the other hand, by the so called "rocket car on a rail track"-problem from the pioneering days of ODE optimal control, we would like to call it "hypersonic rocket car problem". While it features essentially the same ODE-PDE coupling structure as the aircraft problem, the rocket car problem's level of complexity is significantly reduced. Due to this fact it is possible to obtain more easily interpretable results such as an insight into the structure of the active set and the regularity of the adjoints. Therefore, the rocket car problem can be seen as a prototype of an ODE-PDE optimal control problem. The main objective of this paper is the derivation of first order necessary optimality conditions.

Key words: Optimal control of partial differential equations, ODE-PDE-constrained optimization, state constraints

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_37, © Springer-Verlag Berlin Heidelberg 2010

1 Introduction

Realistic mathematical models for applications with a scientific or engineering background often have to consider different physical phenomena and therefore may lead to coupled systems of equations that include partial and ordinary differential equations. While each of the fields of optimal control of partial resp. ordinary differential equations has already been subject to thorough research, the optimal control of systems containing both has not been studied theoretically so far to the best knowledge of the authors.

Recently Chudej et. al. [5] and M. Wächter [12] studied an optimal control problem numerically which describes the flight of an aircraft at hypersonic speed under the objective of minimum fuel consumption. The flight trajectory is described, as usual, by a system of ordinary differential equations (ODE). Due to the hypersonic flight conditions aerothermal heating of the aircraft must be taken into account. This leads to a quasi-linear heat equation with non-linear boundary conditions which is coupled with the ODE. As it is the main objective of the optimization to limit the heating of the thermal protection system, one obtains a pointwise state constraint, which couples the PDE with the ODE reversely. However, anything beyond mere numerical analysis is prohibited by the considerable complexity of this problem. Therefore the present paper's focus is a model problem stripped of all unnecessary content while still including the key features of ODE-PDE optimal control, which will allow a clearer view on the structure of the problem and its solution.

This simplified model problem we would like to call the "hypersonic rocket car problem". To one part it consists of the classical "rocket car on a rail track problem" from the early days of ODE control, first studied by Bushaw [3]. The second part is a one dimensional heat equation with a source term depending on the speed of the car, denoting the heating due to friction.

In contrast to [10], which deals with the same ODE-PDE problem but from the ODE point of view, this paper is dedicated to a PDE optimal control approach.

Another even more complicated optimal control problem for partial integro-differential-algebraic equations including also ODEs, which describes the dynamical behaviour of the gas flows, the electro-chemical reactions, and the potential fields inside a certain type of fuel cells, has been investigated in [6], also numerically only. However, this model does not include a state constraint.

2 The hypersonic rocket car problem

In the following, the ODE state variable w denotes the one-dimensional position of the car depending on time t with the terminal time t_f unspecified. The PDE state variable T stands for the temperature and depends on time as well as the spatial coordinate x describing the position within the car. The control u denotes the acceleration of the car. The PDE is controlled only indirectly via the velocity $\dot{w}$ of the car.

The aim is to drive the car in minimal time from a given starting position and speed (w_0 resp. v_0) to the origin of the phase plane while keeping its temperature below a certain threshold $T_{\max}$.

All in all, the hypersonic rocket car problem is given as follows:

$$\min_{u \in U} \left\{ t_f + \frac{1}{2} \lambda \int_0^{t_f} u^2(t)\, \mathrm{d}t \right\}, \quad \lambda > 0\,, \tag{1a}$$

subject to

$$\begin{aligned}
&\ddot{w}(t) = u(t) \quad \text{in } (0, t_f)\,, && \text{(1b)}\\
&w(0) = w_0\,, \quad \dot{w}(0) = v_0\,, && \text{(1c)}\\
&w(t_f) = 0\,, \quad \dot{w}(t_f) = 0\,, && \text{(1d)}\\
&U := \{u \in L^2(0, t_f) \colon |u(t)| \le u_{\max} \text{ almost everywhere in } [0, t_f]\}\,, && \text{(1e)}
\end{aligned}$$

and

$$\begin{aligned}
&\frac{\partial T}{\partial t}(x,t) - \frac{\partial^2 T}{\partial x^2}(x,t) = g(\dot{w}(t)) \text{ in } (0,l) \times (0, t_f)\,, && \text{(1f)}\\
&T(x,0) = T_0(x) \text{ on } (0,l)\,, && \text{(1g)}\\
&-\frac{\partial T}{\partial x}(0,t) = -\big(T(0,t) - T_0(0)\big)\,, && \\
&\frac{\partial T}{\partial x}(l,t) = -\big(T(l,t) - T_0(l)\big) \text{ on } [0, t_f]\,, && \text{(1h)}
\end{aligned}$$

and finally subject to a pointwise state constraint of type

$$T(x,t) \le T_{\max} \text{ in } [0,l] \times [0, t_f]\,. \tag{1i}$$

The initial temperature T_0 of the car is in the following set to zero. In the numerical experiments the regularisation parameter λ is chosen as $\frac{1}{10}$, the length l of the car and the control constraint $u_{\max}$ both as 1, and the source term $g(\dot{w}(t))$ as $\dot{w}(t)^2$, which models the temperature induced by friction according to Stokes' law (proportional to the square of the velocity).

3 The state-unconstrained problem and its associated temperature profile

For better illustration and to alleviate comparison with the numerical results of section 5 let us first have a brief look at the solution of the state unconstrained (i. e. only ODE) problem; see Fig. 1. This figure describes the optimal solutions for all starting values in the w-$\dot{w}$-phase plane converging into the origin. Unlike the non-regularized problem ($\lambda = 0$) with a pure bang-bang switching structure and optimal solutions having at most one switching point when its trajectories cross the switching curve (dotted black), on which the car finally arrives at the origin, the optimal solutions of the regularized problem ($\lambda > 0$) have a transition phase between two bang-bang subarcs. The smaller the regularization parameter λ is the closer the optimal trajectories (grey) approach the switching curve which serves as their envelope here.

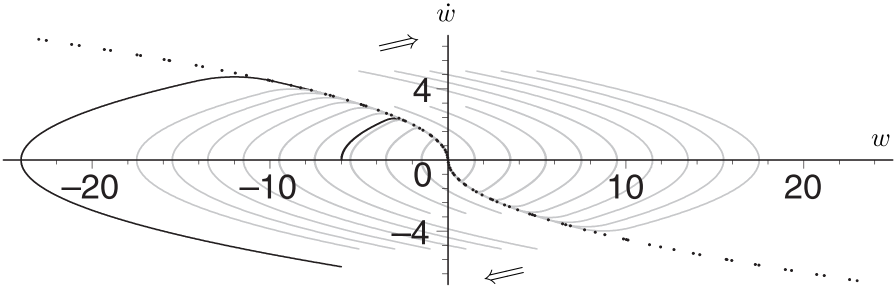


Fig. 1. Optimal trajectories of the regularized minimum-time problem ($\lambda > 0$) in the phase plane (grey). The dotted black curve is the switching curve of the non-regularized problem ($\lambda = 0$). The black curves are the optimal solutions for the starting conditions $w_0 = -6$ and $v_0 = 0$ resp. $w_0 = -6$ and $v_0 = -6$.

Along those two trajectories the following temperature profiles emerge:

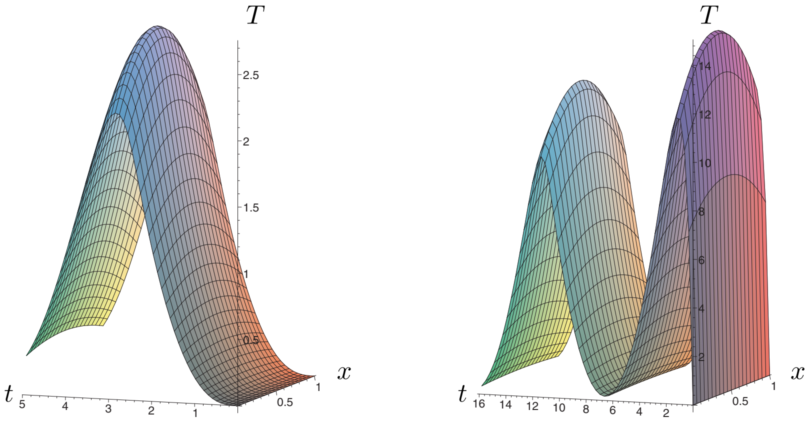


Fig. 2. Temperature profiles along the state-unconstrained trajectories due to the data $w_0 = -6$, $v_0 = 0$ (left), resp. $v_0 = -6$ (right); see Fig. 1.

Those temperature profiles have to be bounded in the following; cp. Fig. 3.

4 Necessary optimality conditions: Interpretation as state-constrained PDE optimal control problem

It is possible to reformulate (1) as a PDE optimal control problem by eliminating the ODE-part:

$$\int_0^{t_f} \left(1 + \frac{\lambda}{2}\, u^2(t)\right) \mathrm{d}t \overset{!}{=} \min_{|u| \le u_{\max}} \tag{2a}$$

subject to

$$T_t(x,t) - T_{xx}(x,t) = \left(v_0 + \textstyle\int_0^t u(s)\,\mathrm{d}s\right)^2 \quad \text{in } (0,l) \times (0,t_f)\,, \tag{2b}$$

$$-T_x(0,t) + T(0,t) = 0\,, \quad T_x(l,t) + T(l,t) = 0 \quad \text{for } 0 < t < t_f\,, \tag{2c}$$

$$T(x,0) = 0 \quad \text{for } 0 \le x \le l\,, \tag{2d}$$

$$\int_0^{t_f} u(t)\,\mathrm{d}t = -v_0\,, \tag{2e}$$

$$\int_0^{t_f}\int_0^t u(s)\,\mathrm{d}s\,\mathrm{d}t = -w_0 - v_0\,t_f \overset{\text{part. int.}}{\Longrightarrow} \int_0^{t_f} t\,u(t)\,\mathrm{d}t = w_0\,, \tag{2f}$$

$$T(x,t) \le T_{\max} \quad \text{in } [0,l] \times [0,t_f]\,. \tag{2g}$$

Here the term $v(t) := v_0 + \int_0^t u(s)\,\mathrm{d}s$ plays the role of a "felt" control for the heat equation. The two isoperimetric conditions (2e, f) are caused by the two terminal conditions (1c) and comprehend the constraints (1b–d) of the ODE part. While this reformulation will alleviate the derivation of first order necessary conditions,it nevertheless comes at a price, namely the nonstandard structure of (2e, f) and especially the source term in (2b). All these terms contain the control under integral signs.

The existence and uniqueness of the solution $T \in W_2^{1,0}((0,l) \times (0,t_f)) \cap C([0,t_f], L^2(0,l))$, the Fréchet-differentiability of the solution operator and the existence of a Lagrange multiplier $\bar{\mu} \in C([0,l] \times [0,t_f])^* = \mathcal{M}([0,l] \times [0,t_f])$ [the set of regular Borel measures on $([0,l] \times [0,t_f])$] under the assumption of a local Slater condition are proven in [8], [9]. Moreover, it turns out, that T is of even higher regularity: T_{tt} and $\partial_x^4 T$ are both of class $L^r(\varepsilon, t_f; L^2(0,l))$ with $0 < \varepsilon < t_f$ and $r \ge 2$ for all controls $u \in L^2(0,t_f)$.

Thereby, we can establish the optimality conditions by means of the Lagrange technique. Furthermore it can be seen that for any given point of time [and for every control $u \in L^2(0,t_f)$] the maximum of T with respect to space is obtained right in the middle at $x = \frac{l}{2}$ (cf. Fig. 2; for a proof see [8]). This implies, that the active set $\mathcal{A}$ is a subset of the line $L := \{x = \frac{l}{2},\ 0 < t < t_f\}$.

Hence the state constraint can equivalently be replaced by $T \leq T_{\max}$ on L. Using this we define the Lagrange-function by

$$\begin{aligned}\mathcal{L} = &\int_0^{t_f} \Big(1 + \frac{\lambda}{2}\,u^2(t)\Big)\,\mathrm{d}t - \int_0^{t_f}\!\!\int_0^l \Big(T_t - T_{xx} - g\big(v_0 + \textstyle\int_0^t u(s)\,\mathrm{d}s\big)\Big)\,q\,\mathrm{d}x\,\mathrm{d}t \\ &- \int_0^{t_f} \big(-T_x(0,t) + T(0,t)\big)\,q(0,t)\,\mathrm{d}t - \int_0^{t_f} \big(T_x(l,t) + T(l,t)\big)\,q(l,t)\,\mathrm{d}t \\ &+\nu_1 \left(\int_0^{t_f} u(t)\,\mathrm{d}t + v_0\right) + \nu_2 \left(\int_0^{t_f} t\,u(t)\,\mathrm{d}t - w_0\right) \\ &+ \int_0^{t_f} \Big(T(\frac{l}{2},t) - T_{\max}\Big)\,\mathrm{d}\mu(t)\,, \end{aligned} \tag{3}$$

with $\mu(t) \in \mathcal{M}(0,t_f)$ and the multipliers q associated with the constraints (2b–c) respectively $\nu_1\,,\nu_2 \in \mathbb{R}$ associated with (2e, f).

By partial integration and differentiation of (3) we find the necessary conditions of first order ($*$ shall in the following denote optimal values):

Adjoint equation:

$$\begin{aligned}\int_0^{t_f^*}\!\!\int_0^l q_t\,\psi - q_x\psi_x\,\mathrm{d}x\,\mathrm{d}t - \int_0^{t_f^*} q(0,t)\,\psi(0,t)\,\mathrm{d}t - \int_0^{t_f^*} q(l,t)\,\psi(l,t)\,\mathrm{d}t \\ + \int_0^{t_f} \psi(\frac{l}{2},t)\,\mathrm{d}\mu(t) = 0 \quad \text{for all } \psi \in W(0,t_f^*)\,,\end{aligned} \tag{4a}$$

$$q(x,t_f^*) = 0 \quad \text{for almost all } x \in [0,l]\,, \tag{4b}$$

Variational inequality:

$$\begin{aligned}&\int_0^{t_f^*} \big(\lambda u^*(t) + \nu_1 + \nu_2\,t\big)\,\big(u(t) - u^*(t)\big)\,\mathrm{d}t \\ &+ \int_0^{t_f^*} g'\big(v_0 + \textstyle\int_0^t u^*(r)\,\mathrm{d}r\big) \left(\displaystyle\int_0^t u(s) - u^*(s)\,\mathrm{d}s\right)\left(\int_0^l q(x,t)\,\mathrm{d}x\right)\mathrm{d}t \geq 0 \\ &\overset{\text{Fubini}}{\Longrightarrow} \int_0^{t_f^*}\left[\lambda u^*(t) + \nu_1 + \nu_2\,t + \int_t^{t_f^*} g'\big(v_0 + \textstyle\int_0^s u^*(r)\,\mathrm{d}r\big)\left(\displaystyle\int_0^l q(x,s)\,\mathrm{d}x\right)\mathrm{d}s\right] \cdot \\ &\qquad \big(u(t) - u^*(t)\big)\,\mathrm{d}t \geq 0\,, \quad \text{for all } u \in U\,,\end{aligned} \tag{4c}$$

Complementarity condition:

$$\mu \geq 0\,, \quad \int_0^{t_f^*} \Big(T^*(\frac{l}{2},t) - T_{\max}\Big)\,\mathrm{d}\mu(t) = 0\,. \tag{4d}$$

The optimality system is completed by a condition for the free terminal time t_f^* and two conditions that give the switching times t_{on}^*, t_{off}^* [i. e. the times where the temperature $T^*(\frac{l}{2}, t)$ hits, resp. leaves the constraint $T_{\max}$, cf. Fig. 3 (right)]. As the derivation of these condition would exceed the scope of this paper they will be published in subsequent papers [8] and [9].

Equations (4a, b) represent the weak formulation of the adjoint equation, which is retrograde in time, and can be formally understood as

$$-q_t(x,t) - q_{xx}(x,t) = \mu(t)\,\delta(x - \frac{l}{2}) \text{ in } (0,l) \times (0,t_f^*)\,, \tag{5a}$$

$$-q_x(0,t) = -q(0,t)\,,\ q_x(l,t) = -q(l,t) \text{ on } [0,t_f^*] \quad \text{and}$$

$$q(x,t_f^*) = 0 \text{ on } [0,l]\,. \tag{5b}$$

Since the adjoints can be interpreted as shadow prices, the line $\{\frac{l}{2}\} \times (t_{\text{on}}^*, t_{\text{off}}^*)$ indicates from where the temperature exerts an influence on the objective functional. This result corresponds to the structure of the solution of the initial-boundary value problem to be expected from (4a, b), in particular $q(x,t) \equiv 0$ for $t_{\text{off}}^* \le t \le t_f^*$; cf. Fig. 5.

A key condition is the optimality condition (4c) which determines the optimal control. It is a complicated integro-variational inequality with a kernel depending on all values of u^* on the interval $[0, t_f^*]$, forward in time, as well as on all values of q on $[t, t_f^*]$, backward in time. Instead (4c), we can determine the optimal control by an integro-projection formula,

$$u^*(t) = P_{[-u_{\max}, u_{\max}]}\left\{ -\frac{1}{\lambda}\left[\nu_1 + \nu_2\, t + \int_t^{t_f^*} g'(v^*(s)) \left(\int_0^l q(x,s)\,\mathrm{d}x \right) \mathrm{d}s \right] \right\}. \tag{6}$$

Comparing this result with the analogous projection formula of [10] it turns out that the second factor [in squared brackets] is just the adjoint velocity $p_{\dot{w}}(t)$ of the equivalent ODE optimal control formulation with the PDE eliminated analytically by a Fourier-type series. This formulation however is also of non-standard form (with a non-local state constraint leading to boundary value problems for systems of complicated integro-ODEs); see [10].

5 Numerical results

The numerical calculations were conducted with the interior point solver IPOPT [7], [11] by A. Wächter and Biegler in combination with the modelling software AMPL [1], with the latter featuring automatic differentiation. This first-discretize-then-optimize (direct) approach was chosen, because even the ostensibly simple and handsome problem (1) proves to be a "redoubtable opponent" for a first-optimize-then-discretize (indirect) method.

After a time transformation $\tau := \frac{t}{t_f}$ to a problem with fixed terminal time (at the cost of spawning an additional optimization variable t_f), applying a simple quadrature formula[1] to (1a), discretizing the ODE with the implicit midpoint rule and the PDE with the Crank-Nicolson scheme, one obtains a nonlinear program to be solved with IPOPT.

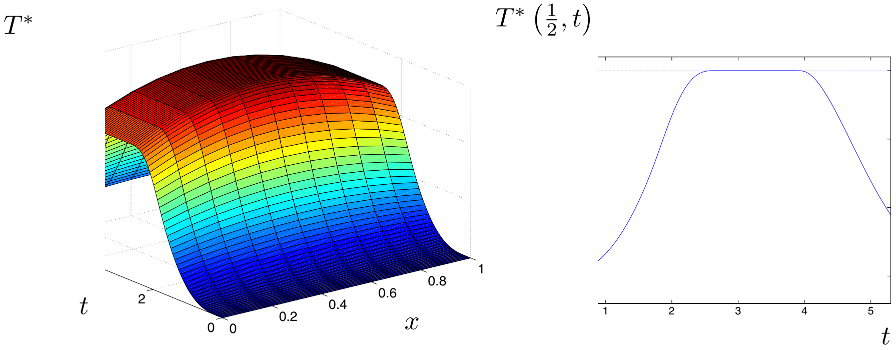


Fig. 3. Temperature $T^*(x,t)$ (left) and cross-section $T^*\left(\frac{1}{2}, t\right)$ (right) along the state-constrained trajectory due to the data $w_0 = -6$, $v_0 = 0$, and $T_{\max} = 1.5$, cf. Figs. 1 and 2 (left).

The approximation of the optimal temperature is shown in Fig. 3. The set of the active state constraint, the line segment $\mathcal{A} = \{\frac{l}{2}\} \times [t^*_{\text{on}}, t^*_{\text{off}}]$, can clearly be seen. The computations used a space-time discretization of 100 by 1000 grid points yielding $t^*_f = 5.35596$, overall objective functional value of 5.51994, $t^*_{\text{on}} = 2.53$ and $t^*_{\text{off}} = 3.96$.

Figure 4 shows the approximations of the optimal control (solid) and the adjoint velocity $p_{\dot{w}}$ (dashed) from the ODE optimal control problem investigated in [10] and also obtained by IPOPT.[2] The perfect coincidence with the projection formula (6) becomes apparent; note the remark to (6).

Figure 5 depicts the approximation of the discrete adjoint temperature yielded by IPOPT[2]. With a closer look at q one can observe a jump discontinuity of its derivative in spatial direction along the relative interior of $\mathcal{A}$. This corresponds to the known jump conditions for adjoints on interior line segments in state-constrained elliptic optimal control [2]. Furthermore one can notice two Dirac measures as parts of the multiplier μ at the entry and exit points of $\mathcal{A}$ in analogy to the behaviour of isolated active points [4]. On the

[1] a linear combination of the trapezoidal sum and the trapezoidal rule with equal weights 1 which indeed approximates a multiple of the integral (2a), but avoids any oscillations of the control.

[2] Note that IPOPT delivers estimates for the adjoint variables with opposite sign compared to our notation.

other hand the multiplier μ contains a smooth part in the relative interior of $\mathcal{A}$ reminiscent of the common behaviour in ODE optimal control.

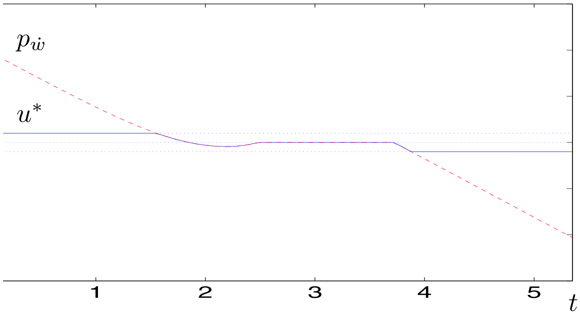


Fig. 4. Optimality check according to the projection formula (6)

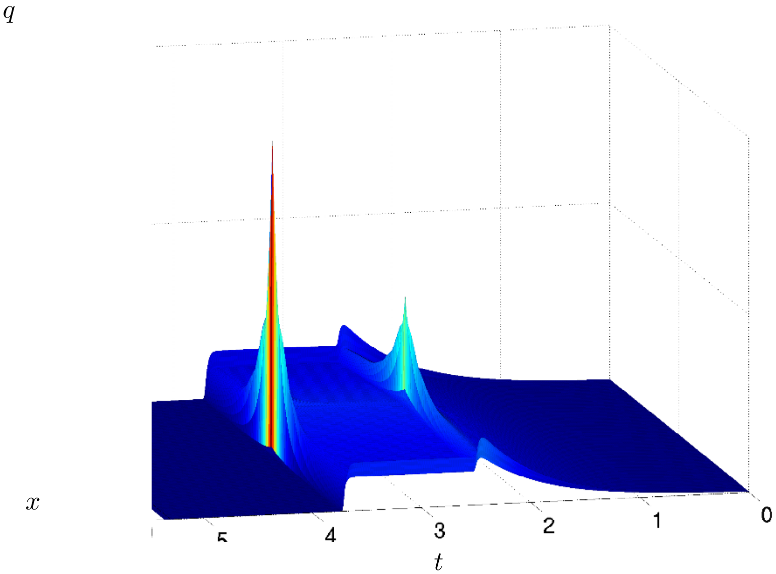


Fig. 5. Adjoint state q of the temperature T.

6 Conclusion

In this paper we studied a prototype of an ODE-PDE optimal control problem. As it is of relatively simple structure, it allows an unobstructed view on its adjoints and optimality system. However an adjoint based method even for such a seemingly simple model problem still remains a formidable task, leaving a direct method as a much more convenient way to go. This of course results in the downside that one has to content oneself with estimates of the continuous

problems' adjoints obtained from the discrete adjoints of the NLP solver used in the *first-discretize-then-optimize* approach.
Transforming the ODE-PDE problem into an only PDE problem, as it has been done in this paper is not the only possibility of tackling it. As it is also viable to transform it into an only ODE problem, which will of course also be pretty nonstandard, an interesting opportunity to compare concepts of ODE and PDE optimal control may arise here such as statements on the topology of active sets. However this is beyond the limited scope of the present paper but can be found in [8], [9].

References

1. The AMPL Company, `http://www.ampl.com/`, 2007.
2. Bergounioux, M., Kunisch, K., *On the Structure of the Lagrange Multiplier for State-constrained Optimal Control Problems*, Systems and Control Letters, Vol. 48, 16–176, 2002.
3. Bushaw, D. W., PhD Thesis, supervised by Solomon Lefschetz, Department of Mathematics, Princeton University, 1952, published as D. W. Bushaw, *Differential Equations with a Discontinuous Forcing Term.* Experimental Towing Tank, Stevens Institute of Technology, Report No. 469, January 1953.
4. Casas, E., *Pontryagin's principle for state-constrained boundary control problems of semilinear elliptic equations*, SIAM J. Control and Otimization, 35:1297-1327, 1997.
5. Chudej, K., Pesch, H. J., Wächter, M., Sachs, G., Le Bras, F., *Instationary Heat Constrained Trajectory Optimization of a Hypersonic Space Vehicle*, to appear in: Proc. of 47th Workshop Variational Analysis and Aerospace Engineering, International School of Mathematics ,,Guido Stampacchia", Erice, Italy, 8.9.–16.9.2007.
6. Chudej, K., Pesch, H.J., Sternberg, K., *Optimal Control of Load Changes for Molten Carbonate Fuel Cell Systems: A Challenge in PDE Constrained Optimization*, SIAM Journal on Applied Mathematics, 70, 2, pp. 621–639, 2009.
7. Laird, C. and Wächter, A., `www.coin-or.org/Ipopt/`; for a documentation including a bibliography see `www.coin-or.org/Ipopt/documentation/`, 2007.
8. Pesch, H. J., Rund, A., von Wahl, W., Wendl, S., *On a Prototype Class of ODE-PDE State-constrained Optimal Control Problems. Part 1: Analysis of the State-unconstrained Problems*, in preparation.
9. Pesch, H. J., Rund, A., von Wahl, W., Wendl, S., *On a Prototype Class of ODE-PDE State-constrained Optimal Control Problems. Part 2: the State-constrained Problems*, in preparation.
10. Pesch, H. J., Rund, A., von Wahl, W., Wendl, S., *On some new phenomena in state-constrained optimal control if ODEs as well as PDEs are involved*, to appear in Control and Cybernetics.
11. Wächter, A., Biegler, L. T., *On the Implementation of an Interior-point Filterline-search Algorithm for Large Scale Nonlinear Programing*, Mathematical Programming, Vol. 106, 25–57, 2006.
12. Wächter, M., *Optimalflugbahnen im Hyperschall unter Berücksichtigung der instationären Aufheizung*, PhD Thesis, Technische Universität München, Faculty of Mechanical Engineering, Munich, Germany, 2004.

Part VII

Engineering Applications of Optimization

Multi-Disciplinary Optimization of an Active Suspension System in the Vehicle Concept Design Stage

Jan Anthonis[1] Marco Gubitosa[1], Stijn Donders[1], Marco Gallo[1], Peter Mas[1], Herman Van der Auweraer[1]

LMS International Interleuvenlaan 68, 3001 Leuven, Belgium
Jan.Anthonis@LMSintl.com

1 Introduction

The automotive industry represents a significant part of the economic activity, in Europe and globally. Common drivers are the improvement of customer satisfaction (performance, personalization, safety, comfort, brand values,) and the adherence to increasingly strict environmental and safety regulations, while at the same time reducing design and manufacturing costs and reducing the time to market. The product evolution is dominated by pushing the envelope on these conflicting demands.

A major evolution currently taking place in this industry is the increase of the electronic and mechatronic content in vehicles. Several studies forecast that the related increase to the vehicle value may well become up to 40 % by 2010 and that up to 80% of the automotive innovation will come from intelligent systems [1], [2], [3], [4]. This of course relates in part to entertainment and telematics systems, but also to the use of many control systems applied to powertrain, chassis and body engineering [5], [6], [7]. One example is the optimization of performance, economy and emissions with engine and transmission controls to realize "green" driving through energy regeneration, automatic start/stop and smart driving control. Another example is the realization of "safe" driving, through the application of ABS and ESP systems for vehicle dynamics control, but also through the adoption of numerous Advanced Driver Assistance Systems (ADAS) such as for lane following, active cruise control, object detection etc. And every vehicle design has ultimately to aim for best customer experience, e.g. by optimizing through control systems ride comfort and handling behaviour and driveability, or by adoption of active systems to control brand sound. This evolution will however not only impact the vehicle product content itself, but also the way vehicle developers (OEM) will cooperate with suppliers in new business models, offering new opportunities for full subsystem responsibility. It will also impact the way the design

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_38, © Springer-Verlag Berlin Heidelberg 2010

and development process itself has to change to enable widespread market introduction in standard vehicles [2], [4], [8], [9]. As a consequence, innovative solutions have lately been introduced for communication and entertainment, engine control and active safety. To a large extent, these innovations however remain on the level of add-on systems and a major need exists to integrate all functionality on the vehicle level through a systems approach. Configuration and performance optimization, system integration, control, component, subsystem and system-level validation of the intelligent systems must be an intrinsic part of the standard vehicle engineering process, just as this is today the case for the structural, vibro-acoustic and kinematic design. This is the goal for Intelligent Vehicle Systems Design Engineering. Such an integrated approach is addressed in this paper. Multiphysics modeling and optimization are key technologies in this process.

2 Engineering Challenges for Intelligent Vehicle Systems

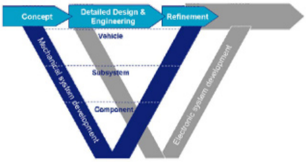

Concept
Detailed Design & Engineering
Refinement
Vehicle
Subsystem
Component
Mechanical system development
Electronic system development

Fig. 1. Double-V process for mechatronic systems.

As Fig. 1 illustrates, the engineering of intelligent systems requires the application of two interconnected "V-shaped" developments: one focusing on the multi-physics system engineering (like the mechanical and electrical components of an electrically powered steering system, including sensors and actuators); and one focused on the controls engineering, the control logic, the software and realization of the control hardware and embedded software. Up to present this process is however very little integrated, with a clearly separated mechanic and electronic design cycle and hence failing to address the need for integrated and maximally frontloaded system modeling. In this paper, the interconnected V-shaped approach is applied on the level of concept modeling, which is situated in the left upper part of the V's in Fig. 1.

The objective of this paper is to perform systems engineering based on multi-physics "plant" models, including the application (and representation of) control. This serves the purpose of configuration design, concept evaluation studies and the optimization of the mechanical system design taking into account the presence of control and certain control laws (or even systems). This will be applied to the design of an active damper.

