**Robotics: Science and Systems 2019**
**Freiburg im Breisgau, June 22-26, 2019**

# Trajectory Optimization on Manifolds: A Theoretically-Guaranteed Embedded Sequential Convex Programming Approach

Riccardo Bonalli, Andrew Bylard, Abhishek Cauligi, Thomas Lew, Marco Pavone

*Abstract*—**Sequential Convex Programming (SCP) has recently gained popularity as a tool for trajectory optimization due to its sound theoretical properties and practical performance. Yet, most SCP-based methods for trajectory optimization are restricted to Euclidean settings, which precludes their application to problem instances where one must reason about manifold-type constraints (that is, constraints, such as loop closure, which restrict the motion of a system to a subset of the ambient space). The aim of this paper is to fill this gap by extending SCP-based trajectory optimization methods to a manifold setting. The key insight is to leverage *geometric embeddings* to lift a manifold-constrained trajectory optimization problem into an equivalent problem defined over a space enjoying a Euclidean structure. This insight allows one to extend existing SCP methods to a manifold setting in a fairly natural way. In particular, we present a SCP algorithm for manifold problems with refined theoretical guarantees that resemble those derived for the Euclidean setting, and demonstrate its practical performance via numerical experiments.**

## I. Introduction

Trajectory optimization is a key problem in robotics, and it has thus been studied extensively through a variety of mathematical frameworks. Examples include sampling-based motion planning techniques [16, 18, 19, 21, 22], variational approaches such as CHOMP and STOMP [33, 17], sum-of-squares methods [27, 38], and sequential convex programming (SCP) techniques such as TrajOpt and GuSTO [6, 26, 28, 36]. Most of these methods, however, are restricted to Euclidean settings, which precludes their application (at least directly) to problem instances where one needs to reason about *manifold-type constraints*. For example, such constraints arise when the motion of a robotic system is forced to evolve on subsets of the ambient space (e.g., due to the presence of closed kinematic chains giving rise to loop closure constraints [15]), which are mathematically modeled as manifolds. Systems having such constraints include quadrotors [2, 29], robots with camera orientation constraints [37, 43], manipulator systems [20, 31] and robotic spacecraft [30, 41], to name a few. For such systems, trajectory optimization methods must ensure that the computed trajectories lie on the relevant manifolds, preventing the planning of infeasible motions. However, this is in general challenging, as manifold-type constraints are often defined only locally (i.e., through local charts) or as implicit constraints (i.e., constraints of the type $E(x) = 0$ where $E$ is a submersion and $x$ is the state vector). As a pedagogical example, consider a two-joint manipulator. Its motion is forced to evolve on the torus $\mathbb{T}^2$ (see Figure 2), which represents a two-dimensional submanifold of $\mathbb{R}^3$. Specifically, each joint variable ($\theta_i$, $i = 1, 2$) evolves on the unitary circle $S^1$, and thus the combined evolution is on the Cartesian product $S^1 \times S^1$, which is diffeomorphic to $\mathbb{T}^2$ (as characterized by implicit and nonlinear equality constraints).

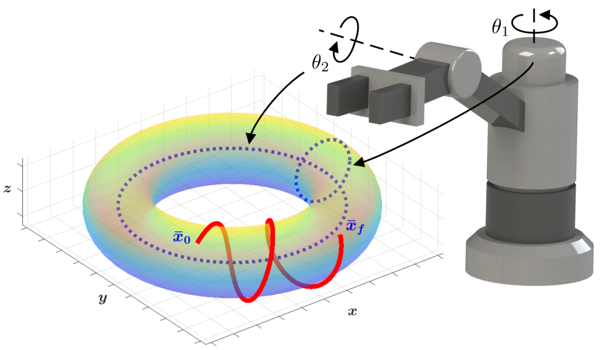

Fig. 1: The torus $\mathbb{T}^2$ is an embedded submanifold of $\mathbb{R}^3$. The red trajectory, starting from $\bar{x}_0$ and finishing at $\bar{x}_f$, is characterized by dynamics with torus-type constraints and can be seen as an embedded curve in $\mathbb{R}^3$ (Section II-B).

Despite the ubiquitous presence of manifold constraints in robotic applications, the set of trajectory optimization tools that handle such constraints is relatively limited. The most naïve technique consists of simply removing, without principled justification, all manifold-type constraints, and then solving a relaxed version of the original problem in the resulting Euclidean space. Since this approach cannot guarantee trajectory feasibility, one needs to resort to post-processing before trajectory execution, often using a heuristic correction step which may be unsuccessful. This has prompted the design of optimization approaches that explicitly account for the presence of manifold constraints, including sampling-based techniques leveraging local chart analysis, [39, 15], methods employing global chart-gluing procedures [43], and methods exploiting properties of specific types of manifolds (in particular, Lie groups), such as invariant metrics [42] and projection operators [34, 35]. These methods, while directly accounting for the presence of manifold constraints, do not in general enjoy theoretical guarantees, and they only consider a subset of the typical constraints arising in robotic applications (e.g., control or goal region constraints are generally not addressed).

R. Bonalli, A. Bylard, A. Cauligi, and M. Pavone are with the Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305. {rbonalli, bylard, acauligi, pavone} @stanford.edu.

T. Lew is with the Institute for Dynamic Systems and Control, ETH Zürich, Zürich, Switzerland. lewt@student.ethz.ch.

This work was supported in part by NASA under the Space Technology Research Program, NASA Space Technology Research Fellowship Grants NNX16AM78H and NNX15AP67H, Early Career Faculty Grant NNX12AQ43G and Early Stage Innovations Grant NNX16AD19G, and by KACST. T. Lew is partially supported by the Master's Thesis Grant of the Zeno Karl Schindler Foundation.

Building on the recent success of SCP-based techniques for trajectory optimization, the aim of this paper is to provide an SCP-based framework for trajectory optimization on manifolds that (1) enjoys theoretical guarantees in terms of convergence, (2) is general, in that it accounts for a vast class of constraints arising in robotic applications (possibly implicitly defined), and (3) provides effective and reliable practical performance. Specifically, SCP entails successively convexifying the cost function and constraints of a nonconvex optimal control problem, seeking a solution to the original problem through a sequence of convex problems. Its attractiveness is due to high computational speed [2, 36], broad applicability [26, 41], and (continuous-time) theoretical guarantees [6, 28]. Extending SCP-based methods, primarily developed for Euclidean settings, to manifold-constrained problems is, however, challenging. In particular, when dealing with manifolds, it is challenging to make linearizations (required by SCP schemes that operate on dynamics) well-posed [8]. The key technical idea of this paper is to leverage *geometric embeddings* – that is, mappings that allow one to recover manifolds as subsets of Euclidean spaces. Leveraging embeddings provides four main advantages. First, it allows one to lift a manifold-constrained problem into an equivalent problem defined over a space enjoying Euclidean structure, where linearizations can be easily computed. Second, embedded problems are often easier to address than their counterparts in local coordinates since, for example, linearity can be partially maintained. Third, embeddings provide a pathway to address implicitly-defined manifolds, as the equality constraints defining them are automatically satisfied in the lifted Euclidean space without the need for explicit enforcement. Fourth, and crucially, for dynamical systems evolving on Lie groups (as it is the case for virtually all robotic systems), there is always a "natural" embedding that can, at least in principle, be leveraged. Indeed, any mechanical system can always be identified with a subgroup $G \subseteq \mathbb{R}^{n\times n}$ of a Cartesian product of Lie groups of matrices. Then, the first-order equation that governs the dynamical evolution of the system is

$$\dot{v}(t) = \Gamma(v(t)) + \sum_{i=1}^{m} u_i(t) \ \mathrm{vlft}(F_i)(v(t)), \quad v \in TG, \qquad (1)$$

where $\Gamma$ is the geodesic spray induced by the kinetic energy and $\mathrm{vlft}(F_i)$ is the vertical lift of the generalized force $F_i$ [7]. This can be reinterpreted as a first-order, control-affine equation with a drift term on the space $G\times\mathfrak{g}$ via the identification $TG \cong G\times\mathfrak{g}$ [7], where $\mathfrak{g}$ denotes the Lie algebra of $G$. A natural embedding is then the inclusion $G\times\mathfrak{g} \subseteq \mathbb{R}^{2n^2}$ (we provide explicit examples in the rest of the paper).

