Coupled 6-DOF Control for Distributed Aerospace Systems

Taylor P. Reynolds and Mehran Mesbahi

***Abstract*— In this paper, we investigate the problem of simultaneously controlling both attitude and position of a network of collaborative aerospace vehicles. In particular, we use unit dual quaternions to model the coupled rotational and translational motion present in many applications. We then derive a simple PD-like feedback controller that simultaneously stabilizes the attitude and position of all vehicles in the network. The analysis reveals global asymptotic stability using LaSalle's invariance principle. We discuss various applications of the control design framework and provide a numerical example of a spacecraft landing on a moving platform. This example demonstrates the utility of the proposed approach to 6-degree-of-freedom multiagent coordination that can be achieved with a single framework.**

I. INTRODUCTION

It is posited that smaller, fractionated spacecraft working harmoniously can achieve the same if not greater mission objectives than a single larger vehicle. Moreover, a distributed approach to spacecraft design increases robustness, and may decrease the overall mission cost. In missions involving several spacecraft, the need to interact and exchange information becomes paramount. To achieve system level performance requirements, this information exchange must be modeled and accounted for in the design of each vehicle's control system.

Information exchange is naturally modeled using graph theoretic methods. Graph theory and networked systems provide a flexible set of tools with which to model many different information-exchange topologies. Different network configurations can subsequently be abstracted using linear algebraic techniques, and the effect on control performance can then be examined [1]. In many applications of interest, a network of aero- or space-vehicles are required to maintain both relative distances and orientations from one another. Examples include space-based interferometry, formation flying maneuvers, aerial surveillance, and as we will demonstrate subsequently, landing on moving platforms.

The attitude control portion of this problem has been studied extensively in [2], [3], [4], [5], [6], [7], among other works. In this direction, a common technique is to use graph theory combined with quaternion based attitude parameterizations to derive a stabilizing controller. Each of the references above takes on either a decentralized or distributed approach to attitude synchronization, and does not consider the position states in the control synthesis. Extensions from quaternions to dual quaternions in order to handle both translational and rotational degrees of freedom in the coordination algorithm become nontrivial due to the introduction of dual numbers and the dual quaternion algebraic operations.

This research has been supported by NASA grant NNX17AH02A SUP02, AFOSR grant FA9550-16-1-0022 and NSERC grant PGSD3-502758-2017.

The authors are with the W.E. Boeing Department of Aeronautics & Astronautics, University of Washington, Seattle, WA, USA. Emails: {tpr6, mesbahi}@uw.edu

The main contribution of this work is to derive a feedback control law that simultaneously stabilizes both position *and* attitude of an arbitrary number of fully-actuated rigid aerospace vehicles, while leveraging information exchange in the underlying network. The 6-degree-of-freedom (6-DOF) motion is modeled using unit dual quaternions and we leverage the coupling of rotational and translational motion in the control design to derive a single, intuitive control law. This approach has widespread applications; indeed modeling with dual quaternions is rather general and can be applied to scenarios beyond just those discussed here. Similar work was presented in [9] where the authors propose a control law for 6-DOF motion using the dual quaternion logarithm. In [10], [11], [12], a two-agent leader-follower scenario is analyzed and the relative motion is controlled using dual quaternions. For a single spacecraft, [13] presents a dual quaternion based feedback control law with notational description that differs from those adopted in this paper.

Controller design for 6-DOF systems without dual quaternions has been studied in more depth, often by modeling the system using a combination of Cartesian variables (position) and quaternions (attitude) [14]. In [15] a consensus type control law is derived using Lie groups and homogeneous matrices on $SE(3)$. Along these lines, [16] proposes a control law derived for 6-DOF motion on $SE(3)$ by separately designing stable feedback controls for rotational and translational motion.

The current work yields a single control law that drives each spacecraft in the network to any desired (possibly time-varying) position and attitude state with global asymptotic convergence guarantees. The presented results differ from the previous works mentioned above in that: (1) we consider an arbitrary number of agents, not necessarily a two agent leader-follower scenario, (2) dual quaternions are used to express rigid body motion, leading to a more compact form and introducing unique algebraic considerations, (3) we do not parameterize the error signal with the quaternion logarithm, leading to – in our view – a more elegant control law. A PD-like feedback component coupled with a dual quaternion consensus term provides the first dual quaternion result that mirrors the elegance of its quaternion counterpart (cf. [2], [17]).

The paper is organized as follows. §II provides the background information on graph theory, dual quaternions and rigid body mechanics. §III states the main results; §III-A provides a numerical simulation to demonstrate the utility of the presented results. Finally, §IV summarizes our contribution and provides directions for future work.

1) Notation: The use of quaternions and dual quaternions necessitates a word on the notation, since there are several conventions. Vector quantities are denoted using lower case bold-faced symbols, while scalar quantities are lower case, regular font symbols, and matrices are upper case symbols. All dual quaternions are denoted using a tilde, $\tilde{\cdot}$, to distinguish them from their quaternion counterparts. A vector $\boldsymbol{r}$ resolved in the frame $\mathcal{F}_b$ is denoted by $\boldsymbol{r}^b$. When multiple body frames, $\mathcal{F}_{b_i}$, $i = 1, \ldots, n$, are used, the same vector is represented using $\boldsymbol{r}^i$. The representations $\boldsymbol{r}^i$ and ${}^i\boldsymbol{r}$ are taken as equivalent. In this work, it is assumed that the scalar part of the quaternion occupies the last spot in the vector representation. The vector part of the quaternion is denoted using the subscript $\boldsymbol{q}_v$ and the scalar part is q_0. The identity quaternion is $\boldsymbol{q}_I = [0\ 0\ 0\ 1]^T$. The vector and scalar parts of a dual quaternion are defined using the same notation, with the tilde adornment serving to distinguish them from regular quaternions. The "vector" part, $\tilde{\boldsymbol{q}}_v$, is thus a 6×1 vector in which both scalar parts have been omitted, and the "scalar" part, $\tilde{\boldsymbol{q}}_s$, is a 2×1 vector containing the real and dual scalar parts.[1]

II. Background

We first provide a brief overview of constructs used in the paper.