The methodology to combine detailed mechanic models with control software is not new in automotive. An example related to the simulation and optimization of a Vehicle Dynamics Control system is discussed in [10]. In [11], the application to motorcycle dynamics control is presented. [12], [13] show that the design of an active noise control system for sound quality control uses the same principles. In general in automotive, during the control design cycle of a system, different stages can be distinguished:

1. "Model-in-the-Loop" (MIL): the combination of the multi-physics simulation model with this of the controller, to enable the design of the control logic and the performance engineering of the intelligent system. The simulation is "off-line", i.e. there is no requirement for Real-Time.
2. "Software-in-the-Loop" (SIL): the development and optimization of the "embedded" control software.
3. "Hardware-in-the-Loop" (HIL): the final testing and calibration of the controller software and hardware, requires the controller to be connected to a multi-physics simulation model of the components, subsystems or system, in a dedicated computing environment. This requires real-time capable simulation models.

Although in these 3 stages, models of different design cycles are coupled to the control design cycle, it can hardly be stated that there is an integration. The models are just used to check functionality, proper implementation and final operation of the controller in the vehicle. The controller itself has no impact on the mechanical design. In this paper, the complete mechatronic system on concept level, including a controller model, will impact the mechanical design and hydraulic specification of the component.

3 The optimal design approach

The active damper considered in the investigation hereafter is a hydraulic single rod cylinder with two valves and a pump providing a continuous flow through the cylinder [14]. The final objective is to optimize cylinder and rod diameters, pump flow and characteristics of the valves with respect to energy consumption while meeting the performance criteria defined by the probability distribution of the mission profiles (force velocity couples at the dampers).

More in detail and considering the scheme proposed in Figure 2, the design optimization is performed in two stages:

- " the first stage, addressed to as "Vehicle Model", calculates the optimal parameters of an ideally controlled vehicle with a skyhook based algorithm;
- " the second stage, referred to as "Actuator Model", aims at optimizing the actuator model properties.

The first step in the optimization, represents the conceptual design stage. In this phase, an optimization on the controller parameters is performed in order to meet certain ride & handling and comfort criteria. As this is a conceptual model, instead of implementing the active shock absorbers in the model, the forces computed by the controller are immediately fed into the suspension. In this way, perfect actuator behavior is assumed. The second stage consists of the modeling of the active damper, built with the hydraulic component design library in LMS Imagine.Lab AMESim. As not all damper configurations are suited to meet the performance at concept level, the set of parameters for the shock absorbers is determined that meets this performance. The selection of the feasible configurations is carried out with a full factorial DOE (Design Of Experiment) considering the active deliverable mechanical power as the restrictive constraint. Between those elected combinations, using force-velocity couples weighted in function of the most occurring road profiles, the optimal damper parameters with respect to energy consumption are selected.

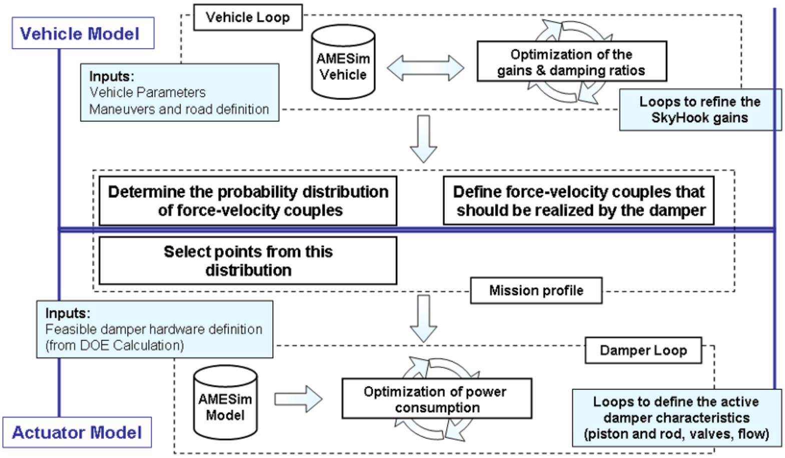

Vehicle Model
Vehicle Loop
Inputs:
Vehicle Parameters
Maneuvers and road definition
AMESim Vehicle
Optimization of the gains & damping ratios
Loops to refine the SkyHook gains
Determine the probability distribution of force-velocity couples
Define force-velocity couples that should be realized by the damper
Select points from this distribution
Mission profile
Inputs:
Feasible damper hardware definition (from DOE Calculation)
Damper Loop
AMESim Model
Optimization of power consumption
Loops to define the active damper characteristics (piston and rod, valves, flow)
Actuator Model

Fig. 2. Process scheme for the design of the active damper.

4 Vehicle model

The vehicle model has been developed in the 1D environment of LMS Imegine.Lab AMESim using a modular approach, as shown in Figure 3 in which a number of blocks representing the different vehicle subsystems are interconnected.

The multibody equations governing the behavior of the system are contained in the central block which details in fact a 15 DOF (Degrees of Freedom) vehicle model:

- car body: 6 DOF
- steering rack: 1 DOF
- rotation of the wheels: 4 DOF

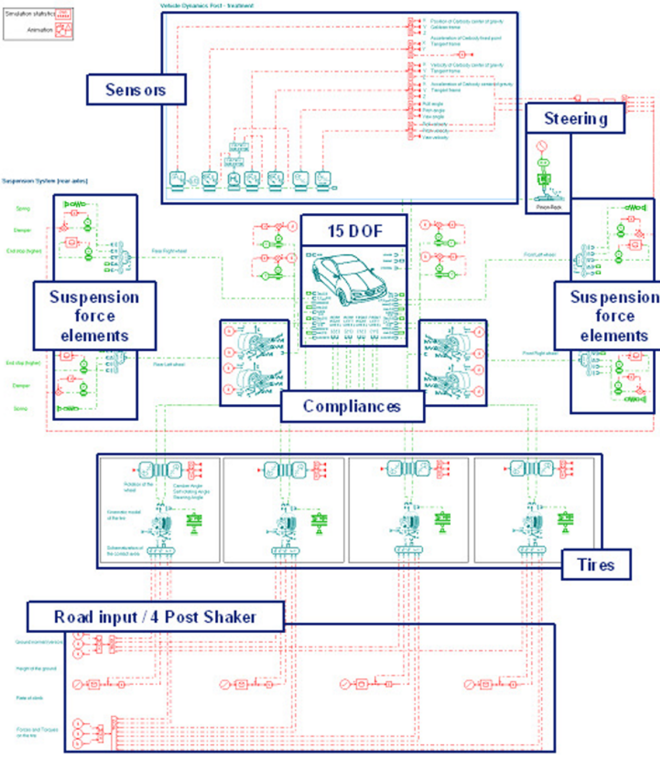

Simulation statistics
Animation
Vehicle Dynamics Post - treatment
Sensors
Steering
Suspension System (rear axles)
Spring
Damper
End stop (higher)
15 DOF
Suspension force elements
Suspension force elements
Compliances
Tires
Road input / 4 Post Shaker

Fig. 3. Model of the vehicle in Imagine.Lab (Amsim).

- vertical displacement of the wheels: 4 DOF

Within this approach a conceptual representation of the suspension behavior is used. More specifically, this means that the different contributions of kinematic characteristics and compliances are addressed as, respectively, look up tables and flexibility matrixes.

Kinematic tables are a functional representation of the axle geometry modification. They describe the variation of track width, wheelbase, steering angle, camber angle and self rotating angle as function of vertical wheel lifts (current, z, and opposite wheel, z_{opp}) and steering rack displacement y_n for front axle system. This allows the definition of the interdependence of the left and right half axle motion as well as the steering input, coming out with the definition of four dimensional matrices.

Flexibility matrices, contained in the compliances blocks, allow the calculation of the contributions in velocity, position, rotary velocity and angle under efforts due to bushing stiffness.

In the modeling of the passively suspended vehicle non linear damping characteristics have been considered as well as end stops, while springs and antiroll bars (present in the front and rear axels) have been included with linear behavior. For all the mentioned force elements acting at the interface between suspended masses nodes and vehicle body connection points, three dimensional tables have been generated to take into account the trigonometrical transformation needed to convert the forces acting within the suspension elements from the pure vertical direction (degree of freedom granted to the tires) to the direction in which the physical elements act and return the forces to be applied on the degrees of freedom of the chassis.

5 Controller modeling

The controller is based on the sky-hook principle [16] and can be illustrated in Figure 4. The principle is illustrated on the "quarter car model". Assuming that every suspension element on each corner of the vehicle works independently (which is a very rough approximation), the behaviour of the car can be represented by 4 independent systems as represented in Figure 4. The sprung and un-sprung masses (M_S and M_{US}) represent respectively one quarter of the body mass and the wheel mass. The connection between each other through a linear spring and damper element, represent the lumped and linearized suspension (subscripted S) stiffness and damping, while M_{US} is connected to the ground with another spring and damper element schematically representing the tire's (subscripted T) vertical behavior. In skyhook damping the suspended mass is connected to an inertial frame: the "sky". "). From the absolute velocity of the suspended mass (d_{ZS}/dt) and the damping coefficient of the skyhook damper (R_{SKY}), one can calculate the force (F_{SKY}) generated by the damper as follows:

$$F_{SKY} = -\frac{dZ_S}{dt}.R_{SKY} \tag{1}$$

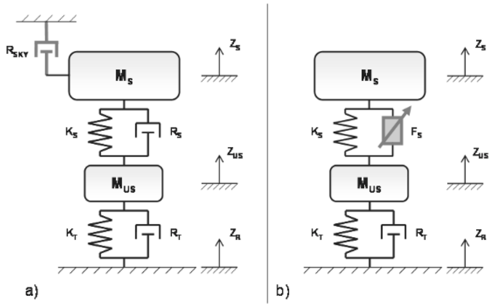


Fig. 4. Ideal skyhook damping (a) and its practical implementation (b).

On top of the skyhook damping, a classical damping force F_{WH}, depending on the relative velocity $(\dot{Z}_S - \dot{Z}_{US})$ between the wheel (unsprung mass) and the car body (sprung mass) is added:

$$F_S = F_{SKY} + F_{WH} = -\frac{dZ_S}{dt}.R_{SKY} - \left(\dot{Z}_S - \dot{Z}_{US}\right).R_{WH} \tag{2}$$

Clearly the two gains R_{SKY} and R_{WH} have a different effect on the vehicle behavior. This is visible in Figure 5, where bode plots of the transfer functions between heave displacement of the body Z_S and road input Z_R are represented, showing the effect of the variation of the two damping coefficients. It can be seen that an increase of R_{SKY} lowers the peak amplitude at the first resonance, while an increase of R_{WH} lowers the amplitude at both
reso ange
abov akes
then ior.

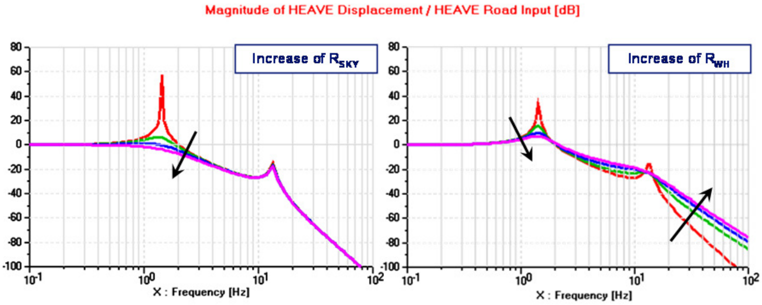


Fig. 5. Effect of the variation of the Skyhook and Wheelhop gains.

The skyhook principle is applied for the following degrees of freedom: heave (linear vertical motion), roll and pitch.

6 Concept model optimization

From Figure 5, it is clear that the skyhook gain doesn't have any influence on the wheel hop damping (second mode around 15 Hz) and the final roll-off of the transfer function. On the other hand, the wheel hop gain affects the body mode (first mode around 1.5 Hz). Therefore, the skyhook gain can be set such that the damping ratio of the body mode is ideally damped i.e. $\frac{1}{2}$. This leads to a nested optimization scheme, which is performed in Noesis OPTIMUS using a differential evolution algorithm. For the tuning of the wheel hop gains, the active full vehicle model is run over three different ride profiles, defined by the vehicle manufacturer. Those profiles, called Road 1, 2 and 3, characterize both common situations as well as extreme events, and can be considered as representing respectively the 5%, 5% and 90% to the definition of the mission profiles on which the components design has to be defined. Given updated wheel hop gains, the skyhook gains are determined such that the damping ratios of the heave, roll and pitch mode are $\frac{1}{2}$. This is performed on the Amesim model discussed in section 4, but linearized around its settled equilibrium position.

7 Shock absorber optimization

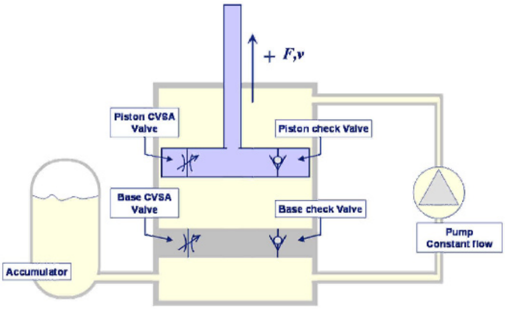


Fig. 6. Schematic representation of the active shock absorber.

The Tenneco shock absorber is shown in Figure 6 and has been modeled in Imagine.Lab, Amesim. Without the pump, it acts like a semi-active damper, in which the damping characteristics are modified by acting on the piston and base CVSA valves. The accumulator is foreseen to take or deliver oil when the shock absorber moves respectively in or out. By adding a pump, the shock absorber becomes active. In case the pressure difference between rod and piston side is small (highest respectively middle chamber of the shock absorber), the larger area at piston side with respect to rod side, creates an upward force on the piston. Note that the pressure on the rod side is always the highest from the three chambers in the damper cylinder. On the other hand, while the pressure difference between rod and piston side is high, a downward force is created.

As stated earlier, following parameters need to be optimized: pump flow, piston and rod diameter, flow characteristics of piston and base valves. Apart from the pump flow, which is constrained to $10l/min$, these parameter values cannot vary continuously. Due to standardization, Tenneco uses a limited number of sizes and valves. From this limited set, a subset must be derived that meets the performance criteria of the concept phase. The performance criteria of the concept phase are translated to power criteria, which are used in a full factorial design of experiments (DOE) in OPTIMUS.

From this subset, the parameters are selected with minimal energy consumption. As equations of hydraulic systems are known to be stiff, small time steps are needed to perform simulations, which makes the simulation of hydraulic systems computationally expensive. Instead of applying the complete mission profiles (force velocity couples of the dampers), obtained from the concept design phase, a limited set of force, velocity couples is selected according their probability. In this way, the original mission profiles are compressed and simulation times are reduced. The final optimization is performed by a genetic algorithm in OPTIMUS.

8 Conclusions

The performance engineering of intelligent vehicle systems mandates simulation and test methods that are capable to simulate, analyze and optimize the performance of such a product, taking into account the interaction of many subsystems and working as active systems with sensors, actuators and interconnections to controllers. The paper presented the optimization of an active damper, in which a conceptual chassis model, including the controller, imposes requirements on the mechanical and hydraulic characteristics of the damper. Nested optimization on the concept level stage, determines the optimal active controller gains. By using full factorial DOE, the set of possible damper parameters is restricted to the ones that meet the performance criteria of the concept design stage. From this restricted set, the configuration is selected that provides minimal energy consumption. Mission profile compression is performed to reduce the computational cost during optimization.

9 Acknowledgements

The presented research is conducted in the context of a coordinated research effort for the development of design engineering solutions for intelligent system. This effort is funded through a number of research projects such as the Flanders' Drive projects ReVAS (Research Vehicle for Active Safety) and InAST (Integrated Active Safety Technology), the IWT project 080067 "MODELISAR" supported by IWT Vlaanderen and the EC Marie Curie ITN project VECOM, funded by the European Commission, from which author, Mr. Marco Gubitosa, holds a grant. We also thank the InAST partners: Tenneco, Triphase and Flanders'Drive for the fruitful discussions about the topic.

References

1. Valsan A (2006) Trends, Technology Roadmaps and Strategic Market Analysis of Vehicle Safety Systems in Europe. In: International Automotive Electronics Congress, Paris oct. 24, 2006.

2. McKinsey's (2004) Automotive & Assembly Practice, HAWK 2015 - Knowledge-based changes in the automotive value chain. McKinsey.
3. JSAE (2007) The automobile in the Year 2030. Society of Automotive Engineers of Japan.
4. Aberdeen Group (2009) System Design: New Product Development for Mechatronics. www.aberdeen.com
5. Laurgeau C (2008) Present and future of intelligent transportation systems. In: Proc ICAT, Int. Conf. on Automotive Technologies, Nov. 13-14, Istanbul (TR)
6. Vahidi A, Eskandarian A (2003) Research advances in intelligent collision avoidance and adapribe cruise control. IEEE Transactions on Intelligent Transportation Systems 4(3):143–153
7. European Commission (2007) Towards Europe-wide Safer, Cleaner and Efficient Mobility: The First Intelligent Car Report. In: Communication from the Commission to the European Parliament, The Council, The European Economic and Social Committee and the Committee of the Regions, COM(2007)541, 17.09.2007
8. Costlow T (2008) Managing Software Growth. Automotive Engineering International Nov. 20, 2008
9. Van der Auweraer A, Vecchio A, Peeters B, Dom S, Mas P (2008) Hybrid Testing in Aerospace and Ground Vehicle Development. In: Saouma V, Mettupalayam S, Sivaselvan M V (eds) Hybrid Simulation: Theory, Implementation and Applications. Taylor Francis Group, London, UK
10. De Cuyper J, Gubitosa M, Kang J, Leth G, Furmann M, Kading D (2008) Vehicle Dynamics Control - A Case Study in LMS Virtual.Lab Motion. In: Proc. 4th Asian Conference on Multibody Dynamics 2008, Aug. 20-23, Seogwipo-city, Korea
11. Moreno Giner D, Brenna C, Symeonidis I, Kavadarlic G (2008) MYMOSA - Towards the Simulation of Realistic Motorcycle Manoeuvres by Coupling Multibody and Control Techniques, In: Proc. IMECE2008, Nov. 2-6, Paper IMECE2008-67297, Boston (USA)
12. de Oliveira L P R, Janssens K, Gajdatsy P, Van der Auweraer H, Varoto P S, Sas P, Desmet W (2009) Active sound quality control of engine induced cavity noise. Mechanical Systems and Signal Processing, 23: 476–488
13. Van der Auweraer H, Mas P, Segaert P, de Oliveira L P R, da Silva M, Desmet W (2007) CAE-based Design of Active Noise Control Solutions. SAE Paper 2007-26-032, Proc. SIAT 2007, Pune (India), 17-20 Jan. 2007.
14. Lauwerys C, Swevers J, Sas P (2005) Robust linear control of an active suspension on a quarter car test-rig, Control Engineering Practice 13:577–586
15. Swevers J, Lauwerys C, Vandersmissen B, Maes M, Reybrouck K, Sas P (2007) A model-free control structure for the on-line tuning of the semi-active suspension of a passenger car, Mechanical Systems and Signal Processing 21:1422–1436
16. Karnopp D C, Crosby M J (1974) Vibration control using semi-active force generators, Journal of Engineering for Industry 96:619–626

Optimal Control of Machine Tool Manipulators

Bahne Christiansen[1], Helmut Maurer[1] and Oliver Zirn[2]

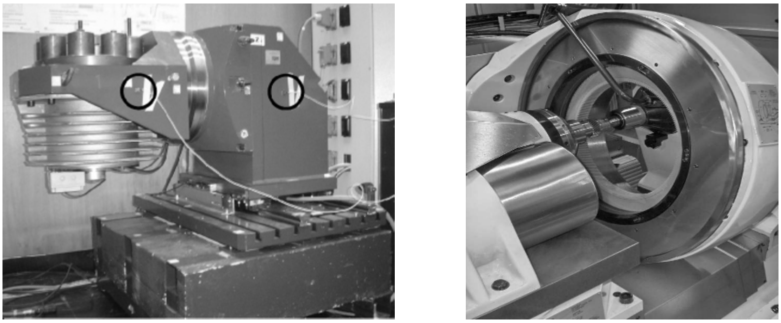
[1] Institut für Numerische Mathematik und Angewandte Mathematik,
Westfälische Wilhelms-Universität Münster,
christiansen@wwu.de, maurer@math.uni-muenster.de

[2] Institut für Prozess- und Produktionsleittechnik
Technische Universität Clausthal,
zirn@ipp.tu-clausthal.de

Summary. Based on the dynamical models for machine tool manipulators in Zirn [11, 12], we compare state-of-the-art feedback controls with optimal controls that either minimize the transfer time or damp vibrations of the system. The damping performance can be improved substantially by imposing state constraints on some of the state variables. Optimal control results give hints for suitable jerk limitations in the setpoint generator of numerical control systems for machine tools.

1 Introduction

Fig. 1 shows two typical machine tool manipulators. Both are representative for the dynamical model presented in the following section.



(a) Milling Machine Tool Workpiece Manipulator

(b) Honing Machine Tool Manipulator

Fig. 1. Typical Machine Tool Manipulators.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_39, © Springer-Verlag Berlin Heidelberg 2010

Fig. 1(a) displays a manipulator on the workpiece side of a 5-axis milling machine with the translatory X-axis driven by a linear motor and two rotary axes. The performance of the linear X-axis is limited significantly by the flexible mechanical structure. Although the voice–coil–motor servo axis that we discussed in [2] also carries a flexible load, Coulombic friction represents the dominating influence on the dynamic axis performance of that example. For the machine tool axis discussed here, the guide friction is comparably small and can be compensated with moderate effort in the axis controller. Fig. 1(b) shows a honing machine tool for the fine finishing of gear wheels. This manipulator has two translatory Z-axes, one for the honing wheel, which is the mid abrasive honing stone, and a second one for the gear wheel.

2 Dynamic control model of a machine tool manipulator

The dynamic process of a machine tool manipulator is considered in the time interval $t \in [0, t_f]$ with t measured in seconds; the final time t_f is either fixed or free. The state variables are as follows: the base position $x_b(t)$, the slider position $x_s(t)$, the slider rotary position $\varphi(t)$, the corresponding velocities $v_b(t)$, $v_s(t)$ and $v_\varphi(t)$ and the X-axis linear motor force $F(t)$. The input variable (control) of the motor is the setpoint motor force $F_{\text{set}}(t)$. The dynamics is given by the following system of linear differential equations, where as usual the dot denotes the time derivative. System parameters are listed in Tab. 1.

$$\begin{aligned}
\dot{x}_b(t) &= v_b(t)\,, & \dot{v}_b(t) &= -\tfrac{1}{m_b}\left(k_b x_b(t) + d_b v_b(t) + F(t)\right)\,,\\
\dot{x}_s(t) &= v_s(t)\,, & \dot{v}_s(t) &= \tfrac{1}{m_s} F(t)\,,\\
\dot{\varphi}(t) &= v_\varphi(t)\,, & \dot{v}_\varphi(t) &= \tfrac{1}{J}\left(rF(t) - k\varphi(t) - d v_\varphi(t)\right)\,,\\
\dot{F}(t) &= \tfrac{1}{T}\left(F_{\text{set}}(t) - F(t)\right)\,.
\end{aligned} \tag{1}$$

The control constraint is given by

$$-F_{\max} \le F_{\text{set}}(t) \le F_{\max}, \quad 0 \le t \le t_f, \tag{2}$$

Base mass	$m_b = 450\,\text{kg}$
Slider mass	$m_s = 750\,\text{kg}$
Slider inertia	$J = 40\,\text{kg}\,\text{m}^2$
Slider centre of gravity excentricity - guides	$r = 0.25\,\text{m}$
Slider centre of gravity excentricity - TCP	$h = 0.21\,\text{m}$
Stiffness of the base anchorage	$k_b = 4.441 \cdot 10^7\,{}^{\text{N}}\!/\!_{\text{m}}$
Damping constant of the base anchorage	$d_b = 8500\,{}^{\text{Ns}}\!/\!_{\text{m}}$
Stiffness of the fictive rotary joint torsion spring	$k = 8.2 \cdot 10^6\,{}^{\text{Nm}}\!/\!_{\text{rad}}$
Damping constant of the fictive rotary joint torsion spring	$d = 1800\,{}^{\text{Nms}}\!/\!_{\text{rad}}$
Current control loop time constant	$T = 2.5\,\text{ms}$
Maximum input force	$\lvert F_{\max}\rvert \le 4\,\text{kN}$

Table 1. List of system parameters

where $F_{\max} \leq 4\,\mathrm{kN}$ holds for mechanical reasons. The initial and terminal conditions for the state vector $x = (x_b, x_s, \varphi, v_b, v_s, v_\varphi, F)^* \in \mathbb{R}^7$ are given by

$$x(0) = (0,0,0,0,0,0,0)^*, \quad x(t_f) = (0, \text{undef.}, 0, 0, 0.1, 0, 0)^*. \tag{3}$$

3 Feedback control performance

The state-of-the-art feedback control for CNC machine tools is shown as a block diagram in Fig. 2.

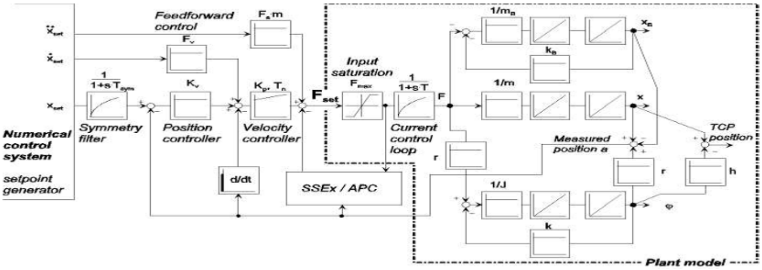
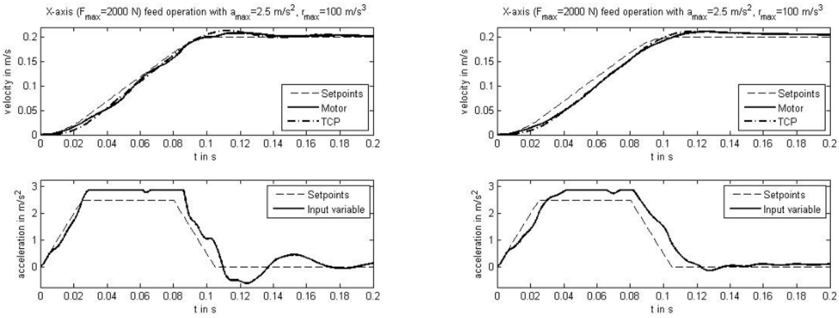


Fig. 2. Block diagramm for feedback control

The velocity-, acceleration- and jerk-limited setpoints generated by the numerical control system are feedback controlled by a cascade controller with velocity and position control loop and a state space control extension to improve the active structural vibration damping for the dominant mode. Compared to robots, where TCP vibrations caused by rough setpoints and flexible structure can hardly be identified by the motor measurement systems and are eliminated by short waiting times after positioning operations, the requirements for machine tool manipulators are much more challenging. Due to the milling or grinding process, all TCP vibrations are visible directly on the workpiece. So it is mandatory that the input variables never excite remarkable structural vibrations at all. Machine tools measure their drive states not only at the motor but also quite near to the TCP, so it is possible to damp structural vibrations by the feedback servo controller. But in practice, the achievable damping is not sufficient and vibration-free control values remain an improtant future issue for machine tools.

The plant model is the block diagram representation of the differential equations discussed in Sec. 2. In Fig. 3(a), the typical setpoints F_{set} that result from such a control concept are shown for an exemplary acceleration operation of the axis: The optimum productivity in combination with suitable damping performance is achieved for significant jerk control, i.e. the maximum acceleration has to be achieved by smooth setpoints. Also with state space control, the suitable setpoints have a typical trapezodial shape, cf. Fig. 3(b). The practical experience with many machine tool axis control applications



(a) Pure cascade controlled system

(b) Axis with state space control extensions

Fig. 3. State-of-the-art feedback control for CNC machine tools

shows that the possible jerk depends on the eigenfrequency of the dominating mode, but no commissioning rules have been derived up to now.

Fig. 3(b) indicates two deficiences of feedback controls: (i) the terminal position is not attained precisely leading to a terminal overshooting, (ii) the transfer time is much larger than the minimal transfer time computed via optimal control methods in the following sections.

4 Optimal control models of machine tool manipulators

The dynamic equation (1) can be written in compact linear form as

$$\dot{x} = f(x, F_{\text{set}}) = Ax + BF_{\text{set}} \tag{4}$$

with a matrix $A \in \mathbb{R}^{7\times 7}$ and a column vector $B \in \mathbb{R}^{7\times 1}$. Since the process duration is an important criterion for the efficient usage of machine tool manipulators, we first consider the control problem of minimizing the final time t_f subject to the conditions (1)–(3). It turns out that some of the time-optimal state trajectories are highly oscillatory. Damping of oscillations can be achieved by an alternative cost functional that is quadratic in control and state variables,

$$\text{minimize} \int_0^{t_f} F_{\text{set}}^2 + c_1 x_b^2 + c_2 \varphi^2 + c_3 v_b^2 + c_4 v_\varphi^2 dt \quad (\text{fixed } t_f > 0), \tag{5}$$

where $c_1, c_2, c_3, c_4 > 0$ are appropriate constants. Of course, the fixed final time t_f in the cost functional (5) must be larger than the minimal time $t_f^{\min}$. Note that we keep the control constraint (2). Another approach to avoid larger oscillations consists in imposing state constraints of the form

$$-c_\varphi \le v_\varphi(t) \le c_\varphi, \quad t \in [0, t_f], \tag{6}$$

with a prescribed constant $c_\varphi > 0$, cf. Sec. 5.2.

5 Time-optimal control

Pontryagin's Minimum Principle involves the adjoint variable (row vector) $\lambda = (\lambda_{x_b}, \lambda_{x_s}, \lambda_{\varphi}, \lambda_{v_b}, \lambda_{v_s}, \lambda_{v_\varphi}, \lambda_F) \in \mathbb{R}^7$ and the Hamiltonian function

$$H(x, \lambda, F_{\text{set}}) = 1 + \lambda(Ax + BF_{\text{set}}). \tag{7}$$

The adjoint λ satisfies the linear adjoint equation $\dot{\lambda} = -\lambda A$,

$$\begin{aligned}
&\dot{\lambda}_{x_b} = \frac{k_b}{m_b}\lambda_{v_b}, \quad \dot{\lambda}_{x_s} = 0, \quad \dot{\lambda}_{\varphi} = \frac{k}{J}\lambda_{v_\varphi}, \quad \dot{\lambda}_{v_b} = -\lambda_{x_b} + \frac{d_b}{m_b}\lambda_{v_b}, \qquad (8)\\
&\dot{\lambda}_{v_s} = -\lambda_{x_s}, \quad \dot{\lambda}_{v_\varphi} = -\lambda_{\varphi} + \frac{d}{J}\lambda_{v_\varphi}, \quad \dot{\lambda}_F = \frac{1}{m_b}\lambda_{v_b} - \frac{1}{m_s}\lambda_{v_s} - \frac{r}{J}\lambda_{v_\varphi} + \frac{1}{T}\lambda_F.
\end{aligned}$$

We have $\lambda_{x_s}(t_f) = 0$, since the terminal state $x_s(t_f)$ is free. Then the adjoint equations (8) imply $\lambda_{x_s}(t) = 0$ and $\lambda_{v_s}(t) = \text{const}$ for all $0 \le t \le t_f$. The optimal control $F_{\text{set}}(t)$ minimizes the Hamiltonian function on the control set $-F_{\max} \le F_{\text{set}}(t) \le F_{\max}$. This gives the control law

$$F_{\text{set}}(t) = -\text{sign}\ (\lambda_F(t))F_{\max}. \tag{9}$$

The linear system (4) is completely controllable, since the 7×7 Kalman matrix

$$C = (B, AB, A^2B, A^3B, A^4B, A^5B, A^6B) \tag{10}$$

has maximal rank 7. Hence, the time–optimal control $F_{\text{set}}(t)$ is of bang–bang type; cf. [5].

