



**Calhoun: The NPS Institutional Archive**  
**DSpace Repository**

---

Faculty and Researchers

Faculty and Researchers' Publications

---

2018-10

A recursively feasible and convergent  
Sequential Convex Programming procedure to  
solve non-convex problems with linear  
equality constraints

Virgili-Llop, Josep; Romano, Marcello

ArXiv

---

Virgili-Llop, Josep, and Marcello Romano. "A recursively feasible and convergent Sequential Convex Programming procedure to solve non-convex problems with linear equality constraints." arXiv preprint arXiv:1810.10439 (2018).

<https://hdl.handle.net/10945/65306>

---

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the

Downloaded from NPS Archive: Calhoun



Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

**Dudley Knox Library / Naval Postgraduate School**  
**411 Dyer Road / 1 University Circle**  
**Monterey, California USA 93943**

<http://www.nps.edu/library>

# A recursively feasible and convergent Sequential Convex Programming procedure to solve non-convex problems with linear equality constraints

Josep Virgili-Llop and Marcello Romano

*Mechanical and Aerospace Engineering Department, Naval Postgraduate School, 1 University Circle, Monterey, CA, 93940*

---

## Abstract

A computationally efficient method to solve non-convex programming problems with linear equality constraints is presented. The proposed method is based on a recursively feasible and descending sequential convex programming procedure proven to converge to a locally optimal solution. Assuming that the first convex problem in the sequence is feasible, these properties are obtained by convexifying the non-convex cost and inequality constraints with inner-convex approximations. Additionally, a computationally efficient method is introduced to obtain inner-convex approximations based on Taylor series expansions. These Taylor-based inner-convex approximations provide the overall algorithm with a quadratic rate of convergence. The proposed method is capable of solving problems of practical interest in real-time. This is illustrated with a numerical simulation of an aerial vehicle trajectory optimization problem on commercial-of-the-shelf embedded computers.

*Keywords:* convex programming, trajectory optimization

---

## 1. Introduction

The ability to solve trajectory optimization problems in real-time on embedded platforms is of paramount importance to achieve a high level of autonomy in aerospace applications. To accomplish this goal, obtaining assurances about an algorithm's computational feasibility, tractability, and strict convergence is a necessity [1]. The growing interest in the computational aspects of trajectory optimization has recently crystallized with the coining of the term Computational Guidance and Control [2].

Convex optimization paired with solvers based on interior-point methods enable to solve convex programming problems to global optimality in polynomial-time [3, 4]. These properties make convex programming suitable for implementation in real-time autonomous applications [5, 6, 7].

Many optimization problems can be cast as convex programming problems [4]. For example, fixed-final-time trajectory optimization problems of linear systems with convex cost and inequality constraints form convex programming problems when discretized [8]. Some problems with apparently non-convex control constraints can also be cast as convex programming problems thanks to lossless convexification or exact relaxation techniques [9, 10, 11, 12, 13, 14]. In trajectory optimization, the use of convex programming techniques has been remarkably fruitful in the past decade [15]. Some of the areas where it has been applied include: powered soft-landing [16, 17, 11, 18, 19], time-optimal path following [14, 20, 21], spacecraft proximity maneuvering [22, 23, 24, 25], re-entry guidance [26, 27, 28], swarm control [29, 30], quadrotor guidance [31], missile guidance [13], and motion planning [32, 33, 7].

Despite these significant advances, there is a large num-

ber of optimization problems of important practical interest that cannot be cast as convex programming problems. For example, problems involving nonlinear systems or problems with non-convex constraints generally produce non-convex programming problems. Although a wide range of approaches are available to solve these non-convex programming problems [34], none of these provide the same computationally attractive properties that interior-point methods offer for convex problems. A notable exception may be the Lasserre's relaxations [35] and their variations [36]. Lasserre-type relaxations are able to relax non-convex programming problems, written in terms of polynomials, to semidefinite programming problems. Although promising, these relaxations dramatically increase the dimensionality of the problem.

In an attempt to bridge some of the attractive properties of convex programming to non-convex problems, the sequential convex programming (SCP) procedure has been proposed [15]. First developed to solve structural problems [37, 38], this method has been adopted by the trajectory optimization community (in addition to previous references see [39, 40, 41, 42, 43, 44]). In an SCP procedure, a convex approximation of the original non-convex problem is repeatedly solved until convergence. Finding the solution of a non-convex programming problem is thus reduced to solving a collection of convex—hence tractable—programming problems.

To obtain the sequence of convex programming problems that constitute an SCP, the non-convex functions are usually linearized around the previous iteration's solution, in what is known as successive linearization [15, 44]. Linearization-based SCP procedures can suffer from inadmissible iterates, infeasible problems, or cost increases between iterations. Trust regions and penalties can be used to alleviate these problems, and when

properly applied can result in a convergent SCP procedure with a superlinear rate of convergence [41, 44].

### 1.1. Contributions

Here we advance the SCP's state-of-the-art by making two critical contributions. First, we present an SCP procedure that offers guaranteed convergence to a locally optimal solution for non-convex problems with linear equality constraints. This procedure is recursively feasible as long as the first convex problem in the sequence is feasible. Trust regions are not needed as the solution to the convexified problem is always admissible. These advantageous properties are obtained when inner-convex approximations are used [45]. These properties were first demonstrated by Marks and Wright [45] but they have remained largely unnoticed, and in some cases, independently rediscovered by the aerospace trajectory optimization community [25, 30]. Here, we generalize and consolidate previously available results and highlight how the use of inner-convex approximations prevents inadmissible solutions, infeasible problems, and cost increases. Additionally, we briefly review the different available methods to obtain inner-convex approximations (namely, sum-of-squares decompositions and regularizations).

Secondly, we introduce a computationally efficient approach to obtain inner-convex approximations. Our approach is based on the convexification of the Taylor series expansion of a non-convex analytic function. These inner-convex approximations allow the SCP to achieve a locally quadratic rate of convergence under mild conditions. As Taylor series may be infinite and therefore truncated, we introduce a high-order regularization approach that allows to truncate the Taylor series and retain the inner-convex properties.

The combination of a recursively feasible and convergent SCP with an efficient method to obtain inner-convex approximations results in an algorithm that is suitable for onboard implementation and real-time use. As each iteration involves solving a convex programming problem, the time per iteration is bounded. Then, the recursively feasible aspect allows the SCP to stop at any iteration and still yield an admissible solution. Finally, the convergent property ensures that each solution is better than the last. These properties make the proposed method particularly attractive for online trajectory optimization on autonomous systems.

### 1.2. Organization of the paper

The rest of this paper is organized as follows. First, the class of non-convex problems with linear equality constraints is introduced in Section 2. Section 3 outlines the generic SCP procedure. A brief discussion on the successive linearization approach is included in Section 4. In Section 5 we introduce the concept of inner-convex approximations and present the recursively feasible and convergent SCP procedure that is obtained when these approximations are used. In the following Section 6 a brief overview of the existing methods that can be used to find inner-convex approximations is provided. Then, in Section 7 we introduce our method to obtain inner-convex approximations based on the convexification of a Taylor series expansion. Finally, and to illustrate the use of the proposed method,

a numerical simulation example, based on the path-planning of an aerial vehicle, is provided in Section 8. The results of an implementation on two different embedded computers are also provided, substantiating the claim that the proposed method is suitable for onboard implementation and real-time use.

## 2. Non-convex programming problems with linear equality constraints

The focus of this work is directed towards non-convex programming problems with linear equality constraints. In canonical form, this class of problems can be written as:

**Problem P. Non-convex programming problem with linear equality constraints**

$$\begin{aligned} &\text{minimize: } f_0(x) \\ &\text{subject to: } f_i(x) \leq 0, & \text{for } i = 1 \dots p & \quad (P) \\ & & h_i(x) = 0, & \text{for } i = 1 \dots q \end{aligned}$$

In this problem,  $x \in \mathbb{R}^n$  is the optimization variable, the equality constraint functions  $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$  are affine, and the cost or objective function  $f_0 : \mathbb{R}^n \rightarrow \mathbb{R}$  as well as the inequality constraint functions  $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$  are non-convex. To simplify notation, generic non-convex functions are simply denoted by  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  in the following.

In general, we call an admissible point of the programming problem  $P$  any point that satisfies the constraints,  $x \in \mathcal{D}_P$ , with  $\mathcal{D}_P$  denoting the problem's admissible set. The programming problem is said to be feasible if  $\mathcal{D}_P$  has a non-empty interior and infeasible if  $\mathcal{D}_P = \emptyset$ .

A locally optimal solution of problem  $P$ , denoted by  $x_P^*$ , is understood as a point where the Karush-Kuhn-Tucker (KKT) conditions are satisfied [4]:

$$f_i(x_P^*) \leq 0, \quad \text{for } i = 1 \dots p \quad (1a)$$

$$h_i(x_P^*) = 0, \quad \text{for } i = 1 \dots q \quad (1b)$$

$$v_{P,i}^* \geq 0, \quad \text{for } i = 1 \dots p \quad (1c)$$

$$v_{P,i}^* f_i(x_P^*) = 0, \quad \text{for } i = 1 \dots p \quad (1d)$$

$$\nabla f_0(x_P^*) + \sum_{i=1}^p v_{P,i}^* \nabla f_i(x_P^*) + \sum_{i=1}^q \mu_{P,i}^* \nabla h_i(x_P^*) = 0 \quad (1e)$$

with  $v_P^* \in \mathbb{R}^p$  and  $\mu_P^* \in \mathbb{R}^q$  denoting the optimal dual variables associated with  $x_P^*$ . Finally,  $\nabla f$  is used to denote the gradient of a function  $f$ .

