Intrinsic Successive Convexification: Trajectory Optimization on Smooth Manifolds

Spencer Kraisler, *Member, IEEE*, Mehran Mesbahi, *Fellow, IEEE*, and Behçet Açıkmeşe, *Fellow, IEEE*

***Abstract*—A fundamental issue at the core of trajectory optimization on smooth manifolds is handling the implicit manifold constraint within the dynamics. The conventional approach is to enforce the dynamic model as a constraint. However, we show that this approach leads to significantly redundant operations, as well as being heavily dependent on the state space representation. Specifically, we propose an intrinsic successive convexification methodology for optimal control on smooth manifolds. This so-called iSCvx is then applied to a representative example involving attitude trajectory optimization for a spacecraft subject to non-convex constraints.**

***Index Terms*—Trajectory optimization, intrinsic convex optimization, optimization over Riemannian manifolds.**

I. Introduction

A FUNDAMENTAL challenge in formulating and solving trajectory optimization problems on manifolds lies in selecting the most appropriate state space representation. For rigid bodies, common representations include dual quaternions and homogeneous matrices. The choice of representation directly influences the solution strategy and the performance of algorithms for solving the corresponding non-linear optimal control problem. For example, the widespread adoption of dual quaternions is due to their advantageous properties in this context [1], [2], [3]. In this letter, we investigate a trajectory optimization framework invariant to the representation of the underlying configuration space, provided that the dynamics and constraints depend exclusively on coordinate-free quantities of the system manifolds. We frame this perspective in the context of the Successive Convexification (SCvx) algorithm, a popular method for fast trajectory optimization [4], [5].

The extrinsic enforcement of dynamics leads to certain redundant operations, particularly in SCvx, by searching over excess dimensions in the convex sub-problem. A systematic analysis of the extrinsic approach has been done in [6]. For example, the unit quaternions $\mathcal{Q}$ is a 3-dimension manifold embedded in a 4-dimensional ambient vector space $\mathbb{R}^4$. The fact that the linearized dynamics of the vehicle are parameterized by matrices $A_k \in \mathbb{R}^{4\times4}$ and $B_k \in \mathbb{R}^{4\times3}$ implies that we have redundancy in our parameterization. The same observation goes for the so-called virtual control $v_k \in \mathbb{R}^4$ that appears in SCvx. This observation implies that our optimizer requires more variables than necessary, resulting in more memory usage and higher time complexity.

Furthermore, the virtual control term introduced in the SCvx setup is essentially responsible for achieving two distinct objectives: satisfying the dynamical as well as the state-control constraints, while also trying to meet the implicit manifold constraint (i.e., $q_k^\top q_k = 1$). This observation highlights that if there was a way to intrinsically embed the SCvx procedure into the system manifold, then the virtual control would only need to deal with enforcing the dynamical and state-control constraints. There would also be fewer "search directions" when solving the resulting local sub-problems. Moreover, since iterates may not remain on the system manifold, additional iterates are generally required for convergence. And most importantly, SCvx is highly dependent on how the dynamics are embedded in the ambient manifold.

The intrinsic approach to general constrained optimization on manifolds is an established area that has recently received renewed interest [7], [8], [9], [10]. For intrinsic *trajectory* optimization, the authors in [11], [12] optimize over the Banach manifold of system trajectories defined on Lie groups. The contribution of this letter is the development of an SCvx-like procedure that is invariant to the representation that we call intrinsic SCvx (iSCvx). We will show that the proposed approach resolves the redundancy and representation-dependent issues mentioned above, with a focus on discrete-time systems over a finite-time horizon.

The outline of this letter is as follows. In Section II we present the problem statement, followed by an overview of required background on smooth manifolds and SCvx in Section III. The main results of this letter are then discussed in Section IV along with theoretical guarantees on the solution of the local sub-problem; this is then followed by an example of the proposed iSCvx algorithm to constrained attitude control in Section V. We conclude with a discussion on the shortcomings of the algorithms and future directions of this line of work in Section VI.

Received 16 March 2025; accepted 27 April 2025. Date of publication 19 May 2025; date of current version 5 June 2025. This work was supported in part by AFOSR under Grant FA9550-20-1-0053, and in part by NASA under Grant 80NSSC24M0212. Recommended by Senior Editor K. Savla. *(Corresponding author: Spencer Kraisler.)*

The authors are with the William E. Boeing Department of Aeronautics and Astronautics, University of Washington, Seattle, WA 98115 USA (e-mail: kraisler@uw.edu; mesbahi@uw.edu; behcet@uw.edu).

Digital Object Identifier 10.1109/LCSYS.2025.3571671

2475-1456 © 2025 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: University of Washington Libraries. Downloaded on March 08,2026 at 06:22:50 UTC from IEEE Xplore. Restrictions apply.

II. Problem Statement

Let $\mathcal{M}$ and $\mathcal{U}$ be Riemannian manifolds and $f\colon \mathcal{M}\times\mathcal{U}\to\mathcal{M}$ represent the system dynamics. The stage and final stage costs are denoted $\phi : \mathcal{M}\times\mathcal{U}\to\mathbb{R}$ and $\phi_f : \mathcal{M}\to\mathbb{R}$. Let $N>0$ be the time horizon of the problem, $\mathbf{x}=(x_0,\dots,x_N)\in\mathcal{M}^{N+1}$ the state trajectory, and $\mathbf{u}=(u_0,\dots,u_{N-1})\in\mathcal{U}^N$ the control sequence. We define the constraints $g : \mathcal{M}\times\mathcal{U}\to\mathbb{R}^{n_g}$; we assume that all functions defined above are twice continuously differentiable. The trajectory optimization problem of interest is solving

$$\min_{\mathbf{x},\mathbf{u}}\ C(\mathbf{x},\mathbf{u}) \coloneqq \sum_{k=0}^{N-1}\phi(x_k,u_k)+\phi_f(x_N) \tag{1a}$$

$$\text{s.t. } x_{k+1}=f(x_k,u_k), \tag{1b}$$

$$g(x_k,u_k)\le 0, \tag{1c}$$

$$x_0 \text{ given.} \tag{1d}$$

The key idea pursued in this letter is solving (1) in an "intrinsic" manner–one that is invariant with respect to how $\mathcal{M}$ has been embedded in an ambient vector space.