A. Graph Theory

Graphs provide a natural abstraction for information exchange between agents operating in a network [1]. An *unweighted* graph is defined by a set of nodes, $\mathcal{N}$, and a set of edges, $\mathcal{E}$. The set of nodes is an enumeration of the form $\mathcal{N} = \{1, 2, \ldots, n\}$, with each agent in the network assigned a fixed $i \in \mathcal{N}$. The edge set is a subset of $\mathcal{N} \times \mathcal{N}$, and we say that $(i, j) \in \mathcal{E}$ if information can be exchanged from agent i to agent j in the network.[2] We say the graph is undirected if $(i, j) \in \mathcal{E}$ implies $(j, i) \in \mathcal{E}$, an assumption made in this work. For an undirected graph, the adjacency matrix is the symmetric matrix such that $A = [a_{ij}]$, where $a_{ij} = 1$ when $(i, j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. The degree matrix, $\Delta = [d_{ij}]$, is the diagonal matrix such that d_{ii} is the number of neighbors of agent $i \in \mathcal{N}$. The graph Laplacian is formed by taking the difference between the degree and adjacency matrices according to $L = \Delta - A$.

If we consider solely the agent's position as the state, then the consensus dynamics (see [1] Ch. 3) lead to simple and distributed control laws using the Laplacian. If we consider the agent's attitude parameterized by the quaternion, then we cannot leverage the same consensus dynamics due to the quaternion algebra. When we further extend to *both* position and attitude using dual quaternions, we encounter the additional algebraic imposition of dual numbers. However, we show here that dual quaternions naturally combine consensus dynamics for the relative position states while incorporating the previously developed quaternion error based attitude consensus.

B. Dual Quaternions

Dual quaternions allow encoding of both relative position and orientation in a single parameter. This in turn facilitates studying the full motion of rigid bodies in a single framework, without assuming that the rotational and translational states are independent from one another. This assumption may be valid in some cases, like when the translational states evolve over time scales that are much longer than those of the rotational states. However, mission level constraints and actuation mechanisms can easily couple these two types of motion. Dual quaternions are an effective method with which we can compute stabilizing controllers that take into account this coupling.

For this work, we denote the set of quaternions as $\mathbb{Q} = \{\boldsymbol{q} \mid \boldsymbol{q} = [\boldsymbol{q}_v^T\ q_0]^T\}$. We write the set of unit quaternions as $\mathbb{Q}_u = \{\boldsymbol{q} \in \mathbb{Q} \mid \boldsymbol{q}^T\boldsymbol{q} = 1\}$. A unit quaternion is generally used to express the attitude of a rigid body with respect to some fixed inertial frame. We adopt the convention throughout that $\boldsymbol{q} = \boldsymbol{q}_{b \leftarrow I}$, where $\mathcal{F}_b$ denotes a coordinate frame fixed to, and rotating with, a rigid body, and $\mathcal{F}_I$ denotes an inertial frame.

Dual quaternions are a generalization of quaternions that use dual numbers as coefficients. A dual number in this setting is the nilpotent quantity ϵ, where $\epsilon^2 = 0$ but $\epsilon \neq 0$. We define the set of dual quaternions as $\mathbb{Q}^2 = \{\tilde{\boldsymbol{q}} = \boldsymbol{q}_1 + \epsilon\boldsymbol{q}_2 \mid \boldsymbol{q}_1, \boldsymbol{q}_2 \in \mathbb{Q}\}$. We call $\boldsymbol{q}_1$ the real (rotational) part of the dual quaternion, and $\boldsymbol{q}_2$ the dual (displacement) part. Analogous to the quaternion case, it is *unit* dual quaternions that will be of use in parameterizing rotational and translational motion.

It is important to take a minute and explain why it is necessary to use unit dual quaternions, rather than dual quaternions, to parameterize rigid body motion. There are six degrees of freedom in general rigid body motion; three from free rotation, and three from free translation. However, the eight parameter dual quaternion would seem to have eight degrees of freedom; four from the real part and four from the dual part. Two constraints are therefore necessary to reduce the number of degrees of freedom to match that of rigid body motion. Unit dual quaternions provide exactly this, as

$$\tilde{\boldsymbol{q}}^T\tilde{\boldsymbol{q}} = \boldsymbol{q}_1^T\boldsymbol{q}_1 + \epsilon\left(\boldsymbol{q}_1^T\boldsymbol{q}_2 + \boldsymbol{q}_2^T\boldsymbol{q}_1\right) = 1, \tag{1}$$

which makes clear the two constraints that are imposed:

$$\boldsymbol{q}_1^T\boldsymbol{q}_1 = 1, \quad \text{and} \quad \boldsymbol{q}_1^T\boldsymbol{q}_2 = 0. \tag{2}$$

The first is the usual unit quaternion constraint that reduces the number of rotational degrees of freedom to three. The second is a constraint on the dual part, reducing the number of translational degrees of freedom to three.

[1]This is a slight abuse of the terms vector and scalar, hence the quotations. They are in reality dual vectors and dual numbers, but can be thought of for simplicity as vectors of the appropriate size.

