A Dual-quaternion Method for Control of Spatial Rigid Body

Dapeng Han, Qing Wei and Zexiang Li

***Abstract*— This paper presents a new formulation for control of spatial rigid bodies. After revealing the geometric structure of dual quaternions, logarithmic feedback are utilized to derive control laws in both kinematic design and dynamic design. The regulation problem and the tracking problem are tackled respectively. Comparison with conventional methods together with simulation results are given to show the effectiveness of the new method.**

I. Introduction

It is well known that dual quaternion is an efficient representation tool for rigid transformations. Since its invention, dual quaternion has been applied in various fields, such as mechanical design[1][2], inertial navigation[3], machine vision[4], and other related problems[5]. Specially it provides convenience in describing spatial rigid body with closed chains[6]. In former research dual quaternion is mostly used in kinematic analysis, except for [7], where dynamics of spatial rigid body is formulated in dual quaternion.

This paper presents another viewpoint for using dual quaternion in dynamic issues. Instead of identifying dual quaternions to the Image Space, as was done in [7], we regard dual quaternions as a manifold. When logarithm of dual quaternion is derived, motion control of spatial rigid bodies can be performed using logarithmic feedback via generalized proportional-derivative(PD) laws. The control laws, which utilize the Lie group structure of normalized dual quaternions, are geometric. It is shown that, the new method can handle the coupling between rotation and translation properly, which is usually neglected. Before dynamic control laws can be developed, kinematic issues are discussed explicitly.

We also regard [8] as a preceding work of our research. In that paper, control problems on $SE(3)$ are discussed. We know dual quaternions have an inherent relation with $SE(3)$. Therefore, some results of this paper are derived by imitating the developments in [8].

This paper is organized as follows. After the review of quaternion, dual vector and dual quaternion in Section 2, Section 3 defines the logarithm of dual quaternion, and reveals the geometric structure of dual quaternions. Based on dual-quaternion kinematic control problems are discussed in Section 4. Section 5 presents dual-quaternion based dynamic control laws, together with some simulation results. The paper is concluded by Section 6.

Dapeng Han and Qing Wei are with College of Mechatronic Engineering and Automation, National University of Defense Technology, Changsha, Hunan Province, CO 410073 China (corresponding author to provide phone: 86-731-4574990; e-mail: dphan@nudt.edu.cn)

Zexiang Li is with Department of Electrical and Electronic Engineering, Hong Kong University of Science and Technology, Clearwater Bay, Kowloon, Hong Kong eezxli@ust.hk

II. Mathematical Preliminaries

This section presents a brief review of unit quaternion and dual quaternion. Readers are referred to [3] and [9] for more details.

A. Unit Quaternion

A quaternion usually has the form

$$q = (q_0, q_{\mathbf{v}}) \tag{1}$$

where q_0 is a scalar, $q_{\mathbf{v}} = (q_1, q_2, q_3)$ is a vector.

Two operations of quaternion are listed as below:

$$q^* = (q_0, -q_{\mathbf{v}})$$
$$\| q \|^2 = q \circ q^*$$

where $\circ$ represents quaternion multiplication. When $\| q \| = 1$, we get a unit quaternion.

Unit quaternion can be defined in another way. Rotation about a unit axis $\mathbf{n}$ with angle ϕ is expressed as

$$q = (\cos(\phi/2), \sin(\phi/2)\mathbf{n}) \tag{2}$$

It is naturally a unit quaternion.

The kinematic equation of rotation is given in [3]:

$$2\dot{q} = q \circ \omega^b \tag{3}$$

where ω^b represents the body angular velocity, explicitly expressed as

$$\omega^b = 2q^* \circ \dot{q} \tag{4}$$

B. Dual Vector

A dual vector is defined as

$$\hat{\mathbf{v}} = \mathbf{l} + \varepsilon\mathbf{m}$$

where $\mathbf{l}$ and $\mathbf{m}$ are both real vectors, ε satisfies $\varepsilon^2 = 0$ and $\varepsilon \neq 0$.