III. Background

We first provide the background notions and constructs that our subsequent analysis and results are built upon.

A. Smooth Geometry

A smooth manifold $\mathcal{M}$ is a topological space that locally behaves like a vector space [13], [14], [15]. Euclidean submanifolds are subsets of $\mathbb{R}^d$ with no corners, cusps, or "non-smooth" artifacts, such as the unit quaternions $\mathcal{Q}\coloneqq\{q\in\mathbb{R}^4 : q^\top q=1\}\subset\mathbb{R}^4$; see Section V.

A smooth curve is a smooth function $\gamma : \mathbb{R}\to\mathcal{M}$. The tangent space $T_x\mathcal{M}$ at x is the set of the tangent vectors $\dot\gamma(0)$ of all smooth curves $\gamma(\cdot)$ with $\gamma(0)=x$. The tangent space is a vector space whose dimension equals the manifold dimension. For example, $T_q\mathcal{Q}\coloneqq\{v\in\mathbb{R}^4 : v^\top q=0\}$ has dimension 3. The disjoint union of tangent spaces $T\mathcal{M}$ is called the tangent bundle. A vector field is a smooth function $V : \mathcal{M}\to T\mathcal{M}$ with $V(x)\in T_x\mathcal{M}$.

Let $f\colon\mathcal{M}\to\mathcal{N}$ be a differentiable map between smooth manifolds. The differential $\mathrm{d}f_x : T_x\mathcal{M}\to T_{f(x)}\mathcal{N}$ of f at x along $v\in T_x\mathcal{M}$ is the linear mapping

$$\mathrm{d}f_x(v)\coloneqq\left.\frac{d}{dt}\right|_{t=0} f\circ\gamma, \tag{2}$$

where $\gamma(\cdot)$ is any smooth curve satisfying $(\gamma(0),\dot\gamma(0))=(x,v)$. The differential is the canonical way of defining the notion of directional derivative.

The following example makes use of the unit quaternions. Fix $y_B\in\mathbb{R}^3$. Define $f(q)\coloneqq q\cdot y_B\cdot q^{-1}$ that rotates y_B via the rotation represented by q. Here, "$\cdot$" denotes quaternion multiplication, and we are identifying $\mathbb{R}^3$ with pure quaternions. From the above, we have

$$\mathrm{d}f_q(v)=q\cdot\left[y_B,q^{-1}\cdot v\right]\cdot q^{-1},$$

where $[a,b]\coloneqq a\cdot b-b\cdot a$.

A retraction is a smooth mapping $R : \mathcal{S}\subset T\mathcal{M}\to\mathcal{M}$ where $\mathcal{S}$ is open, $(x,0_x)\in\mathcal{S}$ for all x, and the curve $\gamma(t)\coloneqq R_x(tv)\equiv R(x,tv)$ satisfies $(\gamma(0),\dot\gamma(0))=(x,v)$ for each $(x,v)\in\mathcal{S}$ [16], [17], [18]. For vector spaces, the canonical retraction is $R_x(v)=x+v$. For a product manifold $\mathcal{M}^k$, the induced product retraction is the retraction applied elementwise: $R_{\mathbf{x}}(\mathbf{v})=(R_{x_1}(v_1),\dots,R_{x_k}(v_k))$.

By the inverse function theorem, there is a neighborhood U_x around each $0\in T_x\mathcal{M}$ such that R_x admits an inverse: $R_x^{-1} : U_x\subset T_x\mathcal{M}\to\mathcal{M}$.

The unit quaternions admit a canonical retraction. Let $\exp : \mathfrak{q}\to\mathcal{Q}$, $\exp(\omega)\coloneqq(\cos\|\omega\|_2,\operatorname{sinc}(\omega)\omega)$ be the quaternion exponential, where $\operatorname{sinc}(\omega)\coloneqq\sin\|\omega\|_2/\|\omega\|_2$. Then

$$R_q(v)\coloneqq q\cdot\exp(\omega); \tag{3}$$

here, we are expressing quaternions as $q=(q_0,q_v)$ and tangent vectors as $v=q\cdot\omega\in T_q\mathcal{Q}$. The inverse of this retraction is

$$R_q^{-1}(p)\coloneqq q\cdot\log\left(q^{-1}\cdot p\right), \tag{4a}$$

$$\log(q)\coloneqq\frac{\operatorname{atan2}(\|q_v\|,q_0)}{\|q_v\|_2}q_v, \tag{4b}$$

where $\log\colon\mathcal{Q}\to\mathfrak{q}$ denotes the quaternion logarithm.

B. Successive Convexification (SCvx)

SCvx descends from the Sequential Convex Programming (SCP) paradigm [19]. Consider a possibly infeasible trajectory $(\mathbf{x},\mathbf{u})$. A convex local sub-problem is constructed to compute the optimal perturbation $(\boldsymbol{\eta},\boldsymbol{\xi})$ around $(\mathbf{x},\mathbf{u})$, minimizing the objective while satisfying the relaxed constraints:

$$\min_{\boldsymbol{\eta},\boldsymbol{\xi},\mathbf{v},\mathbf{s}}\ C(\mathbf{x}+\boldsymbol{\eta},\mathbf{u}+\boldsymbol{\xi})+\sum_{k=0}^{N-1}\lambda_k P(v_k,s_k) \tag{5a}$$

