

# Exponential stability of trajectory tracking control in the orientation space utilizing unit quaternions

Leonidas Koutras and Zoe Doulgeri

**Abstract**—Trajectory tracking in the orientation space utilizing unit quaternions yields non linear error dynamics as opposed to Cartesian position. In this work, we study trajectory tracking in the orientation space utilizing the most popular quaternion error representations and angular velocity errors. By selecting error functions carefully we show exponential convergence in a region of attraction containing large initial errors. We further show that under certain conditions frequently encountered in practice, the formulation respecting the geometric characteristics of the quaternion manifold and its tangent space yields linear tracking dynamics allowing us to guarantee a desired tracking performance by gain selection without tuning. Simulation and experimental results are provided.

## I. INTRODUCTION

In many applications it is necessary for a control system to follow a desired orientation trajectory. Such applications involve the control of robotic manipulators [1], aerial and space agents [2], [3] or underwater vehicles [4], [5]. In the Cartesian position case this problem can be solved by using appropriate feedback terms in the position and velocity, generating linear system error dynamics. This is not however the case in the orientation space. Orientation space representations lie in the  $SO(3)$  group, which is a three dimensional manifold. No minimal representations of  $SO(3)$  exist, free of singularities or discontinuities [6].

A very popular representation of orientation in robotics and control is the unit quaternion [7]. Unit quaternions offer efficiency in calculation and a strong algebraic and kinematics background [8]. Orientation representations using unit quaternions are free of singularities but are not unique by construct. In fact an orientation can be described by either a unit quaternion or its negative. The uniqueness comes by constraining the unit quaternion space. However, when used to form the error between the current and the desired orientation, this constraint can lead to undesired behaviours known as unwinding [9], resulting in non-optimal paths. To alleviate it, instead of constraining the current or desired quaternion space, the error angle, in the equivalent angle axis representation should be confined in the interval  $[-\pi, \pi)$ .

As the inherent quaternion dynamics is not linear, control systems in the orientation space are not linear in general. Furthermore, the orientation error used in such controllers must be defined in  $\mathbb{R}^3$  in order to be compatible with the angular velocity and acceleration. There are three popular

orientation error definitions in  $\mathbb{R}^3$  that have been proposed for quaternion feedback. Initially an error inspired by the angle axis representation is proposed in [10] and still used in recent works [11]. In [12] the utilization of the vector part of the quaternion error is proposed, and has been largely adopted by the robotics and control community [13]–[15]. In recent years, the usage of another quaternion error is gaining popularity [16]–[20] that takes into account the logarithmic mapping from the unit quaternion manifold to the tangent space of angular velocity vectors [21], [22].

Moreover, there exist two different angular velocity error formulations. Earlier works use the error formed from the difference of the desired and the current angular velocity [7], [12], [23]. We will show that this error selection leads to non-autonomous closed loop dynamics. The local convergence properties in this case combined with the vector part of the quaternion error is studied in [24]. More recent works [14], [25]–[28] utilize a different angular velocity error, exploiting the geometry of the quaternion manifold.

The majority of related works in the literature, include proofs of the asymptotic stability of error dynamics. Exponential stability to the best of our knowledge is only addressed in [27] for an attitude tracking control system utilizing rotation matrices. In this work, we will prove exponential stability of the error dynamics with constant control gains in all formulations. This is in contrast to the claim made in some previous works that some cases require time varying gains e.g. [12] for the orientation error proposed in [10] and [29] for the logarithmic error. In contrast to asymptotic stability, exponential stability offers information about the system's convergence rate resulting in more intuitive system design. Furthermore, we will show that under certain conditions frequently encountered in practice, the formulation respecting the geometric characteristics of the quaternion manifold and its tangent space yields linear tracking dynamics; hence transient specifications like convergence rate or avoiding overshoot can be satisfied by appropriate gain selection without tuning. This linear behavior offers predictability which can be vital in high precision tasks.

## II. UNIT QUATERNIONS

A unit quaternion  $\mathbf{Q} = [\eta \ \epsilon^T]^T$  is given by a scalar part  $\eta$  and a three dimensional vector part  $\epsilon$  referred to as the real and imaginary part respectively; unit quaternions expressed by 4-dimensional vectors belong to a sphere of dimension 3 i.e.  $\mathbf{Q} \in \mathcal{S}^3$ . Unit quaternions are the minimal singularity free representation of the orientation of a rigid body expressed by the rotation matrix  $\mathbf{R} \in SO(3)$ . The unit

Authors are with the Automation & Robotics Lab, Dept. of Electrical & Computer Engineering, Aristotle University of Thessaloniki, Greece {kleonidas, doulgeri}@ece.auth.gr

The research leading to these results has received funding from the European Community's Framework Programme Horizon 2020 under grant agreement No 871704, project BACCHUS.

quaternion space  $\mathcal{S}^3$  is mapped to  $SO(3)$  by the mapping:

$$\mathbf{R} = \mathbf{I}_3 + 2\eta\mathbf{S}(\epsilon) + 2\mathbf{S}^2(\epsilon) \quad (1)$$

where  $\mathbf{I}_3$  is the identity matrix of dimension 3 and  $\mathbf{S}(\mathbf{x})$  the skew symmetric matrix constructed by a vector  $\mathbf{x}$ :

$$\mathbf{S}(\mathbf{x}) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}$$

Unit quaternions are related to the equivalent axis  $\mathbf{n}$  and rotation angle  $\theta$  in the following way:

$$\mathbf{Q} = [\eta \quad \epsilon^T]^T = [\cos \frac{\theta}{2} \quad \sin \frac{\theta}{2} \mathbf{n}^T]^T \quad (2)$$

The product of two unit quaternions is also a unit quaternion and is given by:

$$\mathbf{Q}_1 * \mathbf{Q}_2 = \begin{bmatrix} \eta_1 & -\epsilon_1^T \\ \epsilon_1 & \eta_1 \mathbf{I}_3 + \mathbf{S}(\epsilon_1) \end{bmatrix} \mathbf{Q}_2 \quad (3)$$

The inverse of a unit quaternion coincides with its conjugate  $\bar{\mathbf{Q}} = [\eta \quad -\epsilon^T]^T$ .

The unit quaternion space  $\mathcal{S}^3$  is a three dimensional Riemannian manifold. At each point on this manifold corresponds a tangent space of angular velocity vectors. The manifold  $\mathcal{S}^3$  forms a Lie group endowed with the quaternion multiplication (3) [30]. An efficient way to transition between the manifold and its tangent space is the utilization of distance preserving mappings that take into account their geometric characteristics. These are the exponential and logarithmic mappings between a unit quaternion  $\mathbf{Q} \in \mathcal{S}^3$  and a tangent vector  $\mathbf{x} \in \mathbb{R}^3$  stemming from the Lie Algebra of  $\mathcal{S}^3$  and are defined as follows:

$$\exp \mathbf{x} = \begin{cases} \begin{bmatrix} \cos \|\mathbf{x}\| & \sin \|\mathbf{x}\| \frac{\mathbf{x}^T}{\|\mathbf{x}\|} \\ 1 & 0 \end{bmatrix}^T, & \|\mathbf{x}\| \neq 0 \\ \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}^T, & \|\mathbf{x}\| = 0 \end{cases} \quad (4)$$

$$\log \mathbf{Q} = \begin{cases} \frac{\cos^{-1} \eta}{\|\epsilon\|} \epsilon, & \|\epsilon\| \neq 0 \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}^T, & \|\epsilon\| = 0 \end{cases} \quad (5)$$

The quaternion logarithm can also be written as a function of the axis and angle through (2):

$$\log \mathbf{Q} = \begin{cases} \mathbf{n} \frac{\theta}{2}, & \theta \neq 0 \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}^T, & \theta = 0 \end{cases} \quad (6)$$

#### A. Unit Quaternion Kinematics

The derivative of a unit Quaternion is associated with the angular velocity expressed in the inertia frame by:

$$\dot{\mathbf{Q}} = \frac{1}{2} \mathbf{J}_Q(\mathbf{Q}) \boldsymbol{\omega} \quad (7)$$

with

$$\mathbf{J}_Q(\mathbf{Q}) = \begin{bmatrix} -\epsilon^T \\ \eta \mathbf{I}_3 - \mathbf{S}(\epsilon) \end{bmatrix} \quad (8)$$

It is easy to establish some properties for matrix  $\mathbf{J}_Q(\mathbf{Q})$  (8):

$$\mathbf{J}_Q^T \mathbf{J}_Q = \mathbf{I}_3 \quad \mathbf{J}_Q^T \mathbf{Q} = \mathbf{0} \quad (9)$$

Hence the columns of matrix  $\mathbf{J}_Q(\mathbf{Q})$  form an orthogonal base of the tangent space of  $\mathcal{S}^3$  at  $\mathbf{Q}$ .

