Optimal Control on Lie Groups: The Projection Operator Approach

Alessandro Saccon, John Hauser and A. Pedro Aguiar

***Abstract*—Many nonlinear systems of practical interest evolve on Lie groups or on manifolds acted upon by Lie groups. Examples range from aircraft and underwater vehicles to quantum mechanical systems. In this paper, we develop an algorithm for solving continuous time optimal control problems for systems evolving on (noncompact) Lie groups. This algorithm generalizes the projection operator approach for trajectory optimization originally developed for systems on vector spaces. Notions for generalizing system theoretic tools such as Riccati equations and linear and quadratic system approximations are developed. In this development, the covariant derivative of a map between two manifolds plays a key role in providing a chain rule for the required Lie group computations. An example optimal control problem on $SO(3)$ is provided to highlight implementation details and to demonstrate the effectiveness of the method.**

I. Introduction

The optimal control of a continuous time process is among the oldest and most extensively studied problems in control theory. The main pillars of optimal control theory are Bellman's principle of optimality [1] and Pontryagin's maximum principle [2], both developed during the 60's, and the Hamilton-Jacobi-Bellman partial differential equation and its *unique* viscosity solution [3], [4], studied deeply in the 80's. Many books have been written on the subject, a sampling includes [5], [6], [7].

Various numerical methods have been proposed in the literature for solving optimal control problems on $\mathbb{R}^n$. A method is called *indirect* if it seeks to solve the first order necessary optimality conditions of the Pontryagin maximum principle, requiring the solution of a *two-point boundary value problem*. On the other hand, a method is called *direct* if the minimization problem is tackled directly, generating a descending sequence of trajectories. In a direct method, the continuous-time optimal control problem is typically *transcribed* into a finite dimensional constrained optimization problem by discretizing the continuous time dynamics, integral cost, and state-input constraints; the transcribed problem is then solved by using a state-of-the-art nonlinear programming solver.

A. Saccon is with the the Laboratory of Robotics and Systems in Engineering and Science (LARSyS), Instituto Superior Técnico (IST), Lisboa, Portugal. asaccon@isr.ist.utl.pt

A.P. Aguiar is with the Department of Electrical and Computer Engineering, Faculty of Engineering, University of Porto (FEUP), Portugal, and with the Laboratory of Robotics and Systems in Engineering and Science (LARSyS), Lisbon, Portugal. pedro.aguiar@fe.up.pt

J. Hauser is with the Department of Electrical, Computer, and Energy Engineering, University of Colorado at Boulder. john.hauser@colorado.edu

This work was supported in part by projects CONAV/FCT-PT PTDC/EEACRO/113820/2009), FCT (PEst-OE/EEI/LA0009/2011), MORPH (EU FP7 under grant agreement No. 288704), and the US AFOSR grant FA9550-09-1-0470. The first author benefited from a postdoctoral scholarship of FCT.

The purpose of this paper is to present an algorithm for solving continuous time optimal control problems for systems evolving on Lie groups. The proposed numerical algorithm can be used for solving *general* optimal control problems on Lie groups, without restricting the attention to left (or right) invariant optimal control problems. Part of this work has been reported in preliminary form in [8], [9], [10].

Theoretical investigations on optimal control problems on Lie groups has started in the '70, with the pioneering works of Brockett [11] and Baillieul [12]. The literature on optimal control problems on Lie groups has grown steadily since then and the field is still an active area of research [13]. Quite interesting sources are the excellent book of Jurdjevic [14, Chapter 12] and the more recent book of Agrachev and Sachkov [15, Chapter 18] and references therein.

Despite the large and growing literature on geometrical integration [16], [17] and finite dimensional optimization on smooth manifolds [18], [19], there are not so many numerical algorithms available for solving continuous-time optimal control problems on Lie groups. Exceptions to this general statement include the recently-proposed numerical algorithms to address optimal control problems for *mechanical systems* evolving on smooth manifolds (such as Lie groups) presented in [20], [21], [22].

The algorithm proposed in this work is a *direct* method for solving continuous time optimal control problems, generating a descending sequence of system trajectories. In contrast to many direct methods, the continuous-time optimal control problem is not transcribed into a discrete optimization problem, but rather a continuous-time second-order approximation is computed at each iteration. We borrow from and expand the key results of the projection operator approach for optimization of trajectory functionals developed in [23] to the class of systems evolving on Lie groups. The projection operator based optimization approach can handle optimal state transfer [24] and state-control constraints using a barrier function approach [25]. It has been used, in the context of virtual prototyping, to obtain a dynamic inversion procedure for the dynamics of a racing motorcycle [26]. Further applications include [27], [28], and [29].

The algorithm can be viewed as a generalization of Newton's method to the infinite dimensional setting and exhibits a second order convergence rate to a local minimimzer at which the second order sufficient condition (SSC) for optimality holds. At each step, a quadratic model of the original cost functional is constructed about the current trajectory iterate. The quadratic model is obtained from first and second derivatives of the incremental cost, terminal cost, and control system vector field. An interesting property of the algorithm, which connects it to indirect methods, is that it also generates a

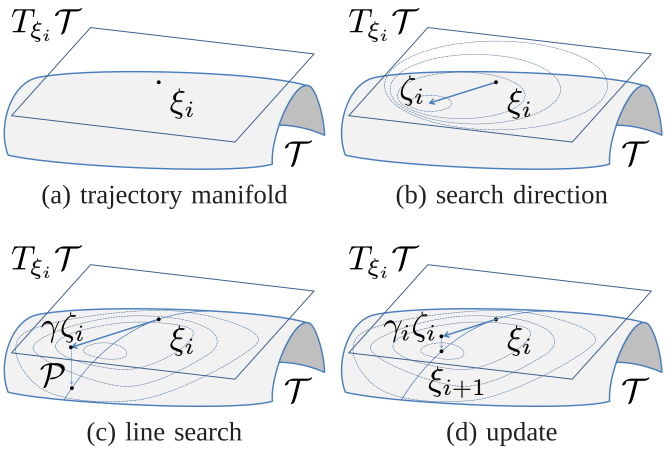

Fig. 1. The projection operator approach; (a) at each iteration, the linearization of the control system about the trajectory ξ_i defines the tangent space to the trajectory manifold $\mathcal{T}$ at ξ_i; (b) the constrained minimization over the tangent space of the second order approximation of the extended cost functional $\tilde{h} = h \circ \mathcal{P}$ yields the *search direction* ζ_i; (c) the optimal *step size* is computed through a line search along ζ_i; (d) the search direction ζ_i and step size γ_i are combined to obtain a new *update* trajectory ξ_{i+1}.

sequence of adjoint state trajectories that converges, as we approach a local minimum, to the adjoint state trajectory of the first order necessary condition. These key properties are maintained in the extension to Lie groups that we propose in this work. We also provide a simple nontrivial optimization example which is worked out in detail to illustrate the method.

The paper is organized as follows. The projection operator approach for the optimization of trajectory functionals in Banach space, originally proposed in [23], is reviewed in Section II. In the same section, after introducing the notation on differential geometry used throughout the paper, a high-level description of the projection operator approach on Lie groups is presented. The remaining sections of the paper provide the low-level details of the method. In particular, Section III introduces the definition of the left-trivialized linearization of a control system on a Lie group and Section IV presents the key concept of *covariant derivative of a map* between two manifolds. Section V defines the Lie group projection operator $\mathcal{P}$ together with its linearization and second covariant derivative. Section VI details the search direction subproblem which is at the heart of the projection operator optimization strategy. A numerical example is presented in Section VII to demonstrate the effectiveness of the method. Conclusions are drawn in Section VIII. Further technical details are collected in the Appendices A and B.

II. The Projection Operator Approach

This section reviews the projection operator approach on a vector space [23], before presenting its extension to Lie groups. The section also introduces the basic notation and symbols that will used frequently throughout the paper.

A. A review of the projection operator approach on a vector space

The projection operator approach to the optimization of trajectory functionals is an iterative algorithm which, in its easiest formulation, allows one to perform local Newton (or quasi-Newton) optimization of the cost functional

$$h(x,u) := \int_0^{t_f} l(x(\tau),u(\tau),\tau)\,d\tau + m(x(t_f)) \qquad (1)$$

over the set of trajectories of a nonlinear system $\dot{x} = f(x,u)$, $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, subject to a fixed initial condition x_0. In this paper, we use the word *trajectory* in an extended sense to indicate the state-control pair $\eta(t) = (x(t), u(t))$, $t \geq 0$, that satisfies $\dot{x}(t) = f(x(t), u(t))$ for all $t \geq 0$. As usual, "all t" means "almost all t" in the sense that

$$x(t) = x(0) + \int_0^t f(x(\tau), u(\tau))d\tau$$

where $\int \ldots d\tau$ is the Lebesgue integral. The cost functional h appearing in (1) — which is defined in terms of the incremental and terminal costs l and m — as well as the control vector field f are assumed to be sufficiently smooth and regular [23].

As shown in [30], the set $\mathcal{T}$ of trajectories of the nonlinear control system $\dot{x} = f(x,u)$ has the structure of a (infinite dimensional) Banach manifold, a fact that allows one to use vector space operations [31] to effectively explore it. To work on the trajectory manifold $\mathcal{T}$, one *projects* state-control curves in the ambient Banach space onto $\mathcal{T}$ by using a local linear time-varying trajectory tracking controller. To this end, suppose that $\xi(t) = (\alpha(t), \mu(t))$, $t \geq 0$, is a bounded state-control curve (an approximate trajectory) and let $\eta(t) = (x(t), u(t))$, $t \geq 0$, be the trajectory of $\dot{x} = f(x,u)$ determined by the nonlinear feedback system

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)), \\ u(t) &= \mu(t) + K(t)(\alpha(t) - x(t)), \end{aligned}$$

with $x(0) = x_0$. Under the hypotheses that the control vector field f is (at least) twice continuously differentiable and the gain K is bounded [30], this feedback system defines a continuous, nonlinear operator

$$\mathcal{P} : \xi = (\alpha, \mu) \mapsto \eta = (x, u) \ .$$

It is straightforward to see that ξ is a fixed point of $\mathcal{P}$, $\xi = \mathcal{P}(\xi)$, if and only if ξ is a trajectory of the control system $\dot{x} = f(x,u)$. This ensures that $\mathcal{P}^2 = \mathcal{P}$ so that $\mathcal{P}$ is a *projection operator*. With this projection operator at hand, one can see [23] that the constrained and unconstrained optimization problems

$$\min_{\xi \in \mathcal{T}} h(\xi) \quad \text{and} \quad \min_{\xi} h(\mathcal{P}(\xi))$$

are essentially equivalent in the sense that a solution to the first *constrained* problem is a solution to the second *unconstrained* problem, while a solution to the second problem is, projected by $\mathcal{P}$, a solution to the first problem. Using these facts, one may develop Newton and quasi-Newton descent methods for trajectory optimization in an effectively unconstrained manner by working with the cost functional $\tilde{h}(\xi) := h(\mathcal{P}(\xi))$. The algorithm proposed in [23] is the following:

Algorithm (Projection operator Newton method)
given initial trajectory $\xi_0 \in \mathcal{T}$

for $i = 0, 1, 2, \ldots$

redesign feedback K if desired/needed

(*search direction*)

$$\zeta_i = \arg \min_{\zeta \in T_{\xi_i}\mathcal{T}} \mathbf{D}\tilde{h}(\xi_i)\cdot\zeta + \tfrac{1}{2}\,\mathbf{D}^2\tilde{h}(\xi_i)\cdot(\zeta,\zeta) \qquad (2)$$

$$\gamma_i = \arg \min_{\gamma\in(0,1]} \tilde{h}(\xi_i + \gamma\zeta_i) \quad (\textit{step size}) \qquad (3)$$

$$\xi_{i+1} = \mathcal{P}(\xi_i + \gamma_i\zeta_i) \quad (\textit{update}) \qquad (4)$$

end

Note that the functional $\tilde{h}$ and the projection operator $\mathcal{P}$ depend on the choice of the feedback K. In (2), $\mathbf{D}\tilde{h}(\xi_i)$ and $\mathbf{D}^2\tilde{h}(\xi_i)$ are the first and second Fréchet derivatives of the Banach space functional $\tilde{h}$. When $\xi \in \mathcal{T}$ and $\zeta \in T_\xi\mathcal{T}$, the first derivative $\mathbf{D}\tilde{h}(\xi)\cdot\zeta$ simply equals $\mathbf{D}h(\xi)\cdot\zeta$, i.e., it does *not* depend on $\mathcal{P}$ (as in this case $\mathbf{D}\mathcal{P}(\xi)\cdot\zeta = \zeta$).

At each step, the minimization of a second order approximation of the extended cost functional $\tilde{h}$ provides a *search direction*. Then an optimal *step size* is computed through a line search (a pure Newton method would use a fixed step size of $\gamma_i = 1$). Combining the search direction ζ_i with step size γ_i a new *update* trajectory ξ_{i+1} is computed by projecting the curve $\xi_i + \gamma_i\zeta_i$ into the trajectory manifold $\mathcal{T}$ and the algorithm restarts (unless a termination condition is met). An illustration of the approach is shown in Figure 1.

The optimal search direction ζ_i computed in (2) is constrained to lie on the tangent space to the trajectory manifold at the current iterate, i.e., $\zeta_i \in T_{\xi_i}\mathcal{T}$. This is not restrictive since, as established in [30, Proposition 3.2], $\mathcal{P}$ can be used to define a bijection between the neighborhood of a trajectory $\xi \in \mathcal{T}$ and the origin of its tangent space $T_\xi\mathcal{T}$. The condition $\zeta_i \in T_{\xi_i}\mathcal{T}$ simply means that $\zeta_i(t) := (z_i(t), v_i(t)) \in \mathbb{R}^n \times \mathbb{R}^m$, $t \geq 0$, is a trajectory of the linearization of the control system $\dot{x} = f(x,u)$ about the current trajectory iterate ξ_i. The search direction subproblem (2) is, in practice, a linear quadratic (LQ) optimal control problem, where the functional to be minimized, $\mathbf{D}\tilde{h}(\xi_i)\cdot\zeta + \frac{1}{2}\mathbf{D}^2\tilde{h}(\xi_i)\cdot(\zeta,\zeta)$, is the quadratic model functional given by the first two terms of the Taylor expansion of the functional $\tilde{h}(\xi_i + \zeta)$ with respect to ζ [23, Section 3]. The LQ problem is defined using first and second order derivatives of the nonlinear system and the incremental and terminal costs about the current (nonlinear system) trajectory iterate. It can be solved by computing the solution to a suitable differential Riccati equation (and an associated adjoint system). In particular, in the vector space case, the usual chain rule applies and one finds that $\mathbf{D}^2\tilde{h}(\xi)\cdot(\zeta,\zeta)$ is a well defined object given by

$$\mathbf{D}^2\tilde{h}(\xi)\cdot(\zeta,\zeta) = \mathbf{D}^2 h(\xi)\cdot(\zeta,\zeta) + \mathbf{D}h(\xi)\cdot\mathbf{D}^2\mathcal{P}(\xi)\cdot(\zeta,\zeta), \qquad (5)$$

for $\xi \in \mathcal{T}$ and $\zeta \in T_\xi\mathcal{T}$ [30]. Note that $\mathbf{D}^2\mathcal{P}(\xi)$ is the second Fréchet derivative of the Banach space *operator* $\mathcal{P}$.

When the system evolves on a Lie group, a number of interesting questions arise. What is the linearization of the system? How do we define and compute a second order approximation of the system? What Riccati equation(s) can we associate with a Lie group trajectory optimization problem? One purpose of this paper is to develop appropriate notions to address these questions.

