2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Pose-Following with Dual Quaternions

Jon Arrizabalaga[1] and Markus Ryll[1,2]

Abstract— **This work focuses on pose-following, a variant of path-following in which the goal is to steer the system's position and attitude along a path with a moving frame attached to it. Full body motion control, while accounting for the additional freedom to self-regulate the progress along the path is an appealing trade-off. Towards this end, we extend the well-established dual quaternion based pose-tracking method into a pose-following control law. Specifically, we derive the equations of motion for the full pose error between the geometric reference and the rigid body in the form of a dual quaternion and dual twist, and subsequently, formulate an almost globally asymptotically stable control law. The global attractivity of the presented approach is validated in a spatial example, while its benefits over pose-tracking are showcased through a planar case-study.**

I. INTRODUCTION

Simultaneous attitude and position – *pose* – control of a rigid body in three-dimensional space is fundamental to many applications, such as for autonomous vehicles, spacecrafts or robotic manipulators.

The simplest approach to address the pose control problem is decoupling it into two separate subproblems [1], [2]. On the one hand, a *position controller* drives the translational motions, and on the other hand, an *attitude controller* regulates the rotational behavior. This separation relates to the de facto representation of the rigid body dynamics, in which the translational and angular motions are expressed separately (as planar and spatial examples, see eq. 7 in [3] and eq. 1 in [4]). However, such partitioning poses a challenge to effectively control the interdependence between the rotational and translational dynamics.

An alternative to this decoupling is representing the system dynamics globally on the configuration manifold of the special Euclidean group $\mathrm{SE}(3)$. Doing so allows for leveraging the group structure to first avoid singularities and second extend proportional derivative (PD) feedback controllers, for the pose-tracking problem [5]. Control methods inspired by these findings have shown very promising results within a plethora of robotic platforms, such as quadrotors [6], robotic manipulators [7], walking robots [8] and spacecrafts [9].

To represent a rigid body in $\mathrm{SE}(3)$, it is customary to combine a three-dimensional vector of the Cartesian coordinates with either a rotation matrix – resulting in a homogeneous transformation matrix (HTM) – or a unit quaternion. A less common choice are *unit dual quaternions*.

Unit quaternion parameterization can be summarized into five advantages: First, when compared to HTMs, unit dual quaternions offer a more compact representation of $\mathrm{SE}(3)$, as it requires only eight parameters (against twelve) to describe the motions of a rigid body – rotations and translations – in a singularity-free and global manner [10]. Second, dual quaternion multiplications are computationally more efficient than HTM multiplications [7]. Third, the utilization of unit dual quaternions is comparatively simpler than the use of Cartesian coordinates and unit quaternions. This can be attributed to the fact that a series of rigid movements can be expressed as a sequence of dual quaternion multiplications, whereas in the other case, the computation of rotation and position is performed independently (for further details, refer to [7], [11]). Fourth, unit dual quaternions yield two closed-loop equilibrium points associated to a quaternion's double coverage of $\mathrm{SO}(3)$, both of which represent the identity rotation matrix, while rotation matrices generate a minimum of four closed-loop equilibrium points, with only one of them relating to the identity [9]. Fifth, in contrast to alternative techniques within $\mathrm{SE}(3)$, such as [6], unit dual quaternions exclusively rely on a single error function, rather than the need for two separate functions addressing position and attitude errors.

In light of these features, unit dual quaternions have been applied across a wide range of disciplines, including but not limited to inertial navigation [12], state estimation [13], inverse kinematics [14], computer graphics [15] and computer vision [16].

In the context of pose control, akin to [5], the authors in [17] and [18] broadened PD-alike feedback controllers to encompass the Lie group of unit dual quaternions using its logarithmic mapping. This resulted in a globally exponentially stable kinematic control law for pose regulation or tracking. These outcomes were subsequently expanded in [19] to also account for rigid body dynamics. Since these findings, the unit dual quaternion-based pose-tracking problem has received considerable attention in the literature. To name a few, the need for linear and angular velocity feedback was dropped in [11], a backstepping control technique to account for robustness was proposed in [20], adaptive control allowed for simultaneous pose-tracking and parameter identification in [9], formation flying was addressed in [21], and optimal control variants in the form of linear quadratic regulator (LQR) and Model Predictive Control (MPC) were formulated in [22] and [23], respectively.

Despite the achievements, all these methods exclusively focus on *pose-tracking*, i.e., they track a time-varying position and attitude reference. However, not all problems fit in such a description. For a more intuitive understanding of this concept we use an illustrative example from [24]: When aiming for precise steering of a robot tool along a geometric reference, the primary concern is to minimize the deviation between the reference and the tool, while the velocity to move along the reference is of secondary

[1] Autonomous Aerial Systems, School of Engineering and Design, Technical University of Munich, Germany. E-mail: jon.arrizabalaga@tum.de and markus.ryll@tum.de

[2] Munich Institute of Robotics and Machine Intelligence (MIRMI), Technical University of Munich

979-8-3503-0123-6/23/$31.00 ©2023 IEEE

interest and can be modified to enhance accuracy. In other words, the problem is not centered on tracking a predefined time-varying reference, but rather on leveraging the velocity to traverse the reference as an additional degree of freedom. Such control problems are denoted as *path-following*. Its appealing properties for a wide range of applications, accompanied by the fact that it is agnostic to the fundamental limitations of reference-tracking [25], account for the significant attention path-following has received in literature. A detailed description of existing approaches can be found in [24], [26]. Among those, most of the existing path-following methods omit the rotational dynamics and focus on path convergence of the translational dynamics.

To close this gap, we assume that the geometric reference consists of a desired path with a moving coordinate frame associated to it, and we define *pose-following* as a generalization of path-following in which the goal is to steer the system's position and attitude[1] along the reference. This begs the question of how to formulate such a pose-following method.

To answer this question, in this paper we derive a unit dual quaternion-based pose-following control approach for rigid body dynamics. To do so, we take advantage of the previously mentioned benefits of unit dual quaternions, namely singularity-free, compactness, computational efficiency and the logarithmic mapping associated with the Lie group, allowing us to extend the PD-alike feedback control from pose-tracking to pose-following. As a result, the freedom and versatility of path-following is augmented to full body motions, i.e., translations, as well as rotations. To the best of our knowledge, this is the first work that explicitly attempts to follow upon both the longitudinal and angular coordinates.

More specifically, the presented scheme consists of the following contributions:

1) We derive the equations of motion for the full pose error between the geometric reference and the rigid body in the form of a dual quaternion and dual twist.
2) We extend the original control law to account for nonlinearities that arise from introducing auxiliary states associated with pose-following. Besides that, we design the additional degree of freedom either to ensure convergence to a desired velocity profile or as feedback. Therefore, the progress along the reference is versatile in the sense that it either accepts any velocity profile or can be utilized to incite a desired behavior around the geometric reference.
3) We prove that the presented control law is almost globally asymptotically stable[2]. When doing so, we take special care of the two equilibria problem, by introducing a switch that guarantees convergence to the closest equilibrium point.

The remainder of this paper is structured as follows: Section II introduces the pose-following problem. Section III presents the solution proposed in this paper, by revisiting the unit dual quaternion-based algebra, transforming the error dynamics into dual quaternion and dual twist form, deriving the control law and conducting a stability analysis. Experimental results are shown in Section IV before Section V presents the conclusions.

Notation: We will use $\dot{(\cdot)} = \frac{\mathrm{d}(\cdot)}{\mathrm{d}t}$ for time derivatives and $\overset{\circ}{(\cdot)} = \frac{\mathrm{d}(\cdot)}{\mathrm{d}\theta}$ for differentiating over pose-parameter θ. We denote three-dimensional vectors in bold $\boldsymbol{v}$, dual numbers as $a + \epsilon \mathrm{b}$ and dual quaternions as $\hat{q}$. We define $\hat{I}$ as $[1,0,0,0] + \epsilon[0,0,0,0]$.