Given two dual vectors $\hat{\mathbf{V}}_1 = \mathbf{l}_1 + \varepsilon\mathbf{m}_1, \hat{\mathbf{V}}_2 = \mathbf{l}_2 + \varepsilon\mathbf{m}_2$, the inner product and the cross product can be defined:

$$\mathbf{V}_1 \cdot \mathbf{V}_2 = \mathbf{l}_1 \cdot \mathbf{l}_2 + \varepsilon(\mathbf{m}_1 \cdot \mathbf{l}_2 + \mathbf{l}_1 \cdot \mathbf{m}_2)$$
$$\hat{v}_1 \times \hat{v}_2 = \mathbf{l}_1 \times \mathbf{l}_2 + \varepsilon(\mathbf{l}_1 \times \mathbf{m}_2 + \mathbf{m}_1 \times \mathbf{l}_2)$$

Given a 6-dimensional real vector

$$\mathbf{k} = (k_1, k_2, k_3 \mid k_4, k_5, k_6) \triangleq (\mathbf{k}_a \mid \mathbf{k}_b)$$

For future derivation we define another type of product

$$\mathbf{k} \cdot \hat{\mathbf{V}}_1 = \mathbf{k}_a \cdot \mathbf{l}_1 + \varepsilon\mathbf{k}_b \cdot \mathbf{m}_1 \tag{5}$$

C. *Dual Quaternion*

A dual quaternion has the form

$$\hat{q} = (\hat{q}_0, \hat{q}_{\mathbf{v}})$$

or

$$\hat{q} = q + \varepsilon q^o \tag{6}$$

where $\hat{q}_0$ is a dual scalar, $\hat{q}_{\mathbf{v}}$ is a dual vector, q and q^o are both quaternions.

Given two dual quaternions

$$\hat{q}_1 = q_1 + \varepsilon q_1^o, \hat{q}_2 = q_2 + \varepsilon q_2^o$$

Quaternion multiplication gives

$$\hat{q}_1 \circ \hat{q}_2 = q_1 \circ q_2 + \varepsilon(q_1^o \circ q_2 + q_1 \circ q_2^0) \tag{7}$$

Operations of dual quaternion are similar to that of quaternion:

$$\begin{aligned} \hat{q}^* &= (\hat{q}_0, -\hat{q}_{\mathbf{v}}) \\ \parallel \hat{q} \parallel^2 &= \hat{q} \circ \hat{q}^* \\ \hat{q}^{-1} &= \hat{q}^* / \parallel \hat{q} \parallel^2 \end{aligned} \tag{8}$$

Suppose there is a rotation q succeeded by a translation $\mathbf{p}$. The whole transformation can be represented using a dual quaternion[3]:

$$\hat{q} = q + \varepsilon \tfrac{1}{2} q \circ \mathbf{p} \tag{9}$$

Note that the quaternion $(0, \mathbf{p})$ is identified with the vector $\mathbf{p}$. It follows that

$$\mathbf{p} = 2q^* \circ q^o \tag{10}$$

Kinematic equation of a rigid body expressed in dual quaternion is:

$$2\dot{\hat{q}} = \hat{q} \circ \hat{\omega}^b \tag{11}$$

where

$$\hat{\omega}^b = \omega^b + \varepsilon(\dot{\mathbf{p}} + \omega^b \times \mathbf{p}) \tag{12}$$

represents the generalized body velocity.

When $q \cdot q^0 = 0$, $\hat{q}$ is said to be normalized[1]. It can be verified that, dual quaternion acquired through (9) is naturally normalized. Moreover, we name a set

$$DQ_u = \{\hat{q} | q \cdot q^0 = 0, \parallel q \parallel = 1\}$$

Given a dual quaternion $\hat{q}$, the sufficient and necessary condition for $\hat{q} \in DQ_u$ can be derived by (7):

$$\hat{q} \circ \hat{q}^* = \hat{O}$$

where $\hat{O} = (1,0,0,0) + \varepsilon(0,0,0,0)$.

DQ_u can be regarded as a manifold with 3 dual dimensions. In the rest of this paper, unless otherwise stated, by dual quaternion we mean an element in DQ_u.

III. Geometric Structure of Dual Quaternions

A. *Logarithm of Normalized Dual Quaternion*

Logarithm of unit quaternion has been presented in [10]:

$$\log q = \frac{\phi}{\sqrt{1-q_0^2}} \mathbf{q_v} \tag{13}$$

where $\phi = 2 \arccos q_0 \in [0, 2\pi]$

If a unit quaternion is expressed as (2), its logarithm reads

$$\log q = \tfrac{1}{2} \phi \mathbf{n}, 0 \leqslant \phi \leqslant 2\pi \tag{14}$$

Note that $\phi \pm 2k\pi$ also satisfies (14) for arbitrary integer.