B. Notation and definitions

We assume that the reader is familiar with the theory of finite dimensional smooth manifolds, matrix Lie groups, and covariant differentiation. We refer to the books [32], [33], [34] for a review on differentiable manifolds and covariant differentiation and to [35], [36], [37] for a review of the theory of Lie groups and Lie Algebra. Many of these topics are also covered in [38] and [39].

Notation

M, N	Smooth manifolds
x	Point on the manifold
T_xM, T_x^*M	Tangent and cotangent spaces of M at x
v, w	Tangent vectors
TM, T^*M	Tangent and cotangent bundles of M
$\pi : TM \to M$	Natural bundle projection from TM to M
$X : M \to TM$	Generic vector field on M
$f : M \to N$	A map from M to N
$\mathbf{D}f : TM \to TN$	Tangent map of f
$\mathbf{D}h(\xi)\cdot\zeta$	Fréchet derivative of the functional h at ξ in the direction ζ
$\mathbf{D}^2h(\xi)\cdot(\zeta_1,\zeta_2)$	Second Fréchet derivative of h at ξ in the directions ζ_1 and ζ_2
$\varphi : M \to N$	Diffeomorphism between M and N
φ^*Y	Pull-back of the vector field Y on N through φ, i.e., $\varphi^*Y(x) := \mathbf{D}\varphi^{-1}(\varphi(x))\cdot Y(\varphi(x))$
φ_*X	Push-forward of the vector field X on M through φ, i.e., $\varphi_*X = (\varphi^{-1})^*X$
∇	Affine connection
$\nabla_X Y$	Covariant derivative of the vector field Y in the direction X
$\mathbb{D}Y\cdot X$	Covariant derivative (alternative notation)
D_t	Covariant differentiation with respect to the parameter t
$\gamma(t)$, $t \in I$	Curve (defined on the interval $I \subset \mathbb{R}$)
$P_\gamma^{t_1\leftarrow t_0}V_0$	Parallel displacement along γ, from $t = t_0$ to $t = t_1$, of the vector $V_0 \in T_{\gamma(t_0)}M$
$\mathbb{D}^2f(x)\cdot(\mathrm{v},\mathrm{w})$	Second covariant derivative of the map f at $x \in M$ evaluated in the directions v, $\mathrm{w} \in T_xM$
G	Lie group
$\mathfrak{g}$	Lie algebra of G
e	Group identity
L_gx, R_gx	Left and right translations of $x \in G$ by $g \in G$
gx, xg,	Shorthand notation for L_gx and R_gx
$g\mathrm{v}$, $\mathrm{v}g$	Shorthand notation for $\mathbf{D}L_g(x)\cdot\mathrm{v}$, $\mathbf{D}R_g(x)\cdot\mathrm{v}$, with $\mathrm{v} \in T_xM$
$[\cdot,\cdot]$	Lie bracket operation
Ad_g	Adjoint representation of G on $\mathfrak{g}$
ad_ϱ	Adjoint representation of $\mathfrak{g}$ onto itself ($\mathrm{ad}_\varrho\varsigma = [\varrho,\varsigma]$)
$\exp : \mathfrak{g} \to G$	Exponential map
$\log : G \to \mathfrak{g}$	Logarithm map (inverse of the $\exp$ in a neighborhood of e)
d	Trivialized tangent of a local diffeormorphism between $\mathfrak{g}$ and G

The (0) connection on a Lie group. On a Lie group, *left-invariant* connections are those for which $(L_g)_* \nabla_X Y = \nabla_{(L_g)_* X} (L_g)_* Y$, while *right-invariant* connections are, similarly, those which commute with the push-forward of the right translation. There is a one-to-one correspondence between left-invariant (respectively, right-invariant) affine connections on G and bilinear maps $\omega : \mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}$ [40, Theorem 8.1] given by

$$\omega(\varrho, \varsigma) = \big(\nabla_{X_\varrho} X_\varsigma\big)(e)\,, \quad \varrho, \varsigma \in \mathfrak{g} \tag{6}$$

for $X_\varrho(g) := \mathbf{D}L_g(e) \cdot \varrho$ (respectively, $:= \mathbf{D}R_g(e) \cdot \varrho$). The bilinear function ω appearing in (6) is termed the *left* (respectively, *right*) *connection function* for ∇. A connection is *bi-invariant* if it is both right- and left-invariant. For bi-invariant connections, the right and left-connection functions coincide and satisfy $\omega(\mathrm{Ad}_g \varrho, \mathrm{Ad}_g \varsigma) = \mathrm{Ad}_g \omega(\varrho, \varsigma)$, for all $g \in G$ and $\varrho, \varsigma \in \mathfrak{g}$. Given a Lie group of dimension n, an invariant connection is uniquely specified by the n^3 *numbers* that characterize the bilinear connection function.

Amongst all possible bi-invariant affine connections, three are particularly useful: they are the $(-)$, $(+)$ and (0) Cartan-Schouten connections. These connections were studied and generalized to homogeneous spaces by Nomizu in [40, Section 11], although in the context of Lie groups they were introduced by E. Cartan and J. Schouten in [41] and further developed by E. Cartan in [42]. For these connections, every 1-parameter subgroup $\gamma_\varrho(t) := \exp(t\varrho)$ is a *geodesic*, meaning that its covariant derivative satisfies $D_t \dot\gamma_\varrho(t) = 0$. The $(-)$ and $(+)$ connections are flat (i.e., the curvature tensor of the connection is identically zero), implying that the associated parallel displacement is *independent* of the path, depending only on its initial and final points.

In this paper, we only make use of the (0)-connection. Its connection function and parallel displacement satisfy, respectively,

$$\omega(\varrho, \varsigma) = \frac{1}{2}\,[\varrho, \varsigma] = \frac{1}{2}\,\mathrm{ad}_\varrho \varsigma, \tag{7}$$

$$^{(0)}P_\gamma^{t_1 \leftarrow t_0} \mathrm{v}_0 = \frac{1}{2}\big(x_1 x_0^{-1} \mathrm{v}_0 + \mathrm{v}_0 x_0^{-1} x_1\big) + o(t_1 - t_0) \tag{8}$$

where ϱ, $\varsigma \in \mathfrak{g}$, $\gamma : \mathbb{R} \mapsto G$ is a curve satisfying $\gamma(t_0) = x_0$ and $\gamma(t_1) = x_1$, and $\mathrm{v}_0 \in T_{x_0}G$. The parallel displacement $^{(0)}P$ is path dependent as the (0)-connection is not flat. The approximate expression given in the left hand side of (8) is a handy and useful formula to compute covariant differentiation. Note that $x_1 x_0^{-1} \mathrm{v}_0 = {}^{(-)}P_\gamma^{t_1 \leftarrow t_0} \mathrm{v}_0$ and $\mathrm{v}_0 x_0^{-1} x_1 = {}^{(+)}P_\gamma^{t_1 \leftarrow t_0} \mathrm{v}_0$ appearing in (8) are, respectively, the (path independent) parallel displacements of the flat $(-)$ and $(+)$ connections [40].

The trivialized tangent of a (local) diffeomorphism between $\mathfrak{g}$ and G. Let G be a Lie group with Lie algebra $\mathfrak{g}$. Consider a (local) diffeomorphism $F : \mathfrak{g} \to G$ between a neighborhood N_0 of the origin of $\mathfrak{g}$ and a neighborhood N_e of the identity of G. Given $\xi \in N_0 \subseteq \mathfrak{g}$ the *(right) trivialized tangent* of F at ξ is the linear map $\mathrm{d}F_\xi : \mathfrak{g} \to \mathfrak{g}$ defined by

$$\mathrm{d}F_\xi \eta := (\mathbf{D}F(\xi) \cdot \eta)\, F(\xi)^{-1} \tag{9}$$

Similarly, given a local diffeomorphism $H : G \to \mathfrak{g}$, the *(right) trivialized tangent* of H at ξ is the linear map $\mathrm{d}H_\xi : \mathfrak{g} \to \mathfrak{g}$ defined by

$$\mathrm{d}H_\xi \eta := \mathbf{D}H(g) \cdot \eta g\,, \tag{10}$$

with $g = H^{-1}(\xi)$. More details on the trivialized tangent and their use can be found in [17] and [43, section 4]. In this paper, we make use of the trivialized tangents of the exponential and logarithm maps, using $F(\xi) = \exp(\xi)$ and $H(g) = \log(g)$.

C. The Lie group projection operator approach

A control vector field on a Lie group G is a (sufficiently smooth) map $f : G \times \mathbb{R}^m \times \mathbb{R} \to TG$, $(g, u, t) \mapsto f(g, u, t)$, such that $\pi f(g, u, t) = g$ for each $(g, u, t) \in G \times \mathbb{R}^m \times \mathbb{R}$. A trajectory of the control system $\dot g = f(g, u, t)$ is a state-control curve $\eta(t) = (g(t), u(t)) \in G \times \mathbb{R}^m$, $t \in \mathbb{R}$, with $g(t)$ absolutely continuous and $u(t)$ integrable, satisfying a.e. the differential equation

$$\dot g(t) = f(g(t), u(t), t). \tag{11}$$

Similarly to what is done on a vector space, given a state-control curve $\xi = (\alpha(t), \mu(t)) \in G \times \mathbb{R}^m$, $t \in \mathbb{R}$, we can assign a cost to it by defining a cost functional

$$h(\xi) := \int_0^{t_f} l(\xi(\tau), \tau)\, d\tau + m(\pi_1 \xi(t_f)) \tag{12}$$

where $\pi_1 \xi(t_f) = \alpha(t_f)$ and $l : G \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}$ and $m : G \to \mathbb{R}$ are given incremental and terminal cost functions. We are interested in minimizing the functional h over the set of trajectories of f starting from a given initial condition $g_0 \in G$.

The projection operator approach on vector spaces [23], reviewed at the beginning of this section, is generalized to Lie groups as follows.

Algorithm (Lie group Projection operator Newton method)
given initial trajectory $\xi_0 \in \mathcal{T}$
for $i = 0, 1, 2, \ldots$
redesign feedback K if desired/needed
(*search direction*)

$$\zeta_i = \arg \min_{\xi_i \zeta \in T_{\xi_i} \mathcal{T}} \mathbf{D}h(\xi_i) \cdot \xi_i \zeta + \tfrac{1}{2}\, \mathbb{D}^2 \tilde h(\xi_i) \cdot (\xi_i \zeta, \xi_i \zeta) \tag{13}$$

$$\gamma_i = \arg \min_{\gamma \in (0,1]} \tilde h(\xi_i \exp(\gamma \zeta_i)) \qquad \textit{(step size)} \tag{14}$$

$$\xi_{i+1} = \mathcal{P}(\xi_i \exp(\gamma_i \zeta_i)) \qquad \textit{(update)} \tag{15}$$

end

The algorithm is closely related to the one proposed for vector spaces. In fact, when $G = \mathbb{R}^n$, it is actually equivalent to it. Note that the perturbation $\zeta_i(t) = (z_i(t), v_i(t))$, $t \in \mathbb{R}$, is now a curve in $\mathfrak{g} \times \mathbb{R}^m$ while the current iterate $\xi_i(t) = (g_i(t), u_i(t))$, $t \in \mathbb{R}$, is a trajectory in $G \times \mathbb{R}^m$. Moreover, the operation $\xi_i + \gamma_i \zeta_i$, which does not make sense on a Lie group, is replaced with the operation $\xi_i \exp(\gamma_i \zeta_i)$, where the exponential acts pointwise in time. Specifically, from now on, we adopt the following convention. Given a curve in $G \times \mathbb{R}^m$, $\xi(t) = (\alpha(t), \mu(t))$, $t \geq 0$, and a curve in $\mathfrak{g} \times \mathbb{R}^m$, $\zeta(t) = (\beta(t), \nu(t))$, $t \geq 0$, we write $\exp(\zeta)$ and $\log(\xi)$ for the pointwise operators defined by $\exp(\zeta)(t) = (\exp(\beta(t)), \nu(t)) \in G \times \mathbb{R}^m$ and

$\log(\xi)(t) = (\log(\alpha(t)), \mu(t)) \in G \times \mathbb{R}^m$, $t \geq 0$. We also adopt the notation $\xi\zeta$ to mean the curve in $T(G \times \mathbb{R}^m)$ defined as $\xi(t)\zeta(t) = (g(t)z(t), v(t))$, $t \geq 0$.

In the following sections, we define the linearization of a control system on a Lie group, the Lie group projection operator $\mathcal{P}$ and detail the search direction subproblem (13). In particular, in Section VI we show that the search direction subproblem (13) is in fact a linear-quadratic problem on the Lie algebra of G.

III. Linearization of control systems on Lie groups

Given a control vector field f on a Lie group, its *left trivialization* is the map $\lambda : G \times \mathbb{R}^m \times \mathbb{R} \to \mathfrak{g}$ defined as $\lambda(g, u, t) := g^{-1} f(g, u, t)$. The left trivialization λ allows one to write equation (11) equivalently as

$$\dot{g}(t) = g(t)\lambda(g(t), u(t), t)\,. \tag{16}$$

As we show in the following, the use of an element on the Lie algebra to uniquely represent a generic tangent vector on the Lie group is key in developing the concept of linearization along a trajectory of the control system. An equivalent theory can be obtained using right translation, the choice between the two depending on the specific application in mind.

Left-trivialized linearization around a trajectory. Let $\eta(t) = (g(t), u(t)) \in G \times \mathbb{R}^m$, $t \geq 0$, be a trajectory of the control system (16), with $g(0) = g_0$.

Definition 3.1: The *left trivialized linearization* of (16) about the state-input trajectory $\eta(t)$, $t \geq 0$, is the linear system

$$\dot{z}(t) = A(\eta(t), t)\, z(t) + B(\eta(t), t)\, v(t)\,,$$

with $z(t) \in \mathfrak{g}$ and $v(t) \in \mathbb{R}^m$ and where

$$A(\eta, t) := \mathbf{D}_1 \lambda(g, u, t) \circ \mathbf{D} L_g(e) - \mathrm{ad}_{\lambda(g,u,t)}\,, \tag{17}$$

$$B(\eta, t) := \mathbf{D}_2 \lambda(g, u, t). \tag{18}$$

In the remaining of this section, we detail in which sense (17) and (18) represent a linearization of (16). Given a bounded curve $v(t) \in \mathbb{R}^m$, $t \geq 0$, and $\varepsilon \in \mathbb{R}$ "small", consider the perturbation of the input defined as $u_\varepsilon(t) := u(t) + \varepsilon v(t)$. Indicating with g_ε the state trajectory associated with u_ε, we have

$$\dot{g}_\varepsilon(t) = g_\varepsilon(t)\lambda(g_\varepsilon(t), u_\varepsilon(t), t), \qquad g_\varepsilon(0) = g_0\,.$$

In the (possibly small) interval $[0, T_\varepsilon)$, the solution g_ε will remain in a neighborhood of the unperturbed trajectory $g(t)$, $t \geq 0$, so that we can use the exponential coordinates to parameterize neighboring trajectories of the nominal state trajectory $g(t)$. To this end, we define the *left-trivialized perturbed trajectory* $z_\varepsilon(t)$, $t \in [0, T_\varepsilon)$, such that $g_\epsilon(t) = g(t)\exp(z_\varepsilon(t))$, $t \in [0, T_\varepsilon)$. The trajectory z_ε satisfies the following differential equation.