The optimal control problem is solved by a discretization approach using Euler's method or a higher order Runge–Kutta integration method. The resulting large–scale optimization problem is implemented via the modeling language AMPL [3] and is solved by the interior point optimization solver IPOPT due to Wächter et al. [10]. Alternatively, we use the optimal control package NUDOCCCS developed by Büskens [1]. Computations with $N = 10000$ grid points show that for all values of $F_{\max} > 0$ the control has the following structure with 5 switching times $0 =: t_0 < t_1 < t_2 < t_3 < t_4 < t_5 < t_f$ and the free final time $t_6 := t_f$:

$$F_{\text{set}}(t) = \left\{ \begin{array}{rl} F_{\max} & \text{for} \ \ t_0 \le t < t_1 \\ -F_{\max} & \text{for} \ \ t_1 \le t < t_2 \\ F_{\max} & \text{for} \ \ t_2 \le t < t_3 \\ -F_{\max} & \text{for} \ \ t_3 \le t < t_4 \\ F_{\max} & \text{for} \ \ t_4 \le t < t_5 \\ -F_{\max} & \text{for} \ \ t_5 \le t \le t_6 \end{array} \right\}. \tag{11}$$

This control structure is not surprising, since one intuitively expects that six degrees of freedom, namely the six variables t_i, $i = 1, \ldots, 6$, would suffice to satisfy the six terminal conditions in (3). The discretization and optimization approach provides switching times that are correct up to 3 – 4 decimals. The arc–parametrization method in [7] then allows us to compute the switching times with higher precision which simultaneously provides a test of optimality [8, 9]. In this method, the *arclengths* of the bang–bang arcs defined by $\xi_j = t_j - t_{j-1}$, $(j = 1, \dots, 6)$, $t_0 := 0$, $t_6 := t_f$ are optimized directly using again the code NUDOCCCS [1].

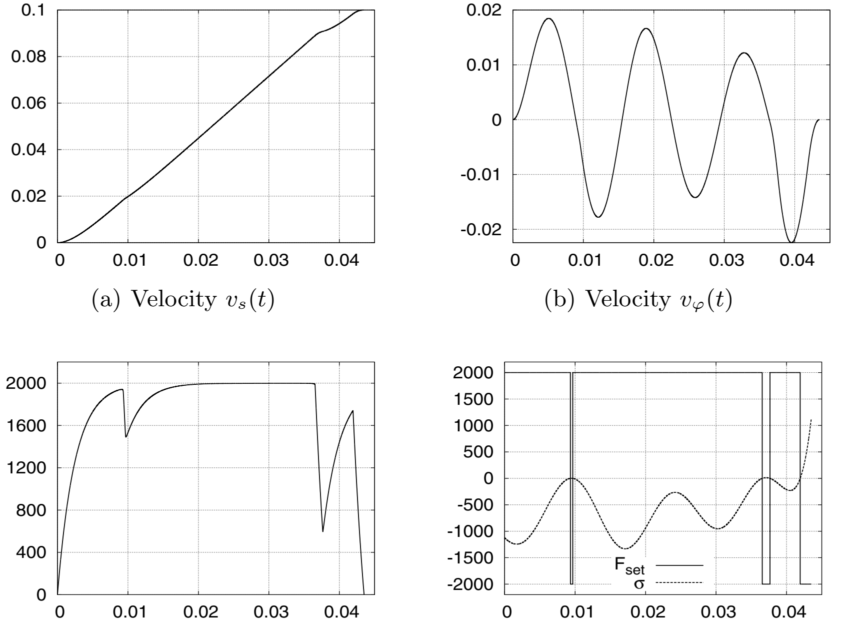
5.1 Numerical results

Fig. 4 displays the optimal solution for the control constraint $F_{\max} = 2\,\text{kN}$. The switching times and final time are computed as

$$\begin{aligned} t_1 &= 0.009337, \; t_2 = 0.009668, \; t_3 = 0.036552, \\ t_4 &= 0.037653, \; t_5 = 0.041942, \; t_f = 0.043505. \end{aligned} \tag{12}$$

The initial value of the adjoint variable $\lambda(t) \in \mathbb{R}^7$ satisfying the adjoint equation (8) is given by

$$\begin{aligned} \lambda(0) = (&-11.87902,\ 0.00000,\ 14.75425,\ 0.05508, \\ &-0.23018,\ 0.01149,\ -1.2503 \cdot 10^{-6}). \end{aligned}$$



(a) Velocity $v_s(t)$

(b) Velocity $v_\varphi(t)$

(c) Motor force $F(t)$

(d) Control $F_{\text{set}}(t)$ and (scaled) switching function $\sigma(t)$

Fig. 4. Time-optimal solution for control bound $F_{\max} = 2\,\text{kN}$

With these values the reader may verify that the switching function $\sigma(t) := H_{F_{\text{set}}}(t) = \lambda_F(t)/T$ obeyes the control law (9) with high accuracy; cf. Fig. 4(d). The local optimality of this trajectory follows from the fact that the 6×6 Jacobian matrix of the terminal conditions computed with respect to the switching times and final time has full rank. Hence, first order sufficient conditions are satisfied for this time–optimal problem; cf. [8, 9].

5.2 State constraints

Higher values of the control bound F_{max} lead to higher vibrations in the machine tool system. Hence, it is reasonable to impose constraints on the oscillating state variables. We restrict the discussion to vibrations of the slider tower φ and consider the state constraint $|v_\varphi(t)| \le c_\varphi$ for the velocity. Following the notations in [4, 6], this can be written as two inequalities

$$S_1(x) := v_\varphi(t) - c_\varphi \le 0, \quad S_2(x) := -c_\varphi - v_\varphi(t) \le 0. \tag{13}$$

Computations show that by imposing these constraints we can also achieve a significant reduction of the deviation $||\varphi(t)||_\infty$. The reader is referred to [4, 6] for the discussion of necessary conditions for state–constrained optimal control problems. It suffices to analyze the component S_1. The constraint has order 2 since the control variable F_{set} appears for the first time in the second time derivative of S_1,

$$\frac{d^2}{dt^2} S_1(x) = \frac{d^2 - k}{J}\, v_\varphi + \frac{dk}{J}\varphi - \left(\frac{r}{JT} + \frac{rd}{J^2} \right) F + \frac{r}{JT}\, F_{\text{set}}.$$

A *boundary arc* $[\,t_{\text{en}},\, t_{\text{ex}}]$ for the constraint $S_1(x) \le 0$ is characterized by the equation $S_1(x(t)) = 0$ for $t_{\text{en}} \le t \le t_{\text{ex}}$ (t_{en} : entry-time, t_{ex} : exit-time). Along a boundary arc the equation $d^2 S_1(t)/dt^2 = 0$ holds, from which we obtain the following feedback expression for the *boundary control*:

$$F^{(b)}_{\text{set}}(x) = \frac{T(k - d^2)}{r}\, v_\varphi - \frac{Tdk}{r}\varphi + \left(1 + \frac{Td}{J}\right) F\,.$$

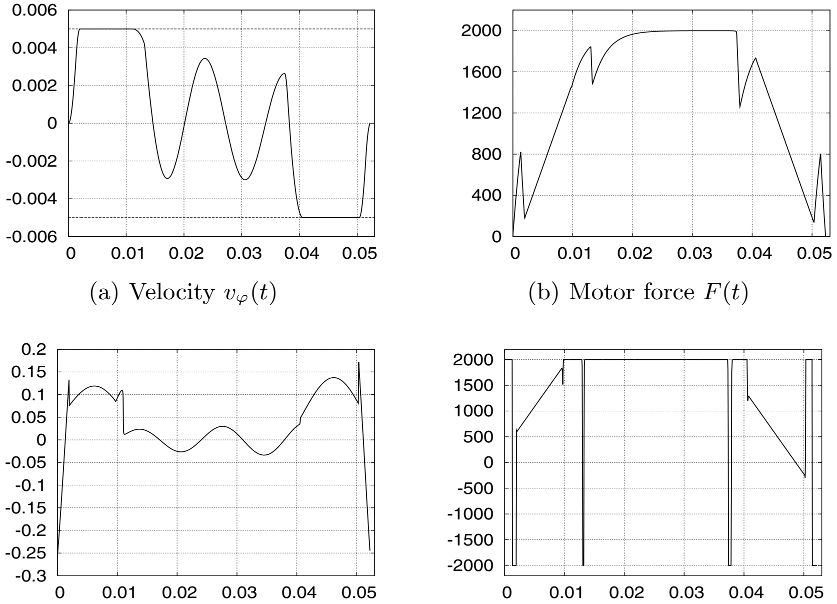
The augmented Hamiltonian $\tilde{H}$ is obtained from the Hamiltonian H by adjoining the state constraint with a multiplier $\mu_1 \in \mathbb{R}$,

$$\tilde{H}(x, F_{\text{set}}, \lambda, \mu_1) = 1 + \lambda(Ax + BF_{\text{set}}) + \mu_1(v_\varphi - c_\varphi).$$

Assuming as in [4, 6] that the boundary control $F^{(b)}_{\text{set}}(x)$ lies in the interior of the control set, it follows from the minimum principle that the switching function vanishes along a boundary arc:

$$\frac{1}{T}\,\lambda_I(t) = \tilde{H}_U(t) = 0 \quad \text{for } \; t_{\text{en}} \le t \le t_{\text{ex}}. \tag{14}$$

From the equation $d^2\lambda_F/dt^2 = 0,\ t_{\text{en}} \le t \le t_{\text{ex}}$, we obtain the relation



(a) Velocity $v_\varphi(t)$

(b) Motor force $F(t)$

(c) Adjoint $\lambda_{v_\varphi}(t)$

(d) Control $F_{\text{set}}(t)$

Fig. 5. Time-optimal solution for control bound $F_{\max} = 2\,\text{kN}$ and state constraint $|v_\varphi(t)| \le c_\varphi = 0.005$

$$\mu_1 = \mu_1(\lambda) = \frac{J}{rm_b}\lambda_{x_b} - \lambda_\varphi - \frac{Jd_b}{rm_b{}^2}\lambda_{v_b}. \tag{15}$$

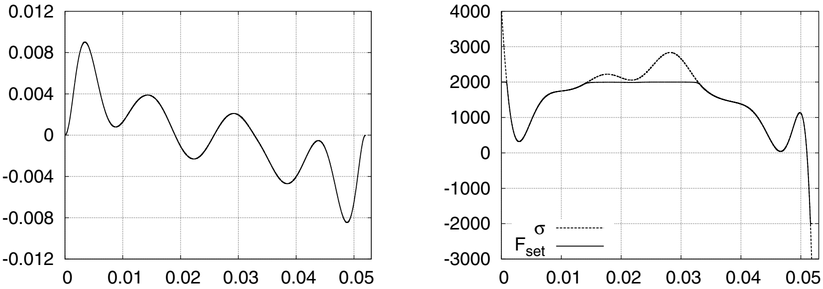
The adjoint variable $\lambda_{v_\varphi}(t)$ may have jumps at the entry and exit time $\tau \in \{t_{\text{en}}, t_{\text{ex}}\}$; cf. [4, 6]. Fig. 5 displays the optimal solution for the rather rigorous bound $c_\varphi = 0.005$. The optimal control has one boundary arc with $v_\varphi(t) = c_\varphi$, one boundary arc with $v_\varphi(t) = -c_\varphi$ and altogether nine interior bang–bang arcs, two of which are located before the first interior arc, five between the interior arcs and two after the last boundary arc. The final time $t_f = 0.0522$ is about 23% higher than in the unconstrained case (12).

6 Damping-optimal control

We consider the “damping-optimal” cost functional (5) of minimizing

$$\int_0^{t_f} (F_{\text{set}}^2 + c_1 x_B^2 + c_2\varphi^2 + c_3 v_B^2 + c_4 v_\varphi^2\,)\,dt,$$

with a fixed final time $t_f > t_f^{\min}$, where $t_f^{\min}$ is the minimal time computed in Sec. 5.1. The weights $c_i, i = 1,..,4$, are equilibrated in such a way that



(a) Velocity $v_\varphi(t)$

(b) Control $F_{\text{set}}(t)$ and (scaled) switching function $\sigma(t)$

Fig. 6. Damping-optimal solution for control bound $F_{\max} = 2\,\text{kN}$, final time $t_f = 0.0522$ and weights (17).

all terms in the quadratic cost functional (5) have the same magnitude. The Hamiltonian

$$\begin{aligned} H(x(t), \lambda(t), F_{\text{set}}) = F_{\text{set}}^2 &+ c_1 x_B^2(t) + c_2\varphi^2(t) + c_3 v_B^2(t) + c_4 v_\varphi^2(t) \\ &+ \lambda(t)(Ax(t) + BF_{\text{set}}) \end{aligned}$$

is regular and admits a unique minimizer

$$F_{\text{set}}(t) = Proj_{[-F_{\max}, F_{\max}]}\,(-\lambda_F(t)\,/\,2T), \tag{16}$$

where $Proj$ denotes the projection onto the control set. Since the convexity conditions of the second order sufficient conditions (SSC) in [4] are satisfied, the optimality of the presented solution is garanteed. The adjoint variables satisfy the adjoint equations $\dot{\lambda} = -2Dx - \lambda A$ with the diagonal matrix $D = \operatorname{diag}(c_1, 0, c_2, c_3, 0, c_4)$. In particular, the control law (16) shows that any optimal control is *continuous*. Fig. 6 displays the optimal solution for the fixed final time $t_f = 0.0522$ that is the minimal time under the state constraint $|v_\varphi(t)| \le c_\varphi = 0.005$. The weigths are given by

$$c_1 = 1.8858\cdot 10^{15},\ c_2 = 1.0961\cdot 10^{15},\ c_3 = 8.6070\cdot 10^{10},\ c_4 = 2.8505\cdot 10^{10}. \tag{17}$$

Fig. 6(b) clearly confirms the control law (16). For this control we get $||\varphi(t)||_\infty = 0.008916$. Though this value is notably higher than the prescribed bound $||\varphi(t)||_\infty = 0.005$ for the time–optimal solution, it is significantly smaller than the value $||\varphi(t)||_\infty = 0.022444$ obtained for the unconstrained time–optimal control.

7 Conclusion

In this paper, we have studied time–optimal and damping–optimal controls for typical machine tool manipulators. Time–optimal controls are bang–bang, for which optimality can be established by second order conditions [7, 8, 9]. The damping performance of time-optimal solutions can be significantly improved by imposing suitable state constraints. Damping-optimal controls are found as solutions to a linear–quadratic control problem with control constraints (saturated control). The numerical results give concrete hints for suitable jerk limitations in the setpoint generator of numerical control systems for machine tools, that otherwise has to be tuned heuristically. This will help to commission control systems for optimal machine performance based on the relevant mechanical system parameters.

References

1. Büskens C (1998) Optimierungsmethoden und Sensitivitätsanalyse für optimale Steuerprozesse mit Steuer- und Zustands-Beschränkungen, Dissertation, Institut für Numerische Mathematik, Universität Münster
2. Christiansen B., Maurer H., Zirn O. (2008) Optimal control of a voice–coil–motor with Coulombic friction, Proceedings 47th IEEE Conference on Decision and Control (CDC 2008), pp. 1557–1562, Cancun, Mexico
3. Fourer R., Gay D.M., Kernighan B.W. (2002) The AMPL Book, Duxbury Press, Brooks–Cole Publishing
4. Hartl R.F., Sethi S.P., Vickson R.G. (1995) A survey of the maximum principles for optimal control problems with state constraints, SIAM Review, 17, pp. 181–218.
5. Hermes H., LaSalle J.P. (1969) Functional analysis and time optimal control, Mathematics in Science and Engineering, 56, Academic Press, New York
6. Maurer H. (1977) On the minimum principle for optimal control problems with state constraints, Schriftenreihe des Rechenzentrums der Universität Münster, ISSN 0344-0842
7. Maurer H., Büskens C., Kim J.-H.R, Kaya C.Y. (2005) Optimization methods for the verification of second order sufficient conditions for bang–bang controls, Optimal Control Applications and Methods, 26, pp. 129–156
8. Maurer M., Osmolovskii N.P. (2004) Second order sufficient conditions for time-optimal bang-bang control problems, SIAM J. Control and Optimization, 42, pp. 2239–2263
9. Osmolovskii N.P., Maurer H. (2005) Equivalence of second order optimality conditions for bang–bang control problems. Part 1: Main results, Control and Cybernetics, 34, pp. 927–950; (2007) Part 2: Proofs, variational derivatives and representations, Control and Cybernetics, 36, pp. 5–45
10. Wächter A., Biegler L.T. (2006) On the implementation of a primal–dual interior point filter line search algorithm for large–scale nonlinear programming, Mathematical Programming, 106, pp. 25–57
11. Zirn O., Weikert S. (2006) Modellbildung und Simulation hochdynamischer Fertigungssysteme, Springer Verlag, Berlin
12. Zirn O.s (2007) Machine Tool Analysis - Modelling, Simulation and Control of Machine Tool Manipulators, Habilitation Thesis, Department of Mechanical and Process Engineering, ETH Zürich

Impact of the Material Distribution Formalism on the Efficiency of Evolutionary Methods for Topology Optimization

Denies J.[1], Dehez B.[1], Glineur F.[2] and Ben Ahmed H.[3]

[1] CEREM - Université catholique de Louvain, Place du Levant, 3, Louvain-la-Neuve, 1348, Belgium, {jonathan.denies, bruno.dehez}@uclouvain.be

[2] ICTEAM & IMMAQ/CORE - Université catholique de Louvain, Voie du Roman Pays, 34, Louvain-la-Neuve, 1348, Belgium, francois.glineur@uclouvain.be

[3] SATIE laboratory - Ecole Normale Supérieure de Cachan antenne de Bretagne, Campus de Ker Lann, 35170, Bruz, France, benahmed@bretagne.ens-cachan.fr[†]

Summary. We consider an evolutionary method applied to a topology optimization problem. We compare two material distribution formalisms (static vs. Voronoi-based dynamic), and two sets of reproduction mechanisms (standard vs. topology-adapted). We test those four variants on both theoretical and practical test cases, to show that the Voronoi-based formalism combined with adapted reproduction mechanisms performs better and is less sensitive to its parameters.

1 Introduction

Optimization methods are used more and more frequently at increasingly early stages in the design process, with the goal of improving performance with respect to cost, weight or other criteria. One can distinguish three paradigms according to the type of design variable used: parametric, geometric and topology optimization.

Parametric optimization deals with a fixed geometry, chosen by the designer, and tries to find optimal choices of geometric parameters such as lengths, widths, etc. Geometric optimization considers instead design parameters which define various shapes in the object under study, using for example spline functions. The designer remains responsible for selecting the initial geometry and choosing which shapes (typically interfaces between materials) are optimized, and how they are parameterized.

In this work, we focus on topology optimization, where design parameters describe the distribution of some materials in a design space. This paradigm

[†] The first author is a FRIA Research Fellow. This text presents research results of the Belgian Program on Interuniversity Poles of Attraction initiated by the Belgian State, Prime Minister's Office, Science Policy Programming. The scientific responsibility is assumed by the authors.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_40, © Springer-Verlag Berlin Heidelberg 2010

offers two significant advantages over the other two. First, it can be started from an empty design space, hence the designer does not have to provide a priori solutions or initial geometries. Second, potential solutions are not restricted in any way, and methods can find designs with a completely arbitrary topology. Topology optimization tools are generally composed of three functional blocks:

1. A material distribution formalism that converts a list of design parameters into a solution (i.e. a design);
2. An evaluation tool that computes the objective function(s) of solutions produced by the material distribution formalism;
3. An optimization algorithm that modifies the solution through its design parameters in order to improve the objective function(s).

These blocks are obviously not completely independent. The choice of one may influence more or less significantly the choice of others and how they are implemented [1, 2, 3, 4]. In this article, we focus on the following two aspects: what is the impact of the material distribution formalism on the performance of the optimization tool, and can the optimization algorithm be adapted to the specific problem of topology optimization.

The material distribution formalisms we consider are based on a subdivision of the design space into cells, each cell being filled homogeneously with a given material. The optimization algorithm used is NSGA-II [5], a genetic algorithm. The choice of an evolutionary meta-heuristic algorithm is motivated by our will to develop a generic optimization tool, completely independent of a particular physics or evaluation tool, that does not require the availability of derivative information and is able to handle discrete parameters (to decide the type of material in each cell) ; other non-evolutionary derivative-free algorithms [6], such as direct search methods, could also be appropriate but fall outside the scope of this work.

This article is structured as follows. Section 2 presents the two different material distribution formalisms we consider, one based on a static division of the design space and the other allowing dynamic divisions using the notion of Voronoi cells. Section 3 proposes one way to adapt, through its reproduction mechanisms, the genetic algorithm to the specific case of topology optimization. Section 4 describes the study cases used in Section 5 to assess the impact of the choice of a material distribution formalism and the adaptation of the optimization algorithm on the quality of the solution found. The results reveal notably that, for a given number of evaluations, a dynamic material distribution formalism leads to solutions with a better objective function, and that the proposed adaptation of the genetic algorithm improves robustness of the results with respect to variations in the number of design parameters used.

2 Material distribution formalisms

Material distribution formalisms can be either static or dynamic. In the first case, subdivision of the design space into cells is decided once and for all before the optimization. Design parameters are then limited to materials constituting each cell, and their total number remains constant. In the second case, the subdivision may evolve during optimization through the number of cells, their shapes and their positions. Design parameters must therefore also include a geometric description of each cell, and their number can vary.

2.1 Static formalism

The static formalism we consider is based on a subdivision of the design space into a regular fixed rectangular grid with m rows and n columns (Fig. 1, left), which is the most frequently used configuration in the literature, see e.g. [8].

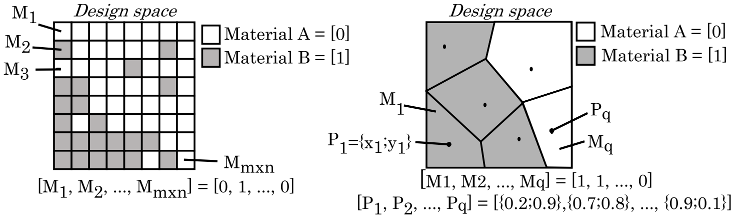

M1
M2
M3
Design space
□ Material A = [0]
▨ Material B = [1]
Mmxn
[M1, M2, ..., Mmxn] = [0, 1, ..., 0]
Design space
□ Material A = [0]
▨ Material B = [1]
M1
P1={x1;y1}
Pq
Mq
[M1, M2, ..., Mq] = [1, 1, ..., 0]
[P1, P2, ..., Pq] = [{0.2;0.9},{0.7;0.8}, ..., {0.9;0.1}]

Fig. 1. Illustration of the static (left) and dynamic Voronoi (right) formalism.

Genetic algorithms manipulate the design parameters via a problem-dependent data structures called chromosomes. In this case, they are arrays where each element, called gene, is a discrete variable indicating the material of the cell. In this work, we only consider two materials, i.e. work with binary variables.

2.2 Dynamic formalism

The dynamic formalism we consider is based on the notion of Voronoi cells [9], whose use in the context of topology optimization was pioneered by Schoenauer (see e.g. [10, 7]). Each of the q cells is defined by its center, and includes all points of the design space that are nearest to this center (Fig. 1, right). In addition to the binary material chromosome of the static case, design parameters include the positions of each cell center, listed in a separate array of real x- and y-coordinates (i.e. $3q$ parameters in total).

3 Reproduction mechanisms

One of the main characteristics of meta-heuristic optimization algorithms is that they can be applied to various problems without requiring special adaptations. Indeed, genetic algorithms can be run as soon as the encoding of the design parameters characterizing a solution (called an individual) into one or more chromosomes is defined. These algorithms evolve a population of individuals by appropriate selection and reproduction mechanisms, with the aim of converging to an optimal solution (or to a set of non-dominated solutions if several objective functions are considered).

3.1 Standard mechanisms

The reproduction mechanisms involved in genetic algorithms are crossover and mutation. Crossover consists in exchanging some of the genes of two individuals, called parents, to produce two new individuals, called children. In its standard version, a pivot is randomly positioned inside the chromosome to determine the genes undergoing the exchange (Fig. 2, left). Mutation consists in modifying the (binary or real) value of a randomly chosen gene (Fig. 2, right).

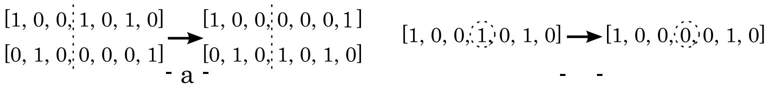


Fig. 2. Standard crossover (left) and mutation (right) reproduction mechanisms

These standard reproduction mechanisms may be applied to both static and dynamic material distribution formalisms (we must nevertheless ensure in the case of the dynamic formalism that crossovers are applied to the same parts of material and position chromosomes). Examples of these standard mechanisms are illustrated on Fig. 3.

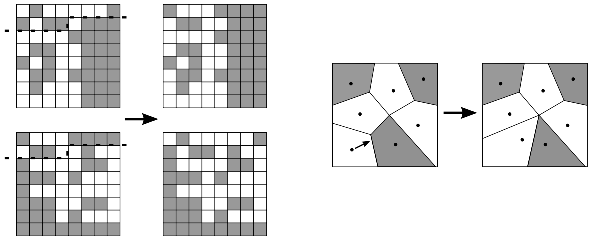


Fig. 3. Examples of standard reproduction mechanisms: crossover with static formalism (left) ; mutation with dynamic formalism (right)

3.2 Adapted mechanisms

The previous selection and reproduction mechanisms are completely generic and independent of the addressed problem. We now propose to use additional mechanisms better suited to the case of topology optimization and its geometric nature. More specifically, we suggest to apply the reproduction mechanisms graphically instead of working directly on chromosomes: a geometric region in the design space will be selected randomly and will then undergo a crossover or a mutation, after which the results will be translated back into the chromosome encoding.

In practice, the adapted crossovers we introduce in the static and dynamic cases are based on a random circle whose center and radius are randomly chosen to fit within the design space. In the static cases, material genes within the circle are exchanged between the parents, while in the dynamic cases both position and material genes are exchanged (see an example on Fig. 4 left).

We also propose to introduce adapted mutations. In the static case, we set a whole randomly selected rectangle (instead of a single gene) to a single type of material (see Fig. 4 right). In the dynamic case, since standard mutations are already effective, we introduce a different type of adapted mutation that consists in randomly adding or deleting a Voronoi cell (note that the adapted crossover mechanism, in contrast with the standard mechanisms, already allows variations in the number of Voronoi cells, see again Fig. 4 left)[4].

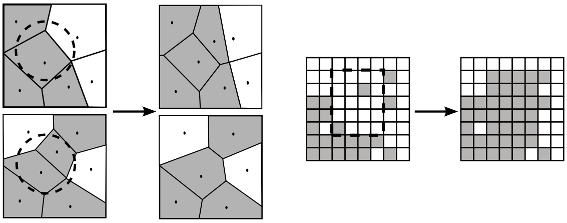


Fig. 4. Adapted mechanisms: dynamic crossover (left) static mutation (right)

4 Study cases

The dominating cost in a typical application of a genetic algorithm to an engineering design problem is the evaluation of the objective function, since computations required for population evolution are typically much cheaper. Therefore, in order to ensure a fair comparison between variants, we run each algorithm for a fixed number of generations, specifically 200 for the experiments reported in Section 5. We also use 1% mutation rates and 80% crossover rates, which have been empirically observed to give good results.

However, like others [10], we first consider a more theoretical test case where the objective function can be evaluated very cheaply. This allows us to run extensive experiments involving all proposed algorithm variants, and derive general observations about them. These conclusions are then validated on an actual engineering problem involving real physics but requiring much higher evaluation times.

4.1 Theoretical case

Our theoretical case study consists in searching for a hidden reference shape (Fig. 5, left). The corresponding objective function to minimize is given by the difference of concordance between the reference shape and that described using the material distribution formalisms. It is evaluated by projecting these two

[4] This however implies that standard crossovers are then no longer possible, because chromosomes can now have different lengths.

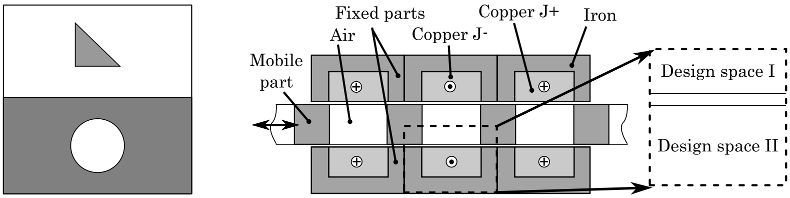


Fig. 5. Diagrams for theoretical (left, reference) and the practical (right) cases.

shapes onto a fine and identical $M \times N$ mesh (with $M \gg m$ and $N \gg n$). The objective function is therefore given by $\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}(p_{ij} \oplus q_{ij})}{M \times N}$, where $\oplus$ denotes the exclusive or operation and p_{ij} and q_{ij} represent components on the fine mesh of the reference solution and of the solution to assess.

4.2 Practical case

Our practical study case concerns the design of a variable reluctance linear actuator (Fig. 5, right). The objective is to maximize the restoring force developed by the actuator between conjunction and opposition positions.

Given this objective and the symmetrical structure imposed on the actuator, the design space can be reduced to a small area (Fig. 5, right). This design space is partitioned into two subspaces, the first related to the mobile part and the other to the fixed part of the actuator.

The objective function to minimize is given by function $f = \psi_{opp} - \psi_{conj}$ [11], where ψ_{conj} and ψ_{opp} are the magnetic flux intercepted by the coils formed by the copper in the conjunction and opposition positions respectively. Evaluation of this function requires the use of a FEM software for calculating magnetic field distribution ; we used version 3.5 of COMSOL [12] (evaluation of a solution takes approximately 2 seconds on a 3 GHz computer).

5 Results and discussion

Whatever the formalism, one can expect that the (initial) number of cells significantly influences the behavior of the topology optimization tool. This is confirmed by results reported on Figs. 5 and 7 for all four combinations (static/dynamic and without/with adaptation). Note first that, in each situation, the smaller the number (initial) cells, the faster the convergence to a solution (a stable solution is even reached before the end of the 200 generations in the two smallest static cases 5×5 and 10×10). This is to be expected since a large number of cells, corresponding to a large number of design parameters, is naturally harder to optimize.