[2]We assume there are no self-loops.

This facilitates the definition of the set of unit dual quaternions as

$$\mathbb{Q}_u^2 = \{\tilde{\boldsymbol{q}} = \boldsymbol{q}_1 + \epsilon \boldsymbol{q}_2 \mid \boldsymbol{q}_1 \in \mathbb{Q}_u,\ \boldsymbol{q}_2 \in \mathbb{Q},\ \boldsymbol{q}_1^T \boldsymbol{q}_2 = 0\}.$$

The set of unit dual quaternions can therefore be thought of as the regular quaternion hypersphere plus each of its tangent hyperplanes. For the majority of this work, unit dual quaternions can be embedded in an eight dimensional Euclidean vector space, and thought of as $\tilde{\boldsymbol{q}} = [\boldsymbol{q}_1^T\ \boldsymbol{q}_2^T]^T \in \mathbb{R}^8$, which will greatly simplify algebraic manipulations. In addition to these definitions, we call $\mathbb{Q}_v$ the set of pure quaternions, which have a zero scalar part. In the case of $\mathbb{Q}_v^2$, both the real and dual parts are pure quaternions.

C. Dual Quaternion Operations

Dual quaternion algebraic operations are defined in terms of their quaternion counterparts. If $\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{q}} \in \mathbb{Q}^2$, then we define dual quaternion multiplication as $\tilde{\boldsymbol{q}} \otimes \tilde{\boldsymbol{p}} = [\tilde{\boldsymbol{q}}]_\otimes \tilde{\boldsymbol{p}}$, where

$$[\tilde{\boldsymbol{q}}]_\otimes = \begin{bmatrix} [\boldsymbol{q}_1]_\otimes & 0_{4\times 4} \\ [\boldsymbol{q}_2]_\otimes & [\boldsymbol{q}_1]_\otimes \end{bmatrix}, \quad [\boldsymbol{q}]_\otimes = \begin{bmatrix} \boldsymbol{q}_v^\times + q_0 I_3 & \boldsymbol{q}_v \\ -\boldsymbol{q}_v^T & q_0 \end{bmatrix}, \quad (3)$$

where quaternion multiplication is defined as $\boldsymbol{q} \otimes \boldsymbol{p} = [\boldsymbol{q}]_\otimes \boldsymbol{p}$ and $\cdot^\times$ is the skew-symmetric cross product operation. We define the dual quaternion cross product as $\tilde{\boldsymbol{q}} \oslash \tilde{\boldsymbol{p}} = [\tilde{\boldsymbol{q}}]_\oslash \tilde{\boldsymbol{p}}$, where

$$[\tilde{\boldsymbol{q}}]_\oslash = \begin{bmatrix} [\boldsymbol{q}_1]_\oslash & 0_{4\times 4} \\ [\boldsymbol{q}_2]_\oslash & [\boldsymbol{q}_1]_\oslash \end{bmatrix}, \quad [\boldsymbol{q}]_\oslash = \begin{bmatrix} \boldsymbol{q}_v^\times + q_0 I_3 & \boldsymbol{q}_v \\ 0_{1\times 3} & 0 \end{bmatrix}. \quad (4)$$

If we consider the vector parts of either dual quaternion matrix in (3) or (4), this amounts to deleting the fourth and eighth rows and columns, and is written as $[\tilde{\boldsymbol{q}}]_{\otimes, v}$.

Finally, we define the dual quaternion conjugate as $\tilde{\boldsymbol{q}}^* = \boldsymbol{q}_1^* + \epsilon \boldsymbol{q}_2^*$, where $\boldsymbol{q}^* = [-\boldsymbol{q}_v^T\ q_0]^T$ is the quaternion conjugate. For both unit quaternions and unit dual quaternions, the conjugate is equal to the quaternion inverse. Therefore we may define the *error quaternion*, (resp. error dual quaternion) as

$$\boldsymbol{q}_e = \boldsymbol{q}^* \otimes \boldsymbol{p} \quad \text{and} \quad \tilde{\boldsymbol{q}}_e = \tilde{\boldsymbol{q}}^* \otimes \tilde{\boldsymbol{p}}. \quad (5)$$

D. Rigid Body Motion

Let $\boldsymbol{q} \in \mathbb{Q}_u$ be the unit quaternion representing the attitude of body-fixed frame $\mathcal{F}_b$ relative to the inertial frame $\mathcal{F}_I$, and $\boldsymbol{r}^I$ be the relative position of the center of $\mathcal{F}_b$ with respect to $\mathcal{F}_I$, resolved in the frame $\mathcal{F}_I$. Then the dual quaternion representing the translation $\boldsymbol{r}^I$ followed by rotation $\boldsymbol{q}$ is

$$\tilde{\boldsymbol{q}} = \left(\boldsymbol{q}_I + \frac{\epsilon}{2}\boldsymbol{r}^I\right) \otimes (\boldsymbol{q} + \epsilon 0) = \boldsymbol{q} + \frac{\epsilon}{2}\boldsymbol{r}^I \otimes \boldsymbol{q}, \quad (6)$$

where $\boldsymbol{r}^I$ is understood to be a *pure* quaternion (zero scalar part) [20]. Equally valid is a rotation followed by a translation. A similar derivation assuming this progression yields $\tilde{\boldsymbol{q}} = \boldsymbol{q} + \frac{\epsilon}{2}\boldsymbol{q} \otimes \boldsymbol{r}^b$, where $\boldsymbol{r}^b$ is the translation vector resolved in the frame $\mathcal{F}_b$.