Proposition 3.1: Let $x_\varepsilon(t) = \exp(z_\varepsilon(t))$, $t \in [0, T_\varepsilon)$. The *left trivialized perturbed trajectory* $z_\varepsilon(t)$, $t \in [0, T_\varepsilon)$, satisfies

$$\dot{z}_\varepsilon = \mathrm{d}\log_{z_\varepsilon} \Big(\mathrm{Ad}_{x_\varepsilon} \lambda\big(g x_\varepsilon, u_\varepsilon, t\big) - \lambda\big(g, u, t\big)\Big), \quad z_\varepsilon(0) = 0. \tag{19}$$

Proof: Since $g_\varepsilon(t) = g(t)x_\varepsilon(t)$ is a trajectory of (16) with input signal $u_\varepsilon(t)$, it satisfies

$$\frac{d}{dt}\,[g(t)x_\varepsilon(t)] = g(t)x_\varepsilon(t)\lambda\big(g(t)x_\varepsilon(t), u_\varepsilon(t), t\big)\,. \tag{20}$$

The left hand side of (20) is equal to $\dot{g}(t)x_\varepsilon(t) + g(t) \cdot \mathbf{D}\exp(z_\varepsilon(t)) \cdot \dot{z}_\varepsilon(t)$. Substituting this expression into (20) and multiplying both sides by $g(t)^{-1}$, we get

$$\mathbf{D}\exp(z_\varepsilon) \cdot \dot{z}_\varepsilon = \Big(\mathrm{Ad}_{x_\varepsilon}\lambda\big(g x_\varepsilon, u_\varepsilon, t\big) - \lambda\big(g, u, t\big)\Big) x_\varepsilon, \tag{21}$$

where for brevity we have dropped the explicit dependence on time. Since the inverse map of $\mathbf{D}\exp(\cdot)$ at z is $\mathbf{D}\log(\exp(z))$ and $\mathbf{D}\log(\exp(z_1)) \cdot (z_2 \exp(z_1)) = \mathrm{d}\log_{z_1} z_2$ for each z_1, $z_2 \in \mathfrak{g}$, the result follows. ■

Proposition 3.2: The left-trivialized perturbed trajectory $z_\varepsilon(t)$, $t \geq 0$, can be expanded to first order as $z_\varepsilon(t) = \varepsilon z(t) + R_2(\varepsilon, t)$, with R_2 of order higher than one in ε and $z(t)$ satisfying

$$\dot{z}(t) = A(\eta(t), t)\, z(t) + B(\eta(t), t)\, v(t), \quad z(0) = 0, \tag{22}$$

where $A(\eta(t), t)$ and $B(\eta(t), t)$ are given by (17) and (18), respectively.

Proof: The result follows from standard perturbation theory (see, e.g., [44, Chapter 8]) realizing that (19) defines a differential equation in the form $\dot{y} = F(y, \varepsilon, t)$ with initial condition $y(0) = 0$, where $F(\cdot, \cdot, \cdot)$ is smooth with respect to ε. Denoting by $y_0(t)$, $t \geq 0$, the solution of (19) for $\varepsilon = 0$, from perturbation theory we get $y_\varepsilon(t) = y_0(t) + \varepsilon z(t) + R_2(\varepsilon, t)$, where $y_0(t)$ is the solution of $\dot{y}_0 = F(y_0, 0, t)$ with $y_0(0) = 0$ and $z(t)$ satisfies

$$\begin{aligned} \dot{z}(t) &= \mathbf{D}_1 F(y_0(t), 0, t) \cdot z(t) + \mathbf{D}_2 F(y_0(t), 0, t) \cdot 1, \\ z(0) &= 0. \end{aligned} \tag{23}$$

Equation (19) is in the form $\dot{y} = F(y, \varepsilon, t) = M(y)\, c_\eta(y, \varepsilon v(t), t)$ with $M(y) = [\mathrm{d}\log_y]$ and

$$c_\eta(y, v, t) := \mathrm{Ad}_{\exp y}\lambda(g(t)\exp y, u(t) + v, t) - \lambda(g(t), u(t), t)\,.$$

Note that $M(0) = I$ and $c_\eta(0, 0, t) = 0$, $t \geq 0$. Since $y(0) = 0$, it follows that $y_0(t) \equiv 0$, $t \geq 0$. Equation (23) can be written as

$$\begin{aligned} \dot{z} &= (\mathbf{D}_1 M(0) \cdot z)\, c_\eta(0, 0, t) \\ &\quad + M(0)(\mathbf{D}_1 c_\eta(0, 0, t) \cdot z + \mathbf{D}_2 c_\eta(0, 0, t) \cdot v), \\ z(0) &= 0. \end{aligned} \tag{24}$$

Since $c_\eta(0, 0, t) \equiv 0$, $t \geq 0$, and $M(0) = I$, we only need to compute the partial derivatives of $c_\eta(\cdot, \cdot, \cdot)$ with respect to the first two arguments around $(0, 0, t)$ to compute the right hand side of (24). Since $\frac{d}{d\varepsilon}\mathrm{Ad}_{\exp(\varepsilon y)}\big|_{\varepsilon=0} = \mathrm{ad}_y$, we obtain

$$\begin{aligned} \mathbf{D}_1 c_\eta(0, 0, t) \cdot z &= \mathrm{ad}_z \lambda(g(t), u(t), t) \\ &\quad + \mathbf{D}_1 \lambda(g(t), u(t), t) \cdot g(t) z\,, \\ \mathbf{D}_2 c_\eta(0, 0, t) \cdot v &= \mathbf{D}_2 \lambda(g(t), u(t), t) \cdot v\,. \end{aligned}$$

The result follows noting that $A(\eta, t) = \mathbf{D}_1 c_\eta(0, 0, t)$ and $B(\eta, t) = \mathbf{D}_2 c_\eta(0, 0, t)$. Recall that $\mathrm{ad}_\varrho \varsigma = -\mathrm{ad}_\varsigma \varrho$, for all $\varsigma, \varrho \in \mathfrak{g}$. ■

IV. Differentiation of maps between manifolds

In this section, we define the *second covariant derivative of a map* between two manifolds that will play a key role in computing the second order approximation of the projection operator $\mathcal{P}$ on Lie groups. This covariant derivative is required to obtain a formula analogous to (5) in the context of Lie groups providing, in particular, an understanding of the second derivative of a map between two manifolds, each endowed with an affine connection. The symbol $\mathbb{D}$ is introduced to indicate this particular notion of covariant differentiation.

A. The second covariant derivative of a map

Let M_1 and M_2 be smooth manifolds endowed with affine connections ${}^1\nabla$ and ${}^2\nabla$, respectively, and let $f : M_1 \to M_2$ be a smooth map. The second covariant derivative is a tool that extends the classical (Leibniz) product rule to the covariant derivative of the "product" $\mathbf{D}f(\gamma_1(t)) \cdot V_1(t)$, where V_1 is a vector field along a curve γ_1 in M_1.

Given $x \in M_1$ and the tangent vectors v and w $\in T_xM_1$, let $\gamma_1 : I \to M_1$ be a smooth curve in M_1 such that $\gamma_1(t_0) = x$ and $\dot\gamma_1(t_0) = \mathrm{w}$. Let V_1 be a smooth vector field along γ_1 such that $V_1(t_0) = \mathrm{v}$. If follows that $V_2(t) := \mathbf{D}f(\gamma_1(t)) \cdot V_1(t) \in T_{f(\gamma_1(t))}M_2$ is a smooth vector field along the curve $\gamma_2(t) := f(\gamma_1(t))$ in M_2.

Definition 4.1: The second covariant derivative of the map $f : M_1 \to M_2$ at $x \in M_1$ in the directions v, w $\in T_xM_1$ is the bilinear map $\mathbb{D}^2 f(x) : T_xM_1 \times T_xM_1 \to T_{f(x)}M_2$ defined by

$$\mathbb{D}^2 f(x) \cdot (\mathrm{v}, \mathrm{w}) = D_t V_2(t_0) - \mathbf{D}f(\gamma_1(t_0)) \cdot D_t V_1(t_0) \qquad (25)$$

where D_tV_1 and D_tV_2 denote covariant differentiation with respect to ${}^1\nabla$ and ${}^2\nabla$, respectively.

Proposition 4.1: Denote by ${}^1P$ and ${}^2P$ the parallel displacements associated with ${}^1\nabla$ and ${}^2\nabla$, respectively. Then, equation (25) is equivalent (for $t = t_0$) to

$$\lim_{\varepsilon \to 0} \frac{1}{\varepsilon}\Big({}^2P_{\gamma_2}^{t \leftarrow t+\varepsilon}\, \mathbf{D}f(\gamma_1(t+\varepsilon)) \cdot {}^1P_{\gamma_1}^{t+\varepsilon \leftarrow t} V_1(t) - \mathbf{D}f(\gamma_1(t)) \cdot V_1(t)\Big). \qquad (26)$$

Proof: The connection ${}^2\nabla$ allows us to compute the covariant derivative of the vector field V_2 along γ_2 as

$$\big(D_t V_2\big)(t) = \lim_{\varepsilon \to 0} \frac{1}{\varepsilon}\left({}^2P_{\gamma_2}^{t \leftarrow t+\varepsilon} V_2(t+\varepsilon) - V_2(t)\right). \qquad (27)$$

The right hand side of equation (27) can be expanded into

$$\lim_{\varepsilon \to 0} \frac{1}{\varepsilon}\Big({}^2P_{\gamma_2}^{t \leftarrow t+\varepsilon} \mathbf{D}f(\gamma_1(t+\varepsilon)) \cdot V_1(t+\varepsilon) - \mathbf{D}f(\gamma_1(t)) \cdot V_1(t)\Big). \qquad (28)$$

Adding and subtracting the term ${}^2P_{\gamma_2}^{t \leftarrow t+\varepsilon}\mathbf{D}f(\gamma_1(t+\varepsilon)) \cdot {}^1P_{\gamma_1}^{t+\varepsilon \leftarrow t} V_1(t)$ inside the parenthesis of the previous expression, and noting that (in TM_2)

$$\lim_{\varepsilon \to 0} \frac{1}{\varepsilon} {}^2P_{\gamma_2}^{t \leftarrow t+\varepsilon}\mathbf{D}f(\gamma_1(t+\varepsilon)) \cdot \big(V_1(t+\varepsilon) - {}^1P_{\gamma_1}^{t+\varepsilon \leftarrow t} V_1(t)\big) = Df(\gamma_1(t)) \cdot D_t V_1(t),$$

the result follows. ∎

Remark 4.1: One can define higher order covariant derivatives of a map ($\mathbb{D}^3 f$, $\mathbb{D}^4 f$, and so on) by requiring that Leibniz's rule holds. Moreover, the symbol $\mathbb{D}$ can be used, e.g., to indicate the covariant derivative of a vector field Y in the direction X, i.e., $\mathbb{D}X \cdot Y$ (whose standard notation is $\nabla_X Y$) as well as the covariant differentiation of the "product" $\mathbf{D}f \cdot X$, with $f : M \to N$ a map and X a vector field over M. Note that covariant differentiation is defined in such a way that Liebnitz's rule holds, so that one obtains, e.g., the identity

$$\mathbb{D}(\mathbf{D}f \cdot X) \cdot Y = \mathbb{D}^2 f \cdot (X, Y) + \mathbf{D}f \cdot (\mathbb{D}X \cdot Y),$$

where X and Y are vector fields over M and $\mathbb{D}^2 f$ is the second covariant derivative of the map f. The vector field X and the tangent map $\mathbf{D}f$ are, in fact, special cases of *two-point tensor fields*, namely a $\left(\begin{smallmatrix}1 & 0\\ 0 & 0\end{smallmatrix}\right)$-tensor field and a $\left(\begin{smallmatrix}0 & 1\\ 1 & 0\end{smallmatrix}\right)$-tensor field, respectively [45]. Two-point tensor fields (sometimes also called double tensor fields) and their covariant derivatives are not commonly encountered in standard differential and Riemannian geometry textbooks and, to our understanding, are mostly encountered in the context of continuum mechanics, quantum physics, and advanced differential geometry applications. They are the natural generalization of vector fields and one forms over maps. From now on, the operator $\mathbb{D}$ will be used to indicate covariant differentiation of a generic two-point tensor field [45].

Remark 4.2: The key role played by covariant differentiation in the context of this paper can be understood through a finite dimensional analogy. Let M and N be differentiable manifolds, each endowed with an affine connection, and let $r : \mathbb{R}^m \to M$, $q : M \to N$, $p : N \to \mathbb{R}^n$ be three given differentiable functions, with m and $n \in \mathbb{N}$ arbitrary. The composition $p \circ q \circ r : \mathbb{R}^n \to \mathbb{R}^m$ is a differentiable function that can be expanded about a given point. In the context of this work, p may be thought of as the cost functional, q as the projection operator, and r as the pointwise exponential operator. Covariant differentiation allows us to use Leibniz's rule to express the second order term using intrinsically defined derivatives of p, q, and r. Indeed, one obtains

$$\begin{aligned} p(q(r(x+z))) = {} & p(q(r(x))) + \mathbf{D}p \cdot \mathbf{D}q \cdot \mathbf{D}r \cdot z \\ & + 1/2 \big(\mathbb{D}^2 p \cdot (\mathbf{D}q \cdot \mathbf{D}r \cdot z, \mathbf{D}q \cdot \mathbf{D}r \cdot z) \\ & + \mathbf{D}p \cdot \mathbb{D}^2 q \cdot (\mathbf{D}r \cdot z, \mathbf{D}r \cdot z) \\ & + \mathbf{D}p \cdot \mathbf{D}q \cdot \mathbb{D}^2 r \cdot (z, z)\big) + o(|z|^2). \end{aligned} \qquad (29)$$

It is this abstract and high level splitting of the second order term into its "elementary" parts that allows us to analyze these parts separately and obtain explicit and computable formulas for use in the projection operator approach.

Due to limited space, we will not present explicit formulas for the covariant derivative of a two-point tensor field. We refer the reader to [45] and references therein for further reading.

In the next subsection, we provide some useful covariant derivatives of maps and vector fields defined on Lie groups that will be used in the derivation of the Lie group projection operator approach. The subsection can be skipped on the first reading of the paper.

B. The covariant derivative on Lie groups: Differentiation rules for the (0) connection

The second and higher order covariant derivatives of a map between two Lie groups can be computed as soon as we specify affine connections on domain and codomain. In this subsection, we restrict our attention to second covariant derivatives with respect to the (0)-connection because this connection is used for computing the second order approximation of the projection operator in Section V-D. The Lie algebra $\mathfrak{g}$, being a vector space, is endowed with the trivial affine connection (the parallel displacement along any curve is the identity map). The following results may be verified by straightforward computations.