$$\text{s.t. } g(x_k,u_k)+S_k\eta_k+T_k\xi_k\le s_k, \tag{5b}$$

$$f(x_k,u_k)-x_{k+1}+A_k\eta_k+B_k\xi_k=v_k, \tag{5c}$$

$$s_k\ge 0,\quad \|\eta_k\|_2\le r,\quad \|\xi_k\|_2\le r. \tag{5d}$$

The linearized constraints ensure the sub-problem remains convex. Specifically, the Jacobians are $S_k\coloneqq D_xg(x_k,u_k)$, $T_k\coloneqq D_ug(x_k,u_k)$, $A_k\coloneqq D_xf(x_k,u_k)$, and $B_k\coloneqq D_uf(x_k,u_k)$. To guarantee feasibility, the linearized constraints are further relaxed with the virtual control $\mathbf{v}$ and buffer zone slack variable $\mathbf{s}$. A penalty term $P : \mathbb{R}^n\times\mathbb{R}^{n_g}\to[0,\infty)$ on the slack variables is appended to the objective in order to "discourage" their usage. A trust region radius $r>0$ constrains the perturbations to ensure boundedness.

Once the sub-problem is solved, the perturbations are added to the trajectory. The trust region radius r is adjusted based on the quality of the approximation of the sub-problem to the original problem, as well as the amount of relative improvement in objective. We define the penalized objective as

$$J(\mathbf{x},\mathbf{u})\coloneqq C(\mathbf{x},\mathbf{u})+\sum_{i=1}^{N-1}\lambda_k P\Big(x_{k+1}-z_k,\left[g(x_k,u_k)\right]_+\Big)$$

with $z_k\coloneqq f(x_k,u_k)$. We call (5a) the localized objective, denoted L. Let $(\boldsymbol{\eta}^*,\boldsymbol{\xi}^*,\mathbf{v}^*,\mathbf{s}^*)$ be the solution of (5). Define the predicted change as

$$\rho\coloneqq\frac{J(\mathbf{x},\mathbf{u})-J\big(\mathbf{x}+\boldsymbol{\eta}^*,\mathbf{u}+\boldsymbol{\xi}^*\big)}{J(\mathbf{x},\mathbf{u})-L\big(\boldsymbol{\eta}^*,\boldsymbol{\xi}^*,\mathbf{v}^*,\mathbf{s}^*\big)}.$$

Authorized licensed use limited to: University of Washington Libraries. Downloaded on March 08,2026 at 06:22:50 UTC from IEEE Xplore. Restrictions apply.

A positive $\rho > 0$ indicates good approximation and an improvement to the objective. The trust region radius is updated according to

$$r^+ = \begin{cases} \alpha r & \text{if } \rho < \rho_1, \\ r & \text{if } \rho_1 \le \rho < \rho_2, \\ \beta r & \text{if } \rho \ge \rho_2 \end{cases}$$

where $0 \le \rho_0 < \rho_1 < \rho_2 < 1$ and $0 < \alpha < 1 < \beta$ are the algorithm parameters. A negative or exceptionally small $\rho < \rho_0$ indicates poor approximation of improvement; the trust region radius is reduced and the sub-problem re-solved until we have an acceptable amount of improvement.

IV. Intrinsic SCvx

In this section, we describe the main mechanisms used in iSCvx. First, we show how to linearize dynamics in an intrinsic way in Section IV-A. Next, we show how to handle the virtual control and buffer zone terms, along with the penalty in Section IV-B. Thereafter, we show how to compute an intrinsic second-order approximation of the trajectory cost in Section IV-C. Then, in Section IV-D, we introduce the geodesic local sub-problem and how to numerically solve it in Section IV-E. Last, we present the iSCvx algorithm in Section IV-F.

A. Linear Approximation on Smooth Manifolds

The key difference between iSCvx and SCvx is how we linearize the dynamics and constraints using the intrinsic differential (2) rather than the ordinary Jacobian operator (5b), (5c). Using the differential, the resulting system matrices will have smaller dimensions, resulting in faster solving time.

Now, let us consider the dynamic constraint (1b). For sufficiently small $r > 0$, there exists a unique $\eta_{k+1} \in T_{x_{k+1}}\mathcal{M}$ satisfying

$$\eta_{k+1} = \left(R^{-1}_{x_{k+1}} \circ f\right)\left(R_{x_k}(\eta_k), R_{u_k}(\xi_k)\right),$$

given $\|\eta_k\|_{x_k}, \|\xi_k\|_{u_k} \le r$. As this mapping is between vector spaces, we can linearize it in the ordinary sense.

Proposition 1: For sufficiently small $r > 0$, the first-order expansions of (1b) and (1c) are

$$\eta_{k+1} = R^{-1}_{x_{k+1}}(z_k) + \mathbf{D}_k \circ (\mathbf{A}_k(\eta_k) + \mathbf{B}_k(\xi_k)) + o(r) \quad (6a)$$

$$g\left(R_{x_k}(\eta_k), R_{u_k}(\xi_k)\right) = g_k + \mathbf{S}_k(\eta_k) + \mathbf{T}_k(\xi_k) + o(r) \quad (6b)$$

where $z_k := f(x_k, u_k)$, $g_k := g(x_k, u_k)$, and

$$\mathbf{A}_k := \mathrm{d}_x f_{(x_k,u_k)} : T_{x_k}\mathcal{M} \to T_{z_k}\mathcal{M} \quad (7a)$$

$$\mathbf{B}_k := \mathrm{d}_u f_{(x_k,u_k)} : T_{u_k}\mathcal{U} \to T_{z_k}\mathcal{M} \quad (7b)$$

$$\mathbf{D}_k := \mathrm{d}R^{-1}_{x_{k+1}}|_{z_k} : T_{z_k}\mathcal{M} \to T_{x_{k+1}}\mathcal{M} \quad (7c)$$

$$\mathbf{S}_k := \mathrm{d}_x g_{(x_k,u_k)} : T_{x_k}\mathcal{M} \to \mathbb{R}^{n_g} \quad (7d)$$

$$\mathbf{T}_k := \mathrm{d}_u g_{(x_k,u_k)} : T_{u_k}\mathcal{U} \to \mathbb{R}^{n_g}. \quad (7e)$$

