

1 Motivation

The proposed research is supported by NSTGRO to improve NASA’s precision landing technology. Precision landing of spacecraft is characterized by the landing ellipse, which is determined by a target landing site and the covariance of spacecraft horizontal positions. Powered descent is the last stage of the landing where engines are reignited so that the spacecraft follows closely along a fuel-optimal trajectory to reach target site. From this perspective, the state of the spacecraft is modeled as a stochastic process, and finding a fuel-minimizing thrust profile during this stage that meets the landing ellipse objective becomes solving a stochastic optimal control problem.

Although solving a generic stochastic optimal control problem is a formidable task, it suffices to approximate the uncertainty with its second moment, the covariance, to meet the landing ellipse requirement. This way, the problem reduces to a *covariance steering* (CS) problem. That is, we optimally steer the initial mean state and covariance of the spacecraft to a prescribed final mean state and covariance, corresponding to the target site and the landing ellipse.

CS for a linear stochastic system in the Euclidean space is more or less solved. For the nonlinear case, all relevant work to the best of my knowledge use the same metastrategy: they iteratively linearize the dynamics in the ambient Euclidean space and apply linear CS. We shall refer to this type of linearization as “naive linearization”. Moreover, none of them has applied their algorithm to rigid-body (6DOF) control. Because rigid-body motion has a rich geometric structure, we aim to fully exploit its geometry to outperform generic nonlinear algorithms.

2 Mathematical Background

We assume basic knowledge of Lie groups and their Lie algebras. For a Lie group $(G, *)$ of dimension n and its Lie algebra $(\mathfrak{g}, [\cdot, \cdot])$, we denote the identification $\mathfrak{g} \cong \mathbb{R}^n$ after choosing a basis by $X \mapsto X^\vee \in \mathbb{R}^n$ and $x^\wedge \mapsto X \in \mathfrak{g}$. The **adjoint action** $\text{Ad} : G \rightarrow \text{Aut}(\mathfrak{g})$ of G on \mathfrak{g} is the infinitesimal action at identity of G acting on itself by conjugation. The

adjoint representation $\text{ad} : \mathfrak{g} \rightarrow \text{Der}(\mathfrak{g})$ of \mathfrak{g} on \mathfrak{g} is the derivative of Ad at e . Note that $\text{ad}_X(Y) = [X, Y]$.

Remark 2.1 The adjoint action for matrix Lie groups is the similarity transformation $\text{Ad}_A(X) = AXA^{-1} = (AX^\vee)^\wedge$, which transforms the coordinates in the post- A -transformation frame back to coordinates in the pre-transformation frame. For matrix Lie groups, the little ad is the commutator $\text{ad}_X(Y) = XY - YX$.

We recall that the semidirect product of groups H and K can be defined by either a left action $\lambda : K \rightarrow \text{Aut}(H)$ or a right action $\rho : K \rightarrow \text{Aut}(H)$. We can convert ρ to a left action ρ^{-1} . That is, we have

$$\begin{aligned} K \rtimes_\lambda H &: (k_1, h_1) \rtimes (k_2, h_2) = (k_1 k_2, h_1 \lambda_{k_1}(h_2)) \\ K \rtimes_{\rho^{-1}} H &: (k_1, h_1) \rtimes (k_2, h_2) = (k_1 k_2, \rho_{k_2}^{-1}(h_1) h_2). \end{aligned}$$

Remark 2.2 The semidirect product under a right action is much less mentioned in the literature. This leads to a significant bias toward the left-action one, and we shall see

The tangent bundle TG of G is a Lie group under the differential of $*$. Since left and right multiplication actions of G on itself have induced actions on the tangent vectors, we can left or right trivialize the tangent bundle TG as a semidirect product of Lie groups. Explicitly, we have Lie group isomorphisms $G \rtimes_{\text{Ad}^{-1}} T_e G \cong TG \cong G \rtimes_{\text{Ad}} T_e G$.