## 3. Sequential Convex Programming (SCP) procedure

In a sequential convex programming procedure a locally optimal solution to the original non-convex problem  $P$  is found by iteratively solving a convex approximation of the original non-convex problem. The iterations within the SCP's sequence are represented by  $k \in \mathbb{N}$  and the solutions to each of the convex programs in the sequence are denoted by  $x_{(k)}$ . The  $k$ -th convex programming problem  $P_{\text{cvx}}^{(k)}$  is obtained by convexifying the

non-convex functions,  $f$ , around the previous iteration's solution  $x_{(k-1)}$ .

The notation  $f_{\text{cvx}}^{(x_e)}$  is used here to denote a convex approximation of  $f$  around the expansion point  $x_e$ . For simplicity, the approximation around the previous iteration's solution  $f_{\text{cvx}}^{(x_{(k-1)})}$ , is simply denoted as  $f_{\text{cvx}}^{(k-1)}$ . With this notation, the convex problem to be repeatedly solved in an SCP procedure is:

**Problem  $P_{\text{cvx}}^{(k)}$ . Convex approximation of problem  $P$  around the previous iteration's solution  $x_{(k-1)}$**

$$\begin{aligned} & \text{minimize: } f_{\text{cvx},0}^{(k-1)}(x) \\ & \text{subject to: } f_{\text{cvx},i}^{(k-1)}(x) \leq 0, \quad \text{for } i = 1 \dots p \quad (P_{\text{cvx}}^{(k)}) \\ & \quad \quad \quad h_i(x) = 0, \quad \text{for } i = 1 \dots q \end{aligned}$$

The first time problem  $P_{\text{cvx}}^{(k)}$  is solved,  $k = 1$ , the problem is convexified around an initial guess,  $x_{(0)}$ . The SCP procedure iteratively solves problem  $P_{\text{cvx}}^{(k)}$  until a stopping criterion is satisfied. For example, when the cost difference between iterates falls below a certain threshold  $\epsilon \geq 0$ , *i.e.*,

$$|f_0(x_{(k-1)}) - f_0(x_{(k)})| \leq \epsilon. \quad (2)$$

#### 4. SCP with successive linearization

A common approach to obtain the convex approximations  $f_{\text{cvx}}^{(x_e)}$  is to linearize  $f$  [15],

$$f_{\text{cvx}}^{(x_e)}(x) = f(x_e) + \delta x^T \nabla f(x_e), \quad (3)$$

$$\delta x = x - x_e. \quad (4)$$

It has been shown that a linearization-based SCP is locally convergent [39]. However, the convergence is not guaranteed outside a small neighborhood around the locally optimal solution. In general, a linearization-based SCP procedure may yield inadmissible solutions. *I.e.*, the solutions,  $x_{(k)}$ , when re-evaluated with the original non-convex functions,  $f$ , may violate the inequality constraints. Furthermore, the cost may be higher than the cost from previous iterations, *i.e.*,  $f_0(x_{(k)}) > f_0(x_{(k-1)})$  [46]. This means that it is not known *a priori* if solving the next convex problem in the sequence will yield a better or even an admissible solution. Additionally, linearizing the cost can lead to unbounded problems, completely preventing the use of a linearization-based SCP.

To mitigate these negative effects, trust regions are commonly used [47]. Trust regions restrict the solution,  $x_{(k)}$ , to a closed bounded convex set,  $x \in \mathcal{T} \subset \mathbb{R}^n$ , where the linearization is expected to be accurate. The heuristic use of trust regions does not eliminate the possibility of an inadmissible solution or cost increase. If that solution can be discarded and the same convex problem re-solved with tighter trust regions, the SCP is shown to be convergent [41].

##### 4.1. Penalty method to resolve infeasibility

When a convex problem in the sequence is infeasible a penalty method can be applied in an attempt to gain feasibility [32, 46].

The penalty method consists on turning the infeasible inequality constraints into parts of the cost function. First, the inequality  $f_i(x) \leq 0$  is relaxed with a slack variable  $s_i$ ,

$$f_i(x) \leq s_i, \quad \text{for } i = 1 \dots l \leq p \quad (5a)$$

$$s_i \geq 0, \quad \text{for } i = 1 \dots l \leq p \quad (5b)$$

and then, the cost function is augmented as

$$f_0(x) + \kappa_{(k)} \sum_{i=1}^l s_i, \quad (6)$$

with  $\kappa_{(k)}$  denoting the penalty's weight or gain.

The theory of exact penalties [48] states that the constrained and the penalty-relaxed problems are equivalent as long as  $\kappa_{(k)}$  is greater than the largest optimal dual variable,  $v_{P,i}^*$ , associated with the relaxed inequalities. The values of  $v_{P,i}^*$  are unlikely to be known *a priori*, which prompts the use of large gains.

The use of penalties is a useful strategy to resolve infeasibilities, and a critical tool to find an admissible solution when an admissible initial guess is not available. The use of penalties also allows the SCP to proceed. However, the SCP may converge to an inadmissible point,  $x_{(\infty)} \notin \mathcal{D}_P$ .

#### 5. SCP with inner-convex approximations

If the convex programming problems within the SCP are constructed with inner-convex approximations the SCP is recursively feasible, descending, and convergent [45]. No trust regions are required and the solution is always improving as the cost decreases between iterations.

Inner-convex approximations are those convex approximations,  $f_{\text{cvx}}^{(x_e)}$ , that meet the three following conditions:

$$f_{\text{cvx}}^{(x_e)}(x) \geq f(x), \quad (7a)$$

$$f_{\text{cvx}}^{(x_e)}(x_e) = f(x_e), \quad (7b)$$

$$\nabla f_{\text{cvx}}^{(x_e)}(x_e) = \nabla f(x_e), \quad (7c)$$

and additionally, as a convex function,  $f_{\text{cvx}}^{(x_e)}$ , must also obey for  $\tau \in [0, 1]$  [4],

$$f_{\text{cvx}}^{(x_e)}(\tau x_1 + (1-\tau)x_2) \geq \tau f_{\text{cvx}}^{(x_e)}(x_1) + (1-\tau) f_{\text{cvx}}^{(x_e)}(x_2). \quad (7d)$$

For completeness we now introduce the theorem and proof of the SCP's recursive feasibility, descent, and convergence, first credited to Marks and Wright [45].

**Theorem 1.** *An SCP that uses inner-convex approximations, as defined in Eq. (7), is an infinite descent algorithm,  $f_0(x_{(k)}) \leq f_0(x_{(k-1)})$ , guaranteeing convergence to a locally optimal solution,  $x_{(\infty)} = x_P^*$ , and with all its iterates remaining admissible points of the original non-convex problem,  $x_{(k)} \in \mathcal{D}_P$ , as long as the first convex problem in the sequence  $P_{\text{cvx}}^{(1)}$  is feasible,  $\mathcal{D}_{P_{\text{cvx}}^{(1)}} \neq \emptyset$ , and as long as the original non-convex problem  $P$  is lower bounded  $f_0(x) > -\infty$ .*

The proof of Theorem 1 is supported by the following two propositions:

**Proposition 1.** *The admissible set of the convexified problem  $P_{\text{cvx}}^{(k)}$  is a subset of the original problem's  $P$  admissible set  $\mathcal{D}_{P_{\text{cvx}}^{(k)}} \subseteq \mathcal{D}_P$ .*

PROOF. Invoking Eq. (7a)

$$f_i(x) \leq f_{\text{cvx},i}^{(k-1)}(x), \quad \text{for } i = 1 \dots p, \quad (8)$$

shows that if a point  $x$  meets the convexified constraints,  $x \in \mathcal{D}_{P_{\text{cvx}}^{(k)}}$ , i.e.,  $f_{\text{cvx},i}^{(k-1)}(x) \leq 0$ , it also meets the original non-convex constraints, i.e.,  $f_i(x) \leq 0$ , hence  $x \in \mathcal{D}_P$   $\square$

**Corollary 1.** *From Proposition 1 it follows that the solution to the convexified problem  $P_{\text{cvx}}^{(k)}$  is an admissible point of the original non-convex problem  $P$ :  $x_{(k)} \in \mathcal{D}_P$ .*

**Proposition 2.** *The convexified problem  $P_{\text{cvx}}^{(k)}$  is feasible if an admissible point to the original non-convex problem  $P$  is used as the expansion point of the convex approximation,  $f_{\text{cvx},i}^{(k-1)}$ .*

PROOF. Given Eq. (7b), if the expansion point  $x_{(k-1)}$  is an admissible point to problem  $P$  then,  $x_{(k-1)}$  is also an admissible point to problem  $P_{\text{cvx}}^{(k)}$ , making the convexified problem  $P_{\text{cvx}}^{(k)}$  feasible as  $x_{(k-1)} \in \mathcal{D}_{P_{\text{cvx}}^{(k)}} \Rightarrow \mathcal{D}_{P_{\text{cvx}}^{(k)}} \neq \emptyset$   $\square$

PROOF (THEOREM 1). The corollary of Proposition 1 shows that if the first convexified problem  $P_{\text{cvx}}^{(1)}$  is feasible then its solution,  $x_{(1)}$ , is an admissible point of the original non-convex problem  $P$ . If that first solution,  $x_{(1)}$ , is used as the expansion point to obtain the next convexified problem  $P_{\text{cvx}}^{(2)}$ , Proposition 2 ensures that this second problem is also feasible. If we continue to use the previous problem's solution as the expansion point to obtain the next convexified problem it follows that all the following convexified problems  $P_{\text{cvx}}^{(k)}$  are feasible and all their solutions are admissible points of the original non-convex problem  $P$ ,  $x_{(k)} \in \mathcal{D}_P$ . Recursive feasibility is then established.

Given Eq. (7a) and (7b),

$$f_0(x_{(k)}) \leq f_{\text{cvx},0}^{(k-1)}(x_{(k)}) \leq f_{\text{cvx},0}^{(k-1)}(x_{(k-1)}) = f_0(x_{(k-1)}), \quad (9)$$

and it is shown that the cost between iterations is non-strictly decreasing, i.e., making the SCP procedure an infinite descent algorithm.