The effect of the proposed adaptation can be observed by comparing the left and right sides of Figs. 5 and 7. On the one hand, for the dynamic formalism, the adaptations are always beneficial, i.e. the final solution is always better. On the other hand, in the static case, results depend on the number

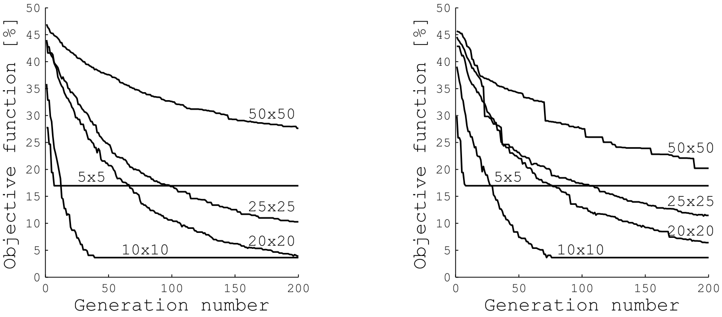


Fig. 6. Convergence of the objective function (theoretical case) for the classical formalism without (left) and with (right) adaptation for different grid sizes.

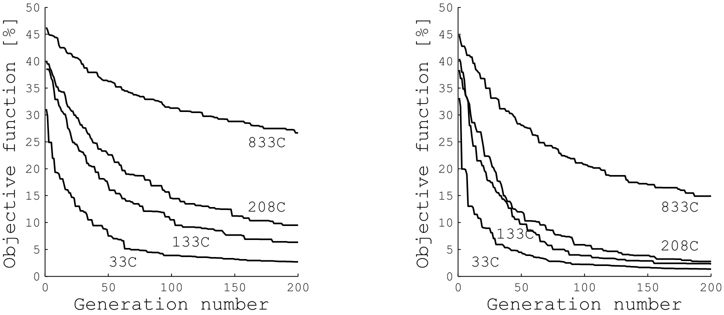


Fig. 7. Convergence of the objective function (theor. case) for the Voronoi formalism without (left) and with (right) adaptation for different initial numbers of cells.

of cells. For small grids, using the standard reproduction mechanisms leads to faster convergence, while the adapted mechanisms perform better for large grids. We explain this by noting that the adapted mutation mechanism, which works with groups of cells, can only speed up convergence when the number of cells is high, allowing more significant changes in the solution at each iteration. For lower number of cells, working with groups of cells has no effect or is even detrimental for the convergence.

Quality of the final solution obtained could be expected to increase when the number of cell increases, because this allows for more precise solutions. This is only partially confirmed by our results: while the static 10×10 result is better than its 5×5 counterpart, this trend does not continue with larger numbers of cells, nor with the dynamic formalism. The reason is that, when the number of cells is large, the 200-generation limit prevents the algorithm from reaching a stable solution. Running with an unlimited number of generations would show that larger numbers of cells lead to better final solutions, but this is of course unrealistic in practice.

Therefore, the initial number of cells becomes a key parameter in a topology optimization process. Too high, the slower convergence rate penalizes the results because the solution does not have time to converge. Too low, the solution converges too quickly to a stable solution with lower quality and generations are wasted. Finding the optimum initial number of cells, one which ensures that the topological optimization tool converges to an optimal solution around the end of the fixed number of generations, is a crucial but difficult challenge, moreover likely to be heavily problem-dependent. Figures 8 and 9 illustrate this tradeoff for our theoretical case study (each box plot stands for 5 experiments).

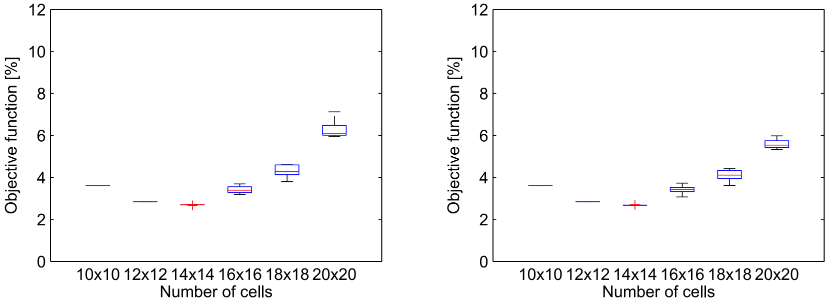


Fig. 8. Result (theoretical case) of the classical formalism without (left) and with (right) adaptation when the number of cells varies

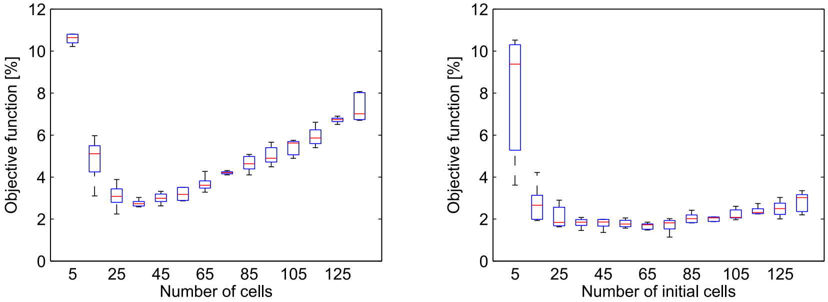


Fig. 9. Result (theoretical case) of the Voronoi formalism without (left) and with (right) adaptation when the number of cells varies

It appears that, when a static formalism is used, or when a dynamic formalism is used without adaptation, quality of the final solution returned by the genetic algorithm is very sensitive to the initial number of cells, the sweet spot for this particular problem being around a 14 × 14 grid or 25-35 Voronoi cells. However, the fourth combination, using a dynamic formalism with adaptations, is clearly much less sensitive to the initial conditions. Recall that this is the only variant where the number of cells can vary from individual to individual. We ascribe its better behaviour to this feature. Indeed, checking the number of cells present in the final solution confirms that this number naturally increases (resp. decreases) when it initially is too low (resp. too high). It is also worth noting that the absolute best objective function among all experiments (around 1.5%) is obtained by this fourth variant.

To conclude this section, we validate these observations on the practical case described at the end of the previous section, with a single run of each of the four versions of the optimization tool, again using a limit of 200 generations. We allocate roughly 200 parameters for both formalisms (a $2 \times 10 \times 10$ grid in the static case, and 67 initial cells in the dynamic case, which corresponds to $3 \times 67 = 201$ design parameters).

Results for the objective function reported in Table 1 are consistent with observations made on the theoretical case (objective values are normalized with respect to the baseline static case without adaptation). The advantage of the dynamic formalism over its static counterpart even seems to be larger than for the theoretical case, with solutions whose objective function is nearly an order of magnitude better than those obtained with the static formalism. Usefulness of the algorithm adaptation is also confirmed, at least in the case of the dynamic formalism.

Distribution formalism	Static	Static	Dynamic	Dynamic
Reproduction mechanisms	Standard	Adapted	Standard	Adapted
Objective function (normalized)	1.00	0.90	6.62	7.20

Table 1. Objective functions obtained after 200 generations in the practical case.

Finally, Fig. 10 displays solutions obtained in the two extreme cases: static formalism with standard reproduction mechanisms (left) and dynamic formalism coupled with adapted mechanisms (right). They suggest that the initial number of cells was too high in the static case, preventing the tool to converge over the course of the 200 generations (observe e.g. the mixture of materials in the lower part of the solution). The solution produced in the second case seems much closer to a stable design. However, the initial number of Voronoi cells was apparently not enough since it rose from 67 to 78 during the optimization. This confirms the observation that the optimization tool based on a combination of a dynamic formalism and an adapted optimization algorithm is much more robust with respect to variations in the initial number of cells.

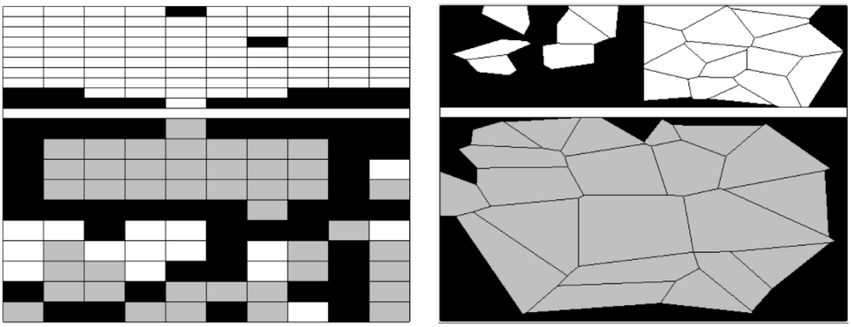


Fig. 10. Actuator design for the practical case obtained with a non-adapted static formalism (left) and adapted dynamic formalism (right).

To conclude, we relate our work with that of Schoenauer et al. (see e.g. [10, 7]), which demonstrates the potential of evolutionary algorithms when applied to the topology optimization of mechanical structures. We confirm their observation that the use of a dynamic formalism with adapted algorithms is beneficial for topology optimization, both on a theoretical case and on a practical application in electromagnetic design.

Our works differs however in several aspects: instead of waiting for convergence of the algorithm, which is unrealistic in many practical situations, we enforce a limit on the number of generations. We demonstrate that the initial number of cells provided to the algorithm is a key parameter influencing the quality of the final solution, but that it cannot be determined a priori. Nevertheless, we show that the quality of the solutions returned by our Voronoi-adapted variant is, through a regulation mechanism on the number of cells, less dependent on the initial number of cells while it converges towards better solutions.

References

1. Dyck D.N., Lowther D.A. (May 1996) Automated design of magnetic devices by optimizing material distribution IEEE Trans. Magn., 32 (3), pp.1188-1193
2. Lowther D. A., Mai W., Dyck D. N., (September 1998) A comparison of MRI magnet design using a hopfield network and the optimized material distribution method, IEEE Trans. Magn., Vol. 34, No 5, pp.2885-2888
3. Dufour S, Vinsard G, Laporte B (July 2000) Generating rotor geometries by using a genetic method, IEEE Trans. Mag., Vol. 36, No 4, pp.1039-1042
4. Denies J., Ben Ahmed H., Dehez B. (2009) Design of a ferrofluid micropump using a topological optimization method, Proc. of Electromotion
5. Deb K., Agrawal S., Pratab A., Meyarivan T. (September 2000) A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multiobjective Optimization : NSGA II, Proc. PPSN VI, 849-858.
6. A. R. Conn, K. Scheinberg, and L. N. Vicente (2009) Introduction to Derivative-Free Optimization, MPS-SIAM Series on Optimization, SIAM, Philadelphia.
7. Schoenauer M. (March 1996) Shape representations and evolution schemes, Proc. of the 5th Annual Conference on Evolutionary Programming, San Diego.
8. Im Chang-Hwa, Jung Hyun-Kyo, Kim Yong-Joo (September 2003) Hybrid genetic algorithm for electromagnetic topology optimization, IEEE Trans. Magn., Vol. 39, No 5, pp.2163-2169
9. Skiena, S. S. (1997) Voronoi Diagrams 8.6.4 in The Algorithm Design Manual. New York, Springer-Verlag, pp. 358–360.
10. Schoenauer M, Jouve F, Leila K (1997) Identification of mechanical inclusions, Evolutionary Computation in Engeneering, pp.477-494
11. Dehez B., Denies J., Ben Ahmed H. (September 2008) Design of electromagnetic actuators using optimizing material distribution methods, Proc. of Int. Conf. on Electrical Machines, Vilamoura (Portugal), ISBN: 978-1-4244-1736-0.
12. COMSOL Multiphyics 3.5a ®, http://www.comsol.com

—

A Variational Model for Image Texture Identification

R. Echegut and L. Piffet

Université d'Orléans, Département de Mathématiques (MAPMO), UFR Sciences,
Bâtiment de mathématiques, Route de Chartres B.P. 6759, 45067 Orléans cedex 2,
FRANCE
echegut.romain@wanadoo.fr, Loic.Piffet@univ-orleans.fr

Summary. A second order variational model is tested to extract texture from an image. An existence result is given. A fixed point algorithm is proposed to solve the discretized problem. Some numerical experiments are done for two images.

Variational models in image processing have been extensively studied during the past decade. They are used for segmentation processes (geodesic or geometric contours), restoration and textures extraction purpose as well. Roughly speaking image restoration problems are severely ill posed and a Tikhonov-like regularization is needed. The general form of such models consists in the mimization of an "energy" functional :

$$\mathcal{F}(u) = \|u - u_d\|_X + \mathcal{R}(u) \ , u \in Y \subset X \ ,$$

where X, Y are (real) Banach spaces, $\mathcal{R}$ is a regularization operator, u_d is the observed (or measured) image and u is the image to recover. Here, we are interested in textures extraction and/or image denoising. Recent works were based on the assumption that an image can be decomposed in many components, each component describing a particular property of the image (see [6, 12, 14] for example). We follow this idea and assume that the image f we want to recover from the data u_d can be decomposed as $f = u + v$ where u and v are functions that belong to different functional spaces: u is the "texture" part which involves (periodic or not) details (and noise as well) while v is a more regular part (usually called the "cartoon" component).

In a first section, we present the functional framework, introducing the BV^2 space, and the general variational model we consider. In section 2, we focus on numerical implementation and present the discretization process. Numerical tests are reported in the last section.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_41, © Springer-Verlag Berlin Heidelberg 2010

1 Functional framework and model

1.1 The $BV^2(\Omega)$ space

Let Ω be an open bounded subset of $\mathbb{R}^n$, $n \geq 2$ (practically $n = 2$). Following Demengel [9], we define the space of Hessian bounded functions that we call $BV^2(\Omega)$. We recall that the space $BV(\Omega)$ of bounded variation functions (see [2, 4, 3]) is defined as

$$BV(\Omega) = \{u \in L^1(\Omega) \mid \Phi(u) < +\infty\},$$

where

$$\Phi(u) = \sup \left\{ \int_\Omega u(x) \operatorname{div} \xi(x)\, dx \mid \xi \in \mathcal{C}_c^1(\Omega),\ \|\xi\|_\infty \leq 1 \right\}. \tag{1}$$

The space $BV(\Omega)$, endowed with the norm $\|u\|_{BV(\Omega)} = \|u\|_{L^1} + \Phi(u)$, is a Banach space. The derivative in the sense of the distributions of every $u \in BV(\Omega)$ is a bounded Radon measure, denoted Du, and $\Phi(u) = \int_\Omega |Du|$ is the total variation of Du. We extend this definition to the second derivative (in the distributional sense). Recall that the Sobolev space

$$W^{1,1}(\Omega) = \{\ u \in L^1(\Omega) \mid \nabla u \in L^1(\Omega)\ \}$$

where ∇u stands for the first order derivative of u (in the sense of distributions).

Definition 1. *A function $u \in W^{1,1}(\Omega)$ is Hessian bounded if*

$$|u|_{BV^2(\Omega)} := \sup \left\{ \int_\Omega \langle \nabla u, div(\phi) \rangle_{\mathbb{R}^n} \mid \phi \in \mathcal{C}_c^2(\Omega, \mathbb{R}^{n \times n}),\ \|\phi\|_\infty \leq 1 \right\} < \infty,$$

where

$$div(\phi) = (div(\phi_1), div(\phi_2), \ldots, div(\phi_n)),$$

with

$$\forall i,\ \phi_i = (\phi_i^1, \phi_i^2, \ldots, \phi_i^n) \in \mathbb{R}^n \quad and \quad div(\phi_i) = \sum_{k=1}^n \frac{\partial \phi_i^k}{\partial x_k}.$$

For more information on the $BV^2(\Omega)$ space, see [9, 13].

1.2 The variational model

We now assume that the image we want to recover from the data u_d can be written as $f = u + v$ where u is in $BV(\Omega)$ and v is in $BV^2(\Omega)$. Such decompositions have already been performed [5, 6, 4] using the "Meyer" space of oscillating function [10] instead of $BV^2(\Omega)$. So far, the model we propose is not the same: the oscillating component will be included in the non regular part u while v involves the cartoon and the contours. We consider the following function defined on $BV(\Omega) \times BV^2(\Omega)$:

$$F(u,v) = \frac{1}{2}\|u_d - u - v\|^2_{L^2(\Omega)} + \lambda |u|_{BV(\Omega)} + \mu |v|_{BV^2(\Omega)} + \delta \|\nabla v\|_{W^{1,1}(\Omega)}, \quad (2)$$

where $\lambda, \mu, \delta \geq 0$ are weigths. We are looking for a solution to the optimisation problem

$$\inf_{(u,v) \in BV(\Omega) \times BV^2(\Omega)} F(u,v) \quad (\mathcal{P})$$

The first term $\|u_d - u - v\|^2_{L^2(\Omega)}$ of F is the fitting data term. Other terms are Tychonov-like regularization terms. Note that the δ-term is not useful from the modelling point of view. It is only a tool that allows to prove existence of solutions. We shall choose $\delta = 0$ for numerical tests.

If the image is noisy, the noise is considered as a texture and will be included in u: more precisely v will be the part of the image without the oscillating component, that is the denoised part. In a previous work, [7], we focused on the denoising process taking only v into account (and assuming that $u = 0$ so that $u_d - v$ is the noise). We now give an existence and uniqueness result for the general problem $(\mathcal{P})$ (see [7] for the proof).

Theorem 1. *Assume that $\lambda > 0, \mu > 0$ and $\delta > 0$. Problem $(\mathcal{P})$ has a unique solution (u,v).*

2 Numerical implementation

2.1 Discretization of the problem

We assume for simplicity that the image is square with size $N \times N$. We denote $X := \mathbb{R}^{N \times N} \simeq \mathbb{R}^{N^2}$ endowed with the usual inner product and the associated euclidean norm

$$\langle u, v \rangle_X := \sum_{1 \leq i,j \leq N} u_{i,j} v_{i,j}, \quad \|u\|_X := \sqrt{\sum_{1 \leq i,j \leq N} u^2_{i,j}} \,. \quad (3)$$

It is classical to define the discrete total variation as follows (see for example [4]) : the discrete gradient of the numerical image $u \in X$ is $\nabla u \in X^2$ defined by

$$(\nabla u)_{i,j} = \left((\nabla u)^1_{i,j} , (\nabla u)^2_{i,j} \right), \quad (4)$$

where

$$(\nabla u)^1_{i,j} = \begin{cases} u_{i+1,j} - u_{i,j} & \text{if } i < N \\ 0 & \text{if } i = N, \end{cases} \quad \text{and} \quad (\nabla u)^2_{i,j} = \begin{cases} u_{i,j+1} - u_{i,j} & \text{if } j < N \\ 0 & \text{if } j = N. \end{cases}$$

The (discrete) total variation $|u|_{BV(\Omega)}$ is given by

$$J_1(u) = \sum_{1 \le i,j \le N} \left\| (\nabla u)_{i,j} \right\|_{\mathbb{R}^2}, \tag{5}$$

where

$$\left\| (\nabla u)_{i,j} \right\|_{\mathbb{R}^2} = \left\| \left((\nabla u)^1_{i,j}, (\nabla u)^2_{i,j} \right) \right\|_{\mathbb{R}^2} = \sqrt{\left((\nabla u)^1_{i,j} \right)^2 + \left((\nabla u)^2_{i,j} \right)^2}.$$

The discrete divergence operator div is the adjoint operator of the gradient operator ∇ :

$$\forall (p,u) \in X^2 \times X, \quad \langle -\text{div } p, u \rangle_X = \langle p, \nabla u \rangle_{X^2},$$

so that

$$(\text{div } p)_{i,j} = \begin{cases} p^1_{i,j} - p^1_{i-1,j} & \text{if } 1 < i < N \\ p^1_{i,j} & \text{if } i = 1 \\ -p^1_{i-1,j} & \text{if } i = N \end{cases} + \begin{cases} p^1_{i,j} - p^2_{i,j-1} & \text{if } 1 < j < N \\ p^2_{i,j} & \text{if } j = 1 \\ -p^1_{i,j-1} & \text{if } i = N. \end{cases} \tag{6}$$

To define a discrete version of the second order total variation we have to introduce the discrete Hessian operator. As for the gradient operator, we define it by finite differences. So, for any $v \in X$, the Hessian matrix of v, denoted Hv is identified to a X^4 vector:

$$(Hv)_{i,j} = \left((Hv)^{11}_{i,j}, (Hv)^{12}_{i,j}, (Hv)^{21}_{i,j}, (Hv)^{22}_{i,j} \right).$$

The discrete second order total variation $|v|_{BV^2(\Omega)}$ of v is defined as

$$J_2(v) = \sum_{1 \le i,j \le N} \| (Hv)_{i,j} \|_{\mathbb{R}^4}. \tag{7}$$

As in the BV case, we may compute the adjoint operator of H (which is the discretized "second divergence" operator) :

$$\forall p \in X^4, \ \forall v \in X \qquad \langle H^* p, v \rangle_X = \langle p, Hv \rangle_{X^4}. \tag{8}$$

and we deduce a numerical expression for H^* from the equality (8). The discretized problem stands

$$\inf_{(u,v) \in X^2} \frac{1}{2} \| u_d - u - v \|^2_X + \lambda J_1(u) + \mu J_2(v) + \delta(|v| + J_1(v)), \tag{$\mathcal{P}_d$}$$

where

$$|v| := \sum_{1 \le i,j \le N} |v_{i,j}| \ .$$

In the finite dimensional case we still have an existence result.

Theorem 2. *Problem $\mathcal{P}_d$ has a unique solution for every $\lambda > 0$, $\mu > 0$ and $\delta > 0$.*

For numerical purpose we shall set $\delta = 0$. In fact, we have performed tests with $\delta = 0$ and very small $\delta \neq 0$ (as required by the theory to get a solution to problem $\mathcal{P}_d$) and results where identical. So, to simplify numerical implementation, we consider the following discretized problem :

$$\inf_{(u,v)\in X^2} \frac{1}{2}\|u_d - u - v\|_X^2 + \lambda J_1(u) + \mu J_2(v). \qquad (\tilde{\mathcal{P}}_d)$$

2.2 Algorithm

Using non smooth analysis tools (for convex functions) it is easy to derive (necessary and sufficient) optimality conditions. More precisely (u,v) is a solution of $(\tilde{\mathcal{P}_d})$ if and only if

$$\begin{cases} 0 \in \partial \left(\lambda J_1(u) + \frac{1}{2}\|u_d - u - v\|^2\right) \\ 0 \in \partial \left(\mu J_2(v) + \frac{1}{2}\|u_d - u - v\|^2\right), \end{cases} \tag{9}$$

where ∂J is the classical subdifferential of J. Using subdifferential properties, we see that (9) is equivalent to

$$\begin{cases} u = u_d - v - \Pi_{\lambda K_1}(u_d - v) \\ v = u_d - u - \Pi_{\mu K_2}(u_d - u). \end{cases} \tag{10}$$

where K_1 and K_2 are closed convex sets. Chambolle [8] proved that

$$K_1 = \{\text{div } p \mid p \in X^2, \ \|p_{i,j}\|_{\mathbb{R}^2} \leq 1 \ \forall i,j = 1,\dots,N\} \tag{11}$$

in the $BV(\Omega)$ setting and we may prove similarly that

$$K_2 = \{H^*p \mid p \in X^4, \ \|p_{i,j}\|_{\mathbb{R}^4} \leq 1, \ \forall i,j = 1,\dots,N\}, \tag{12}$$

(see [7]). Moreover, Chambolle [8] proposed a fixed point algorithm to compute $\Pi_{\lambda K_1}$ and we are able to extend this result to the second order case.

$$p^0 = 0 \tag{13a}$$

$$p_{i,j}^{n+1} = \frac{p_{i,j}^n - \tau(H[H^*p^n - u_d/\lambda])_{i,j}}{1 + \tau\|(H[H^*p^n - u_d/\lambda])_{i,j}\|_{\mathbb{R}^4}}. \tag{13b}$$

which convergence is proved in [7] :

Theorem 3. *Let $\tau \leq 1/64$. Then $\lambda\left(H^*p^n\right)_n$ converges to $\Pi_{\lambda K_2}(u_d)$.*

So, we propose the following algorithm :

- ***Step 1 :*** *We choose u_0 et v_0 (for example, $u_0 = 0$ et $v_0 = u_d$) and* $0 < \alpha < 1$.
- ***Step 2 :*** *define the sequences $((u_n, v_n))_n$ as follows:*

$$\begin{cases} u_{n+1} = u_n + \alpha \left(u_d - v_n - \Pi_{\lambda K_1} \left(u_d - v_n \right) - u_n \right) \\ v_{n+1} = v_n + \alpha \left(u_d - u_n - \Pi_{\mu K_2} \left(u_d - u_n \right) - v_n \right) . \end{cases}$$

- ***Step 3 :*** *if a stopping criterion is not satisfied, set $k := k+1$ and go back to 2.*

We can show that the algorithm converges for $\alpha \in]0, 1/2[$. In practice, we observed that the convergence is faster for $\alpha = 0.6$.

3 Numerical tests and comments

We test the model on two images: The first one is a synthetic image where texture has been artificially added, and the second one is the well known "Barbara" benchmark, often used in texture extraction.

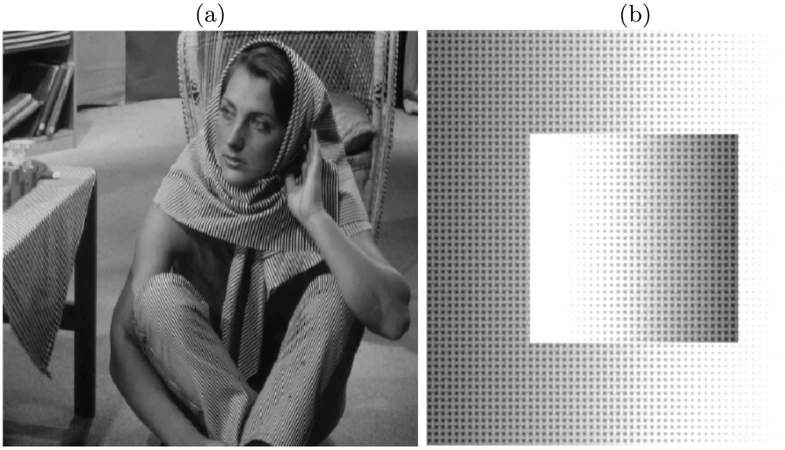


Fig. 1. Original images.

We perform many tests with respect to the different parameters. We only present here the most significant : $\alpha = 0.6$, $\lambda = 0.5$ and $\mu = 100$. Let us first report on the iterations number effect with image (a).

If we are only interested in the texture part, we can observe in fig 2 that we get back all the textures. Unfortunately, most of the geometrical information (that we don't want) is also kept, and we observe that the involved geometric part is getting more important as the iteration number is growing.

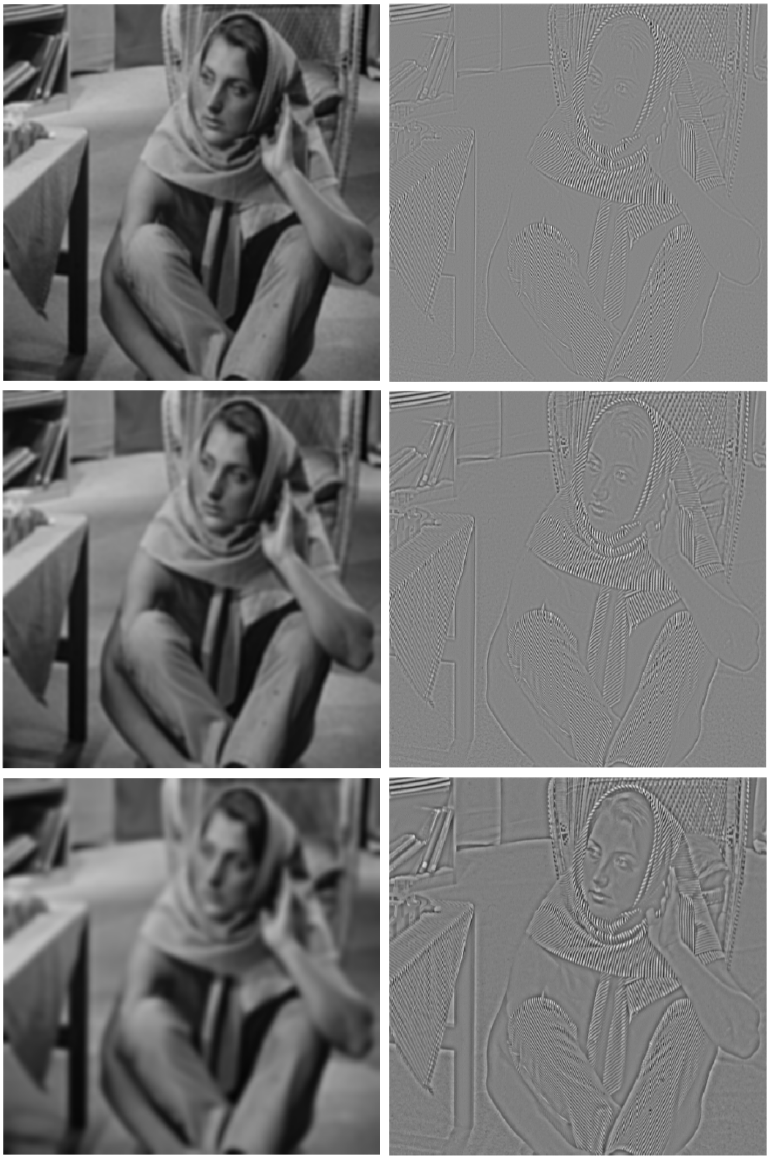


Fig. 2. Number of iterations: first line: 60; second line: 200; third line: 2000.

We see in fig 3 that we can choose a large number of iterations for the texture extraction of image (b) because of its inner structure.

On the other hand, we have to limit this number for the image (a). We give an example image (b) with a very large iterations number.

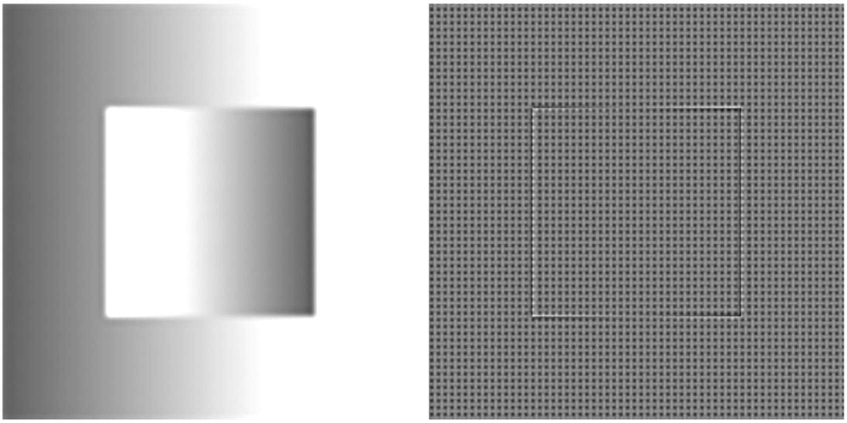


Fig. 3. Number of iterations: 2000.