When $\phi = 0$, $q = (1,0,0,0) \triangleq O$; When $\phi = 2\pi$, $q = -O$. Arbitrary $\mathbf{n}$ fits these two cases. Their logarithms are defined specially as $\log O = \log(-O) = (0,0,0)$.

Given a dual quaternion, an interesting conclusion is given in [9]:

$$q + \varepsilon q^0 = q e^{\varepsilon \gamma}, \text{with } \gamma = q^0 / q \tag{15}$$

Substituting (9) into (15) yields

$$\gamma = \tfrac{1}{2} \mathbf{p}$$

Computing the logarithm on both sides of (15) gives

$$\log(q + \varepsilon q^0) = \log q + \varepsilon \tfrac{1}{2} \mathbf{p}$$

Let $\theta = \phi \mathbf{n}$, it follows that

$$\log \hat{q} = \tfrac{1}{2}(\theta + \varepsilon \mathbf{p}) \tag{16}$$

Logarithm of a dual quaternion is a vector with 3 dual dimensions. For future discussion, we use the symbol $\hat{\mathbb{V}}^3$ to represent all 3-dimensional(3D) dual vectors.

Just like the case of unit quaternions, logarithms of $\hat{O}$ and $-\hat{O}$ are defined specially as $(0,0,0) + \varepsilon(0,0,0)$. From the viewpoint of control, $\hat{O}$ and $-\hat{O}$ are the equilibriums of (11).

B. *Normalized Dual Quaternion as a Lie Group*

It has been shown that dual quaternion is a natural generalization of quaternion by the principle of transference[11]. We have known that all unit quaternions form a group[12]. Applying the principle of transference, DQ_u is also a group.

Claim 3.1: DQ_u is a group.

A detailed proof is given in Appendix A.

As DQ_u is also a manifold, it follows that DQ_u is a Lie group.

Logarithmic feedback has been proved to be effective for control on Lie groups[8]. With the logarithm of dual quaternion defined, next we will discuss control on DQ_u.

C. *More Definitions and Discussions*

Given $\hat{q} \in DQ_u$ and its logarithm $\hat{v} \in \hat{\mathbb{V}}^3$, a reasonable definition of the inner products on $\hat{\mathbb{V}}^3$ is

$$< \hat{v}, \hat{v} >= 4 \log \hat{q} \cdot (\log \hat{q})^T = \theta \cdot \theta + 2\varepsilon \theta \cdot \mathbf{p}$$

Now we can define a new type of norm for dual quaternions:

$$\mathcal{R}(\hat{q}) = \sqrt{< \log(\hat{q}), \log(\hat{q}) >} \tag{17}$$

Given $\hat{q} \in DQ_u, \hat{w} \in DQ_u$ and $\hat{v} \in \hat{\mathbb{V}}^3$, adjoint mapping Ad is defined as

$$\begin{aligned} Ad_{\hat{q}}\hat{w} &= \hat{q} \circ \hat{w} \circ \hat{q}^* \\ Ad_{\hat{q}}\hat{v} &= \hat{q}\hat{v}\hat{q}^* \end{aligned} \quad (18)$$

Given two dual quaternions $\hat{q}_1, \hat{q}_2$, we evaluate their difference by

$$\hat{e} = \hat{q}_1^* \circ \hat{q}_2$$

When $\hat{q}_1 = \hat{q}_2$, $\hat{e} = \hat{O}$.

Let $\hat{q}_1 = q_1 + \frac{1}{2}q_1 \circ \mathbf{p}_1, \hat{q}_2 = q_2 + \frac{1}{2}q_2 \circ \mathbf{p}_2$, it can be calculated that

$$\begin{aligned} \hat{e} &= q_e + \tfrac{1}{2}q_e \circ \mathbf{p}_e \\ q_e &= q_1^* \circ q_2 \\ p_e &= \mathbf{p}_2 - Ad_{q_e^*}\mathbf{p}_1 \end{aligned} \quad (19)$$

Note that formula (19) contradicts traditional viewpoint. It considers the coupling between rotation and translation. Similar definition on $SE(3)$ is given in [8].

Given a sequence of transformation $\hat{q}(t) \in DQ_u$,

$$\hat{\mathbf{r}}(t) = \log \hat{q}(t) \text{ and } \hat{\omega}^b(t) = 2\hat{q}^* \circ \dot{\hat{q}}$$

can now be computed. A useful lemma follows.