As the problem is lower bounded,  $f_0(x) > -\infty$ , the sequence must be convergent:

$$f_{\text{cvx},0}^{(\infty-1)}(x_{(\infty)}) = f_{\text{cvx},0}^{(\infty)}(x_{(\infty+1)}) = f_0(x_{(\infty)}), \quad (10)$$

$$x_{(\infty)} = x_{(\infty+1)}. \quad (11)$$

As  $x_{(\infty)}$  is the solution to problem  $P_{\text{cvx}}^{(\infty)}$  it automatically satisfies the KKT conditions for problem  $P_{\text{cvx}}^{(\infty)}$ . Then, as  $x_{(\infty)}$  is an admissible point of the non-convex problem  $P$  it satisfies its primal feasibility conditions, Eqs (1a) and (1b). Using Eq. (7a) and (7c) it is shown that

$$f_{\text{cvx}}^{(\infty)}(x_{(\infty)}) = f(x_{(\infty)}), \quad (12)$$

$$\nabla f_{\text{cvx}}^{(\infty)}(x_{(\infty)}) = \nabla f(x_{(\infty)}), \quad (13)$$

and thus  $x_{(\infty)}$ , and its associated optimal dual variables  $\nu_{(\infty)}, \mu_{(\infty)}$ , also meet the dual feasibility, (Eq. 1c) complementary slackness (Eq. 1d), and stationarity (Eq. 1e) conditions of the non-convex problem  $P$ . This allows to conclude that  $x_{(\infty)}$  is a locally optimal solution to the non-convex problem  $P$ :  $x_{(\infty)} = x_P^*$   $\square$

Two brief remarks are now offered about the proposed SCP. 1) to ensure that the first convexified problem in the sequence is feasible, an admissible initial guess,  $x_{(0)} \in \mathcal{D}_P$ , can be used to obtain the first convexified problem. If an admissible initial guess is not available, then the first problem in the SCP procedure may be infeasible, even if the original non-convex problem  $P$  is feasible. 2) the SCP procedure is local. The final solution potentially depends on the initial guess provided. Multiple initial guesses can be used to seed multiple SCP procedures, and eventually select the converged solution with the lowest cost.

## 6. Approaches to obtain inner-convex approximations

In the literature we can find several methods to generate inner-convex approximations. These are briefly reviewed in this section.

If a function is expressed as the difference of two convex functions  $c_1(x), c_2(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ ,

$$f(x) = c_1(x) - c_2(x). \quad (14)$$

an inner-convex approximation can be obtained by linearizing the concave component, i.e.,  $-c_2(x)$ , as follows:

$$f_{\text{cvx}}^{(x_e)}(x) = c_1(x) - c_2(x_e) - \delta x^T \nabla c_2(x_e). \quad (15)$$

These difference of convex functions are typically referred to as d.c., and the resulting SCP obtained when the concave component is linearized is known as the convex-concave procedure [49, 50]. This convex-concave procedure is thus a particular case of the proposed SCP with inner-convex approximations.

Problem formulations with d.c. functions naturally appear in many optimization problems [51, 52]. A common example in trajectory optimization are keep-out zones, which are usually formulated as concave inequality constraints (i.e., a d.c. with  $c_1(x) = 0$ ). Several authors have rediscovered the recursively feasible and convergent properties that an SCP enjoys when concave inequalities are linearized. [30, 25, 7].

It has been shown that d.c. decompositions exist for any twice differentiable function  $f \in C^2$  [53, 54], and several approaches exist to find them. For example, if  $f$  is a polynomial, a d.c. decomposition that is made by convex sum-of-squares polynomials (sos-convex in the following) can be found [55]. There are infinite sos-convex representations and thus in [55] different methods to optimize the sos-convex decomposition for a faster rate of convergence are proposed. Interestingly enough, finding these optimized sos-convex decompositions is a semidefinite programming problem (second order cone or linear programming problem with some additional relaxations) [55].

Another approach to find d.c. decompositions is the regularization method [54, 56, 57, 58, 59]. In a regularization, a d.c. decomposition is obtained by using a sufficiently strong convex function  $\phi(x) : \mathbb{R}^n \rightarrow \mathbb{R}$  such that

$$f(x) = \underbrace{\phi(x)}_{c_1(x)} - \underbrace{(\phi(x) - f(x))}_{c_2(x)}, \text{ for } x \in \mathcal{T}, \quad (16)$$

is d.c. in the trust region  $\mathcal{T}$ .

If the  $f$  function is Lipschitz continuous on  $x \in \mathcal{T}$ , with  $K$  denoting the Lipschitz constant, a d.c. representation of  $f$  over  $\mathcal{T}$  is

$$f(x) = \underbrace{K\|x\|^2}_{c_1(x)} - \underbrace{(K\|x\|^2 - f(x))}_{c_2(x)}, \text{ for } x \in \mathcal{T}. \quad (17)$$

Using the same principle, and using the Lipschitz constant of the Hessian, inner-convex approximations based on higher-order regularizations are also possible. For example, [60] introduces a cubic regularization that enjoys a quadratic rate of convergence.

Up to this point, we have shown that inner-convex approximations for d.c. functions are easy to obtain. However, if the function is not d.c., either a semidefinite program needs to be solved to obtain a sos-convex decomposition or, for regularization-based approaches, we need to estimate the Lipschitz constant.

## 7. Taylor-based inner-convex approximations

A new computationally inexpensive method to obtain an inner-convex approximation of a non-convex analytic function is here presented. This approach can be seen as a high order regularization based on the convexification of the high-order terms of a Taylor series expansion. With these inner-convex approximations the SCP's local rate of convergence is superlinear, or even quadratic when converging to a non-degenerate minima.

To simplify the notation of the high-order terms of the Taylor series let's use a multi-index notation, with an  $n$ -tuple of natural numbers  $\mathbb{N}$ , denoted by  $\alpha$  [61, Section 1.1]:

$$\alpha = (\alpha_1, \alpha_2 \dots \alpha_n), \quad \text{with } \alpha_j \in \mathbb{N} \quad (18a)$$

$$|\alpha| = \alpha_1 + \alpha_2 + \dots + \alpha_n \quad (18b)$$

$$\alpha! = \alpha_1! \alpha_2! \dots \alpha_n! \quad (18c)$$

$$x^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} \dots x_n^{\alpha_n} \quad (18d)$$

$$\partial^\alpha f = \frac{\partial^{|\alpha|} f}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \dots \partial x_n^{\alpha_n}} \quad (18e)$$

Using a multi-index notation, the Taylor series of the analytic function  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is written as:

$$f(x) = \sum_{|\alpha|=0}^{\infty} \frac{\partial^\alpha f(x_e)}{\alpha!} \delta x^\alpha. \quad (19)$$

To obtain an inner-convex approximation of  $f$  we propose to convexify all of the terms corresponding to the derivatives of

second order and higher. The Hessian,  $\mathbf{H}$ , as a symmetric matrix, can be decomposed in a positive and a negative semi-definite part [62, Section 7.2]:

$$\mathbf{H} = \mathbf{H}^+ + \mathbf{H}^- \quad (20)$$

$$\mathbf{H}^+ \geq 0, \quad \mathbf{H}^- \leq 0. \quad (21)$$

The positive semi-definite part is recovered using the matrix of eigenvalues  $\mathbf{\Lambda}$  and the matrix of eigenvectors  $\mathbf{W}$  of the Hessian:

$$\mathbf{H}^+ = \mathbf{W}\mathbf{\Lambda}^+\mathbf{W}^T, \quad (22)$$

where  $\mathbf{\Lambda}^+$  is the matrix of eigenvalues containing only the positive ones.

The higher order terms ( $|\alpha| \geq 3$ ) of the Taylor series form homogeneous, polynomial forms

$$\sum_{|\alpha|} \frac{\partial^\alpha f(x_e)}{\alpha!} \delta x^\alpha = \mathbf{T} \delta x^\alpha = \sum_{j_1 \dots j_{|\alpha|}=1}^n T_{j_1 \dots j_{|\alpha|}} \delta x_{j_1} \dots \delta x_{j_{|\alpha|}}, \quad (23)$$

with  $\mathbf{T} \in \mathbb{R}^{n \times n \times \dots \times n}$  denoting a supersymmetric tensor—invariant under permutation on any of its indices.

If the eigenvalues and eigenvectors of these tensors could be extracted a similar procedure to the one followed to convexify the Hessian could be pursued. Although the eigenvalues of tensors can be computed using a sum-of-squares and a semidefinite programming approach [63] (showing a connection with sos-convex decompositions, Lasserre's relaxations, and non-convex programming), the computational cost quickly becomes prohibitive as the order, or number of variables and constraints grows.

Our proposed approach takes a computationally efficient route and a convex overestimation of the terms of third order or higher,  $\text{cvx}(\mathbf{T} \delta x^\alpha)$ , is obtained as follows:

$$\text{cvx}(\mathbf{T} \delta x^\alpha) = \sum_{i=1}^n \left( |T_{\text{diag},i} \delta x_i^{|\alpha|}|^+ + T_{\text{cvx},i} |\delta x_i^{|\alpha|}| \right). \quad (24)$$

The  $T_{\text{diag},i}$  coefficient is the element of the tensor's main diagonal associated with  $\delta x_i$ ,

$$T_{\text{diag},i} = T_{i,i,\dots,i}. \quad (25)$$

As  $|\cdot|^+$  denotes the positive part operator, the expression  $|T_{\text{diag},i} \delta x_i^{|\alpha|}|^+$  is convex and it is an overestimator of the  $\mathbf{T} \delta x^\alpha$  terms associated with the  $\mathbf{T}$  tensor's main diagonal.