In addition, we see that too many geometrical information remains together with the texture in the oscillating part: this is a bad point. Nevertheless, our main goal is to locate the texture and we don't need to work with the cartoon part anymore once it has been identified. We do not need to recover all the texture but only a significant part to identify it. In that case, we propose a method that permits to improve the results significantly: we modify the Hessian operator to make it anisotropic. More precisely, we reinforce chosen directions. As texture is made of oscillating information, we hope that we shall keep most of it while many contour lines disappear. We specially act on the vertical and horizontal components of the hessian operator. To deal with non vertical and horizontal lines, we just have to let the image rotate. In the following test, we have replaced the Hessian operator by the operator H' defined for all $v \in X^4$ by :

$$\forall (i,j) \in \{1, ..., N\}^2, \quad (H'v)_{i,j} = \big(0, (Hv)^{12}_{i,j}, (Hv)^{21}_{i,j}, 0\big) \,.$$

We can see on fig 4 that we keep most of the texture without geometrical information. Of course, this method is only efficient on contour lines which are beelines, and permits to deal with only two directions which are necessarily perpendicular. We will propose, in a forthcoming work, a local method to eliminate contour lines in every directions.

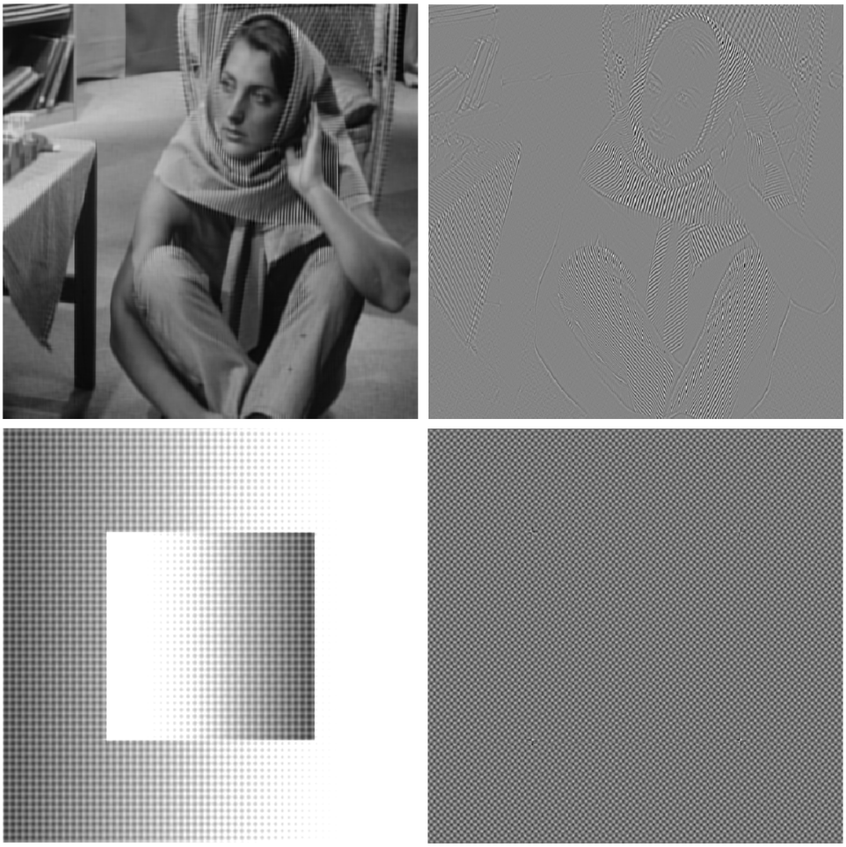


Fig. 4. Test with the anisotropic operator H'. Number of iterations: first line: 60; second line: 2000.

4 Conclusion

The model permits to extract texture from an image, but the texture part still contains too much geometric information. Thus, to recover what we are interested in, we have to use the algorithm with a limited number of iterations. Moreover, we have noticed that we recover too many contour lines as well. The asset of this model is that we can make it anisotropic, modifying the hessian operator in an appropriate way. Therefore we get rid of geometrical information, but we lose part of the texture as well. Nevertheless, if our goal is just to locate texture on an image, this loss remains acceptable.

References

1. Acar R., Vogel C. R. (1994) Analysis of bounded variation penalty methods for ill-posed problems. Inverse Problems, 10: 1217–1229
2. Ambrosio L., Fusco N., Pallara D. (2000) Functions of bounded variation and free discontinuity problems. Oxford mathematical monographs, Oxford University Press.
3. Attouch H., Buttazzo, Michaille G. (2006) Variational analysis in Sobolev and BV spaces : applications to PDEs and optimization. MPS-SIAM series on optimization
4. Aubert G., Kornprobst P. (2006) Mathematical Problems in Image Processing, Partial Differential Equations and the Calculus of Variations. Applied Mathematical Sciences 147, Springer Verlag
5. Aubert G., Aujol J.-F. (2005) Modeling very oscillating signals, application to image processing. Applied Mathematics and Optimization, 51: 163-182
6. Aubert G., Aujol J.-F., Blanc-Feraud L., Chambolle A. (2005) Image decomposition into a bounded variation component and an oscillating component. Journal of Mathematical Imaging and Vision, 22: 71–88
7. Bergounioux M., Piffet L. (2010) A $BV^2(\Omega)$ model for image denoising and/or texture extraction , submitted, Set Valued Analysis
8. Chambolle A. (2004) An algorithm for total variation minimization and applications. Journal of Mathematical Imaging and Vision, 20: 89–97
9. Demengel F. (1984) Fonctions à hessien borné. Annales de l'institut Fourier, 34: 155–190
10. Meyer Y. (2002) Oscillating patterns in image processing and nonlinear evolution equations. Vol. 22 of University Lecture Series, AMS
11. Osher S., Fatemi E., Rudin L. (1992) Nonlinear total variation based noise removal algorithms. Physica D 60:259-268
12. Osher S., Vese L., (2004) Image denoising and decomposition with total variation minimization and oscillatory functions. Special issue on mathematics and image analysis. Journal of Mathematical Imaging and Vision, 20: 7–18
13. Piffet L. (2010) Modèles variationnels pour l'extraction de textures 2D, PhD Thesis, Université d'Orléans
14. Yin W., Goldfarb D., Osher S. (2007) A comparison of three total variation based texture extraction models. Journal of Visual Communication and Image Representation, 18: 240–252
15. Echegut R. (2009) Rapport de stage de Master 1, Université d'Orléans

Optimization Study of a Parametric Vehicle Bumper Subsystem Under Multiple Load Cases

Laszlo Farkas[1], Cedric Canadas[1], Stijn Donders[1], Herman Van der Auweraer[1], and Danny Schildermans[2]

[1] LMS International, Interleuvenlaan 68, B-3001 Leuven, Belgium
laszlo.farkas@lmsintl.com

[2] PUNCH Metals N.V., Nobelstraat 2, B-3930 Hamont-Achel, Belgium
danny.schildermans@punchmetals.com

Summary. This paper deals with the design and optimization of a vehicle bumper subsystem, which is a key scenario for vehicle component design. More than ever before, the automotive industry operates in a highly competitive environment. Manufacturers must deal with competitive pressure and with conflicting demands from customers and regulatory bodies regarding the vehicle functional performance and the environmental and societal impact, which forces them to develop products of increasing quality in even shorter time. As a result, bumper suppliers are under pressure to increasingly limit the weight, while meeting all relevant design targets for crashworthiness and safety. In the bumper design process, the structural crashworthiness performance as the key attribute taken into account, mainly through the Allianz crash repair test, but also through alternative tests such as the impact to pole test. The structural bumper model is created, parameterizing its geometric and sectional properties. A Design of Experiments (DOE) strategy is adopted to efficiently identify the most important design parameters. Subsequently, an optimization is performed on efficient Response Surface Models (RSM), in order to minimize the vehicle bumper weight, while meeting all design targets.

1 Integrated methodology

A methodology is developed and presented to support early balancing between different crash attributes of the vehicle bumper system. Figure 1 presents the schematic representation of the bumper optimization process, starting from geometric design. The process consists of 3 main elements. The first element incorporates design modification and pre-processing in LMS Virtual.Lab [1]. In the second phase, the impact problem is solved with LS-DYNA [2]. The full process of the crash scenario is then captured in the third element OPTIMUS [3], which allows the process integration and design optimization of the sequence in an automated way.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_42, © Springer-Verlag Berlin Heidelberg 2010

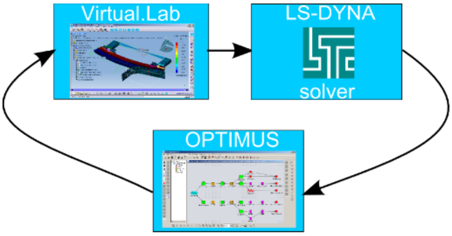


Fig. 1. Schematic representation of the automated process

1.1 Integrated solution for geometry based multi-attribute simulation

A key element in this integrated process is LMS Virtual.Lab, which addresses multi-attribute model assembly and analysis areas to perform end-to-end assessment of a design with respect to multiple performance attributes long before committing to expensive tooling and physical prototypes. For the vehicle bumper subsystem of interest, engineers can start from the CAD design, define a generic assembly model, define multi-attribute simulation models and meshes, as well as multiple analysis cases (see figure 2). The entire process is fully associative, enabling automated iteration of design and model changes, which is key towards an efficient optimization process.

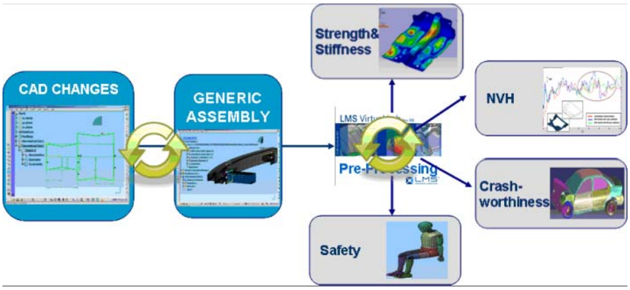


Fig. 2. Integrated solution: from CAD changes to FE models

1.2 Process integration and automation for optimization purpose

In order to automate the entire design procedure from parameter changes to analysis results processing, the above process has been formalized and integrated. For the present case, the OPTIMUS software package has been used to apply the selected analysis methodology and to integrate the different analysis tools for parameter pre-processing, mesh regeneration, crash analysis as well as output extraction and post-processing. The process integration workflow has enabled the automatic execution of the different analysis phases in order to automatically iterate during the optimization process and find the optimal design. Figure 3 shows the workflow of the multi-attribute optimization process, which has been captured.

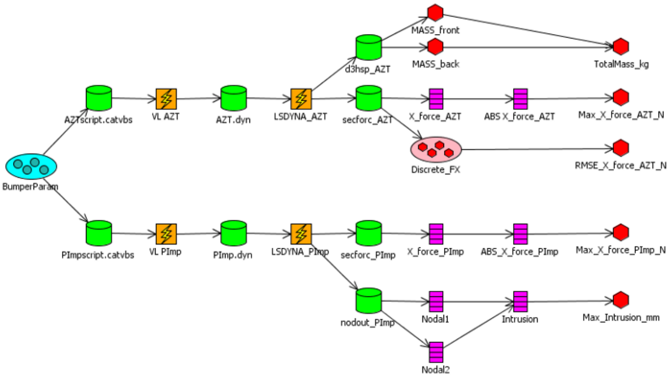


Fig. 3. Process integration workflow in OPTIMUS

1.3 Design exploration and optimization tools

Design of Experiments (DOE)

DOE is a technique [5] that in a statistics context allows the analysis of correlations or shows the statistical significance of an effect, but it is also used for screening purposes or to build meta-models. OPTIMUS provides wide a range of DOE methods for different kinds of applications, such as factorial designs, Box-Behnken, Latin Hypercube, Taguchi or Monte Carlo sampling [4]. In the bumper optimization process, the DOE strategy is used with double purpose: on the one hand it allows the extraction of global sensitivities or so called degree of influence (DOI) [7], on the other hand, the DOE experiments serve as a basis for response surface models (RSM).

Degree of Influence (DOI)

In order to identify the most significant parameters in an optimization process, a large scale sensitivity analysis is performed. Opposed to the generally applied local sensitivity measures based on finite differences, this approach provides large-scale sensitivity information that is calculated based on DOE. Given that for each parameter i, a specific output o is available at 3 different levels (minimum, centre, maximum), the variation of the output o with respect to parameter i is approximated: the large-scale sensitivity is given by $VAR_i^o = (|\Delta 1| + |\Delta 2|)$ (see figure 4). The DOI for each parameter-output pair is expressed with the following formula:

$$DOI_i^o = VAR_i^o / \sum_i VAR_i^o \tag{1}$$

The DOI information is used to select a subset of parameters that have strong influence on the outputs. Parameters with a minor influence can be omitted form further analysis. This way, the computational burden on the optimization is relaxed.

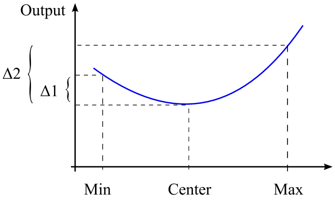


Fig. 4. Variation of output w.r.t. an input parameter

Response Surface Modelling (RSM)

DOE is often used to build a RSM [6]: a meta-model is estimated from the experimental data, to build an approximate functional relationship between the input parameters and the true response. In this context, OPTIMUS offers a whole range of meta-models, from the simple polynomial approximations to more advanced Radial Basis Functions or Kriging models [4].

1.4 Multi-objective Optimization

In many cases, design engineers are faced with multiple objectives, possibly conflicting with each other, so that some trade-off between the optimality criteria is needed. Multi-objective optimization methods that construct the so-called Pareto front allow the solution of such problems. The goal of the different methods that generate the Pareto front is to find a number of points on the Pareto surface, by giving different weights to the different objectives [4]. In order to limit the total computational effort required for a full optimization process, a hybrid optimization approach has been used, taking advantage of DOE and RSM techniques, which is summarized in the following steps:

- Design space exploration with DOE
- Response surface modelling of the functional performance
- Multi-objective optimization, based on the response surface model
- Validation of the obtained results

For the present paper, given the computational time required for one single execution of the complete analysis, the DOE approach limits the total computational effort that needs be spent. The optimization relies on the creation of response models to considerably speed up the process. To guarantee the validity of the optimum found with the efficient RSM analyses, the results of the optimization process obtained with the RSM are verified, which allows assessing the local error at the optimum between the predictive response model and the simulation analysis.

2 Application: mass optimization of a bumper system

To illustrate the methodology described in section 2 of this paper, an optimization study is performed on an industrial parametric CAD bumper system.

This application case has been defined by LMS and PUNCH as a representative bumper design scenario of semi-industrial complexity, which will be used in this paper to demonstrate the structural simulation optimization methodologies.

2.1 Bumper system

The bumper geometry has been taken from an industrial design practice with a mesh density that is both acceptable for the predictions of interest and also feasible in terms of computational effort. The geometry consists of a cross section made of 2 chambers. Subsequently, an assembly is made to connect with the bumper, the longitudinal beams through brackets using seamweld connections and rigid connections (see figure 5).

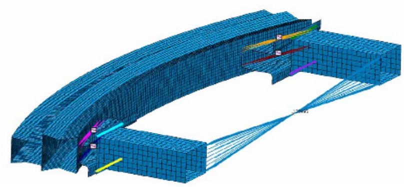


Fig. 5. CAD-based mesh and assembly of the bumper system

2.2 Load cases: reparability low speed impact

Two load cases are considered for the evaluation of the crashworthiness performance of the vehicle bumper system: the Allianz crash repair test and the impact to pole test. The Allianz test (AZT) is the most important low speed load case in the vehicle bumper design. This test aims at evaluating the reparability cost, and is used by insurance companies to determine the insurance fee of a vehicle. The more damage the vehicle will endure in this impact case, the higher the insurance fee will be. The AZT test protocol prescribes a 40% offset impact at $16 km/h$ against a rigid barrier(see figure 6 left). To minimize the reparability cost, the deformation should be limited within the bumper system and the load transferred to the longitudinal members should be limited to avoid permanent deformations. Frontal pole-impact test is used to study the intrusion during a frontal impact with a rigid pole. Similarly to the AZT test, it allows evaluating the repairability cost of the bumper system

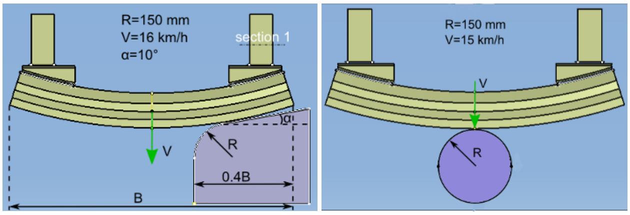


Fig. 6. The Allianz and impact to pole load cases

in a different typical crash scenario. The larger the intrusion, the higher the risk of damaging costly parts, such as the engine cooling system. This test consists of a $15km/h$ central impact against a rigid pole (see figure 6 right).

3 Optimization

The goal of the optimization process is to obtain an optimized bumper profile in terms of mass and Allianz test crash performance, while satisfying a set of design constraints. Multi-objective optimization ensures an optimal trade-off between the two selected objectives. At each iteration of the DOE experiments, 2 parallel analyses are performed, one analysis for each load case.

3.1 Input parameters

In order to optimize the bumper system, 9 parameters are considered. Parameters L_1, H_1, H_2, G_1, G_2, D_1 and D_2 are geometrical parameters that define the profile of the bumper, while t_1 and t_2 represent shell thickness values. The cross-sectional length of the bumper is considered to be fixed to $L = 150mm$. The parameter ranges and the nominal values are presented in table 1.

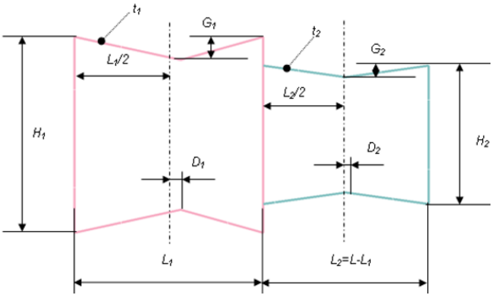


Fig. 7. Bumper parameters

Table 1. Design parameters

Parameter	L_1	H_1	H_2	G_1	G_2	D_1	D_2	t_1	t_2
Min[mm]	60	70	55	5	0	-15	-15	2	2
Max[mm]	100	100	65	15	10	15	15	4	4
Nom[mm]	85	85	60	10	5	0	0	3	3.3

3.2 Objectives and constraints

Nowadays, with the increasing awareness of the environmental footprint of the vehicle, mass reduction of the different vehicle subcomponents is mandatory. Reducing the mass of the bumper is therefore the primary objective. To optimize energy absorption potential of the bumper for the Allianz test, the deviation with respect to an ideal $85kN$ constant curve is considered. The target curve is the ideal force level to absorb the total kinetic energy of a $1200kg$ car that crashes into the rigid barrier in conformity with the Allianz test, with an initial velocity of $16km/h$. The target force level is equivalent to $11,9kJ$ (total initial kinetic energy), based on a deformation length of $140mm$ (total collapse of the bumper section). The average deviation of the actual force-deflection curve from this ideal curve is expressed with the root mean squared error (RMSE) formula that is based on 10 sample points:

$$RMSE\_F_x = \sqrt{\textstyle\sum_{i=1}^{10}(F_x^i - 85kN)^2/10} \qquad (2)$$

Figure 8 shows the AZT load case sectional force X at section 1 for the nominal bumper variant. The red line represents the ideal force curve, while the black dots represent the sampled data for the RMSE calculation. The optimization is subject to two constraints: the X force level at section 1 during the AZT test is limited to $120kN$, and the intrusion for the pole impact scenario is limited to $100mm$.

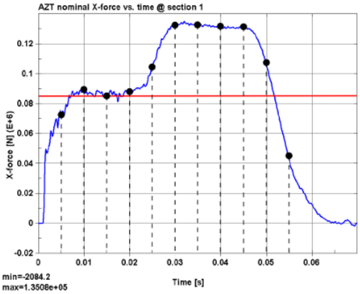


Fig. 8. X-force at section 1 vs. time

Table 2. Summary of the objectives

Objectives	Abbreviation	Nominal value
Total bumper mass	$Mass$	$5.54kg$
AZT test: RMSE of X-force w.r.t. $85kN$	$RMSE\_F_x$	$33kN$

Table 3. Summary of the constraints

Constraints	Abbreviation	Nominal value	Limit value
AZT test: highest X force section 1	Max_F_x	$135kN$	$120kN$
Pole impact test: largest bumper intrusion	Max_Int	$52mm$	$100mm$

3.3 First screening results: DOI

In order to identify the most significant parameters with respect to the objectives and constraints, a first output screening based on the DOE is performed. The objective of this step is to reduce the number of parameters from 9 to 5. This parameter reduction results in a reduced number of experiments used as basis for the RSM. For a 3-level full factorial (3FF) design, the full set of 9 parameters would result in 19683 experiments. 3FF design based on the reduced set of parameters results in a feasible number of 243 experiments. Given 70 minutes CPU time for 1 experiment, the 3FF design could be covered within 12 days. The DOE adopted for the large scale sensitivities (DOI's), consists of a set of experiments that includes the central point and the extreme points, requiring a total number of 19 evaluations. Based on the DOI results (see figure 9), a set containing 5 parameters is selected: L_1, H_1, H_2, t_1, t_2.

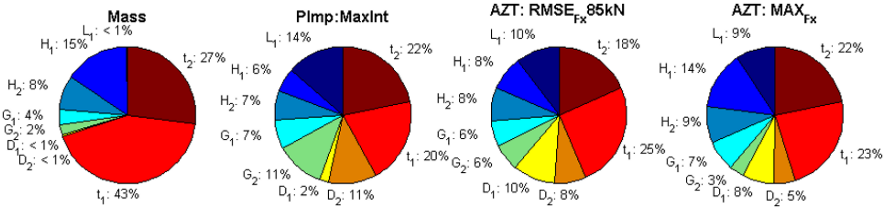


Fig. 9. DOI of the 9 parameters with respect to objectives and constraints

3.4 DOE and RSM selection

The 5 considered parameters are used for a DOE based on 3FF design, to ensure uniform sampling of the design space. The experimental results of the objectives and constraints are then used to build a meta-model for each objective and constraint. The Radial Basis Functions-based (RBF) interpolating response models [8] amended with quadratic polynomial functions are adopted for this purpose, and subsequently used in the multi-objective optimization.

3.5 Bumper Design Optimization

The multi-objective optimization problem is solved with the Normal-Boundary Intersection (NBI) method which searches the Pareto front that represents the set of optimal trade-off solutions [9]. Table 4 summarizes 5 selected Pareto-optimal solutions that are obtained with 1367 iterations based on the RSM using the NBI method. As a final step, the optimum with weight of 0.5 for both objectives has been selected and validated (see table 5). The validation of the optimum shows some difference (13%) as compared to the RMSE objective, which indicates room for improvement of the RSM for this specific output. This potential improvement however was not adressed in this study. The start and optimized geometries of the bumper are showed in figure 10. Figure 11 compares the normal sectional force profile for both the initial and

Table 4. 5 different trade-off optimums

	L_1	H_1	H_2	t_1	t_2	$Mass$	Weight	$RMSE\_F_x$	Weight	Max_F_x	Max_Int
	$[mm]$	$[mm]$	$[mm]$	$[mm]$	$[mm]$	$[kg]$	$Mass$	$[kN]$	$RMSE$	$[kN]$	$[mm]$
Opt1	60	75.6	56.7	2.29	2.89	4.38	1	19.3	0	119	94
Opt2	60.5	75.1	55.9	2.34	2.88	4.41	0.75	17.7	0.25	119	92
Opt3	61.5	74.4	55.2	2.44	2.89	4.49	0.5	16.3	0.5	120	88
Opt4	63.8	74.5	55	2.62	2.94	4.68	0.25	15.1	0.75	117	80
Opt5	73.3	82.7	55.8	2.92	2.94	5.12	0	14.5	1	107	61

Table 5. The selected optimum and validation

	L_1	H_1	H_2	t_1	t_2	$Mass$	$RMSE\_F_x$	Max_F_x	Max_Int
	$[mm]$	$[mm]$	$[mm]$	$[mm]$	$[mm]$	$[kg]$	$[kN]$	$[kN]$	$[mm]$
Start	80	85	60	3	3.3	5.54	33.7	135	52.5
RSM	61.57	74.43	55.27	2.44	2.89	4.49	16.3	120	88.5
Simulation	∼	∼	∼	∼	∼	4.5	18.8	118.5	90.5
Relative error						0.2%	13%	1.2%	2.2%

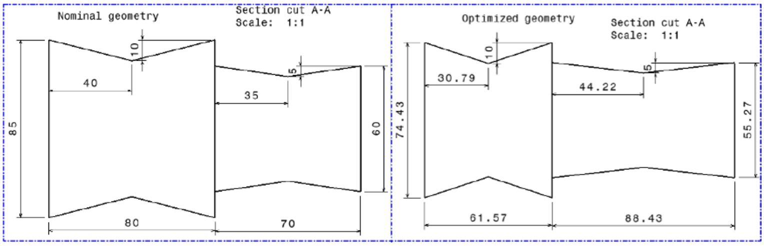


Fig. 10. The initial and the optimized bumper geometries

the optimized design. The optimized bumper has an improved performance: the mass is reduced with 18.7% and the RMSE of the normal sectional force as compared to the ideal force profile is reduced with 44%, while the imposed constraints are satisfied.

4 Conclusions and discussion

This paper presents a generic methodology for automated crash performance optimization, which is illustrated on a real-case scenario. LMS Virtual.Lab offers an integrated solution for CAD-based simulations with the benefits of decreasing analysis time by means of quick model updates, by offering an integrated platform for multi-attribute system modelling for design engineers. The crash design process from parametric model modification and preprocessing in LMS Virtual.Lab and the solution of the crash problems with LS-DYNA is captured with the use of OPTIMUS, which is a dedicated platform for

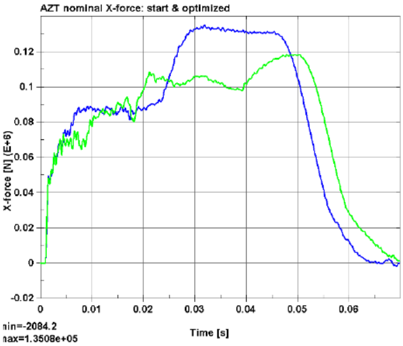


Fig. 11. The original design (blue) and the optimized design (green)

process automation that enables multi-disciplinary design optimization. The automated methodology is illustrated on a vehicle bumper system that is subject to multiple load cases. It is shown that the multi-objective optimization process based on DOE and RSM significantly improves the crash performance of the bumper while reducing mass and satisfying different crash criterias.

Acknowledgements

The work presented in this paper has been performed in the frame of the ongoing research project IWT-070401 ”Simulation-Based Development of an Innovative Bumper Beam Concept with Integrated Crashbox Functionality (**I-CRASH**)”, which is carried out by PUNCH Metals N.V. and LMS International, and which is supported by IWT Vlaanderen.

References

1. LMS International (2009) LMS Virtual.Lab. Leuven
2. Livermore Software Technology Corporation (2009) LS-DYNA Keyword Users's Manual, Version 971, Volume I–II
3. Noesis Solutions N.V. (2009). Livermore OPTIMUS. Leuven
4. Noesis Solutions N.V. (2008) OPTIMUS Theoretical Background. Leuven
5. Montgomery D. C. (1984) Design and Analysis of Experiments. John Wiley, New York
6. Barthelemy J.F.M., Haftka R.T. (1993) Approximation concepts for optimum design - a review. Structural Optimization 5:129–144
7. Farkas L., Moens D., Vandepitte D., Desmet W. (2007) Application of fuzzy numerical techniques for product performance analysis in the conceptual and preliminary design stage. Computers & Structures, 86/10:1061–1079
8. Hongbing F., Mark F.H. (2005) Metamodeling with radial basis functions. 46th AIAA/ASME/ASCE/ ASC Structures, Structural Dynamics and Materials Conference, AIAA-2005-2059
9. Das I., Dennis J.E. (1996) Normal-Boundary Intersection: An Alternate Approach for Generating Pareto-optimal Points in Multicriteria Optimization Problems. ICASE-NASA Tech. Report 96–62

Application of Learning Automata for Stochastic Online Scheduling

Yailen Martinez[1,2], Bert Van Vreckem[1]*, David Catteeuw[1], and Ann Nowe[1]

[1] Computational Modeling Lab, Vrije Universiteit Brussel, Pleinlaan 2, 1050 Brussels, Belgium {ymartine,bvvrecke,dcatteeu,ann.nowe}@vub.ac.be
[2] Central University of Las Villas, Santa Clara, Cuba, yailenm@uclv.edu.cu

Summary. We look at a stochastic online scheduling problem where exact job-lenghts are unknown and jobs arrive over time. Heuristics exist which perform very well, but do not extend to multi-stage problems where all jobs must be processed by a sequence of machines.

We apply Learning Automata (LA), a Reinforcement Learning technique, successfully to such a multi-stage scheduling setting. We use a Learning Automaton at each decision point in the production chain. Each Learning Automaton has a probability distribution over the machines it can chose. The difference with simple randomization algorithms is the update rule used by the LA. Whenever a job is finished, the LA are notified and update their probability distribution: if the job was finished faster than expected the probability for selecting the same action is increased, otherwise it is decreased.

Due to this adaptation, LA can learn processing capacities of the machines, or more correctly: the entire downstream production chain.

1 Introduction

Multi-stage scheduling over parallel machines

Batch chemical plants usually consist of a series of one or more processing stages with parallel processing units at each stage. A new trend in production processes is to operate flexible, adaptive multi-purpose plants. We look at an application based on the chemical production plant of Castillo and Roberts [1, 2]. It is a two-stage process with four times two parallel machines, see Figure 1.

Each order (created at P_1 and P_2) must be handled first by a 'stage-1' machine M_{1-} and afterwards by a 'stage-2' machine M_{2-}. At each stage, a scheduler must choose between two parallel machines. *Parallel machines* can handle the same type of tasks, but may differ in speed. The possible choice in

* This research is partially funded by the IWT-SBO Project (DiCoMAS) "Distributed Collaboration using Multi-Agent System Architectures"

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_43, © Springer-Verlag Berlin Heidelberg 2010

parallel machines is depicted by the arrows in the figure. All machines have a FIFO-queue and execute jobs non-preemptively.

Stochastic online scheduling

The length of the jobs varies according to an exponential distribution. Only the average joblength is known by the schedulers. Also, the machines' speeds are unknown. Even the expected processing time of the jobs is unknown. However, when a job is finished, the scheduler has access to its exact processing time.

Moreover, it is not known in advance when a new order will arrive. I.e. we have an *online* scheduling problem. In an offline problem, all product orders are known in advance. An optimal algorithm will find the best feasible schedule if time and memory restrictions allow it to be computed. In an online scheduling problem, an algorithm has to make decisions based on the history (i.e. information of already released or finished jobs) and the current product request. It is obvious this makes for a more challenging problem. Moreover, no algorithm can find the optimal schedule for all possible input sequences.

Approaches

This problem is particulary hard since it is stochastic, online and multi-stage at the same time.