*Statement of Contributions:* In this paper, we leverage geometric embeddings to extend SCP-based methods for trajectory optimization to manifold-constrained problems. Specifically, the contribution of this paper is fourfold. First, we introduce the notion of embedded SCP, a trajectory optimization method that exploits geometric embeddings to recast optimization on manifolds as a sequence of convex optimal control problems within Euclidean spaces. Importantly, a large number of trajectory optimization problems can be "naturally" (in the sense above) embedded in Euclidean spaces, which makes this step generally straightforward. Second, leveraging such a reformulation and extending recent results on theoretical guarantees for SCP-based methods in Euclidean spaces [6], we provide convergence guarantees for embedded SCP in the sense of the geometric Pontryagin Maximum Principle (PMP) [32], i.e., in the sense of convergence of both the solution and corresponding Lagrange multipliers to stationary points satisfying necessary conditions for optimality and complying with the structure of the manifold characterizing the problem. In particular, a key aspect of our theoretical analysis entails showing how one can avoid manifold-type constraints in the optimization process, and yet can still guarantee that the computed solution lies on the manifold – thus providing a computationally efficient pathway to deal with implicit manifold constraints. Third, by merging techniques from indirect optimal control and differential geometry, we extend the theoretical results to a large variety of settings, e.g., goal region constraints and pointwise state constraints arising in multitask scenarios. Fourth, again inspired by analogue results in the Euclidean setting [6], we harness the insights gained through our theoretical analysis to develop a convergence acceleration scheme for trajectory optimization on manifolds based on shooting methods [3].

One must note that while geometric embeddings allow a principled and systematic development of SCP methods for manifold-constrained problems, they come with two key drawbacks. First, embedding the manifold into an Euclidean space requires solving an optimization problem on a space having a higher dimension than the original manifold (albeit of simpler structure). Nevertheless, SCP-based algorithms scale rather well with problem dimensionality, and thus this drawback is offset by the simplification in the problem structure. Second, a straightforward and simple leveraging of embeddings is possible only if globally defined dynamical equations are at our disposal, i.e., it is easy to write

$$\dot{x} = f(x,u), \ x \in \mathbb{R}^N \text{ with the additional constraint } x \in M, \quad (2)$$

for some $N \in \mathbb{N}$, where $M$ is a $n$-dimensional manifold. Indeed, though recovering the expression above is always theoretically possible thanks to (1), it could be hard to practically describe complex dynamics as in (2) (for example, for second-order dynamics that are defined only by local coordinates). Still, most of the systems commonly used in robotics are in the form given by (2), which makes the proposed method applicable to most robotic systems.

*Structure of the Paper:* In Section II, we define the problem of trajectory optimization on manifolds and introduce the tool of embeddings that allows us to reformulate the problem in a standard Euclidean space on which we can proceed by linearization. In Section III, we introduce the embedding trajectory optimization algorithm and the primary theoretical contributions that are geometrically consistent with the structure of the manifold. Finally, Sections V and VI present experiments, conclusions and future extensions for this work.

*Notation:* We denote by $M$ an $n$-dimensional manifold with tangent bundle $TM$ and cotangent bundle $T^*M$ (both manifolds of dimension $2n$). The tangent space of $M$ at $x \in M$ will be denoted by $T_xM$. Moreover, we recall that a smooth vector field on $M$ is a mapping $f : M \to TM$ such that $f(x) \in T_xM$, for every $x \in M$. The interested reader is referred to [24] for related concepts in differential geometry.

## II. Problem Formulation and Sequential Convex Programming on Manifolds

We begin in Section II-A by formulating trajectory optimization for dynamical systems as an optimal control problem on manifolds. Then, in Section II-B, we introduce a procedure for appropriately embedding the original problem on manifolds into an Euclidean space. This allows us to exploit classical SCP frameworks for trajectory optimization in Euclidean spaces to solve the problem in Section II-C.

### A. Trajectory Optimization on Generic Manifolds

In this paper we consider a continuous-time formulation to ensure that the theoretical guarantees we derive are independent of the discretization scheme that is employed. A discussion about the impact of discretization schemes on the proposed methodology will be discussed in Section III-D.

Specifically, consider an initial point $\bar{x}_0 \in M$ and smooth mappings $g^i : M \to \mathbb{R}^{r_i}$, $i = 1, \ldots, \ell$, which are submersions at 0. Here, $g^i$ represent pointwise state constraints that are used to mathematically model multitask scenarios; in particular, $g^\ell$ represents goal region constraints. Without loss of generality, we require $\mathrm{dist}(\bar{x}_0, (g^\ell)^{-1}(0)) > 0$, where dist is a point-set distance evaluated w.r.t. some Riemannian metric on $M$. For times $0 < t_1 < \cdots < t_\ell$, we model the dynamical evolution of the system by the following drift control-affine system in $M$

$$\begin{cases} \dot{x}(t) = f(x(t), u(t)) = f_0(x(t)) + \displaystyle\sum_{j=1}^{m} u^j(t) f_j(x(t)) \\ x(0) = \bar{x}_0 \quad , \quad g^i(x(t_i)) = 0 \ , \ i = 1, \ldots, \ell, \quad t \in [0, t_\ell], \end{cases} \tag{3}$$

where $f_j : M \to TM$, $j = 0, \ldots, m$ are $C^1$ vector fields. The pointwise state constraints $g^i(x(t_i)) = 0$ are useful in multitask scenarios where one seeks to jointly optimize subtrajectories connecting different waypoints. We emphasize that, as previously mentioned, the dynamics of every mechanical system can be written as in Eq. (3) by substituting the manifold $M$ with its tangent bundle $TM$.

We pose trajectory optimization as an optimal control problem with penalized state constraints. Specifically, we define the Optimal Control Problem (**OCP**) as minimizing the cost

$$\begin{aligned} J(t_\ell, x, u) = \int_0^{t_\ell} f^0(x(t), u(t)) \, \mathrm{d}t = \\ \int_0^{t_\ell} \Big( \|u(t)\|_R^2 + u(t) \cdot f_u^0(x(t)) + g(x(t)) \Big) \, \mathrm{d}t \end{aligned} \tag{4}$$

under dynamics and pointwise constraints (3), among all control trajectories $u \in L^\infty([0, t_\ell], \mathbb{R}^m)$ satisfying $u(t) \in U$ almost everywhere in $[0, t_\ell]$, where the measurable set $U \subseteq \mathbb{R}^m$ represents control constraints. Here, $f_u^0 : M \to \mathbb{R}^m$, $g : M \to \mathbb{R}$ are $C^1$, $\| \cdot \|_R$ is the weighted norm defined by a constant positive-definite matrix $R \in \mathbb{R}^{m \times m}$, and times $t_i$, $i = 1, \ldots, \ell$ are fixed. We remark that hard enforcement of dynamical and intermediate/final goal set constraints is naturally imposed by (3). The function $g = g_a + \omega g_b$ accumulates contributions from a purely state-dependant cost $g_a$ and state constraint penalty function $g_b$ (e.g., stemming from collision-avoidance constraints), weighted by $\omega \geq 1$. Penalizing state constraints (e.g., collision avoidance) provides both theoretical and numerical benefits: it allows us to obtain theoretical guarantees in the sense of the Pontryagin Maximum Principle [32, 1, 10], necessary conditions for optimality that are stronger than standard Lagrange multiplier rules (see also Theorem 3 below), and it provides numerical flexibility by often allowing simple trajectories that violate constraints such as obstacle avoidance to be exploited for initialization. Indeed, given correct design of an SCP algorithm, we can still guarantee that returned solutions satisfy state constraints up to a user-defined tolerance.

