Nonlinear Control using Coordinate-free and Euler Formulations: An Empirical Evaluation on a 3D Pendulum

Avinash Siravuru [*] **Koushil Sreenath** [**]

[*] *Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15231, USA. (e-mail: avinashs@cmu.edu).*
[**] *Department of Mechanical Engineering, University of California, Berkeley, USA. (e-mail: koushils@berkeley.edu)*

Abstract: Pendulum dynamics are widely utilized in robotics control literature to test and evaluate novel control design techniques. They exhibit many interesting features commonly seen in real-world nonlinear systems and yet they are simple enough for quick prototyping, further analysis, and benchmarking. In this work, we study the impact of a $3D$ pendulum's orientation parametrization on stabilization performance. Mainly, we show that using a global or coordinate-free formulation for dynamics and control is not only singularity-free but also more input-efficient. We validate this empirically by running over 700 stabilization simulations across the full configuration space of a $3D$ pendulum and compare the performance of a geometric and a Euler-parametrized controller. We show that the geometric controller is able to leverage the inherent manifold curvature and flow along geodesics for efficient stabilization.

1. INTRODUCTION

Pendular systems are widely studied in the robotics and control community to discover and characterize nonlinear dynamical phenomena like symmetries, bifurcations, orbital stability, etc. (Lewis et al. (1992); Chaturvedi et al. (2011b); Lee et al. (2011)). The deeper understanding of these behaviors helps control theorists devise nonlinear control techniques for effective stabilization and tracking (Chung and Hauser (1995); Åström and Furuta (2000); Shiriaev et al. (1999)). The impact of pendular systems in robotics cannot be understated. Most complex robotic systems commonly apply pendular abstractions to model and control their dominant behaviors. For example, see pendulum-like models in robotic manipulation (Lefrançois and Gosselin (2010); Cunningham and Asada (2009); Zanotto et al. (2011)), inverted pendulum models in legged locomotion (Kajita et al. (2001); Chevallereau et al. (2018); Poulakakis and Grizzle (2009)), and multi-link pendular models in brachiation (Saito et al. (1994); Farzan et al. (2019)). To date, multi-link pendula, remain the best nonlinear system models for benchmarking performance of new control algorithms on typical real-world challenges like underactuation, model uncertainty, stochasticity, etc.

Traditionally, nonlinear control design techniques used for stabilization or swinging up of a $3D$ or *spherical* pendulum (a.k.a $3D$ pendulum that cannot yaw) used Euler angles to define the pendulum configuration (Shiriaev et al. (1999); Liu et al. (2005); Aguilar-Ibanez et al. (2006)). However, more recently, coordinate-free formulations have been proposed to define pendulum configuration and corresponding dynamical models (Chaturvedi et al. (2011b); Lee et al. (2011); Bittner and Sreenath (2016)) and suitable control laws (Chaturvedi et al. (2011a); Lee (2011)) have been de-

vised. These globally-defined dynamical formulations are *singularity-free*, i.e., they are devoid of kinematic singularities like gimbal-lock seen in locally-defined formulations like Euler angles (For example, the ZXY-ordering Euler-angles ($[\phi\ \theta\ \psi]^T$) exhibit a singularity when $\phi = \pi/2$). The singularity-free property of global formulations has implications in control design, particularly, recovery from large angle disturbances is possible and the controllers can be designed to be *almost-globally* attractive (as opposed to weaker local attractivity properties of Euler-based control designs). A new sub-field in nonlinear control, called *geometric control* has emerged that exploits these control laws and its been heavily applied in UAV literature, see Lee et al. (2013); Sreenath and Kumar (2013); Lee (2017); Mueller (2018) and more recently in legged locomotion, see Sreenath and Sanyal (2015); Siravuru et al. (2018); Ding et al. (2019).

Prior literature on geometric control has primarily focused on providing mathematical rigor and experimental validation to geometric control, specifically the almost-global attractivity property and the freedom from singularities. While these two properties are more meaningful for developing UAV maneuvers to quickly recover from large disturbances, for legged robots and other systems, the use of geometric control needs further motivation. Large angle recoveries are uncommon (restricted to acrobatic motions) and may be impractical during regular locomotion tasks like walking, running, etc. due to limited control authority, finite time (to impact) and narrow region-of-attraction.

In this work, we demonstrate the other advantages of geometric control that make it an appealing choice even for legged locomotion. Through a comprehensive empirical evaluation of $3D$ Pendulum stabilization to its hanging equilibrium position, we show that geometric control is more *input-efficient* than traditional Euler angles-based
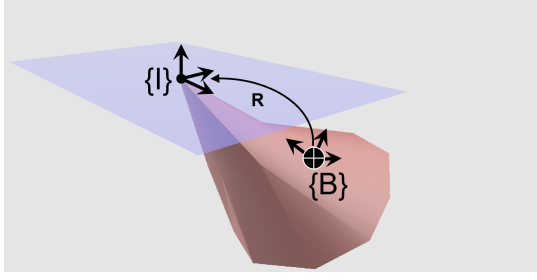
Fig. 1. A 3D pendulum is a rigid body pinned at one end (pivot) restricting its motion to be purely rotational.

nonlinear control. This fact purely stems from the exploitation of inherent curvature in the $\mathbb{SO}(3)$ manifold space that is thrown away when an Euler-parameterization is used. The feedback laws built directly on $\mathbb{SO}(3)$ use the geodesics for error correction and are therefore more efficient. As both nonlinear controllers are equivalent to the first order, the efficiency of the geometric controller is more pronounced for larger error recovery.