II. THE POSE-FOLLOWING PROBLEM

Classical path-following approaches steer a system's position, while leaving the attitude unattended. In this work, we tighten the original path-following problem by focusing on full rigid body motions, i.e, position and attitude following.

A. Rigid body dynamics

The three-dimensional rigid body dynamics in the body frame are given by

$$\ddot{\boldsymbol{p}}^b(t) = \boldsymbol{f}^b(t) m^{-1}\,, \tag{1a}$$

$$\dot{\boldsymbol{\omega}}^b(t) = \mathrm{J}^{-1}\left(\boldsymbol{\tau}^b(t) - \boldsymbol{\omega}^b(t) \times \mathrm{J}\,\boldsymbol{\omega}^b(t)\right)\,, \tag{1b}$$

where $\{\boldsymbol{p}^b, \boldsymbol{\omega}^b, \boldsymbol{f}^b, \boldsymbol{\tau}^b\} \in \mathbb{R}^3$ refer to the rigid body's position, angular velocity, control forces and control torques, while $m \in \mathbb{R}$ and $J \in \mathbb{R}^{3\times 3}$ are the mass and inertia matrix. From now onward, since frame superscripts remain constant, they will be dropped. By taking position $\boldsymbol{p}$, longitudinal velocity $\dot{\boldsymbol{p}}$, attitude $q \in \mathrm{SO}(3)$ and angular velocity $\boldsymbol{\omega}$ as states $\boldsymbol{x}(t) = [\boldsymbol{p}(t), \dot{\boldsymbol{p}}(t), q(t), \boldsymbol{\omega}(t)]$ with forces $\boldsymbol{f}$ and torques $\boldsymbol{\tau}$ as inputs $\boldsymbol{u}(t) = [\boldsymbol{f}(t), \boldsymbol{\tau}(t)]$, and introducing the respective kinematic relationships, the dynamics in (1) can be written in the standard form:

$$\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), \boldsymbol{u}(t))\,. \tag{2}$$

B. Geometric reference representation

Let Γ refer to a geometric reference and be defined as a path with a moving frame attached to it. Its respective desired position and attitude are given by two functions, $\boldsymbol{p}_d : \mathbb{R} \mapsto \mathbb{R}^3$ and $q_d : \mathbb{R} \mapsto \mathrm{SO}(3)$, that depend on pose-parameter θ and are at least $\mathcal{C}^2$:

$$\Gamma = \{\theta \in [\theta_0, \theta_f] \subseteq \mathbb{R} \mapsto \boldsymbol{p}_d(\theta) \in \mathbb{R}^3, q_d(\theta) \in \mathrm{SO}(3)\} \tag{3}$$

It should be noted that the C^2 requirement for q_d enables the calculation of the desired angular velocity $\boldsymbol{\omega}_d(\theta) : \mathbb{R} \mapsto \mathbb{R}^3$ from its kinematic equations.

C. Problem statement

To incorporate the additional freedom inherited from path-following, we augment the rigid body dynamics in (2) by adding the pose-parameter $\theta(t)$ and its first time derivative

[1] In contrast to [26], [27], where the rotational convergence is reduced to the heading $\psi \in \mathbb{R}$ and the remaining two euler angles are left unattended, we seek *true attitude convergence*, i.e. $q \in \mathrm{SO}(3)$.

[2] "almost" globally asymptotically stability is the optimal performance achievable by a continuous controller for rotational motion, owing to the fact that the group of rotation matrices SO(3) constitutes a compact manifold [28].

$\dot{\theta}(t)$ as virtual states and assign the second time derivative $\ddot{\theta}(t)$ as a virtual input. The resulting system is denoted as

$$\dot{\boldsymbol{x}}_\Gamma(t) = f_\Gamma(\boldsymbol{x}_\Gamma(t), \boldsymbol{u}_\Gamma(t)) \,. \tag{4}$$

where $\boldsymbol{x}_\Gamma(t) = \left[\boldsymbol{p}(t), \dot{\boldsymbol{p}}(t), q(t), \boldsymbol{\omega}(t), \theta(t), \dot{\theta}(t)\right]$ and $\boldsymbol{u}_\Gamma(t) = \left[\boldsymbol{f}(t), \boldsymbol{\tau}(t), \ddot{\theta}(t)\right]$. The augmented system f_Γ contains two additional equations of motion that relate to the integration chain of the pose-parameter $\theta(t)$, implying that the virtual input $\ddot{\theta}(t)$ is associated to its acceleration. Consequently, the time evolution $\theta(t)$, and thereby the pose reference $\{\boldsymbol{p}_d(\theta(t)), q_d(\theta(t))\}$, are controlled via the virtual input $\ddot{\theta}(t)$. This leads to the definition of the *pose-following error* as

$$\boldsymbol{e}_\Gamma(t) = \triangle\left[\{\boldsymbol{p}(t), q(t)\}, \{\boldsymbol{p}_d(\theta(t)), q_d(\theta(t))\}\right] \,, \tag{5}$$

where $\triangle : \{\mathbb{R}^3, \mathrm{SO}(3)\} \times \{\mathbb{R}^3, \mathrm{SO}(3)\} \mapsto \mathbb{R}$ is a function that outputs the deviation between the rigid body's pose and reference pose, and will only be 0 if both are equal, i.e., $\triangle[a, b] = 0 \iff a = b$. Due to the structure of SE(3), this function is dependent on the control design approach, and thus, will be defined in the upcoming Section III. For the remainder of this work, we address the following problem:

Problem 1 (Pose-Following): *Given the geometric reference* Γ *in* (3) *and the augmented rigid body dynamics in* (4)*, formulate a controller* $\boldsymbol{u}_\Gamma(t) = \left[\boldsymbol{f}(t), \boldsymbol{\tau}(t), \ddot{\theta}(t)\right]$ *that fulfills:*

P1.1 ***Pose convergence:*** *The pose-following error vanishes asymptotically* $\lim_{t\to\infty} \boldsymbol{e}_\Gamma(t) = 0$.

P1.2 ***Convergence on pose-parameter:*** *The system converges to the end of the geometric reference* $\lim_{t\to\infty} \theta_f - \theta(t) = 0$.

For specific applications, it might be of interest to traverse the reference according to a desired velocity profile $\theta_{vd}(\theta(t))$. For example, inspection and manufacturing processes might require a lower traverse velocity at critical sections of the geometric reference. For this reason, in contrast to [24], instead of directly depending on time, our velocity profile θ_{vd} depends on the pose-parameter $\theta(t)$. Notice that this problem differs from reference tracking, for further details refer to Sec II. in [24]. Remaining consistent with the existing literature, we denote this problem as *pose-following with velocity assignment*.

Problem 2 (Pose-Following with velocity assignment): *Given the geometric reference* Γ *in* (3) *and the augmented rigid body dynamics in* (4)*, formulate a controller* $\boldsymbol{u}_\Gamma(t) = \left[\boldsymbol{f}(t), \boldsymbol{\tau}(t), \ddot{\theta}(t)\right]$ *that fulfills:*

P2.1 ***Pose convergence:*** *P1.1 from Problem 1.*

P2.2 ***Velocity convergence:*** *The velocity of the pose-parameter converges to a desired velocity profile* $\lim_{t\to\infty} \dot{\theta}(t) - \theta_{vd}(\theta(t)) = 0$.