Proposition 4.2: Let $X(g) := g\xi(g)$ and $Y(g) = \xi(g)g$ be vector fields on G, where $g \mapsto \xi(g)$ is a differentiable $\mathfrak{g}$-valued function. Then

$$\begin{aligned} \mathbb{D}X(g)\cdot g\eta &= g\big(\mathbf{D}\xi(g)\cdot g\eta + 1/2\,[\eta,\xi(g)]\big) \\ \mathbb{D}Y(g)\cdot \eta g &= \big(\mathbf{D}\xi(g)\cdot \eta g + 1/2\,[\xi(g),\eta]\big)g\,. \end{aligned} \tag{30}$$

Proposition 4.3: Let $G \ni g \mapsto \xi(g) \in \mathfrak{g}$ be defined as $g^{-1}X(g) = \mathbf{D}L_{g^{-1}}(g)\cdot X(g)$, where $X(g)$ is a vector field. Then

$$\mathbf{D}\xi(g)\cdot g\eta = g^{-1}\Big(\mathbb{D}X(g)\cdot g\eta\Big) + 1/2\,\big[g^{-1}X(g),\eta\big]\,.$$

Proposition 4.4: For each $\varrho, \varsigma \in \mathfrak{g}$, we have

$$\mathbb{D}^2\exp(0)\cdot(\varrho,\varsigma) = 0, \qquad \text{and} \qquad \mathbb{D}^2\log(e)\cdot(\varrho,\varsigma) = 0.$$

Proposition 4.5: Let $t \mapsto V(t)$ be a vector field along the curve $\gamma \subset G$, $g \in G$ a constant, and $W(t) := g\,V(t)$ a vector field along the curve $g\gamma$. Then,

$$D_t W(t) = g\,D_t V(t)\,.$$

V. The projection operator on Lie groups

In this section we define the projection operator for a dynamical system evolving on a Lie group. The standard projection operator for a nonlinear system evolving on a vector space, introduced in [30], was reviewed in Section II-A.

A. The projection operator

Let $f : G \times \mathbb{R}^m \times \mathbb{R} \to TG$ be a control vector field on G. A state-input trajectory $\xi(t) = (\alpha(t), \mu(t))$, $t \geq 0$, is called *exponentially stabilizable* if (and only if) there is a feedback law $u(t) = k(g(t), \alpha(t), \mu(t), t)$, with $k(\alpha(t), \alpha(t), \mu(t), t) = \mu(t)$ for all $t \geq 0$, such that α is an exponentially stable (state) trajectory of the closed loop system

$$\dot g(t) = f\big(g(t), k\big(g(t), \alpha(t), \mu(t), t\big), t\big)\,, \tag{31}$$

that is, there exist $M < \infty$, $\lambda > 0$, and $\delta > 0$ such that

$$\|\log(g(t)^{-1}\alpha(t))\| \leq M\,e^{-\lambda(t-t_0)}\|\log(g(t_0)^{-1}\alpha(t_0))\|$$

for all $t \geq t_0 \geq 0$ and all $g(t_0)$ in a neighborhood of $\alpha(t_0)$ such that $\|\log(g(t_0)^{-1}\alpha(t_0))\| < \delta$.

In the following, we would also impose some smoothness and boundedness conditions on k and restrict, without loss of generality, our attention to a feedback of the form

$$\begin{aligned} u(t) &= k(g(t), \alpha(t), \mu(t), t) \\ &= \mu(t) + K(t)\log(g(t)^{-1}\alpha(t)), \end{aligned} \tag{32}$$

since, as for any control system on $\mathbb{R}^n$, a trajectory ξ of a C^1 nonlinear system is exponentially stabilizable if and only if there is a bounded gain matrix K that stabilizes the linearization of f about ξ. Note that $K(t)$ is a linear map from $\mathfrak{g}$ to $\mathbb{R}^m$. It will be evident from next section that the linearization of the closed loop system (31) with feedback (32) around a state trajectory α is given by the linear differential equation

$$\dot z(t) = [A(\xi(t),t) - B(\xi(t),t)K(t)]z(t), \tag{33}$$

with A and B defined by (17), (18).

*Definition 5.1 (**Projection operator** $\mathcal{P}$):* Equation (31) with the initial condition $g(0) = g_0$ and feedback (32) defines a causal operator, called the *projection operator*, which maps a state-input curve $\xi(t) = (\alpha(t), \mu(t)) \in G \times \mathbb{R}^m$, $t \geq 0$, into the state-input trajectory $\eta(t) = (g(t), u(t)) \in G \times \mathbb{R}^m$, $t \geq 0$, that satisfies

$$\dot g = g\lambda_K(g, \xi(t), t)\,, \tag{34}$$

$$u(t) = u_K(g, \xi(t), t)\,, \tag{35}$$

$$g(0) = g_0\,, \tag{36}$$

where

$$\lambda_K(g,\xi,t) := \lambda(g, u_K(g,\xi,t), t)\,, \tag{37}$$

$$u_K(g,\xi,t) := \mu + K(t)\log(g^{-1}\alpha), \tag{38}$$

and $\lambda : G \times \mathbb{R}^m \times \mathbb{R} \to \mathfrak{g}$ denotes the left trivialization of the control system $\dot g = f(g,u,t)$. In short, we write $\eta = \mathcal{P}_K^{g_0}(\xi)$ or, when g_0 and K are clear from the context, simply $\eta = \mathcal{P}(\xi)$. As in the vector space case, the projection operator satisfies the projection property $\mathcal{P}(\xi) = \mathcal{P}(\mathcal{P}(\xi)) =: \mathcal{P}^2(\xi)$.

B. The local projection operator and its properties

We are interested in studying the effect of a perturbation of the curve ξ in the direction ζ, that is, we study the map $\mathcal{P}(\xi\exp(\varepsilon\zeta))$, for $\varepsilon \in \mathbb{R}$ "small". We can parameterize $\mathcal{P}(\xi\exp(\varepsilon\zeta))$ using the left-trivialized perturbed trajectory $\chi_\varepsilon(t) \in \mathfrak{g} \times \mathbb{R}^m$, $t \geq 0$, defined by

$$\mathcal{P}(\xi\exp(\varepsilon\zeta)) = \mathcal{P}(\xi)\exp(\chi_\varepsilon)\,. \tag{39}$$

*Definition 5.2 (**The local projection operator** $\mathcal{N}_\xi$):* The left-trivialized *local projection operator* around the curve ξ, written as $\chi = \mathcal{N}_\xi(\zeta)$, is the operator that takes the curve $\zeta(t) = (\beta(t), \nu(t)) \in \mathfrak{g} \times \mathbb{R}^m$, $t \geq 0$, to the left-trivialized trajectory $\chi(t) = (y(t), w(t)) \in \mathfrak{g} \times \mathbb{R}^m$, $t \geq 0$, given by

$$\chi = \log(\mathcal{P}(\xi)^{-1}\mathcal{P}(\xi\exp(\zeta))) =: \mathcal{N}_\xi(\zeta). \tag{40}$$

Proposition 5.1: Given curves $\xi = (\alpha, \mu)$ and $\eta = (g, u)$ with $\eta = P(\xi)$, the map $(y_\varepsilon, w_\varepsilon) = \chi_\varepsilon = \mathcal{N}_\xi(\varepsilon\zeta) = \mathcal{N}_\xi(\varepsilon\beta, \varepsilon\nu)$ can be computed explicitly by using

$$\dot{y}_\varepsilon = \mathrm{d}\log_{y_\varepsilon}\left[\mathrm{Ad}_{\exp y_\varepsilon}\lambda_K(g\exp y_\varepsilon, \xi\exp(\varepsilon\zeta), t) - \lambda_K(g,\xi,t)\right], \tag{41}$$

$$w_\varepsilon(t) = u_K(g\exp y_\varepsilon, \xi\exp(\varepsilon\zeta), t) - u_K(g,\xi,t), \tag{42}$$

$$y_\varepsilon(0) = 0, \tag{43}$$

where Ad is the adjoint representation of the group G on its Lie algebra and $\mathrm{d}\log$ denotes the trivialized tangent of $\log$ as defined in (10).

Proof: By definition, $\mathcal{P}(\xi\exp(\varepsilon\zeta)) = \mathcal{P}(\xi)\exp(\mathcal{N}_\xi(\varepsilon\zeta))$. Since

$$\big(\mathcal{P}(\xi)\exp(\mathcal{N}_\xi(\varepsilon\zeta))\big)(t) = (g(t)\exp y_\varepsilon(t), u(t) + w_\varepsilon(t)),$$

with $t \geq 0$, is a trajectory of the closed loop control system (34)-(36), it satisfies

$$\frac{d}{dt}(g\exp y_\varepsilon)(t) = g\exp y_\varepsilon \lambda_K(g\exp y_\varepsilon, \xi\exp(\varepsilon\zeta), t), \tag{44}$$

$$w_\varepsilon(t) = u_K(g\exp y_\varepsilon, \xi\exp(\varepsilon\zeta), t) - u_K(g,\xi,t), \tag{45}$$

$$g(0)\,\exp(y_\varepsilon(0)) = g_0. \tag{46}$$

It is now clear that (42) and (43) follow immediately from (45) and (46). One can also conclude that (44) implies (41) by mimicking what was done in the proof of Proposition 3.1. ■

Differentiating the local projection operator $\mathcal{N}_\xi$, defined in (40), in the direction ζ_1 and evaluating it at $\zeta \equiv 0$, we obtain

$$\mathbf{D}\mathcal{N}_\xi(0)\cdot\zeta_1 = \mathcal{P}(\xi)^{-1}\mathbf{D}\mathcal{P}(\xi)\cdot\xi\zeta_1$$

and, by differentiating it twice and evaluating it at $\zeta \equiv 0$,

$$\begin{aligned}&\mathbf{D}^2\mathcal{N}_\xi(0)\cdot(\zeta_1,\zeta_2) = \\ &\mathcal{P}(\xi)^{-1}\left(\mathbb{D}\mathcal{P}^2(\xi)\cdot(\xi\zeta_1,\xi\zeta_2) + \mathbf{D}\mathcal{P}(\xi)\cdot\xi\mathbb{D}^2\exp(0)\cdot(\zeta_1,\zeta_2)\right) \\ &+ \mathbb{D}^2\log(e)\cdot(\mathbf{D}\mathcal{N}_\xi(0)\cdot\zeta_1, \mathbf{D}\mathcal{N}_\xi(0)\cdot\zeta_2).\end{aligned}$$

As mentioned in Section IV-B, using the (0) connection we have $\mathbb{D}^2\exp(0)\cdot(\zeta_1,\zeta_2) = 0$ and $\mathbb{D}^2\log(e)\cdot(\zeta_1,\zeta_2) = 0$. Therefore, we obtain the following result.

Proposition 5.2: The first and second covariant derivatives with respect to the (0) connection of the projection operator $\mathcal{P}$ satisfy

$$\mathbf{D}\mathcal{N}_\xi(0)\cdot\zeta_1 = \mathcal{P}(\xi)^{-1}\mathbf{D}\mathcal{P}(\xi)\cdot\xi\zeta_1 \tag{47}$$

and

$$\mathbf{D}^2\mathcal{N}_\xi(0)\cdot(\zeta_1,\zeta_2) = \mathcal{P}(\xi)^{-1}\mathbb{D}^2\mathcal{P}(\xi)\cdot(\xi\zeta_1,\xi\zeta_2). \tag{48}$$

Note that we write $\mathbf{D}^2\mathcal{N}_\xi$ instead of $\mathbb{D}^2\mathcal{N}_\xi$ to highlight the fact that $\mathcal{N}_\xi$ is an operator between two vector spaces. The next two subsections detail how (47) and (48) can be computed.

C. The first derivative of the projection operator

The following proposition provides the explicit expressions for computing the first derivative of the projection operator $\mathcal{P}$. Its proof is based on perturbation theory and uses the same arguments as in the proof of Proposition 3.2.

Proposition 5.3: The left-trivialized trajectory $\chi_\varepsilon = \mathcal{N}_\xi(\varepsilon\zeta)$ can be expanded to first order as $\chi_\varepsilon(t) = \varepsilon\gamma(t) + R_2(\varepsilon, t)$ with R_2 of order higher than one in ε. The curve $\gamma(t) = (z(t), v(t))$, $t \geq 0$, satisfies

$$\gamma = \mathbf{D}\mathcal{N}_\xi(0)\cdot\zeta = \mathcal{P}(\xi)^{-1}\mathbf{D}\mathcal{P}(\xi)\cdot\xi\zeta \tag{49}$$

and can be computed using

$$\dot{z} = A(\eta(t),t)\,z + B(\eta(t),t)\,v, \tag{50}$$

$$v = \nu + K(t)\mathrm{d}\log_{\log(g^{-1}\alpha)}(\mathrm{Ad}_{g^{-1}\alpha}\beta - z), \tag{51}$$

$$z(0) = 0,$$

where $A(\eta(t),t)$ and $B(\eta(t),t)$ are given by (17) and (18). Note that, when $\xi = \mathcal{P}(\xi)$, (51) is simply equal to $v = \nu + K(t)(\beta - z)$.

D. The second covariant derivative of the projection operator

Recall the definition of the (left-trivialized) local projection operator $\mathcal{N}_\xi$ given in (40). The proof of the following key result is developed in Appendix A.

Theorem 5.4: Given a trajectory $\eta = (g,u)$, $\eta = \mathcal{P}(\eta)$, the second derivative of $\mathcal{N}_\eta$ about zero in the directions ζ_1 and ζ_2, namely

$$\begin{aligned}(y,w) &= \mathbf{D}^2\mathcal{N}_{(g,u)}(0)\cdot((\beta_1,\nu_1),(\beta_2,\nu_2)) \\ &= \mathbf{D}^2\mathcal{N}_\eta(0)\cdot(\zeta_1,\zeta_2) = \mathcal{P}(\eta)^{-1}\mathbb{D}^2\mathcal{P}(\eta)\cdot(\eta\zeta_1,\eta\zeta_2),\end{aligned}$$

is given by

$$\begin{aligned}\dot{y} &= A(\eta,t)y + B(\eta,t)w \\ &\quad + 1/2\left(\mathrm{ad}_{z_1}\mathbf{D}\lambda_t(\eta)\cdot\eta\gamma_2 + \mathrm{ad}_{z_2}\mathbf{D}\lambda_t(\eta)\cdot\eta\gamma_1\right) \\ &\quad + \mathbb{D}^2\lambda_t(\eta)\cdot(\eta\gamma_1,\eta\gamma_2),\end{aligned} \tag{52}$$

$$w = -K(t)\big[y + 1/2\left([z_1,\beta_2] + [z_2,\beta_1]\right)\big], \tag{53}$$

with $y(0) = 0$, $\gamma_i = (z_i, v_i) = D\mathcal{N}_\eta(0)\cdot\zeta_i$, $i = 1, 2$, and where $\lambda_t(\eta) := \lambda(\eta, t)$ and $A(\eta,t)$ and $B(\eta,t)$ are defined as in (17) and (18), respectively.

Note that for brevity we have suppressed the argument t in the expressions (52) and (53). Equations (52) and (53) generalize to Lie groups the second derivative of the projection operator given in [30, subsection 1.3]. Also, when $\xi\zeta_i \in T_\eta\mathcal{T}$, that is $\gamma_i = \mathbf{D}\mathcal{N}_\eta(0)\cdot\zeta_i = \zeta_i$, $i = \{1,2\}$, equation (53) reduces to $w = -K(t)y$.

VI. The search direction subproblem in detail

The search direction subproblem (13) requires the minimization of the functional $\mathbf{D}h(\xi)\cdot\xi\zeta + \frac{1}{2}\mathbb{D}^2\tilde{h}(\xi)\cdot(\xi\zeta,\xi\zeta)$ over the Banach space $T_\xi\mathcal{T}$. Leveraging on the results obtained in the previous sections, we detail how this functional can be constructed and minimized.

Proposition 6.1: Let $\mathbf{e}_i$, $i = 1,\dots,n+m$, be a basis for $\mathfrak{g}\times\mathbb{R}^m$, so that each $(z,v)\in\mathfrak{g}\times\mathbb{R}^m$ can be uniquely written as $(z,v) = z^1\mathbf{e}_1 + \cdots + z^n\mathbf{e}_n + v^1\mathbf{e}_{n+1} + \cdots + v^m\mathbf{e}_{n+m}$.