The rest of this paper is organized as follows. In Section 2, we summarize the $3D$ pendulum dynamical model using both Euler and geometric formulations. Next, in Section 3, we define feedback linearization control laws for both the Euler and geometric formulations and compare the error metric choices of the two formulations. In Section 4, we comprehensively evaluate the performance of the two models, and finally, we provide concluding remarks in Section 5.

2. MODEL DEFINITIONS

To define the mathematical model of the 3D pendulum, we first need to define a stationary fixed frame (inertial frame) about which the pendulum configuration is measured. Denote this frame as $\{I\}$ and fix it at the origin, which is also the pivot for the pendulum. Second, we attach a moving frame (body frame), denoted as $\{B\}$, to the center-of-mass (CoM) of the pendulum. Having defined the frames, the next step is to choose a suitable parametrization to represent a *3D rotation* - this translates to estimating the orientation of $\{B\}$ w.r.t. $\{I\}$, as shown in Fig. 1. In this work, we use two methods, 1) Euler-based (ZYX ordering): $q \in \mathbb{R}^3$, and 2) Geometric: $R \in \mathbb{SO}(3)$. Other model parameters used for modeling and control design are defined and summarized in Table 1.

2.1 3D Pendulum Dynamics

Pendulum dynamics using both Euler and $\mathbb{SO}(3)$ formulations can be compactly expressed as,

Euler:
$$\boldsymbol{D_e(q)\ddot{q} + H_e(q,\dot{q})\dot{q} = B_e u_e,} \tag{1}$$

where $q = [\phi\ \theta\ \psi]^T$, $\dot{q} = [\dot{\phi}\ \dot{\theta}\ \dot{\psi}]^T$, and $u_e \in \mathbb{R}^3$.

$\mathbb{SO}(3)$:
$$\boldsymbol{D_s(R)\dot{\Omega} + H_s(R,\Omega)\Omega = B_s u_s.} \tag{2}$$

where $R \in \mathbb{SO}(3)$, $\Omega \in T_R\mathbb{SO}(3)$, and $u_s \in T_R^*\mathbb{SO}(3)$. Here, $T_R\mathbb{SO}(3)$ and $T_R^*\mathbb{SO}(3)$ are *tangent* and *co-tangent* spaces on $\mathbb{SO}(3)$ at the configuration R.

$\{I\}$	Inertial frame fixed to the pivot.
$\{B\}$	Body frame fixed at the CoM of pendulum.
$e_3 \in \mathbb{R}^3$	Standard unit vector pointing upward.
$m \in \mathbb{R}$	Mass of the pendulum link.
$J \in \mathbb{R}^{3\times 3}$	Inertia of the pendulum expressed in frame $\{B\}$.
$l_c \in \mathbb{R}$	Length of the pendulum CoM w.r.t the pivot.
$R \in \mathbb{SO}(3)$	Rotation matrix from $\{B\}$ to $\{I\}$.
$\Omega \in \mathbb{R}^3$	Angular velocity of the pendulum expressed in the body frame $\{B\}$.
$\phi,\theta,\psi \in \mathbb{R}$	Roll, Pitch and Yaw angles used to express pendulum configuration using an Euler parametrization.
$\widehat{(.)}$	*hat* map is a linear mapping from $\mathbb{R}^3$ to $\mathfrak{so}(3)$.
$(.)^\vee$	*vee* map is a linear mapping from $\mathfrak{so}(3)$ to $\mathbb{R}^3$.

Table 1. Symbols and parameter definitions for $3D$ Pendulum modeling and control design.

The Euler dynamics can be easily derived and we omit it here for brevity. However, $D_e(q)$ will be used later to check for singularities and to define suitable mappings to go back and forth between Euler and geometric inputs. The $\mathbb{SO}(3)$ dynamics equations of motion are compact, as shown below:

$$\dot{R} = R\widehat{\Omega}, \tag{3}$$
$$\dot{\Omega} = J^{-1}(-\widehat{\Omega}J\Omega - mg\widehat{l_c}R^T e_3 + u_s). \tag{4}$$

For the derivation of these equations, see Lee et al. (2017) and Chaturvedi et al. (2011b). The *hat* map used in equation 4, and its inverse - the *vee* map - are defined in Table 1. Note that the dynamics cannot be directly compared and we need to define suitable mapping between the Euler angles, rates and inputs to their counterparts in $\mathbb{SO}(3)$.