III. Solution Approach

A. Mathematical preliminaries

For the sake of making this paper self-contained we briefly recall the dual quaternion algebra. In doing so, we follow the notation and content of [19], which pioneered the use of unit dual quaternion-based pose-tracking for rigid body dynamics. To begin, we define the foundational concepts of the quaternion and the dual number, which serve as the building blocks of the dual quaternion. For more comprehensive information on these concepts, we refer the reader to [10], [18], [29].

1) Quaternions: Quaternions extend the notion of a complex number to the four-dimensional space $\mathbb{R}^4$ and can be expressed as $q = a + b\,i + c\,j + d\,k = [s, \boldsymbol{v}]$, where s is the *scalar part*, $\boldsymbol{v} \in \mathbb{R}^3$ is the *vector part* and $\{i, j, k\}$ is the standard $\mathbb{R}^4$ basis, i.e., $i^2 = j^2 = k^2 = -1$ and $ij = k,\ jk = i,\ ki = j$. When a three-dimensional vector is expressed as a quaternion with a zero scalar component, the term *vector quaternion* is used. Furthermore, quaternions fulfilling $a^2 + b^2 + c^2 + d^2 = 1$ are named *unit quaternions* and allow to describe rotations, i.e., a rotation around a unit axis $\boldsymbol{n}$ by an angle of $|\phi| < 2\pi$ can be expressed as a unit quaternion in the form of $q = [\cos(\phi/2), \sin(\phi/2)\boldsymbol{n}]$. Additionally, unit quaternions constitute a Lie group $\mathcal{Q}_u$ over multiplication, and its logarithmic map is $\ln \hat{q} = \phi/2$ with $\phi = [0,\ |\phi|\boldsymbol{n}]$. Limiting to $\phi \in [0,\ 2\pi)$ and defining υ as the Lie algebra of $\mathcal{Q}_u$, i.e., all logarithmic mappings of unit quaternions, the adjoint transformation is $\mathrm{Ad}_q V = q \circ V \circ q^-1 = q \circ V \circ q^*$, where '$\circ$' is the quaternion multiplication and $V \in \upsilon$.

2) Dual numbers and vectors: Dual numbers are defined as $\hat{a} = a + \epsilon b$ with $\epsilon^2 = 0$, $\epsilon \neq 0$ – ϵ is nilpotent –, and $\{a, b\} \in \mathbb{R}$. In this case a and b are denoted as the *real part* and *dual part*, respectively. Dual vectors are a generalization of dual numbers, where both the real and dual part are three-dimensional vectors such that $\hat{\boldsymbol{a}} = \boldsymbol{a} + \epsilon \boldsymbol{b}$, where $\{\boldsymbol{a}, \boldsymbol{b}\} \in \mathbb{R}^3$.

3) Dual quaternions: A dual quaternion features dual components instead of its regular ones, i.e., $\hat{q} = [\hat{s}, \hat{\boldsymbol{v}}]$, where $\hat{s}$ is a dual number and $\hat{\boldsymbol{v}}$ is a dual vector. Following the introduced notation, a dual quaternion with a vanishing scalar part is named as a *dual vector quaternion*. We define the dot product of two dual vector quaternions $\hat{v} = \left[\hat{0}, \hat{\boldsymbol{v}}\right]$ with $\hat{\boldsymbol{v}} = \boldsymbol{v}_r + \epsilon \boldsymbol{v}_d$ and $\hat{k} = [\hat{0}, \hat{\boldsymbol{k}}]$ with $\hat{\boldsymbol{k}} = \boldsymbol{k}_r + \epsilon \boldsymbol{k}_d = [k_{r1}, k_{r2}, k_{r3}] + \epsilon\,[k_{d1}, k_{d2}, k_{d3}]$ as

$$\hat{k} \odot \hat{v} = \left[\hat{0}, K_r \boldsymbol{v}_r\right] + \epsilon \left[\hat{0}, K_d \boldsymbol{v}_d\right]$$

with $K_r = \mathrm{diag}(k_{r1}, k_{r2}, k_{r3})$ and $K_d = \mathrm{diag}(k_{d1}, k_{d2}, k_{d3})$. Alternatively, a dual quaternion can also be expressed as $\hat{q} = q_d + \epsilon q_r$, where q_d and q_r are quaternions. To operate with dual quaternions, we introduce the following operations:

$$\begin{aligned}
&\hat{q}_1 + \hat{q}_2 = [\hat{s}_1 + \hat{s}_2, \hat{\boldsymbol{v}}_1 + \hat{\boldsymbol{v}}_2] = (q_{r1} + q_{r2}) + \epsilon\,(q_{d1} + q_{d2}) \,, \\
&\lambda\hat{q} = [\lambda\hat{s}, \lambda\hat{\boldsymbol{v}}] = \lambda q_r + \epsilon\lambda q_d \,, \\
&\hat{q}_1 \circ \hat{q}_2 = \left[\hat{s}_1\hat{s}_2 - \hat{\boldsymbol{v}}_1^T \odot \hat{\boldsymbol{v}}_2, \hat{s}_1\hat{\boldsymbol{v}}_2 + \hat{s}_2\hat{\boldsymbol{v}}_1 + \hat{\boldsymbol{v}}_1 \times \hat{\boldsymbol{v}}_2\right] = \\
&\qquad\quad = q_{r1} \circ q_{r_2} + \epsilon\,(q_{r1} \circ q_{d2} + q_{d1} \circ q_{r2}) \,,
\end{aligned}$$

where $\hat{q}_1$ and $\hat{q}_2$ are dual quaternions, $\lambda \in \mathbb{R}$ and the operator '$\circ$' is the (dual) quaternion multiplication, which is associative and distributive but not commutative. Other relevant properties of the dual quaternion are the conjugate $\hat{q}^* = [\hat{s}, -\hat{\boldsymbol{v}}]$ and the multiplicative inverse $\hat{q}^{-1} = (1/\hat{q} \circ \hat{q}^*) \circ \hat{q}^*$. It follows that dual quaternions that meet the condition $\hat{q} \circ \hat{q}^* = \hat{I}$ also satisfy $\hat{q}^{-1} = \hat{q}^*$. In such instances, the dual quaternion in question is referred to as a *unit dual quaternion*.

Just as unit quaternions permit the representation of rotations, unit dual quaternions provide a means of describing three-dimensional transformations that encompass both translation and rotation. Specifically, a transformation consisting of a translation vector $\boldsymbol{p}$ and a rotation quaternion q corresponds to a screw motion, which entails a translation along axis $\boldsymbol{n}$ by a distance d, and a rotation of angle $|\phi|$. Such a transformation is expressed as a dual quaternion in the following form:

$$\hat{q} = \left[\cos\left(\hat{\phi}/2\right), \sin\left(\hat{\phi}/2\right)\hat{\boldsymbol{n}}\right] = q + \epsilon/2\, p \circ q\,, \tag{6}$$

where $\hat{\boldsymbol{n}}$ is the dual screw axis and $\hat{\phi} = |\phi| + \epsilon d$ is the dual screw angle. Similar to unit quaternions, unit dual quaternions form a Lie group $\mathcal{DQ}_u$ with respect to the dual quaternion multiplication and its logarithmic map is also a dual quaternion given by

$$\ln \hat{q} = 1/2\left(\phi + \epsilon p\right)\,, \tag{7}$$

where $\phi = [0,\, |\phi|\, \boldsymbol{n}]$. In a similar way to unit quaternions, naming $\hat{v}$ as the Lie algebra for $\mathcal{DQ}_u$, the adjoint transformation for the dual quaternions is $\mathrm{Ad}_{\hat{q}}\hat{V} = \hat{q} \circ \hat{V} \circ \hat{q}^{-}1 = \hat{q} \circ \hat{V} \circ \hat{q}^*$, where $\hat{V} \in \hat{v}$. For additional information regarding the Lie group of unit dual quaternions, please refer to [18].