Let  $\tilde{\mathbf{Q}} \in \mathcal{S}^3$  denote the error between a desired quaternion  $\mathbf{Q}_d$  and the current quaternion  $\mathbf{Q}$  along the geodesic connecting the two points expressed in the inertia frame:

$$\tilde{\mathbf{Q}} = \mathbf{Q} * \bar{\mathbf{Q}}_d = [\tilde{\eta} \quad \tilde{\epsilon}^T]^T \quad (10)$$

To ensure that the quaternion error represents the minimal rotation we constrain its scalar part  $\tilde{\eta} \geq 0$  resulting through (2) to error angles in  $[-\pi, \pi)$ , since  $\tilde{\eta} = \cos \frac{\tilde{\theta}}{2}$ .

Differentiating  $\mathbf{Q} * \bar{\mathbf{Q}}_d$  in (10) using (7), (8) yields the derivatives of the quaternion error's scalar and vector parts:

$$\dot{\tilde{\eta}} = -\frac{1}{2} \tilde{\epsilon}^T (\boldsymbol{\omega} - \boldsymbol{\omega}_d) \quad (11)$$

$$\dot{\tilde{\epsilon}} = \frac{1}{2} (\tilde{\eta} \mathbf{I}_3 - \mathbf{S}(\tilde{\epsilon})) (\boldsymbol{\omega} - \boldsymbol{\omega}_d) + \mathbf{S}(\boldsymbol{\omega}_d) \tilde{\epsilon} \quad (12)$$

Equations (11) and (12) can also be written as:

$$\dot{\tilde{\eta}} = -\frac{1}{2} \tilde{\epsilon}^T (\boldsymbol{\omega} - \tilde{\mathbf{R}} \boldsymbol{\omega}_d) \quad (13)$$

$$\dot{\tilde{\epsilon}} = \frac{1}{2} (\tilde{\eta} \mathbf{I}_3 - \mathbf{S}(\tilde{\epsilon})) (\boldsymbol{\omega} - \tilde{\mathbf{R}} \boldsymbol{\omega}_d) \quad (14)$$

and in compact form:

$$\dot{\tilde{\mathbf{Q}}} = \frac{1}{2} \mathbf{J}_Q(\tilde{\mathbf{Q}}) (\boldsymbol{\omega} - \tilde{\mathbf{R}} \boldsymbol{\omega}_d) \quad (15)$$

where  $\tilde{\mathbf{R}}$  is derived from  $\tilde{\mathbf{Q}}$  through (1). Notice the similarity of (15) with (7). Hence  $\boldsymbol{\omega} - \tilde{\mathbf{R}} \boldsymbol{\omega}_d$  represents the angular velocity on the tangent space of  $\mathcal{S}^3$  at  $\tilde{\mathbf{Q}}$ .

### III. TRACKING CONTROL VARIANTS

The control objective is the tracking of a desired orientation trajectory  $\mathbf{Q}_d$ ,  $\boldsymbol{\omega}_d$ ,  $\dot{\boldsymbol{\omega}}_d$ . Consider systems that can be put in the form:

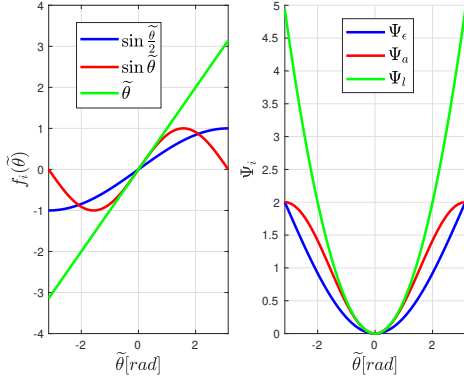
$$\dot{\boldsymbol{\omega}} = \mathbf{u} \quad (16)$$

where  $\dot{\boldsymbol{\omega}}$  is the system's angular acceleration and  $\mathbf{u}$  is the control input. Equation (16) expresses the second order kinematics of a rigid body rotational motion. The respective rigid body dynamics (e.g. a robot's end-effector) can further take this form via an inner inverse dynamics control loop. In the literature, two types of control structures have been proposed. They differ with respect to the angular velocity error they utilize. These are:

$$\tilde{\boldsymbol{\omega}}_E \triangleq \boldsymbol{\omega} - \boldsymbol{\omega}_d \quad (17)$$

$$\tilde{\boldsymbol{\omega}}_G \triangleq \boldsymbol{\omega} - \tilde{\mathbf{R}} \boldsymbol{\omega}_d \quad (18)$$

The first is the trivial case, employing the Euclidean difference between the current and desired angular velocity [7], [12], [23]. The second angular velocity error represents the angular velocity at  $\tilde{\mathbf{Q}}$  as discussed in the end of Section II, which takes into account the geometry of  $\mathcal{S}^3$  [14], [25]–[28]. Matrix  $\tilde{\mathbf{R}}$  in (18) operates as a transport map which allows velocities belonging to different tangent spaces to be compared [31]. In contrast, the angular velocities  $\boldsymbol{\omega}$  and



**Fig. 1:** Scalar function of the error angle  $\tilde{\theta}$  for each quaternion orientation error and  $\Psi_i$  function.

$\omega_d$  in (17) only belong in the same tangent space for small errors. The respective control inputs are given by:

$$\mathbf{u}_E = \dot{\omega}_d - d\tilde{\omega}_E - \kappa \mathbf{e}_i \quad (19)$$

$$\mathbf{u}_G = \frac{d(\tilde{\mathbf{R}}\omega_d)}{dt} - d\tilde{\omega}_G - \kappa \mathbf{e}_i \quad (20)$$

where  $d, \kappa > 0$  are positive damping and stiffness gains and  $\mathbf{e}_i$  is an expression of the orientation error in  $\mathbb{R}^3$ . Notice that  $\tilde{\mathbf{R}}$  involved in  $\frac{d(\tilde{\mathbf{R}}\omega_d)}{dt}$  in (20) is given by  $\tilde{\mathbf{R}} = \mathbf{S}(\tilde{\omega}_G)\tilde{\mathbf{R}}$ .

To construct a control system on the manifold of  $\mathcal{S}^3$ , one needs to define an appropriate orientation error function  $\Psi : \mathcal{S}^3 \times \mathcal{S}^3 \rightarrow \mathbb{R}$ , which is positive definite and measures the error between two orientations [32]. The orientation and angular velocity errors emerge by differentiating  $\Psi$ . Related to the three choices of  $\mathbf{e}_i$  suggested in the literature we select the following orientation error functions  $\Psi_i$ :

- The vector part of  $\tilde{\mathbf{Q}}$  [1], [12], [14], [15]:

$$\mathbf{e}_\epsilon \triangleq \tilde{\boldsymbol{\epsilon}} = \sin \frac{\tilde{\theta}}{2} \tilde{\mathbf{n}} \quad (21)$$

$$\Psi_\epsilon \triangleq 2(1 - \tilde{\eta}) \quad (22)$$

- Double the product of the scalar and the vector part of  $\tilde{\mathbf{Q}}$  [10], [11]:

$$\mathbf{e}_a \triangleq 2\tilde{\eta}\tilde{\boldsymbol{\epsilon}} = \sin \tilde{\theta} \tilde{\mathbf{n}} \quad (23)$$

$$\Psi_a \triangleq 2\tilde{\epsilon}^T \tilde{\boldsymbol{\epsilon}} \quad (24)$$

- Double the quaternion logarithm (6) of  $\tilde{\mathbf{Q}}$  [16]–[19]:

$$\mathbf{e}_l \triangleq 2 \log \tilde{\mathbf{Q}} = \tilde{\theta} \tilde{\mathbf{n}} \quad (25)$$

$$\Psi_l \triangleq \frac{1}{2} \tilde{\theta}^2 \quad (26)$$

For every orientation error, it is easy to prove that the derivative of the respective orientation error function yields:

$$\dot{\Psi}_i = \mathbf{e}_i^T \tilde{\omega}_G = \mathbf{e}_i^T \tilde{\omega}_E \quad \forall i = \{\epsilon, a, l\} \quad (27)$$

Notice that each orientation error is a product of the error rotation axis  $\tilde{\mathbf{n}}$  with a scalar function of the error angle  $f_i(\tilde{\theta})$  that is depicted in Figure 1, along with function  $\Psi_i$ .