2.2 Transfer Maps

Using the ZXY ordering of Euler angles, we define $\mathcal{R}(q) : \mathbb{R}^3 \to \mathbb{SO}(3)$ whose expression is given as,

$$\mathcal{R}(q) = \begin{pmatrix} c\theta\,c\psi - s\phi\,s\theta\,s\psi & -c\phi\,s\psi & c\psi\,s\theta + c\theta\,s\phi\,s\psi \\ c\theta\,s\psi + c\psi\,s\phi\,s\theta & c\phi\,c\psi & s\theta\,s\psi - c\theta\,c\psi\,s\phi \\ -c\phi\,s\theta & s\phi & c\phi\,c\theta \end{pmatrix}, \tag{5}$$

s.t. $R = \mathcal{R}(q)$. Here, we compactly represent $cos(\alpha)$ and $sin(\alpha)$ as $c\alpha$ and $s\alpha$, respectively, and α is a placeholder for any Euler angle in q.

Next, we define $\mathcal{T}_{\dot{q}}(q) : \mathbb{R}^3 \to \mathbb{R}^{3\times 3}$ to convert Euler rates ($\dot{q}$) to angular velocities (Ω) as shown below:

$$\mathcal{T}_{\dot{q}}(q) = \begin{pmatrix} c\theta & 0 & -c\phi\,s\theta \\ 0 & 1 & s\phi \\ s\theta & 0 & c\phi\,c\theta \end{pmatrix}, \qquad \text{s.t. } \Omega = \mathcal{T}_{\dot{q}}(q)\dot{q}. \tag{6}$$

Finally, to convert Euler input (u_e) to $\mathbb{SO}(3)$ input (u_s), we define $\mathcal{T}_u(q,R) : \mathbb{R}^3 \times \mathbb{SO}(3) \to \mathbb{R}^{3\times 3}$ as,

$$\mathcal{T}_u(q,R) = (D_s^{-1}B_s)^{-1}\mathcal{T}_{\dot{q}}(q)(D_e^{-1}B_e), \tag{7}$$
$$\text{s.t } u_s = \mathcal{T}_u(q,R)u_e.$$

$\mathcal{T}_{\dot{q}}$ **Derivation**: We first define $R_w(\alpha) : \mathbb{R} \to \mathbb{SO}(3)$ as a mapping from an Euler angle α to its corresponding axis-specific rotation matrix R_{w_i}. Following the *ZXY*

Euler sequencing, we have $\alpha \in [\psi \ \phi \ \theta]^T$ representing a rotation about an axis denoted by $w_i \in [Z \ X \ Y]^T$ and whose unit vector is denoted by e_i. Accordingly, we have $\dot{R}_{w_i} = R_{w_i}\widehat{\dot\alpha e_i}$. Using this notation, we express R as a product of its individual axis-wise rotation elements $R_z(\psi)$, $R_x(\phi)$, and $R_y(\theta)$. We can then compute its first derivative, $\dot{R}$, as a function of Euler angles q and their Euler rates $\dot{q}$, as shown below:

$$R = R_z R_x R_y,$$
$$\implies \dot{R} = \dot{R}_z R_x R_y + R_z \dot{R}_x R_y + R_z R_x \dot{R}_y,$$
$$= R_z\widehat{\dot\psi e_3}R_x R_y + R_z R_x\widehat{\dot\phi e_1}R_y + R_z R_x R_y\widehat{\dot\theta e_2}.$$

From equation (3), we know that $\Omega = (R^T \dot{R})^\vee$. Therefore,

$$\widehat{\Omega} = R_y^T R_x^T R_z^T R_z\widehat{\dot\psi e_3}R_x R_y + R_y^T R_x^T R_z^T R_z R_x\widehat{\dot\phi e_1}R_y +$$
$$R_y^T R_x^T R_z^T R_z R_z R_x R_y\widehat{\dot\theta e_2},$$
$$= R_y^T R_x^T\widehat{\dot\psi e_3}R_x R_y + R_y^T\widehat{\dot\phi e_1}R_y + \widehat{\dot\theta e_2},$$
$$= \widehat{R_y^T R_x^T \dot\psi e_3} + \widehat{R_y^T \dot\phi e_1} + \widehat{\dot\theta e_2}.$$

Finally, we apply a *vee* map on both sides to get,

$$\Omega = [R_y^T e_1 \ \ e_2 \ \ R_y^T R_x^T e_3]\begin{pmatrix} \dot\phi \\ \dot\theta \\ \dot\psi \end{pmatrix}, \qquad (8)$$

$$\implies \mathcal{T}_{\dot q}(q) = [R_y^T e_1 \ \ e_2 \ \ R_y^T R_x^T e_3] = \begin{pmatrix} c\theta & 0 & -c\phi\, s\theta \\ 0 & 1 & s\phi \\ s\theta & 0 & c\phi\, c\theta \end{pmatrix}$$