Lemma 3.1:

$$\frac{1}{2}\frac{d}{dt}\mathcal{R}^2(\hat{q}) = 2 < \hat{\mathbf{r}}, \hat{\omega}^b > \quad (20)$$

Proof is given in Appendix B.

IV. Control Law Design: The Kinematic Case

To stabilize system (11), a control law using logarithmic feedback is presented:

$$\hat{\omega}^b = -2k_p \log \hat{q} \quad (21)$$

To prove the stability, consider the candidate Lyapunov function:

$$W(\hat{q}) = \frac{1}{2}\mathcal{R}^2(\hat{q}) \quad (22)$$

Differentiating (22) and substituting (20)(21) gives

$$\dot{W}(\hat{q}) =< 2\log\hat{q}, -2k_p \log \hat{q} >= -k_p W$$

Thus the logarithmic control law ensures exponential stability.

By "exponential stability" we mean that $\log \hat{q}$ converges to zero exponentially. That is, (21) will drive any initial posture to $\hat{O}$. Note that there is an exception. If the initial posture is $-\hat{O}$, by (21) $\hat{q}$ will stay at $-\hat{O}$.

Actually $\hat{O}$ and $-\hat{O}$ are physically identical. When the initial posture $\hat{q}_0$ is near $-\hat{O}$, that is to say, when the scalar part of $\hat{q}_0$'s quaternion part is negative, it is more reasonable to take $-\hat{O}$ as the equilibrium; otherwise the system will follow a "longer" trajectory leading to $\hat{O}$. To handle this multi-equilibrium problem, we introduce a parameter λ for a given initial posture $\hat{q}(0) = (q_{s0}, q_{\mathbf{v}0}) + \varepsilon(q_{s0}^0, q_{\mathbf{v}0}^0)$:

$$\lambda = \begin{cases} 1, & \text{if } q_{s0} \geqslant 0 \\ -1, & \text{otherwise} \end{cases}$$

Then a revision of (21)

$$\hat{\omega}^b = -2k_p\lambda \log(\lambda\hat{q})$$

will drive the system to $\hat{O}$ or $-\hat{O}$ as demanded.

Note that k_p is not necessary a scalar. It can be replaced by a 6-dimensional vector, and the calculation can follow the special inner product defined in (5). Similar trick will be inherited in the following derivations.

A. The Tracking Problem

Given a reference trajectory $\hat{q}_d(t)$, the demanded velocity is express as

$$\hat{\omega}_d = 2\hat{q}_d^* \circ \dot{\hat{q}}_d$$

As $\hat{q}_d$ is normalized, differentiating the equality

$$\hat{q}_d \circ \hat{q}_d^* = \hat{O}$$

yields

$$2\dot{\hat{q}}_d^* = -\hat{\omega}_d \circ \hat{q}_d^* \quad (23)$$

The tracking error is calculated as

$$\hat{e} = \hat{q}_d^* \circ \hat{q} \quad (24)$$

Differentiating (24), applying (23) and rearranging the terms gives the kinematic error system:

$$\begin{cases} 2\dot{\hat{e}} = \hat{e} \circ \hat{\omega}_e \\ \hat{\omega}_e = \hat{\omega}^b - Ad_{\hat{e}^*}\hat{\omega}_d \end{cases} \quad (25)$$

Combining The tracking law is easily derived by applying (21) to (25):

$$\hat{\omega}^b = -2k_p \log \hat{e} + Ad_{\hat{e}^*}\hat{\omega}_d \quad (26)$$

B. Comparison and Analysis

In this section, we take omnidirectional control as an example to show the difference of the new method from the conventional one.

The posture of a omnidirectional robot is described by (x, y, θ), with x, y being the Cartesian coordinates, and θ being the angle between the heading direction and the x-axis. Corresponding to arbitrary element in $SE(2)$ there is a normalized dual quaternion $\hat{q} = (q_1, q_2, q_3, q_4) + \varepsilon(q_5, q_6, q_7, q_8)$. Its relation with (x, y, θ) is

$$\begin{aligned} q_1 &= \cos(\theta/2) \\ q_4 &= \sin(\theta/2) \\ q_6 &= x\cos(\theta/2) + y\sin(\theta/2) \\ q_7 &= y\cos(\theta/2) - x\sin(\theta/2) \\ q_2 &= q_3 = q_5 = q_8 = 0 \end{aligned} \quad (27)$$