Remark 2.3 The vector space $T_e G$ can be either viewed as a Lie group under addition or a Lie algebra under the trivial bracket. Context should easily differentiate the two. We reserve \mathfrak{g} to denote the corresponding Lie algebra of G with the ad bracket.

The **unit quaternions** \mathbb{H}_u is isomorphic to $\text{SU}(2)$ as Lie groups and therefore is a double-cover of $\text{SO}(3)$. Its Lie algebra \mathfrak{h}_u is the set of pure quaternions \mathbb{H}_p that can be identified as \mathbb{R}^3 with cross product bracket. The **dual unit quaternions** $\tilde{\mathbb{H}}_u$ uses dual numbers to attach tangent vectors to \mathbb{H}_u , making it an explicit parametrization of $T\mathbb{H}_u$, a double-cover of $T\text{SO}(3)$.

3 Research Proposal

To solve for the optimal control for the CS problem, we have the following strategy:

- (1) Fully understand the left-trivialized Lie group geometry of the state space and derive necessary Lie group operations.
- (2) Use the derived operations to set up controlled dynamics of mean and covariance using a Lie-theoretic uncertainty propagation model.
- (3) Solve the optimal control by selecting a suitable numerical scheme.
- (4) Prove optimality/extremality of the solution using optimal control theorems generalized from the Euclidean case.

3.1 Lie Group Geometry of State Space

Rigid-body motions, including rotations and translations, together form a Lie group called the special Euclidean group $SE(3)$. We can identify rotations with $SO(3)$ and translations/positions with \mathbb{R}^3 , and the frame-dependency of translations is precisely captured by the adjoint action of rotations on translations if we make the identification $\mathbb{R}^3 \cong T_I SO(3)$. This yields two semidirect products, one corresponding to left-trivialization $SO(3) \ltimes_{Ad^{-1}} \mathbb{R}^3$ of $TSO(3)$ and the other to right-trivialization $SO(3) \ltimes_{Ad} \mathbb{R}^3$. By convention, the \mathbb{R}^3 in the left trivialization corresponds to the body frame position whereas right corresponds to spatial frame position. Either one can be used as the configuration space of 6DOF spacecraft dynamics. Similarly, we can trivialize $\tilde{\mathbb{H}}_u$ either way and use this double-cover of $SE(3)$ as the configuration space as well.

In 6DOF dynamics state space, we include angular and linear velocities, which live in the tangent bundle of the configuration space. This allows us to trivialize again and obtain a twice-trivialized Lie group as the state space. For twice-left-trivialized dual unit quaternions, this looks like $(\mathbb{H}_u \ltimes_{Ad_0^{-1}} \mathbb{H}_p) \ltimes_{Ad_1^{-1}} (\mathfrak{h}_u \ltimes_{ad_0} T_1 \mathbb{H}_p)$, where subscript denotes the Lie group indexed by the number of trivializations it goes through. For attitude quaternion \mathbf{q} , body position r , body angular velocity ω , and body velocity v , multiplication in this group is

$$((\mathbf{q}_1, r_1), (\omega_1, v_1)) \ltimes ((\mathbf{q}_2, r_2), (\omega_2, v_2))$$

$$= ((\mathbf{q}_1 \mathbf{q}_2, \mathbf{q}_2^* r_1 \mathbf{q}_2 + r_2), (\mathbf{q}_2^* \omega_1 \mathbf{q}_2 + \omega_2, \mathbf{q}_2^* v_1 \mathbf{q}_2 + v_2 + (\mathbf{q}_2^* \omega_1 \mathbf{q}_2) \times r_2)).$$

Knowing its multiplication structure allows us to derive other important operations on this group.

Although it is more common to see right-trivialized $\text{SE}(3)$ in the literature, for control purposes it is crucial to use the left trivialization to include velocities because the control enters the system through the body frame. Simplification occurs when we left-trivialize twice compared to a mix of left and right trivializations, so we shall stick with twice-left-trivialized state space.