Similarly, we encode the velocity components using dual quaternions as follows. Let $\boldsymbol{\omega}, \boldsymbol{v} \in \mathbb{R}^3$ define the angular and linear velocity of a rigid body, respectively. These are understood to be resolved in the body frame, $\mathcal{F}_b$. If $\boldsymbol{\omega}^b$ and $\boldsymbol{v}^b$ are represented as pure quaternions, then the dual velocity is

$$\tilde{\boldsymbol{\omega}}^b = \boldsymbol{\omega}^b + \epsilon \boldsymbol{v}^b \in \mathbb{Q}_v^2. \quad (7)$$

Note that $\boldsymbol{v}^b = \dot{\boldsymbol{r}}^b + {\boldsymbol{\omega}^b}^\times \boldsymbol{r}^b$ in accordance with the Transport theorem. Note also that there is no requirement that this is a unit dual quaternion; since both $\boldsymbol{\omega}^b$ and $\boldsymbol{v}^b$ are pure quaternions, there are only six free parameters.

Kinematics: The dual quaternion kinematics are simple to derive, and can be found by directly computing a time derivative of (6). For $\tilde{\boldsymbol{q}} \in \mathbb{Q}_u^2$ this gives

$$\dot{\tilde{\boldsymbol{q}}} = \frac{1}{2}\tilde{\boldsymbol{q}} \otimes \tilde{\boldsymbol{\omega}}, \quad (8)$$

which nicely resembles the quaternion kinematics.

Dynamics: Let $J \in \mathbb{S}_{++}^3$ denote the positive definite inertia matrix of the rigid body, and $m \in \mathbb{R}_+$ its mass. We will assume that both of these quantities are constant. Using Newton's second law and Euler's equations, we can write the dual quaternion dynamics as

$$\boldsymbol{J}\dot{\tilde{\boldsymbol{\omega}}} + \tilde{\boldsymbol{\omega}} \oslash \boldsymbol{J}\tilde{\boldsymbol{\omega}} = \tilde{\boldsymbol{u}}, \quad (9)$$

where,

$$\boldsymbol{J} = \begin{bmatrix} \begin{matrix} \\ 0_{4\times 4} \end{matrix} & \begin{matrix} mI_3 & 0_{3\times 1} \\ 0_{1\times 3} & 1 \end{matrix} \\ \begin{matrix} J & 0_{3\times 1} \\ 0_{1\times 3} & 1 \end{matrix} & \begin{matrix} \\ 0_{4\times 4} \end{matrix} \end{bmatrix}, \quad \tilde{\boldsymbol{u}} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{\tau} \end{bmatrix},$$

where $\boldsymbol{f}, \boldsymbol{\tau} \in \mathbb{Q}_v$ are the externally applied forces and torques, respectively. We assume that these systems are fully actuated and have access to full state information. See [8] for more details on the dual quaternion kinematic and dynamic model.

III. Main Results

Problem Statement: Given n aerospace vehicles, whose motion is governed by (8)-(9), design control signals $\{\tilde{\boldsymbol{u}}_i\}_{i=1}^n$ such that all vehicles are stabilized about a desired position and attitude.

We now provide two technical lemmas on which solutions to this problem may be constructed.

Lemma 3.1: Let $(\tilde{\boldsymbol{q}}_i, \tilde{\boldsymbol{\omega}}_i^i)$ and $(\tilde{\boldsymbol{q}}_j, \tilde{\boldsymbol{\omega}}_j^j)$ both satisfy the dual quaternion kinematics and describe the rotation and translation of agent i and j, respectively. Then $(\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i, \tilde{\boldsymbol{\omega}}_i^i - \tilde{\boldsymbol{\omega}}_j^i)$ also satisfies the dual quaternion kinematics and if

$$V_{\tilde{\boldsymbol{q}}} = \|\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i - \tilde{\boldsymbol{q}}_I\|^2, \quad (10)$$

then

$$\dot{V}_{\tilde{\boldsymbol{q}}} = (\tilde{\boldsymbol{\omega}}_i^i - \tilde{\boldsymbol{\omega}}_j^i)^T \left[\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i - \tilde{\boldsymbol{q}}_I\right]_v, \quad (11)$$

where $\tilde{\boldsymbol{q}}_I = [\boldsymbol{q}_I^T\ 0_{1\times 4}]^T$ is the identity dual quaternion.

Proof: The proof follows from an application of Lemma 3.1 from [2] that in turn uses a result from [17]. ∎

The single caveat in the proof above is that the reference frame in which the dual velocities are resolved must be kept track of. Angular velocities are additive, and hence no such care was needed in [2]. However, the dual part of the dual velocity is a linear velocity, which is not additive between reference frames. Hence the frame in which $\tilde{\boldsymbol{\omega}}_j$ is resolved is important here.

Lemma 3.2: Assume that a communication network between vehicles is represented by the graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ and that $\mathcal{G}$ is undirected and connected. Suppose the control force for the i^{th} agent is given by

$$\tilde{\boldsymbol{u}}_i = -k_P \tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i - k_D \tilde{\boldsymbol{\omega}}_i^i - \sum_{j=1}^{n} a_{ij}\left(k_{ij}\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i + b_{ij}(\tilde{\boldsymbol{\omega}}_i^i - \tilde{\boldsymbol{\omega}}_j^i)\right), \quad (12)$$

where $k_D \succ 0$, $K = [k_{ij}] \geq 0$ is symmetric and

$$k_P > \sum_{j=1}^{n} a_{ij}k_{ij}(r_{\max} + 1) \quad (13)$$

for each $i = 1, \ldots, n$, with $r_{\max}$ as a global upper bound on the relative positions of the agents. Then we have

$$\tilde{\boldsymbol{q}}_i \to \tilde{\boldsymbol{q}}_j \to \tilde{\boldsymbol{q}}_d, \quad \text{and} \quad \tilde{\boldsymbol{\omega}}_i \to \tilde{\boldsymbol{\omega}}_j \to 0,$$

asymptotically.