Proof: Using the fact that $\mathrm{d}R_{x_k}|_0(\cdot)$ is identity and

$$\mathrm{d}\left(R^{-1}_{x_{k+1}} \circ f \circ (R_{x_k}, R_{u_k})\right) = \mathrm{d}R^{-1}_{x_{k+1}} \circ (\mathrm{d}_x f + \mathrm{d}_u f),$$

through the chain rule, we obtain (6a). The same process holds for (6b). ∎

B. Slack Variables and Penalty

Like SCvx, we use slack variables $s_k \ge 0$, $v_k \in T_{x_{k+1}}\mathcal{M}$ to ensure that the sub-problem is feasible:

$$g(x_k, u_k) + \mathbf{S}_k(\eta_k) + \mathbf{T}_k(\xi_k) \le s_k,$$
$$R^{-1}_{x_{k+1}}(z_k) + \mathbf{D}_k \circ (\mathbf{A}_k(\eta_k) + \mathbf{B}_k(\xi_k)) + v_k = \eta_{k+1}.$$

We also require an exact penalty term in order to minimize usage of these slack variables. Define a function $P : T\mathcal{M} \times \mathbb{R}^{n_g} \to \mathbb{R}$ using only coordinate-free quantities. From here, we can define the penalized objective as

$$J(\mathbf{x}, \mathbf{u}) := C(\mathbf{x}, \mathbf{u}) + \sum_{k=0}^{N-1} \lambda_k P\left(R^{-1}_{x_{k+1}}(z_k), \left[g_k\right]_+\right) \quad (8)$$

C. Sub-Problem Objective

We will describe how to construct the localized trajectory cost for the sub-problem. We require that our stage and final stage costs are geodesically convex with respect to the Riemannian structures of $\mathcal{M}$ and $\mathcal{U}$. This will imply that the second-order approximation is a positive semi-definite quadratic function, thereby making the sub-problem convex.

A Riemannian metric is a smooth assignment of an inner product $\langle ., . \rangle_x : T_x\mathcal{M} \times T_x\mathcal{M} \to \mathbb{R}$ to each point $x \in \mathcal{M}$. A smooth manifold paired with a Riemannian metric is called a Riemannian manifold. Riemannian metrics allow us to define generalized notions of the gradient.

Euclidean submanifolds equipped with $\langle v, w \rangle_x := v^\top w$ are labeled isometric. In the context of optimization, the "right" Riemannian metric can speed up the convergence rate of the corresponding Riemannian gradient method by orders of magnitude [20].

The Riemannian gradient $\nabla f(x) \in T_x\mathcal{M}$ of $f \colon \mathcal{M} \to \mathbb{R}$ is the vector field that satisfies

$$\langle \nabla f(x), v \rangle_x = \mathrm{d}f_x(v)$$

at all $(x, v) \in T\mathcal{M}$. While the Riemannian Hessian $\nabla^2 f(x) : T_x\mathcal{M} \to T_x\mathcal{M}$ requires additional theory to define, for isometric Euclidean submanifolds it becomes

$$\nabla^2 f(x)[v] := \mathrm{Proj}_{T_x\mathcal{M}}\left(\overline{\nabla}^2 f(x) v\right).$$

Here, $\overline{\nabla}^2 f(x) \in \mathbb{R}^{d \times d}$ is the ordinary Hessian matrix. We also write $\nabla^2 f(x)[v, w] := \langle \nabla^2 f(x)[v], w \rangle_x$.

We refer to [13] for definitions of geodesic convexity for sets and functions on manifolds. There is one property of geodesically convex functions that will be utilized subsequently: f is geodesically convex if and only if $\nabla^2 f \succeq 0$ and the domain is strongly geodesically convex. Then, the second-order approximation about any $x \in U$ is a positive semi-definite quadratic:

$$\hat{f}_x(v) := f(x) + \mathrm{d}f_x(v) + \frac{1}{2}\nabla^2 f(x)[v, v].$$

Suppose the stage costs $\phi(\cdot, \cdot), \phi_f(\cdot)$ are geodesically convex and $\nabla^2_{xu}\phi = 0$. Then their second-order approximations are convex quadratics:

$$\hat{\phi}|_{(x,u)}(\eta, \xi) = \phi(x, u) + \mathrm{d}_x\phi_{(x,u)}(\eta) + \mathrm{d}_u\phi_{(x,u)}(\xi)$$

Authorized licensed use limited to: University of Washington Libraries. Downloaded on March 08,2026 at 06:22:50 UTC from IEEE Xplore. Restrictions apply.

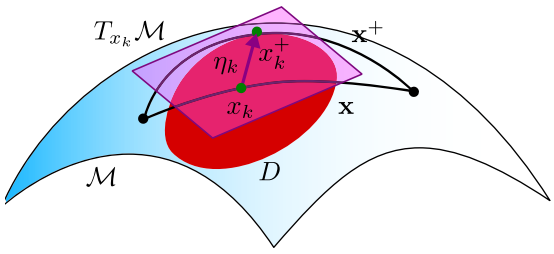

Fig. 1. Visualization of perturbing a trajectory along a tangent space.

$$+\frac{1}{2}\nabla^2_{xx}\phi(x,u)[\eta,\eta]+\frac{1}{2}\nabla^2_{uu}\phi(x,u)[\xi,\xi],$$

$$\hat{\phi}_f|_x(\eta)=\phi_f(x)+\mathrm{d}_x\phi_f|_x(\eta)+\frac{1}{2}\nabla^2\phi_f(x)[\eta,\eta].$$

This implies that the trajectory cost is geodesically convex, and so its second-order approximation will be a convex quadratic:

$$\hat{C}_{(\mathbf{x},\mathbf{u})}(\boldsymbol{\eta},\boldsymbol{\xi})=\sum_{k=0}^{N-1}\hat{\phi}_{(x_k,u_k)}(\eta_k,\xi_k)+\hat{\phi}_f|_{x_N}(\eta_N). \tag{9}$$