The  $T_{\text{cvx},i}$  coefficient is computed as the sum of the absolute values of all of the coefficients in  $\mathbf{T}$  associated with  $\delta x_i$  (with the exception of the coefficients in the main diagonal):

$$T_{\text{cvx},i} = \sum_{j_1 \dots j_{|\alpha|=1}}^n \begin{cases} 0 & \text{if all } j_1 \dots j_{|\alpha|} \text{ are different than } i \\ 0 & \text{if all } j_1 \dots j_{|\alpha|} \text{ are equal to } i \\ |T_{j_1 \dots j_{|\alpha|}}| & \text{otherwise} \end{cases} \quad (26)$$

As  $T_{\text{cvx},i} \geq 0$  the term  $T_{\text{cvx},i} |\delta x_i^{|\alpha|}|$  is convex and it follows that

$$\text{cvx}(\mathbf{T} \delta x^\alpha) \geq \mathbf{T} \delta x^\alpha. \quad (27)$$

If the coefficients  $T_{\text{cvx},i}$  and  $T_{\text{diag},i}$  of a particular order,  $|\alpha|$ , are denoted by  $T_{\text{cvx},i}^{(|\alpha|)}$  and  $T_{\text{diag},i}^{(|\alpha|)}$ , the proposed inner-convex approximation of  $f$ , satisfying the conditions in Eq. (7), is

$$f_{\text{cvx}}^{(x_e)}(x) = f(x_e) + \delta x^T \nabla f(x_e) + \frac{1}{2} \delta x^T \mathbf{H}^+(x_e) \delta x + \sum_{|\alpha|=3}^{|\alpha|=\infty} \sum_{i=1}^n \left( \left| T_{\text{diag},i}^{(|\alpha|)} \delta x_i^{|\alpha|} \right|^+ + T_{\text{cvx},i}^{(|\alpha|)} |\delta x_i^{|\alpha|} \right). \quad (28)$$

### 7.1. Truncating the series

The inclusion of high order terms might need to be truncated as the Taylor series may be infinite or the computational effort required to compute high-order terms may be too high. To truncate the series and preserve the inner-convex properties a high-order regularization can be added as follows,

$$f_{\text{cvx}}^{(x_e)}(x, M) = \overbrace{f_{\text{cvx}}^{(x_e)}(x)}^{\text{truncated at order } |\alpha|} + \frac{M}{(|\alpha| + 1)!} \|\delta x\|^{|\alpha|+1}, \quad (29)$$

for  $M \geq 0$ .

If the condition  $f_{\text{cvx}}^{(k-1)}(x_{(k)}, M) \geq f(x_{(k)})$  is not met after solving the convex problem  $P_{\text{cvx}}^{(k)}(M)$ , the problem is re-solved with a new, larger  $M$ , i.e.,  $M_{\text{new}} > M_{\text{old}}$ . As long as the  $|\alpha|+1$  order derivative of  $f$  is Lipschitz continuous the  $f_{\text{cvx}}^{(k-1)}(x_{(k)}, M) \geq f(x_{(k)})$  condition will be eventually met by a large-enough  $M$ . Adding the high-order regularization may potentially reduce the rate of convergence and therefore it is advised to start with  $M = 0$ , only increasing  $M$  when strictly required.

This truncation with a high-order regularization can be seen as an extension of the cubic regularization approach developed by Nesterov [60].

### 7.2. Quadratic rate of convergence

The proposed inner-convex approximation of the Taylor series provides the SCP with a local superlinear rate of converge, which can be proven to be quadratic when converging to a non-degenerate minima.

It has been shown that a linearization-based SCP is super-linearly convergent around its convergence point  $x_{(\infty)}$  [44]. As the SCP converges,

$$\lim_{k \rightarrow \infty} \delta x = 0, \quad (30)$$

making the high-order terms become increasingly small

$$f_{\text{cvx}}^{(x_e)}(x) = f(x_e) + \delta x^T \nabla f(x_e) + O(\|\delta x\|^2), \quad (31)$$

and thus the Taylor-based convexification inherits the superlinear convergence of linearization-based SCPs.

But as the Taylor-based convexification preserves the Hessian, as long as  $\mathbf{H}(x) > 0$ , the resulting SCP exhibits a quadratic rate of convergence for non-degenerate minima.

**Theorem 2.** *The SCP exhibits a quadratic rate of convergence,*

$$\|x_{(k+1)} - x_{(k)}\| \leq \gamma_{(k)} \|x_{(k)} - x_{(k-1)}\|^2,$$

for some  $\gamma_{(k)} > 0$ , around a non-degenerate local minima,  $\mathbf{H}(x) > 0$ , of Problem  $P$ .

The cubic regularization approach proposed by Nesterov [60] also has a quadratic rate of convergence, and the proof provided here is based on the proof provided for that case.

**PROOF.** Let us start by introducing some additional notation. Let the spectrum of a of an  $n \times n$  symmetric matrix  $\mathbf{A}$  be denoted by  $\{\lambda_i(\mathbf{H})\}_{i=1}^n$ . Also, let's assume that the eigenvalues are ordered in a decreasing sequence:

$$\lambda_1(\mathbf{A}) \geq \dots \geq \lambda_n(\mathbf{A}), \quad (32)$$

and that the norm of a real symmetric matrix is defined with its standard spectral form as [64, Chapter 5]

$$\|\mathbf{A}\| = \sqrt{\lambda_1(\mathbf{A}^T \mathbf{A})}. \quad (33)$$

Having introduced this notation, let us suppose that the optimal dual variables at each iteration of the SCP are known, then, the Lagrangian of the problem with the non-convex functions is

$$L_{(k)}(x) = f_0(x) + \sum_{i=1}^p v_{(k),i}^* f_i(x) + \sum_{i=1}^q \mu_{(k),i}^* h_i(x). \quad (34)$$

Solving the constrained problem  $P$  is equivalent to solving an unconstrained problem with the Lagrangian in Eq. 34 as the cost function (dual problem). Without any loss of generality, let's use this unconstrained version of the problem for the proof.

Further assume that the Hessian of the Lagrangian,  $\mathbf{H}_L$  is locally Lipschitz continuous,

$$\|\mathbf{H}_L(x_2) - \mathbf{H}_L(x_1)\| \leq K_L \|x_2 - x_1\|. \quad (35)$$

Integrating both sides of the inequality yields:

$$\|\nabla L(x_2) - \nabla L(x_1) - \mathbf{H}_L(x_1)(x_2 - x_1)\| \leq \frac{K_L}{2} \|x_2 - x_1\|^2. \quad (36)$$

Using the inner-convex approximations at iteration  $k + 1$  the convexified Lagrangian is,

$$L_{\text{cvx}}^{(k)}(x) = f_{\text{cvx},0}^{(k)}(x) + \sum_{i=1}^p v_{(k+1),i}^* f_{\text{cvx},i}^{(k)}(x) + \sum_{i=1}^q \mu_{(k+1),i}^* h_i(x), \quad (37)$$

and, as a sum of convex and affine functions,  $L_{\text{cvx}}^{(k)}$  is convex and takes the form of a convexified Taylor series as defined in Eq. (28).

The stationary condition (see Eq. (1e)) imposes that the gradient of the convexified Lagrangian  $L_{\text{cvx}}^{(k)}$  at the optimal point  $x_{(k+1)}$  is zero.

$$\nabla L_{\text{cvx}}^{(k)}(x_{(k+1)}) = 0. \quad (38)$$

Expanding it as Taylor series yields

$$\nabla L(x_{(k)}) + \mathbf{H}_L^+(x_{(k)})(x_{(k+1)} - x_{(k)}) + \dots = 0. \quad (39)$$

If it is further assumed that the current iterate  $x_{(k)}$  is close to a non-degenerate local minima with  $\mathbf{H}_L > 0$ . With this assumption, and noting that the high-order terms of  $L_{\text{cvx}}^{(k)}$  are all convex, it follows that

$$\|x_{(k+1)} - x_{(k)}\| = \|(\mathbf{H}_L(x_{(k)}) + \dots)^{-1} \nabla L(x_{(k)})\| \leq \frac{\|\nabla L(x_{(k)})\|}{\lambda_n(\mathbf{H}_L(x_{(k)}))}. \quad (40)$$

Additionally, the Hessian of the convexified Lagrangian  $\mathbf{H}_{L_{\text{cvx}}^{(k)}}$  is locally Lipschitz continuous

$$\left\| \mathbf{H}_{L_{\text{cvx}}^{(k)}}(x) - \mathbf{H}_{L_{\text{cvx}}^{(k)}}(x_{(k)}) \right\| \leq K_{L_{\text{cvx}}^{(k)}} \|x - x_{(k)}\|, \quad (41)$$

allowing to bound the high order terms

$$\begin{aligned} L_{\text{cvx}}^{(k)}(x) &\leq L(x_{(k)}) + \nabla L(x_{(k)})(x - x_{(k)}) \\ &+ (x - x_{(k)})^T \mathbf{H}_L(x_{(k)})(x - x_{(k)}) + \frac{K_{L_{\text{cvx}}^{(k)}}}{6} \|x - x_{(k)}\|^3. \end{aligned} \quad (42)$$

With this observation, the optimality condition in Eq. (40) yields the following inequality:

$$\left\| \nabla L(x_{(k)}) + \mathbf{H}_L(x_{(k)})(x_{(k+1)} - x_{(k)}) \right\| \leq \frac{1}{2} K_{L_{\text{cvx}}^{(k)}} \|x_{(k+1)} - x_{(k)}\|^2. \quad (43)$$

Combining Eq. (36) with Eq. (43) and observing the triangle inequality yields the following expression:

$$\left\| \nabla L(x_{(k+1)}) \right\| \leq \frac{K_L + K_{L_{\text{cvx}}^{(k)}}}{2} \|x_{(k+1)} - x_{(k)}\|^2. \quad (44)$$

Therefore it follows that

$$\left\| x_{(k+1)} - x_{(k)} \right\| \leq \frac{\left\| \nabla L(x_{(k)}) \right\|}{\lambda_n(\mathbf{H}_L(x_{(k)}))} \leq \frac{(K_L + K_{L_{\text{cvx}}^{(k)}})}{2\lambda_n(\mathbf{H}_L(x_{(k)}))} \|x_{(k)} - x_{(k-1)}\|^2, \quad (45)$$

thus establishing the algorithm's local quadratic rate of convergence  $\square$

## 8. Application to a trajectory optimization problem

To illustrate the use of the proposed SCP procedure with Taylor-based inner-convex approximations, an illustrative example is offered. In this example, inspired by the one provided in [46, 44], the trajectory of an aerial vehicle subject to drag, control bounds, and a keep-out zone constraint is optimized. The dynamics of the system is written as

$$\ddot{r} = \frac{1}{m} (F - k_d \|\dot{r}\| \dot{r}), \quad (46)$$

with  $r \in \mathbb{R}^3$  denoting the position of the vehicle,  $m$  its mass,  $F \in \mathbb{R}^3$  its thrust, and  $k_d$  the vehicle's drag coefficient.