$\mathcal{T}_u$ **Derivation**: We can obtain $\mathcal{T}_u$ by taking the time derivative of equation (6), then substituting the dynamics from equations (12) and (13) for $\ddot{q}$ and $\dot{\Omega}$. Finally, we equate the input vector fields on both sides to finish the derivation. In particular,

$$\Omega = \mathcal{T}_{\dot q}(q)\dot{q},$$
$$\implies \dot{\Omega} = \dot{\mathcal{T}}_{\dot q}\dot{q} + \mathcal{T}_{\dot q}\ddot{q},$$
$$\underbrace{-D_s^{-1}H_s}_{=:\tilde{A}_s} + \underbrace{D_s^{-1}B_s u_s}_{=:\tilde{B}_s} = \underbrace{\dot{\mathcal{T}}_{\dot q}\dot{q} - \mathcal{T}_{\dot q}D_e^{-1}H_e}_{=:\tilde{A}_e} + \underbrace{\mathcal{T}_{\dot q}(D_e^{-1}B_e u_e)}_{=:\tilde{B}_e}.$$
$$(9)$$

Note that, the equality in equation (9) must hold for all u_e and u_s. Setting $u_e = u_s = \mathbf{0}$ results in $\tilde{A}_e = \tilde{A}_s$. Eliminating, $\tilde{A}_e$ and $\tilde{A}_s$, we can define,

$$u_s = \underbrace{(D_s^{-1}B_s)^{-1}\mathcal{T}_{\dot q}(D_e^{-1}B_e)}_{\mathcal{T}_u} u_e,$$
$$\implies \mathcal{T}_u = (D_s^{-1}B_s)^{-1}\mathcal{T}_{\dot q}(D_e^{-1}B_e).$$

Using these transfer maps it is easy to compare the Euler and geometric control laws which are going to be defined in the next section.

2.3 Pendulum State Sampling:

In the following sections, we empirically evaluate kinematic and dynamical defects like singularities, control design attributes like error metrics, input profiles, etc. for both $\mathbb{SO}(3)$ and Euler models for a wide range of pendulum configurations/states. The main emphasis of this
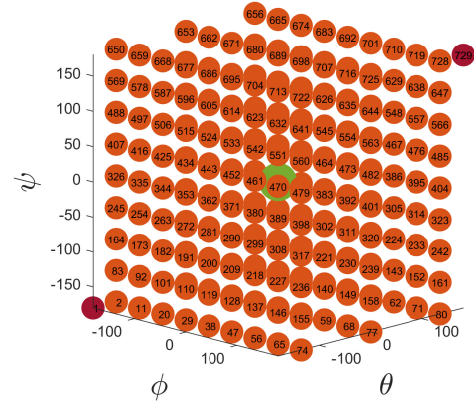

Fig. 2. The figure shows a coarse grid of the sampling points used in this study along with their *sample number*. The starting and final experiment indices are highlighted in **red**, and correspond to ($\phi = \theta = \psi = -\pi$) and ($\phi = \theta = \psi = \pi$), respectively. To better visualize the control studies, the desired final position (hanging equlibrium) is also shown in **green** and corresponds to ($\phi = \theta = \psi = 0$).
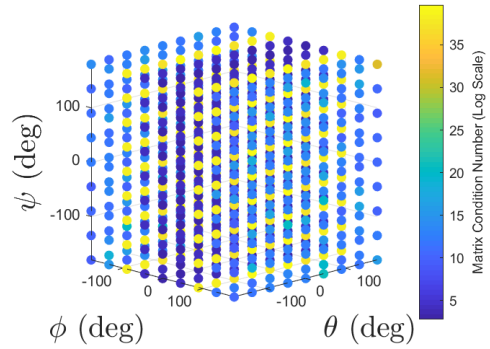

Fig. 3. A scatter plot of $D_s(q)$ matrix condition number (on the log scale) for all the state samples from Sec. 2.3. All the **yellow** points are singular.

work is to use this comprehensive empirical evaluation to highlight the benefits of *geometric control* and complement mathematically rich previous works.

The states are sampled from a $[-\pi, \pi]$ range of ϕ, θ, and ψ values with a resolution of $\pi/4$. The corresponding $\mathbb{SO}(3)$ states can be obtained using transfer map $\mathcal{R}$. For dynamic and kinematic studies, these configurations can be treated as a potential intermediate state in the pendulum motion trajectory. For control studies, they are used as initial conditions from which the pendulum is stabilized to its hanging equilibrium position. Some of these state samples are shown in Fig. 2 along-with the sample numbers to be used in plots that follow.