The vector part of  $\tilde{\mathbf{Q}}$  (21), proposed in [12] is the most popular to this day, as it facilitates Lyapunov stability analysis of control systems. However, it yields slower response than other selections [16]. The second error (23) is continuous when the error angle crosses  $\pm\pi$  as opposed to the other choices but it is a non-monotonous function of the error angle as shown in Figure 1. Hence, its behavior deteriorates when angle errors are larger than  $\frac{\pi}{2}$  [27]. Moreover  $\mathbf{e}_a = \mathbf{0}$  implies  $\tilde{\boldsymbol{\epsilon}} = \mathbf{0}$ , (which in turn implies  $\mathbf{Q} = \mathbf{Q}_d$ ) or  $\tilde{\eta} = 0$ ; when  $\tilde{\eta} = 0$  ( $\tilde{\theta} = -\pi$ ),  $\tilde{\boldsymbol{\epsilon}} \neq \mathbf{0}$  and hence  $\mathbf{Q}$  does not track  $\mathbf{Q}_d$ . The logarithmic error (25) is gaining popularity in recent research works [16]–[19] as it offers a more mathematically sound approach utilizing the Lie Algebra of  $\mathcal{S}^3$  by employing the distance preserving logarithmic mapping.

Differentiating the orientation error using (11)–(14) we get:

$$\dot{\mathbf{e}}_i = \mathbf{J}_i \tilde{\omega}_G \quad (28)$$

$$\dot{\mathbf{e}}_i = \mathbf{J}_i \tilde{\omega}_E - \mathbf{S}(\mathbf{e}_i) \omega_d \quad (29)$$

where matrix  $\mathbf{J}_i$  is given for each error selection in Appendix. System (16) with control inputs (19), (20) yield the following closed loop tracking systems:

$$\dot{\tilde{\omega}}_E + d\tilde{\omega}_E + \kappa \mathbf{e}_i = \mathbf{0} \quad (30)$$

$$\dot{\tilde{\omega}}_G + d\tilde{\omega}_G + \kappa \mathbf{e}_i = \mathbf{0} \quad (31)$$

In the following sections, we express (30) and (31) in state space, find their equilibria, show the exponential convergence to the origin, determine the region of attraction and indicate conditions that render the tracking system linear and thus allow gain selection that satisfy predefined specifications.

#### IV. STABILITY ANALYSIS

Define the state space vectors for  $i = \{\epsilon, a, l\}, j = \{G, E\}$ :

$$\boldsymbol{\xi}_i^j = [\mathbf{e}_i^T \quad \tilde{\omega}_j^T]^T \quad (32)$$

In state space form the system is written as:

$$\dot{\boldsymbol{\xi}}_i^j = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{J}_i \\ -\kappa \mathbf{I}_3 & -d \mathbf{I}_3 \end{bmatrix} \boldsymbol{\xi}_i^j \quad (33)$$

where  $\mathbf{A}_{11} = \mathbf{0}$  for  $j = G$  and  $\mathbf{A}_{11} = \mathbf{S}(\omega_d)$  for  $j = E$ . Notice that due to the time dependant term  $\mathbf{S}(\omega_d)$ , system (30), with angular velocity error  $\tilde{\omega}_E$  is non-autonomous.

**Proposition 1.** *In the region  $\tilde{\theta} \in [-\pi, \pi)$  the origin  $\boldsymbol{\xi}_i^j = \mathbf{0}$  is a unique equilibrium of (33) for  $i = \{\epsilon, a, l\}, j = \{G, E\}$*

*Proof.* Setting  $\tilde{\omega}_j = \mathbf{0}$ , the second equation of (33) yields  $\tilde{\omega}_j = -\frac{\kappa}{d} \mathbf{e}_i$ . For  $i = \epsilon$  setting  $\tilde{\boldsymbol{\epsilon}} = \mathbf{0}$  implies  $\tilde{\eta} = 0$  owing to the unit quaternion norm constraint. Substituting the above in (11) or (13) implies  $-\frac{\kappa}{d} \tilde{\epsilon}^T \tilde{\boldsymbol{\epsilon}} = 0$  and in turn  $\boldsymbol{\xi}_\epsilon^j = \mathbf{0}$ . For  $i = l$ , setting  $\tilde{\boldsymbol{\epsilon}}_l = \mathbf{0}$ , the first equation of (33), using  $\mathbf{J}_l \mathbf{e}_l = \mathbf{e}_l$  from the Appendix, yields  $\mathbf{e}_l = \mathbf{0}$  for  $j = G$  and  $(-\frac{\kappa}{d} \mathbf{I}_3 + \mathbf{S}(\omega_d)) \mathbf{e}_l = \mathbf{0}$  for  $j = E$ . Notice that matrix  $(-\frac{\kappa}{d} \mathbf{I}_3 + \mathbf{S}(\omega_d))$  is full rank, thus the origin is the only equilibrium of (33) for  $i = l$ . For  $i = a$ , as before we get  $\mathbf{J}_a \mathbf{e}_a = \mathbf{0}$  for  $j = G$ ; substituting  $\mathbf{J}_a$  from (55) and  $\mathbf{e}_a$  from (23) we get after some calculations

$\mathbf{J}_a \mathbf{e}_a = (2\tilde{\eta}^2 - 1)\mathbf{e}_a = \mathbf{0}$ . For  $j = E$ , we find respectively  $(-\frac{\kappa}{d}(2\tilde{\eta}^2 - 1)\mathbf{I}_3 + \mathbf{S}(\boldsymbol{\omega}_d))\mathbf{e}_a = \mathbf{0}$ . Notice that the expression in the parenthesis is zero for the value of  $\tilde{\eta} = \frac{1}{\sqrt{2}}$  which implies  $\tilde{\boldsymbol{\omega}}_j = -\frac{\sqrt{2}\kappa}{d}\tilde{\boldsymbol{\epsilon}}$ . Substituting this result in (11) or (13) yields  $\dot{\tilde{\eta}} = \frac{\kappa}{\sqrt{2}d}\tilde{\boldsymbol{\epsilon}}^T \tilde{\boldsymbol{\epsilon}} \neq 0$  and hence the solution cannot stay at this value. Thus, in the above identities  $\mathbf{e}_a = \mathbf{0}$  is the only solution. Notice that  $\mathbf{e}_a = \mathbf{0}$  does not guarantee tracking as discussed in Section III for  $\tilde{\eta} = 0$ . It is however possible to show that this is not a stable state.  $\square$

**Theorem 1.** *The origin  $\boldsymbol{\xi}_i^j = \mathbf{0}$  of (33) for  $i = \{\epsilon, a, l\}, j = \{G, E\}$  is exponentially stable and an estimate of its the region of attraction is:*

$$\Psi_i(0) < 2, i = \{\epsilon, a\}, \quad \Psi_l(0) < \frac{1}{2}\pi^2 \quad (34)$$

$$\|\tilde{\boldsymbol{\omega}}_j(0)\|^2 < \begin{cases} 2\kappa(2 - \Psi_i(0)), i = \{\epsilon, a\} \\ 2\kappa(\frac{1}{2}\pi^2 - \Psi_i(0)), i = l \end{cases} \quad (35)$$