D. Geodesic Local Sub-Problem

The geodesic local sub-problem about $(\mathbf{x},\mathbf{u})$ is

$$\min_{\boldsymbol{\eta},\boldsymbol{\xi},\mathbf{v},\mathbf{s}}\ \hat{C}_{(\mathbf{x},\mathbf{u})}(\boldsymbol{\eta},\boldsymbol{\xi})+\sum_{k=0}^{N-1}\lambda_k P(v_k,s_k) \tag{10a}$$

$$\text{s.t.}\ \ \eta_{k+1}=R^{-1}_{x_{k+1}}(z_k)+\tilde{\mathbf{A}}_k(\eta_k)+\tilde{\mathbf{B}}_k(\xi_k)+v_k, \tag{10b}$$

$$g(x_k,u_k)+\mathbf{S}_k(\eta_k)+\mathbf{T}_k(\xi_k)\le s_k, \tag{10c}$$

$$s_k\ge 0,\quad \|\eta_k\|_{x_k}\le r,\quad \|\xi_k\|_{u_k}\le r. \tag{10d}$$

where $\tilde{\mathbf{A}}_k\coloneqq\mathbf{D}_k\circ\mathbf{A}_k$ and $\tilde{\mathbf{B}}_k\coloneqq\mathbf{D}_k\circ\mathbf{B}_k$. We call (10a) the localized objective and denote it $L(\boldsymbol{\eta},\boldsymbol{\xi},\mathbf{v},\mathbf{s})$. It generalizes the objective (5a).

Lemma 1: If the stage costs are continuously differentiable and geodesically convex over a chosen domain, then a solution exists for (10).

Proof: Recall that the sub-problem is feasible if the feasibility set is non-empty. Due to the virtual control term v_k, any state in the feasible region of the convex sub-problem is reachable in finite time. Furthermore, the virtual buffer zone variables s_k ensure that the feasible set defined by the linearized state and control constraints is non-empty. Hence, an optimal solution exists. ∎

E. Coordinate Representations

Problem 10 is written in terms of tangent vectors and linear operators. To solve it, we must choose a frame and compute the coordinate representations of the quantities appearing in the problem. The optimization variables will therefore be the corresponding components.

For $U\subset\mathcal{M}$, a local frame is an ordered set of n linearly independent vector fields $E_i: U\to T\mathcal{M}$. A global frame for the unit quaternions is $E_i(q)\coloneqq q\cdot e_i$, where $e_i\in\mathbb{R}^3$ is the ith standard basis vector. Let $(x,v)\in T\mathcal{M}$. Then the components of v is the coordinate vector $[v]\coloneqq(v^1,\dots,v^n)\in\mathbb{R}^n$ which uniquely satisfies $v=\sum_{i=1}^n v^i E_i(x)$. The components of the Riemannian metric will be a positive definite matrix $G_{ij}(x)=\langle E_i(x),E_j(x)\rangle_x$. For $w\in T_x\mathcal{M}$, we can write $\langle v,w\rangle_x=[v]^\top[G(x)][w]$.

Let $\mathcal{N}$ be a Riemannian manifold of dimension k and (F_i) a local frame about $y\in\mathcal{N}$. Let $\mathbf{A}: T_x\mathcal{M}\to T_y\mathcal{N}$ be linear. Its components is the matrix $[\mathbf{A}]\in\mathbb{R}^{k\times n}$ defined as follows. Set $A_j\coloneqq\mathbf{A}(E_j(x))\in T_y\mathcal{N}$. Then $[A_j]\in\mathbb{R}^k$ will be the jth column of $[\mathbf{A}]$. Now we can apply this methodology to turn the abstract Problem 10 into a computable convex problem.

The coordinate representation of the final stage cost is

$$\phi_f(x)+\left[\mathrm{d}\phi_f|_x\right][\eta]+\frac{1}{2}[\eta]^\top\left[\nabla^2\phi_f(x)\right][\eta],$$

which is a convex quadratic function. Applying the same methodology to the stage cost and adding the penalty term, we can then represent the sub-problem objective (10a) as a computable convex function.

The coordinate representation of (10b) is the linear equation

$$\left[\eta_{k+1}\right]=\left[R^{-1}_{x_{k+1}}(z_k)\right]+[\mathbf{D}_k]([\mathbf{A}_k][\eta_k]+[\mathbf{B}_k][\eta_k])+[v_k].$$

A similar process can be done for the coordinate representation of (10c).

The bounds are represented as $[\eta]^\top[G(x)][\eta]\le r^2$ and similar for $\|\xi\|_u$. Using these coordinate representations, we can represent (10) as a computable convex program.

F. iSCvx Procedure

We present Algorithm 1 with the following remarks. First, it adheres to the SCvx protocol laid out in Section III-B and further detailed in [4, Algorithm 2.1]. The quantities J and L are defined in (8) and (10a), respectively. Second, since Problem 10 optimizes over tangent vectors coordinatized by a chosen frame, the sub-problem involves fewer optimization variables. This results in faster computation. Because the perturbations η_k, ξ_k, v_k are tangent vectors, we are effectively optimizing over the necessary directions, reducing the magnitude and need to satisfy the implicit manifold constraint. Third, since Problem 10 is entirely written in intrinsic manifold quantities, the computational outcome is independent of representation.

V. Example: Constrained Attitude Guidance iSCvx

In this section, we compare iSCvx to its "Euclidean" realization for constrained attitude guidance.

Our dynamics are the discrete kinematics for rotating bodies using unit quaternions:

$$q_{k+1}=f(q_k,\omega_k)=q_k\cdot\exp(\Delta t\omega_k)$$

where $\Delta t>0$ is the time step. This describes rotating an object with attitude q_k about $\omega_k\in\mathbb{R}^3$ by an angle $\Delta t\|\omega_k\|_2$. We fix an initial q_0 and desired q_f. The chosen retraction is (3). The global frame is the one described in Section IV-E.