Given the trajectory $\xi(t) = (g(t), u(t)) \in G \times \mathbb{R}^m$, $t \in [0,T]$, of (11), the search direction step (13) is equivalent to solving the optimal control problem

$$\min_{(z,v)(\cdot)} \int_0^{t_f} a(\tau)^T z(\tau) + b(\tau)^T v(\tau) + \frac{1}{2}\begin{bmatrix} z(\tau) \\ v(\tau) \end{bmatrix}^T W(\tau) \begin{bmatrix} z(\tau) \\ v(\tau) \end{bmatrix} d\tau + r_1^T z(t_f) + \frac{1}{2} z(t_f)^T P_1 z(t_f)\,, \tag{54}$$

subject to the dynamic constraint

$$\dot z(t) = A(\xi(t),t) z(t) + B(\xi(t),t) v(t), \quad z(0) = 0, \tag{55}$$

with $z(t) \in \mathfrak{g}$ and $v(t) \in \mathbb{R}^m$. In the above linear-quadratic problem, $A(\xi,t)$ and $B(\xi,t)$ are given, respectively, by (17) and (18), while $a(t)$, $b(t)$, r_1, and P_1 satisfy

$$\langle a(t), z\rangle = \mathbf{D}_1 l(g(t), u(t), t) \cdot g(t) z, \tag{56}$$
$$\langle b(t), v\rangle = \mathbf{D}_2 l(g(t), u(t), t) \cdot v, \tag{57}$$
$$\langle r_1, z\rangle = \mathbf{D} m(g(t_f)) \cdot g(t_f) z, \tag{58}$$
$$\langle P_1 z_2, z_1\rangle = \mathbb{D}^2 m(g(t_f)) \cdot (g(t_f) z_1, g(t_f) z_2). \tag{59}$$

The matrix $W(t)$, appearing in (54), is the symmetric $(n+m)\times(n+m)$ matrix with elements

$$w_{ij}(t) = l_{ij}(\xi(t),t) + \langle p(t), \lambda_{ij}(\xi(t))\rangle \tag{60}$$

where $p(t) \in \mathfrak{g}^*$, the *adjoint state*, satisfies (64) below, while $l_{ij}(\xi,t) \in \mathbb{R}$ and $\lambda_{ij}(\xi) \in \mathfrak{g}$ are

$$l_{ij}(\xi,t) := \mathbb{D}^2 l_t(\xi) \cdot (\xi \mathbf{e}_i, \xi \mathbf{e}_j) \tag{61}$$
$$\lambda_{ij}(\xi) := \mathbb{D}^2 \lambda_t(\xi) \cdot (\xi \mathbf{e}_i, \xi \mathbf{e}_j) \tag{62}$$
$$+ 1/2 \left(\mathrm{ad}_{\varpi_1(\mathbf{e}_i)} \mathbf{D}\lambda_t(\xi) \cdot \xi \mathbf{e}_j + \mathrm{ad}_{\varpi_1(\mathbf{e}_j)} \mathbf{D}\lambda_t(\xi) \cdot \xi \mathbf{e}_i\right), \tag{63}$$

where $\varpi_1 : \mathfrak{g} \times \mathbb{R}^m \to \mathfrak{g}$, $\varpi_1(z,v) = z$, $\lambda_t(\xi) := \lambda(\xi,t)$, and $l_t(\xi) := l(\xi,t)$. The adjoint state $p(t) \in \mathfrak{g}^*$ satisfies

$$-\dot p(t) = A_{cl}(t)^T p(t) + a(t) - K(t)^T b(t), \quad p(T) = r_1, \tag{64}$$

with $A_{cl}(t) = A(\xi(t),t) - B(\xi(t),t)\, K(t)$.

Proof: Using the projection operator $\mathcal{P}$ defined in (34)-(38), the functional $\tilde h$ over the space of curves in $G \times \mathbb{R}^m$ is constructed as

$$\tilde h(\xi) = h(\mathcal{P}(\xi)), \tag{65}$$

with h defined as in (12). To construct the projection operator based optimization algorithm we need to find a quadratic approximation of $\tilde h$ around a given curve ξ. To this end, for a given curve $\xi(t) \in G \times \mathbb{R}^m$, $t \geq 0$ and perturbation $\zeta(t) \in \mathfrak{g} \times \mathbb{R}^m$, $t \geq 0$, we expand with respect to ε the expression

$$\tilde h(\xi \exp \varepsilon\zeta) = h\big(\mathcal{P}(\xi \exp \varepsilon\zeta)\big). \tag{66}$$

Note that the above expression, as a function of ε and for fixed ξ and ζ, defines a real function on $\mathbb{R}$. Using Leibnitz's rule and the identities $\mathbf{D}\exp(0)\cdot\zeta = \zeta$ and $\mathbb{D}^2\exp(0) = 0$, one obtains

$$h(\mathcal{P}(\xi\exp(\varepsilon\zeta))) = h(\mathcal{P}(\xi)) + \varepsilon\, \mathbf{D}h(\mathcal{P}(\xi)) \cdot \mathbf{D}\mathcal{P}(\xi)\cdot\xi\zeta + (1/2)\,\varepsilon^2 \left[\mathbb{D}^2 h(\mathcal{P}(\xi)) \cdot (\mathbf{D}\mathcal{P}(\xi)\cdot\xi\zeta, \mathbf{D}\mathcal{P}(\xi)\cdot\xi\zeta) + \mathbf{D}h(\mathcal{P}(\xi)) \cdot \mathbb{D}^2\mathcal{P}(\xi)\cdot(\xi\zeta,\xi\zeta)\right] + o(\varepsilon^2) \tag{67}$$

where the first covariant derivative of h is

$$\mathbf{D}h(\xi)\cdot\xi\zeta = \int_0^{t_f} \mathbf{D}l(\xi(\tau),\tau)\cdot\xi(\tau)\zeta(\tau)\, d\tau + \mathbf{D}m(\pi_1\xi(t_f)) \cdot \mathbf{D}\pi_1(\xi(t_f))\cdot\xi(t_f)\zeta(t_f) \tag{68}$$

and $\mathbb{D}^2 h(\xi)\cdot(\xi\zeta_1, \xi\zeta_2)$ equals

$$\int_0^{t_f} \mathbb{D}^2 l(\xi(\tau),\tau)\cdot(\xi(\tau)\zeta_1(\tau), \xi(\tau)\zeta_2(\tau))\, d\tau + \mathbb{D}^2 m(\pi_1\xi_f) \cdot \big(\mathbf{D}\pi_1(\xi_f)\cdot\xi_f\zeta_1(t_f), \mathbf{D}\pi_1(\xi_f)\cdot\xi_f\zeta_2(t_f)\big)$$

where $\xi_f := \xi(t_f)$ and $\mathbf{D}\pi_1(\xi(t))\cdot\xi(t)\zeta(t) = \alpha(t)\beta(t)$.

The expressions for the first and second (covariant) derivatives of the projection operator $\mathcal{P}$, appearing in (67), have been presented previously in Proposition 5.3 and Theorem 5.4.

Recalling (67), assuming $\xi \in \mathcal{T}$ and $\xi\zeta \in T_\xi\mathcal{T}$, one gets

$$\mathbb{D}^2 \tilde h(\xi)\cdot(\xi\zeta,\xi\zeta) = \mathbb{D}^2 h(\xi)\cdot(\xi\zeta,\xi\zeta) + \mathbf{D}h(\xi)\cdot\mathbb{D}^2\mathcal{P}(\xi)\cdot(\xi\zeta,\xi\zeta)\,. \tag{69}$$

The result follows by mimicking the proof of Proposition 3.2 in [23], replacing the expressions for the second derivative of the projection on vector spaces with those given in (52)-(53) and noting that $\mathbf{D}h(\xi)\cdot\xi\zeta = \int_0^{t_f} a(\tau)^T z(\tau) + b(\tau)^T v(\tau)\, d\tau + r_1^T z(t_f)$. ■

The linear quadratic optimal control problem appearing in Proposition 6.1 can be solved by standard techniques (see, e.g., [6]). The optimal control solution is given in form of a time-varying *affine* state feedback obtained by solving a linear and a Riccati differential equation backward in time.

Indirect methods and the projection operator approach. Let $\hat H^-(g,p,u,t) := l(g,u,t) + \langle p, \lambda(g,u,t)\rangle$ be the left-trivialized pre-Hamiltonian which is naturally associated to the optimal control problem of interest. Recall that the necessary conditions for optimality of the (left-trivialized) Pontryagin Maximum Principle are

$$g^{-1}\dot g = \frac{\partial \hat H}{\partial p}^- (g,p,u^*(g,p,t),t) \tag{70}$$

$$\dot p = \mathrm{ad}^*_{g^{-1}\dot g} p - (\mathbf{D}L_g(e))^* \frac{\partial \hat H}{\partial g}^- (g,p,u^*(g,p,t),t) \tag{71}$$

$$u^*(g,p,t) = \arg\min_u \hat H^-(g,p,u,t)\,, \tag{72}$$

with boundary conditions $g(0) = g_0$ and $p(T) = r_1$ (see, e.g., [14, Chapter 12, Corollary 1]). The following proposition shows in which sense the projection operator based Newton method is related to indirect methods for solving optimal control problems, by linking (64) with the adjoint equation (71).

Proposition 6.2: Equation (64) is a stabilized version of the adjoint equation (71). The two equations coincides when $\xi(t) = (g(t), u(t))$ satisfies the first order optimality conditions.

Proof: In (71), $(\mathbf{D}L_g(e))^* : T_g^*G \mapsto \mathfrak{g}^*$ is the dual map of the linear operator $\mathbf{D}L_g(e) : \mathfrak{g} \mapsto T_gG$. Recalling the definition of $A(\xi(t),t)$ and $a(t)$, it is straightforward to verify that (71) equals $-\dot p = A^T(\xi(t),t)\, p + a(t)$. Note that (64), instead, is equal to $-\dot p = A^T(\xi(t),t)p +$

$a(t) - K^T(t)(b(t) + B(\xi(t),t)^T p)$. The necessary condition (72) implies $\partial \hat{H}^-(g,p,u^*(g,p),t)/\partial u = 0$, i.e., $b^T(t) + p^T(t)B(\xi(t),t) = 0$. Therefore, approaching a (local) optimal solution, $p(t)$ in (64) converges to the solution of (71), since $b^T(t) + p^T(t)B(\xi(t),t)$ will converge to zero. Note the stabilization (backward in time) of equation (64) due to the presence of the feedback K. ■

VII. A WORKED EXAMPLE

This section presents numerical results obtained by using the algorithm detailed in Section II-C to solve an optimal control problem on SO(3). The problem considered is one of the simplest examples of an optimal control problem for a system evolving on a *nonabelian* Lie group and it is a generalization of the classical linear quadratic regulator (LQR) problem on vector spaces to the group of rotation matrices SO(3) [46]. Its relative simplicity allows us to give details for the linear quadratic optimal control problem (54)-(55), providing explicit formulas for the matrices A, B, W, and P_1 and vectors a, b, and r_1. Furthermore, the computations indicate that the algorithm provides, as known for the flat case, second order convergence to a (local SSC) minimum.

Let $\|M\|_P$ denote the weighted *Frobenius* matrix norm defined as $\sqrt{\mathrm{tr}(M^T P M)}$, with $M, P \in \mathbb{R}^{3\times 3}$ and $P = P^T > 0$,

Let $\big(g_d(t), u_d(t)\big) \in \mathrm{SO}(3) \times \mathbb{R}^3$, $t \in [0,T]$, be a desired state-control curve (i.e., not necessarily a system trajectory). Let $\bar{Q}$, $\bar{R}$, and $\bar{P}_f \in \mathbb{R}^{3\times 3}$ be symmetric positive define matrices and g_0 and g_f two elements of $SO(3)$. We define the *hat* operator $\wedge : \mathbb{R}^3 \mapsto \mathbb{R}^{3\times 3}$ as the Lie algebra isomorphism

$$\mathbb{R}^3 \ni \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = x \mapsto \hat{x} = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \in \mathfrak{so}(3)\,. \tag{73}$$

The goal is to find a trajectory $(g(t), u(t)) \in \mathrm{SO}(3) \times \mathbb{R}^3$, $t \in [0,T]$, satisfying the *dynamic constraint*

$$\dot{g}(t) = g(t)\hat{u}(t), \qquad g(0) = g_0, \tag{74}$$

that *minimizes*

$$\int_0^{t_f} l(g,u,\tau)\, d\tau + m(g(t_f))\,, \tag{75}$$

with $l(g,u,\tau) := (1/2)\,\|e - g_d^{-1}(\tau)\,g\|^2_{\bar{Q}} + (1/2)\,\|u - u_d(\tau)\|^2_{\bar{R}}$ and $m(g) := (1/2)\,\|e - g_f^{-1} g\|^2_{\bar{P}_f}$ being, respectively, the incremental and terminal costs.

Since (74) is already in the left-trivialized form (16) with $\lambda(g,u,t) = u$, given a trajectory $\xi(t) = (g(t), u(t))$, $t \in [0,T]$, its left-trivialized linearization is

$$\begin{aligned} A(\xi(t),t) &:= \mathbf{D}_1\lambda(g(t),u(t),t) \circ \mathbf{D}L_{g(t)}(e) - \mathrm{ad}_{\lambda(g(t),u(t),t)} \\ &= -\mathrm{ad}_{u(t)} = -\hat{u}(t)\,, \end{aligned} \tag{76}$$

$$B(\xi(t),t) := \mathbf{D}_2\lambda(g(t),u(t),t) = I\,. \tag{77}$$

The expression for the vectors $a(t)$, $b(t)$ and r_1 are

$$a(t)^T z = \mathbf{D}_1 l(g(t),u(t),t) \cdot g(t)\hat{z} = -\mathrm{tr}(\bar{Q} g_d^T(t) g(t)\hat{z})\,, \tag{78}$$

$$b(t)^T v = \mathbf{D}_2 l(g(t),u(t),t) \cdot v = (u(t) - u_d(t))^T \bar{R} v\,, \tag{79}$$

$$r_1^T z = \mathbf{D}_1 m(g(T)) \cdot g(T)\hat{z} = -\mathrm{tr}(\bar{P}_f {g_f}^T g(T)\hat{z})\,. \tag{80}$$

The matrices $W(t)$ and P_1 can be computed once the second covariant derivative of the function

$$F(g) := \frac{1}{2}\|e - g_1^{-1} g\|^2_{\bar{P}}\,, \tag{81}$$

with $g_1 \in \mathrm{SO}(3)$ and $\bar{P} = \bar{P}^T > 0$, is known. Note how the function $F(g)$ appears in the expressions of the incremental and terminal costs. The first and second covariant derivatives of $F(g)$ are given by

$$\mathbf{D}F(g) \cdot gz = -\mathrm{tr}(\bar{P} g_1^T g\hat{z})\,, \tag{82}$$

$$\mathbb{D}^2 F(g) \cdot (gz_1, gz_2) = -\mathrm{tr}\left(\bar{P} g_1^T g \frac{\hat{z}_2\hat{z}_1 + \hat{z}_1\hat{z}_2}{2}\right). \tag{83}$$

In principle, one could obtain the vector and matrix representations of the above derivatives by using the identities $\mathrm{tr}(\hat{x}^T A) = x^T(A - A^T)^\vee$ and $\mathrm{tr}(\hat{x}^T A\hat{y}) = y^T((\mathrm{tr}A)I - A)x$, valid for each $x, y \in \mathbb{R}^3$ and $A \in \mathbb{R}^{3\times 3}$ (the *vee* operator $^\vee$ is just the inverse of the *hat* operator $^\wedge$ defined in (73)). However, we found that simpler and more elegant expressions for these derivatives can be obtained when parametrizing SO(3) by unit quaternions. Define the matrix P according to the transformation

$$P = (\mathrm{tr}\bar{P})I - \bar{P}\,, \tag{84}$$

with inverse

$$\bar{P} = \mathrm{tr}P\,(1/2)\,I - P\,, \tag{85}$$

and let $q \in \mathbb{R}^4$ be one of the two unit quaternions corresponding to the rotation matrix $g_1^{-1}g$ in (81) above. Let $q_s \in \mathbb{R}$ and $q_v \in \mathbb{R}^3$ denote, respectively, the scalar and vector parts of the unit quaternion $q = (q_s; q_v)$, where ; denotes row concatenation. Remarkably, the following identity holds

$$F(g) = \frac{1}{2}\,(2q_v)^T P (2q_v)\,. \tag{86}$$

Note that the formula is, as it must be in order to be a function defined on SO(3), invariant under the antipodal symmetry $(q_s, q_v) \mapsto (-q_s, -q_v)$. From (86), one may then obtain

$$\mathbf{D}F(g) \cdot g\hat{z} = 2\, q_v^T P (q_s I + \hat{q}_v) z\,, \tag{87}$$

$$\begin{aligned} &\mathbb{D}^2 F(g) \cdot (g\hat{z}_1, g\hat{z}_2) = \\ &\qquad z_2^T((q_s I + \hat{q}_v)^T P (q_s I + \hat{q}_v) - (q_v^T P q_v) I) z_1\,. \end{aligned} \tag{88}$$

Due to limited space, we do not provide a proof of these formulas. They can be easily checked numerically against the equivalent expressions (82) and (83).