*Proof.* We will initially prove that (34), (35) is an invariant set. The inequality (34) holds since  $\theta \in (-\pi, \pi)$ . Consider the function:

$$W(\boldsymbol{\xi}_i^j) = \kappa\Psi_i + \frac{1}{2}\tilde{\boldsymbol{\omega}}_j^T \tilde{\boldsymbol{\omega}}_j \quad (36)$$

It is easy to show that  $\dot{W} = -d\tilde{\boldsymbol{\omega}}_j^T \tilde{\boldsymbol{\omega}}_j \leq 0$ , which implies  $W(t) \leq W(0)$ . Using (34), (35),  $W(0)$  is bounded by  $W(0) < 2\kappa$  for  $i = \{\epsilon, a\}$  and  $W(0) < \frac{1}{2}\kappa\pi^2$  for  $i = l$ . Since,  $\kappa\Psi_i(t) \leq W(t) \leq W(0)$ ,  $\Psi_i(t) < 2$  for  $i = \{\epsilon, a\}$  and  $\Psi_i(t) < \frac{1}{2}\pi^2$  for  $i = l \forall t > 0$  which means that the orientation errors are well defined.

Define the candidate Lyapunov function:

$$V(\boldsymbol{\xi}_i^j) = \kappa\Psi_i + \frac{1}{2}\tilde{\boldsymbol{\omega}}_j^T \tilde{\boldsymbol{\omega}}_j + c\mathbf{e}_i^T \tilde{\boldsymbol{\omega}}_j \quad (37)$$

Notice that:

$$\boldsymbol{\zeta}_i^{jT} \mathbf{L}_i \boldsymbol{\zeta}_i^j \leq V \leq \boldsymbol{\zeta}_i^{jT} \mathbf{U}_i \boldsymbol{\zeta}_i^j \quad (38)$$

where

$$\boldsymbol{\zeta}_i^j = [\|\mathbf{e}_i\| \quad \|\tilde{\boldsymbol{\omega}}_j\|]^T \quad (39)$$

and

$$\mathbf{L}_i = \begin{bmatrix} L_{11} & \frac{-c}{2} \\ \frac{-c}{2} & \frac{1}{2} \end{bmatrix} \quad \mathbf{U}_i = \begin{bmatrix} U_{11} & \frac{c}{2} \\ \frac{c}{2} & \frac{1}{2} \end{bmatrix} \quad (40)$$

where  $L_{11} = \kappa, U_{11} = 2\kappa$  for  $i = \epsilon$ ,  $L_{11} = U_{11} = \frac{\kappa}{2\tilde{\eta}^2}$  for  $i = a$  and  $L_{11} = U_{11} = \frac{\kappa}{2}$  for  $i = l$ . Notice that (34) implies  $\tilde{\eta} > 0$ . Matrices  $\mathbf{L}_i, \mathbf{U}_i$  are positive definite when  $c < \sqrt{2}\kappa$ , for  $i = \epsilon$  and  $c < \sqrt{\kappa}$ , for  $i = \{a, l\}$ . Differentiating (37) for  $j = G$ , using (31), (27) yields:

$$\begin{aligned} \dot{V} &= -d\tilde{\boldsymbol{\omega}}_G^T \tilde{\boldsymbol{\omega}}_G - c\kappa\mathbf{e}_i^T \mathbf{e}_i - c d\mathbf{e}_i^T \tilde{\boldsymbol{\omega}}_G + c\tilde{\boldsymbol{\omega}}_G^T \mathbf{J}_i \tilde{\boldsymbol{\omega}}_G \leq \\ &(-d + c\|\mathbf{J}_i\|)\|\tilde{\boldsymbol{\omega}}_G\|^2 - c\kappa\|\mathbf{e}_i\|^2 + cd\|\tilde{\boldsymbol{\omega}}_G\|\|\mathbf{e}_i\| \leq \\ &-\boldsymbol{\zeta}_i^{GT} \mathbf{M}_i^G \boldsymbol{\zeta}_i^G \end{aligned} \quad (41)$$

Differentiating (37) for  $j = E$  using (30), (27) yields:

$$\begin{aligned} \dot{V} &= -d\tilde{\boldsymbol{\omega}}_E^T \tilde{\boldsymbol{\omega}}_E - c\kappa\mathbf{e}_i^T \mathbf{e}_i - c d\mathbf{e}_i^T \tilde{\boldsymbol{\omega}}_E + c\tilde{\boldsymbol{\omega}}_E^T \mathbf{J}_i \tilde{\boldsymbol{\omega}}_E + \\ &c\tilde{\boldsymbol{\omega}}_E^T \mathbf{S}(\boldsymbol{\omega}_d)\mathbf{e}_i \leq (-d + c\|\mathbf{J}_i\|)\|\tilde{\boldsymbol{\omega}}_E\|^2 - c\kappa\|\mathbf{e}_i\|^2 + \\ &c(d + B)\|\tilde{\boldsymbol{\omega}}_E\|\|\mathbf{e}_i\| \leq -\boldsymbol{\zeta}_i^{ET} \mathbf{M}_i^E \boldsymbol{\zeta}_i^E \end{aligned} \quad (42)$$

where  $B \geq \|\boldsymbol{\omega}_d\|$  is the maximum norm value of the desired angular velocity and  $\mathbf{M}_i^j$  are given by:

$$\mathbf{M}_i^G = \begin{bmatrix} c\kappa & -\frac{cd}{2} \\ -\frac{cd}{2} & d - c\|\mathbf{J}_i\| \end{bmatrix} \quad \mathbf{M}_i^E = \begin{bmatrix} c\kappa & -c\frac{d+B}{2} \\ -c\frac{d+B}{2} & d - c\|\mathbf{J}_i\| \end{bmatrix} \quad (43)$$

Using (60), matrices (43) are positive definite when  $c < \frac{4\kappa d}{2\kappa p_i + q_j^2}$ , with  $p_\epsilon = 1, p_a = 2, p_l = \pi$  and  $q_G = d, q_E = d + B$ . Since there exists a value for  $c$  that satisfies this constraint, the origin is exponentially stable [33], with:

$$\|\boldsymbol{\zeta}_i^j(t)\| \leq \sqrt{\frac{\lambda_{\max}(\mathbf{U}_i)}{\lambda_{\min}(\mathbf{L}_i)}} \exp\left(-\frac{\lambda_{\min}(\mathbf{M}_i^j)}{2\lambda_{\max}(\mathbf{U}_i)}t\right) \|\boldsymbol{\zeta}_i^j(0)\| \quad (44)$$

**Remark 1.** *We will now show that  $\tilde{\eta} = 0$  is not a stable state for  $i = a$ . Consider the set:*

$$\boldsymbol{\xi}_S = \{\mathbf{e}_a \in \mathbb{R}^3, \tilde{\boldsymbol{\omega}}_j \in \mathbb{R}^3 \mid \tilde{\eta} = 0, \tilde{\boldsymbol{\omega}}_j = \mathbf{0}\} \quad (45)$$

We have already shown that for function (36)  $\dot{W} \leq 0$ . Notice that  $W(\boldsymbol{\xi}_S) = 2\kappa$ , since  $\|\tilde{\boldsymbol{\epsilon}}\| = 1$  in this case. Assume a point  $\boldsymbol{\xi}_\sigma$  emerging from a small disturbance  $\tilde{\eta} = \sigma$ . Then,  $W(\boldsymbol{\xi}_\sigma) = 2\kappa(1 - \sigma^2) < W(\boldsymbol{\xi}_S)$ . Thus if  $\boldsymbol{\xi}_S$  is perturbed,  $\boldsymbol{\xi}_a^j$  will never return to  $\boldsymbol{\xi}_S$ , rendering it unstable.