We next set a boresight direction y_B fixed in the body frame, a keep-out direction t_o fixed in the inertial frame, and a zone angular radius $\theta_{\max}\in[0,\pi)$. The keep-out zone constraint is

$$g(q)\coloneqq t_o^\top y_o-\cos\theta_{\max}.$$

Authorized licensed use limited to: University of Washington Libraries. Downloaded on March 08,2026 at 06:22:50 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 iSCvx Algorithm

Input: Choose initial trajectory $(\mathbf{x}, \mathbf{u}) \in \mathcal{M}^{N+1} \times \mathcal{U}^N$. Select tolerance ϵ_{tol}, trust region $r > 0$, penalty weight $\lambda > 0$, and parameters $0 \leq \rho_0 < \rho_1 < \rho_2 < 1$ and $0 < \alpha < 1 < \beta$, and sufficiently large $\Delta J > 0$.
while $|\Delta J| > \epsilon_{\text{tol}}$ **do**
  Solve Problem (10) to get $\boldsymbol{\eta}, \boldsymbol{\xi}, \mathbf{v}, \mathbf{s}$.
  Compute

$$\Delta J \leftarrow J(\mathbf{x}, \mathbf{u}) - J(R_{\mathbf{x}}(\boldsymbol{\eta}), R_{\mathbf{u}}(\boldsymbol{\xi}))$$
$$\Delta L \leftarrow J(\mathbf{x}, \mathbf{u}) - L(\boldsymbol{\eta}, \boldsymbol{\xi}, \mathbf{v}, \mathbf{s})$$

  Compute $\rho \leftarrow \Delta J / \Delta L$.
  if $\rho < \rho_0$ **then**
    $r \leftarrow \alpha r$
  else
    Update $\mathbf{x} \leftarrow R_{\mathbf{x}}(\boldsymbol{\eta})$, $\mathbf{u} \leftarrow R_{\mathbf{u}}(\boldsymbol{\xi})$
    Update $r \leftarrow \begin{cases} \alpha r, & \text{if } \rho < \rho_1; \\ r, & \text{if } \rho_1 \leq \rho < \rho_2; \\ \beta r, & \text{if } \rho \geq \rho_2. \end{cases}$
  end if
end while
return $\mathbf{x}$, $\mathbf{u}$

Thus, $g(q) \leq 0$ implies the angle between t_o and $y_o := q \cdot y_B \cdot q^{-1}$ is at least $\theta_{\max}$. For both SCvx and iSCvx, we use the penalty

$$P(v, s) := \|[v]\|_1 + \|s\|_1 = \|q^{-1} \cdot v\|_1 + \|s\|_1.$$

For SCvx, the (Euclidean) trajectory cost is

$$\sum_{k=0}^{N-1} \left(\lambda_q \|q_k - q_f\|_2^2 + \lambda_\omega \|\omega_k\|_2 \right) + \lambda_f \|q_N - q_f\|_2^2,$$

where $\lambda_q, \lambda_\omega, \lambda_f \geq 0$ are the reward coefficients.

Now, we will describe the iSCvx setup. To construct geodesically convex stage and final stage costs, we use the squared geodesic distance: $\rho(q) := \|\log(q_f^{-1} q)\|_2^2$. This function is geodesically convex over the domain $D := \{q : \sqrt{\rho(q)} < \pi/2\}$ [21]. The stage costs are

$$\phi(q, \omega) := \lambda_q \rho(q) + \lambda_\omega \|\omega\|_2^2,$$
$$\phi_f(q) := \lambda_f \rho(q).$$

Their derivatives are

$$\mathrm{d}\phi_f|_q(v) = \lambda_f \mathrm{d}\rho_q(v), \mathrm{d}_q\phi_{(q,\omega)}(v) = \lambda_q \mathrm{d}\rho_q(v),$$
$$\nabla^2 \phi_f(q) = \lambda_f \nabla^2 \rho(q), \nabla^2_{qq}\phi(q, \omega) = \lambda_q \nabla^2 \rho(q)$$
$$\mathrm{d}_\omega \phi_{(q,\omega)}(\xi) = 2\lambda_\omega \xi^\top \omega, \nabla^2_{\omega\omega}\phi(q, \omega) = 2\lambda_\omega I$$

with $\nabla^2_{q\omega}\phi = 0$. Next, the derivatives of ρ are

$$\mathrm{d}\rho_q(v) = 2v^\top \left(q \cdot \log(q_f^{-1} \cdot q) \right),$$
$$\nabla^2 \rho(q)[v, v] = 2v^\top \left(uu^\top + \theta \cot\theta (I - qq^\top - uu^\top) \right) v.$$

where $\theta = \rho(q)$ and $u = \frac{(I - qq^\top)}{\sin\theta} q_f$. We will designate (1a) the geodesic trajectory cost.

TABLE I
SCvx and iSCvx Comparison: Case 1

	$\theta_{\max} = 10°$		$\theta_{\max} = 30°$	
	SCvx	**iSCvx**	**SCvx**	**iSCvx**
Avg. Iter.	40.21	24.89	45.8	26.8
Std. Iter.	9.23	2.14	18.28	1.88
Time (s)	6.26	4.40	7.10	4.70
Avg. Geo. Cost	4.98	4.73	6.50	5.67
Avg. Eucl. Cost	7.21	6.91	9.65	8.17

Results compare SCvx and iSCvx for spacecraft attitude control under different angle constraints with $N = 30$ time steps and discretization rate $\Delta t = 0.1$ sec. Values are averages over multiple runs.