A representative example (that will serve as running example) of a trajectory optimization problem evolving on manifolds is the minimum-energy optimal control of a spacecraft avoiding collisions in a microgravity environment, which can be stated as (under the previous formalism, $g$ provides state constraints):

$$(\mathbf{Ex}) \begin{cases} \min \displaystyle\int_0^{t_\ell} \Big( \|u_1(t)\|^2 + \|u_2(t)\|^2 + g(r(t), q(t)) \Big) \, \mathrm{d}t \\ \dot{r} = v \ , \ \dot{v} = \dfrac{u_1}{m} \ , \ \dot{q} = \dfrac{1}{2} \Omega(w) q \ , \ \dot{w} = J^{-1}(u_2 - w \times Jw) \\ (r, v, q, w)(0) = \bar{x}_0 \ , \ (r, v, q, w)(t_\ell) = \bar{x}_\ell \ , \ (u_1, u_2) \in U \\ (r, v, q, w) \in \mathbb{R}^6 \times S^3 \times \mathbb{R}^3 \ , \ S^3 = \{q \in \mathbb{R}^4 : \|q\| = 1\} \end{cases}$$

where $r$ is the position of the vehicle, $v$ its tangential velocity, $q$ its orientation (expressed via quaternions), $w$ its angular velocity ($\Omega(w)$ is the usual skew-symmetric matrix depending on $w$) and the manifold $S^3 \subseteq \mathbb{R}^4$ characterizes quaternions. Controls are represented by the thrust $u_1$ and the torque $u_2$. A naïve way to approach our running example (**Ex**) would entail removing, without principled justification, the constraint $q \in S^3$, and then solving the relaxed problem in the resulting Euclidean space – this could result, however, in computation of infeasible trajectories. Better justified approaches could exploit local charts or Lie group properties, as mentioned in Section I. However, in what follows, we demonstrate another method to tackle the implicit manifold constraint $q \in S^3$, that hinges on embeddings and provides a way to lift SCP methods to manifold-constrained problems.

### B. Embedding the Problem into the Euclidean Space

We would like to solve (**OCP**) via SCP, i.e., by an iterative procedure based on the linearization of all nonlinear mappings around the solution at the previous iteration. This requires us to compose a notion of linearized vector fields of $M$ around curves. In [40, 25, 8], the authors adapt such a definition on manifolds by recasting differential equations as algebraic equations of operators in $TM$. However, in many applications concerning dynamical systems, $M$ naturally appears as subset of the Euclidean space, in which case the most intuitive linearization is the one operating in the ambient Euclidean space. This insight motivates our approach: to recast (**OCP**) into an appropriate Euclidean space via geometric embeddings, i.e., mappings $e : M \to \mathbb{R}^N$ for $N \in \mathbb{N}$, and then to linearize in the ambient space.

Following the previous discussion, we assume that $M$ is a closed submanifold of $\mathbb{R}^N$, for some $N \in \mathbb{N}$. This means that we fix a particular embedding, which is given by the canonical inclusion $e : M \to \mathbb{R}^N : x \mapsto x$. This choice is made without loss of generality because, due to Whitney-type theorems [24], such a mapping always exists. Moreover, consistent with the previous motivating discussion, we assume that the mappings defining (**OCP**) naturally extend to $\mathbb{R}^N$, i.e., there exist $C^1$ vector fields $F_j : \mathbb{R}^N \to T\mathbb{R}^N \cong \mathbb{R}^N$, $j = 1,\ldots,m$ (with an abuse of notation, see [24]), and $C^1$ functions $F_u^0 : \mathbb{R}^N \to \mathbb{R}$, $G : \mathbb{R}^N \to \mathbb{R}$ and $G^i : \mathbb{R}^N \to \mathbb{R}$, $i = 1,\ldots,\ell$, that are smooth submersions at 0, such that $F_j|_M = f_j$, $G^i|_M = g^i$, $F_u^0|_M = f_u^0$ and $G|_M = g$. This setup allows us to transform the dynamics (3) into the following drift control-affine system in $\mathbb{R}^N$:

$$\begin{cases} \dot{x}(t) = F(x(t),u(t)) = F_0(x(t)) + \displaystyle\sum_{j=1}^{m} u^j(t) F_j(x(t)) \\ x(0) = \bar{x}_0 \in M \subseteq \mathbb{R}^N \quad , \quad G^i(x(t_i)) = 0 \ , \ i = 1,\ldots,\ell. \end{cases} \tag{5}$$

Therefore, (**OCP**) can be embedded in $\mathbb{R}^N$ by considering the following Embedded Optimal Control Problem (**EOCP**), which consists of minimizing the integral cost

$$\begin{aligned} J(t_\ell, x, u) = \int_0^{t_\ell} F^0(x(t),u(t))\,\mathrm{d}t = \\ \int_0^{t_\ell} \Big( \|u(t)\|_R^2 + u(t)\cdot F_u^0(x(t)) + G(x(t)) \Big)\,\mathrm{d}t \end{aligned} \tag{6}$$

under dynamics (5), among all control trajectories $u \in L^\infty([0,t_\ell],\mathbb{R}^m)$ satisfying $u(t) \in U$ almost everywhere in $[0,t_\ell]$. At this step, it is worth noting that this embedding approach is justified only if solving (**EOCP**) is equivalent to solving (**OCP**). Fortunately, this is actually the case: every couple $(x,u)$ is optimal for (**OCP**) if and only if it is optimal for the embedded problem (**EOCP**). The validity of the whole scheme hinges on this crucial remark, which is summarized in the statement below,

**Lemma 1** (Embedding Lemma)**.** *A tuple $(x,u)$ is optimal for (**EOCP**) if and only if it is optimal for (**OCP**).*

*Proof:* The proof makes use of standard tools in differential geometry, whose definitions can be found in [24] and are omitted here due to space limitations. We retrace the main steps of the proof. Denote by $e : M \to \mathbb{R}^N$ the canonical inclusion. It follows that $F$ is $e$-correlated to $f$, which implies their flows satisfy $\Phi_F(t,x) = \Phi_F(t,e(x)) = e(\Phi_f(t,x)) = \Phi_f(t,x)$ for every $(t,x) \in \mathbb{R}\times M$ for which the flow $\Phi_f$ is defined. Since $\bar{x}_0 \in M$, we obtain that $(x,u)$ satisfies the dynamics in (5) if and only if it satisfies the embedded dynamics in (3). From this, the optimality of $(x,u)$ follows from the similarity between (4) and (6). ■

**Remark 1.** *Crucially, from Lemma 1, we see that the satisfaction of implicit manifold-type constraints for (**EOCP**) is induced by hard enforcement of dynamical constraints. Therefore, any numerical strategy used to solve (**EOCP**) must provide hard enforcement of dynamics – otherwise, the solution trajectory is not guaranteed to lie on the manifold!*

Let us show how this embedding framework applies to our running example (**Ex**). It is sufficient to note that the only components of the dynamics evolving on a manifold are given by the mapping

$$S^3 \times \mathbb{R}^3 \subseteq \mathbb{R}^7 \to T(S^3 \times \mathbb{R}^3) : (q,w) \mapsto \Big( \frac{1}{2}\Omega(w)q, J^{-1}(u_2 - w \times Jw) \Big),$$

and that this mapping is also defined when $(q,w) \in \mathbb{R}^7$. In other words, the original dynamics are equivalent to $\left(\frac{1}{2}\Omega(w)q, J^{-1}(u_2 - w \times Jw)\right)$ restricted to the subset $S^3 \times \mathbb{R}^3$. Therefore, the embedded dynamics related to this mapping are exactly the same but extended on $\mathbb{R}^7$, which shows that the embedded version of our example problem coincides with the original (**OCP**). Luckily, for many robotics applications which include trajectory optimization on manifolds, (**EOCP**) is equivalent to (**OCP**), which is also the case when formulation (1) is met. This is among the main motivations for developing such an embedded framework (see also our discussion at the end of Section II-A).