Further,

$$\begin{aligned} \log\hat{q} &= (0, 0, \tfrac{\theta}{2}) + \varepsilon(\tfrac{x}{2}, \tfrac{y}{2}, 0) \\ \hat{\omega}^b &= (0, 0, \omega_z) + \varepsilon(v_x - \omega_z y, v_y + \omega_z x, 0) \end{aligned} \quad (28)$$

Traditionally the three degree-of-freedoms(DOFs) of an omnidirectional robot are decoupled when performing control. Given a reference trajectory $(x_d(t), y_d(t), \theta_d(t))$, the tracking control law reads

$$
\begin{aligned}
\dot{\theta} &= -k_0\theta_e + \dot{\theta}_d \\
\dot{x} &= -k_1 x + \dot{x}_d \\
\dot{y} &= -k_2 y + \dot{y}_d
\end{aligned}
\tag{29}
$$

where $\theta_e = \theta - \theta_d$.

A new tracking law can be derived by substituting (27)-(28) into (26), though the calculation is somewhat tedious:

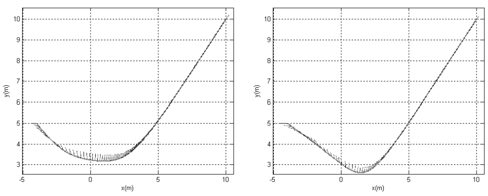
$$
\begin{aligned}
\dot{\theta} &= -k_0\theta_e + \dot{\theta}_d \\
\dot{x} &= -k_1 x + (\dot{x}_d \cos\theta_e - \dot{y}_d \sin\theta_e) \\
&\quad -k_1(x_d \cos\theta_e - y_d \sin\theta_e) \\
&\quad -k_1\theta_e(x_d \sin\theta_e + y_d \cos\theta_e) \\
\dot{y} &= -k_2 y + (\dot{x}_d \sin\theta_e + \dot{y}_d \cos\theta_e) \\
&\quad -k_2(x_d \sin\theta_e + y_d \cos\theta_e) \\
&\quad +k_2\theta_e(x_d \cos\theta_e - y_d \sin\theta_e)
\end{aligned}
\tag{30}
$$

Comparing (30) to (29), we find extra terms. It is not easy to understand from the traditional viewpoint. A simple explanation is that, these terms represent the effect of rotation on translation.

Given a reference trajectory

$$
\begin{aligned}
x_d(t) &= t \\
y_d(t) &= t \\
\theta_d(t) &= \tfrac{1}{4}\pi
\end{aligned}
$$

Starting with the initial posture $(-4, 5, 0)$, taking $k_0 = k_1 = k_2 = 1$, applying (30), the actual trajectory is given in Fig. 1(a). As a comparison, simulation result by (29) is shown in Fig. 1(b). As can be seen, resulting trajectory of the new tracking law has lower curvatures. For a vehicle with high speed, it can lead to less risk of slippage.



(a) (b)

Fig. 1. Kinematic tracking simulation. The starting posture is $(-4, 5, \frac{\pi}{2})$. The short bar perpendicular to the trajectory represents the orientation angle. On the left is the result of the new tracking law (30), on the right is the result of (29).

V. Control Law Design: The Dynamic Case

A. Review of Geometric Method

In [8], control problems on Euclidean group are discussed, and geometric control laws are developed. The authors present a general dynamic model for all control problems on $SE(3)$:

$$
\begin{aligned}
\dot{g} &= g \circ \omega^b \\
\dot{V} &= f(g, V) + U
\end{aligned}
$$

where $g \in SE(3)$, V is the generalized velocity, $f(g, V)$ and U vary with particular systems.

For the regulation problem, a generalized proportional-derivative law that guarantees approximate exponential convergence is proposed

$$
U = -k_p \log g - k_d \omega^b - f(g, V) \tag{31}
$$

B. Regulation on DQ_u

Firstly we establish a general dynamic model using dual quaternion.