B. Unit dual quaternion dynamics

In this subsection we aim to transform the rigid body dynamics in (2) to a unit dual quaternions representation. For this purpose, in a similar way to [19], we start by derivating (6) in time, which results in the following kinematic equations:

$$\dot{\hat{q}}(t) = \frac{1}{2}\hat{\omega}(t) \circ \hat{q}(t)\,, \tag{8a}$$

$$\hat{\omega}(t) = [0, \boldsymbol{\omega}(t)] + \epsilon\,[0, \dot{\boldsymbol{p}}(t) + \boldsymbol{p}(t) \times \boldsymbol{\omega}(t)] \tag{8b}$$

where $\hat{\omega}(t)$ is the dual twist. Taking its time derivative, we get

$$\dot{\hat{\omega}}(t) = \dot{\boldsymbol{\omega}}(t) + \epsilon\left(\ddot{\boldsymbol{p}}(t) + \dot{\boldsymbol{p}}(t) \times \boldsymbol{\omega}(t) + \boldsymbol{p}(t) \times \dot{\boldsymbol{\omega}}(t)\right)\,, \tag{8c}$$

and combining it with the rigid body dynamics in (1), leads to

$$\begin{aligned}\dot{\hat{\omega}}(t) &= \left(\boldsymbol{a} + \mathrm{J}^{-1}\boldsymbol{\tau}\right) + \epsilon\left(\boldsymbol{f}/m + \dot{\boldsymbol{p}} \times \boldsymbol{\omega} + \boldsymbol{p} \times \left(\boldsymbol{a} + \mathrm{J}^{-1}\boldsymbol{\tau}\right)\right)\\ &= \underbrace{\boldsymbol{a} + \epsilon\left(\boldsymbol{p} \times \boldsymbol{a} + \dot{\boldsymbol{p}} \times \boldsymbol{\omega}\right)}_{\hat{F}} + \underbrace{\mathrm{J}^{-1}\boldsymbol{\tau} + \epsilon\left(\boldsymbol{f}/m + \boldsymbol{p} \times \mathrm{J}^{-1}\boldsymbol{\tau}\right)}_{\hat{U}}\\ &= \hat{F}(t) + \hat{U}(t)\end{aligned} \tag{8d}$$

with $\boldsymbol{a} = \mathrm{J}^{-1}\boldsymbol{\omega} \times \mathrm{J}\,\boldsymbol{\omega}$. For readability, in the first two lines of (8d) dependencies on time $(\cdot)(t)$ have been omitted. Notice that $\hat{F}(t)$ is fully defined by the rigid body's state $\boldsymbol{x}(t)$, while the force and torque control inputs $\boldsymbol{u}(t)$ only appear in $\hat{U}(t)$.

Model 1 (**Unit dual quaternion-based dynamics**): *Letting eq.* (6) *describe a screw motion for translation $\boldsymbol{p}(t)$ and rotation $q(t)$, and defining the linear and angular velocity as in eq.* (2)*, then the unit dual quaternion-based rigid body dynamics are*

$$\dot{\hat{q}}(t) = \frac{1}{2}\hat{\omega}(t) \circ \hat{q}(t)\,, \tag{9a}$$

$$\hat{\omega}(t) = \boldsymbol{\omega}(t) + \epsilon\left(\dot{\boldsymbol{p}}(t) + \boldsymbol{p}(t) \times \boldsymbol{\omega}(t)\right)\,, \tag{9b}$$

$$\dot{\hat{\omega}}(t) = \hat{F}(t) + \hat{U}(t) \tag{9c}$$

where $\hat{F}(t)$ and $\hat{U}(t)$ are given in (8d).

C. Unit dual quaternion error dynamics

Drawing upon the dynamics obtained in the previous subsection, we proceed to derive the unit dual quaternion-based error dynamics for the pose-following problem. By applying the dual quaternion and twist definitions in eqs. (6) and (8b), it is possible to transform the geometric reference Γ in (3) into a desired dual quaternion and a desired dual twist:

$$\hat{q}_d(\theta) = q_d(\theta) + \epsilon/2\, p_d(\theta) \circ q_d(\theta)\,, \tag{10a}$$

$$\hat{\omega}_d(\theta) = [0, \boldsymbol{\omega}_d(\theta)] + \epsilon\,[0, \mathring{\boldsymbol{p}}_d(\theta) + \boldsymbol{p}_d(\theta) \times \boldsymbol{\omega}_d(\theta)]\,, \tag{10b}$$

where $\theta \in [\theta_0, \theta_f]$. Combining (10) with the kinematics in (8), the equations of motion for the desired pose are obtained:

$$\mathring{\hat{q}}_d(\theta) = \frac{1}{2}\hat{\omega}_d(\theta) \circ \hat{q}_d(\theta)\,, \tag{11a}$$

$$\begin{aligned}\mathring{\hat{\omega}}_d(\theta) &= [0, \mathring{\boldsymbol{\omega}}_d(\theta)] +\\ &\epsilon\,[0, \mathring{\mathring{\boldsymbol{p}}}_d(\theta) + \mathring{\boldsymbol{p}}_d(\theta) \times \boldsymbol{\omega}_d(\theta) + \boldsymbol{p}_d(\theta) \times \mathring{\boldsymbol{\omega}}_d(\theta)]\,,\end{aligned} \tag{11b}$$

From (11) it is apparent that in contrast to the pose-tracking case [19], the desired pose in (10) does not evolve according to *time t*, but with respect to the *pose-parameter θ*. The error between the geometric reference and the rigid body's pose can be expressed in the form of a unit dual quaternion:

$$\hat{q}_e(t) = \hat{q}(t) \circ \hat{q}_d^*(\theta(t))\,, \tag{12}$$

Derivating the dual quaternion error (12) in time[3] leads to

$$\dot{\hat{q}}_e(t) = \dot{\hat{q}}(t) \circ \hat{q}_d^*(\theta(t)) + \dot{\theta}(t)\hat{q}(t) \circ \mathring{\hat{q}}_d^*(t)\,.$$

Combining it with (8a), (11a), (12) and the property $(\hat{q}_1 \circ \hat{q}_2)^* = \hat{q}_2^* \circ \hat{q}_1^*$ results in

$$\dot{\hat{q}}_e(t) = \frac{1}{2}\left(\hat{\omega}(t) \circ \hat{q}_e(t) + \dot{\theta}(t)\hat{q}_e(t) \circ \hat{\omega}_d^*(\theta(t))\right)\,.$$

Noticing that $\hat{q}_e \circ \hat{\omega}_d^* = (\hat{q}_e \circ \hat{\omega}_d^* \circ \hat{q}_e^*) \circ \hat{q}_e$, the equation above can be rearranged to

$$\dot{\hat{q}}_e(t) = \frac{1}{2}\left(\hat{\omega}(t) + \dot{\theta}(t)\hat{q}_e(t) \circ \hat{\omega}_d^*(\theta(t)) \circ \hat{q}_e^*(t)\right) \circ \hat{q}_e(t)\,,$$

which takes the same form as (8a):

$$\dot{\hat{q}}_e(t) = \frac{1}{2}\hat{\omega}_e(t) \circ \hat{q}_e(t)\,, \tag{13a}$$

$$\hat{\omega}_e(t) = \hat{\omega}(t) + \dot{\theta}(t)\mathrm{Ad}_{\hat{q}_e(t)}\hat{\omega}_d^*(\theta(t))\,. \tag{13b}$$

When compared to the pose-tracking case, the first time derivative of the pose-parameter $\dot{\theta}(t)$ appears to be multiplying the second term of the dual twist error. Similar