Proof: Note that in (12) we only consider the vector part of the right hand side, since $\tilde{\boldsymbol{u}} \in \mathbb{Q}_v^2$. In places that this distinction is key to the proof, it will be explicitly stated. The result will be proved using a LaSalle's invariance principal type argument. Consider the Lyapunov function candidate

$$V = k_P \sum_{i=1}^{n} \|\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i - \tilde{\boldsymbol{q}}_I\|^2 + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}k_{ij}\|\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i - \tilde{\boldsymbol{q}}_I\|^2 + \frac{1}{2}\sum_{i=1}^{n} \tilde{\boldsymbol{\omega}}_i^i \oslash \boldsymbol{J}_i \tilde{\boldsymbol{\omega}}_i^i. \quad (14)$$

Using Lemma 3.1 and the dual quaternion dynamics (9) we can write

$$\begin{aligned}\dot{V} &= k_P \sum_{i=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_i^{\,T} (\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i) \\ &\quad + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}k_{ij}(\tilde{\boldsymbol{\omega}}_i^i - \tilde{\boldsymbol{\omega}}_j^i)^T \tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i \\ &\quad + \sum_{i=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_i^{\,T} \left(\tilde{\boldsymbol{u}}_i - \tilde{\boldsymbol{\omega}}_i^i \oslash \boldsymbol{J}_i \tilde{\boldsymbol{\omega}}_i^i\right).\end{aligned}$$

Let us examine the second term in more detail. Following [2], we have

$$\begin{aligned}&\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}k_{ij}(\tilde{\boldsymbol{\omega}}_i^i - \tilde{\boldsymbol{\omega}}_j^i)^T \tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i \\ &= \frac{1}{2}\sum_{i=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_i^{\,T} \sum_{j=1}^{n} a_{ij}k_{ij}\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}k_{ij}\,{}^i\tilde{\boldsymbol{\omega}}_j^{\,T} \tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i, \\ &= \frac{1}{2}\sum_{i=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_i^{\,T} \sum_{j=1}^{n} a_{ij}k_{ij}\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i + \frac{1}{2}\sum_{j=1}^{n}\sum_{i=1}^{n} a_{ji}k_{ji}\,{}^i\tilde{\boldsymbol{\omega}}_j^{\,T} \tilde{\boldsymbol{q}}_i^* \otimes \tilde{\boldsymbol{q}}_j, \\ &= \frac{1}{2}\sum_{i=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_i^{\,T} \sum_{j=1}^{n} a_{ij}k_{ij}\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i + \frac{1}{2}\sum_{j=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_j^{\,T} \sum_{i=1}^{n} a_{ij}k_{ij}\tilde{\boldsymbol{q}}_i^* \otimes \tilde{\boldsymbol{q}}_j\,, \\ &= \sum_{i=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_i^{\,T} \left(\sum_{j=1}^{n} a_{ij}k_{ij}\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i\right).\end{aligned}$$

Note that for the second equality we have used the fact that $a_{ij} = a_{ji}$ (the communication graph is undirected), $k_{ij} = k_{ji}$ and the *vector part* satisfies $\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i = -\tilde{\boldsymbol{q}}_i^* \otimes \tilde{\boldsymbol{q}}_j$. The subsequent steps are algebraic manipulations that lead to the final result. Returning to the derivative of the Lyapunov candidate function, we now have

$$\dot{V} = \sum_{i=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_i^{\,T} \left[k_P \tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i - \sum_{j=1}^{n} a_{ij}k_{ij}\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i + \tilde{\boldsymbol{u}}_i\right].$$

Using the proposed control law (12),

$$\begin{aligned}\dot{V} &= \sum_{i=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_i^{\,T} \left[-k_D \tilde{\boldsymbol{\omega}}_i^i - \sum_{j=1}^{n} a_{ij}k_{ij}(\tilde{\boldsymbol{\omega}}_i^i - \tilde{\boldsymbol{\omega}}_j^i)\right], \\ &= -\sum_{i=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_i^{\,T} k_D \tilde{\boldsymbol{\omega}}_i^i - \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}k_{ij}\,{}^i\tilde{\boldsymbol{\omega}}_i^{\,T}(\tilde{\boldsymbol{\omega}}_i^i - \tilde{\boldsymbol{\omega}}_j^i), \\ &= -\sum_{i=1}^{n} {}^i\tilde{\boldsymbol{\omega}}_i^{\,T} k_D \tilde{\boldsymbol{\omega}}_i^i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}k_{ij}\|\tilde{\boldsymbol{\omega}}_i^i - \tilde{\boldsymbol{\omega}}_j^i\|^2,\end{aligned}$$

which can be obtained using the same trick from above in reverse order. Since we have assumed $k_D \succ 0$ and $k_{ij} \geq 0$, we have $\dot{V} \leq 0$ along trajectories of the system.