Differentiating (12) yields

$$
\dot{\hat{\omega}}^b = \dot{\omega}^b + \varepsilon(\ddot{\mathbf{p}} + \omega^b \times \dot{\mathbf{p}} + \dot{\omega}^b \times \mathbf{p}) \tag{32}
$$

A general dynamic model can be written as

$$
\begin{aligned}
2\dot{\hat{q}} &= \hat{q} \circ \hat{\omega}^b \\
\dot{\hat{\omega}}^b &= f(\hat{q}, \hat{\omega}^b) + U
\end{aligned}
$$

with

$$
\begin{aligned}
&f(\hat{q}, \hat{\omega}^b) = \varepsilon\omega^b \times \dot{\mathbf{p}} \\
&U = \dot{\omega}^b + \varepsilon(\ddot{\mathbf{p}} + \dot{\omega}^b \times \mathbf{p})
\end{aligned}
$$

Formula (31) is rewritten as

$$
U = -2k_p \log \hat{q} - k_d \hat{\omega}^b - f(\hat{q}, \hat{\omega}^b) \tag{33}
$$

where we have replaced $\log g$ and V by $2\log\hat{q}$ and $\hat{\omega}^b$ respectively.

To handle the multi-equilibrium problem, we still use the indicating parameter λ. Let $\bar{\hat{q}} = \lambda\hat{q}$ and set

$$
\begin{aligned}
\bar{\hat{\omega}}^b &= \lambda\hat{\omega}^b \\
\bar{U} &= \lambda U \\
\bar{f} &= \lambda f(\hat{q}, \hat{\omega}^b)
\end{aligned}
$$

Formula (33) can now be revised as

$$
\bar{U} = -k_p \log \bar{\hat{q}} - k_d \bar{\hat{\omega}}^b - \bar{f}(\hat{q}, \hat{\omega}^b)
$$

which implies a new regulation law:

$$
U = -k_p \lambda \log \lambda\hat{q} - k_d \hat{\omega}^b - f(\hat{q}, \hat{\omega}^b) \tag{34}
$$

The conventional way to perform control on $SE(3)$, called double geodesic control law, is addressed in [8]. Next we compare it with the new method.

To give an intuitionistic result, we restrict an omnidirectional robot to move on the x-y plane while rotating about z-axis. The posture is described by $\mathbf{p} = (x, y, 0)$ and $\log q = \frac{1}{2}(0, 0, \theta_z)$.

Applying (34), simulation result is presented in Fig. 2(a). Initially the robot is at the position $(-5, 5, 0)$ with $\theta_3 = \frac{\pi}{2}$. The gains k_p, k_d is chosen as

$$
\begin{aligned}
k_p &= (1, 1, 1, 4, 4, 4) \\
k_d &= (0.5, 0.5, 0.5, 2, 2, 2)
\end{aligned}
$$

Note that short bars are used to indicate the heading angle θ_z in the following figures.

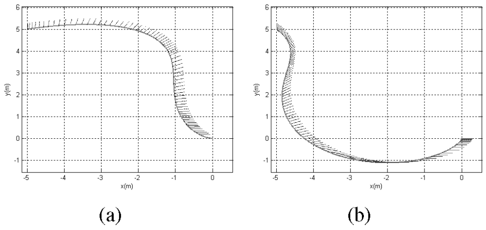


Fig. 2. A rigid body moving on the x-y plane is driven to the origin. The short bar perpendicular to the curve represents the heading angle. On the left is the result of the dual-quaternion method, on the right is the result of the conventional method.

Double geodesic law using equivalent gains is also applied to repeat the simulation, and the result is given in Fig. 2(b). Compared to the dual-quaternion based regulation law, the double geodesic law generates a spiraling (x, y) motion. This is due to the neglect of the interacting term $\omega^b \times p$ in (12).

C. Tracking Problem

The kinematic error system has been constructed as (25), from which the dynamic error system can be derived.

Firstly, following (23) we have

$$2\dot{\hat{e}}^* = -\hat{\omega}_e \circ \hat{e}^*$$

Also we know

$$\hat{\omega}_e \times Ad_{\hat{e}^*}\hat{\omega}_d = -Ad_{\hat{e}^*}\hat{\omega}_d \times \hat{\omega}_e$$

Let $\hat{\Omega}_d$ be the demanded acceleration. In (25) the reference trajectory brings a compensating term $Ad_{\hat{e}^*}\hat{\omega}_d$. Differentiating the term yields

$$(Ad_{\hat{e}}\hat{\omega}_d)' = Ad_{\hat{e}^*}\hat{\Omega}_d + Ad_{\hat{e}^*}\hat{\omega}_d \times \hat{\omega}_e$$