[3] Time derivations over pose-parameter θ dependent variables, such as the $\hat{q}_d(\theta(t))$ requires using the chain rule, i.e., $\frac{\mathrm{d}(\cdot)}{\mathrm{d}t} = \frac{\mathrm{d}(\cdot)}{\mathrm{d}\theta}\frac{\mathrm{d}\theta}{\mathrm{d}t} = \mathring{(\cdot)}\dot{\theta}(t)$

derivations result in $p_e(t) = p(t) + \mathrm{Ad}_{q_e(t)} p_d^*(\theta(t))$ and $w_e(t) = w(t) + \dot{\theta}(t)\mathrm{Ad}_{q_e(t)} \omega_d^*(\theta(t))$. These expressions allow to ensure that the right-hand side of (13b) is equivalent to

$$\hat{\omega}_e(t) = [0, \boldsymbol{\omega}_e(t)] + \epsilon\,[0, \dot{\boldsymbol{p}}_e(t) + \boldsymbol{p}_e(t) \times \boldsymbol{\omega}_e(t)] \ . \tag{14}$$

For brevity, the present manuscript excludes these derivations; however, they are provided in the Appendix of the supplementary material[4]. Other than that, to fully define the error dynamics, we still need to compute the time derivative of the dual twist error. Towards this end, we take the time derivative of (13b) and we obtain

$$\begin{aligned}\dot{\hat{\omega}}_e(t) = \ & \dot{\hat{\omega}}(t) + \ddot{\theta}(t)\,\mathrm{Ad}_{\hat{q}_e(t)} \hat{\omega}_d^*(\theta(t)) + \\ & \dot{\theta}(t) \Big[\dot{\hat{q}}_e(t) \circ \hat{\omega}_d^*(\theta(t)) \circ \hat{q}_e(t) + \\ & \qquad \hat{q}_e(t) \circ \dot{\theta}(t) \mathring{\hat{\omega}}_d^*(\theta(t)) \circ \hat{q}_e(t) + \\ & \qquad \hat{q}_e(t) \circ \hat{\omega}_d^*(\theta(t)) \circ \dot{\hat{q}}_e(t) \Big] \ .\end{aligned}$$

Model 2 (**Unit dual quaternion-based error dynamics**): *For a given dual quaternion state $\hat{q}(t)$ and a desired configuration $\hat{q}_d(\theta(t))$ – associated to pose-parameter $\theta(t)$ –, the dynamics of the dual quaternion error in* (12) *are*

$$\dot{\hat{q}}_e(t) = \frac{1}{2} \hat{\omega}_e(t) \circ \hat{q}_e(t) \,, \tag{15a}$$

$$\hat{\omega}_e(t) = [0, \boldsymbol{\omega}_e(t)] + \epsilon\,[0, \dot{\boldsymbol{p}}_e(t) + \boldsymbol{p}_e(t) \times \boldsymbol{\omega}_e(t)] \ , \tag{15b}$$

$$\dot{\hat{\omega}}_e(t) = \hat{F}(t) + \hat{U}(t) + \ddot{\theta}(t)\,Ad_{\hat{q}_e(t)} \hat{\omega}_d^*(\theta(t)) + \tag{15c}$$

$$\begin{aligned} & \dot{\theta}(t) \Big[\dot{\hat{q}}_e(t) \circ \hat{\omega}_d^*(\theta(t)) \circ \hat{q}_e(t) + \\ & \qquad \hat{q}_e(t) \circ \dot{\theta}(t) \mathring{\hat{\omega}}_d^*(\theta(t)) \circ \hat{q}_e(t) + \\ & \qquad \hat{q}_e(t) \circ \hat{\omega}_d^*(\theta(t)) \circ \dot{\hat{q}}_e(t) \Big] \ , \end{aligned} \tag{15d}$$

where $\hat{q}_e(t) = \hat{q} \circ \hat{q}_d^*(\theta(t))$, $p_e(t) = p(t) + Ad_{q_e(t)} p_d^*(\theta(t))$, $w_e(t) = w(t) + \dot{\theta}(t) Ad_{q_e(t)} \omega_d^*(\theta(t))$, and $\hat{F}(t)$ and $\hat{U}(t)$ are given in (8d).

The first two equations, i.e., the time derivative of the dual quaternion error and the dual twist error, show the same structure as in pose-tracking. However, differences arise in the time derivative of the dual twist error, as it involves additional terms multiplied by the first and second time derivatives of the pose-parameter $\theta(t)$.

D. Control law

Considering the error dynamics in (15), the *pose-convergence* definition in P1.1 can be reformulated as $\lim_{t\to\infty} \hat{q}_e(t) = \pm\hat{I}$ and $\lim_{t\to\infty} \hat{\omega}_e(t) = \hat{0}$. To accomplish this, in this subsection we derive a control law for the term $\hat{U}$ in equation (15d). Its design is significantly influenced by the forthcoming stability analysis. Once the control law is determined, we will be able calculate the command forces $\boldsymbol{f}$ and torques $\boldsymbol{\tau}$ from eq. (8d). In a similar way to [5] and [19], we decouple it into a feedforward (FF) and a feedback term (FB). The former eliminates nonlinearities in (15d) and the latter ensures stability.

$$\hat{U} = \hat{U}_{\text{FF}} + \hat{U}_{\text{FB}} \tag{16}$$

From (15d) it is apparent that the first and the last term can readily be cancelled out by the feedforward compensation. However, this does not hold true for the second adjoint term, which is multiplied by the virtual input $\ddot{\theta}(t)$. To account for this, we choose $\ddot{\theta}(t) = U_\theta(x_\Gamma(t))$, where $U_\theta(\cdot)$ is the – yet to be defined – *pose-parameter control law* dependent on the augmented state vector $x_\Gamma(t)$ in (4). This choice allows us to also include the adjoint term into the feedforward[5]:

$$\begin{aligned}\hat{U}_{\text{FF}}(t, U_\theta) = & -\hat{F}(t) - U_\theta(x_\Gamma(t))\,\mathrm{Ad}_{\hat{q}_e(t)} \hat{\omega}_d^*(\theta(t)) - \\ & \dot{\theta}(t) \Big[\dot{\hat{q}}_e(t) \circ \hat{\omega}_d^*(\theta(t)) \circ \hat{q}_e(t) + \\ & \hat{q}_e(t) \circ \dot{\theta}(t) \mathring{\hat{\omega}}_d^*(\theta(t)) \circ \hat{q}_e(t) + \hat{q}_e(t) \circ \hat{\omega}_d^*(\theta(t)) \circ \dot{\hat{q}}_e(t) \Big] \ .\end{aligned} \tag{17}$$

Regarding the feedback term, following the pose-tracking formulation in [19], we leverage the logarithmic mapping associated to the Lie group of unit dual quaternions $\mathcal{DQ}_u$ to design a proportional derivative feedback as

$$\hat{U}_{\text{FB}}(t) = -2\hat{\boldsymbol{k}}_p \odot \ln \lambda \hat{q}_e(t) - \hat{\boldsymbol{k}}_v \odot \hat{\omega}_e(t) \,, \tag{18}$$

where $\hat{\boldsymbol{k}}_p$ and $\hat{\boldsymbol{k}}_v$ are vector dual quaternion control gains and $\lambda \in \{-1, 1\}$ is a switching parameter to account for both equilibrium points $\pm\hat{I}$. This is defined as $\lambda = 1$, if $\hat{q}_{e_1}(t) >= 0$ and -1 otherwise, where $\hat{q}_{e_1}$ refers to the first component of $\hat{q}_e(t)$.

E. Stability analysis

In the present subsection, we establish the necessary conditions for control laws $\hat{U}$ and U_θ to almost global asymptotic stability[6].