### *C. Reformulating Problem (**OCP**) via SCP in Euclidean Space*

Given that (**EOCP**) evolves in the Euclidean space, we may solve it using SCP. Below, we describe a particular SCP formulation that enjoys geometrically consistent theoretical convergence guarantees.

Under the assumption that $U$ is convex, we iteratively linearize the nonlinear contributions of (**EOCP**) around local solutions, thus recursively defining a sequence of simplified problems. Specifically, at the end of iteration $k$, assume we have some continuous curves $x_k : [0,t_\ell] \to \mathbb{R}^N$ and $u_k : [0,t_\ell] \to \mathbb{R}^m$, continuously extended in the interval $(0,+\infty)$. Then, at iteration $k+1$, the Linearized Embedded Optimal Control Problem $(\textbf{LEOCP})_{k+1}$ consists of minimizing the new cost

$$\begin{aligned} J_{k+1}(t_\ell, x, u) = \int_0^{t_\ell} F_{k+1}^0(t, x(t), u(t))\,\mathrm{d}t = \\ \int_0^{t_\ell} \Big( \|u(t)\|_R^2 + h_k(\|x(t) - x_k(t)\|^2 - \Delta_k) \Big)\,\mathrm{d}t \\ + \int_0^{t_\ell} u(t) \cdot \left( F_u^0(x_k(t)) + \frac{\partial F_u^0}{\partial x}(x_k(t)) \cdot (x(t) - x_k(t)) \right) \mathrm{d}t \\ + \int_0^{t_\ell} \left( G_k(x_k(t)) + \frac{\partial G_k}{\partial x}(x_k(t)) \cdot (x(t) - x_k(t)) \right) \mathrm{d}t \end{aligned} \tag{7}$$

where, consistent with the notation of Section II-B, $G_k = G_a + \omega_k G_b$ and $h_k(s)$ is any smooth approximation of $\max\{0,s\}$ [24, Chapter 10]. Function $h_k$ provides trust-region guarantees on the *updates* in the state trajectories and constraints via the bounds $0 \le \Delta_k \le \Delta_0$ and weights $1 \le \omega_0 \le \omega_k \le \omega_{\max}$. The dynamical constraint for $(\textbf{LEOCP})_{k+1}$ is

$$\begin{cases} \dot{x}(t) = F_{k+1}(t, x(t), u(t)) = \\ \left( F_0(x_k(t)) + \displaystyle\sum_{j=1}^{m} u^j(t) F_j(x_k(t)) \right) \\ + \left( \dfrac{\partial F_0}{\partial x}(x_k(t)) + \displaystyle\sum_{j=1}^{m} u_k^j(t) \dfrac{\partial F_j}{\partial x}(x_k(t)) \right) \cdot (x(t) - x_k(t)) \\ x(0) = \bar{x}_0 \in M \quad , \quad G_{k+1}^i(x(t_i)) = G^i(x_k(t_i)) + \\ \qquad \dfrac{\partial G^i}{\partial x}(x_k(t_i)) \cdot (x(t_i) - x_k(t_i)) = 0 \ , \ i = 1,\ldots,\ell, \end{cases} \tag{8}$$

obtained from the linearized expansion of all nonlinear mappings. We minimize among all controls $u \in L^\infty([0,t_\ell],\mathbb{R}^m)$ satisfying $u(t) \in U$ almost everywhere in $[0,t_\ell]$. Inductively, the curves $x_{k+1} : [0,t_\ell] \to \mathbb{R}^N$ and $u_{k+1} : [0,t_\ell] \to \mathbb{R}^m$ are defined as the optimal solution for problem $(\textbf{LEOCP})_{k+1}$, continuously extended in the interval $(0, +\infty)$. Ideally, SCP algorithms may vary $\Delta_k$ and $\omega_k$ at each iteration to smoothen the process towards convergence, for example as in [6].

A convexified formulation similar to (7)-(8) has already been introduced in [6]. However, we stress the fact that this new formulation deals with the presence of the manifold $M$ and of pointwise state constraints. The introduction of these two new features necessitates a considerable revision of the proof of theoretical guarantees (see the extended version [5]).

The sequence of problems (**LEOCP**)$_k$ is well-posed if, for each iteration $k \geq 1$, an optimal solution for (**LEOCP**)$_k$ exists. For this, we consider the following assumptions:

($A_1$) The set $U$ is compact and convex. Moreover, the differentials of mappings $G^i$, $i = 1, \dots, \ell$, are of full rank.

($A_2$) Mappings $f^0$, $g$, vector fields $f_j$, $j = 0, \dots, m$ and their differentials have compact supports (and do $F^0$, $G$, $F_j$).

($A_3$) At every iteration $k \geq 1$, problem (**LEOCP**)$_k$ is feasible.

Under these assumptions, classical existence Filippov-type arguments [12, 23] (applied to the reduced form of (**LEOCP**)$_k$, see also the extended version [5]) show that, at each iteration $k \geq 1$, the problem (**LEOCP**)$_k$ has at least one optimal solution. We remark that similar assumptions have been considered in [6]; in the present contribution, ($A_1$)-($A_3$) gather the assumptions in [6] and appropriately adapt them to the context of manifolds and pointwise state constraints. Comments on their validity for very general trajectory optimization problems are easily adapted from [6, Section II.B].

Coming back to our running example problem (**Ex**), since we have already proved that the embedded problem coincides with (**Ex**), the linearization technique above applies directly to (**Ex**) without any additional step. This is particularly useful and happens every time the embedded problem is equivalent to the original one, which is common in trajectory optimization as highlighted previously. We remark that in nearly all scenarios having natural control constraints $U$, Assumptions ($A_1$)-($A_3$) are easily satisfied by (**Ex**).

## III. Algorithm Overview and Theoretical Guarantees

In Section III-A, we detail a general algorithm for the solution of (**OCP**) which combines SCP-based procedures with the embedded framework defined previously. Its convergence guarantees, in the sense of the Pontryagin Maximum Principle [32], are studied in Section III-B to III-D, where we show that these respect the original structure of the manifold in (**OCP**), despite solving a sequence of linearized versions of the embedded problem. Notably, this procedure allows one to solve (**OCP**) on manifolds defined implicitly via nonlinear equalities *without explicit representation*.

### A. SCP-based Trajectory Optimization on Manifolds

The first two steps in the algorithm above consist of transforming (**OCP**) into the optimal control problem (**EOCP**) on the Euclidean space via embedding procedures and successively linearizing it as detailed in Section II-B. In the third step, one finally applies some SCP scheme on Euclidean spaces. It is important to remark that E-SCP provides the user with the freedom to choose any sequential convex procedure to solve the sequence of problems (**LEOCP**)$_k$. However, we show in Section III-B that specific choices of SCP (e.g. using hard enforcement of dynamical constraints) provide theoretical guarantees for E-SCP which are also consistent with the presence of the manifold within the original problem (**OCP**).