The vehicle is subject to the following control constraint,

$$\|F\| \leq F_{\max}, \quad (47)$$

and it must respect the following keep-out zone constraint,

$$(r_1^2 + r_2^2)^2 + r_3^4 - b^4 - 10r_3(r_1^2 r_2 - r_2^2 r_1) \geq 0. \quad (48)$$

The initial and final conditions, as well as the time to complete the maneuver,  $t_f$ , are specified:

$$r(0) = r_0, \quad \dot{r}(0) = \dot{r}_0, \quad (49a)$$

$$r(t_f) = r_f, \quad \dot{r}(t_f) = \dot{r}_f. \quad (49b)$$

The cost to be minimized is expressed as

$$J = \int_0^{t_f} \|F\| dt. \quad (50)$$

This optimal control problem is discretized with a direct transcription method with  $N$  nodes, applying a first-order-hold on the accelerations  $\ddot{r}$ , and estimating the cost with a trapezoidal integration scheme. The resulting non-convex programming problem is:

$$\text{minimize: } J = \sum_{i=1}^{N-1} \frac{\|F^{[i+1]}\| - \|F^{[i]}\|}{2} (t^{[i+1]} - t^{[i]}) \quad (51)$$

$$\text{subject to: } F^{[i]} = m\ddot{r}^{[i]} + k_d \|\dot{r}^{[i]}\| \dot{r}^{[i]}, \quad (46)$$

$$\|F^{[i]}\| \leq F_{\max}, \quad (47)$$

$$(r_1^{[i]2} + r_2^{[i]2})^2 - 10r_3^{[i]}(r_1^{[i]2} r_2^{[i]} - r_2^{[i]2} r_1^{[i]}) + r_3^{[i]4} - b^4 \geq 0 \quad (48)$$

$$r^{[1]} = r_0, \quad \dot{r}^{[1]} = \dot{r}_0 \quad (49a)$$

$$r^{[N]} = r_f, \quad \dot{r}^{[N]} = \dot{r}_f \quad (49b)$$

Note that this problem is of the class  $P$  with  $x = \{r^{[1]} \dots r^{[N-1]}\}$ . The cost as well as the control and keep-out constraints are non-convex functions.

For the cost and control constraints, the underlying non-convex functions have a Taylor series expansion with infinite terms and an inner-convex approximation truncated to order  $|\alpha| = 3$  is used during the SCP. No regularization is added.

The keep-out zone constraint can be re-written as follows

$$\underbrace{-(r_1^2 + r_2^2)^2 - r_3^4 + b^4}_{\text{sos-concave}} + \underbrace{+10r_3(r_1^2 r_2 - r_2^2 r_1)}_{\text{non-convex}} \leq 0. \quad (52)$$

showing that it has a concave part—specifically sos-concave—and a non-convex part. The Taylor series of the non-convex part is finite, reaching fourth order. To obtain inner-convex approximation of the non-convex part we convexify all the terms of the Taylor series expansion up to fourth order. Once the non-convex part of Eq. (52) has been convexified the function becomes d.c. and thus we know that the concave part needs to be linearized to obtain an inner-convex approximation. This example problem shows how different methods can be used to generate inner-convex approximations.

A Monte Carlo analysis with a total of 1000 cases has been conducted. The parameters used are provided in Table 1. Based on the randomized initial and final conditions an initial guess is analytically generated for each case. In particular, a bang-bang type control keeping the acceleration at two different constant values during the first and second halves of the maneuver is used to generate the initial guess. This solution meets the initial and final conditions but it may violate the thrust and/or keep-out zone constraints. As a consequence, the initial guess may not be an admissible solution of the non-convex problem. Therefore, we cannot guarantee a recursively feasible SCP procedure based on an inadmissible initial guess.

To generate an admissible solution from the initial guess a penalty method is used. This step serves to illustrate how inner-convex approximations can also be applied in conjunction with



Table 1: Simulation parameters (non-dimensional).

Parameter	Value
<i>Problem</i>	
Mass $m$	1
Final time $t_f$	15
Drag coefficient $k_d$	0.25
Max. thrust $F_{\max}$	1.5
Keep-out zone parameter $b$	3.5
Number of nodes $N$	25
SCP stopping criterion $\epsilon$	$0.01 f_0(\mathbf{x}_{(k)}^*)$
Max. num. of SCP iterations	50
<i>Monte Carlo</i>	
Number of cases	1000
Initial position $r_0$	Random with $\ r_0\  = 6$
Final position $r_f$	$r_f = -r_0$
Initial velocity $\dot{r}_0$	Random with $\ \dot{r}_0\  = 1$
Final velocity $\dot{r}_f$	Random with $\ \dot{r}_f\  = 1$

penalty methods and help obtain an admissible solution. In the penalty method version of the problem, the maximum thrust constraint is relaxed with a slack variable  $s$ ,

$$\|F^{[i]}\| - F_{\max} \leq s^{[i]}, \quad (53a)$$

$$s^{[i]} \geq 0. \quad (53b)$$

and the cost function in Eq. (51) replaced by

$$J = \sum_{n=1}^N s^{[i]}. \quad (54)$$

The penalty method version of the problem is then solved using the same SCP approach until an admissible solution is obtained. Given that inner-convex approximations are used and that the problem's feasibility is guaranteed by the introduction of the slack variable, the sequence is bound to converge. However, the converged solution of the penalty version of the problem may be an inadmissible point of the original problem.

When an admissible solution is found, the slack variable is removed, the original cost function in Eq. (51) restored, and the SCP continues. As an admissible solution is now used as a seed, the remaining iterations of the SCP procedure are recursively feasible and guaranteed to converge to a locally optimal solution.

### 8.1. Illustrative result of one case

For illustration purposes, the results of one case with the initial and terminal conditions indicated in Table 2 are shown in Figs. 1-3. The keep-out zone and the trajectories corresponding to the initial guess, the first admissible solution, and the converged solution are shown in Fig. 1. In this particular case, four iterations with the penalty method were needed to produce an admissible solution. Eight more iterations were then required to converge down to the specified level, bringing the total number of convex problems solved to 12. Figure 2 shows the decreasing cost along with the inner-convex approximation of the cost,

Table 2: Initial and terminal condition of illustrative case.

Parameter	Value
Initial position $r_0$	$r_0 = [-2.61, 0.53, -5.38]^T$
Final position $r_f$	$r_f = -r_0$
Initial velocity $\dot{r}_0$	$\dot{r}_0 = [-0.62, 0.77, -0.14]^T$
Final velocity $\dot{r}_f$	$\dot{r}_f = [0.64, 0.75, 0.15]^T$

which is always overestimating it. The non-convex thrust and keep-out zone constraints are also overestimated by the inner-convex approximation,  $f_{\text{cvx},i}^{(k-1)} - f_i \geq 0$ , as shown in Figs. 3a and 3b. In all of the SCP iterations within the 1000 cases, the thrust was always overestimated, showing that when the Taylor series is truncated a high order regularization may not be required in all cases.

### 8.2. Results of the Monte Carlo analysis

Analyzing the Monte Carlo results provides further insight into the proposed method, specifically with respect to its computational properties.

Figure 4 shows the convergence of the proposed method on this illustrative problem. The penalty method is able to find an admissible solution in 98.1% of the cases. Once an admissible solution is found the SCP enjoys guaranteed convergence, thus achieving a 100 % success rate.

Figure 5 compares the cost of the solution offered by the proposed SCP with the solution obtained with the pseudospectral optimal control package GPOPS-II [65]. Here the GPOPS-II solution is used as a proxy for the globally optimal solution. When the cost of both methods are compared it can be argued that the proposed SCP produces competitive solutions with only minor overcosts. For example, 50% of the cases show an overcost below 5%, while the overcost of 90% of the cases is below 12%.

Where the proposed method shows one of its most remarkable advantages is in its low computational cost. A single-threaded C implementation of the proposed SCP has been tested on a laptop and on two embedded computing platforms: an NVIDIA TX2 and Raspberry Pi 3 B+ (RPi). To solve the convex programming problems the interior-point solver IPOPT [66], paired with HSL's MA27 linear solver routine [67], is used.

Figure 6 shows an histogram of the computational time required to solve the problems on these three platforms, with the solid lines indicating the averages. For reference, the average time taken by GPOPS-II on the laptop is also shown. The GPOPS-II solution uses the default settings (with IPOPT used as the underlying nonlinear programming solver). On a laptop machine the proposed SCP is shown to be a 100 times faster than GPOPS-II. Additionally the results on the two consumer-grade embedded computers provide empirical evidence that the proposed approach is suitable for implementation on embedded computers and fast enough to be used for real-time applications.