Theorem 1 (**Stability of pose-following**): *Consider the geometric reference* (3)*, the augmented system* (4)*, the control law $\hat{U}$ in* (16) *with the feedforward and feedback terms in* (17) *and* (18)*, and suppose that the following conditions are satisfied:*

i *The dual quaternion control gains are chosen as $\hat{\boldsymbol{k}}_p > \hat{0}$ with $k_{pd1} = k_{pd2} = k_{pd3}$, i.e., equivalent terms in the dual part of $\hat{\boldsymbol{k}}_p$, and $\hat{\boldsymbol{k}}_v > \hat{0}$.*

ii *The pose-parameter control law ensures that the velocity of the pose-parameter is positive, i.e., $U_\theta(x_\Gamma(t)) \implies \dot{\theta}(t) > 0$, $\forall\, \theta \in [\theta_0, \theta_f]$.*

Then, the closed-loop control scheme defined by system (2) *and control law* (16) *solves the pose-following Problem 1.*

Proof. Starting with *pose convergence* in P1.1, since the feedforward term (17) was designed to eliminate all the nonlinearities in (15d), substituting (16) in (15d) results in $\dot{\hat{\omega}}_e = \hat{U}_{\text{FB}}$. In addition, considering that (14) also remains true for pose-following, the stability analysis in [19] holds.

[4] See Appendix in `https://arxiv.org/pdf/2308.09507.pdf`.

[5] Model 1 in (9) and Model 2 in (15) enable the conversion of $x_\Gamma(t)$ into a dual quaternion error and a dual twist error, as well as the conversion of these errors back into $x_\Gamma(t)$.

[6] We assume perfect and instantaneous state measurements, and thus, the presented global attractivity might be jeopardized by noises and delays that arise in practical applications. This can formally be addressed by combining the proposed method with robust control.

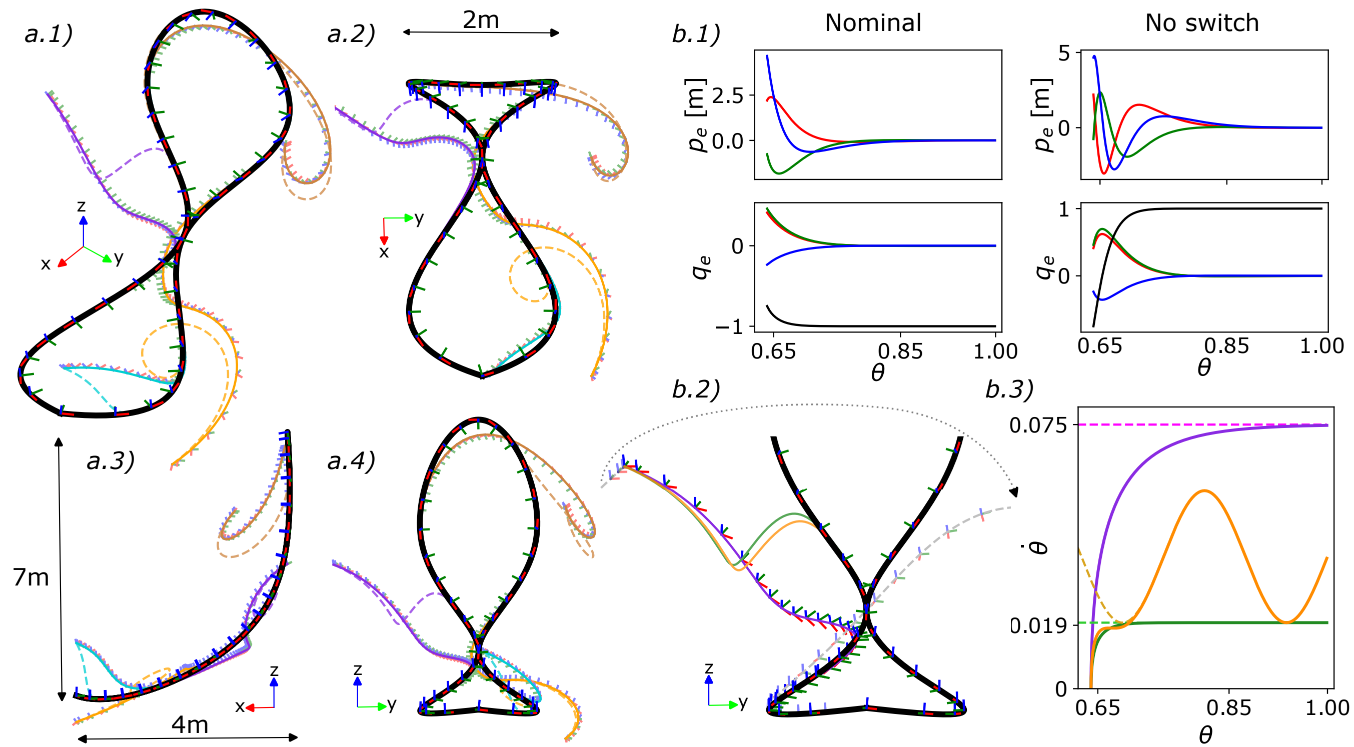

Fig. 1: Rigid body motions obtained from applying control law (16) – with $\hat{U}_\theta$ defined as in Theorem 2 – to the dynamics in (2). The left column (a.1 - a.4) shows the almost global asymptotic stability of the presented pose-following control law by starting from different initial poses (purple, orange, yellow, cyan) from different perspectives. Each starting point is evaluated according to two different constant velocity profiles. The motions associated to the fast profiles are depicted by a continuous line, as well as their orientations, while the motions related to the slow profiles are given by dashed lines. The plots at the top right (b.1), together with the gray dashed line in (b.2), showcase the consequences of deactivating the λ switch. The figure (b.3) shows that the pose-parameter's velocity (continuous line) converges to the desired velocity profile (dashed line) by evaluating three different cases (fast in purple, slow in green and sinusoidal in orange).

This implies that Model 2 converges to the closest equilibrium point $\{\pm\hat{I}, \hat{0}\}$ asymptotically, which directly translates to the fulfillment of pose convergence.

Regarding *convergence on pose-parameter* in P1.2, combining the Lyapunov function $V = ||\theta(t) - \theta_f||^2$ with $\dot{\theta}(t) > 0$ – from (ii) – shows that θ_f is an asymptotically stable equilibrium point. □

Theorem 2 (**Stability of pose-following with velocity assignment**): *Consider the geometric reference* (3)*, the augmented system* (4)*, the control law $\hat{U}$ in* (16) *with the feed-forward and feedback terms in* (17) *and* (18)*, and suppose that the following conditions are satisfied:*

i *The dual quaternion control gains are chosen as $\hat{\boldsymbol{k}}_p > \hat{0}$ with $k_{pd1} = k_{pd2} = k_{pd3}$ and $\hat{\boldsymbol{k}}_v > \hat{0}$.*

ii *The pose-parameter control law is given by $U_\theta(x_\Gamma(t)) = -k_\theta\left(\dot{\theta}(t) - \theta_{vd}(\theta(t))\right) + \dot{\theta}(t)\mathring{\theta}_{vd}(\theta(t))$, where $k_\theta \in \mathbb{R}_{>0}$.*

Then, the closed-loop control scheme defined by system (2) *and control law* (16) *solves the pose-following with velocity assignment Problem 2.*

Proof. The proof for *pose convergence* in P2.1 remains the same as P1.1 in Theorem 1. When it comes to *velocity convergence* in P2.2, the utilization of the Lyapunov function $V = ||\dot{\theta}(t) - \theta_{vd}(\theta(t))||^2$ in conjunction with $U_\theta(x_\Gamma(t))$ as given in (iii), and the recognition that $\ddot{\theta}(t) = U_\theta(x_\Gamma(t))$, indicates that the velocity of the pose-parameter asymptotically converges to the desired velocity profile. □

IV. NUMERICAL EXPERIMENTS

In order to assess the effectiveness of our methodology, we focus on two case studies. The first one focuses on the key properties of the derived control law, including its almost global asymptotic stability and its capability to converge to a predetermined velocity profile. In the second case-study, we demonstrate the advantages of the proposed pose-following control law in comparison to its predecessor, the pose-tracking control law.