There exist heuristics for online stochastic scheduling in the single-stage scenario. But these cannot be easily mapped to a multi-stage problem, in this case we do not only need the information about the immediate available machines, but also the information about the machines of the coming stages and this, of course, increases the complexity. In Section 3 we discuss one such heuristic.

In the next section, we introduce Reinforcement Learning and Learning Automata. We propose to apply these techniques for difficult scheduling problems such as the one described above. Later, we will compare Learning Automata to the heuristic of Section 3 in an easy setting.

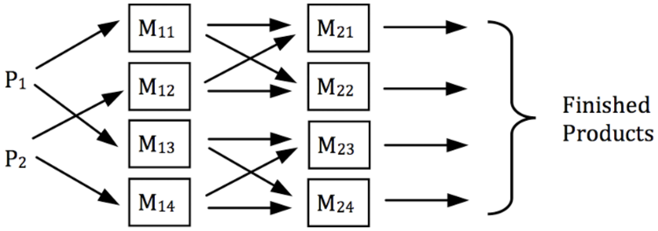


Fig. 1. A two-stage chemical production plant. For both product types P_1 and P_2, there are two parallel machines at the first stage. At the second stage of the process, there are also two parallel machines.

2 Reinforcement Learning

Reinforcement Learning (RL), as noted by Kaelbling, Littman and Moore in [4], dates back to the early days of cybernetics and work in statistics, psychology, neuroscience, and computer science. It has attracted increasing interest in the machine learning and artificial intelligence communities during the past fifteen years.

RL is learning what to do in which situation to maximize a numerical reward signal. The learner is not told which actions to take, as in most forms of machine learning, but instead must discover which actions yield the most reward by trial-and-error. In the most interesting and challenging cases, actions may affect not only the immediate reward but also the next situation and, through that, all subsequent rewards. These two characteristics, trial-and-error search and delayed reward, are the two most important distinguishing features of RL [3].

In the standard RL paradigm, an agent is connected to its environment via perception and action, as depicted in Figure 2. In each step of interaction, the agent senses the current state s of its environment, and then selects an action a which may change this state. The action generates a reinforcement signal r, which is received by the agent. The task of the agent is to learn a policy for choosing actions in each state to receive the maximal long-run cumulative reward.

One of the challenges that arise in RL is the trade-off between *exploration and exploitation*. To obtain a high reward, an RL agent must prefer actions that it has tried in the past and found to be effective in producing reward. But to discover such actions, it has to try actions that it has not selected before. The agent has to exploit what it already knows in order to obtain reward, but it also has to explore in order to make better action selections in the future. The dilemma is that neither exploration nor exploitation can be pursued exclusively without failing at the task. The agent must try a variety of actions and progressively favor those that appear to be best.

In many cases the environment is stochastic. This means, (i) rewards are drawn from a probability distribution and (ii) for each current state s and the chosen action a there is a probability distribution for the transition to any other state. As long as the environment is stationary (i.e. the transition and reward probabilities do not change over time) RL agents can learn an optimal policy. This was e.g. proven for the well-known Q-Learning algorithm [5].

In the next section we look at a particular RL method: Learning Automata.

2.1 Learning Automata

Learning Automata (LA) [6] keep track of a probability distribution over their actions.[3] At each timestep an LA selects one of its actions according to its

[3] Here we look only at 'Linear Reward' LA, there are many more described in literature [6], but this is probably the most widely used.

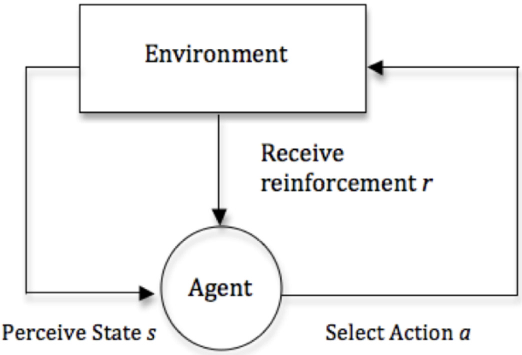


Fig. 2. The RL Paradigm: an agent repeatedly perceives the state of the environment and takes action. After each action the agent receives a reinforcement signal. The goal of the agent is to collect as much reward as possible over time.

probability distribution. After taking the chosen action i, its probability p_i is updated based on the reward $r \in \{0,1\}$, see Equation 1, first line. The other probabilities p_j (for all actions $j \neq i$) are adjusted in a way that keeps the sum of all probabilities equal to 1 ($\sum_i p_i = 1$), see Equation 1, second line. This algorithm is based on the simple idea that whenever the selected action results in a favorable response, the action probability is increased; otherwise it is decreased.

$$\begin{aligned} p_i &\leftarrow p_i + \alpha r(1-p_i) - \beta(1-r)p_i \ , \\ p_j &\leftarrow p_j - \alpha r p_j + \beta(1-r)\left(\tfrac{1}{n-1} - p_j\right) , \quad \forall j \neq i . \end{aligned} \tag{1}$$

The parameters α and β ($\alpha, \beta \in [0,1]$) are the reward and penalty learning rate. In literature, three common update schemes are defined based on the values of α and β:

- Linear Reward-Inaction (L_{R-I}) for $\beta = 0$,
- Linear Reward-Penalty (L_{R-P}) for $\alpha = \beta$,
- Linear Reward-ϵ-Penalty ($L_{R-\epsilon P}$) for $\beta \ll \alpha$.

2.2 Application to Scheduling

To apply LA to a scheduling problem we need to define the actions of all agents, the rewards and the problem's state space. We define an *action* of the LA as submitting a job to one of the parallel machines. Thus, for the problem described in Section 1 we have 6 agents: two receive product orders P_1 and P_2 and decide which 'stage-1' machine will be used. The other four agents receive partially processed jobs from a 'stage-1' machine M_{1-} and send them to a 'stage-2' machine M_{2-}. Note, the agent cannot wait to submit a job and cannot stop a job preemptively. In other settings these could be added as extra actions.

When a job j is completely finished, the two agents that decided the path of that job are notified of this. Based on the completion time C_j and the release times R_j for both stages a *reward* $r \in \{0, 1\}$ is created, see Equation 2. Note, (i) completion time is the time at which the job has finished both stage 1 *and* stage 2, and (ii) release times are the times at which the job starts stage 1 *or* stage 2 depending on the agent.

$$r = \begin{cases} 0 & \text{if } T > T_{avg}, \\ 1 & \text{otherwise}, \end{cases} \tag{2}$$

where the flowtime $T = C_j - R_j$ and T_{avg} is the average flowtime over the last n number of jobs. The larger n is the more accurate the LA's belief of the average flowtime of the jobs. The smaller n the faster the LA will adapt his belief of the average flowtime when for example a machine breaks down or the performance of a machine increases.

For this problem, it is unnecessary to define a *state-space*. From the agents' point of view the system is always in the same state.

3 WSEPT Heuristic for Stochastic Online Scheduling

We do not know of any good approximation algorithm for scheduling problems that are online, stochastic and multi-stage at the same time. For the single-stage case, however, there exists a very good heuristic: Weighted Shortest Expected Processing Time (WSEPT) [7].

It works in the following setup: orders are arriving over time and must be processed by one out of several parallel machines. The objective is to reduce the total weighted completion time ($\sum_j w_j C_j$, for all jobs j). Each time an order arrives, the WSEPT rule submits the job to the machine that is *expected* to finish it first. To this end it polls each machine for its current expected makespan (including the new job). If, for example, all jobs have equal expected processing time, and each machine the same average speed, then the expected makespan is the queuelength (including the currently processed job if any). In [7] lower bounds on the total weighted completion time ($\sum_j w_j C_j$) are given for the WSEPT heuristic.

In the next section we will compare the WSEPT and the Learning Automata in a simple single-stage scheduling task.

4 Experimental Results

4.1 WSEPT Heuristic versus Learning Automata

We ran some experiments on single-stage scheduling with $N = 4, 5$ or 6 identical machines. One scheduler receives a sequence of jobs. The joblengths are generated by an exponential distribution with average $\mu = 100$. The identical machines have unit processing speed $s_i = 1$, for $i = 1, \ldots, N$. I.e. a machine needs 100 timesteps to process an average job.

To make sure the system can actually handle the load, we set the probability of creating a job at any timestep to 95% of the total processing speed divided by the average job length: $0.95 \sum_i s_i/\mu$. To keep things easy, all jobs have unit weight $w_j = 1$.

We tested the following agents on the same sequence of orders:

RND: uniformly distributes the jobs over all machines,
WSEPT: uses the WSEPT heuristic as described in Section 3,
LA: a Learning Automaton as described in Section 2.1 with $\alpha = \beta = 0.02$.

Results

The experiments show that the LA clearly performs better than the RND scheduler. This is not at all to be expected. The optimal (but static) distribution of jobs of equal expected processing length on identical machines is the uniform distribution. Which is exactly what RND uses. However, due to the high variance in processing times, adapting the load distribution is more efficient at keeping the queues short.

Obviously, WSEPT outperforms LA. Note, the heuristic uses information which both LA and RND cannot access. The length of the queues over time show that WSEPT balances the load better: queues are 4 to 5 times shorter. On the other hand, the total weighted completion time ($\sum_j w_j C_j$) does not show huge differences between WSEPT and LA (in the order of 0.001 to 0.01).

Although the WSEPT heuristic outperforms the Reinforcement Learning approach, the LA are not useless. WSEPT requires access to more information and only works in single-stage loadbalancing problems. In the next section, we test LA in the multi-stage setting as described in Section 1.

4.2 Multi-Stage Scheduling

We look at two slightly different settings, see Table 1. Setting 1 is copied from [2]. In both cases, the average joblength is 100 and the jobrate is 1/45 for both product types P_1 and P_2. The performance is measured by total flowtime of the jobs through entire processing chain.

The first type of LA we test are Linear Reward-Inaction LA (L_{R-I}). After some time, the queues started growing indefinitely. This was caused by some automata converging prematurely to a pure strategy. I.e. they end up selecting the same action forever. This is due to the fact that L_{R-I} never penalize bad actions ($\beta = 0$). Although this may be favorable for many RL problems, it will almost never be for load-balancing. The only obvious exception is when one machine is able to process all jobs before any new order arrives.

The $L_{R-\epsilon P}$ generated better and better results when ϵ is increased. Finally, when $\epsilon = 1$ we have L_{R-P}, where penalty and reward have an equally large influence on the probabilities. This gives the best results.

The value of α and β, which determines the learning speed, seems best around 0.01 for this problem. Table 2 shows the average policy for each of the six agents. For example, the fourth agent receives jobs partially finished by machine M_{13} and distributes them over M_{23} and M_{24}.

The second setting shows that the agents take into account the time needed for a job to go through all stages. Machines M_{13} and M_{14} are 10 times faster as in the first setting. This does not increase the total system capacity, since the machines in the second stage would create a bottleneck. The result is that the first two agents still favor M_{11} and M_{12}, but slightly less. For example, the first agent in Table 2 distributes 71% of the load on M_{11} in the second setting, as opposed to 74% in the first setting.

Table 1. Processing speeds of all machines for two different settings.

Machine	M_{11}	M_{12}	M_{13}	M_{14}	M_{21}	M_{22}	M_{23}	M_{24}
Speed setting 1	3.33	2	1	1	3.33	1	1	1
Speed setting 2	3.33	2	10	10	3.33	1	1	1

Table 2. Average probabilities of all agents through an entire simulation.

Machine	M_{11}	M_{13}	M_{12}	M_{14}	M_{21}	M_{22}	M_{23}	M_{24}	M_{21}	M_{22}	M_{23}	M_{24}
Setting 1	.74	.26	.69	.31	.76	.24	.50	.50	.77	.23	.50	.50
Setting 2	.71	.29	.61	.39	.77	.23	.50	.50	.78	.22	.50	.50

5 Discussion

Following advantages of LA make them very applicable in difficult scheduling scenarios:

- They can cope with uncertainty: unknown joblengths, unknown future jobs and unknown machine speeds.
- The decisions based on the probability distribution and the updates of those distribution are very straightforward. They can be performed in a minimum of time and require only very limited resources.

The performed experiments show that LA can learn processing capacities of entire downstream chains. Note however that the rewards are delayed.

While waiting for a submitted job to be finished, other jobs must already be scheduled. In our scheduling problem, this is not a problem for the LA. When more stages would be added to the system, the LA could be equipped with so-called eligibility traces [3].

Since LA are very adaptive, it should even be possible to detect changes in processing speed, such as machine break downs.

Finally, when applying any randomization technique (such as LA) to balance a load, one is always better off with many short jobs than very few long ones (cf. the law of large numbers). It remains to be seen how large the effect of fewer but longer jobs will be in our setting.

References

1. Castillo I, Roberts CA (2001) Real-time control/scheduling for multi-purpose batch plants. Computers & Industrial Engineering,
2. Peeters M (2008) Solving Multi-Agent Sequential Decision Problems Using Learning Automata. PhD Thesis, Vrije Universiteit Brussel, Belgium.
3. Sutton R, Barto A (1998) Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA.
4. Kaelbling LP, Littman M, Moore A (1996) Reinforcement Learning: a survey. Journal of Artificial Intelligence Research 4: 237-285.
5. Watkins C, Dayan P (1992) Technical note Q-Learning. Machine Learning. Springer Netherlands 8: 279-292.
6. Narendra KS, Thathachar MAL (1974) Learning automata - A survey. IEEE Transactions on Systems, Man, and Cybernetics, SMC-4(4): 323-334.
7. Megow N, Uetz M, and Vredeveld T (2006) Models and algorithms for stochastic online scheduling. Mathematics of Operations Research, 31(3): 513-525.

Global Optimization with Expensive Functions - Sample Turbomachinery Design Application

Caroline Sainvitu, Vicky Iliopoulou and Ingrid Lepot

Cenaero, Numerical Methods and Optimization Group
Bâtiment EOLE, Rue des Frères Wright, 29
B-6041 Gosselies, Belgium - caroline.sainvitu@cenaero.be

Abstract

This contribution presents some of the tools developed at Cenaero to tackle industrial multidisciplinary designs. Cenaero's in-house optimization platform, Minamo implements mono- and multi-objective variants of Evolutionary Algorithms strongly accelerated by efficient coupling with surrogate models. The performance of Minamo will be demonstrated on a turbomachinery design application.

1 Introduction

Nowadays, with the continuous increase in computing power, a widespread practice in engineering is that of simulation-based design optimization. Indeed, design of complex engineering systems, which is synonymous with the use of accurate high-fidelity simulationmodels (e.g. Computational Fluid Dynamics (CFD) analysis or Finite Element Method (FEM)), has become a reality. However, even with today's computational power, it is rarely conceivable to thoroughly search the design space using the high-fidelity simulations. Since optimization procedures are mandatory to quickly provide optimal designs, an adequate and general answer to optimization based on computationally expensive analysis lies in the exploitation of surrogate models. *Surrogate-Based Optimization (SBO)* essentially exploits surrogates or approximations instead of the expensive analysis results to contain the computational time within affordable limits (see [4, 12]), with occasional recourse to the high-fidelity model. The performance of such methods is known to be largely dependent on the following key elements: the initial sample set used to build the surrogate model(s), the underlying optimization algorithm(s), the surrogate model(s) training and the surrogate model(s) management schemes.

This paper is structured as follows. First, the SBO methodology implemented in Minamo is exposed with a focus on the design of experiments and the surrogate modeling. Subsequently, the performance of our in-house optimization platform is demonstrated on a turbomachinery design application.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_44, © Springer-Verlag Berlin Heidelberg 2010

2 Optimization Methodology

In most engineering design optimization, every evaluation of functions involved in the problem is expensive and their derivatives are, generally, unavailable or available at a prohibitive cost. Typical optimization techniques, like gradient-based methods[11], are not applicable or not efficient in such contexts. Despite their speed of convergence, these methods are indeed known to lack space exploration. They are appropriate if the derivatives are available or can be inexpensively approximated and if a good starting point is known. Moreover, they are restricted to mono-objective problems and only permit to solve multi-objective problems by using an aggregation of the objectives with pre-defined weights for each objective. Derivative-free algorithms [3] have been developed for local optimization of computationally expensive functions, but most of the time engineers are interested by a global optimum.
For these reasons, Minamo implements mono- and multi-objective Evolutionary Algorithms (EAs)sing real coded variables. These methods are stochastic, population-based search techniques and widely used as efficient global optimizers in the engineering world. Such zero-order optimization techniques are indeed robust and able to cope with noisy, discontinuous, non-differentiable, highly non-linear and uncomputable functions. Most importantly, they also permit to simultaneously handle multiple physics as well as multiple objectives. They are also less prone to getting trapped in local optima than other optimization algorithms as gradient-based methods. Moreover, EAs provide a list of optimal solutions from which the user/engineer can choose the best design according to his/her experience (see the two families of promising designs obtained in Section 3). However one drawback of EAs is that they may suffer from slow convergence due to their probabilistic nature. As a consequence, for engineering applications involving expensive high-fidelity simulations, the CPU time required for a pure EA is usually not practical. This highlights the importance to reduce the number of calls to these simulations. Therefore, the optimization process in Minamo is significantly accelerated by the use of cheap-to-evaluate surrogate models, also known as metamodels or response surface models.

2.1 Surrogate-Based Optimization

The heart of the proposed methodology consists of a surrogate modeling optimization strategy. As already underlined, SBO refers to the idea of accelerating optimization processes by exploiting surrogates for the objective and constraint functions. An SBO design cycle consists of several major elements as shown in Figure 1. It is worth underlying the major importance of the first step, namely the problem definition and optimization specification, which can include the parameterization, the definition of the bounds, the objectives and the constraints. The second step consists of building an initial database by

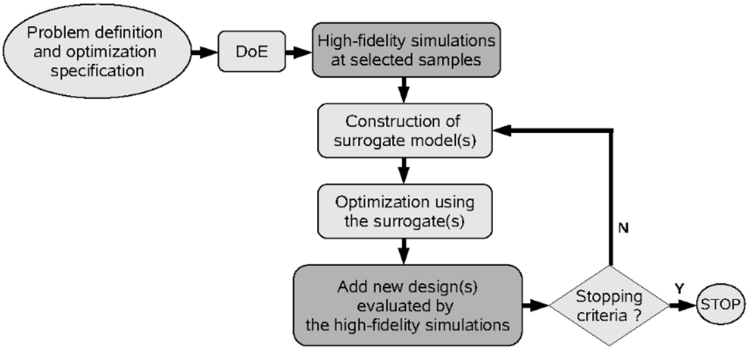

Problem definition and optimization specification
DoE
High-fidelity simulations at selected samples
Construction of surrogate model(s)
Optimization using the surrogate(s)
Add new design(s) evaluated by the high-fidelity simulations
Stopping criteria ?
N
Y
STOP

Fig. 1. Online surrogate-based optimization framework.

choosing a set of points in the design space and conducting high-fidelity simulations at these points. This exercise is called the *Design of Experiments (DoE)* Based on this DoE, surrogate models are constructed in order to build an analytical relationship between the design parameters and the expensive simulation responses. This phase provides cheap responses to be used by an optimizer. Using the surrogate models to evaluate the objective and constraint functions, an optimization is then carried out to identify the optimum, at least in the sense of the surrogates. The accurate simulation is used to evaluate the objective function and constraint values for this optimum in order to check the accuracy of the surrogates at the optimal solution. The new simulation result (and possibly simulation results at other infill points) is (are) added to the database which is therefore continuously improved with new design points, leading to increasingly accurate approximate models all along the design. This design loop is repeated until the maximum number of optimization cycles specified by the user is reached. In this contribution, an EA is employed to optimize the surrogate model(s) because this optimizer choice allows any kind of surrogate models without particular properties such as differentiability of the surrogates and also permits to deal with multiple objectives. It is important to note that our SBO scheme can incorporate the derivative information, when it is available, in different ways without any major modifications. For instance, the derivatives could be exploited directly in the construction of the metamodels. The periodic retraining of the surrogates ensures that the metamodels continue to be representative of the newly-defined search regions. Furthermore, in order to obtain a better approximate solution, a framework for managing surrogate models is used. Based on effectiveness of approximations, a *move-limit*procedure adapts the range of the variables along the design process, focusing the optimization search on smaller regions of the design space and exploiting local models. As the optimization proceeds, the idea is to enlarge or restrict the search space in order to refine the candidate optimal region. The main advantage of this is that it assures that the optimization does not generate new points in regions where the surrogates are not valid.

In order to guarantee diversity in the population, Minamo also exploits a *merit function*which is combined with the objective function of each candidate solution [15]. This function takes into account the average distance of a candidate with the other candidate solutions, and favors the solutions far away from their neighbours. A good approach for SBO seeks a balance between exploitation and exploration search, or refining the approximate model and finding the global optimum. Our strategy also allows the addition of several new design points evaluated in parallel at each cycle. Typically, the design point coming from the optimization of the surrogate(s) is added and other update points may be appended to the database as well. Using several research criteria per iteration allows to combine exploitation (optimization of the approximate function) and exploration (to systematically aim for a better global capture) within a single iteration, speeding up the restitution time of the optimization. In other words, although most of the optimizers based on the Kriging model use one single refinement criterion per iteration (such as the Expected Improvement criterion), Minamo is capable to proceed by iteratively enhancing with more than one point per iteration by using a balancing between model function minimization and uncertainty minimization. This process builds upon multiples high-fidelity simulations (e.g. CFD runs) in parallel.

The efficiency of our SBO algorithm is illustrated in the search of the global minimum of the Ackley function which is a well-known multimodal function. The left plot of Figure 2 depicts the function with 2 design parameters, while the optimization has been carried out on the same function but generalized to 5 dimensions within $[-2, 2]$ for every parameter. The optimization is first performed using the EA alone, with a population of 50 individuals for 40 generations (*i.e.* 2000 function evaluations). These results are compared with those obtained by the method combining the surrogate model with the EA. An initial database comprising 20 sample points is used and then only 100 design iterations are performed. The convergence histories are displayed in the right plot of Figure 2. The results clearly indicate that, for a given fixed number of actual function evaluations, the SBO approach drastically outperforms a pure EA optimization using actual function evaluations.

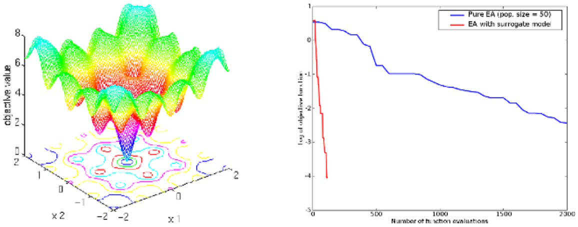


Fig. 2. Ackley function and convergence history comparison.

In Minamo, particular attention has been paid to handling simulation failures *i.e.* experiments where the simulation fails to converge. Indeed, when optimization is carried out using high-fidelity simulations, it is an inevitable fact that not all simulations provide reliable results (due to an inappropriate mesh, failed geometry regeneration, etc.). The best practice is to try to make the simulation chain as robust as possible, and let the optimizer take care of the simulation failures. In Minamo, the simulation failures are recorded for every sample point through a boolean response, called the success/failure flag. Two separate surrogate models are maintained simultaneously, namely the response model(s) (used for the evaluation of objective and constraint functions) and the failure prediction model (used for the evaluation of the simulation failure). The idea is to bias the search away from failed sample points by penalizing, via a constraint, regions containing simulation failures.

2.2 Design of Experiments

The *design of experiments* is the sampling plan in the design parameter space. This is a crucial ingredient of the SBO procedure, especially when the function evaluations are expensive, because it must concentrate as much information as possible. The qualities of surrogate models are mainly related to the good choice of the initial sample points. The challenge is in the definition of an experiment set that will maximize the ratio of the model accuracy to the number of experiments, as the latter is severely limited by the computational cost of each sample point evaluation. Minamo features various DoE techniques aiming at efficient and systematic analysis of the design space. Besides classical space-filling techniques, such as Latin Hypercube Sampling (LHS), Minamo's DoE module also offers Centroidal Voronoi Tessellations (CVT) and Latinized CVT (LCVT) [14]. A drawback of LHS is that sample points could cluster together due to the random process by which the points are generated. CVT efficiently produces a highly uniform distribution of sample points over large dimensional parameter spaces. However, a CVT dataset (in a hypercube) has the tendency for the projections of the sample points to cluster together in any coordinate axis. LCVT technique tries to achieve good dispersion in two opposite senses: LHS and CVT senses. The idea is to compute a CVT dataset and then apply a Latinization on this set of points. Latinizing a set of points means transforming it into another set of neighbouring points that fulfills the Latin hypercube property. The aim of this Latinization of CVT sample points is to improve the discrepancy of the set of points. LCVT technique has both lower discrepancy than pure CVT and higher volumetric uniformity than pure LHS (see Figure 3). The discrepancyis a measure of a point set's uniformity of projection onto all the coordinate axes. As uniformity increases, discrepancy decreases. All these space-filling techniques, independent of the design space dimensionality and of the type of surrogates, constitute good first choices to generate an *a priori* sample set in large dimensions. The DoE can be generated quickly by making use of massively parallel computers.

Since the computation of the response functions can typically take several

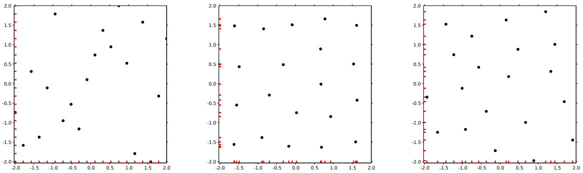


Fig. 3. LHS, CVT and LCVT, respectively, sample sets showing discrepancies of point projections (in red) onto coordinate axes.

hours on tens of computational cores, next to LCVT implementation, further research effort has been put to achieve a good accuracy of approximate models with a reasonable number of samples by incorporating function knowledge. In order to further tailor the sampling and to better capture the responses underlying physics, Minamo exhibits an *auto-adaptive DoE* technique. The idea is to locally increase the sampling intensity where it is required, depending on the response values observed at previous sample points. Such auto-adaptive techniques are also known as capture/recapture sampling or *a posteriori* sequential sampling (see [8, 9]). They incorporate information on the true function in sample distribution, explaining the term *a posteriori*. The aim is to automatically explore the design space while simultaneously fitting a metamodel, using predictive uncertainty to guide subsequent experiments. Our method consists in iteratively refining the sample dataset where the model exhibits its maximum of error, with the error indicator provided by a Leave-One-Out (LOO) procedure [10]. The use of adaptive sampling helps shorten the time required for the construction of a surrogate model of satisfactory quality. Figure 4 shows the performance of this sampling technique on a mathematical function with 2 design parameters. It allows to directly and correctly identified the region of the global optimum, whereas, using the same type of model and the same number of samples from LHS, the optimum is misplaced and the optimization will therefore be stuck in a local optimum of the original function.

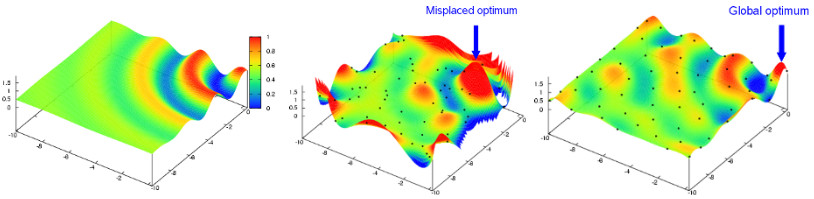


Fig. 4. The exact function, the model built using 60 LHS points and the one with 60 points generated by auto-adaptive LCVT sampling technique, respectively.

2.3 Surrogate Modeling

The challenge of the surrogate modeling is similar to that of the DoE: the generation of a surrogate that is as good as possible, using as few expensive evaluations as possible. Polynomial fitting surfaces are generally not well-suited for high dimensional and highly multimodal problems. Several non-linear data-fitting modeling techniques can be used to build the surrogates, e.g. artificial neural networks, Radial Basis Functions (RBF) networks, Kriging or support vector machines [2]. Contrary to polynomial models, these techniques have the advantage of decoupling the number of free parameters with respect to the number of design parameters. Furthermore, they can describe complex and multimodal landscapes. The Minamo surrogate module offers several generic interpolators such as RBFetworks, ordinary and universal Kriging. In the training process, a trade-off must be attained between the accuracy of the surrogates and their computational cost. For our RBF network, the models are generated without the user's prescription of the type of basis function and model parameter values. Our method autonomously chooses the type of basis functions (between Gaussian or multiquadric) and adjusts the width parameter of each basis function in order to obtain an accurate surrogate model. RBF implementation is built on the efficient LOO procedure proposed by Rippa [13], while for our Kriging implementation, the parameters defining the model are estimated by solving the log-likelihood estimation problem using our EA as this problem is known to be multimodal.

3 Sample Turbomachinery Design Application

he performance of Minamo is demonstrated with the multi-point aerodynamic optimization of a non axisymmetric hub for a high pressure compressor single-row rotor blade. This work has been performed within the NEWAC project (NEW Aero engine Core concepts, project co-funded by the European Commission within the Sixth Framework Program for Research and Technological Development), aimed at technological breakthroughs for the field of aero engines efficiency and emissions. The objective was to identify the hub endwall parameter values that create a non axisymmetric hub endwall leading to significant global losses reduction with respect to the axisymmetric case at design point, while preserving the total-to-total pressure ratio close to stall.

Computer-Aided Design (CAD)systems have become an entire and critical part of the design process in many engineering fields. Therefore, it is of prime importance to exploit the native CAD system and CAD model directly within the design loop in order to avoid translation, manipulation/regeneration errors resulting from different geometry kernels. For the works presented in [6, 7], the CAPRI CAD integration middleware [5] has been exploited to provide direct CAD access without manual interventions in the CAD system during the optimization loops. Based on CAPRI, an object-oriented framework has

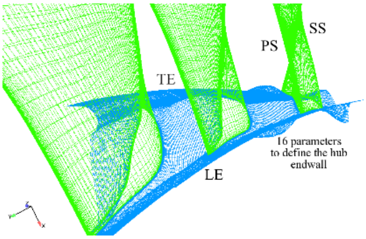


Fig. 5. Hub and blade surface mesh for the non axisymmetric hub optimizations.

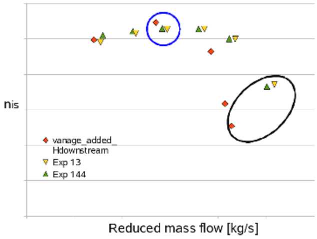


Fig. 6. Performance map with the baseline axisymmetric and optimized hub endwalls (individuals 13 and 144).

been developed for Minamo to: interact with the underlying CAD system transparently, modify the shape design variables, regenerate the CAD model and provide an updated native geometry representation to be used for the analyses.