TABLE II
SCvx and iSCvx Comparison: Case 2

	$\theta_{\max} = 10°$		$\theta_{\max} = 30°$	
	SCvx	**iSCvx**	**SCvx**	**iSCvx**
Avg. Iter.	67.9	24.75	65.72	25.65
Std. Iter.	34.86	2.22	17.02	2.45
Time (s)	22.18	9.09	21.43	9.37
Avg. Geo. Cost	9.04	8.96	10.59	10.50
Avg. Eucl. Cost	11.94	13.34	13.98	15.42

$N = 60$ time steps and discretization rate $\Delta t = 0.05$ sec.

The differentials of the system dynamics and constraints (7) can now be computed as

$$\mathbf{A}_k(\eta_k) = \mathrm{d}_q f_{(q_k,\omega_k)}(\eta_k) = \eta_k \cdot \exp(\Delta t \omega_k),$$
$$\mathbf{B}_k(\xi_k) = \mathrm{d}_\omega f_{(q_k,\omega_k)}(\xi_k) = \Delta t q_k \cdot \mathrm{d}\exp_{\Delta t \omega_k}(\xi_k),$$
$$\mathbf{S}_k(\eta_k) = \mathrm{d}g_{(q_k,\omega_k)}(\eta_k) = t_o^\top \left(q_k \cdot \left[q_k^{-1} \cdot \eta_k, y_B \right] \cdot q_k^{-1} \right),$$
$$\mathbf{D}_k(v) = q_{k+1} \cdot \mathrm{d}\log_{q_{k+1}^{-1} \cdot z_k} \left(q_{k+1}^{-1} \cdot v \right),$$

where $z_k := f(q_k, \omega_k)$. These expressions make use of the derivatives of the quaternion exponential $\mathrm{d}\exp_\eta : \mathbb{R}^3 \to T_{\exp(\eta)}\mathcal{Q} \subset \mathbb{R}^4$ and logarithm $\mathrm{d}\log_q : T_q\mathcal{Q} \subset \mathbb{R}^4 \to \mathbb{R}^3$. These have the following closed form expressions [22], [23], [24]:

$$\mathrm{d}\exp_\omega(\eta) = \begin{bmatrix} -\mathrm{sinc}(\omega)\omega^\top \\ \mathrm{sinc}(\omega) I_3 + \nabla \mathrm{sinc}(\omega) \omega^\top \end{bmatrix} \eta,$$
$$\mathrm{d}\log_q(v) = \begin{bmatrix} -\|q_v\|_2^{-2} q_v & \alpha \|q_v\|_2^{-1} I_3 - \alpha \|q_v\|_2^{-3} q_v q_v^\top \end{bmatrix} v,$$
$$\nabla \mathrm{sinc}(\omega) = \left(\frac{\cos\|\omega\|_2}{\|\omega\|_2^2} - \frac{\sin\|\omega\|_2}{\|\omega\|_2^3} \right) \omega.$$

with $\alpha := \mathrm{atan2}(\|q_v\|, q_0)$.

A. Numerical Results

The numerical simulation[1] begins with randomly chosen initial q_0 and desired q_f attitudes. The initial trajectory was derived through spherical linear interpolation. We implemented both SCvx and iSCvx with algorithm parameters $\epsilon_{\text{tol}} = 10^{-5}$, $r = 1$, $\alpha = .5$, $\beta = 3.2$, $\rho_0 = 0$, $\rho_1 = .25$, and $\rho_2 = .7$. The objective coefficients are $\lambda_q = 1$, $\lambda_f = 10$, $\lambda_\omega = .1$, and $\lambda = 10^5$. We repeated this process 100 times and recorded the average and standard deviation of the number of iterations for each algorithm, as well as the average clock time, geodesic, and Euclidean trajectory costs. We also stopped the algorithms if the iteration count exceeded $M = 100$. The corresponding results are presented in Tables I and II.

In Table I, iSCvx outperforms SCvx in every category for $\theta_{\max} = 30°$ and $\theta_{\max} = 10°$. Paired with the fact that

[1]Code available at github.com/Rainlabuw/intrinsic-scvx.

Authorized licensed use limited to: University of Washington Libraries. Downloaded on March 08,2026 at 06:22:50 UTC from IEEE Xplore. Restrictions apply.

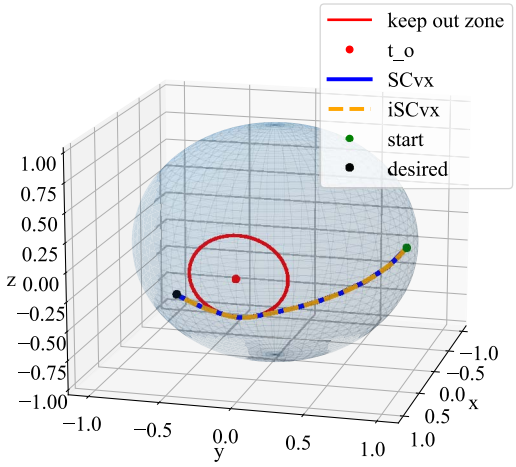

Fig. 2. The iterates of SCvx and iSCvx for the constrained attitude guidance problem with parameters $N = 30$, $\Delta t = 0.05$, and $\theta_{\max} = 20°$.

iSCvx achieves significantly lower average iteration count, standard deviation of iteration counts, and wall clock time, this experiment shows iSCvx has a natural advantage over SCvx for this particular optimal control problem. In Table II, while we see SCvx achieving a lower Euclidean trajectory cost, the difference is small, especially considering iSCvx has significantly more attractive results in the other categories. Figure 2 shows the optimal trajectories under both algorithms, and hence both costs. As shown, the trajectories are nearly identical.

VI. Conclusion

This letter proposes the intrinsic successive convexification algorithm for solving non-convex trajectory optimization problems. We outlined the key ingredients of the proposed algorithm and discussed the computational benefits of the intrinsic geometry of the underlying smooth manifold during the iterates. A numerical example, demonstrating the application of the so-called iSCvx to constrained attitude guidance is then discussed.

iSCvx has certain shortcomings. For example, in the SCvx framework, rather than linearizing all constraints, the convex constraints can instead be enforced directly in the sub-problem. This convenience is the result of convexity in the Euclidean setting. Geodesic convexity lacks this property, requiring all constraints to be linearized. This shortcoming will make iSCvx comparatively slower in such scenarios.