3.2 Uncertainty Propagation on Lie Groups

A natural way to define a stochastic process on a Lie group is to define an SDE on the Lie algebra and repeatedly inject it (for each infinitesimal time step) onto the group via the exponential map, a technique called **McKean-Gangolli injection**. Then, using Lie-theoretic Fokker-Planck equation, one can derive a closed-form approximation, called EMD2, of the left-trivialized mean and covariance dynamics. An alternative way is to parametrize the group and derive an SDE such that if we evolve it on the parameter space the entire time and then map it to the group, the trajectory coincides with that from McKean-Gangolli injection. Either way we obtain the same approximation. We compare the propagation performance of EMD2 with naive linearization and gold-standard UKF-M for attitude dynamics.

At time t , let $\mu(t) \in G$ be the mean state, $\Sigma(t) \in (n)$ be the covariance, and $\mathcal{U}(t)$ be the set of admissible controls. CS approximates stochastic OCP with the following deterministic problem:

$$\begin{aligned} \min_{u \in \mathcal{U}} \quad & \int_0^{t_f} \frac{1}{2} \|u(t)\|^2 dt \\ \text{subject to} \quad & \mu^{-1} \dot{\mu} = f(\mu, \Sigma, u, t), \\ & \dot{\Sigma} = F(\mu, \Sigma, u, t), \quad \Sigma \succeq 0, \\ & \mu(0) = \mu_0, \quad \mu(t_f) = \mu_{t_f}, \\ & \Sigma(0) = \Sigma_0, \quad \Sigma(t_f) = \Sigma_{t_f}, \\ & \text{free } t_f, \end{aligned}$$

3.3 Optimality of the Candidate Solutions

Our main theoretical tool for proving optimality/extremality is **Pontryagin’s Maximum Principle** (PMP) adapted for left-trivialized dynamics on Lie groups. It provides the necessary conditions of optimal control. We can try to generalize Legendre–Clebsch’s second-order condition to prove the local optimality of the solutions.

3.4 Numerical Scheme

Although PMP provides enough conditions to reduce the optimal control problem to a BVP problem, BVP solvers (shooting, collocation) are not as flexible as direct optimization when there is additional constraints, including the semidefinite cone constraint on the covariance. Sequential convex programming (SCP) is a highly efficient way to solve for optimal control. It discretizes an optimization problem in an infinite dimensional space to optimization on finite number of nodes, where each neighboring pair satisfy discretized dynamics constraint. All nonconvex constraints are linearized at current states so that an intermediate solution can be found quickly using convex programming. Then the algorithm updates the current states and iterate until convergence. However, SCP in its current form commits naive linearization including linearizing the highly nonlinear exponential map.

Inspired by Invariant EKF [?] and error-state MPC [?] , we propose a novel error-state SCP variant where decision variables are not the states, but the errors from the current trajectory in the Lie algebra. All linearizations will be done in the Lie algebra with no direct linearization of the exponential map. This relaxed problem can be solved efficiently using semidefinite programming. By using the twice-trivialized dual unit quaternions, we use fewer decision variables than matrices and aim to make the algorithm fast enough that we can recompute optimal trajectory frequently during flight when the spacecraft deviates from nominal trajectory.

3.5 From Theory to Application

As part of the NSTGRO experience, I will visit multiple NASA centers over three years to turn my theoretical and algorithmic developments into technology that can be used for

precision landing on the moon, Mars, and beyond.

3.6 Further Directions

Once this problem is decisively solved, I will investigate strategies to derive and compute feedback optimal control, where the control is a function of the state instead of time. Both optimal transport and dynamic programming can be used to find feedback control in the linear/Euclidean case, and both have some partial generalizations to Riemannian manifolds. It is perhaps a fruitful endeavor to further generalize them to left-trivialized Lie groups.