Finally, Fig 7 shows how the time is spent solving each convex program within the SCP. Percentages for each of the major tasks are approximately the same across all platforms, but

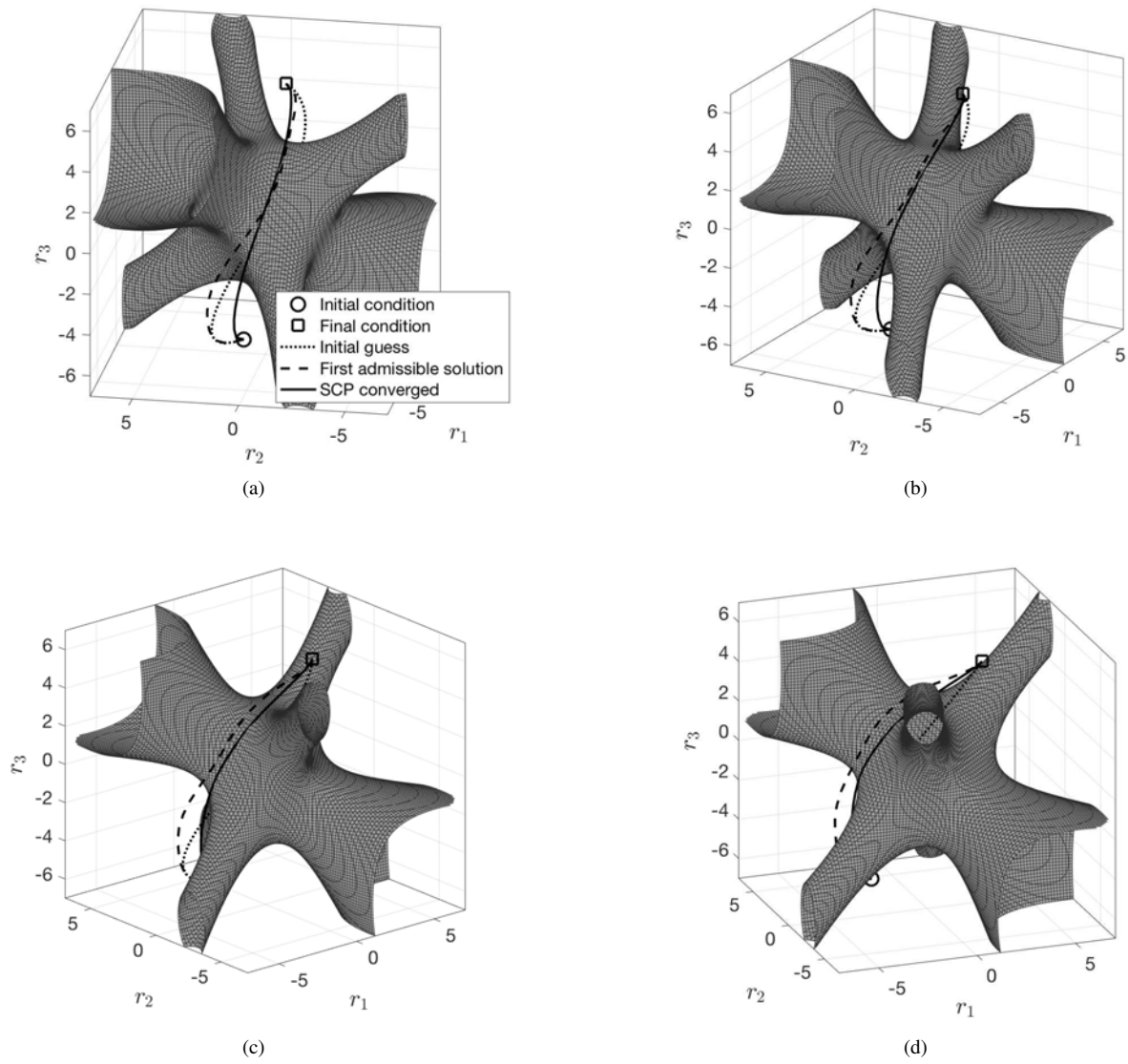


Figure 1: Initial guess, first admissible solution, and converged trajectory at different viewing angles.

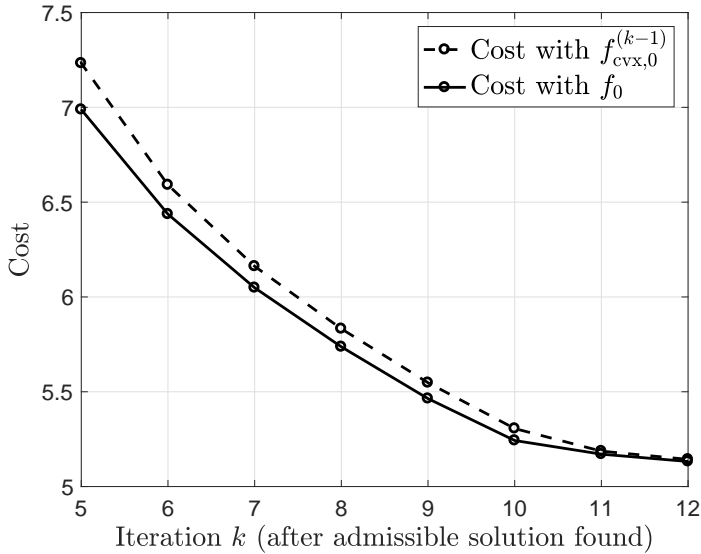


Figure 2: Evolution of the cost during the SCP.

for readability they are only explicitly given for the RPi. What Fig 7 shows is that the proposed Taylor-based approach to obtain inner-convex approximations is computationally efficient, taking less than 1.5% of the time per iteration. Also the time taken to evaluate the nonlinear functions (*i.e.*, Eq. (28)) is small, just below 6.5%. The bulk of the time, > 92%, is used to solve a linear system at each iteration of the IPOPT's interior-point method.

### 8.3. Applicability to other problems

The proposed SCP and Taylor-based convexification approach can be applied to a wide range of problems. The example provided here is only one of these potential applications. The authors have applied this optimization approach to other problems, namely to the spacecraft attitude maneuver [68].

## 9. Concluding remarks

When inner-convex approximations are used, a recursively feasible, descending, and convergent sequential convex programming (SCP) procedure is obtained for the wide class of non-convex programming problems with linear equality constraints. This SCP procedure avoids the common pitfalls of infeasible iterates and cost increases of linearization-based SCP procedures without the need of trust regions. By convexifying each term of the Taylor series expansion of a non-convex function we obtain a computationally efficient and widely applicable method to generate inner-convex approximations. These Taylor-based approximations preserve the positive semidefinite part of the Hessian and thus provide a locally quadratic rate of convergence when the local minima is non-degenerate. Given the attractive computational properties of the proposed approach it appears that the proposed SCP is suitable for onboard implementation and real-time use. The results obtained when solving a trajectory optimization problem on two embedded computers empirically substantiate this claim.

## References

- [1] P. Tsiotras, M. Mesbahi, Toward an algorithmic control theory, *Journal of Guidance, Control, and Dynamics* 40 (2) (2017) 194–196. doi:10.2514/1.G002754.
- [2] P. Lu, Introducing computational guidance and control, *Journal of Guidance, Control, and Dynamics* 40 (2) (2017) 193–193. doi:10.2514/1.G002745.
- [3] Y. Nesterov, A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, Society for Industrial and Applied Mathematics, 1994. doi:10.1137/1.9781611970791.
- [4] S. Boyd, L. Vandenberghe, *Convex optimization*, Cambridge university press, 2004.
- [5] D. Dueri, B. Açıkmeşe, D. P. Scharf, M. W. Harris, Customized real-time interior-point methods for onboard powered-descent guidance, *Journal of Guidance, Control, and Dynamics* (2016) 1–16doi:10.2514/1.G001480.
- [6] D. P. Scharf, B. Açıkmeşe, D. Dueri, J. Benito, J. Casoliva, Implementation and experimental demonstration of onboard powered-descent guidance, *Journal of Guidance, Control, and Dynamics* (2016) 1–17doi:10.2514/1.G000399.
- [7] J. Virgili-Llop, J. V. Drew, R. Zappulla, M. Romano, A convex programming based guidance algorithm to capture a tumbling object on orbit using a spacecraft equipped with a robotic manipulator, *International Journal of Robotics Research* (Accepted. In press).
- [8] D. G. Hull, Conversion of optimal control problems into parameter optimization problems, *Journal of Guidance, Control, and Dynamics* 20 (1) (1997) 57–60. doi:10.2514/2.4033.
- [9] B. Açıkmeşe, L. Blackmore, Lossless convexification of a class of optimal control problems with non-convex control constraints, *Automatica* 47 (2) (2011) 341–347. doi:10.1016/j.automatica.2010.10.037.
- [10] L. Blackmore, B. Açıkmeşe, J. M. Carson, Lossless convexification of control constraints for a class of nonlinear optimal control problems, *Systems & Control Letters* 61 (8) (2012) 863–870. doi:10.1109/ACC.2012.6314722.
- [11] B. Açıkmeşe, J. M. Carson, L. Blackmore, Lossless convexification of nonconvex control bound and pointing constraints of the soft landing optimal control problem, *IEEE Transactions on Control Systems Technology* 21 (6) (2013) 2104–2113. doi:10.1109/TCST.2012.2237346.
- [12] M. W. Harris, B. Açıkmeşe, Lossless convexification of non-convex optimal control problems for state constrained linear systems, *Automatica* 50 (9) (2014) 2304–2311.
- [13] X. Liu, Z. Shen, P. Lu, Exact convex relaxation for optimal flight of aerodynamically controlled missiles, *IEEE Transactions on Aerospace and Electronic Systems* 52 (4) (2016) 1881–1892. doi:10.1109/TAES.2016.150741.
- [14] D. Verscheure, B. Demeulenaere, J. Swevers, J. D. Schutter, M. Diehl, Time-optimal path tracking for robots: A convex optimization approach, *IEEE Transactions on Automatic Control* 54 (10) (2009) 2318–2327. doi:10.1109/TAC.2009.2028959.
- [15] X. Liu, P. Lu, B. Pan, Survey of convex optimization for aerospace applications, *Astrodynamics* 1 (1) (2017) 23–40. doi:10.1007/s42064-017-0003-8.
- [16] B. Acikmese, S. R. Ploen, Convex programming approach to powered descent guidance for mars landing, *Journal of Guidance, Control, and Dynamics* 30 (5) (2007) 1353–1366. doi:10.2514/1.27553.
- [17] L. Blackmore, B. Açıkmeşe, D. P. Scharf, Minimum-landing-error powered-descent guidance for mars landing using convex optimization, *Journal of guidance, control, and dynamics* 33 (4) (2010) 1161. doi:10.2514/1.47202.
- [18] M. Sagliano, Pseudospectral convex optimization for powered descent and landing, *Journal of Guidance, Control, and Dynamics* (2017) 1–15doi:10.2514/1.G002818.
- [19] R. M. Pinson, P. Lu, Trajectory design employing convex optimization for landing on irregularly shaped asteroids, *Journal of Guidance, Control, and Dynamics* (2018) 1–14doi:10.2514/1.G003045.
- [20] F. Debrouwere, W. Van Loock, G. Pipeleers, Q. T. Dinh, M. Diehl, J. De Schutter, J. Swevers, Time-optimal path following for robots with convex-concave constraints using sequential convex programming, *IEEE Transactions on Robotics* 29 (6) (2013) 1485–1495. doi:10.1109/TR0.2013.2277565.