Note that both $\mathcal{T}_{\dot q}$ and D_e lose rank when $\phi = \pi/2$. This is the singularity issue that plagues Euler parametrization. Singular states are littered all over the configuration space, as shown in Fig. 3, making large disturbance recovery challenging using Euler parametrization.
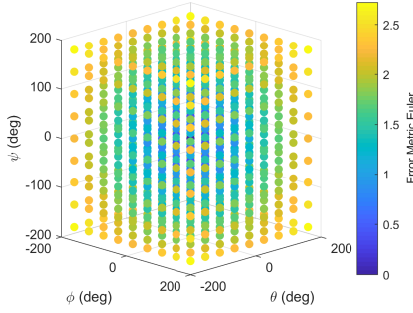
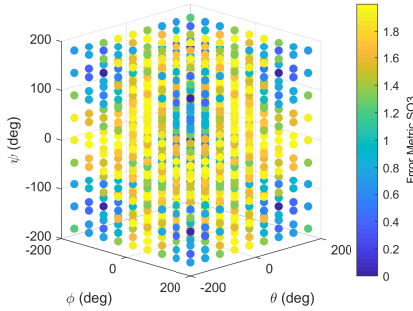Fig. 4. Ψ_e for $q_d = [0\ 0\ 0]^T$ and for all q in Sec. 2.3.

Fig. 5. Ψ_s for $R_d = I$ and for all R in Sec. 2.3.

3. CONTROL LAWS

Error Metrics: Before designing controllers and testing their performance, it is worth defining suitable error metrics for the Euclidean and $\mathbb{SO}(3)$ spaces to measure error growth between desired and actual pendulum states as they spread apart. This metric is directly proportional to the control expense involved. This helps us visualize apriori configurations that require greater control effort.

Euler: Define non-negative function $\Psi_e : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$ as,

$$\Psi_e = \frac{1}{2}\sqrt{(q - q_d)^T(q - q_d)}. \tag{10}$$

$\mathbb{SO}(3)$: Define $\Psi_s : \mathbb{SO}(3) \times \mathbb{SO}(3) \to \mathbb{R}$ as,

$$\Psi_s = \frac{1}{2}tr[I - R_d^T R]. \tag{11}$$

Assuming the hanging equilibrium configuration (i.e. $q_d = [0\ 0\ 0]^T$ or $R_d = I$) is desired, we pick q from all the states defined in the earlier section (Fig. 2) and plot Ψ_e in Fig. 4 and Ψ_s in Fig. 5.

On the $\mathbb{SO}(3)$ manifold, configurations where either one or two Euler angles $\approx \pi$ (antipodal configurations) are the farthest from q_d. These configurations are depicted by the cross-type bands around face centers on the samples cube in Fig. 5. On the other hand, note that the highest error points in Fig. 4 pertain to instances where all the three Euler angles $\approx \pi$ (depicted by the edges of the samples cube). This is non-intuitive as these points are actually very close to the desired configuration on the $\mathbb{SO}(3)$ manifold and should require minimal effort to reach q_d. Even before evaluating the controller performance, the error metrics clearly highlight the inefficiencies induced by an Euler-based controller design.

Control Design: In this work, we choose Feedback Linearization (FL) for pendulum stabilization. We begin by rewriting equations (1) and (2) in the control-affine form as,

$$\dot{x}_e = f_e(x_e) + g_e(x_e)u_e, \tag{12}$$
$$\dot{x}_s = f_s(x_s) + g_s(x_s)u_s, \tag{13}$$

where, $x_e = [q\ \dot{q}]^T$ and $x_s = [R\ \Omega]^T$. Note that x_s is a *set*. The Feedback Linearization schemes for both models are summarized below:

Euler: For the Euler case, it is fairly straightforward to derive an appropriate feedback linearizing policy by defining a suitable output as $y_e = h_e(q) = q - q_d$. Here, $q_d(t)$ can be time-varying and the control problem transitions to tracking from regulation. The output is relative degree 2. The control goal is to drive the output $y_e \to 0$, i.e.,

$$y_e := h_e(q) = q - q_d(t) \to 0, \tag{14}$$
$$\implies \dot{y}_e = L_{f_e} h_e = \dot{q} - \dot{q}_d(t), \tag{15}$$
$$\implies \ddot{y}_e = L_{f_e}^2 h_e + (L_{g_e} L_{f_e} h_e)u_e - \ddot{q}_d(t) = \ddot{q} - \ddot{q}_d(t), \tag{16}$$

where, $L_{f_e} h_e$ is the lie derivative and $L_{f_e} h_e(q) = \frac{\partial h_e}{\partial q} f$. Now lets define suitable *feedforward* and *feedback* terms,

$$u_e^{ff} := -(L_{g_e} L_{f_e} h_e)^{-1}(L_{f_e}^2 h_e), \tag{17}$$
$$u_e^{fb} := -(L_{g_e} L_{f_e} h_e)^{-1}(K_p y_e + K_d \dot{y}_e). \tag{18}$$

Applying $u_e := u_e^{ff} + u_e^{fb}$ in equation (16) results in a closed-loop linear system,

$$\ddot{y}_e + K_d \dot{y}_e + K_p y_e = 0. \tag{19}$$

$\mathbb{SO}(3)$: Feedback Linearization for $\mathbb{SO}(3)$ dynamics is non-trivial. Here we use geometric PD error functions previously introduced in Lee (2012, 2011) and presented in (20). Here e_R is the configuration error akin to angle errors in the Euclidean space. Similarly, e_Ω is angular velocity error. The extra term $(R^T R_d)$ in e_Ω is called a *transport map*. Since the tangent spaces on $\mathbb{SO}(3)$ change with configuration ($\Omega \in T_R\mathbb{SO}(3)$ & $\Omega_d \in T_{R_d}\mathbb{SO}(3)$), the transport map helps project desired angular velocities onto the tangent space at the current configuration ($T_{R_d}\mathbb{SO}(3) \to T_R\mathbb{SO}(3)$) for correct comparison.