Now, let $E = \{\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i - \tilde{\boldsymbol{q}}_I, \tilde{\boldsymbol{\omega}}_i^i \mid \dot{V} = 0,\ i = 1, \ldots, n\}$; on E, we must have $\tilde{\boldsymbol{\omega}}_i^i = \tilde{\boldsymbol{\omega}}_j^i = 0$. For trajectories contained in E, we have $\dot{\tilde{\boldsymbol{\omega}}}_i^i = 0$ for each $i = 1, \ldots, n$. Considering only the vector portion of the control law (12), and combining this with the system dynamics (9), we can write

$$\begin{aligned}0 &= k_P[\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i]_v + \sum_{j=1}^{n} a_{ij}k_{ij}[\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_i]_v, \\ &= k_P[\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i]_v \sum_{j=1}^{n} a_{ij}k_{ij}[\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_d \otimes \tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i]_v, \\ &= k_P[\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i]_v + \sum_{j=1}^{n} a_{ij}k_{ij}([\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_d]_{\otimes,v}[\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i]_v \\ &\quad + [\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i]_s \otimes [\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_d]_v), \\ &= -\sum_{j=1}^{n} a_{ij}k_{ij}[[\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i]_s]_{\otimes,v} \otimes [\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_j]_v \\ &\quad + \left(k_P I + \sum_{j=1}^{n} a_{ij}k_{ij}[\tilde{\boldsymbol{q}}_j^* \otimes \tilde{\boldsymbol{q}}_d]_{\otimes,v}\right)[\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i]_v,\end{aligned}$$

which must hold for all $i = 1, \dots, n$. Motivated by [2] we can then write this as $0 = (P(t) \otimes_K I_n)Q_e$, where $\otimes_K$ is the Kronecker product, $Q_e \in \mathbb{R}^{6n}$ is a vector stack of the dual error quaternions $\tilde{\boldsymbol{e}}_i = [\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i]_v$ for each agent, and the matrix $P \in \mathbb{R}^{6\times 6}$ can be written by $P_{ll} = k_P + \sum_{j=1}^{n} a_{lj} k_{lj} \boldsymbol{e}_{j,0}$ and $P_{lm} = -a_{lm} k_{lm} \boldsymbol{e}_{l,s}$ if $l = 4, 5, 6$ and $m = 1, 2, 3$, and $P_{lm} = -a_{lm} k_{lm} \boldsymbol{e}_{l,0}$ otherwise. Here, $\boldsymbol{e}_{i,0}$ is the scalar entry in the attitude error quaternion for the ith agent, and therefore lies between $[-1, 1]$. Further, $\boldsymbol{e}_{i,s}$ is the scalar entry in the dual part of the dual error quaternion.

While we can bound the real (attitude) part, the dual part depends on the position of the ith agent, and cannot be bounded in the same way as the real part. However, if we know that $r_{\max}$ is an upper bound on the relative positions of all agents, then P is strictly diagonally dominant if and only if $k_P > \sum_{j=1}^{n} a_{ij} k_{ij}(r_{\max} + 1)$. Since our control gains are chosen to satisfy this condition, we conclude that $Q_e = 0$ since P is full rank and therefore has a trivial null space. Hence $\tilde{\boldsymbol{q}}_i = \tilde{\boldsymbol{q}}_d$ for all $i = 1, \dots, n$ and the largest invariant set contained in E is $(0, 0)$. By LaSalle's theorem we have that $\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i - \tilde{\boldsymbol{q}}_I \to 0$ and $\tilde{\boldsymbol{\omega}}_i \to 0$, for all $i = 1, \dots, n$ as $t \to \infty$. ■

By applying Lemma 3.2 directly, we are able to achieve a regulating controller; one that drives the dual velocity of each agent to zero, while ensuring convergence to a common desired attitude and position. This in itself is quite powerful, since it unifies two commonly separate control goals. By blending the control of position and attitude, we can achieve maneuvers that leverage the coupling between these dynamic states. While Lemma 3.2 offers rich results, it is the following two results that may be of greater interest in practice.

Theorem 3.3: Assume that the communication network is represented by $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ and that $\mathcal{G}$ is undirected and connected. If the control torque for the i^{th} spacecraft is given by (12) with $\tilde{\boldsymbol{q}}_i$ replaced by $\tilde{\boldsymbol{q}}_i \otimes \delta\tilde{\boldsymbol{q}}_i$, and both $k_D \succ 0$ and (13) hold, then we have $\tilde{\boldsymbol{q}}_i \to \tilde{\boldsymbol{q}}_d \otimes \delta\tilde{\boldsymbol{q}}_i^*$ and $\tilde{\boldsymbol{\omega}}_i \to \tilde{\boldsymbol{\omega}}_j \to 0$, where $\delta\tilde{\boldsymbol{q}}_i$ defines an offset (in both attitude and position) from $\tilde{\boldsymbol{q}}_d$ for the i^{th} spacecraft.

Proof: The proof follows immediately by an application of Lemma 3.2 with $\tilde{\boldsymbol{q}}_i$ replaced by $\tilde{\boldsymbol{q}}_i \otimes \delta\tilde{\boldsymbol{q}}_i$ wherever it appears. ■

The difference between Theorem 3.3 and Lemma 3.2 is that the former causes convergence to a common attitude and position, whereas the latter allows for each agent to be at an offset from a common point. Since each offset is arbitrary and not linked between agents, this result is most useful for formation flying or aerial surveillance type maneuvers.

Theorem 3.4: Assume that the communication network is represented by $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ and that $\mathcal{G}$ is undirected and connected. Let $\tilde{\boldsymbol{\nu}}_i$ be the control given in (12) for the i^{th} spacecraft. If the applied control for the i^{th} spacecraft is

$$\tilde{\boldsymbol{u}}_i = \tilde{\boldsymbol{\nu}}_i + \tilde{\boldsymbol{\omega}}_i^i \oslash \boldsymbol{J}_i \tilde{\boldsymbol{\omega}}_i^i + \boldsymbol{J}_i \dot{\tilde{\boldsymbol{\omega}}}_d, \tag{15}$$

and both $k_D \succ 0$ and (13) are met, then we have

$$\tilde{\boldsymbol{q}}_i \to \tilde{\boldsymbol{q}}_j \to \tilde{\boldsymbol{q}}_d(t) \quad \text{and} \quad \tilde{\boldsymbol{\omega}}_i \to \tilde{\boldsymbol{\omega}}_j \to \tilde{\boldsymbol{\omega}}_d(t)$$

asymptotically, where $\tilde{\omega}_d(t)$ is a possibly time varying desired dual velocity.