The non axisymmetric hub surface has been parameterized under CATIA V5 and imported into the AutoGrid5 mesh generation tool for meshing purposes. The flow computations have been performed with $3D$ Reynolds-Averaged Navier-Stokes simulations using the elsA code developed at ONERA [1]. These tools have then been coupled with Minamo. Most importantly, this optimization chain can be easily applied to any blade/endwall geometry with only minor adjustments. The hub endwall has been parameterized with 16 design parameters, that can create circumferential $3D$ bumps and hollows that follow the blade curvature. The 2.2 million grid points mesh deformation at the hub is illustrated in Figure 5. Reference [6] has focused on the description of the optimization chain and methodology that have been set up, with presentation of the mono-point optimization results. Indeed, before the multi-point optimization was conducted, only one operating point was considered in order to gain first experience with limited computational cost and let the optimizer as free as possible to explore the search space. The objective was to maximize the isentropic efficiency of the compressor while imposing no additional operational or manufacturing constraints. The initial DoE was performed with LHS and held 97 sample points among which 74 experiments were considered as a success ($\approx$ 4.5 times the number of parameters). The type of surrogate models used was RBF network. This first optimization allowed indentification of a non axisymmetric surface yielding an isentropic efficiency gain of about 0.4%. This increase may be seen as quite important, when considering that the geometry changes very locally, only at the hub endwall. However, the total-to-total pressure ratio decreased by 0.4%. This highlights one of the main drawbacks of the mono-point optimization that lead to the specification of a second robust optimization [7], now considering two operating points. The first operating point was again chosen close to peak efficiency (design point) and the second point was chosen closer to the stall region (stall point), in order to better represent the performance map of the compressor. The objective was to maximize the efficiency at the design point while preserving at least the same total-to-total pressure ratio at the stall point. The mass flow at design point was also constrained to remain within 0.5% of the reference axisymmetric flow value and some manufacturing constraints were also imposed (limited digging/removal of material). The number of success experiments for the DoE was 71 over the 97 experiments. The most interesting sample of this new DoE appeared to be the hereafter noted individual 13, which yielded an increase in terms of isentropic efficiency of about 0.39% with respect to the axisymmetric case, while it increased the total-to-total pressure ratio by 0.31% at stall without exceeding the limit on the mass flow at design point. A series of promising individuals were then found along the optimization phase in itself. Some of them were quite close in terms of performance and shape to the best DoE experiment. However, most interestingly, a second family of promising designs, quite different and somewhat smoother in terms of $3D$ surface definition, was found. This illustrates the ability of the EA to globally search the space and possibly offer a panel of solutions to the designer. Let us point out one design in this second family, individual 144, which yields an increase of efficiency of 0.35% with respect to the reference axisymmetric case, while increasing the total-to-total pressure ratio by 0.1% at stall without exceeding the limit on the mass flow at design point and satisfying the manufacturing constraints (this was not the case of individual 13). Interestingly also, individual 134 appeared quite close in shape to the interesting designs found from the mono-point optimization. The isentropic efficiency curves of the rotor with the optimized non axisymmetric hub endwalls and with the baseline axisymmetric hub are shown in Figure 6 for the two-point optimization results. The pressure contours on the blade suction side are displayed in Figure 7 and indicate that the main loss mechanism results from the shock and acceleration system along the blade suction side. The different non-axisymmetric endwall geometries decreased the losses until 50% of the blade span in the region just downstream the blade trailing edge at the hub compared to the reference axisymmetric hub geometry. The optimized designs decreased the losses downstream the shock, during the flow acceleration between 10 and 50% span.

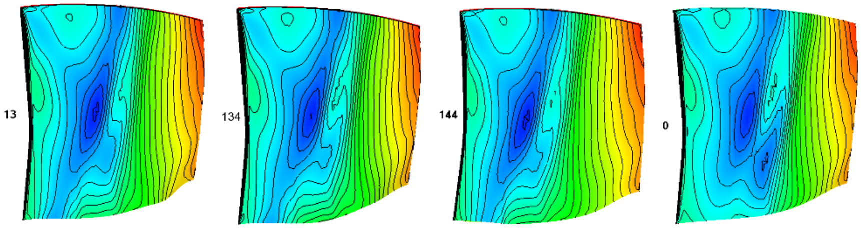


Fig. 7. Pressure contours on the blade suction side at design point for the two-point optimization - Non axisymmetric individuals 13, 134, 144 and axisymmetric case 0.

4 Conclusion

This paper has presented our in-house optimization platform, Minamo, implementing an SBO scheme. Its capabilities have been demonstrated in a truly industrial framework with an aerodynamic design optimization. With Minamo, multi-physics multi-criteria designs tackling over a hundred parameters within a heavily constrained setting are successfully handled on a day-to-day basis.

References

1. Cambier L, Gazaix M (2002) elsA: an Efficient Object-Oriented Solution to CFD Complexity. Proceedings of the 40th AIAA Aerospace Sciences Meeting and Exhibit, Reno, USA
2. Chen V C P, Tsui K-L, Barton R R, Meckesheimer M (2006) A review on design, modeling and applications of computer experiments. IIE Transactions, Volume 38, Pages 273-291
3. Conn A R, Scheinberg K, Toint Ph L (1997) Recent progress in unconstrained nonlinear optimization without derivatives. Mathematical Programming, Volume 79, Pages 397-414
4. Forrester A I J, Keane, A J (2009) Recent advances in surrogate-based optimization. Progress in Aerospace Sciences, Volume 45, Issues 1-3, Pages 50-79
5. Haimes R, Follen G (1998) Computational Analysis Programming Interface. International Conference on Numerical Grid Generation in Computational Field Simulations, University of Greenwich, United Kingdom
6. Iliopoulou V, Lepot I, Geuzaine P (2006) Design optimization of a HP compressor blade and its hub endwall. ASME Paper GT2008-50293, Berlin
7. Iliopoulou V, Mengistu T, Lepot I, (2008) Non Axisymmetric Endwall Optimization Applied to a High Pressure Compressor Rotor Blade. AIAA-2008-5881, 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Victoria, British Columbia, Canada
8. Jin R, Chen W, Sudjianto A (2002) On Sequential Sampling for Global Metamodeling in Engineering Design. DETC-DAC34092, 2002 ASME Design Automation Conference, Montreal, Canada
9. Jones D R , Schonlau M, Welch W J (1998) Efficient Global Optimization of Expensive Black-Box Functions. Journal of Global Optimization, Volume 13, Number 4, Pages 455-492
10. Meckesheimer M, Barton R R, Simpson T W, Booker A J (2002) Computationally Inexpensive Metamodel Assessment Strategies. AIAA Journal, Volume 40, Number 10, Pages 2053-2060
11. Nocedal J, Wright S J (1999) Numerical Optimization. Springer, New York
12. Queipo N V, Haftka R T, Shyy W, Goel T, Vaidyanathan R, Tucker P K (2005) Surrogate-based analysis and optimization. Progress in Aerospace Sciences, Volume 41, Issue 1, Pages 1-28
13. Rippa S (1999) An algorithm for selecting a good value for the parameter c in radial basis function interpolation. Advances in Computational Mathematics, Volume 11, Number 2-3, Pages 193-210

14. Saka Y, Gunzburger M, Burkardt J (2007) Latinized, improved LHS, and CVT point sets in hypercubes. International Journal of Numerical Analysis and Modeling, Volume 4, Number 3-4, Pages 729-743
15. Torczon V, Trosset M W (1998) Using approximations to accelerate engineering design optimization. AIAA-98-4800 in the Proceedings of the 7th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization

Adaptive Alternating Minimization for Fitting Magnetic Resonance Spectroscopic Imaging Signals

Diana M. Sima, Anca Croitor Sava, and Sabine Van Huffel

Katholieke Universiteit Leuven, Department of Electrical Engineering,
Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium
diana.sima@esat.kuleuven.be, anca.croitor@esat.kuleuven.be,
sabine.vanhuffel@esat.kuleuven.be

Summary. In this paper we discuss the problem of modeling Magnetic Resonance Spectroscopic Imaging (MRSI) signals, in the aim of estimating metabolite concentration over a region of the brain. To this end, we formulate nonconvex optimization problems and focus on appropriate constraints and starting values for the model parameters. Furthermore, we explore the applicability of spatial smoothness for the nonlinear model parameters across the MRSI grid. In order to simultaneously fit all signals in the grid and to impose spatial constraints, an adaptive alternating nonlinear least squares algorithm is proposed. This method is shown to be much more reliable than independently fitting each signal in the grid.

1 Introduction

Magnetic Resonance (MR) is widely used in hospitals to distinguish between normal and abnormal tissue. Among the established MR techniques, Magnetic Resonance Imaging (MRI) has a high spatial resolution and is able to provide detailed pictures reflecting differences in tissue, but this technique has a low spectral resolution since it mainly represents the density of water. A second important technique is Magnetic Resonance Spectroscopy (MRS), which provides a signal from a small localized region called voxel, and has a high spectral resolution, *i.e.*, many metabolites (chemicals) are identifieble from an MR spectrum. Thirdly, Magnetic Resonance Spectroscopic Imaging (MRSI) is a multi-voxel technique that combines imaging and spectroscopy in order to provide a trade-off between spatial and spectral resolution.

An MRS signal is a complex-valued time-domain signal y induced by a population of nuclei immersed in a magnetic field after applying a radio-frequency pulse. This time-domain signal is a superposition of many exponentially decaying components. The problem of metabolite quantification amounts to fitting a certain model to the MRS signal.

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_45, © Springer-Verlag Berlin Heidelberg 2010

In this paper we focus on modeling and fitting MRSI data, which is a challenging computational problem because of relatively low spectral resolution and high level of noise in the signals. To overcome low data quality, it is important to formulate appropriate constraints and to use good starting values in the nonconvex metabolite quantification optimization problems. In particular, we focus on the spatial smoothness of the nonlinear model parameters across the MRSI grid. In order to simultaneously fit all signals in the grid and to impose spatial constraints, an alternating nonlinear least squares algorithm is proposed. This method is adaptive, in the sense that each subproblem may tune some hyperparameters at run-time, instead of always keeping them fixed.

The paper is organized as follows. In Section 2, the state-of-the-art model for MRS signals, as well as details on the optimization methods used for single-voxel MRS signals, are presented. Further, we pursue in Section 3 the topic of MRSI data quantification, where we first motivate the need to impose spatial relations between the grid's signals; then, the optimization problem and solution method for simultaneous MRSI data quantification are described. Finally, numerical illustrations on simulated noisy MRSI grids are found in Section 4.

2 Metabolite quantification of MRS signals

2.1 MRS model

An MRS signal can be modeled in the time-domain as a sum of complex damped exponentials $\sum_{k=1}^{K'} a_k \exp(j\phi_k)\exp(-d_k t + \ 2\pi j f_k t)$, where a_k are amplitudes, ϕ_k phases, d_k damping factors and f_k frequencies, $j = \sqrt{-1}$ and t denotes a particular time instant among the discrete measuring times $t_0, \ldots, t_{m-1}$. In this parametric model, the frequencies are characteristic to the metabolites under investigation, while the amplitudes are proportional to the concentration of the respective molecule.

Due to the fact that many metabolites resonate in a well-defined pattern at more than one frequency, depending on the molecular configuration, a more sophisticated model is currently used for MRS signals,

$$\widehat{y}(t) = \sum_{k=1}^{K} a_k \exp(j\phi_k)\exp(-d_k t + 2\pi j f_k t)\, v_k(t), \tag{1}$$

where we point out that v_k, with $k = 1, \ldots, K$, denotes a pure metabolite signal, which can be measured *in vitro* or simulated using quantum mechanical knowledge. In this case the factor $\exp(j\phi_k)\exp(-d_k t + 2\pi j f_k t)$ accounts for corrections to the ideal metabolite signal v_k, such as small frequency shifts f_k, small damping corrections d_k and phase corrections ϕ_k, while a_k stands for the total amplitude of metabolite k.

2.2 Model fitting

Metabolite quantification amounts to a nonlinear least squares problem of fitting model (1) to a measured signal $y(t)$.[1] Figure 1 (left) shows a typical basis set of metabolite spectra v_k that can be used for fitting *in vivo* measured MRS signals from the human brain. Figure 1, bottom right, illustrates the fitting of a noisy signal with the metabolite basis set; to this end, the metabolite spectra are appropriately modified, as shown in Figure 1, top right, by broadening the peaks (*i.e.*, increasing d_k), by slightly shifting the spectra along the frequency axis (with f_k Hz), and by scaling each of them to an appropriate amplitude a_k.

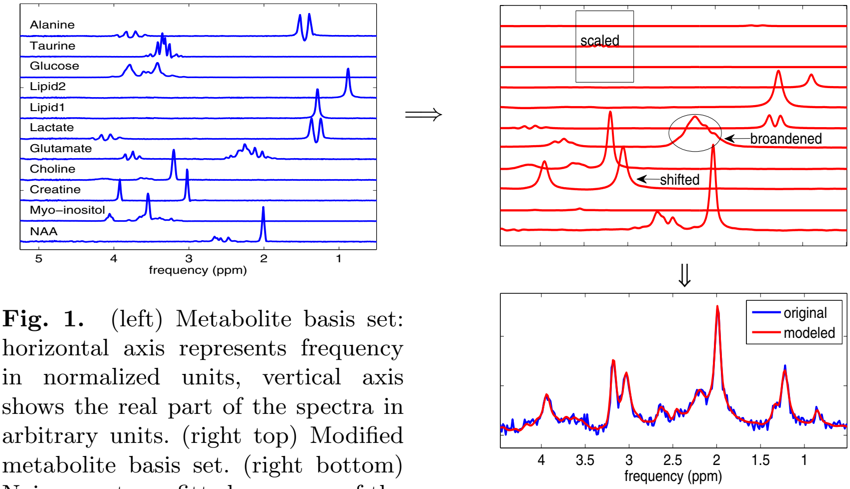


Fig. 1. (left) Metabolite basis set: horizontal axis represents frequency in normalized units, vertical axis shows the real part of the spectra in arbitrary units. (right top) Modified metabolite basis set. (right bottom) Noisy spectrum fitted as a sum of the modified metabolite basis set.

The nonlinear least squares problem mentioned above involves also bounds on the considered parameters, which come from the physical meaning of these parameters. In mathematical terms, this problem reads:

$$\min_{\substack{a_k,\phi_k,d_k,f_k\\ k=1,\ldots,K}} \|y-\widehat{y}\|^2 \quad \text{s.t.} \quad a_k \geq 0,\ \phi_k \in [0,2\pi], d_k \in (-\epsilon_d,\epsilon_d),\ f \in (-\epsilon_f,\epsilon_f) \tag{2}$$

It is important to notice the two important hyperparameters ϵ_d and ϵ_f, which specify the allowed variation of the damping corrections and of the frequency

[1] There are several acquisition conditions that lead to distortions or artifacts of the considered model (1) and for which specialized preprocessing steps exists. They will not be discussed in this paper; see, *e.g*, [12] for more details.

shifts, respectively. Since *in vivo* spectra may be quite different from each other, there are no predetermined optimal values for these hyperparameters, however such bounds are needed in order to preserve the physical meaning of each metabolite spectrum. Their chosen values might be critical for the estimated model parameters (a_k, etc.).

2.3 Variable projection approach

In the considered model (1), the *complex amplitudes* $\alpha_k = a_k \exp(j\phi_k)$ appear as coefficients of a linear combination of nonlinear functions in the parameters d_k, f_k. Thus, for any fixed values of the *nonlinear parameters* d_k, f_k, $k = 1, \ldots, K$, one can obtain corresponding optimal values for all α_k using linear least squares. The constraints $a_k \geq 0$, $\phi_k \in [0, 2\pi]$ are then readily satisfied if we take $a_k = |\alpha_k|$ and $\phi_k = \text{angle}(\alpha_k)$.

The variable projection approach [4, 11] is an optimization framework where the coefficients α_k are projected out, such as to obtain an optimization problem only in the remaining nonlinear variables. The projected functional will be denoted $\phi(\theta)$, where $\theta \in \Re^{2K}$ stands for the vector of parameters $d_1, \ldots, d_K, f_1, \ldots, f_K$. Function and Jacobian evaluations needed by optimization solvers such as Gauss-Newton, Levenberg-Marquardt, or trust region, are slightly more computationally expensive than for the original problem formulation. Still, it is well known and proven by theory [10] and practice that variable projection always converges in less iterations than the original full functional approach. This includes convergence in cases when the full functional approach diverges. Another advantage of this formulation is that no starting values are needed for the linear parameters, and that the number of parameters is halved.

The Levenberg-Marquardt algorithm [6] applied to the variable projection functional is implemented in the quantification method AQSES (Accurate Quantification of Short Echo-Time MRS Signals) [8]. The starting values for the nonlinear parameters d_k and f_k are set by default in AQSES to zero, with the motivation that d_k and f_k represent *small corrections* to the metabolite profiles in the basis set.

3 Metabolite quantification of MRSI signals

3.1 Characteristics of MRSI data

MRSI signals can be modeled with the same mathematical formulation as the MRS signals (1). A straightforward approach to quantify metabolites in a grid of MRSI voxels would be to apply a single-voxel quantification method, such as AQSES, to each signal in the grid individually. As opposed to single-voxel measurements, the MRSI signals usually have a much lower quality, due to the spatial/spectral trade-off for the available measuring time. Thus, they are more prone to quantification errors, since metabolites present in low concentration are almost embedded in noise. Moreover, a lower spectral resolution also implies that metabolite components become more strongly overlapping in frequency.

It is obvious that supplementary information expressed as constraints on the optimization parameters would be very valuable in analysing this type of data. Since MRSI signals are obtained during a single scan using a certain acquisition protocol, many characteristics of the signals within the same grid are related [3]. Differences in the signals may appear due to two main causes: the heterogeneity of the tissue under investigation, and the magnetic field applied in the scanner, which cannot be kept perfectly constant over the whole volume under investigation.[2] In particular, the damping factors and frequency location of each individual exponential decay are directly related to the local magnetic field. Assuming there are no abrupt changes in the magnetic field, the damping and frequency parameters exhibit smooth maps over the considered MRSI grid.

3.2 Smoothness of parameter maps

Smoothness of a 2D parameter map can be locally measured at every voxel (ℓ, κ) in the grid by using the parameter value at the current location and the values in a certain neighborhood. We denote that two voxels are neighbors by $(\ell_1, \kappa_1) \sim (\ell_2, \kappa_2)$. Because MRSI grids are rather coarse, we usually focus on 3×3 regions with the current voxel (ℓ, κ) in the center. When (ℓ, κ) is on the border of the MRSI grid, only the available neighbors are used. A possible measure for the smoothness at point (ℓ, κ) is given by the first order difference norm

$$\sum_{(i,j)\sim(\ell,\kappa)} (p_{\ell\kappa} - p_{ij})^2, \tag{3}$$

where p stands for any of the parameters d_k or f_k, for any k. Second order formulas are also possible, such as the second order differences

$$(2p_{\ell\kappa} - p_{\ell-1,\kappa} - p_{\ell+1,\kappa})^2 + (2p_{\ell\kappa} - p_{\ell,\kappa-1} - p_{\ell,\kappa+1})^2, \tag{4}$$

$$(4p_{\ell\kappa} - p_{\ell-1,\kappa} - p_{\ell+1,\kappa} - p_{\ell,\kappa-1} - p_{\ell,\kappa+1})^2, \tag{5}$$

3.3 Simultaneous optimization of MRSI signals

A complete optimization problem for fitting all signals in the MRSI grid and, simultaneously, penalizing all the parameter maps for smoothness (with, *e.g.*, a penalty of type (3)) is formulated as (see also Kelm [5])

[2] Other causes of spectral differences could be differences in temperature or in pH, but we assume them constant over the grid.

$$\min_{\Theta \in I} \sum_{\ell,\kappa} \phi_{\ell\kappa}(\theta_{\ell\kappa}) + \sum_{(i,j)\sim(\ell,\kappa)} \lambda_{(i,j),(\ell,\kappa)} \| W(\theta_{\ell\kappa} - \theta_{ij}) \|_2^2, \tag{6}$$

where Θ stands for the entire set of parameters $\theta_{\ell\kappa} \in \Re^{2K}$, for all voxels (ℓ, κ), and I denotes the box defined by the hyperparameters ϵ_d, ϵ_f. Moreover, the diagonal $2K \times 2K$ matrix W is used to account for different scaling of the d_k and f_k parameters in $\theta_{\ell\kappa}$, and the scalars $\lambda_{(i,j),(\ell,\kappa)}$ are regularization hyperparameters that affect the trade-off between data fitting and parameter map smoothing.

This optimization problem is highly dimensional, having $2KMN$ variables, where $M \times N$ is the grid size. (In practice we may have grids of at least 16×16 voxels and at least 10 metabolite signals in the basis set, leading to a total of more than 5000 nonlinear variables.) However, the objective function is a sum of squares, where each term contains only a few variables. Assuming all variables fixed, except for the vector $\theta_{\ell\kappa}$, we obtain tractable subproblems of the form

$$\min_{\theta_{\ell\kappa} \in I_{\ell\kappa}} \phi_{\ell\kappa}(\theta_{\ell\kappa}) + \sum_{(i,j)\sim(\ell,\kappa)} \lambda_{(i,j),(\ell,\kappa)} \| W(\theta_{\ell\kappa} - \theta_{ij}) \|_2^2, \tag{7}$$

with $I_{\ell\kappa}$ denoting the box corresponding to the vector $\theta_{\ell\kappa}$. Thus, the total optimization problem (6) is a natural candidate for an alternating minimization procedure, where subproblems of the type (7) are solved for each voxel in several sweeps through the grid, until convergence.

Remark 1. In a statistical setting, this type of alternating minimization has been introduced in the field of computer vision under the name *iterated conditional modes* (ICM). An extension of ICM to MRSI data is proposed in [5] under the name block-ICM, where instead of minimizing only over $\theta_{\ell\kappa}$, each subproblem takes a set of parameters corresponding to a neighborhood of voxels as free variables.

3.4 Adaptive alternating minimization

Alternating minimization algorithms are known to converge under very mild conditions [9, 2]. Recently, convergence properties have been analyzed for the situation when the problem statement slightly changes from sweep to sweep [7]. In [7] the variables are partitioned in only two sets, while here we apply adaptive alternative minimization with a large number of subsets (one subset per voxel). Slight changes in problem formulation are expressed, in our case, as modifications of the hyperparameters of the problem. These are, essentially, the bounds on d_k and f_k, which define at each sweep $w = 1, 2, \ldots$ a box $I^w_{\ell,\kappa}$ for each voxel (ℓ, κ). Thus, the new subproblem at sweep w for the voxel (ℓ, κ) is very similar to (7), except that the box constraint may vary at each sweep, and so do the regularization factors. Updates for the box constraints, for the regularization factors and for the starting values of each subproblem are proposed in [1].

4 Numerical results

In this section we illustrate several aspects of the new method on realistically simulated signals. The simulated signals follow model (1), where 11 *in vitro* measured metabolite profiles v_k are used, and the model parameters take biologically relevant values.[3] In order to better approximate MRSI situations, we artificially smoothen the parameter maps of the nonlinear variables. We set random, but realistic values for the amplitude maps, except in the case of two metabolites: for the first, we create a smooth map and for the second a map with an abrupt change in value. This is done for the purpose of checking whether the method is able to capture such specific situations, although the amplitudes are not explicitly constrained. The signals are finally perturbed with additive white noise with various signal-to-noise ratios. The size of the simulated grids is 5×5, the considered neighborhoods are 3×3, and only maximum 4 neighbors (up, down, left, right) are considered when imposing spatial constraints. The spatial constraint in these simulations involves the second order difference (5). Results with the first order difference (3) are comparable, but a bit less suitable for this particular simulation with very smooth parameter maps for the nonlinear variables.

Figure 2 depicts the estimated amplitude values for a grid of signals, and Figure 3 the corresponding frequency shifts, when the signals contain a high level of noise (SNR = 5). We clearly see that the results of the multi-voxel approach are much closer to the true simulated values compared to the single-voxel based method AQSES. For lower noise levels the differences are not as pronounced, since in that case AQSES performs already well enough.

Further we illustrate in Figure 4 the effect of the hyperparameters ϵ_d, ϵ_f. These bounds are computed at each sweep as the median value of the corresponding parameters from the neighboring voxels plus/minus a fraction of the previous length of the interval.

A final illustration of the importance of the considered box constraints is given by the contour plot in Figure 5. All parameters are set to the optimal values computed by the new method, except for two frequency shifts, f_2 and f_4, corresponding to metabolites that partially overlap in frequency. The projected objective function, although regularized with the smoothness penalty terms, and having excellent values for 20 out of 22 model parameters, is highly nonconvex. Still, the obtained optimal solution is very close to the minimum and also close to the true simulated values.

5 Conclusions

We discussed an alternating minimization algorithm with varying values for the hyperparameters, applied to the simultaneous, spatially constrained fitting

[3] See [8] for more details on the measured metabolite profiles and on how meaningful values for the model parameters are obtained.

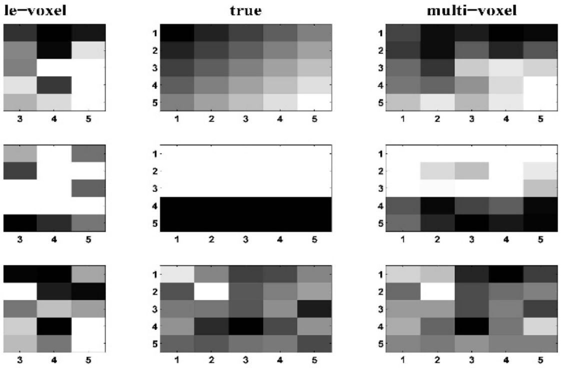


Fig. 2. Amplitude values on a 5 × 5 grid for a selection of 3 out of 11 metabolites (the 3 rows of grids), namely two metabolites with smooth and abrupt amplitude maps, and a third one with random entries. The middle column corresponds to the true values, while the left and right columns correspond to the estimated amplitudes provided by the single-voxel and the multi-voxel approaches, respectively.

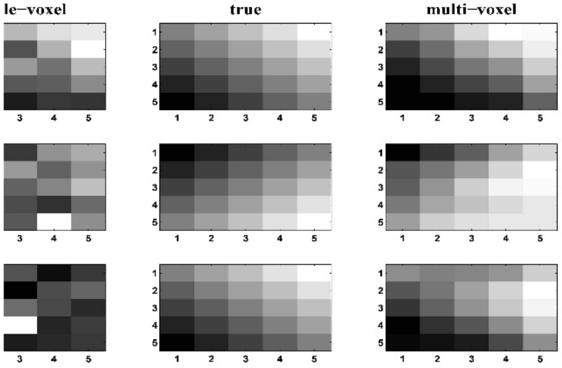


Fig. 3. Frequency values for the same example as in Figure 2. Damping maps are similar, although not shown here.

of Magnetic Resonance Spectroscopic Imaging signals. This approach is more accurate than individually fitting each signal in the grid. Still, some issues must be further studied, such as what smoothness measure is more appropriate for *in vivo* data, or how to automatically safeguard against decreasing the constraint box too much; ideas from trust region methods could be adapted for this purpose. Finally, the relevance of this approach to clinical data obtained from brain tumor patients is being evaluated in [1].

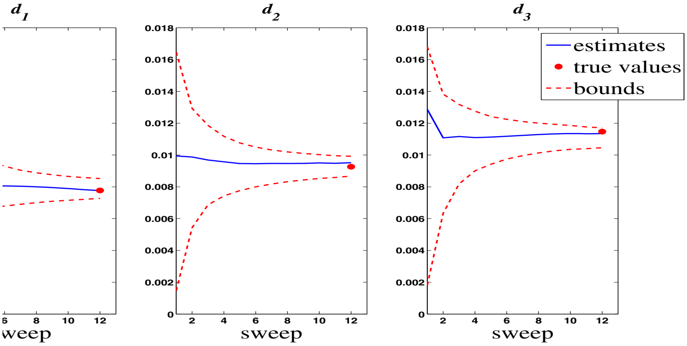


Fig. 4. Convergence of three damping estimates corresponding to the middle voxel of a 5 × 5 simulated MRSI grid with SNR=10. The estimated values for 12 sweeps are plotted together with the corresponding upper and lower bounds; the true values are also shown as big dots.

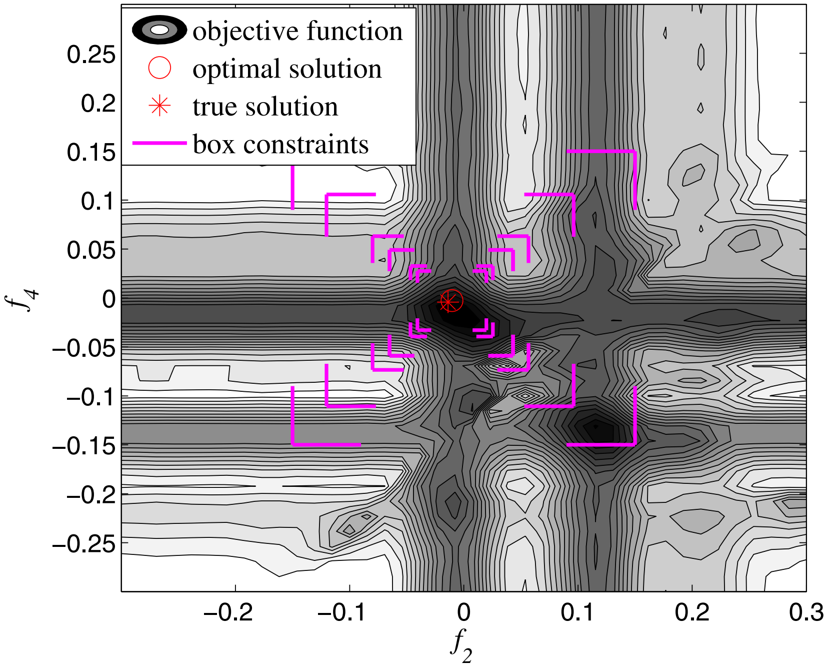


Fig. 5. Contour plot for the objective function projected onto the (f_2, f_4)-plane of a subproblem corresponding to the voxel $(1, 1)$ during the last sweep of the multi-voxel method, in a simulated MRSI grid with SNR=10. True and estimated parameters are shown as a star and a circle, respectively. The successive box constraints are also sketched.