Error Functions on $\mathbb{SO}(3)$

$$e_R = \frac{1}{2}[R_d^T R - R^T R_d]^\vee \tag{20a}$$

$$e_\Omega = \Omega - (R^T R_d)\Omega_d \tag{20b}$$

$$\dot{e}_R = \underbrace{\frac{1}{2}[tr(R^T R_d)I - R^T R_d]}_{=:\Upsilon(R,R_d)} e_\Omega \tag{20c}$$

$$\dot{e}_\Omega = \dot{\Omega} + \widehat{\Omega}R^T R_d\Omega_d - R^T R_d\dot{\Omega}_d. \tag{20d}$$

Similar to equation (19), we desire an exponentially stable closed-loop error dynamics of the form

$$\dot{e}_\Omega + K_d e_\Omega + K_p e_R = 0. \tag{21}$$

Using a non-negative constant c, we define a candidate *Lyapunov function* $\mathcal{V}$ as

$$\mathcal{V} = \frac{1}{2}e_\Omega^T J e_\Omega + J K_p \Psi_s(R, R_d) + c e_R^T e_\Omega, \qquad (22)$$

where Ψ_s is an error metric on $\mathbb{SO}(3) \times \mathbb{SO}(3)$ defined earlier in (11), and for a given R_d, $\Psi_s \leq 2$. Further, the time derivative of Ψ_s is $\dot{\Psi}_s = e_R^T e_\Omega$ (for proof, see Lee et al. (2010)). Using this we can compute the time derivative of $\mathcal{V}$ as,

$$\dot{\mathcal{V}} = e_\Omega^T J \dot{e}_\Omega + J K_p e_R^T e_\Omega + c \dot{e}_R^T e_\Omega + c e_R^T \dot{e}_\Omega. \qquad (23)$$

Using equations (2), (20c), and (20d) and defining u_s as

$$u_s = \underbrace{\widehat{\Omega} J \Omega + m g \widehat{l}_c R^T e_3 - J \widehat{\Omega} R^T R_d \Omega_d + J R^T R_d \dot{\Omega}_d}_{u_s^{ff}}$$

$$\underbrace{- J K_p e_R - J K_d e_\Omega}_{u_s^{fb}}, \qquad (24)$$

we get,

$$\dot{\mathcal{V}} = -(J K_d - c\Upsilon)\|e_\Omega\|^2 - c K_p \|e_R\|^2 - c K_d e_R^T e_\Omega. \quad (25)$$

Since Υ in (20c) satisfies $\|\Upsilon\| \leq 1$, this is bounded by

$$\dot{\mathcal{V}} \leq -\eta^T W \eta, \qquad (26)$$

where $\eta = [\|e_R\|, \ \|e_\Omega\|]^T$, and the matrix $W \in \mathbb{R}^{2 \times 2}$ is

$$W = \begin{bmatrix} c K_p & -\frac{c K_d}{2} \\ -\frac{c K_d}{2} & \lambda_M(J)K_d - c \end{bmatrix}, \qquad (27)$$

with c, K_p, and K_d chosen such that W is *positive definite*. Finally, $\lambda_M(J)$ denotes the largest eigen value of J. Thus, using $\mathcal{V}$ and u_S defined in (22) and (24), we can convert error dynamics in (20d) to the desired closed-loop exponentially stable dynamics in (21). A more detailed convergence proof can be derived following the process shown in Lee et al. (2010)(Appendix B).

4. RESULTS AND DISCUSSION

The control objective in this study is to stabilize the $3D$ pendulum to its hanging equilibrium position. Euler rates in $\dot{q}_d$ are randomized for each trial with a max of 4 rad/s per state and Ω_d is determined using the $\mathcal{T}_{\dot{q}}$ mapping. The initial condition of the pendulum could be any state from the samples defined in Sec. 2.3. All the experiments are run for a fixed time T. For the pendulum model, we choose, mass as $m = 1 \ kg$, length as $l = 0.5 \ m$, and inertia w.r.t body-frame as $J = diag(0.1625, 0.1625, 0.01) \ kg \ m^2$. The controller gains are kept same for both the Euler and geometric controllers (as $K_p = 100$, $K_d = 20$) to better visualize and compare performance. We applied an Euler controller for the Euler dynamics and a geometric controller for the $\mathbb{SO}(3)$ dynamics and logged key performance metrics like input integral over time, power integral over time, max input, and max power.