Proof: The proof follows from an application of Lemma 3.2 with the following substitutions. Whenever they appear, $\tilde{\boldsymbol{q}}_i$ is replaced with $\tilde{\boldsymbol{q}}_d^* \otimes \tilde{\boldsymbol{q}}_i$ and $\tilde{\boldsymbol{\omega}}_i^i$ is replaced with $\tilde{\boldsymbol{\omega}}_i^i - \tilde{\boldsymbol{\omega}}_d^i$. Note that by Lemma 3.1 these substitutions obey the dual quaternion kinematics. The extra two terms present in (15) are a result of this substitution for the dual velocity. ■

It is worth mentioning that the control law for the regulation and offset cases, (12), is model independent in the sense that no knowledge of the inertia matrix, or mass, is required. We do not enjoy the same luxury in the case of the tracking control law (15). This result is consistent with previous findings, such as those in [2] and [17].

A. Numerical Example

In this section, we provide a numerical example that highlights the utility of the dual quaternion based control design presented in the previous section. To demonstrate the wider reaching applications of the proposed methods, this case considers a single agent landing on a moving platform. The platform can be modeled as an agent in the network provided a communication link exists between the platform and the agent (e.g., a beacon signal). Note that the problem setup is effectively a rendezvous-docking problem.

The landing spacecraft has mass 1kg and inertia matrix $J = \text{diag}(1, 1, 2)$ kg m^2. The moving landing pad is modeled with mass 10 kg and inertia $J = \text{diag}(10, 10, 25)$ kg m^2.

The landing spacecraft uses the control law of Theorem 3.4 to track and land on the moving target, achieving a low velocity at touchdown. The information being passed to the landing agent may be in the form of optically sensed information aboard the landing agent, or as a beacon signal emitted from the platform. We use a single "stubborn" agent to model a moving landing platform; it ignores the information received from the landing agent, and proceeds at a constant dual velocity. For this example, the landing pad is travelling at a velocity $\boldsymbol{v}(0) = [-0.1\ 0.5\ 0]^T$m/s from the initial position $\boldsymbol{r}(0) = [0\ 0\ 0.5]^T$m in an inertial frame of reference with zero angular velocity. For this example both the initial dual quaternion and dual velocity of the landing spacecraft were chosen at random. The control gains are $k = 5 + \epsilon 5$, $b = 2 + \epsilon 2$, $k_P = 5 + \epsilon 5$ and $k_D = 5I_3 + \epsilon I_3$.

The resulting trajectories are presented in Figures 1 and 2.

IV. CONCLUSIONS AND FUTURE WORK

We have presented new results for the coupled attitude and position control of rigid bodies. Similar control laws can be applied for many different scenarios; including the cases of regulation to, offset from and tracking of reference signals. In the example, a moving landing pad was modeled using the dual quaternion equations (8)–(9). A second spacecraft used the control law from Theorem 3.4 to track and land on the moving pad. In all cases (regulation, offset, tracking), the agents perform maneuvers that would not otherwise be observed if separate attitude and position controllers were used. The phenomenon of screw motion is characteristic

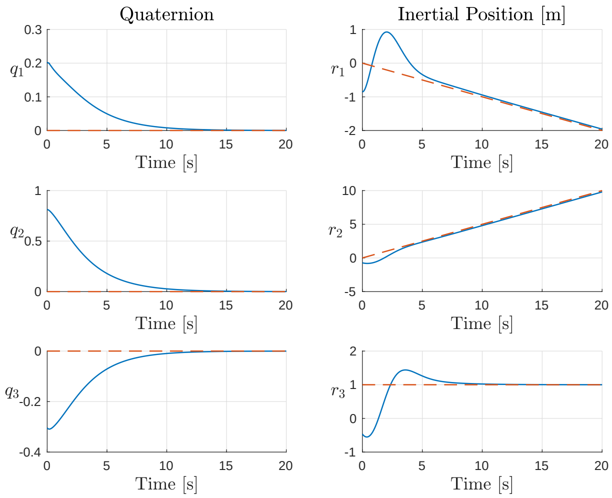

Fig. 1. Quaternion vector part and inertial position values for spacecraft and pad during a landing maneuver. The solid blue line represents the landing spacecraft.

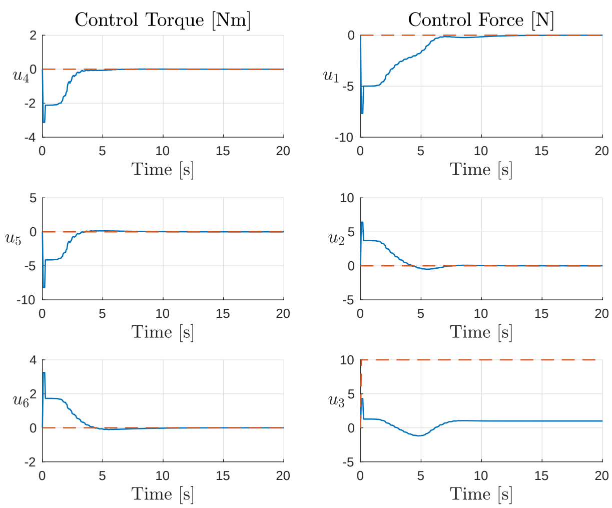

Fig. 2. Torque and force values for spacecraft and pad during a landing maneuver. The solid blue line represents the landing spacecraft. The constant offset in u_3 is the landing spacecraft accounting for gravity.

of dual quaternions and is especially observed using the controllers presented here [18], [19]. Screw motion can be thought of as a simultaneous twist and translation (like the threads of a screw). As a result, rigid bodies are seen to translate and rotate at the same time, but in a coordinated way that ensures cohesive movement. It is clear that dual quaternions offer a fresh look at the control of rigid body systems, and open the door to new possibilities for motion planning and feedback control of both networked multi-agent systems and single agent maneuvers.