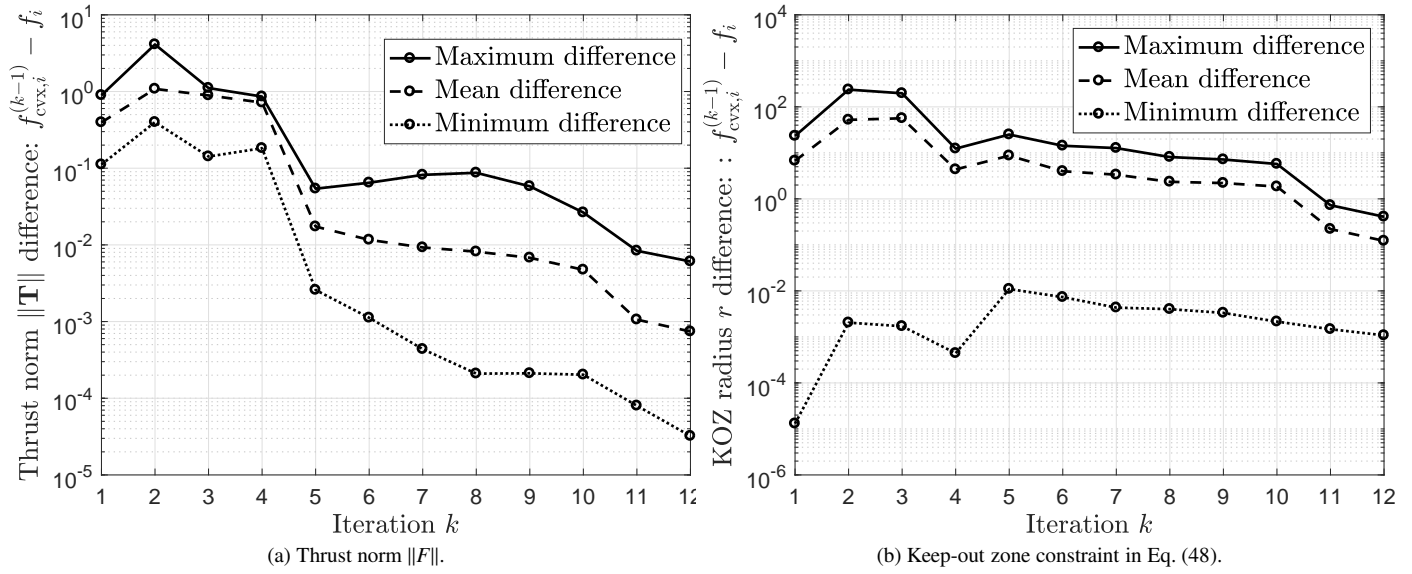


Figure 3: Evolution of the constraint differences,  $f_{cvx,i}^{(k-1)} - f_i \geq 0$ , during the SCP.

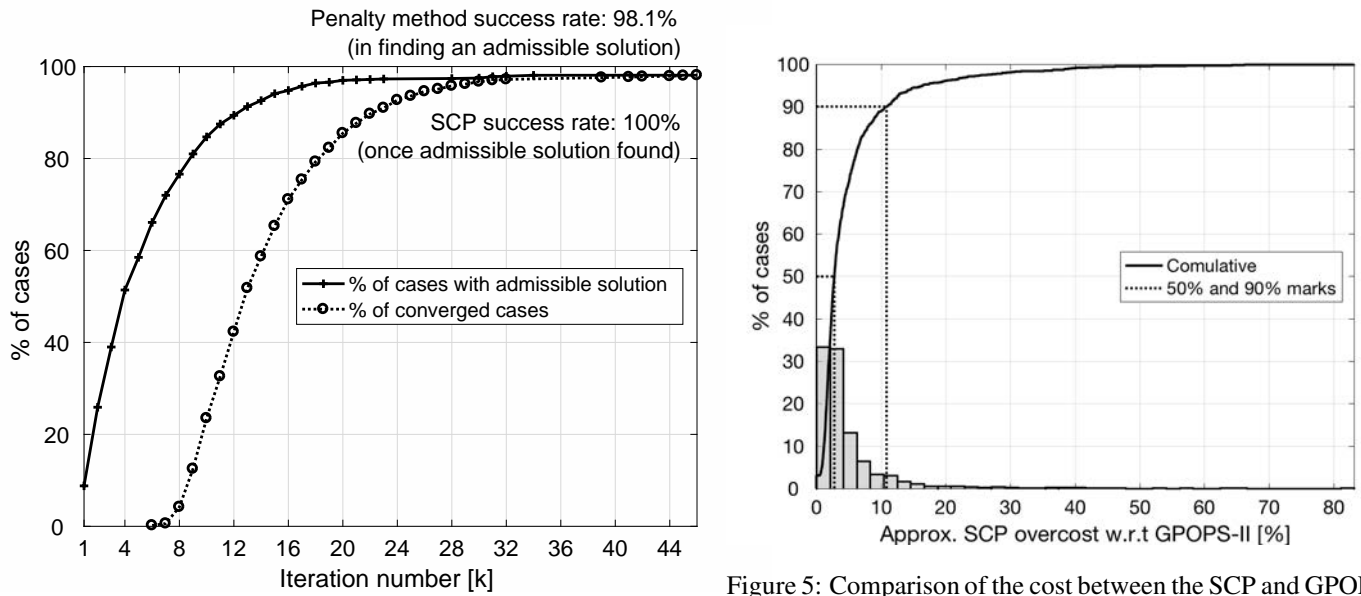


Figure 4: Convergence speed.

Figure 5: Comparison of the cost between the SCP and GPOPS-II.

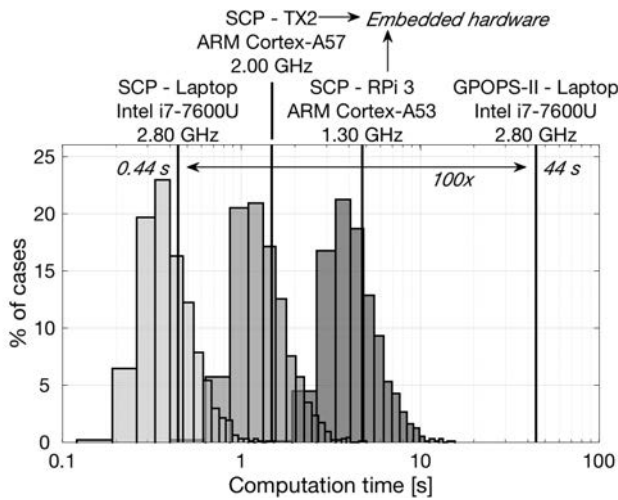


Figure 6: SCP's computational time on different computing platforms, compared to GPOPS-II.

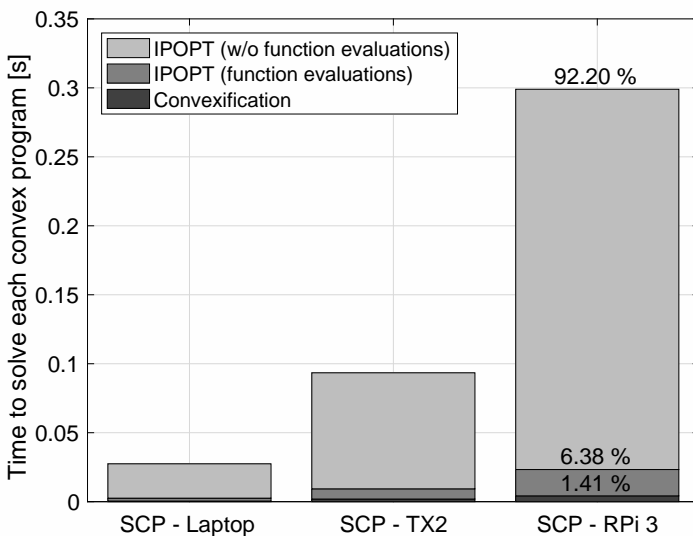


Figure 7: Average time to solve a single convex programming problem within the SCP sequence.