## V. TRACKING SYSTEM LINEARITY CONDITIONS AND GAIN SELECTION

We will now show that controller (20) utilizing the angular velocity error  $\tilde{\boldsymbol{\omega}}_G$  (18) with the logarithmic orientation error  $\mathbf{e}_l$  (25) yields linear tracking error dynamics with certain initial conditions. The latter is typical in many applications. We will further show that for small error angles the system remains linear, regardless of the initial conditions, even in the presence of disturbances. This property is useful considering a tracking system at steady state under modelling errors and measurement noise. In these cases, one can select control gain values for  $\kappa, d$  that satisfy specifications about the speed of response and its damping ratio (e.g being critically damped) or even satisfy a desired compliant behaviour under external forces.

**Proposition 2.** *System (31) with the geometric angular velocity error and the logarithmic orientation error  $\mathbf{e}_l$  (25) yields linear tracking error dynamics in case the initial angular velocity error  $\tilde{\boldsymbol{\omega}}_G(0)$  is zero or the initial orientation error  $\mathbf{e}_l(0)$  is zero.*

*Proof.* Using (15), (9) and (2) we get:

$$\tilde{\boldsymbol{\omega}}_G = 2\mathbf{J}_Q^T(\tilde{\mathbf{Q}})\dot{\tilde{\mathbf{Q}}} = \dot{\tilde{\theta}}\tilde{\mathbf{n}} + \sin\tilde{\theta}\dot{\tilde{\mathbf{n}}} + (1 - \cos\tilde{\theta})\mathbf{S}(\tilde{\mathbf{n}})\dot{\tilde{\mathbf{n}}} \quad (46)$$

Notice that (46) has three orthogonal components, since  $\dot{\tilde{\mathbf{n}}}$  is the derivative of the unit vector  $\tilde{\mathbf{n}}$  and is orthogonal to it by construct, and  $\mathbf{S}(\tilde{\mathbf{n}})\dot{\tilde{\mathbf{n}}}$  represents the cross product between  $\tilde{\mathbf{n}}$  and  $\dot{\tilde{\mathbf{n}}}$ . Differentiating (25) yields:

$$\dot{\mathbf{e}}_l = \dot{\tilde{\theta}}\tilde{\mathbf{n}} + \tilde{\theta}\dot{\tilde{\mathbf{n}}} \quad (47)$$

Notice from (46) and (47) that when  $\dot{\tilde{\mathbf{n}}} = \mathbf{0}$ , the geometric angular velocity error is equal to the derivative of the logarithmic orientation error. Define the projection matrices  $\mathbf{P}_{\parallel} = \tilde{\mathbf{n}}\tilde{\mathbf{n}}^T$ ,  $\mathbf{P}_{\perp} = \mathbf{I}_3 - \tilde{\mathbf{n}}\tilde{\mathbf{n}}^T$ . Projecting (46) yields:

$$\mathbf{P}_{\parallel}\tilde{\boldsymbol{\omega}}_G = \dot{\tilde{\theta}}\tilde{\mathbf{n}} \quad (48)$$

$$\mathbf{P}_{\perp}\tilde{\boldsymbol{\omega}}_G = \sin\tilde{\theta}\dot{\tilde{\mathbf{n}}} + (1 - \cos\tilde{\theta})\mathbf{S}(\tilde{\mathbf{n}})\dot{\tilde{\mathbf{n}}} \quad (49)$$

Differentiating (48) and (49) using (31), with  $i = l$  and (46) yields:

$$\frac{d}{dt}(\mathbf{P}_{\parallel}\tilde{\boldsymbol{\omega}}_G) = (\|\dot{\tilde{\mathbf{n}}}\|^2 \sin\tilde{\theta} - d\dot{\tilde{\theta}} - \kappa\tilde{\theta})\tilde{\mathbf{n}} + \dot{\tilde{\theta}}\dot{\tilde{\mathbf{n}}} \quad (50)$$

$$\begin{aligned} \frac{d}{dt}(\mathbf{P}_{\perp}\tilde{\boldsymbol{\omega}}_G) = & -d(\sin\tilde{\theta}\dot{\tilde{\mathbf{n}}} + (1 - \cos\tilde{\theta})\mathbf{S}(\tilde{\mathbf{n}})\dot{\tilde{\mathbf{n}}}) - \\ & \|\dot{\tilde{\mathbf{n}}}\|^2 \sin\tilde{\theta}\tilde{\mathbf{n}} - \dot{\tilde{\theta}}\dot{\tilde{\mathbf{n}}} \end{aligned} \quad (51)$$

Let the initial orientation error be  $\mathbf{e}_l(0) = \tilde{\theta}(0)\tilde{\mathbf{n}}(0)$  and the initial angular velocity error  $\tilde{\boldsymbol{\omega}}_G(0) = \mathbf{0}$ . Given that  $\tilde{\theta}(0) \neq 0$ , the initial condition  $\tilde{\boldsymbol{\omega}}_G(0) = \mathbf{0}$  implies from (46) that  $\dot{\tilde{\mathbf{n}}}(0) = \mathbf{0}$ ,  $\dot{\tilde{\theta}}(0) = 0$ . Hence from (50) and (51) we deduce that  $\frac{d}{dt}(\mathbf{P}_{\perp}\tilde{\boldsymbol{\omega}}_G)$  remains zero and  $\tilde{\boldsymbol{\omega}}_G$  evolves along the direction of  $\tilde{\mathbf{n}}(0)$  implying  $\dot{\tilde{\mathbf{n}}} = \mathbf{0}$ .

Consider now  $\mathbf{e}_l(0) = \mathbf{0}$  and  $\tilde{\boldsymbol{\omega}}_G(0) = \tilde{\boldsymbol{\omega}}_0 \neq \mathbf{0}$ . Zero initial orientation error implies  $\tilde{\theta}(0) = 0$ , which from (46) implies that  $\tilde{\boldsymbol{\omega}}_0 = \dot{\tilde{\theta}}(0)\tilde{\mathbf{n}}(0)$ . From (47) we get  $\dot{\mathbf{e}}_l(0) = \dot{\tilde{\theta}}(0)\tilde{\mathbf{n}}(0)$ , thus the orientation error tends to evolve in the direction of  $\tilde{\mathbf{n}}(0)$ , implying  $\dot{\tilde{\mathbf{n}}}(0) = \mathbf{0}$ . Hence again from (50) and (51) we again deduce that  $\frac{d}{dt}(\mathbf{P}_{\perp}\tilde{\boldsymbol{\omega}}_G)$  remains zero and  $\tilde{\boldsymbol{\omega}}_G$  evolves along the direction of  $\tilde{\mathbf{n}}(0)$  implying  $\dot{\tilde{\mathbf{n}}} = \mathbf{0}$ .

From the above analysis, we deduce that in both cases, the system is linear as  $\dot{\mathbf{e}}_l = \tilde{\boldsymbol{\omega}}_G = \dot{\tilde{\theta}}\tilde{\mathbf{n}}$ .  $\square$

**Remark 2.** System (31) with orientation errors  $\mathbf{e}_e$  (21) or  $\mathbf{e}_a$  (23) yields an angular velocity error  $\tilde{\boldsymbol{\omega}}_G$  along  $\tilde{\mathbf{n}}(0)$  following the same analysis performed in the above proof. However the resulting dynamics is not linear as the orientation error derivatives are  $\dot{\mathbf{e}}_e = \frac{\dot{\tilde{\theta}}}{2} \cos \frac{\tilde{\theta}}{2} \tilde{\mathbf{n}} + \sin \frac{\tilde{\theta}}{2} \dot{\tilde{\mathbf{n}}}$  and  $\dot{\mathbf{e}}_a = \dot{\tilde{\theta}} \cos \tilde{\theta} \tilde{\mathbf{n}} + \sin \tilde{\theta} \dot{\tilde{\mathbf{n}}}$  respectively.