**Algorithm 1** Embedded SCP (E-SCP)

```
Input: Trajectory $x_0$ and control $u_0$ defined in $(0, \infty)$.
Output: Solution for (LEOCP)$_k$ at iteration $k$.
Data: Parameters for the used SCP procedure.
Transform (OCP) into problem (EOCP) in the Euclidean space, as in Section II-B;
Linearize (EOCP) by defining a sequence of convex problems (LEOCP)$_k$ (an example of such linearization is given in Section II-C);
Select a SCP procedure on Euclidean spaces to solve the sequence of problems (LEOCP)$_k$ for $(x_k, u_k)$;
return $(x_k, u_k)$ at the last iteration.
```

Problem (**LEOCP**)$_1$ is linearized around an initial curve tuple $(x_0, u_0)$, where these initialization curves should be as close as possible to a feasible or even optimal curve for (**LEOCP**)$_1$, although we do not require that $(x_0, u_0)$ is feasible for the embedded problem (**EOCP**). This allows one to initialize E-SCP with simple, even infeasible, guesses for solutions of (**EOCP**), such as a straight line in the manifold, as detailed in [6, Section III.A].

### B. Necessary Conditions for Optimality

Although the strategy for solving problems (**LEOCP**)$_k$ in E-SCP is up to the user, we show that specific choices of solvers allow one to recover important theoretical guarantees for the convergence of E-SCP to critical points for the original problem (**OCP**) on manifolds. Specifically, we can show the convergence of E-SCP towards a trajectory satisfying first-order necessary conditions for optimality under the Pontryagin Maximum Principle [32] when the sequence of problems (**LEOCP**)$_k$ is chosen as in Section II-C. However, we must adapt the proof for the classical Euclidean setting to take into account the presence of both manifold and pointwise state constraints. This will be done by leveraging fundamental results from differential geometry and optimal control, i.e., Hamiltonian systems and the Pontryagin Maximum Principle with pointwise state constraints. For self-containtment, we summarize some of these results in the following discussion (see, e.g., [24, 1, 10] for an extended treatment).

For consistency with the existing presentation of these results, denote $f^0$ as a time-varying function, i.e., $f^0 = f^0(t, x, u)$, and fix some measurable control time-series $u : \mathbb{R} \to \mathbb{R}^m$. Thanks to Assumptions ($A_1$), ($A_2$), for every $t_0 \in \mathbb{R}$, the trajectories arising for the augmented system

$$\begin{cases} \dot{x}^0(t) = f^0(t, x(t), u(t)), \quad \dot{x}(t) = f(t, x(t), u(t)) \\ (x^0, x)(t_0) = (0, y_0), \quad (x^0, x)(t) \in \mathbb{R} \times M \end{cases} \tag{9}$$

exist in $\mathbb{R}$ for every $y_0 \in M$. Notice that $x^0(t)$ is simply the accumulated cost at time $t$. Therefore, the flow of system (9) is defined for every $(t_0, y_0) \in \mathbb{R} \times M$ and we denote it by $\exp_{(f^0_u, f_u)}(\cdot; \cdot, \cdot) : \mathbb{R}^2 \times M \to \mathbb{R} \times M$, such that $\exp_{(f^0_u, f_u)}(\cdot; t_0, y_0)$ represents the trajectory of (9) starting from $(0, y_0)$ at $t_0$.

By standard identifications, we denote $(p^0, p) \in T^*(\mathbb{R} \times M)$ and we define the Hamiltonian function related to (9) as

$$H : \mathbb{R} \times T^*(\mathbb{R} \times M) \times \mathbb{R}^m \to \mathbb{R}$$
$$(t, p^0, p, u) \mapsto \langle p, f(t, \pi(p), u) \rangle + p^0 f^0(t, \pi(p), u) \quad (10)$$

where $\pi : T^*M \to M$ is the canonical projection and $\langle \cdot, \cdot \rangle$ denotes the duality in $T^*M$. As a classical result on Hamiltonian systems [1, 24], for every $(t, u) \in \mathbb{R}^{m+1}$, one can uniquely associate to (10) the so-called Hamiltonian vector field $\vec{H}(t, \cdot, \cdot, u) : T^*(\mathbb{R} \times M) \to T(T^*(\mathbb{R} \times M))$ by the rule $\sigma_{(p^0,p)}(\cdot, \vec{H}(t, \cdot, \cdot, u)) = d_{(p^0,p)} H(t, \cdot, \cdot, u)$, $\sigma$ being the canonical symplectic form of $T^*(\mathbb{R} \times M)$ (see, e.g., [1]). Combining the classical geometric Pontryagin Maximum Principle [1] with the reduction scheme for pointwise state constraints developed in [10] yields the following extended geometric Pontryagin Maximum Principle (see extended version [5] for a proof sketch).

**Theorem 2** (Geometric Pontryagin Maximum Principle with Pointwise State Constraints)**.** *Let $x$ be an optimal trajectory for (**OCP**), associated with the control $u$ in $[0, t_\ell]$ and with fixed interior and final times $t_i$, $i = 1, \ldots, \ell$. There exists a nonpositive constant scalar $p^0$ and a piecewise absolutely continuous function $p : [0, t_\ell] \to T^*M$, called the adjoint vector, satisfying $p(t) \in T^*_{x(t)} M$, with $(p^0, p) \neq 0$, such that, almost everywhere in $[0, t_\ell]$, the following relations hold:*

- ***Adjoint Equations***

$$\frac{d(p^0, p)}{dt}(t) = \vec{H}(t, p^0, p(t), u(t)) \quad (11)$$

- ***Maximality Condition***

$$H(t, p^0, p(t), u(t)) = \max_{u \in U} H(t, p^0, p(t), u) \quad (12)$$

- ***Transversality Conditions***
  *For each $i = 1, \ldots, \ell - 1$, the adjoint vector satisfies*

$$p(t_i^-) - p(t_i^+) \perp \ker \frac{\partial g^i}{\partial x}(x(t_i)) \ , \ p(t_\ell) \perp \ker \frac{\partial g^\ell}{\partial x}(x(t_\ell)). \quad (13)$$

*The tuple $(p^0, p, u)$ is a (Pontryagin) extremal. We say that $(p^0, p, u)$ is normal if $p^0 \neq 0$, and abnormal otherwise.*

It is important to recall that the statement of optimality in Theorem 2 supersedes the classical Euclidean case as it additionally addresses pointwise state constraints (which consequently imply discontinuity of the adjoint vector), in context of *nonlinear* manifolds. These features considerably complexify the proof (cf. extended version [5]).

### C. Convergence with Geometric Consistency

The convergence of E-SCP can be inferred by leveraging one further commonly adopted regularity assumption concerning optimal controls:

$(A_4)$ At every iteration $k \geq 1$, the optimal control $u_k$ of $(\textbf{LEOCP})_k$ is piecewise continuous in every subinterval $[0, t_1]$ and $[t_{i-1}, t_i]$ for $i = 2, \ldots, \ell$.

The key convergence result is stated next:

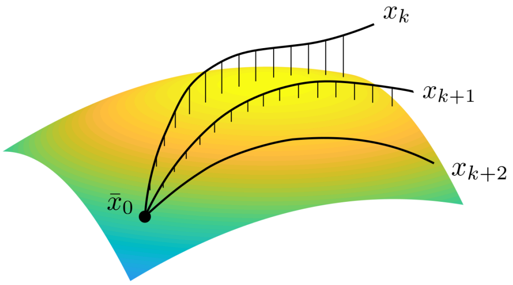

Fig. 2: (**EOCP**) uses hard enforcement of dynamical constraints to guarantee convergence to trajectories which lie on the manifold of the system, without explicitly enforcing or penalizing manifold constraints.