How to compare the two? After applying the two controllers and logging performance data across a range of initial conditions, we compare both of them in the Euler-space. We use the maps defined in Sec. 2.2 to map $R(t)$, $\Omega(t)$, and $u_s(t)$ to $\tilde{q}(t)$, $\dot{\tilde{q}}(t)$, and $\tilde{u}_e$, respectively. Therefore,

$$\tilde{q} = \mathcal{R}^{-1}(R), \ \dot{\tilde{q}} = \mathcal{T}_{\dot{q}}^{-1}(\Omega), \ \tilde{u}_e = \mathcal{T}_u^{-1}(u_s). \qquad (28)$$

Geometric Control is more efficient:

Having logged both u_e and $\tilde{u}_e$, we plot joint-wise integrals for each experiment in Fig. 6. Note that, $\tilde{u}_\phi$ is particularly
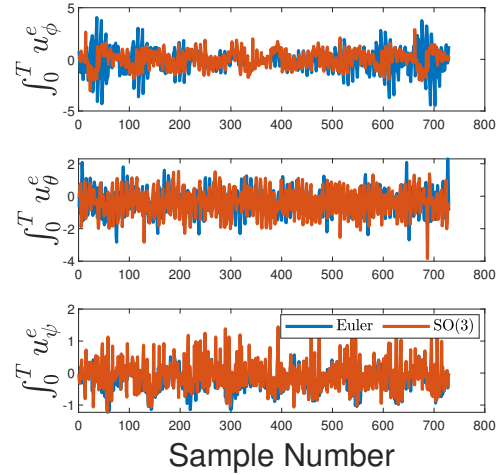

Fig. 6. Individual joint input integrals for each experiment.
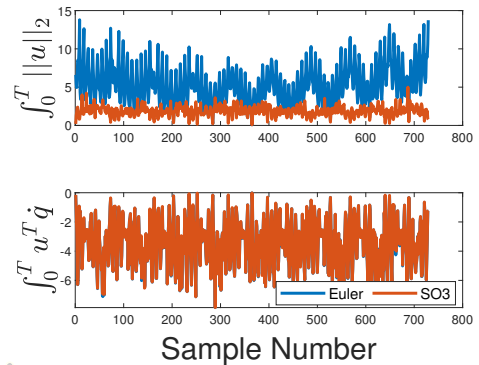

Fig. 7. The top plot shows normed input integral and the bottom plot shows power integral for each experiment. Note that *orange* points indicate $\tilde{u}_e$ and *blue* points indicate u_e.

efficient compared to u_ϕ. The existence of singularity in the ϕ direction impacts all trajectories in the Euler model with large ϕ errors.

Next, we evaluate metrics for full attitude control performance. We show normed input integral and power integral plots in Fig. 7 and max input and max power plots in Fig. 8. Note that both over time and in magnitude the geometric controllers are consistently more efficient. Also, the max input and power values for Euler controller are particularly high in some cases, mainly while crossing the singularity configuration where the numerical integrator applies arbitrarily large inputs. However, on a real system with strict input saturation this difference would be less severe.

5. CONCLUSIONS AND FUTURE WORK

In this work, we presented two formulations for modeling and control of a $3D$ pendulum - one is Euler-parameterized and the other is coordinate-free geometric formulation in the $\mathbb{SO}(3)$ manifold space. Through comprehensive empirical evaluation, we demonstrate that geometric control is generally more input efficient than the more conventional Euler-parametrized nonlinear control, not just for large error recovery as was mainly shown so far.
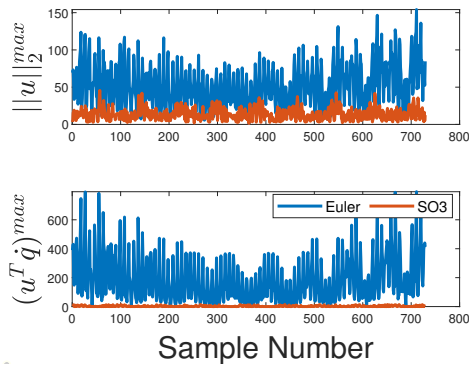
Fig. 8. The top plot shows max input norm and the bottom plot shows max power for each experiment.

We hope that this study assists in a wider adoption of coordinate-free modeling and control design beyond aerial robotics. As a part of future work, we wish to extend our empirical performance assessment studies to underactuated double pendular systems popularly studied in the legged and brachiating robot communities. Further, we also plan to develop trajectory optimization routines to discover optimal motion plans directly in the $\mathbb{SO}(3)$ space.

REFERENCES

Aguilar-Ibanez, C., Gutierrez, F.O., and Sossa-Azuela, H. (2006). Lyapunov approach for the stabilization of the inverted spherical pendulum. In *IEEE Conference on Decision and Control (CDC)*.

Åström, K.J. and Furuta, K. (2000). Swinging up a pendulum by energy control. *Automatica*, 36(2).