We will now consider system (31) with a disturbance input  $\mathbf{u}_{Dist}$ :

$$\dot{\tilde{\boldsymbol{\omega}}}_G + d\tilde{\boldsymbol{\omega}}_G + \kappa\mathbf{e}_i = \mathbf{u}_{Dist} \quad (52)$$

Consider small disturbances which can occur by unmodelled environment dynamics. Such disturbances induce small errors to the system, disrupting the linear behavior of the above analysis. However, for small error angles  $\tilde{\theta} < \frac{\pi}{6}$  the approximations  $\sin\tilde{\theta} \approx \tilde{\theta}$  and  $\cos\tilde{\theta} \approx 1$  hold, thus (46) can be written as:

$$\tilde{\boldsymbol{\omega}}_G = \dot{\tilde{\theta}}\tilde{\mathbf{n}} + \tilde{\theta}\dot{\tilde{\mathbf{n}}} = \dot{\mathbf{e}}_l \quad (53)$$

making the system again linear.

**Remark 3.** Considering the other two error selections, (53) also holds for the angle axis orientation error (23). Therefore, near the system's steady state the behavior of the two orientation errors is similar. However, as we discussed

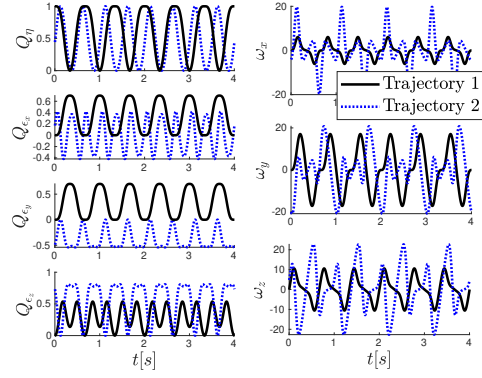


Fig. 2: Desired orientation and angular velocity trajectories.

in Section III this error selection's performance deteriorates for large error angles.

## VI. SIMULATION RESULTS

We will demonstrate the performance of controllers (19), (20) using all three types of orientation errors (21), (23), (25). For the numerical integration of the quaternion error we have used the exponential mapping (4) in order to preserve its unit norm:  $\tilde{\mathbf{Q}}(t + \Delta t) = \exp(\frac{1}{2}\tilde{\boldsymbol{\omega}}_G(t + \Delta t)\Delta t) * \tilde{\mathbf{Q}}(t)$

We present the results of two sets of simulations, using two different desired trajectories shown in Figure 2. Their main difference is that the second trajectory has a non zero initial angular velocity. For the first trajectory the system was initialized with  $\tilde{\theta}_0 = \frac{3\pi}{4}$ , and zero angular velocity error, while for the second with  $\tilde{\theta}_0 = 0$  and angular velocity errors with initial norm  $\|\tilde{\boldsymbol{\omega}}_G\| = \|\tilde{\boldsymbol{\omega}}_E\| = 20.9\text{rad/s}$ . Therefore the two systems would cover the two cases of linearity of Proposition 2. The control gains were set  $\kappa = 30.25$  and  $d = 11$ , for the linear system to be critically damped with settling time  $t_s \approx 1\text{s}$ . Figures 3 and 4 show responses of the error angle and the norm of the angular velocity error respectively for both systems, where (30) is depicted with dotted lines and (31) is depicted with solid lines and for all orientation error selections; the left subplot includes responses for the case of zero initial angular velocity while the right subplot those with a non-zero angular velocity. In Figure 3 we call "geometric", error angle responses from (31) and "Euclidean" those from (30). In Figure 4 we show the response of  $\tilde{\boldsymbol{\omega}}_E$  and  $\tilde{\boldsymbol{\omega}}_G$ . The subplot within this figure is a closer to the x-axis view. Responses converge exponentially to the origin as expected. For both trajectories,  $\mathbf{e}_l$  in system (31) responds with the expected speed as it is related to a linear tracking system according to Proposition 2. In all cases, it is evident that system (31) converges faster than (30), owing to its autonomous nature. Indeed, as shown in (43), the system's convergence rate is dependant on the bound of the norm of the desired angular velocity, rendering slower than (31). Notice that in both figures, due to the non-autonomous nature of system (30) the sinusoidal nature of the input is also present in the error system response.

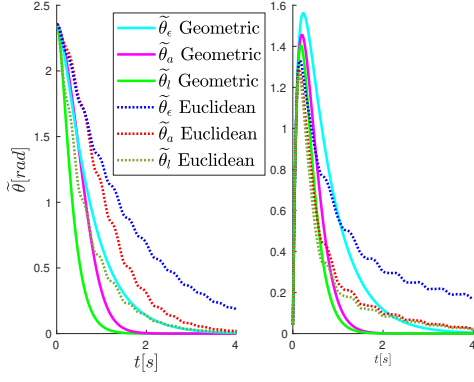


Fig. 3: Simulation Result: Error angle convergence.

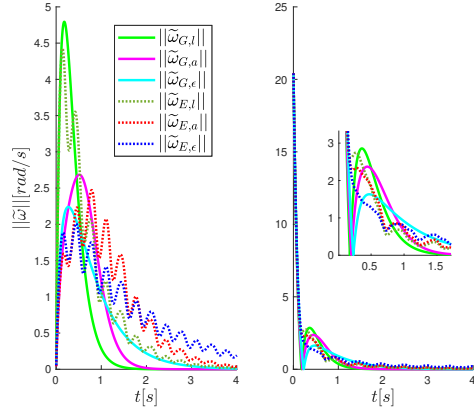


Fig. 4: Simulation Result: Angular velocity error convergence.

## VII. EXPERIMENTAL RESULTS

We implemented the methods studied in the previous sections using a UR5e robotic manipulator operating with control frequency  $500Hz$ . The planned trajectory was designed to resemble that of a polishing task of the interior of a spherical surface. The position of the wrist was kept constant; we can imagine this as the center of the sphere with the polishing tool having the length of the sphere's radius. We again conducted two experiments, one with zero initial angular velocity and one with non zero. In both cases the system was initialized with error angle  $\tilde{\theta} = \frac{\pi}{3}$ . The two trajectories are shown in Figure 5. We integrated system (33) producing a reference trajectory  $Q, \omega$ . Then the joint reference trajectory was provided to the robot utilizing a Closed Loop Inverse Kinematics scheme [34]. In our experiments we set as requirement a compliant behavior, thus system parameters were set to  $\kappa = d = 4$ , making the linear system critically damped with settling time  $t_s \approx 3$ .

Experimental results are shown in Figures 6, 7. All error selections are shown to converge exponentially to zero. In the first trajectory, the system using the logarithmic orientation error and the geometric angular velocity error, shows linear response. The second trajectory only demonstrates the exponential convergence of all systems, as it does not obey the conditions of Proposition 2, since for  $e_l(0) \neq 0$  and

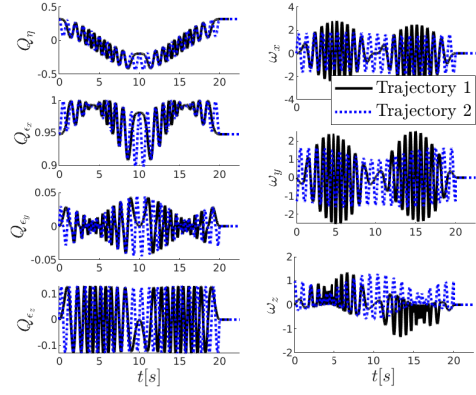


Fig. 5: Desired robot orientation and angular velocity trajectories.

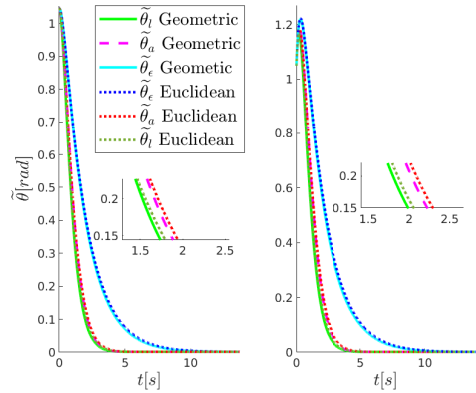


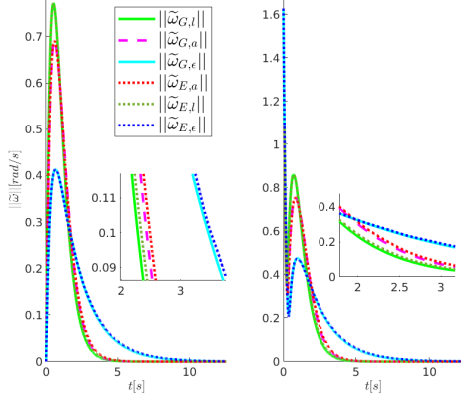
Fig. 6: Experimental Result: Error angle convergence.