**Theorem 3** (Convergence Theorem)**.** *Suppose that $(A_1)$-$(A_4)$ hold. Given any sequence of trust region radii and weights $((\Delta_k, \omega_k))_{k \in \mathbb{N}} \subseteq [0, \Delta_0] \times [\omega_0, \omega_{\max}]$, let $((x_k, u_k))_{k \in \mathbb{N}}$ be any sequence such that, for every $k \geq 1$, $(x_k, u_k)$ is optimal for $(\textbf{LEOCP})_k$ in $[0, t_\ell]$, where problems $(\textbf{LEOCP})_k$ are built as detailed in Section II-C. Up to some subsequence:*

- *$x_k \to \tilde{x} \in C^0([0, t_\ell], M)$, for the strong topology of $C^0$, and*
- *$u_k \to \tilde{u} \in L^\infty([0, t_\ell], U)$, for the weak topology of $L^2$,*

*where $(\tilde{x}, \tilde{u})$ is feasible for the original problem (**OCP**). Moreover, there exists a nonpositive constant scalar $\tilde{p}^0$ and a piecewise absolutely continuous function $\tilde{\gamma} : [0, t_\ell] \to \mathbb{R}^N$, with $(\tilde{p}^0, \tilde{\gamma}) \neq 0$, such that the tuple $((\tilde{p}^0, \tilde{p}) = \mathrm{Pr}(\tilde{p}^0, \tilde{x}, \tilde{\gamma}), \tilde{u})$, where* Pr *is the orthogonal projection of $T^*\mathbb{R}^{N+1}|_{\mathbb{R} \times M}$ onto $T^*(\mathbb{R} \times M)$ (cf. extended version [5]), represents a geometric Pontryagin extremal (in the sense of Theorem 2) for the original problem (**OCP**) on $M$. In particular, as $k$ tends to infinity, up to some subsequence:*

- *$p^0_k \to \tilde{p}^0$, and*
- *$\gamma_k|_{[t_i, t_{i+1})} \to \tilde{\gamma}|_{[t_i, t_{i+1})} \in C^0([t_i, t_{i+1}), \mathbb{R}^N)$, $i = 1, \ldots, \ell - 1$, and $\gamma_k|_{[t_{\ell-1}, t_\ell]} \to \tilde{\gamma}|_{[t_{\ell-1}, t_\ell]} \in C^0([t_{\ell-1}, t_\ell], \mathbb{R}^N)$ for the strong topology of $C^0$,*

*where $(p^0_k, x_k, \gamma_k, u_k)$ is a Pontryagin extremal of $(\textbf{LEOCP})_k$.*

Due to space limitations, we present the proof of Theorem 3 in the extended version [5]. In short, Theorem 3 asserts that there exists a sequence of solutions for problems $(\textbf{LEOCP})_k$ that converges (under appropriate topologies) to a critical point for (**OCP**), in the (strong) sense of Theorem 2. Importantly, the limiting trajectory lies on $M$, despite solving the linearized, embedded versions $(\textbf{LEOCP})_k$ (see Figure 2). The termination properties of Algorithm E-SCP are stated next:

**Corollary 1** (E-SCP Termination)**.** *Assume (i) $(A_1)$-$(A_4)$ hold, (ii) problems $(\textbf{LEOCP})_k$ are built as detailed in Section II-C, (iii) the SCP procedure adopted for E-SCP (line **6** of Algorithm 1) enforces hard dynamical constraints (as opposed to a penalized implementation – see Remark 1), (iv) $\Delta_k \in [0, \Delta_0]$ for all $k$ and $\omega_k < \omega_{\max}$ are chosen such that for every iteration $k \geq 1$, an output $(x_k, u_k)$ is always provided, and (v) the algorithm is terminated if $\omega_k > \omega_{\max}$. Then, in solving (**OCP**) by E-SCP only three mutually exclusive situations arise:*

1) *There exists an iteration $k \geq 1$ for which $\omega_k > \omega_{\max}$. Then, E-SCP terminates, providing a solution for $(\textbf{LEOCP})_k$ satisfying only soft state constraints.*
2) *There exists an iteration $k \geq 0$ for which $(x_{k+1}, u_{k+1}) = (x_k, u_k)$. Then, E-SCP terminates, providing a stationary point, in the sense of the Pontryagin Maximum Principle, for the original problem (**OCP**).*

3) *We have $(x_{k+1}, u_{k+1}) \neq (x_k, u_k)$, for every iteration $k \geq 0$. Then, E-SCP builds a sequence of optimal solutions for $(\textbf{LEOCP})_k$ that has a subsequence converging (with respect to appropriate topologies) to a stationary point, in the sense of the Pontryagin Maximum Principle, for the original problem (**OCP**).*

*Proof:* The assumptions on the internal SCP procedure for E-SCP imply that only these three cases may happen and that they are mutually exclusive. Moreover, case 3) is a direct consequence of Theorem 3 and, for case 2), it is sufficient to apply Theorem 3 to the sequence of solutions

$$\Big((x_0, u_0), (x_1, u_1), \dots, (x_{k-1}, u_{k-1}), (x_k, u_k), (x_k, u_k), (x_k, u_k), \dots\Big)$$

because it clearly converges to $(\tilde{x}, \tilde{u}) = (x_k, u_k)$. ■

In case 1), SCP fails because no feasible strategies can be computed, which also occurs in other state-of-the-art trajectory optimization solvers such as TrajOpt [36]. Since the convergence of numerical methods usually leverages a termination threshold, case 3) is the most common outcome. Fortunately, Theorem 3 ensures that we are converging to a stationary point that satisfies strong necessary conditions for optimality in the sense of Theorem 2 and manifold-type constraints.

### *D. Geometric Consistency and Discrete-Time Convergence*

Despite the fact that E-SCP entails solving a sequence of linearized problems for the embedded reformulation (**EOCP**), i.e., without explicit representation of the manifold, Lemma 1 and Corollary 1 ensure that the numerical solution converges to trajectories that satisfy the manifold constraints and also ensure that the limiting solution satisfies strong first-order necessary conditions for optimality that respect the geometric structure of the original manifold.

The additional advantage of working within a continuous-time setting is that the validity of the theoretical guarantees is independent of the time-discretization scheme used to solve the ODEs. For instance, choosing schemes such as Euler or Simpson's rule lead to well well-posed convex optimization problems when the dynamics are linearized within each SCP step. With a sufficiently small time-step, one can ensure that the discrete solution provided by SCP stays close to the solution of the continuous-time problem (**EOCP**). Finally, Corollary 1 further implies that this discretized solution stays close to the solution for the original continuous-time (**OCP**).

## IV. Convergence Acceleration via Differential Shooting Method

An important result provided by Theorem 3 is the convergence of Pontryagin extremals related to the sequence of solutions for problem $(\textbf{LEOCP})_k$ in the Euclidean space towards a Pontryagin extremal related to the solution for (**OCP**) in the manifold, found by E-SCP. As a consequence, we can extend the acceleration procedure proposed in [6] to the manifold and pointwise state constraint case, i.e., warm-starting shooting methods [3] with E-SCP.

The key idea is that since the convergence of the adjoint vectors is provided in $\mathbb{R}^N$, one can leverage them to re-state a shooting method on the manifold within $\mathbb{R}^N$. However, unlike the framework proposed in [6], the main difficulty concerns the pointwise state constraints that introduce discontinuities for the multipliers. Fortunately, we can still use this hybrid method by leveraging the knowledge of adjoint vectors at intermediate times. Assuming SCP is converging, the Lagrange multipliers $\lambda^i_k$ related to the pointwise condition $G^i_k(x(t_i)) = 0$ for the finite dimensional discretization of problems $(\textbf{LEOCP})_k$ approximate the values $\gamma_k(t_i)$ of the adjoint vectors related to the continuous-time $(\textbf{LEOCP})_k$ (see [13] and the problem reduction provided in the extended version [5]). Then, up to some subsequence, for every small $\delta > 0$, there exists an iteration $k_\delta \geq 1$ for which, for every iteration $k \geq k_\delta$, one has $\|\tilde{\gamma}(t_i) - \lambda^i_k\| < \delta$, $i = 1, \dots, \ell$, where $\tilde{\gamma}$ is an adjoint vector related to the solution of (**OCP**) found by SCP (see Theorem 3). This means that, starting from some iteration $k \geq k_\delta$, we can run a shooting method to solve (**OCP**), initializing using $\lambda^i_k$, $i = 1, \dots, \ell$. At each iteration of SCP, we use the values $\lambda^i_k$ provided by the solver to initialize the shooting method until convergence is achieved. This provides a theoretically guaranteed method to accelerate convergence for SCP towards a more accurate solution.