Let $\hat{\Omega}$ be the generalized acceleration. After differentiation on (25), the dynamic error system is stated as

$$\begin{aligned} 2\dot{\hat{e}} &= \hat{e} \circ \hat{\omega}_e \\ \dot{\hat{\omega}}_e &= f(\hat{q}, \hat{\omega}) - U_{tr} + U \end{aligned} \tag{35}$$

where

$$U_{tr} = Ad_{\hat{e}^*}\hat{\Omega}_d + Ad_{\hat{e}^*}\hat{\omega}_d \times \hat{\omega}_e$$

Following (33) the dynamic tracking control law is easily obtained:

$$\hat{U} = -k_p \log(\hat{e}) - k_d\hat{\omega}_e + U_{tr} - f(\hat{q}, \hat{\omega}^b) \tag{36}$$

Simulation is done to compare (36) with the double geodesic law. Using equivalent gains the control laws are both applied on an omnidirectional ground robot. The starting posture is $(-4, 0, 0)$, and the reference trajectory to be tracked is

$$\begin{aligned} x &= t \\ y &= t \\ \theta_z &= \tfrac{3}{4}\pi \end{aligned}$$

When the derivative parameter k_d has a small amplitude, the two resulting trajectories together with the reference trajectory are shown in Fig. 3(a) and Fig. 3(b) respectively. As can be seen, when the new dynamic tracking law is applied the resulting trajectory is more smooth.

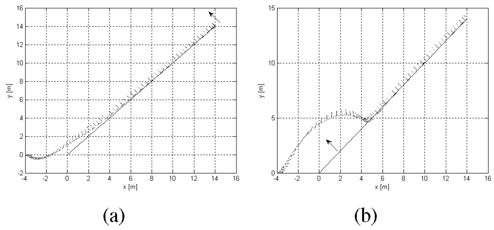


Fig. 3. Tracking of a line with fixed orientation angle. The short bars attached to the trajectory represent the heading angle. The arrow represents the demanded heading direction. On the left is the result of the dual-quaternion method, on the right is the result of the conventional method.

VI. Concluding Remarks

Based on dual quaternion we develop a new method for control of spatial rigid bodies in this paper. Both kinematic and dynamic problems are tackled. Compared to the conventional method, the control laws can handle the coupling between rotation and translation properly, making the resulting trajectory more natural.

Due to its computational efficiency, dual quaternion is a wonderful algebraic tool for motion design. This paper reveals the geometric structure of dual quaternion and presents new control laws. Combining dual-quaternion based trajectory planning schemes with the geometric control laws will make a prospective solution for many applications, such as industrial robots and cooperative manipulators, which will be covered by future research.

Appendix

A. Proof of Claim 3.1

According to the definition, a dual quaternion $\hat{Q}$ is a normalized one if and only if

$$\hat{Q} \circ \hat{Q}^* = \hat{O}$$

Obviously, if $\hat{Q} \in DQ_u$, $\hat{Q}^* \in DQ_u$.

Given two quaternions $\hat{Q}_1$ and $\hat{Q}_2$, it can be verified by direct calculation that

$$(\hat{Q}_1 \circ \hat{Q}_2)^* = \hat{Q}_2^* \circ \hat{Q}_1^*$$

If $\hat{Q}_1$ and $\hat{Q}_2$ are elements in DQ_u, it follows that

$$(\hat{Q}_1 \circ \hat{Q}_2)(\hat{Q}_1 \circ \hat{Q}_2)^* = \hat{Q}_1 \circ \hat{Q}_2 \circ \hat{Q}_2^* \circ \hat{Q}_1^* = \hat{O}$$

Therefore, $\hat{Q}_1 \circ \hat{Q}_2 \in DQ_u$. Moreover, given another dual quaternion $\hat{Q}_3 \in DQ_u$, it can also be verified that

$$(\hat{Q}_1 \circ \hat{Q}_2) \circ \hat{Q}_3 = \hat{Q}_1 \circ (\hat{Q}_2 \circ \hat{Q}_3)$$

Hence, with $\hat{O}$ being the identity element, and with $\hat{Q}^*$ being the inverse of $\hat{Q}$, DQ_u is a group.