Numerical implementation: For all evaluations, the parameters are kept constant as $m = 1$kg, $J = \text{diag}(0.01, 0.01, 0.01)\,\text{Kg/m}^{-2}$, $\hat{\boldsymbol{k}}_p = \hat{\boldsymbol{k}}_v = \hat{\boldsymbol{3}}$ and $k_\theta = 1$.

A. Almost global asymptotic stability on pose-following with velocity assignment

The primary focus of this study is to validate the outcome of Theorem 2: almost global asymptotic stability for pose-following with velocity assignment. When doing so, we intend to verify that, regardless of the initial state, the pose of the rigid body converges to the geometric reference. To this end, we initialize the system at four distinct poses. In addition, we also want to show that this convergence is upheld irrespective of the velocity assignment. To achieve this goal, we evaluate each starting condition according to two distinct profiles, namely a slow one with $\theta_{vd,\text{slow}} = 0.019$ and a fast one with $\theta_{vd,\text{fast}} = 0.075$. As an exemplary geometric reference, we select the same three-dimensional curve as in [30] and to conform to the requirements of (3),

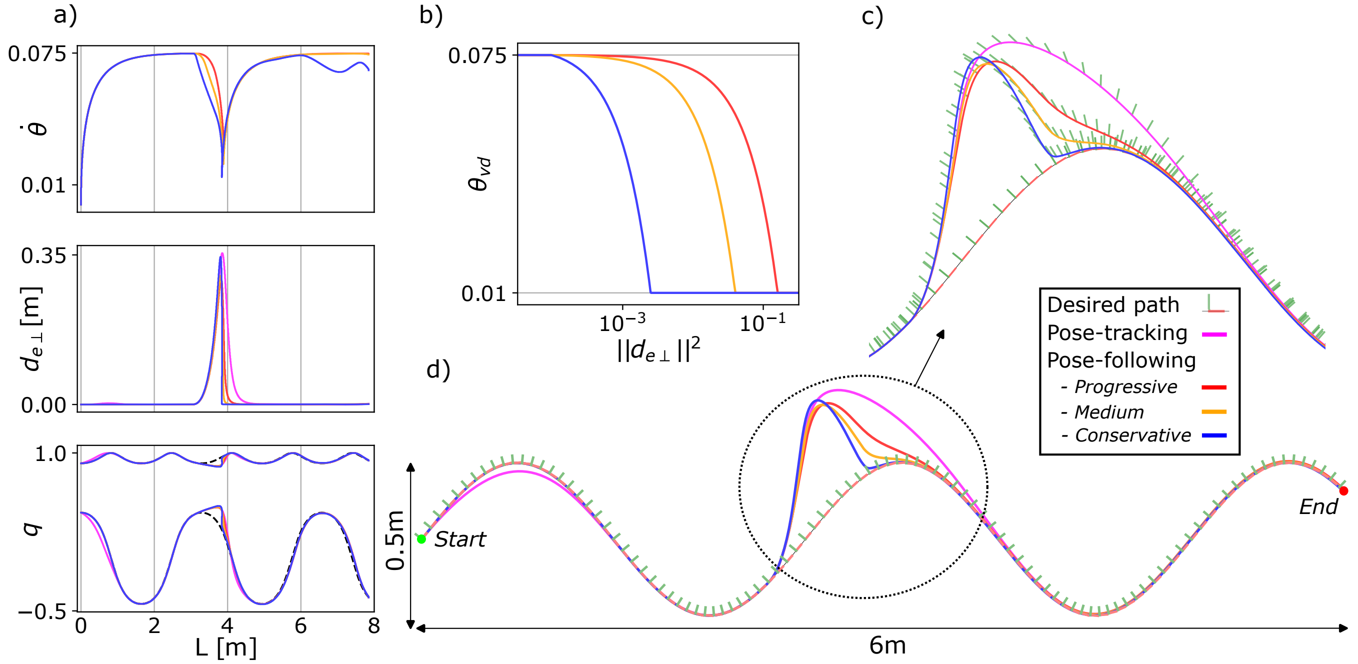

Fig. 2: A comparison between pose-tracking [19] (magenta) and three variants of the presented pose-following control approach (progressive as red, medium as orange and conservative as blue) in the presence of a disturbance. The geometric reference is given by a thin black line and its moving frame. For the rest of the motions, only the normal components are shown for clarity. a) From top to bottom, velocity of pose-parameter, transverse distance to the geometric reference and first and last components of the unit quaternion, where the dashed black line refers to the geometric reference's orientation. b) The desired velocity profile as a mapping from transverse distance to the reference. c) Zoomed comparison in the location of the disturbance and d) Overview of the case-study.

we assign a moving frame to it. For this purpose, akin to [31], [32], we rely on *Pythagorean Hodograph curves*, allowing us to overcome the singularities and discontinuities of the well-known *Frenet-Serret* frame.

The motions resulting from applying the control law (16) – with the pose-parameter control $\hat{U}_\theta$ defined as in Theorem 2 – to the rigid body dynamics in (2) are depicted in the left side of Fig. 1. The motions respective to the runs with the faster velocity profiles are shown by a continuous line and their respective orientations, while the ones related to the slow profiles are represented by dashed lines.

These motions manifest two noteworthy characteristics. The first one being that all of them demonstrate asymptotic convergence towards the geometric reference. The second characteristic, which aligns with common intuition, is that motions corresponding to the slower velocity profiles attain convergence at an earlier stage.

For a more comprehensive analysis, we direct attention to the purple case-study, which refers to the motion located at the top-left corner of Fig. 1(a.1). A magnified view of this case is presented on the right-hand side of Fig. 1(b.2). We hereby validate that the velocity of the pose-parameter, $\dot\theta(t)$, achieves convergence with the desired velocity profile, θ_{vd}. To accomplish this, as demonstrated in Fig. 1(b.3), we examined the convergence in not only slow (green) and fast (purple) constant velocity profiles but also in a sinusoidal profile (orange).

Lastly, we showcase the importance of taking care of the existence of two equilibrium points $\pm\hat{I}$, which in our approach is handled by the switching term λ (18). Within the same case-study as in the previous paragraph, we show that if this switching term is deactivated, the control law only converges to $\hat{I}$ resulting in unnecessarily lengthy and large motion. This can be visualized in the position and quaternion errors, as well as in the resultant motions colored in light gray in Fig.1(b.1-2).

B. Comparison to pose-tracking

Having analyzed the most relevant properties of the presented control law, in this second case-study we compare the performance of the proposed pose-following approach against pose-tracking [19]. For this purpose, we pick a planar sinusoidal curve with a moving frame attached to it as a geometric reference. The task at hand consists of traversing the geometric reference from a zero-velocity pose. However, at the middle of the navigation a longitudinal and angular disturbance is introduced. To ensure a fair comparison, both the pose-tracking and pose-following have been tuned to ensure that the navigation time is the same if no disturbance occurs.

In this experiment the desired velocity profile function is chosen to be dependent on the distance to the geometric reference: $U_\theta(x_\Gamma(t)) = -k_\theta\left(\dot\theta(t) - \theta_{vd}(d_{e\perp}(x_\Gamma(t))\right)$[7] Intuitively, if the system is far away from the reference, it slows down until it is close enough to increase the speed. This mapping is regarded as a tuning parameter that the user can tailor based on system properties and task at hand. In a illustrative manner, we design three variants: progressive (red), medium (orange) and conservative (blue). These velocity profiles alongside their associated motions can be visualized in Fig. 2.