$\tilde{\omega}_G(0) \neq 0$ . Again with  $e_l$  convergence is faster followed closely by the response with the use of  $e_a$ . This is because the initial angle error in the experiment is much less to the one used in the simulations ( $\frac{\pi}{3}$  vs  $\frac{3\pi}{4}$ ) and soon  $\tilde{\theta}$  becomes small enough for the approximation  $\sin \tilde{\theta} \approx \tilde{\theta}$  to hold; the latter implies  $e_l \approx e_a$ . Furthermore responses from the geometric and Euclidean system are similar, since the bound of the desired angular velocity norm is not as large as in the simulations to create notable differences.

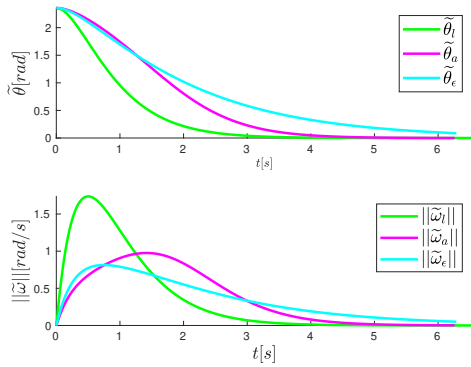
We performed an additional experiment with regulation control, with a large initial error angle  $\tilde{\theta}(0) = \frac{3\pi}{4}$ . Notice that in the case of regulation control, where  $\omega_d = 0 \forall t$ , the two angular velocity errors become identical. Therefore there is not difference between system (31) and (30). The results are shown in Figure 8. The logarithmic orientation error is shown to reach faster the desired orientation, with settling time again  $t_s \approx 3s$ , as expected since it exhibits linear behavior. Due to the larger initial error angle, the performance of the angle axis orientation error is shown to deteriorate from the previous experiment.

We tested the performance of the logarithmic orientation error external inputs with an admittance control setup, using (52). A polishing tool was mounted to the robot's end effector, being in contact with a mechanical part. The setup is shown in Figure 9. To execute the task, Trajectory 1 from





**Fig. 7:** Experimental Result: Angular velocity error convergence.



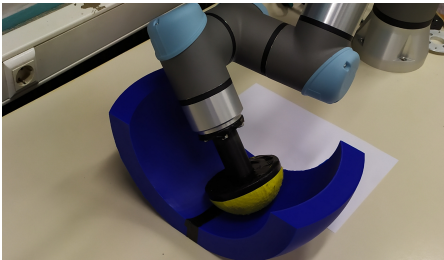
**Fig. 8:** Experimental Result: Error angle and angular velocity error convergence during regulation control.

Figure 5 was fed to (52), with the measurements from the robot's Force/Torque sensor used as an external input.

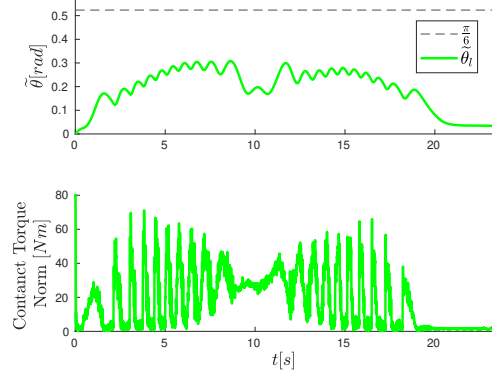
Experimental results are shown in Figures 10 and 11. In Figure 10 the contact torque norm is shown to induce small error angles, less than  $\frac{\pi}{6}$ . Thus the approximations discussed in Section V hold and the system should remain linear. This claim is further demonstrated in Figure 11, where the geometric angular velocity error is shown to be equal with the derivative of the logarithmic orientation error.

## VIII. CONCLUSIONS

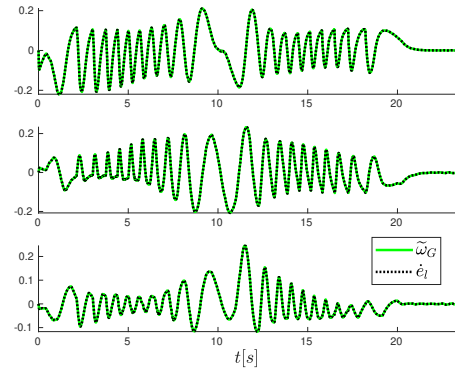
In this work we studied trajectory tracking in the orientation space using the unit quaternion representation. We presented existing variants regarding the orientation error



**Fig. 9:** Experimental setup for the polishing task.



**Fig. 10:** Experimental Result: Error angle and disturbance norm during polishing task.



**Fig. 11:** Experimental Result: Angular velocity error and derivative of logarithmic orientation error during polishing task.

and angular velocity error. We have associated each of these choices with the appropriate orientation error function and rigorously proved exponential error convergence in all cases. We have further showed that the formulation respecting the geometric characteristics of the quaternion manifold and its tangent space yields linear tracking dynamics under conditions frequently encountered in practice. Thus transient specification can be satisfied by appropriate gain selection without tuning. Theoretical results are validated by simulations and experiments.

## APPENDIX

### ORIENTATION ERROR JACOBIAN MATRICES

We will now provide the Jacobian matrices used in equations (28), (29). When using the vector part orientation error (21), we can write (14) and (12) as (28) and (29) with:

$$J_\epsilon = \frac{1}{2}(\tilde{\eta}I_3 - S(\tilde{\epsilon})) \quad (54)$$

For the angle axis orientation error we differentiate (23) substituting  $\tilde{\eta}$  and  $\tilde{\epsilon}$  from (13) (14) for  $j = G$  and (11) (12) for  $j = E$  to get:

$$J_a = \tilde{\eta}^2 I_3 - \tilde{\epsilon}\tilde{\epsilon}^T - \tilde{\eta}S(\tilde{\epsilon}) \quad (55)$$

For the logarithmic orientation error when differentiating (25) we need  $\tilde{\theta}$  and  $\tilde{n}$ , which can be found by differentiating

the axis and angle representations of (2):

$$\dot{\tilde{\theta}} = \tilde{\mathbf{n}}^T \tilde{\boldsymbol{\omega}}_G = \tilde{\mathbf{n}}^T \tilde{\boldsymbol{\omega}}_E \quad (56)$$

$$\dot{\tilde{\mathbf{n}}} = -\frac{1}{2} \begin{pmatrix} \cos \frac{\tilde{\theta}}{2} \\ \sin \frac{\tilde{\theta}}{2} \end{pmatrix} \mathbf{S}(\tilde{\mathbf{n}}) + \mathbf{I}_3 \bigg) \mathbf{S}(\tilde{\mathbf{n}}) \tilde{\boldsymbol{\omega}}_G \quad (57)$$

and

$$\dot{\tilde{\mathbf{n}}} = -\frac{1}{2} \begin{pmatrix} \cos \frac{\tilde{\theta}}{2} \\ \sin \frac{\tilde{\theta}}{2} \end{pmatrix} \mathbf{S}(\tilde{\mathbf{n}}) + \mathbf{I}_3 \bigg) \mathbf{S}(\tilde{\mathbf{n}}) \tilde{\boldsymbol{\omega}}_E - \mathbf{S}(\tilde{\mathbf{n}}) \boldsymbol{\omega}_d \quad (58)$$

Thus, we get:

$$\mathbf{J}_l = -\frac{\tilde{\theta} \cos(\frac{\tilde{\theta}}{2})}{\sin \frac{\tilde{\theta}}{2}} \mathbf{S}^2(\tilde{\mathbf{n}}) - \frac{\tilde{\theta}}{2} \mathbf{S}(\tilde{\mathbf{n}}) + \tilde{\mathbf{n}} \tilde{\mathbf{n}}^T \quad (59)$$

It is easy to prove using (25) that  $\mathbf{J}_l \mathbf{e}_l = \mathbf{e}_l$ . In fact  $\mathbf{J}_l$  has an eigenvalue of one and hence  $\mathbf{e}_l$  is the associated eigenvector. Notice that this property applies only for the logarithmic orientation error.