There are several avenues to explore as extensions of this work. For example, there is a rather intriguing tendency for formation flying maneuvers to be naturally collision free. Even if deeper analysis reveals that collision avoidance is not guaranteed, adjustments to the control laws could readily be made to ensure collision avoidance. Either repulsive terms or the notion of graph rigidity can be exploited to this end. Time varying communication graphs are another natural extension. This is useful for practical purposes, since during maneuvers it is possible that previously connected spacecraft are no longer in communication range, or previously unconnected spacecraft may establish communication.

References

[1] Mesbahi, M. and Egerstedt, M., *Graph Theoretic Methods in Multiagent Networks*, Princeton Univeristy Press, Princeton, NJ; 2010.

[2] Ren, W., "Distributed attitude alignment in spacecraft formation flying," *International Journal of Adaptive Control and Signal Processing*, vol. 21, 2007, pp 95–113.

[3] Dimarogonas, D.V., Tsiotras, P. and Kyriakopoulos, K.J., "Leader-follower cooperative attitude control of multiple rigid bodies," *Systems and Control Letters*, vol. 58, no. 6, 2009, pp 429–435.

[4] Mesbahi, M. and Hadaegh, F.Y., "Formation Flying Control of Multiple Spacecraft via Graphs, Matrix Inequalities, and Switching," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 2, 2001, pp 369–377.

[5] Dai, R. and Maximoff, J. and Mesbahi, M., "Formation of Connected Networks for Fractionated Spacecraft," *AIAA Guidance, Navigation, and Control Conference*, Minneapolis, MN, 2012, pp 5047–5061.

[6] Lee, U. and Mesbahi, M., "Feedback Control for Spacecraft Reorientation Under Attitude Constraints via Convex Potentials," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 4, 2014, pp 2578–2592.

[7] Ren, W. and Beard, R.W., "Decentralized Scheme for Spacecraft Formation Flying via the Virtual Structure Approach," *Journal of Guidance, Control, and Dynamics*, vol. 27, no. 1, 2004, 73–82.

[8] Lee, U. and Mesbahi, M., "Constrained Autonomous Precision Landing via Dual Quaternions and Model Predictive Control," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 2, 2016, pp 292–308.

[9] Wang, X., Yu, C. and Lin, Z. "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.

[10] Wang, J.Y., Liang, H.Z., Sun, W., Wu, S.N. and Zhang, S.J., "Relative motion coupled control based on dual quaternion," *Aerospace Science and Technology*, vol. 25, no.1 pp. 102–113, 2013.

[11] Filipe, N. and Tsiotras, P., "Adaptive position and attitude tracking controller for satellite proximity operations using dual quaternions," *Advances in the Astronautical Sciences*, vol. 150, no. 4, pp. 2313–2332, 2014.

[12] Wu, J., Han, D., Liu, K. and Xiang, J., "Nonlinear suboptimal synchronized control for relative position and relative attitude tracking of spacecraft formation flying," *Journal of the Franklin Institute*, vol. 352, no. 4, pp. 1495–1520, 2015.

[13] Filipe, N. and Tsiotras, P., "Rigid Body Motion Tracking Without Linear and Angular Velocity Feedback Using Dual Quaternions," *European Control Conference*, Zurich, Switzerland, 2013, pp. 329–334.

[14] Kristiansen, R., Nicklasson, P.J. and Gravdahl, J.T., "Spacecraft coordination control in 6DOF: Integrator backstepping vs passivity-based control," *Automatica*, vol. 44, no. 11, pp. 2896–2901, 2008.

[15] Dong, R., Geng, Z., Runsha, D. and Zhiyong, G., "Consensus Control for Dynamic Systems of Lie Groups," *32nd Chinese Control Conference*, pp 6856–6851, 2013.

[16] Lee, T., Leok, M., McClamroch, N.H. and Mar, O.C., "Geometric Tracking Control of a Quadcopter UAV on SE(3)," *IEEE Conference on Decision and Control*, Atlanta, GA, 2010, pp 5420–5425.

[17] Wen, J.T-Y., Kreutz-Delgado, K., The Attitude Control Problem, *IEEE Transactionas on Automatic Control,* vol. 36, no. 10, 1991, pp 1148–1162.

[18] Buchheim, A., "A Memoir on Biquaternions," *American Journal of Mathematics*, Vol. 7, No. 4, 1885, pp. 293–326.

[19] Clifford, W., "Preliminary Sketch of Bi-quaternions," *Proceedings of the London Mathematical Society*, 1873, pp. 381–395.

[20] Kenwright, B., "A Beginners Guide to Dual-Quaternions: What They Are, How They Work, and How to Use Them for 3D Character Hierarchies," *The 20th International Conference on Computer Graphics, Visualization and Computer Vision*, 2012, pp. 1–13.