Equations (87) and (88) provide immediately the vector and matrix representations that we need to compute the matrices $W(t)$, $t \in [0,T]$, and P_1. Define $Q = Q^T > 0$ from $\bar{Q}$ according to (84). Using (60), $W(t)$ results in

$$\begin{bmatrix} (q_s I + \hat{q}_v)^T Q (q_s I + \hat{q}_v) - (q_v^T Q q_v) I & -1/2\hat{p}(t) \\ 1/2\hat{p}(t) & \bar{R} \end{bmatrix}, \tag{89}$$

where $q = (q_s, q_v^T)^T$ is the unit quaternion representation of $g_d(t)^{-1}g(t)$.

Equation (89) has been obtained as follow. Recall the definitions of l_{ij} and λ_{ij} given in (61) and (63), respectively, and let $\zeta_{1,k}$, $\zeta_{2,k} \in \mathbb{R}$, $k = 1, \ldots, n+m$ be the components of $\zeta_1 = (z_1, v_1)$, $\zeta_2 = (z_2, v_2) \in \mathfrak{g} \times \mathbb{R}^m$ with respect

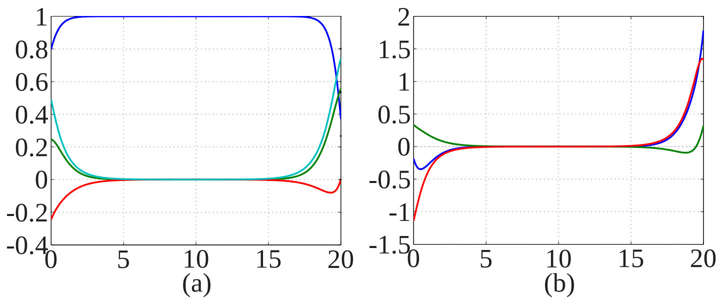

Fig. 2. Optimal state-control trajectory. Parts (a) and (b) show the optimal state and control trajectories versus time. The state is represented using unit quaternions.

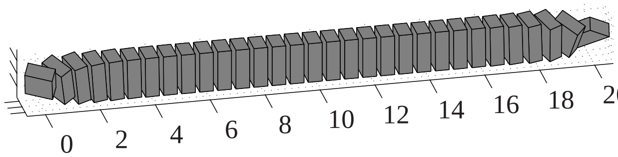

Fig. 3. Graphical representation of the optimal state trajectory. The plot shows the optimal attitude matrix $g(t) \in \mathrm{SO}(3)$, $t \in [0, 20]$, using a rectangular box that is centered at $(t, 0, 0)$ and rotated by $g(t)$. Thirty snapshots are shown.

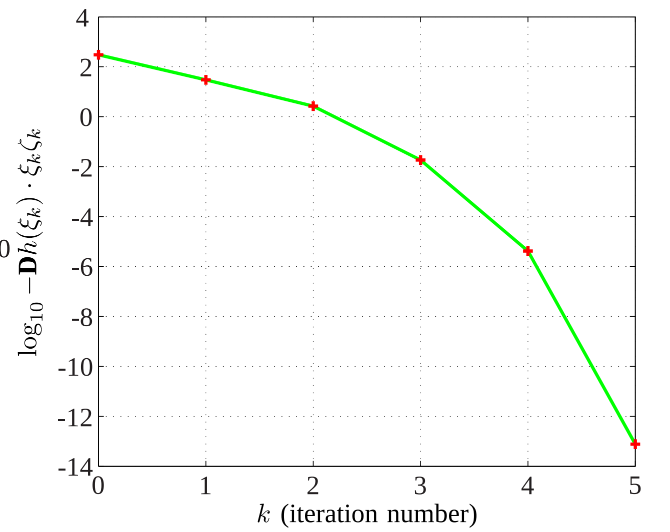

Fig. 4. Quadratic convergence rate. The plot shows $\log_{10} -\mathbf{D}h(\xi_k) \cdot \xi_k\zeta_k$ as a function of the number of iterations.

the basis $\mathbf{e}_k$, $k = 1, \ldots, n+m$. The diagonal entries of $W(t)$ in (89) are derived from the matrix representation of $\mathbb{D}^2 l(\xi(t), t) \cdot (\xi(t)\zeta_1, \xi(t)\zeta_2) = l_{ij}(\xi(t), t)\zeta_{1,i}\zeta_{2,j}$ which is obtained, concerning the state part, from (88). The off diagonal terms are obtained computing $p_k(t)\lambda^k_{ij}(\xi(t), t)\zeta_{1,i}\zeta_{2,j}$ which, since $\mathbb{D}^2\lambda(\xi) \equiv 0$, is equal to $\langle p(t), 1/2\left(\mathrm{ad}_{z_1} v_2 + \mathrm{ad}_{z_2} v_1\right)\rangle$. Finally, $P_1 = (q_s I + \hat{q}_v)^T P_f (q_s I + \hat{q}_v) - (q_v^T P_f q_v) I$, with (q_s, q_v) the unit quaternion representation of $g_f^{-1} g(T)$.

In Figure 2, we show the optimal solution obtained by applying the proposed descent algorithm detailed in Section II-C to the problem (74)-(75). To provide a visual representation, the optimal solution $g(t)$, $t \in [0, 20]$ is represented in Figure 3 using a rectangular box. The width and height of the box (corresponding to the y and z body axes, respectively) are, respectively, two and three times the depth (the x body axis). For each t, the box is centered at the point $(t, 0, 0)$ and thirty snapshots (three every two seconds) are shown.

The following set of parameters was chosen. The initial condition g_0 is the rotation matrix corresponding to the unit quaternion $[0.7986\,, 0.2457\,, -0.2457\,, 0.4914]^T$. The final time is $t_f = 20$s. The desired trajectory $\xi_d(t) = (g_d(t), u_d(t))$, $t \in [0, t_f]$, appearing in the incremental cost (75), is the trivial trajectory identically equal to $(e, 0)$ for each $t \in [0, t_f]$. The weighting matrix $\bar{Q}$ is equal to $\bar{Q} = (1/2\,\mathrm{tr}\,Q)I - Q$, the inverse of the transformation (84), with $Q = \mathrm{diag}(2\,, 5\,, 3)$. The weighting matrix $\bar{R}$ is equal to $\mathrm{diag}(1\,, 6\,, 3)$. The rotation matrix g_f in the terminal cost (75) corresponds to the unit quaternion $q_f = [0.2673\,, 0.5345\,, 0\,, 0.8018]^T$, while the weighting matrix $\bar{P}_f$ is obtained from $P_f = \mathrm{diag}(20, 20, 20)$, in the same way $\bar{Q}$ is obtained from Q.

The initial trajectory $\xi(t) = (g(t), u(t))$, $t \in [0, t_f]$, is the constant trajectory $(g_0, 0)$, $t \in [0, t_f]$. At each iteration, the projection operator feedback $K(t)$, $t \in [0, t_f]$, is designed by solving a time-varying LQR problem (with diagonal weighting matrices, both equal to the identity) about the current trajectory iterate. The differential equations required at each iteration of the algorithm are solved numerically using the `ode45` solver in Mathworks Matlab/Simulink, storing all the trajectories with a sampling period of 0.005s. The absolute and relative tolerances of the ODE solver are set to 10^{-14} and 10^{-7}, respectively. The termination condition is $-\mathbf{D}h(\xi_k) \cdot \xi_k\zeta_k \leq 10^{-8}$. The algorithm takes about 3 seconds to solve this problem on a laptop equipped with a Intel Core 2 Duo CPU P8600 2.40 GHz. The algorithm is coded as an m-file script which calls a series of S-functions written in `C` for integrating the differential equations.

Figure 4 shows that the algorithm takes only 5 iterations to converge. In the first iteration, the backtracking line search reduced the step size to $\gamma_0 = 0.7^4 \approx 0.24$ (using $c = 0.4$ and $\alpha = 0.7$ in the notation of Algorithm 3.1 in [47]) as the local quadratic model of the functional does not approximate the cost functional very well over long steps on this *curved* manifold. However, beginning with the second iteration, full Newton steps are taken ($\gamma_k = 1$) and Figure 4 provides an indication of quadratic rate of convergence to the locally minimizing trajectory ξ^*. Indeed, since $-\mathbf{D}h(\xi_k) \cdot \xi_k\zeta_k = \mathbb{D}^2\tilde{h}(\xi_k) \cdot (\xi_k\zeta_k, \xi_k\zeta_k)$ is a scaled L_2 norm (squared) of ζ_k, we see that the Newton "step" ζ_k for this problem converges to zero in L_2 with a quadratic rate. While this does not ensure that the error $\log(\xi_k^{-1}\xi^*)$ converges to zero in L_∞ with quadratic rate, we know that if it does then so must ζ_k (in L_∞ and hence in L_2). An examination of $-\mathbf{D}h(\xi_k) \cdot \xi_k\zeta_k$ versus k may thus be used to rule out quadratic convergence. In finite dimensions, the size of the Newton step provides a direct indication of the size of the error [48] making a plot such as that in Figure 4 especially useful. The Banach space projection operator is known to provide quadratic convergence to local SSC minimizers [23, Section 5], providing further support for such convergence in the Lie group case.

VIII. Conclusions

In this paper, we have extended the projection operator based trajectory optimization approach to the class of nonlinear systems that evolve on Lie groups. This has required the introduction of a covariant derivative notion for the repeated differentiation of a map between two Lie groups, endowed with affine connections. With this tool, chain rule like formulas have been used to develop the expressions for the basic objects needed for trajectory optimization. The resulting algorithm requires one to solve, at each iteration, a time-varying linear quadratic optimal control problem associated with the current trajectory.

A numerical example on the Lie group $\mathrm{SO}(3)$ has been presented, highlighting implementation details. Computational results indicate a second order convergence rate for this problem. Second order convergence to a local SSC minimizer is well known for Newton's method in finite dimensions and has also been shown to hold for the Banach space projection operator approach [23]—we believe that this result continues to hold in the Lie groups setting although a formal proof has not yet been worked out in detail. The numerical example presented provides useful formulas that can also be used to solve trajectory optimization problems for mechanical systems whose configuration manifold is $\mathrm{SE}(3)$, e.g., for trajectory planning and parameter identification of unmanned aerial vehicles or underwater autonomous vehicles [49]. Preliminary tests have shown that, with respect to the standard projection operator approach, the Lie group version of the projection operator approach can have computational advantages in solving the same optimization problem (solved, with the standard approach, using a set of local coordinates). We suspect that this is related to the absence, in the Lie group version of the algorithm, of the double differentiation of the functions that describe the attitude matrix in terms of a set of local parameters (e.g., Euler angles).

For Lie groups for which a known closed formulas for the exponential and logarithm maps are not available or hard to compute, we expect that the use of approximations (such as, e.g., the Cayley map on $\mathrm{SO}(n)$) that agree with the first and second covariant derivatives of those mappings at the origin of $\mathfrak{g}$ and at the identity of G, respectively, will be effective and will maintain the second order convergence rate of the algorithm.

The choice of the (0)-connection for defining the second covariant derivative of a map between Lie groups has been motivated by the observation that the obtained formulas are somehow simpler than the ones resulting by choosing different connections, mainly because in this case $\mathbb{D}^2\exp(0)$ and $\mathbb{D}^2\log(e)$ are zero. In *finite* dimension optimization on Lie groups, it has been shown [19] that a Newton like algorithm defined using any of the Cartan-Schouten connections displays the local quadratic convergence characteristic of Newton algorithms. The rate of convergence is not affected by the choice of the connection as the geometric Hessian at a critical point (for a smooth function, the geometric Hessian is equivalent to the second covariant derivative discussed in our work) is always the same, independently of the choice of the connection. In those algorithms, however, using a connection different than the (0) connection does not correspond to minimizing a truncated Taylor expansion of the original cost function.

Further investigations are required to clarify all these issues and to fully explore the strengths and weaknesses of the proposed Lie group method.

As a final remark, we would like to emphasize that the tangent bundle of a Lie group G *is* itself a Lie group. This means that the method developed in this paper is directly applicable, e.g., to the optimal control of mechanical systems evolving on Lie groups [10], either holonomic or nonholonomic.

Acknowledgements

The authors would like to express their heart-felt appreciation to the anonymous reviewers for the valuable comments and suggestions on how to improve the quality and presentation of the paper. The authors are also grateful to Hans Z. Munthe-Kaas for fruitful discussion on covariant differentiation of two-point tensors, and to Franco Cardin, Rui Loja Fernandes, Mario Micheli, and Jochen Trumpf for valuable help on understanding some subtleties of parallel displacement on Lie groups.

Appendix A
Proof of Theorem 5.4

Obtaining the expression for $\mathbb{D}^2\mathcal{P}(\xi)\cdot(\xi\zeta_1,\xi\zeta_2) = \mathcal{P}(\xi)\,\mathbf{D}^2\mathcal{N}_\xi(0)\cdot(\zeta_1,\zeta_2)$ given in Theorem 5.4 is not trivial and involves tedious computations. The proof of Theorem 5.4 is given in this appendix, while Appendix B contains most of the technical details. The second derivative $\mathbf{D}^2\mathcal{N}_\xi(0)\cdot(\zeta_1,\zeta_2)$ can be computed by differentiating $\zeta_0\mapsto\mathbf{D}\mathcal{N}_\xi(\zeta_0)\cdot\zeta_1$ around $\zeta_0\equiv 0$ in the direction ζ_2. We begin this task by computing $\mathbf{D}\mathcal{N}_\xi(\zeta_0)\cdot\zeta_1$.