## V. Numerical Experiments and Discussion

In this section, we provide implementation details and examples to demonstrate various facets of our approach. We focus on three important aspects: (1) providing comparisons between E-SCP and standard SCP on Euclidean spaces, (2) providing comparisons between E-SCP and state-of-the-art algorithms for trajectory optimization, and (3) analyzing convergence acceleration for E-SCP via shooting methods.

Simulations are provided by considering two problems: our running example (**Ex**), defined in Section II-A, and a trajectory optimization problem for a 7 degree-of-freedom manipulator in a cluttered environment. Consistent with our embedding framework, rather than describing the manipulator via joint angle variables (i.e., local variables), the states are characterized by tuples $(x_i, y_i) \in \mathbb{S}^1$, $i = 1, \dots, 7$ (one for each joint), so that the system evolves in the 7-dimensional torus, i.e., $(x_1, y_1, \dots, x_7, y_7) \in \mathbb{T}^7$, which is naturally embedded in $\mathbb{R}^{14}$. The dynamics are given by the kinematic equations of a manipulator, that is, each joint $i$ satisfies:

$$f_i : \mathbb{T}^1 \times \mathbb{R} \to T\mathbb{T}^1 : (x_i, y_i, u_i) \mapsto (-y_i u_i, x_i u_i) \qquad (14)$$

where each $u_i \in \mathbb{R}$ is a control variable. As for our running example (**Ex**), (14) naturally represents a dynamical system in $\mathbb{R}^2$, so that, problem (**EOCP**) coincides with (**OCP**).

The examples and algorithms presented in this work were implemented in the Julia programming language [4] using the GuSTO.jl package located at `https://github.com/StanfordASL/GuSTO.jl`, with optimization problems solved using Gurobi [14]. We chose GuSTO [6] as the SCP procedure for E-SCP (line **4** of Algorithm 1), so that Corollary 1 held. For each compared SCP method, the continuous-time optimal control problem was discretized using a trapezoidal approximation of the dynamics, assuming a zero-order hold for the control, and the discrete-time cost considered for each problem was the energy $\sum_{k=1}^{d-1} ||u_k||_2^2 \Delta t$, where $d$ is the number of discretization points for the trajectory. Additionally, obstacle avoidance constraints served as our non-convex state constraints. For each set of simulations presented, we report results for 50 experiments with different start and goal configurations. A SCP trial is marked as successful if the algorithm converged and the resulting solution was collision-free. We used the Bullet Physics engine to calculate signed distances for obstacle avoidance constraints or penalties [9, 11]. For each experiment, the variables were initialized using straight-line initializations on the manifold.

### *A. Comparison with State-of-the-Art*

In this section, we demonstrate the benefits obtained when using E-SCP to solve trajectory optimization problems on manifolds, rather than standard SCP approaches. More specifically, the main advantage of E-SCP is that, when considering hard enforcement of dynamical constraints, the limiting numerical solution is guaranteed to lie on the manifold, even if such a constraint is not explicitly enforced. This approach is in contrast to enforcing the nonlinear manifold-type equality constraints that appear when using standard Euclidean-based SCP approaches. Since added equality constraints increase the complexity of the problem and may adversely affect efficiency, removing manifold-type constraints in SCP provides greater flexibility in solving the sequential problems, given that these constraints are implicitly satisfied.

For this comparison, we considered the 7-DoF manipulator using 120 discretization points over a trajectory time of 30 seconds [22]. The achieved results are shown in Table I. Here, comparisons are given between E-SCP with GuSTO as the internal solver, TrajOpt [36] without any enforced manifold-type constraints, and versions of GuSTO and TrajOpt where the manifold-type constraints are penalized (denoted by SCP and TrajOpt-P, respectively, in Table I).

E-SCP has the best performance in reduction of dynamical constraint error (even if negligible). As can be expected, the methods that explicitly penalize the presence of the manifold achieve the highest precision for manifold-constraint satisfaction (even if also negligible). However, a keen analysis of Table I shows that this comes with the tradeoff that the additional state-constraint penalties produce a greater tendency to fall into high-cost local minima, since the penalization affects the way the internal convex optimization algorithm reduces the cost. Consequently, the resulting true cost of the non-penalizing algorithms is on average lower than their penalizing counterparts. In other words, E-SCP provides better optimal solutions than state-of-the-art penalization approaches.

In addition, we note that although unit-norm constraints like $x \in \mathbb{T}^n$ can be easily formulated as penalty expressions and satisfied through penalization, more complex manifold constraints (e.g., those associated with SO(3), closed-kinematic chains, etc.) are more difficult to formulate and satisfy in this way, and would be better handled using E-SCP. Indeed, from Theorem 3 we see that manifold constraint error in E-SCP scales with the dynamical constraint error, which in turn depends only on the discretization scheme. Thus, we expect the negligible manifold constraint error for E-SCP in the previous example to carry over to more complex manifolds.

| | E-SCP | SCP | TrajOpt | TrajOpt-P |
|---|---|---|---|---|
| Dynamical constraint error | $1.9{\times}10^{-3}$ | $5.8{\times}10^{-3}$ | $8.5{\times}10^{-3}$ | $4.8{\times}10^{-3}$ |
| $\mathbb{T}^7$ manifold constraint error | $3.9{\times}10^{-2}$ | $3.6{\times}10^{-6}$ | $1.6{\times}10^{-2}$ | $3.1{\times}10^{-6}$ |
| True cost | 0.105 | 1.0 | 0.215 | 0.792 |
| Computation time | 0.466 | 0.129 | 0.321 | 1.0 |

TABLE I: Averaged results of manipulator arm experiments, with normalized cost, computation times and non-normalized values for constraint errors.

### *B. Convergence Acceleration via Shooting Methods*

In this section, we provide numerical simulations that highlight convergence benefits that are obtained when E-SCP is combined with shooting methods. In particular, we consider trajectory optimization of a spacecraft having the dynamics and embedding given in (**Ex**) and navigating through a highly cluttered environment, using 100 discretization points over 50-second trajectories. As shown in Table II, the results are very promising: on average, the shooting method cuts significantly the number of SCP iterations required to converge to a trajectory, resulting in an overall 59.4% increase in speed. The difference in performance is made more stark by the fact that the shooting method can occasionally converge in cases where SCP is unable to converge at all due to the use of naive in-collision straight-line initialization. Indeed, it is also interesting to note that in cases where the shooting method provides the final convergence, the final trajectory cost is always lower (if sometimes only slightly) than the cost returned by SCP alone, as the shooting method relies on the Newton method and thus tends to achieve a much higher proximity to the optimal on convergence than SCP.

| | E-SCP Only | E-SCP + Shooting |
|---|---|---|
| SCP Iterations | 32 | 13 |
| Reported Cost | 0.221 | 0.176 |

TABLE II: Averaged results of experiments using a shooting method to accelerate the convergence of E-SCP, while resulting in lower trajectory cost than using E-SCP alone.

## VI. Conclusions

In this paper we provided an SCP-based method for trajectory optimization with manifold constraints. Our key insight was to leverage geometric embeddings to lift a manifold-constrained trajectory optimization problem into an equivalent problem defined over a space enjoying Euclidean structure, where SCP can be readily applied. We derived sound theoretical guarantees and validated the proposed methodology via numerical experiments. Among other benefits, our method can easily accommodate implicitly-defined manifold constraints.

This work opens the field to many future avenues of research. First, we plan to study the setting with free final times $t_i$, $i = 1, \dots, \ell$, both from a theoretical and numerical standpoint. Second, related to the previous direction, it is of interest to design accurate numerical schemes that can handle multi-shooting methods and free final time settings. Third, we would like to leverage techniques from Lie group theory to improve performance when applying E-SCP to such specific class of submanifolds. Finally, we plan to evaluate our method on hardware platforms, such as robotic manipulators and test beds for free-flying robotic spacecraft.