Bittner, B. and Sreenath, K. (2016). Symbolic computation of dynamics on smooth manifolds. In *Workshop on Algorithmic Foundations of Robotics (WAFR)*.

Chaturvedi, N., Sanyal, A.K., McClamroch, N.H., et al. (2011a). Rigid-body attitude control. *IEEE Control Systems Magazine (CSM)*, 31(3).

Chaturvedi, N.A., Lee, T., Leok, M., and McClamroch, N.H. (2011b). Nonlinear dynamics of the 3d pendulum. *Journal of Nonlinear Science*, 21(1).

Chevallereau, C., Razavi, H., Six, D., Aoustin, Y., and Grizzle, J. (2018). Self-synchronization and self-stabilization of 3d bipedal walking gaits. *Robotics and Autonomous Systems (RAS)*, 100.

Chung, C.C. and Hauser, J. (1995). Nonlinear control of a swinging pendulum. *Automatica*, 31(6).

Cunningham, D. and Asada, H.H. (2009). The winch-bot: A cable-suspended, under-actuated robot utilizing parametric self-excitation. In *IEEE International Conference on Robotics and Automation (ICRA)*.

Ding, Y., Pandala, A., and Park, H.W. (2019). Real-time model predictive control for versatile dynamic motions in quadrupedal robots. In *International Conference on Robotics and Automation (ICRA)*.

Farzan, S., Hu, A.P., Davies, E., and Rogers, J. (2019). Feedback motion planning and control of brachiating robots traversing flexible cables. In *American Control Conference (ACC)*.

Kajita, S., Kanehiro, F., Kaneko, K., Yokoi, K., and Hirukawa, H. (2001). The 3d linear inverted pendulum mode: A simple modeling for a biped walking pattern generation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 1.

Lee, T. (2011). Geometric tracking control of the attitude dynamics of a rigid body on so(3). In *American Control Conference (ACC)*.

Lee, T. (2012). Exponential stability of an attitude tracking control system on so (3) for large-angle rotational maneuvers. *Systems & Control Letters*, 61.

Lee, T. (2017). Geometric control of quadrotor uavs transporting a cable-suspended rigid body. *IEEE Transactions on Control Systems Technology*.

Lee, T., Leok, M., and McClamroch, N.H. (2010). Control of complex maneuvers for a quadrotor uav using geometric methods on se (3). *arXiv preprint arXiv:1003.2005*.

Lee, T., Leok, M., and McClamroch, N.H. (2011). Stable manifolds of saddle equilibria for pendulum dynamics on s 2 and so (3). In *IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*.

Lee, T., Leok, M., and McClamroch, N.H. (2017). *Global Formulations of Lagrangian and Hamiltonian Dynamics on Manifolds*.

Lee, T., Sreenath, K., and Kumar, V. (2013). Geometric control of cooperating multiple quadrotor uavs with a suspended payload. In *IEEE Conference on Decision and Control (CDC)*.

Lefrançois, S. and Gosselin, C. (2010). Point-to-point motion control of a pendulum-like 3-dof underactuated cable-driven robot. In *IEEE International Conference on Robotics and Automation (ICRA)*.

Lewis, D., Ratiu, T., Simo, J., and Marsden, J.E. (1992). The heavy top: a geometric treatment. *Nonlinearity*, 5(1).

Liu, G., Nešić, D., and Mareels, I. (2005). Modelling and stabilisation of a spherical inverted pendulum. *IFAC Proceedings Volumes*, 38(1).

Mueller, M.W. (2018). Multicopter attitude control for recovery from large disturbances. *arXiv preprint arXiv:1802.09143*.

Poulakakis, I. and Grizzle, J.W. (2009). The spring loaded inverted pendulum as the hybrid zero dynamics of an asymmetric hopper. *IEEE Transactions on Automatic Control*, 54(8).

Saito, F., Fukuda, T., and Arai, F. (1994). Swing and locomotion control for a two-link brachiation robot. *IEEE Control Systems Magazine (CSM)*, 14(1).

Shiriaev, A., Ludvigsen, H., and Egeland, O. (1999). Swinging up of the spherical pendulum. *IFAC Proceedings Volumes*, 32(2).

Siravuru, A., Viswanathan, S.P., Sreenath, K., and Sanyal, A.K. (2018). The reaction mass biped: Geometric mechanics and control. *Journal of Intelligent and Robotic Systems*, 89(1-2).

Sreenath, K. and Kumar, V. (2013). Dynamics, control and planning for cooperative manipulation of payloads suspended by cables from multiple quadrotor robots. *Robotics: Science and Systems (R:SS)*.

Sreenath, K. and Sanyal, A.K. (2015). The reaction mass biped: Equations of motion, hybrid model for walking and trajectory tracking control.

Zanotto, D., Rosati, G., and Agrawal, S.K. (2011). Modeling and control of a 3-dof pendulum-like manipulator. In *IEEE International Conference on Robotics and Automation (ICRA)*.