Proposition A.1: The first derivative of $\mathcal{N}_\xi$ at ζ_0 along ζ_1

$$(y_1,w_1)=\mathbf{D}\mathcal{N}_{(\alpha,\mu)}(\beta_0,\nu_0)\cdot(\beta_1,\nu_1)=\mathbf{D}\mathcal{N}_\xi(\zeta_0)\cdot\zeta_1,$$

can be computed as

$$\begin{aligned}\dot y_1 &= \mathbb{D}^2\log(\exp(z_{(1,0)}))\cdot\Big(X^K_{(1,0)},\mathbf{D}\exp(z_{(1,0)})\cdot y_1\Big)\\ &\quad+\mathbf{D}\log(\exp(z_{(1,0)}))\cdot D_\tau X^K_{(1,0)}\,, \qquad (90)\\ w_1 &= D_\tau U^K_{(1,0)}\,, \qquad (91)\end{aligned}$$

where $(z_{(1,0)},v_{(1,0)}) = \mathcal{N}_\xi(\zeta_0)$ and $X^K_{(\sigma,0)}(t)$, $D_\tau X^K_{(\sigma,0)}(t)$, and $D_\tau U^K_{(\sigma,0)}(t)$ are defined as in (100), (102), and (103) in Appendix B, respectively.

Proof: We compute $\mathbf{D}\mathcal{N}_\xi(\zeta_0)\cdot\zeta_1$ taking the limit $\lim_{\varepsilon\to 0}1/\varepsilon\left[\mathcal{N}_\xi(\zeta_0+\varepsilon\zeta_1)-\mathcal{N}_\xi(\zeta_0)\right]$. From Proposition 5.1, it follows that $(z_{(1,\varepsilon)},v_{(1,\varepsilon)})=\mathcal{N}_\xi(\zeta_0+\varepsilon\zeta_1)$ can be computed as

$$\begin{aligned}\dot z_{(1,\varepsilon)} &= \mathrm{d}\log_{z_{(1,\varepsilon)}}\!\big(\mathrm{Ad}_{\exp z_{(1,\varepsilon)}}\lambda_K(g\exp z_{(1,\varepsilon)},\xi\exp(\zeta_0+\varepsilon\zeta_1),t)\\ &\qquad -\lambda_K(g,\xi,t)\big)\,,\\ v_{(1,\varepsilon)} &= u_K(g\exp(z_{(1,\varepsilon)}),\xi\exp(\zeta_0+\varepsilon\zeta_1),t)-u_K(g,\xi,t)\,.\end{aligned}$$

Defining $x_{(1,\varepsilon)}(t) = \exp\big(z_{(1,\varepsilon)}(t)\big)$, the previous two equations can be written as

$$\dot z_{(1,\varepsilon)}(t) = \mathbf{D}\log(x_{(1,\varepsilon)}(t)) \cdot X^K_{(1,\varepsilon)}(t), \tag{92}$$

$$v_{(1,\varepsilon)}(t) = U^K_{(1,\varepsilon)}(t), \tag{93}$$

with $X^K_{(\cdot,\cdot)}$ and $U^K_{(\cdot,\cdot)}$ defined as in (98) and (99), respectively. For small ε, $(z_{(1,\varepsilon)}, v_{(1,\varepsilon)}) = (z_{(1,0)}, v_{(1,0)}) + \varepsilon(y_1, w_1) + o(\varepsilon)$. Thus,

$$\begin{aligned}\mathbf{D}\log(x_{(1,\varepsilon)}) \cdot X^K_{(1,\varepsilon)} &= \mathbf{D}\log(x_{(1,0)}) \cdot X^K_{(1,0)} \\ &+\varepsilon\Big[\mathbb{D}^2\log(x_{(1,0)}) \cdot \big(X^K_{(1,0)}, x'_{(1,0)}\big) + \mathbf{D}\log(x_{(1,0)}) \cdot D_\tau X^K_{(1,0)}\Big] \\ &+ o(\varepsilon)\end{aligned} \tag{94}$$

$$U^K_{(1,\varepsilon)} = U^K_{(1,0)} + \varepsilon\, D_\tau U^K_{(1,0)} + o(\varepsilon) \tag{95}$$

where $x'_{(1,0)}(t) := \partial/\partial\tau\, \exp(z_{(1,\tau)})(t)|_{\tau=0}$. The result follows. ■

Remark A.1: Recall that $\mathbf{D}\mathcal{N}_\xi(0)\zeta_1$ was computed in Proposition 5.3. For the case $\zeta_0 \equiv 0$, one can show that (90) and (91) simplify to

$$\dot y_1 = D_\tau X^K_{(1,0)}\,, \qquad w_1 = D_\tau U^K_{(1,0)}\,, \qquad y(0) = 0\,, \tag{96}$$

which coincide with (50) and (51).

Proposition A.2 (Left-trivialized second derivative of $\mathcal{P}$): The second derivative of $\mathcal{N}_\xi$ at $\zeta_0 \equiv 0$ evaluated in the directions ζ_1 and ζ_2,

$$\begin{aligned}(y, w) &= \mathbf{D}^2\mathcal{N}_{(\alpha,\mu)}(0) \cdot ((\beta_1, \nu_1), (\beta_2, \nu_2)) \\ &= \mathbf{D}^2\mathcal{N}_\xi(0) \cdot (\zeta_1, \zeta_2) = \mathcal{P}(\xi)^{-1}\mathbb{D}^2\mathcal{P}(\xi) \cdot (\xi\zeta_1, \xi\zeta_2)\,,\end{aligned}$$

can be computed as

$$\dot y = D_\sigma D_\tau X^K_{(0,0)}\,, \quad w = D_\sigma D_\tau U^K_{(0,0)}\,, \quad y(0) = 0\,, \tag{97}$$

for $D_\sigma D_\tau X^K_{(0,0)}$ and $D_\sigma D_\tau U^K_{(0,0)}$ defined in (104) and (105).

Proof: This is a straightforward application of the differentiation rule for the covariant derivative to the result of Proposition A.1. During the derivation the term $\mathbb{D}^3\log$ shows up but, since it is a linear operator and one of its argument is $X^K_{(0,0)} \equiv 0$, it does not appear in the final expression. ■

Finally, Theorem 5.4 can be proven using the results contained in Proposition A.2 for the special case $(g, u) = \eta = \xi = \mathcal{P}(\xi) = \mathcal{P}(\alpha, \mu)$. Again, the computations are straightforward but tedious. On the contrary, the resulting expressions are elegant and closely related to the vector space ones. In particular, one finds that the first and second derivatives of u_K are $\mathbf{D}_1 u_K(g,\eta,t)\cdot gz = -K(t)z$, $\mathbf{D}_2 u_K(g,\eta,t)\cdot\eta\zeta = \nu + K(t)\beta$, $\mathbb{D}^2_1 u_K(g,\eta,t)\cdot(gz_1, gz_2) = 0$, $\mathbf{D}_{1,2}u_K(g,\eta,t)\cdot(gz,\eta\zeta) = 1/2\,K(t)\,[z,\beta]$, $\mathbf{D}_{2,1}u_K(g,\eta,t)\cdot(\eta\zeta, gz) = 1/2\,K(t)\,[z,\beta]$, and $\mathbb{D}^2_2 u_K(g,\eta,t)\cdot(\eta\zeta_1,\eta\zeta_2) = 0$.

APPENDIX B
TECHNICAL DETAILS: $X^K_{(\sigma,\tau)}$ AND $U^K_{(\sigma,\tau)}$

This appendix contains a series of technical results which are used for computing the second covariant derivative of the projection operator $\mathcal{P}$. In the following, the Greek letters σ and τ are used to indicate "small" quantities, much as we have used ε thus far.

Definition B.1: Let the curve $\gamma_{(\sigma,\tau)}(t) \in \mathfrak{g}\times\mathbb{R}^m$, $t \geq 0$, be defined as $\gamma_{(\sigma,\tau)} = \big(z_{(\sigma,\tau)}, v_{(\sigma,\tau)}\big) := \mathcal{N}_\xi(\sigma\zeta_0 + \tau\zeta_1)$. and $x_{(\sigma,\tau)}(t) := \exp\big(z_{(\sigma,\tau)}(t)\big)$, $t \geq 0$.

Definition B.2: Define $X^K_{(\sigma,\tau)}(t) \in T_{x_{(\sigma,\tau)}(t)}G$, $t \in \mathbb{R}$, as the vector field along the curve $x_{(\sigma,\tau)}$ such that

$$\begin{aligned}X^K_{(\sigma,\tau)} &:= x_{(\sigma,\tau)}\lambda_K\big(g x_{(\sigma,\tau)}, \xi\exp(\sigma\zeta_0 + \tau\zeta_1), t\big) \\ &\quad - \lambda_K(g,\xi,t)\, x_{(\sigma,\tau)}\,,\end{aligned} \tag{98}$$

and $U^K_{(\sigma,\tau)}(t) \in \mathbb{R}^m$, $t \in \mathbb{R}$, the curve

$$\begin{aligned}U^K_{(\sigma,\tau)} &:= u_K\big(g\,x_{(\sigma,\tau)}, \xi\exp(\sigma\zeta_0 + \tau\zeta_1), t\big) \\ &\quad - u_K\big(g,\xi,t\big)\,.\end{aligned} \tag{99}$$

Note that

$$\begin{aligned}X^K_{(\sigma,0)} &= x_{(\sigma,0)}\lambda_K(g x_{(\sigma,0)}, \xi\exp(\sigma\zeta_0), t) \\ &\quad - \lambda_K(g,\xi,t)\, x_{(\sigma,0)},\end{aligned} \tag{100}$$

$$\begin{aligned}U^K_{(\sigma,0)} &= u_K\big(g\,x_{(\sigma,0)}, \xi\exp(\sigma\zeta_0)), t\big) \\ &\quad - u_K\big(g,\xi,t\big),\end{aligned} \tag{101}$$

Lemma B.1: Let $(y_\sigma, w_\sigma) := \mathbf{D}\mathcal{N}_\xi(\sigma\zeta_0)\cdot\zeta_1$. Then, the following holds.

$$\begin{aligned}&D_\tau X^K_{(\sigma,0)} = \\ &x_{(\sigma,0)}\big(\mathbf{D}_1\lambda_K(g\,x_{(\sigma,0)}, \xi\exp(\sigma\zeta_0), t)\cdot g\mathbf{D}\exp(z_{(\sigma,0)})\cdot y_\sigma \\ &+\mathbf{D}_2\lambda_K(g\,x_{(\sigma,0)}, \xi\exp(\sigma\zeta_0), t)\cdot \xi\mathbf{D}\exp(\zeta_{(\sigma,0)})\cdot\zeta_1\big) \\ &- 1/2\big(\big[\lambda_K\big(g,\xi,t\big)\,,\,\mathrm{d}\exp_{z_{(\sigma,0)}} y_\sigma\big] \\ &+\big[\mathrm{Ad}_{x_{(\sigma,0)}}\lambda_K\big(g x_{(\sigma,0)}, \xi\exp(\sigma\zeta_0), t\big), \mathrm{d}\exp_{z_{(\sigma,0)}} y_\sigma\big]\big)x_{(\sigma,0)}\,,\end{aligned} \tag{102}$$

$$\begin{aligned}&D_\tau U^K_{(\sigma,0)} = \\ &\mathbf{D}_1 u_K(g x_{(\sigma,0)}, \xi\exp(\sigma\zeta_0), t)\cdot g\mathbf{D}\exp(z_{(\sigma,0)})\cdot y_\sigma \\ &+ \mathbf{D}_2 u_K(g x_{(\sigma,0)}, \xi\exp(\sigma\zeta_0), t)\cdot\xi\mathbf{D}\exp(\zeta_{(\sigma,0)})\cdot\zeta_1\,,\end{aligned} \tag{103}$$

Lemma B.2: Let now $\zeta_0 = \zeta_2$ and define $(y, w) := \mathbf{D}^2\mathcal{N}_\xi(0)\cdot(\zeta_1,\zeta_2)$, $(z_1, v_1) := \mathbf{D}\mathcal{N}_\xi(0)\cdot\zeta_1$, and $(z_2, v_2) := \mathbf{D}\mathcal{N}_\xi(0)\cdot\zeta_2$. We have

$$\begin{aligned}D_\sigma D_\tau X^K_{(0,0)} &= \Big(\mathbf{D}_1\lambda_K(g,\xi,t)\circ\mathbf{D}L_g(e) - \mathrm{ad}_{\lambda_K(g,\xi,t)}\Big)y \\ &+ \mathbb{D}^2_1\lambda_K(g,\xi,t)\cdot(gz_1, gz_2) + \mathbf{D}_{1,2}\lambda_K(g,\xi,t)\cdot(gz_1,\xi\zeta_2) \\ &+ \mathbf{D}_{2,1}\lambda_K(g,\xi,t)\cdot(\xi\zeta_1, gz_2) + \mathbb{D}^2_{2,2}\lambda_K(g,\xi,t)\cdot(\xi\zeta_1,\xi\zeta_2) \\ &+ 1/2\,\mathrm{ad}_{z_2}\big(\mathbf{D}_1\lambda_K(g,\xi,t)\cdot gz_1 + \mathbf{D}_2\lambda_K(g,\xi,t)\cdot\xi\zeta_1\big) \\ &+ 1/2\,\mathrm{ad}_{z_1}\big(\mathbf{D}_1\lambda_K(g,\xi,t)\cdot gz_2 + \mathbf{D}_2\lambda_K(g,\xi,t)\cdot\xi\zeta_2\big)\,,\end{aligned} \tag{104}$$

$$\begin{aligned}D_\sigma D_\tau U^K_{(0,0)} &= \mathbf{D}_1 u_K(g,\xi,t)\cdot gy \\ &+ \mathbb{D}^2_1 u_K(g,\xi,t)\cdot(gz_1, gz_2) + \mathbf{D}_{1,2}u_K(g,\xi,t)\cdot(gz_1,\xi\zeta_2) \\ &+ \mathbf{D}_{2,1}u_K(g,\xi,t)\cdot(\xi\zeta_1, gz_2) + \mathbb{D}^2_2 u_K(g,\xi,t)\cdot(\xi\zeta_1,\xi\zeta_2)\,.\end{aligned} \tag{105}$$

Remark B.1: The proofs of the two previous lemmas are obtained applying the classical differentiation rules of the covariant derivative and the specific differentiation rules described in Section IV-B for the (0) connection. Note that $\partial/\partial\tau\, x_{(\sigma,\tau)}|_{\tau=0} = \mathbf{D}\exp(z_{(\sigma,0)})\cdot y_\sigma$. Also, one uses that fact that for $\varrho, \varsigma \in \mathfrak{g}$, we have $d/d\varepsilon \, (\mathbf{D}\exp(\varepsilon\varsigma)\varrho) \exp(\varepsilon\varsigma)^{-1}|_{\varepsilon=0} = 1/2\,[\varsigma, \varrho]$, $d/d\varepsilon \, \exp(\varepsilon\varsigma)^{-1}(\mathbf{D}\exp(\varepsilon\varsigma)\cdot\varrho)|_{\varepsilon=0} = 1/2\,[\varrho, \varsigma]$. Given the curves $g(\varepsilon) \in G$ and $\xi(\varepsilon) \in \mathfrak{g}$, $\varepsilon \in \mathbb{R}$, one also has $d/d\varepsilon \, \mathrm{Ad}_{g(\varepsilon)}\xi(\varepsilon) = \mathrm{Ad}_{g(\varepsilon)}\big(\xi'(\varepsilon) + [g(\varepsilon)^{-1}g'(\varepsilon), \xi(\varepsilon)]\big)$ with $\xi'(\varepsilon) := d/d\varepsilon\, \xi(\varepsilon)$ and $g'(\varepsilon) := d/d\varepsilon\, g(\varepsilon)$.

First and second derivatives of $u_K(g,\xi,t)$.