By evaluating the singular values of  $\mathbf{J}_i$ , it can be shown that the norm of  $\mathbf{J}_i$  is given by:

$$\|\mathbf{J}_i\| = \begin{cases} \frac{1}{2} & i = \epsilon \\ \max(\tilde{\eta}, |1 - 2\tilde{\eta}^2|) & i = a \\ h(\tilde{\theta}) & i = l \end{cases} \quad (60)$$

where

$$h(\tilde{\theta}) = \frac{\frac{\tilde{\theta}}{2}}{\sin\left(\frac{\tilde{\theta}}{2}\right)} \quad (61)$$

Notice that  $1 \leq h(\tilde{\theta}) \leq \frac{\pi}{2} \forall \tilde{\theta} \in [-\pi, \pi)$  and  $\|\mathbf{J}_a\| \leq 1$ .

## REFERENCES

- [1] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*, 1st ed. Springer Publishing Company, Incorporated, 2008.
- [2] P. C. Hughes, *Spacecraft Attitude Dynamics*, 1st ed. Dover Publications, 2004.
- [3] J. T. . Wen and K. Kreutz-Delgado, "The attitude control problem," *IEEE Transactions on Automatic Control*, vol. 36, no. 10, pp. 1148–1162, Oct 1991.
- [4] Fjellstad and Fossen, "Quaternion feedback regulation of underwater vehicles," in *1994 Proceedings of IEEE International Conference on Control and Applications*, Aug 1994, pp. 857–862 vol.2.
- [5] J. Rodriguez, H. Castañeda, and J. Gordillo, "Lagrange modeling and navigation based on quaternion for controlling a micro auv under perturbations," *Robotics and Autonomous Systems*, vol. 124, p. 103408, 2020.
- [6] R. M. Murray, S. S. Sastry, and L. Zexiang, *A Mathematical Introduction to Robotic Manipulation*, 1st ed. Boca Raton, FL, USA: CRC Press, Inc., 1994.
- [7] S. Chiaverini and B. Siciliano, "The unit quaternion: A useful tool for inverse kinematics of robot manipulators," *Systems Analysis Modelling Simulation*, vol. 35, no. 1, pp. 45–60, Jan. 1999.
- [8] J. C. K. Chou, "Quaternion kinematic and dynamic differential equations," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 1, pp. 53–64, Feb 1992.
- [9] S. P. Bhat and D. S. Bernstein, "A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon," *Systems & Control Letters*, vol. 39, pp. 63 – 70, 2000.
- [10] J. Luh, M. Walker, and R. Paul, "Resolved-acceleration control of mechanical manipulators," *IEEE Transactions on Automatic Control*, vol. 25, no. 3, pp. 468–474, June 1980.
- [11] R. Mahony, T. Hamel, and J. Pfimlin, "Nonlinear complementary filters on the special orthogonal group," *IEEE Transactions on Automatic Control*, vol. 53, no. 5, pp. 1203–1218, June 2008.
- [12] J. S. Yuan, "Closed-loop manipulator control using quaternion feedback," *IEEE Journal on Robotics and Automation (IEEE J Robot Autom)*, vol. 4, no. 4, pp. 434–440, Aug 1988.
- [13] F. Vigoriti, F. Ruggiero, V. Lippiello, and L. Villani, "Control of redundant robot arms with null-space compliance and singularity-free orientation representation," *Robotics and Autonomous Systems*, vol. 100, pp. 186 – 193, 2018.
- [14] A. Tayebi, "Unit quaternion-based output feedback for the attitude tracking problem," *IEEE Transactions on Automatic Control (IEEE T AUTOMAT CONTR)*, vol. 53, no. 6, pp. 1516–1520, July 2008.
- [15] —, "Unit quaternion-based output feedback for the attitude tracking problem," *IEEE Transactions on Automatic Control*, vol. 53, no. 6, pp. 1516–1520, 2008.
- [16] A. Ude, B. Nemec, T. Petrić, and J. Morimoto, "Orientation in cartesian space dynamic movement primitives," in *2014 IEEE International Conference on Robotics and Automation*, May 2014, pp. 2997–3004.
- [17] S. Kim, R. Haschke, and H. Ritter, "Gaussian mixture model for 3-dof orientations," *Robotics and Autonomous Systems*, vol. 87, pp. 28 – 37, 2017.
- [18] L. Koutras and Z. Doulgeri, "A correct formulation for the orientation dynamic movement primitives for robot control in the cartesian space," in *Proceedings of the Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 100. PMLR, 30 Oct–01 Nov 2020, pp. 293–302.
- [19] L. Koutras and Z. Doulgeri, "A novel dmp formulation for global and frame independent spatial scaling in the task space," in *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2020, pp. 727–732.
- [20] L. Koutras and Z. Doulgeri, "Exponential stability of an attitude trajectory tracking controller utilizing unit quaternions," in *2021 European Control Conference (ECC)*, 2021, pp. 1126–1131.
- [21] M. J. A. Zeestraten, I. Havoutis, J. Silvério, S. Calinon, and D. G. Caldwell, "An approach for imitation learning on riemannian manifolds," *IEEE Robotics and Automation Letters (RA-L)*, vol. 2, no. 3, pp. 1240–1247, July 2017.
- [22] K. Li, F. Pfaff, and U. D. Hanebeck, "Grid-based quaternion filter for so(3) estimation," in *2020 European Control Conference (ECC)*, 2020, pp. 1738–1744.
- [23] F. Caccavale, C. Natale, B. Siciliano, and L. Villani, "Resolved-acceleration control of robot manipulators: A critical review with experiments," *Robotica*, vol. 16, no. 5, p. 565–573, 1998.
- [24] R. Campa, R. Kelly, and E. Garcia, "On stability of the resolved acceleration control," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, vol. 4, May 2001, pp. 3523–3528 vol.4.
- [25] J. Biggs, "A quaternion-based attitude tracking controller for robotic systems," in *IMA Conference on Mathematics of Robotics*, GBR, September 2015.
- [26] Z. Yu and G. Li, "Quaternion-based robust sliding mode control for spacecraft attitude tracking," in *2019 Chinese Control And Decision Conference (CCDC)*, June 2019, pp. 2495–2500.
- [27] T. Lee, "Exponential stability of an attitude tracking control system on so(3) for large-angle rotational maneuvers," *Systems & Control Letters*, vol. 61, no. 1, pp. 231 – 237, 2012.
- [28] X. Liu, M. Zhang, J. Chen, and B. Yin, "Trajectory tracking with quaternion-based attitude representation for autonomous underwater vehicle based on terminal sliding mode control," *Applied Ocean Research*, vol. 104, p. 102342, 2020.
- [29] M. Saveriano, F. Franzel, and D. Lee, "Merging position and orientation motion primitives," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 7041–7047.
- [30] J. Solà, J. Deray, and D. Atchuthan, "A micro lie theory for state estimation in robotics," *CoRR*, vol. abs/1812.01537, 2018. [Online]. Available: <http://arxiv.org/abs/1812.01537>
- [31] F. Bullo and R. M. Murray, "Tracking for fully actuated mechanical systems: a geometric framework," *Automatica*, vol. 35, no. 1, pp. 17–34, 1999.
- [32] F. Bullo and A. D. Lewis, *Geometric Control of Mechanical Systems*, ser. Texts in Applied Mathematics. New York-Heidelberg-Berlin: Springer Verlag, 2004, vol. 49.
- [33] H. Marquez, *Nonlinear Control Systems: Analysis and Design*. Wiley-Interscience, 11 2002.
- [34] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer-Verlag, 2007, ch. 11.