B. *Proof of Lemma 3.1*

Starting from (17), standard calculation gives

$$\begin{aligned} \tfrac{1}{2}\tfrac{d}{dt}\mathcal{R}^2(\hat{q}) &= < 2\log(\hat{q}), 2\tfrac{d}{dt}\log(\hat{q}) > \\ 2\tfrac{d}{dt}\log(\hat{q}) &= (\dot{\phi}\mathbf{n} + \varepsilon\dot{p}\mathbf{s}) + (\phi\dot{\mathbf{n}} + \varepsilon\phi\dot{\mathbf{n}}) \\ &\triangleq C_{\parallel} + C_{\perp} \end{aligned}$$

As $\mathbf{n}$ and $\mathbf{s}$ are both unit vectors,

$$\mathbf{n} \cdot \dot{\mathbf{n}} = \mathbf{s} \cdot \dot{\mathbf{s}} = 0$$

Together with (16) it follows that

$$< 2\log(\hat{q}), C_{\perp} >= 0$$

$$\tfrac{1}{2}\tfrac{d}{dt}\mathcal{R}^2(\hat{q}) =< 2\log(\hat{q}), C_{\parallel} >$$

Following (2)(4)(12), $\hat{\omega}^b$ is calculated as

$$\begin{aligned} \hat{\omega}^b &= (\dot{\phi}\mathbf{n} + \varepsilon\dot{p}\mathbf{s}) + (\sin\phi\dot{\mathbf{n}} + 2\sin^2\tfrac{\phi}{2}\dot{\mathbf{n}} \times \mathbf{n}) \\ &\quad +\varepsilon(p\dot{\mathbf{s}} + \dot{p}\dot{\phi}\mathbf{n} \times \mathbf{s} + p\dot{\phi}\mathbf{n} \times \dot{\mathbf{s}} + \phi\dot{p}\dot{\mathbf{n}} \times \mathbf{s}) \\ &\quad +\varepsilon(\phi p\dot{\mathbf{n}} \times \dot{\mathbf{s}} + 2\sin^2\tfrac{\phi}{2}\dot{\mathbf{n}} \times \mathbf{s} \times \dot{\mathbf{s}}) \\ &\triangleq C_{\parallel} + \bar{C}_{\perp} \end{aligned}$$

Obviously

$$< 2\log(\hat{q}), \bar{C}_{\perp} >= 0$$

Thus

$$< 2\log(\hat{q}), \hat{\omega}^b >=< 2\log(\hat{q}), C_{\parallel} >$$

Then (20) follows.

REFERENCES

[1] Q.J. Ge and B.Ravani. Computer aided geometric design of motion interpolants. ASME Journal of Mechanical Design, 116(3): 756-762, Sept. 1994.

[2] A. Perez and J. M. McCarthy. Dual Quaternion Synthesis of Constrained Robotic Systems. Journal of Mechanical Design, 126(3): 425-435, May 2004.

[3] Y.X. Wu, X.P. Hu, D.W. Hu, and J. X. Lian. Strapdown Inertial Navigation System Algorithms Based on Dual Quaternions. IEEE Transaction on Aerospace and Electronic Systems, 41(1):110-132, Jan. 2005.

[4] J. James Samuel Goddard. Pose and motion estimation from vision using dual quaternion-based extended Kalman filtering. PhD thesis, The University of Tennessee, Knoxville, 1997.

[5] K. Daniilidis. Hand-Eye Calibration Using Dual Quaternions. The International Journal of Robotics Research, 18(3):286-298, March 1999.

[6] J.R. Dooley and J.M. McCarthy. On the Geometric Analysis of Optinum Trajectories for Cooperating Robots using Dual Quaternion Coordinates. Proceedings of the 1993 International Conference on Robotics and Automation, pages:1031-1036, 1993.

[7] J.R. Dooley and J.M. McCarthy. Spatial Rigid Body Dynamics using Dual Quaternion Components. Proceedings of the 1991 International Conference on Robotics and Automation, Sacramento California, pages:90-95, April 1991.

[8] F. Bullo and R.M. Murray. Proportional derivative (PD) control on the Euclidean group, CDS Technical Report 95-010, 1995.

[9] O.Bottema and B.Roth. Theoretical Kinematics. North-Holland Publishing Company,New York,1979.

[10] M.-J. Kim and M.-S. Kim. A Compact Differential Formula for the First Derivative of a Unit Quaternion Curve. Journal of Visualization and Computer Animation, 7(1): 43-57, July 1996.

[11] L.M. Hsia and A.T. Yang. On the Principle of Transference in Three-Dimensional Kinematics. ASME Journal of Mechnical Design, 103:652-656, 1981.

[12] R.M. Murray, Z.X. Li, and S.S. Sastry. An Mathematical Introduction to Robotic Manipulation. CRC Press, 1994.