[7] $d_{e,\perp}(x_\Gamma(t))$ is the transverse distance to the geometric reference.

When compared to pose-tracking (in magenta), two differences can be spotted. First, at the very beginning of the trajectory, the tracking method shows a small deviation from the reference. This is due to the fact that the rigid body initially is standing still and needs to catch up with the moving time-reference. In contrast the presented pose-following is aware of its initial state and progressively increases its velocity along the reference. Second, as soon as the disturbance is over, the additional degree of freedom inherited from augmenting the system allows all three variants to slow down and converge back to the geometric reference. This can clearly be visualized in the evolution of $\dot{\theta}$. As expected, the convergence rate directly correlates to how conservative the desired velocity profile mapping is. On the other hand, pose-tracking lacks this additional degree of freedom and has no choice but to catch up with the time-based reference, causing a large deviation error.

V. CONCLUSIONS

In this paper we have formulated a unit dual quaternion-based pose-following control approach for rigid body dynamics. Initially, we have derived the equations of motion for the full pose error between the rigid body and the geometric reference in the form of a dual quaternion and dual twist. Subsequently, we have extended the original control law to account for nonlinearities arising from the introduction of auxiliary states associated with pose-following and designed the additional degree of freedom either to achieve convergence to a desired velocity profile or as a feedback mechanism. When doing so, we have also established almost global asymptotic stability. Lastly, we have numerically validated our findings with two illustrative simulations.

REFERENCES

[1] O.-E. Fjellstad and T. I. Fossen, "Position and attitude tracking of auv's: a quaternion feedback approach," *IEEE Journal of Oceanic Engineering*, vol. 19, no. 4, pp. 512–518, 1994.

[2] D. T. Stansbery and J. R. Cloutier, "Position and attitude control of a spacecraft using the state-dependent riccati equation technique," in *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No. 00CH36334)*, vol. 3. IEEE, 2000, pp. 1867–1871.

[3] J. Arrizabalaga, N. van Duijkeren, M. Ryll, and R. Lange, "A caster-wheel-aware mpc-based motion planner for mobile robotics," in *2021 20th International Conference on Advanced Robotics (ICAR)*. IEEE, 2021, pp. 613–618.

[4] J. Arrizabalaga and M. Ryll, "Towards time-optimal tunnel-following for quadrotors," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 4044–4050.

[5] F. Bullo and R. M. Murray, "Proportional derivative (pd) control on the euclidean group," 1995.

[6] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor uav on se (3)," in *49th IEEE conference on decision and control (CDC)*. IEEE, 2010, pp. 5420–5425.

[7] L. Figueredo, B. V. Adorno, J. Y. Ishihara, and G. A. Borges, "Robust kinematic control of manipulator robots using dual quaternion representation," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 1949–1955.

[8] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, "Mit cheetah 3: Design and control of a robust, dynamic quadruped robot," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 2245–2252.

[9] N. Filipe and P. Tsiotras, "Adaptive position and attitude-tracking controller for satellite proximity operations using dual quaternions," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 4, pp. 566–577, 2015.

[10] A. T. Yang, *Application of quaternion algebra and dual numbers to the analysis of spatial mechanisms*. Columbia University, 1963.

[11] N. Filipe and P. Tsiotras, "Rigid body motion tracking without linear and angular velocity feedback using dual quaternions," in *2013 European Control Conference (ECC)*. IEEE, 2013, pp. 329–334.

[12] Y. Wu, X. Hu, D. Hu, T. Li, and J. Lian, "Strapdown inertial navigation system algorithms based on dual quaternions," *IEEE transactions on aerospace and electronic systems*, vol. 41, no. 1, pp. 110–132, 2005.

[13] Y. Zu, U. Lee, and R. Dai, "Distributed motion estimation of space objects using dual quaternions," in *AIAA/AAS Astrodynamics Specialist Conference*, 2014, p. 4296.

[14] D. Gan, Q. Liao, S. Wei, J. Dai, and S. Qiao, "Dual quaternion-based inverse kinematics of the general spatial 7r mechanism," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 222, no. 8, pp. 1593–1598, 2008.

[15] L. Kavan, S. Collins, C. O'Sullivan, and J. Zara, "Dual quaternions for rigid transformation blending," *Trinity College Dublin, Tech. Rep. TCD-CS-2006–46*, 2006.

[16] K. Daniilidis, "Hand-eye calibration using dual quaternions," *The International Journal of Robotics Research*, vol. 18, no. 3, pp. 286–298, 1999.

[17] D.-P. Han, Q. Wei, and Z.-X. Li, "Kinematic control of free rigid bodies using dual quaternions," *Int. J. Autom. Comput.*, vol. 5, no. 3, pp. 319–324, 2008.

[18] X. Wang, D. Han, C. Yu, and Z. Zheng, "The geometric structure of unit dual quaternion with application in kinematic control," *Journal of Mathematical Analysis and Applications*, vol. 389, no. 2, pp. 1352–1364, 2012.

[19] X. Wang and C. Yu, "Unit dual quaternion-based feedback linearization tracking problem for attitude and position dynamics," *Systems & Control Letters*, vol. 62, no. 3, pp. 225–233, 2013.

[20] F. Zhang and G. Duan, "Robust integrated translation and rotation finite-time maneuver of a rigid spacecraft based on dual quaternion," in *Aiaa guidance, navigation, and control conference*, 2011, p. 6396.

[21] X. Wang, C. Yu, and Z. Lin, "A dual quaternion solution to attitude and position control for rigid-body coordination," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1162–1170, 2012.

[22] M. M. Marinho, L. Figueredo, and B. V. Adorno, "A dual quaternion linear-quadratic optimal controller for trajectory tracking," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 4047–4052.

[23] U. Lee and M. Mesbahi, "Constrained autonomous precision landing via dual quaternions and model predictive control," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 2, pp. 292–308, 2017.

[24] T. Faulwasser and R. Findeisen, "Nonlinear model predictive control for constrained output path following," *IEEE Transactions on Automatic Control*, vol. 61, no. 4, pp. 1026–1039, 2015.

[25] A. P. Aguiar, J. P. Hespanha, and P. V. Kokotovic, "Path-following for nonminimum phase systems removes performance limitations," *IEEE Transactions on Automatic Control*, vol. 50, no. 2, pp. 234–239, 2005.

[26] N. Hung, F. Rego, J. Quintas, J. Cruz, M. Jacinto, D. Souto, A. Potes, L. Sebastiao, and A. Pascoal, "A review of path following control strategies for autonomous robotic vehicles: Theory, simulations, and experiments," *Journal of Field Robotics*, vol. 40, no. 3, pp. 747–779, 2023.

[27] J. Płaskonka, "Different kinematic path following controllers for a wheeled mobile robot of (2, 0) type," *Journal of Intelligent & Robotic Systems*, vol. 77, pp. 481–498, 2015.

[28] 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, no. 1, pp. 63–70, 2000.

[29] B. Kenwright, "A beginners guide to dual-quaternions: what they are, how they work, and how to use them for 3d character hierarchies," 2012.

[30] S. Kumar and R. Gill, "Path following for quadrotors," in *2017 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2017, pp. 2075–2081.

[31] J. Arrizabalaga and M. Ryll, "Spatial motion planning with pythagorean hodograph curves," in *2022 IEEE 61st Conference on Decision and Control (CDC)*. IEEE, 2022, pp. 2047–2053.

[32] J. Arrizabalaga and M. Ryll, "SCTOMP: Spatially constrained time-optimal motion planning," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023.