Additionally, the SCvx literature includes discretization techniques and the machinery for handling free final time, fixed terminal constraints, and non-smooth constraints and objectives. iSCvx currently lacks these capabilities. Unlike the convex constraint issue, these limitations are not the result of a non-Euclidean setting. Notably, first-order holds and subgradients can be generalized to smooth manifolds, suggesting a pathway to resolving these shortcomings.

Future directions include implementing first-order hold techniques for discretizing continuous dynamics, as well as handling a free-final time objective. We plan to test the algorithm on more complicated examples such as the 6-DOF powered descent and generalize this methodology to other SCP paradigms [25, Ch. 12].

References

[1] U. Lee and M. Mesbahi, "Optimal power descent guidance with 6-DoF line of sight constraints via unit dual quaternions," in *Proc. AIAA Guid., Navigation, Control Conf.*, 2015, pp. 1–21.
[2] U. Lee and M. Mesbahi, "Dual quaternions, rigid body mechanics, and powered-descent guidance," in *Proc. IEEE Conf. Decis. Control*, 2012, pp. 3386–3391.
[3] M. Szmuk, T. Reynolds, B. Acikmese, M. Mesbahi, and J. M. Carson, "Successive convexification for 6-DoF powered descent guidance with compound state-triggered constraints," in *Proc. AIAA Scitech Forum*, 2019, pp. 1–16.
[4] Y. Mao, M. Szmuk, X. Xu, and B. Açikmese, "Successive convexification: A superlinearly convergent algorithm for non-convex optimal control problems," 2018, *arXiv:1804.06539*.
[5] D. Malyuta et al., "Convex optimization for trajectory generation: A tutorial on generating dynamically feasible trajectories reliably and efficiently," *IEEE Control Syst. Mag.*, vol. 42, no. 5, pp. 40–113, Oct. 2022.
[6] R. Bonalli, A. Bylard, A. Cauligi, T. Lew, and M. Pavone, "Trajectory optimization on manifolds: A theoretically-guaranteed embedded sequential convex programming approach," 2019, *arXiv:1905.07654*.
[7] S. T. Smith, "Optimization techniques on Riemannian manifolds," 2014, *arXiv:1407.5965*.
[8] C. Liu and N. Boumal, "Simple algorithms for optimization on Riemannian manifolds with constraints," *Appl. Math. Optim.*, vol. 82, no. 3, pp. 949–981, 2020.
[9] J. Hu, X. Liu, Z.-W. Wen, and Y.-X. Yuan, "A brief introduction to manifold optimization," *J. Oper. Res. Soc. China*, vol. 8, pp. 199–248, Apr. 2020.
[10] R. Bergmann and R. Herzog, "Intrinsic formulation of KKT conditions and constraint qualifications on smooth manifolds," *SIAM J. Optim.*, vol. 29, no. 4, pp. 2423–2444, 2019.
[11] A. Saccon, J. Hauser, and A. P. Aguiar, "Optimal control on lie groups: The projection operator approach," *IEEE Trans. Autom. Control*, vol. 58, no. 9, pp. 2230–2245, Sep. 2013.
[12] J. Hauser and D. G. Meyer, "The trajectory manifold of a nonlinear control system," in *Proc. IEEE Conf. Decis. Control*, 1998, pp. 1034–1039.
[13] N. Boumal, *An Introduction to Optimization on Smooth Manifolds*. Cambridge, U.K.: Cambridge Univ. Press, 2023.
[14] J. M. Lee, *Introduction to Smooth Manifolds*. Cham, Switzerland: Springer, 2012.
[15] J. M. Lee, *Introduction to Riemannian Manifolds*. Cham, Switzerland: Springer, 2018.
[16] N. Boumal, P.-A. Absil, and C. Cartis, "Global rates of convergence for nonconvex optimization on manifolds," *IMA J. Numer. Anal.*, vol. 39, no. 1, pp. 1–33, 2019.
[17] T. T. Truong, "Unconstrained optimisation on Riemannian manifolds," 2020, *arXiv:2008.11091*.
[18] M. Lezcano-Casado, "Trivializations for gradient-based optimization on manifolds," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–12.
[19] T. P. Reynolds and M. Mesbahi, "The crawling phenomenon in sequential convex programming," in *Proc. Amer. Control Conf.*, 2020, pp. 3613–3618.
[20] S. Kraisler and M. Mesbahi, "Output-feedback synthesis orbit geometry: Quotient manifolds and LQG direct policy optimization," *IEEE Control Syst. Lett.*, vol. 8, pp. 1577–1582, 2024.
[21] R. Tron, B. Afsari, and R. Vidal, "Riemannian consensus for manifolds with bounded curvature," *IEEE Trans. Autom. Control*, vol. 58, no. 4, pp. 921–934, Apr. 2013.
[22] J. Sola, "Quaternion kinematics for the error-state Kalman filter," 2017, *arXiv:1711.02508*.
[23] W. Rossmann, *Lie Groups: An Introduction Through Linear Groups*. Oxford, U.K.: Oxford Univ. Press, 2006.
[24] E. Hairer, M. Hochbruck, A. Iserles, and C. Lubich, "Geometric numerical integration," *Oberwolfach Rep.*, vol. 3, no. 1, pp. 805–882, 2006.
[25] A. A. Agrachev and Y. Sachkov, *Control Theory From the Geometric Viewpoint*, vol. 87. Cham, Switzerland: Springer, 2013.

Authorized licensed use limited to: University of Washington Libraries. Downloaded on March 08,2026 at 06:22:50 UTC from IEEE Xplore. Restrictions apply.