- [21] T. Lipp, S. Boyd, Minimum-time speed optimisation over a fixed path, *International Journal of Control* 87 (6) (2014) 1297–1311. doi:10.1080/00207179.2013.875224.
- [22] J. B. Mueller, R. Larsson, Collision avoidance maneuver planning with robust optimization, in: *International ESA Conference on Guidance, Navigation and Control Systems*, Tralee, County Kerry, Ireland, 2008.
- [23] M. W. Harris, B. Açıkmeşe, Minimum time rendezvous of multiple spacecraft using differential drag, *Journal of Guidance, Control, and Dynamics* 37 (2) (2014) 365–373. doi:10.2514/1.61505.
- [24] P. Lu, X. Liu, Autonomous trajectory planning for rendezvous and proximity operations by conic optimization, *Journal of Guidance, Control, and Dynamics* 36 (2) (2013) 375–389. doi:10.2514/1.58436.
- [25] X. Liu, P. Lu, Solving nonconvex optimal control problems by convex optimization, *Journal of Guidance, Control, and Dynamics* 37 (3) (2014) 750–765. doi:10.2514/6.2013-4725.
- [26] X. Liu, Z. Shen, P. Lu, Solving the maximum-crossrange problem via successive second-order cone programming with a line search, *Aerospace Science and Technology* 47 (2015) 10–20.
- [27] X. Liu, Z. Shen, P. Lu, Entry trajectory optimization by second-order cone programming, *Journal of Guidance, Control, and Dynamics* 39 (2) (2015) 227–241.
- [28] D.-J. Zhao, Z.-Y. Song, Reentry trajectory optimization with waypoint and no-fly zone constraints using multiphase convex programming, *Acta Astronautica* 137 (2017) 60 – 69. doi:10.1016/j.actaastro.2017.04.013.
- [29] J. Alonso-Mora, T. Naegeli, R. Siegwart, P. Beardsley, Collision avoidance for aerial vehicles in multi-agent scenarios, *Autonomous Robots* 39 (1) (2015) 101–121.
- [30] D. Morgan, G. P. Subramanian, S.-J. Chung, F. Y. Hadaegh, Swarm assignment and trajectory optimization using variable-swarm, distributed auction assignment and sequential convex programming, *The International Journal of Robotics Research* 35 (10) (2016) 1261–1285.
- [31] M. Szmuk, C. A. Pascucci, D. Dueri, B. Açıkmeşe, Convexification and real-time on-board optimization for agile quad-rotor maneuvering and obstacle avoidance, in: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 4862–4868. doi:10.1109/IR0S.2017.8206363.
- [32] J. Schulman, Y. Duan, J. Ho, A. Lee, I. Awwal, H. Bradlow, J. Pan, S. Patil, K. Goldberg, P. Abbeel, Motion planning with sequential convex optimization and convex collision checking, *The International Journal of Robotics Research* 33 (9) (2014) 1251–1270.
- [33] G. Misra, X. Bai, Optimal path planning for free-flying space manipulators via sequential convex programming, *Journal of Guidance, Control, and Dynamics* 40 (11) (2017) 3026–3033. doi:10.2514/1.G002487.
- [34] J. Nocedal, S. Wright, *Numerical optimization*, Springer Science & Business Media, 2006.
- [35] J. B. Lasserre, Global optimization with polynomials and the problem of moments, *SIAM Journal on Optimization* 11 (3) (2001) 796–817. doi:10.1137/S1052623400366802.
- [36] J. Nie, An exact jacobian sdp relaxation for polynomial optimization, *Mathematical Programming* 137 (1-2) (2013) 225–255.
- [37] K. Svanberg, The method of moving asymptotes—a new method for structural optimization, *International journal for numerical methods in engineering* 24 (2) (1987) 359–373.
- [38] C. Zillober, Global convergence of a nonlinear programming method using convex approximations, *Numerical Algorithms* 27 (3) (2001) 265–289. doi:10.1023/A:1011841821203.
- [39] Q. T. Dinh, M. Diehl, Local convergence of sequential convex programming for nonconvex optimization, in: *Recent Advances in Optimization and its Applications in Engineering*, Springer, 2010, pp. 93–102.
- [40] M. Szmuk, B. Açıkmeşe, Successive convexification for fuel-optimal powered landing with aerodynamic drag and non-convex constraints, in: *AIAA Guidance, Navigation, and Control Conference*, Vol. 378, 2016.
- [41] Y. Mao, D. Dueri, M. Szmuk, B. Açıkmeşe, Successive convexification of non-convex optimal control problems with state constraints, *IFAC-PapersOnLine* 50 (1) (2017) 4063 – 4069, 20th IFAC World Congress. doi:10.1016/j.ifacol.2017.08.789.
- [42] B. Alrifaee, J. Maczjewski, D. Abel, Sequential convex programming mpc for dynamic vehicle collision avoidance, in: *Control Technology and Applications (CCTA)*, 2017 IEEE Conference on, IEEE, 2017, pp. 2202–2207.

- [43] M. Szmuk, B. Acikmese, Successive Convexification for 6-DoF Mars Rocket Powered Landing with Free-Final-Time, American Institute of Aeronautics and Astronautics, 2018. doi:doi:10.2514/6.2018-0617.
- [44] Y. Mao, M. Szmuk, B. Açıkmeşe, Successive convexification: A super-linearly convergent algorithm for non-convex optimal control problems, Preprint Submitted to Automatica.
- [45] B. R. Marks, G. P. Wright, A general inner approximation algorithm for nonconvex mathematical programs, *Operations research* 26 (4) (1978) 681–683.
- [46] Y. Mao, D. Dueri, M. Szmuk, B. Açıkmeşe, Successive convexification of non-convex optimal control problems with state constraints, arXiv preprint arXiv:1701.00558.
- [47] A. R. Conn, N. I. Gould, P. L. Toint, *Trust region methods*, SIAM, 2000.
- [48] G. Di Pillo, Exact penalty methods, in: *Algorithms for Continuous Optimization*, Springer, 1994, pp. 209–253.
- [49] A. L. Yuille, A. Rangarajan, The concave-convex procedure, *Neural computation* 15 (4) (2003) 915–936.
- [50] T. Lipp, S. Boyd, Variations and extension of the convex–concave procedure, *Optimization and Engineering* 17 (2) (2016) 263–287.
- [51] H. Tuy, *Dc optimization: theory, methods and algorithms*, in: *Handbook of global optimization*, Springer, 1995, pp. 149–216.
- [52] P. D. Tao, et al., The dc (difference of convex functions) programming and dca revisited with dc models of real world nonconvex optimization problems, *Annals of operations research* 133 (1-4) (2005) 23–46.
- [53] P. Hartman, On functions representable as a difference of convex functions, *Pacific Journal of Mathematics* 9 (3) (1959) 707–713.
- [54] J.-B. Hiriart-Urruty, Generalized differentiability / duality and optimization for problems dealing with differences of convex functions, in: J. Ponsstein (Ed.), *Convexity and Duality in Optimization*, Springer Berlin Heidelberg, Berlin, Heidelberg, 1985, pp. 37–70.
- [55] A. A. Ahmadi, G. Hall, Dc decomposition of nonconvex polynomials with algebraic techniques, *Mathematical Programming*doi:10.1007/s10107-017-1144-5.
- [56] A. D. Aleksandrov, On the surfaces representable as difference of convex functions., *Sibirskie Elektronnye Matematicheskie Izvestiia* 9.
- [57] Q. T. Dinh, W. Michiels, S. Gros, M. Diehl, An inner convex approximation algorithm for bmi optimization and applications in control, in: *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, IEEE, 2012, pp. 3576–3581.
- [58] T. Wang, L. Vandendorpe, Successive convex approximation based methods for dynamic spectrum management, in: *Communications (ICC), 2012 IEEE International Conference on*, IEEE, 2012, pp. 4061–4065.
- [59] G. Scutari, F. Facchinei, L. Lampariello, P. Song, Distributed methods for constrained nonconvex multi-agent optimization-part I: theory, arXiv preprint arXiv:1410.4754.
- [60] Y. Nesterov, B. Polyak, Cubic regularization of newton method and its global performance, *Mathematical Programming* 108 (1) (2006) 177–205. doi:10.1007/s10107-006-0706-8.
- [61] X. Saint Raymond, *Elementary introduction to the theory of pseudodifferential operators*, Studies in Advanced Mathematics, CRC Press, 2000 Corporate Blvd., N.W., Boca Raton, FL, 33431, USA., 1991.
- [62] C. D. Meyer, *Matrix analysis and applied linear algebra*, Vol. 71, Siam, 2000.
- [63] C.-F. Cui, Y.-H. Dai, J. Nie, All real eigenvalues of symmetric tensors, *SIAM Journal on Matrix Analysis and Applications* 35 (4) (2014) 1582–1601.
- [64] R. A. Horn, C. R. Johnson, *Matrix Analysis.*, Cambridge University Press, 1990.
- [65] M. A. Patterson, A. V. Rao, Gpops-ii: A matlab software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming, *ACM Transactions on Mathematical Software (TOMS)* 41 (1) (2014) 1.
- [66] A. Wächter, T. L. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, *Mathematical Programming* 106 (1) (2005) 25–57. doi:10.1007/s10107-004-0559-y.
- [67] HSL, HSL. a collection of fortran codes for large scale scientific computation. <http://www.hsl.rl.ac.uk/>.
- [68] J. Virgili-Llop, A. Sharp, M. Romano, Reorientation of rigid spacecraft using onboard convex optimization, in: *AAS/AIAA Astrodynamics Specialist Conference*, 19 - 23 August 2018, Snowbird, Utah. AAS 18-422, 2018.