Proposition B.3: The first and second covariant derivatives relative to the (0) Cartan-Schouten and Euclidean connections of the *closed loop feedback* u_K, defined by (38), with respect to the first and second arguments g and ξ are

$$\mathbf{D}_1 u_K(g,\xi,t)\cdot gz = -K(t)\,\mathbf{D}\log(g^{-1}\alpha)\cdot(zg^{-1}\alpha)\,, \tag{106}$$

$$\mathbf{D}_2 u_K(g,\xi,t)\cdot \xi\zeta = \nu + K(t)\,\mathbf{D}\log(g^{-1}\alpha)\cdot(g^{-1}\alpha\beta)\,, \tag{107}$$

and

$$\mathbb{D}_1^2 u_K(g,\xi,t)\cdot(gz_1,gz_2) = -K(t)\,\mathbb{D}^2\log(g^{-1}\alpha)\cdot(z_1\,g^{-1}\alpha, z_2\,g^{-1}\alpha) \tag{108}$$

$$\mathbf{D}_{1,2} u_K(g,\xi,t)\cdot(gz,\xi\zeta) = -K(t)\Big(\mathbb{D}^2\log(g^{-1}\alpha)\cdot(z\,g^{-1}\alpha, g^{-1}\alpha\beta\,) + 1/2\,\mathbf{D}\log(g^{-1}\alpha)\cdot\big[z, \mathrm{Ad}_{g^{-1}\alpha}\beta\big]g^{-1}\alpha\Big)\,, \tag{109}$$

$$\mathbf{D}_{2,1} u_K(g,\xi,t)\cdot(\xi\zeta,gz) = -K(t)\Big(\mathbb{D}^2\log(g^{-1}\alpha)\cdot(g^{-1}\alpha\beta, z\,g^{-1}\alpha) + 1/2\,\mathbf{D}\log(g^{-1}\alpha)\cdot\big[z, \mathrm{Ad}_{g^{-1}\alpha}\beta\big]g^{-1}\alpha\Big)\,, \tag{110}$$

$$\mathbb{D}_2^2 u_K(g,\xi,t)\cdot(\xi\zeta_1,\xi\zeta_2) = K(t)\,\mathbb{D}^2\log(g^{-1}\alpha)\cdot(g^{-1}\alpha\beta_1, g^{-1}\alpha\beta_2)\,. \tag{111}$$

Proof: This is a straightforward application of the differentiation rules. Note, in particular, that for $f_1(\alpha) := zg^{-1}\alpha$ and $f_2(g) := g^{-1}\alpha\beta$ we have $\mathbb{D}f_1(\alpha)\cdot\alpha\beta = 1/2\,\big[z, \mathrm{Ad}_{g^{-1}\alpha}\beta\big]g^{-1}\alpha$, and $\mathbb{D}f_2(g)\cdot gz = 1/2\,\big[\mathrm{Ad}_{g^{-1}\alpha}\beta, z\big]g^{-1}\alpha$. The functions f_1 and f_2 derive from the differentiation of $g \mapsto g^{-1}\alpha$ and $\alpha \mapsto g^{-1}\alpha$ in the directions gz and $\alpha\beta$. ■

First and second derivatives of $\lambda_K(g,\xi,t)$. The proof of the following proposition follows from differentiation rules.

Proposition B.4: The first derivative of the $\lambda_K(g,\xi,t) = \lambda(g,u_K(g,\xi,t),t)$ is

$$\begin{aligned}\mathbf{D}_1\lambda_K(g,\xi,t)\cdot gz &= \mathbf{D}_1\lambda(g,u_K(g,\xi,t),t)\cdot gz \\ &\quad + \mathbf{D}_2\lambda(g,u_K(g,\xi,t),t)\cdot\mathbf{D}_1 u_K(g,\xi,t)\cdot gz\,, \\ \mathbf{D}_2\lambda_K(g,\xi,t)\cdot\xi\zeta &= \mathbf{D}_2\lambda(g,u_K(g,\xi,t),t)\cdot\mathbf{D}_2 u_K(g,\xi,t)\cdot\xi\zeta\,.\end{aligned}$$

Proposition B.5: The second covariant derivative of the (left trivialized) projection operator vector field $\lambda_K(g,\xi,t) = \lambda(g,u_K(g,\xi,t),t)$ is

$$\begin{aligned}&\mathbb{D}_1^2\lambda_K(g,\xi,t)\cdot(gz_1,gz_2) = \mathbb{D}_1^2\lambda(g,u_K,t)\cdot(gz_1,gz_2) \\ &+ \mathbf{D}_{1,2}\lambda(g,u_K,t)\cdot(gz_1,\mathbf{D}_1 u_K(g,\xi,t)\cdot gz_2) \\ &+ \mathbf{D}_{2,1}\lambda(g,u_K,t)\cdot(\mathbf{D}_1 u_K(g,\xi,t)\cdot gz_1, gz_2) \\ &+ \mathbf{D}_2^2\lambda(g,u_K,t)\cdot(\mathbf{D}_1 u_K(g,\xi,t)\cdot gz_1, \mathbf{D}_1 u_K(g,\xi,t)\cdot gz_2) \\ &+ \mathbf{D}_2\lambda(g,u_K,t)\cdot\mathbb{D}_1^2 u_K(g,\xi,t)\cdot(gz_1,gz_2)\end{aligned}$$

$$\begin{aligned}&\mathbf{D}_{1,2}\lambda_K(g,\xi,t)\cdot(gz_1,\xi\zeta_2) = \\ &\mathbf{D}_{1,2}\lambda(g,u_K,t)\cdot(gz_1,\mathbf{D}_2 u_K(g,\xi,t)\cdot\xi\zeta_2) \\ &+ \mathbf{D}_2^2\lambda(g,u_K,t)\cdot(\mathbf{D}_1 u_K(g,\xi,t)\cdot gz_1, \mathbf{D}_2 u_K(g,\xi,t)\cdot\xi\zeta_2) \\ &+ \mathbf{D}_2\lambda(g,u_K,t)\cdot\mathbf{D}_{1,2} u_K(g,\xi,t)\cdot(gz_1,\xi\zeta_2)\end{aligned}$$

$$\begin{aligned}&\mathbf{D}_{2,1}\lambda_K(g,\xi,t)\cdot(\xi\zeta_1,gz_2) = \\ &\mathbf{D}_{2,1}\lambda(g,u_K)\cdot(\mathbf{D}_2 u_K(g,\xi,t)\cdot\xi\zeta_1, gz_2) \\ &+ \mathbf{D}_2^2\lambda(g,u_K,t)\cdot(\mathbf{D}_2 u_K(g,\xi,t)\cdot\xi\zeta_1, \mathbf{D}_1 u_K(g,\xi,t)\cdot gz_2) \\ &+ \mathbf{D}_2\lambda(g,u_K,t)\cdot\mathbf{D}_{2,1} u_K(g,\xi,t)\cdot(\xi\zeta_1,gz_2)\,.\end{aligned}$$

$$\begin{aligned}&\mathbb{D}_2^2\lambda_K(g,\xi,t)\cdot(\xi\zeta_1,\xi\zeta_2) = \\ &\mathbf{D}_2^2\lambda(g,u_K,t)\cdot(\mathbf{D}_2 u_K(g,\xi,t)\cdot\xi\zeta_1, \mathbf{D}_2 u_K(g,\xi,t)\cdot\xi\zeta_2) \\ &+ \mathbf{D}_2\lambda(g,u_K,t)\cdot\mathbb{D}_2^2 u_K(g,\xi,t)\cdot(\xi\zeta_1,\xi\zeta_2)\,,\end{aligned}$$

where it is understood that u_K in the above expressions is evaluated at (g,ξ,t).

References

[1] R. Bellman, *Dynamical Programming*. Princeton University Press, 1957.

[2] L. Pontryagin, V. Boltyasnkii, R. Gamkrelidze, and E. Mischenko, *The Mathematical Theory of Optimal Processes*. Interscience Publishers, 1962.

[3] M. G. Crandall and P. L. Lions, "Conditions d'unicité pour les solutions generalises des equations d'Hamilton-Jacobi du premier ordre," *C. R. Acad. Sci. Paris Sér. I Math.*, vol. 292, pp. 487–502, 1981.

[4] M. G. Grandall and P. L. Lions, "Viscosity solutions of Hamilton-Jacobi equations," *Transactions of the American Mathematical Society*, vol. 277, pp. 1–42, 1983.

[5] E. B. Lee and L. Markus, *Foundations of Optimal Control Theory*. Wiley, 1967.

[6] B. D. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*. Prentice Hall, 1989.

[7] M. Bardi and I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, ser. Systems & Control: Foundations & Applications. Birkhauser, 1997.

[8] A. Saccon, J. Hauser, and A. P. Aguiar, "Optimal control on non-compact Lie groups: A projection operator approach," in *the proceedings of the 49th IEEE Conference on Decision and Control (CDC)*, Atlanta, Georgia, USA, 2010, pp. 7111–7116.

[9] ——, "Optimal control on Lie groups: Implementation details of the projection operator approach," in *the proceedings of the 18th IFAC World Congress*, Milan, Italy, 2011.

[10] A. Saccon, A. P. Aguiar, and J. Hauser, "Lie group projection operator approach: Optimal control on $T\mathrm{SO}(3)$," in *the proceedings of the 50th Conference on Decision and Control (CDC) and European Control Conference (ECC)*, Orlando, Florida, USA, 2011.

[11] R. W. Brockett, "Lie theory and control systems defined on spheres," *SIAM Journal on Applied Mathematics*, pp. 213–225, 1973.

[12] J. Baillieul, "Geometric methods for nonlinear optimal control problems," *Journal of optimization theory and applications*, vol. 25, no. 4, pp. 519–548, 1978.

[13] A. M. Bloch, P. E. Crouch, J. E. Marsden, and A. K. Sanyal, "Optimal Control and Geodesics on Quadratic Matrix Lie Groups," *Foundations of Computational Mathematics*, vol. 8, no. 4, pp. 469–500, 2008.

[14] V. Jurdjevic, *Geometric control theory*. Cambridge University Press, 1997.

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

[16] E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*, ser. Computational Mathematics. Springer, 2006.

[17] A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna, "Lie-group methods," *Acta Numerica*, vol. 9, pp. 1–148, 2000.

[18] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization algorithms on Matrix Manifolds*. Princeton University Press, 2008.

[19] R. Mahony and J. H. Manton, "The geometry of the Newton method on non-compact Lie groups," *Journal of Global Optimization*, vol. 23, pp. 309–327, 2002.

[20] O. Junge, J. E. Marsden, and S. Ober-Blöbaum, "Discrete mechanics and optimal control," in *the proceedings of the 16th IFAC World Congress*, 2005.

[21] T. Lee, M. Leok, and N. H. Mc Clamroch, "Optimal attitude control of a rigid body using geometrically exact computations on SO(3)," *Journal of Dynamical and Control Systems*, vol. 14, no. 4, p. 465487, 2008.

[22] M. B. Kobilarov and J. E. Marsden, "Discrete Geometric Optimal Control on Lie Groups," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 641–655, Aug. 2011.

[23] J. Hauser, "A projection operator approach to the optimization of trajectory functionals," in *the proceedings of the 15th IFAC World Congress*, Barcelona, Spain, 2002.

[24] ——, "On the computation of optimal state transfers with application to the control of quantum spin systems," in *the proceedings of the American Control Conference (ACC)*, 2003.

[25] J. Hauser and A. Saccon, "A barrier function method for the optimization of trajectory functionals with constraints," in *the proceedings of the 45th IEEE Conference on Decision and Control (CDC)*, 2006.

[26] A. Saccon, J. Hauser, and A. Beghi, "Trajectory exploration of a rigid motorcycle model," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 2, pp. 424–437, 2012.

[27] P. Martin, E. Johnson, T. Murphey, and M. Egerstedt, "Constructing and Implementing Motion Programs for Robotic Marionettes," *IEEE Transactions on Automatic Control (TAC)*, vol. 56, no. 4, pp. 902–907, 2011.

[28] G. Notarstefano and J. Hauser, "Computing feasible trajectories for constrained maneuvering systems : the PVTOL example," *Automatica (accepted for publication in)*, 2011, arXiv:1112.0879v1.

[29] M. B. Horowitz and J. W. Burdick, "Combined Grasp and Manipulation Planning as a Trajectory Optimization Problem," in *the proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 584–591.

[30] J. Hauser and D. G. Meyer, "The trajectory manifold of a nonlinear control system," in *the proceedings of the 37th IEEE Conference of Decision and Control (CDC)*, vol. 1, 1998, pp. 1034–1039.

[31] D. G. Luemberger, *Optimization by Vector Space Methods*. John Wiley & Sons, 1969.

[32] W. Boothby, *An introduction to differentiable manifolds and Riemannian geometry*, 2nd ed., ser. Pure and applied mathematics. Academic Press, Boston, 1986.

[33] R. Abraham, J. E. Marsden, and T. Ratiu, *Manifolds, tensor analysis, and applications*. Springer-Verlag, 1988.

[34] J. M. Lee, *Riemannian manifolds: an introduction to curvature*. Springer, 1997.

[35] V. Varadarajan, *Lie groups, Lie algebras, and their representations*. Springer-Verlag, 1984.

[36] J. Marsden and T. Ratiu, *Introduction to Mechanics and Symmetry*, 2nd ed. Springer, 1999.

[37] W. Rossmann, *Lie groups. an introduction through linear groups*. Oxford University Press, 2002.

[38] F. Bullo and A. D. Lewis, *Geometric Control of Mechanical Systems*, ser. Texts in Applied Mathematics. Springer-Verlag, 2004, vol. 49.

[39] A. M. Bloch, *Nonholonomic Mechanics and Control*. Springer-Verlag, 2003.

[40] K. Nomizu, "Invariant affine connections on homogeneous spaces," *American Journal of Mathematics*, vol. 76, pp. 33–65, 1954.

[41] E. Cartan and J. Schouten, "On the Geometry of the Group-manifold of simple and semi-simple groups," *Koninklijke Akademie van Wetenschappen (Royal Academy of Studies, Proceedings of)*, vol. 24, pp. 803–815, 1926.

[42] E. Cartan, "La géométrie des groupes de transformations," *Journal de Mathématiques pures et appliquées*, vol. 6, no. 9, pp. 1–119, 1927.

[43] N. Bou-Rabee and J. Marsden, "Hamilton-Pontryagin integrators on Lie groups part I: Introduction and structure-preserving properties," *Foundations of Computational Mathematics (FOCM)*, vol. 9, no. 2, pp. 197–219, April 2009.

[44] H. Khalil, *Nonlinear Systems*, 2nd ed. Prentice Hall, 1996.

[45] J. E. Marsden and T. J. R. Hughes, *Mathematical Foundations of Elasticity*. Prentice-Hall, 1983.

[46] A. Saccon, J. Hauser, and A. P. Aguiar, "Exploration of kinematic optimal control on the Lie group $\mathrm{SO}(3)$," in *the proceedings of the 8th IFAC Symposium on Nonlinear Control Systems (NOLCOS)*, Bologna, Italy, 2010.

[47] J. Nocedal and S. J. Wright, *Numerical optimization*. Springer-Verlag, 1999.

[48] C. T. Kelley, *Iterative Methods for Optimization*, ser. Frontiers in Applied Mathematics, 18. SIAM, 1999.

[49] A. Saccon, A. P. Aguiar, A. J. Häusler, J. Hauser, and A. M. Pascoal, "Constrained motion planning for multiple vehicles on SE(3)," in *the proceedings of the 51th IEEE Conference on Decision and Control (CDC)*, Maui, Hawaii, US, 2012.