Acknowledgments

D.M. Sima is a postdoctoral fellow of the Fund for Scientific Research-Flanders. S. Van Huffel is a full professor at the Katholieke Universiteit Leuven, Belgium. Research supported by

- Research Council KUL: GOA-AMBioRICS, GOA MaNet, CoE EF/05/006 Optimization in Engineering (OPTEC), IDO 05/010 EEG-fMRI, IDO 08/013 Autism, IOF-KP06/11 FunCop, several PhD/postdoc and fellow grants;
- Flemish Government:
  - FWO: PhD/postdoc grants, projects, G.0519.06 (Noninvasive brain oxygenation), FWO-G.0321.06 (Tensors/Spectral Analysis), G.0302.07 (SVM), G.0341.07 (Data fusion), research communities (ICCoS, ANMMM);
  - IWT: TBM070713-Accelero, TBM070706-IOTA3, PhD Grants;
- Belgian Federal Science Policy Office IUAP P6/04 (DYSCO, 'Dynamical systems, control and optimization', 2007-2011);
- EU: ETUMOUR (FP6-2002-LIFESCIHEALTH 503094), Healthagents (IST200427214), FAST (FP6-MC-RTN-035801), Neuromath (COST-BM0601)
- ESA: Cardiovascular Control (Prodex-8 C90242)

References

1. A. Croitor Sava, D.M. Sima, J.B. Poullet, A. Heerschap, and S. Van Huffel, *Exploiting spatial information to estimate metabolite levels in 2D MRSI of heterogeneous brain lesions*, Tech. Report 09-182, ESAT-SISTA, K.U. Leuven, 2009.
2. I. Csiszár and G. Tusnády, *Information geometry and alternating minimization procedures*, Statistics & Decisions **Supplement Issue** (1984), no. 1, 205237.
3. R. de Graaf, *In Vivo NMR Spectroscopy. Principles and Techniques*, 2nd ed., John Wiley & Sons, 2007.
4. G.H. Golub and V. Pereyra, *Separable nonlinear least squares: the variable projection method and its applications*, Inverse Problems **19** (2003), no. 2, 1–26.
5. B. M. Kelm, *Evaluation of vector-valued clinical image data using probabilistic graphical models: Quantification and pattern recognition*, Ph.D. thesis, Ruprecht-Karls-Universität, Heidelberg, 2007.
6. J. J. Moré, *The Levenberg-Marquardt algorithm: Implementation and theory*, Numerical Analysis: Proceedings of the Biennial Conference held at Dundee, June 28-July 1, 1977 (Lecture Notes in Mathematics #630) (G. A. Watson, ed.), Springer Verlag, 1978, pp. 104–116.
7. U. Niesen, D. Shah, and G.W. Wornell, *Adaptive alternating minimization algorithms*, IEEE Transactions on Information Theory **55** (2009), no. 3, 1423–1429.
8. J.B. Poullet, D.M. Sima, A. Simonetti, B. De Neuter, L. Vanhamme, P. Lemmerling, and S. Van Huffel, *An automated quantitation of short echo time MRS spectra in an open source software environment: AQSES*, NMR in Biomedicine **20** (2007), no. 5, 493–504.
9. M.J.D. Powell, *On Search Directions for Minimization Algorithms*, Mathematical Programming **4** (1973), 193–201.
10. A. Ruhe and P.A. Wedin, *Algorithms for separable nonlinear least squares problems*, SIAM Review **22** (1980), 318–337.
11. D.M. Sima and S. Van Huffel, *Separable nonlinear least squares fitting with linear bound constraints and its application in magnetic resonance spectroscopy data quantification*, Journal of Computational and Applied Mathematics **203** (2007), 264–278.
12. D.M. Sima, J. Poullet, and S. Van Huffel, *Exponential data fitting and its applications*, ch. Computational aspects of exponential data fitting in Magnetic Resonance Spectroscopy, Bentham eBooks, 2009.

Optimization of Partial Differential Equations for Minimizing the Roughness of Laser Cutting Surfaces

Georg Vossen, Jens Schüttler and Markus Nießen

Lehr- und Forschungsgebiet Nichtlineare Dynamik der Laserfertigungsverfahren,
RWTH Aachen, Steinbachstr. 15, 52074 Aachen
georg.vossen@nld.rwth-aachen.de, jens.schuettler@nld.rwth-aachen.de and
markus.niessen@nld.rwth-aachen.de

Summary. This work introduces a mathematical model for laser cutting which involves two coupled nonlinear partial differential equations. The model will be investigated by linear stability analysis to study the occurence of ripple formations at a cutting surface. We define a measurement for the roughness of the cutting surface and give a method for minimizing the roughness with respect to process parameters. A numerical solution of this nonlinear optimization problem will be presented and compared with the results of the linear stability analysis.

1 Introduction

Laser cutting is a thermal separation process widely used in shaping and contour cutting applications. There are, however, gaps in understanding the dynamics of the process, especially issues related to cut quality. One essential problem in laser cutting is the occurence of ripple structures at the cutting surface, cf. Figure 1. Such structures can be induced by fluctuations in the

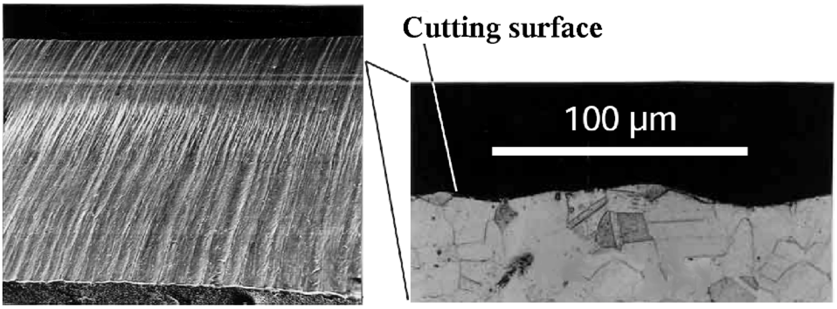


Fig. 1. Image of a cutting surface with ripple structures

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0_46, © Springer-Verlag Berlin Heidelberg 2010

melt flow during the process. Typical tasks in laser cutting applications involve finding process parameters like laser power or cutting speed such that ripple structures at the cutting front are minimal.

Research work has been done in the fields of modeling, model analysis and numerical simulation of laser cutting. One of the major challenges is the treatment of the arising melt and its free boundaries in the process. An overview on state-of-the-art and new developments in the field of modeling on the basis of asymptotic expansions, integral (or variational) methods and spectral methods is presented in [9]. Numerical simulation involving Level Set methods and adaptive sparse grids has been applied in [7]. Nonlinear stability analysis of melt flows has been carried out in [12]. The special problem of ripple formations has been investigated in [10, 3]. An optimization on the basis of the model in the latter reference has been applied in [11].

2 A model for the dynamical behavior of the melt surfaces

We introduce a model in scaled and dimensionless coordinates for the surfaces of the melt arising in a laser cutting process. Figure 2 shows the melt bounded by three free boundaries: the melt front and the absorption front (intersecting at $z = 0$) and the lower boundary along $z = 1$. The position of the laser beam axis (dashed) is $x = x_L := t$ and the laser with beam radius m_0 propagates vertically in z–direction. The melt front with position $x = x_M := x_L + M$ where $M = M(z, t)$ is the distance from the laser beam axis is given by the

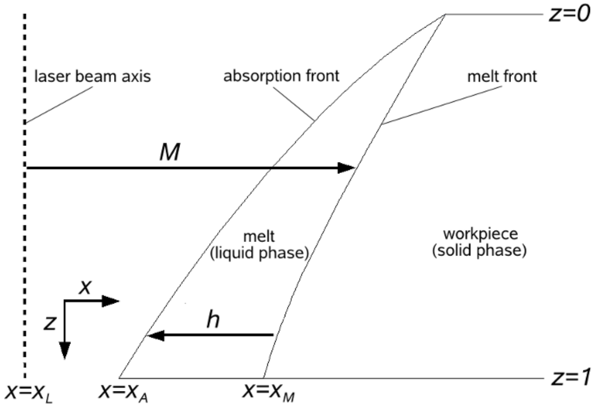


Fig. 2. Schematic 2D view of a laser cutting process

phase boundary where molten material from the solid phase enters the liquid phase. The cutting gas expelling the melt downwards and the laser beam hit the melt at the absorption front with position $x = x_A := x_M - h$ where $h = h(z,t)$ is the melt film thickness. It intersects the melt front at $(x,z) = (m_0, 0)$. The lower boundary of the melt is given as the connecting arc between the lower end points of the melt and the absorption front along $z = 1$.

A model for the dynamical behavior of the melt and the absorption front and their interactions is given by the initial/boundary value problem

$$\frac{\partial h}{\partial t} + 2h\frac{\partial h}{\partial z} = v_p, \quad \frac{\partial M}{\partial t} = v_p - 1, \quad v_p = Q_A - Q_s, \quad Q_s = \frac{c_p \mathit{\Delta} T}{H_m} \tag{1}$$

$$Q_A = \nu\mu\mathcal{A}(\mu), \quad \nu = \frac{P_L}{\pi w_0^2 \rho H_m v_0}, \quad \mathcal{A}(\mu) = \frac{4\mu\iota}{2\mu^2 + 2\mu\iota + \iota^2} \tag{2}$$

$$\mu = \alpha\left(\frac{\partial h}{\partial z} - \frac{\partial M}{\partial z}\right), \quad \alpha = \frac{d_m}{d}, \quad d_m = \sqrt{\frac{2\eta_l v_0 d}{\tau_g}} \tag{3}$$

$$h(0,t) = 0, \;\; M(0,t) = m_0 = w_0/d_m, \;\; h(z,0) = h_i(z), \;\; M(z,0) = M_i(z) \tag{4}$$

for z, $t \geq 0$. Here, $v_p = v_p(z,t)$ denotes the dimensionless in-flow velocity of the melt in normal direction at the melt front. Furthermore, $Q_A = Q_A(z,t)$ and the constant Q_s are dimensionless heat flow densities at the absorption front and in the solid phase at the melt front, respectively. Here, Q_A is a function of the Fresnel absorption $\mathcal{A} = \mathcal{A}(\mu)$ and the cosine $\mu = \mu(z,t)$ of the angle of incidence of the laser beam onto the absorption front which involves spatial derivatives of h and M. Thus, (1)–(3) yield a nonlinear coupled system of partial differential equations with initial and boundary conditions (4) where m_0 is the distance of the melt front at $z = 0$, and h_i, M_i are initial distributions for h and M at $t = 0$.

Constant positive parameters in this model are the specific heat capacity c_p, the difference $\mathit{\Delta} T$ between melting and ambient temperature, the enthalpy of fusion H_m, the beam radius w_0, the mass density ρ, the material absorption parameter ι, the thickness d of the workpiece, the dynamical viscosity η_l and the shear stress τ_g of the cutting gas along the absorption front. Parameters which can be used as optimization variables are the laser power P_L and the cutting velocity v_0. All parameters are given in corresponding physical units.

We use scaled and dimensionless coordinates $x = \tilde{x}/d_m$, $z = \tilde{z}/d$, $t = v_0\tilde{t}/d_m$ and obtain the scalings $h = \tilde{h}/d_m$, $M = \tilde{M}/d_m$ where the $\sim$ superscript indicates that the quantity is given in its corresponding physical unit. The quantity d_m is a typical length for the melt film thickness. The in-flow velocity v_p is scaled by $v_p = \tilde{v}_p/v_0$ whereas the heat flow densities are scaled by $Q = \tilde{Q}/(v_0\rho H_m)$ where, again, $\tilde{v}_p$ and $\tilde{Q}$ are given in corresponding physical units.

To deduce the model, we consider the implicit description

$$\mathit{\Phi}_M(x,z,t) := x_L(t) + M(z,t) - x = 0 \tag{5}$$

for the melt front at $x = x_M$ and the dynamical behavior of a particle (x, z) along this surface which yields

$$\frac{d}{dt}\left(\Phi_M(x(t), z(t), t)\right) = 1 + v_{z,M}\frac{\partial M}{\partial z} + \frac{\partial M}{\partial t} - v_{x,M} = 0 \tag{6}$$

Here, $v_{x,M} = v_{x,M}(z, t)$ and $v_{z,M} = v_{z,M}(z, t)$ denote the dimensionless velocities of the melt front in x– and z–direction, respectively, which can be substituted by means of the dimensionless in-flow velocity v_p of the melt. Denoting $n_M = n_M(z, t)$ as the scaled unit length outer normal vector of the melt along the melt front, we obtain

$$v_p = \langle (v_{x,M}, v_{z,M}), n_M \rangle = \frac{1}{\sqrt{1 + \left(\alpha \frac{\partial M}{\partial z}\right)^2}} \left(v_{x,M} - v_{z,M}\frac{\partial M}{\partial z} \right) \tag{7}$$

at the melt front. Combining (6) with (7) yields

$$\frac{\partial M}{\partial t} = v_p \sqrt{1 + \left(\alpha \frac{\partial M}{\partial z}\right)^2} - 1 = v_p - 1 + O(\alpha^2) \tag{8}$$

For the absorption front $x = x_A$, we consider the transformation $\bar{x} = x_M - x$ and obtain the implicit form

$$\Phi_A(\bar{x}, z, t) := h(z, t) - \bar{x} = 0 \tag{9}$$

which can be used to deduce a kinematic boundary condition from

$$\frac{d}{dt}\left(\Phi_A(\bar{x}(t), z(t), t)\right) = v_{z,A}\frac{\partial h}{\partial z} + \frac{\partial h}{\partial t} - v_{\bar{x},A} = 0 \tag{10}$$

In [7] we obtain that the relative velocities $v_{\bar{x},A}$ and $v_{z,A}$ of the absorption front in $\bar{x}$– and z–direction are given by

$$v_{\bar{x},A} = v_p + O(\alpha), \quad v_{z,A} = 2h + O(\alpha) \tag{11}$$

The expression for d_m can be deduced from [8] which implies $\alpha \ll 1$ in a realistic cutting process. Neglecting therefore terms of order $O(\alpha)$ in (8) and (11) yields the two first-order partial differential equations in (1). The coupling can be deduced by means of the so-called Stefan condition [4] where the inflow velocity v_p is given by the jump of the heat flow density at the melt front. Due to the thinness of the melt, we assume that, for fixed z, the heat flow density in the liquid phase is constant which leads to the third equation in (1). The expressions for Q_A and Q_s as well as the other formulae in (2), (3) can be found in [8] where, again, higher order terms in α are neglected. We note that the approximation for μ is only good for values around zero, i.e. for nearly vertical absorption fronts, which is the case in a typical cutting process. The x–position m_0 of the two fronts at $z = 0$ is given by the scaled value w_0/d_m of the beam radius. For this model we assume no interaction of the lower boundary at $z = 1$ with the melt. Hence, we consider $z \in [0, \infty)$ in the following theoretical discussions.

3 Linear stability analysis

In this section, we perform a linear stability analysis of the system (1)–(4). We introduce a perturbation parameter $\epsilon > 0$ and investigate (1)–(4) by using the initial condition

$$h_i = h_0 + \epsilon g_h, \quad M_i = M_0 + \epsilon g_M \tag{12}$$

where $h_0 = h_0(z)$ and $M_0 = M_0(z)$ are stationary solutions of (1)–(4) and $g_h = g(z)$, $g_M = g_M(z)$ are initial perturbations in the system, e.g. given by a sinusoidal wave with fixed frequency. We partition the solution by

$$h = h_0 + \epsilon h_1, \quad M = M_0 + \epsilon M_1 \tag{13}$$

where $h_1 = h_1(z,t)$ and $M_1 = M_1(z,t)$ describe the dynamical behavior of the initial perturbations g_h, g_M. We consider the Taylor expansion of the absorption $\mathcal{A}$ around the stationary value μ_0 of μ given by

$$\mathcal{A}(\mu) = \mathcal{A}(\mu_0) + \epsilon\mu_1\mathcal{A}'(\mu_0) + O(\epsilon^2), \quad \mu = \mu_0 + \epsilon\mu_1. \tag{14}$$

where the partition of μ is a direct consequence of (13) and (3).

In the following, we suppose $\mu_0 \geq 0$ since $\mu_0 < 0$ implies that the laser beam hits the absorption front from inside the melt which is, from the physical point of view, not reasonable.

Lemma 1. *A stationary solution of (1)–(4) with $\mu_0 \geq 0$ exists if and only if*

$$0 < r < 2\iota, \quad r = \frac{1+Q_s}{\nu} \tag{15}$$

holds. In this case, the solution is unique and given by

$$h_0(z) = \sqrt{z}, \quad M_0(z) = \sqrt{z} - \frac{1}{\alpha}\mu_0 z + m_0, \quad \mu_0 = \frac{\iota r}{\sqrt{(4\iota - r)r} - r} \tag{16}$$

Proof. We substitute (13) and (14) into (1)–(4) and consider terms of order $O(1)$ in ϵ to obtain two solutions $\mu_0^{(1)}$ and $\mu_0^{(2)}$ given by

$$\mu_0^{(1)} = \frac{\iota r}{\sqrt{(4\iota - r)r} - r}, \quad \mu_0^{(2)} = \frac{\iota r}{-\sqrt{(4\iota - r)r} - r}, \quad r = \frac{1+Q_s}{\nu} \tag{17}$$

for the stationary value μ_0 of μ. Hence, for $r > 4\iota$, i.e. large values of v_0 or small values of P_L, there exists no stationary solution. Furthermore, for all $r \leq 4\iota$, we obtain $\mu_0^{(2)} < 0$ which contradicts the assumption $\mu_0 \geq 0$. Since $r > 2\iota$ implies $\mu_0^{(1)} < 0$ and $r = 2\iota$ provides no solution for $\mu_0^{(1)}$, the only possible setting is $r < 2\iota$ (i.e. small v_0 or large P_L) which leads to the stationary solutions given in (16).

Remark 1. We note that in this model μ_0 and hence the angle of incidence of the laser beam onto the absorption front is constant for all z.

Considering terms of order $O(\epsilon)$ yields the linear perturbation system

$$\frac{\partial y}{\partial t} + F\frac{\partial y}{\partial z} = Ny, \quad y(0,t) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad y(z,0) = \begin{pmatrix} g_h(z) \\ g_M(z) \end{pmatrix} \tag{18}$$

$$F = F(z) = \begin{pmatrix} 2h_0 - c_0 & c_0 \\ -c_0 & c_0 \end{pmatrix}, \quad N = N(z) = \begin{pmatrix} -1/h_0 & 0 \\ 0 & 0 \end{pmatrix} \tag{19}$$

for the vector perturbation $y = (h_1, A_1)^T$ with, using (16),

$$c_0 = \alpha\nu \left(\mathcal{A}(\mu_0) + \mu_0\mathcal{A}'(\mu_0)\right) = \alpha(1+Q_s)\frac{2\iota(\iota+\mu_0)}{\mu_0(\iota^2+2\iota\mu_0+2\mu_0^2)} > 0 \tag{20}$$

Lemma 2. *For $z > 0$, the system (18), (19) is hyperbolic, elliptic or parabolic if and only if the term $h_0(z) - 2c_0$ is positive, negative or zero, respectively.*

Proof. The eigenvalues of F are given by $h_0 \pm \sqrt{h_0^2 - 2c_0h_0}$. For $z > 0$, i.e. $h_0 > 0$, we obtain two real, two complex or one multiple eigenvalue if the radicant is positive, negative or zero, respectively, which proves Lemma 2.

Remark 2. The system (18), (19) yields an interesting example of a system whose property changes from elliptic via parabolic to hyperbolic while z decreases.

In general, a solution of (18), (19) cannot be given since F and N depend on z. To investigate further properties, we consider for a fixed position $z_0 > 0$, the solution in a small neighborhood $|z-z_0| < \delta$, $z_0-\delta > 0$, where the variation of $h_0(z)$ is small. We denote $c_1 := h_0(z_0)$ and obtain

$$\frac{\partial y}{\partial t} + \bar{F}\frac{\partial y}{\partial z} = \bar{N}y, \quad \bar{F} = \begin{pmatrix} 2c_1 - c_0 & c_0 \\ -c_0 & c_0 \end{pmatrix}, \quad \bar{N} = \begin{pmatrix} -1/c_1 & 0 \\ 0 & 0 \end{pmatrix} \tag{21}$$

with initial condition $y(z,0) = (g_h(z), g_M(z))^T$. Note that we are interested in stability of the system (18), (19), i.e. in particular large variations in h_1, M_1. Thus, it is reasonable to investigate the solutions of (21) which will give, for small times $t \geq 0$, local approximations of the solutions of (18), (19). For a rigorous investigation of error bounds between the solutions of both systems, we refer to [1].

Proposition 1. *The system (21) is linearly unstable.*

Proof. Using Fourier transform with respect to z yields the ordinary differential equation

$$\frac{\partial Y}{\partial t} = \bar{R}Y, \; \bar{R} = \bar{N} - ik\bar{F}, \; Y = Y(k,t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} y(w,t)\exp(-ikw)\,dw \tag{22}$$

Stability of (22) and hence, cf. [6], of (21), can be analyzed by means of the real parts of the two complex eigenvalues $\sigma_{1/2} = \sigma^r_{1/2} + i\sigma^i_{1/2}$ of R partitioned in real and imaginary parts given by

$$\sigma^r_{1/2} = -\frac{1}{2c_1} \pm \sqrt{\frac{\xi + \sqrt{\xi^2 + \zeta^2}}{2}}, \quad \sigma^i_{1/2} = -kc_1 \pm \sqrt{\frac{-\xi + \sqrt{\xi^2 + \zeta^2}}{2}} \tag{23}$$

$$\xi = k^2 c_1(2c_0 - c_1) + \frac{1}{4c_1^2}, \quad \zeta = k\left(1 - \frac{c_0}{c_1}\right) \tag{24}$$

Since $\zeta = 0$ implies $c_0 = c_1$ and hence $\xi > 0$, we have $\sigma^r_1 > \sigma^r_2$. Therefore, (22) is stable if and only if $\sigma^r_1 \leq 0$ holds. Basic calculations yield that this is equivalent to the three conditions

$$(\mathrm{I}): \ \frac{1}{2c_1} \geq 0, \quad (\mathrm{II}): \ \frac{1}{2c_1^2} - \xi \geq 0, \quad (\mathrm{III}): \ -c_0^2 \geq 0 \tag{25}$$

Condition (I) is fulfilled for all $z > 0$. In view of Lemma 2, Condition (II) is satisfied if and only if the system is not elliptic. However, condition (III) is not fulfilled since from (20) we obtain $c_0 \neq 0$. This implies $\sigma^r_1 > 0$ and hence instability of the system.

As shown in Proposition 1, system (21) is unstable since $c_0 \neq 0$. However, the proof illustrates that $c_0 = 0$ implies $\sigma^r_1 = 0$ and hence marginal stability. Therefore, the value of c_0 can be interpreted as a measurement for instability and for decreasing values $c_0 \to 0$ the process becomes more stable. From (20) we conclude that $c_0 \to 0$ holds for $\mu_0 \to \infty$ which, due to (15), is obtained for $r \to 2\iota$. In the limit case $r = 2\iota$, we obtain a marginal stability curve

$$\mathcal{N} = \{(v_0, P_L) \in \mathbb{R}^2 : P_L = C\, v_0\}, \quad C = \frac{(1 + Q_s)\pi w_0^2 \rho H_m}{2\iota} \tag{26}$$

which, as mentioned above, cannot be achieved in practice since μ_0 is not defined in this case. Figure 3 shows the nonlinear dependency of c_0 on p

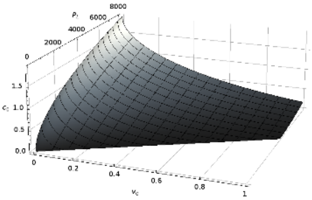


Fig. 3. c_0 as a function of the process parameters v_0 and P_L

where the plot is cut at the curve $\mathcal{N}$ for realistic parameters for stainless steel

$$c_p = 550, \quad \Delta T = 1500, \quad H_m = 277 \cdot 10^3, \quad w_0 = 300 \cdot 10^{-6} \tag{27}$$

$$\rho = 7000, \quad \iota = 0.25, \quad d = 4 \cdot 10^{-3}, \quad \eta_l = 2 \cdot 10^{-3}, \quad \tau_g = 500 \tag{28}$$

It can be deduced that c_0 is strictly monotonuously increasing with P_L and decreasing with v_0 which implies that large values of v_0 and small values of P_L may provide a stationary solution with only small instabilities.

4 Minimizing the roughness of the surfaces

To extend our results about the connection between the process parameters v_0, P_L and stability of the system, we will investigate a nonlinear optimization problem. The goal is to find a process parameter vector

$$p = (v_0, P_L)^T \in P, \quad P \subset P_{\mathrm{ad}} := \{p \in \mathbb{R}^2 : P_L > C\, v_0 > 0\} \tag{29}$$

with C from (26) such that the melt surfaces stay close to the stationary solution. Here, P is an arbitrary non-empty compact and convex subset of P_{ad} where the condition $P_L > C\, v_0$ (which is equivalent to $r < 2\iota$, cf. (15)) in the definition of P_{ad} ensures the existence of a stationary solution due to Lemma 1 and $v_0 > 0$ (note that $C > 0$) is a physically reasonable bound. The problem is to find $p \in P$ which minimizes the roughness

$$\mathcal{R}(p) := \frac{1}{2} \int_0^1 \int_0^{t_f} \Big[(h(z,t;p) - h_0(z;p))^2 + \lambda (M(z,t;p) - M_0(z;p))^2 \Big]\, dt\, dz \tag{30}$$

where $h(z,t;p)$, $M(z,t;p)$ are solutions of (1)–(4) with inital condition (12) using $h_0(z;p)$, $M_0(z;p)$ from (16) as stationary solutions, λ is a weighting parameter and t_f is a suitable chosen final time. We will assume that h_0, h, M_0, M are unique solutions sufficiently smooth with respect to p. Note that for all p the system is not stable. Hence, a solution of the optimization problem will yield parameters where the surface roughness is as small as possible.

We present a numerical solution of problem (30). The spatial and time domain is partitioned into $N_z = 80$ and $N_t = 1600$ intervals of length h_z and h_t, respectively. We use the Lax-Wendroff [5] and an Euler-forward scheme for the equation for h and M, respectively. The derivatives in μ are treated by an upwind method. The cost functional (30) is approximated by the composite trapezoidal rule. Using data (27), (28) yields $C = 4362.164$. We choose

$$\epsilon = 0.025, \quad g_h(z) = \sin(5 \cdot 2\pi z) = 10 g_M(z), \quad t_f = 0.8, \quad \lambda = 10 \tag{31}$$

The domain $P \in P_{\mathrm{ad}}$ is taken as

$$0.01 \le v_0 \le 0.2, \quad 100 \le P_L \le 6000, \quad P_L \ge 1.5\, C\, v_0 \tag{32}$$

Using the code IPOPT [13] together with the modeling language AMPL [2], we obtain two local minima p_1 and p_2 of $R(p)$ in P given by

$$p_1 = (0.019995, 180.80)^T, \quad p_2 = (0.2, 6000)^T \tag{33}$$

with $R(p_1) = 0.2500914087$ and $R(p_2) = 0.4751724227$ where p_1 is close to $\mathcal{N}$ and p_2 is at the boundary of P far away from $\mathcal{N}$. Hence, there exists a domain where the roughness decreases for p approaching $\mathcal{N}$ which is equal to the results in the previous section. However, p_1 is strictly inside P close to the boundary and there exists a second minimum. A possible interpretation for these discrepancies with the linear stability analysis is that there are nonlinear effects in the system which can lead to a surface with small roughness although the process is strongly linear instable. Figure 4 shows the solution for h and M for the parameter p_1.

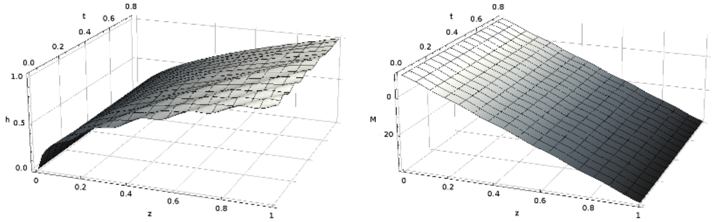


Fig. 4. Melt thickness h (left) and position M of the melt front (right)

We emphasize that p_1 is no realistic parameter vector since P_L is not large enough to melt the workpiece. This can also be seen from the mathematical point of view since μ_0 in this case is so large that at $z = 1$ the absorption front has left the area $[-m_0, m_0]$ of the laser beam, i.e. $M_0(1) < -m_0$. Adjusting P by adding this constraint $\mu_0 \leq \alpha(1 + 2m_0)$, we obtain the only minimum p_2.

5 Conclusions and Outlook

We presented a model for the dynamical behavior of the free melt surfaces in a laser cutting process which involves two nonlinear coupled partial differential equations. We identified parameter domains for the existence of a stationary solution and showed uniqueness in this case. We applied a linear stability analysis to an approximate model and obtained that the system is linearly unstable. This investigation implied that the distance of the parameter vector to a practically not achievable neutral stability curve is a measurement for instability of the system providing rough cutting surfaces. As a second approach, we formulated a nonlinear optimization problem. The goal was to find parameters which minimize the roughness of the cutting surface defined by a tracking cost functional measuring the L_2 distance to the stationary solution. A numerical solution was presented which showed that in a certain domain the results correspond to the linear stability analysis. However, presumedly due to nonlinear effects, we obtained a second local minimum far away from the neutral stability curve. We finally identified a further condition for the technically relevant parameter domain leading to only this second minimum.

Future works comprise extension of the model by non-vertical beam incidence, nonlinear stability analyis (which may lead to explanations for the second minimum), study of necessary and sufficient optimality conditions and the consideration of further, also spatial and time dependent optimization variables which leads to optimal control problems.

6 Acknowledgement

The research related to this paper is supported by the German Research Foundation DFG as part of the Cluster of Excellence "Integrative Production Technology for High-Wage Countries" at RWTH Aachen University.

References

1. Colombo RM, Mercier M, Rosini MD (2009) Stability estimates on general scalar balance laws. C R Acad Sci Paris, Ser I 347
2. Fourer R, Gay DM, Kernighan (1990) A Modeling Language for Mathematical Programming. Management Science 36:519554
3. Friedrich R, Radons G, Ditzinger T, Henning A (2000) Ripple Formation through an Interface Instability from Moving Growth and Erosion Sources. Phys Rev Lett 85:4884-4887
4. Lamé G, Clapeyron BD (1831) Mémoire sur la solidification par refroidissement d'un globe liquide. Ann Chimie Physique, 47:250-256
5. Lax PD, Wendroff B (1960) Systems of conservation laws. Commun Pure Appl Math 13:217-237
6. Kevorkian J (2000) Partial Differential Equations: Analytical Solution Techniques. 2nd Edition. Springer, New York
7. Nießen M (2005) Numerische Modellierung freier Randwertaufgaben und Anwendung auf das Laserschneiden. PhD thesis, RWTH Aachen University
8. Schulz W (2003) Die Dynamik des thermischen Abtrags mit Grenzschichtcharakter. Aachen, Shaker-Verlag, Habilitation thesis, RWTH Aachen University
9. Schulz W, Nießen M, Eppelt U, Kowalick K (2009) Simulation of Laser Cutting. In: Dowden JM (ed) The Theory of Laser Materials Processing: Heat and Mass Transfer in Modern Technology. Springer Series in Materials Science 119
10. Schulz W, Kostrykin V, Nießen M, Michel J, Petring D, Kreutz EW, Poprawe R (1999) Dynamics of Ripple Formation and Melt Flow in Laser Beam Cutting. J Phys D: Appl. Phys. 32:1219-1228
11. Theißen K (2006) Optimale Steuerprozesse unter partiellen Differentialgleichungs-Restriktionen mit linear eingehender Steuerfunktion. PhD thesis, University of Münster
12. Vossen G, Schulz W (2009) Multiple Scale Methods for Nonlinear Stability Analysis in Laser Cutting Processes. Technical Report, RWTH Aachen. Online: http://www.nld.rwth-aachen.de/
13. Wächter A, Biegler LT (2006) On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming. Mathematical Programming 106(1):25-57

Author Index

M. Diehl et al. (eds.), *Recent Advances in Optimization and its Applications in Engineering*,
DOI 10.1007/978-3-642-12598-0, © Springer-Verlag Berlin Heidelberg 2010

Subject Index