## References

[1] A. A. Agrachev and Y. Sachkov. *Control Theory from the Geometric Viewpoint*. Springer, 2004.

[2] F. Augugliaro, A. P. Schoellig, and R. D'Andrea. Generation of collision-free trajectories for a quadrocopter fleet: A sequential convex programming approach. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2012.

[3] J. T. Betts. Survey of numerical methods for trajectory optimization. *AIAA Journal of Guidance, Control, and Dynamics*, 21(2):193–207, 1998.

[4] J. Bezanson, S. Karpinski, V. B. Shah, and A. Edelman. Julia: A fast dynamic language for technical computing, 2012. Available at http://arxiv.org/abs/1209.5145.

[5] R. Bonalli, A. Bylard, A. Cauligi, T. Lew, and M. Pavone. Trajectory optimization on manifolds: A theoretically-guaranteed embedded sequential convex programming approach. In *Robotics: Science and Systems*, 2019. Extended version available at https://arxiv.org/abs/1905.07654.

[6] R. Bonalli, A. Cauligi, A. Bylard, and M. Pavone. GuSTO: Guaranteed sequential trajectory optimization via sequential convex programming. In *Proc. IEEE Conf. on Robotics and Automation*, 2019.

[7] F. Bullo and A. D Lewis. *Geometric Control of Mechanical Systems*. Springer-Verlag, 1 edition, 2004.

[8] F. Bullo and A. D. Lewis. Reduction, linearization, and stability of relative equilibria for mechanical systems on Riemannian manifolds. *Acta Applicandae Mathematicae*, 99(1):53–95, 2007.

[9] E. Coumans. Bullet physics. Available at http://bulletphysics.org/.

[10] A. V. Dmitruk and A. M. Kaganovich. Maximum principle for optimal control problems with intermediate constraints. *Computational Mathematics and Modeling*, 22(2):2180–215, 2011.

[11] C. Ericson. *Real-Time Collision Detection*. CRC Press, 2004.

[12] A. F. Filippov. On certain questions in the theory of optimal control. *SIAM Journal on Control*, 1(1):76–84, 1962.

[13] L. Göllmann, D. Kern, and H. Maurer. Optimal control problems with delays in state and control variables subject to mixed control-state constraints. *Optimal Control Applications and Methods*, 30(4):341–365, 2008.

[14] Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*, 2018. Available at http://www.gurobi.com.

[15] L. Jaillet and J. M. Porta. Path planning with loop closure constraints using an atlas-based RRT. In *Int. Symp. on Robotics Research*, 2016.

[16] L. Janson, E. Schmerling, A. Clark, and M. Pavone. Fast Marching Tree: a fast marching sampling-based method for optimal motion planning in many dimensions. *Int. Journal of Robotics Research*, 34(7):883–921, 2015.

[17] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor, and S. Schaal. STOMP: Stochastic trajectory optimization for motion planning. In *Proc. IEEE Conf. on Robotics and Automation*, 2011.

[18] S. Karaman and E. Frazzoli. Sampling-based algorithms for optimal motion planning. *Int. Journal of Robotics Research*, 30(7):846–894, 2011.

[19] L. E. Kavraki, P. Švestka, J.-C. Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580, 1996.

[20] K. Khorasani and P. V. Kokotovic. Feedback linearization of a flexible manipulator near its rigid body manifold. *System and Control Letters*, 6(3):187–192, 1985.

[21] S. M. LaValle. *Planning Algorithms*. Cambridge Univ. Press, 2006.

[22] S. M. LaValle and J. J. Kuffner. Rapidly-exploring random trees: Progress and prospects. In *Workshop on Algorithmic Foundations of Robotics*, 2000.

[23] E. B. Lee and L. Markus. *Foundations of Optimal Control Theory*. John Wiley & Sons, 1967.

[24] J. M. Lee. *Introduction to Smooth Manifolds*. Springer, 1 edition, 2003.

[25] A. D. Lewis and D. R. Tyner. Jacobian linearisation in a geometric setting. In *Proc. IEEE Conf. on Decision and Control*, 2003.

[26] X. Liu and P. Lu. Solving nonconvex optimal control problems by convex optimization. *AIAA Journal of Guidance, Control, and Dynamics*, 37(3):750 – 765, 2014.

[27] A. Majumdar and R. Tedrake. Funnel libraries for real-time robust feedback motion planning. *Int. Journal of Robotics Research*, 36(8):947–982, 2017.

[28] Y. Mao, M. Szmuk, and B. Açikmeşe. Successive convexification of non-convex optimal control problems and its convergence properties. In *Proc. IEEE Conf. on Decision and Control*, 2016.

[29] D. Mellinger, N. Michael, and V. Kumar. Trajectory generation and control for precise aggressive maneuvers with quadrotors. *Int. Journal of Robotics Research*, 31 (5):664–674, 2012.

[30] G. Misra and X. Bai. Task-constrained trajectory planning of free-floating space-robotic systems using convex optimization. *AIAA Journal of Guidance, Control, and Dynamics*, 40(11):2857 – 2870, 2017.

[31] R. M. Murray, Z. Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, first edition, 1994.

[32] L. S. Pontryagin. *Mathematical Theory of Optimal Processes*. Taylor & Francis, 1987.

[33] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa. CHOMP: Gradient optimization techniques for efficient motion planning. In *Proc. IEEE Conf. on Robotics and Automation*, 2009.

[34] A. Saccon, A. P. Aguiar, A. J. Häusler, J. Hauser, and A. M. Pascoal. Constrained motion planning for multiple vehicles on SE(3). In *Proc. IEEE Conf. on Decision and Control*, 2012.

[35] A. Saccon, J. Hauser, and A. P. Aguiar. Optimal control on Lie groups: The projection operator approach. *IEEE Transactions on Automatic Control*, 58(9):2230–2245, 2013.

[36] J. Schulman, Y. Duan, J. Ho, A. Lee, I. Awwal, H. Bradlow, J. Pan, S. Patil, K. Goldberg, and P. Abbeel. Motion planning with sequential convex optimization and convex

collision checking. *Int. Journal of Robotics Research*, 33 (9):1251–1270, 2014.

[37] S. Shen, Y. Mulgaonkar, N. Michael, and V. Kumar. Vision-based state estimation and trajectory control towards high-speed flight with a quadrotor. In *Robotics: Science and Systems*, 2013.

[38] S. Singh, M. Chen, S. L. Herbert, C. J. Tomlin, and M. Pavone. Robust tracking with model mismatch for fast and safe planning: an SOS optimization approach. In *Workshop on Algorithmic Foundations of Robotics*, 2018. In Press.

[39] C. Suh, T. T. Um, B. Kim, H. Noh, M. Kim, and F. C. Park. Tangent space RRT: A randomized planning algorithm on constraint manifolds. In *Proc. IEEE Conf. on Robotics and Automation*, 2011.

[40] H. J. Sussman. An introduction to the coordinate-free maximum principle. In *Geometry of Feedback and Optimal Control*. 1998.

[41] J. Virgili-llop, C. Zagaris, R. Zappulla II, A. Bradstreet, and M. Romano. A convex-programming-based guidance algorithm to capture a tumbling object on orbit using a spacecraft equipped with a robotic manipulator. *Int. Journal of Robotics Research*, 38(1):40 – 72, 2019.

[42] M. Watterson, T. Smith, and V. Kumar. Smooth trajectory generation on SE(3) for a free flying space robot. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, 2016.

[43] M. Watterson, S. Liu, K. Sun, T. Smith, and V. Kumar. Trajectory optimization on manifolds with applications to SO(3) and R3XS2. In *Robotics: Science and Systems*, 2018.