AAS 14-XXX

DUAL QUATERNION BASED SPACECRAFT RENDEZVOUS WITH ROTATIONAL AND TRANSLATIONAL FIELD OF VIEW CONSTRAINTS

Unsik Lee* **and Mehran Mesbahi†**
University of Washington, Seattle, WA 98195-2400

In this paper, we address spacecraft rendezvous with a tumbling asteroid. In order for the spacecraft to know its exact position relative to the asteroid, a vision sensor system must maintain continuous exposure to the asteroid while the spacecraft maneuvers. Such constrained motion not only restricts the approach trajectory toward the asteroid but also impacts the spacecraft's attitude orientation during the maneuver. Together these restrictions constitute a challenging coupled rotationally and translationally constrained motion planning problem. In this direction, taking advantage of the unit dual quaternion parameterization which captures this combined motion simultaneously, we compactly represent the relative translational and rotational dynamics in deep space. We proceed to construct a convex representable subset to represent all feasible transitional and rotational motions. A convex quadratically constrained quadratic optimization program is formulated to find the control for the chasing spacecraft over this proposed convex subset based on the application of a piece-wise affine model predictive control framework. Brief stability analysis and simulation results for a typical constrained rendezvous mission are addressed to demonstrate the viability of the proposed methodology.

INTRODUCTION

The asteroid retrieval mission is the main objective of NASA's Asteroid Initiative and the next step in humanity's exploration of the solar system. Its main goal is to capture an entire near-Earth asteroid and return it to a cislunar orbit.[1] A novel methodology for proximity operations and rendezvous is one of the many obstacles for this mission. Modern autonomous rendezvous algorithms often lose accuracy when forced to rely on limited sensor-based estimation. This means the spacecraft's Guidance, Navigation, and Control (GN&C) subsystem is required to both rendezvous with the target and hold the target within the sensor's field of view. The coupled translational and rotational dynamics of the spacecraft make this a challenging problem for conventional autonomous rendezvous algorithms.

This challenge is illustrated by considering a spacecraft rendezvousing with an uncontrolled tumbling object in deep space. The spacecraft is required to approach the object through its principal rotation axis to simplify the docking mechanics. Furthermore, the spacecraft needs real time relative position and attitude information to correctly plan its approach. Thus the vision sensor should be continuously exposed to the target during the approach. The challenge here is that the exposure of the sensor is in fact affected not only by the rotational motion but also the translational motion of

*Postdoctoral Researcher, W. E. Boeing Department of Aeronautics & Astronautics, unsik@uw.edu
†Professor, W. E. Boeing Department of Aeronautics & Astronautics, mesbahi@uw.edu

the spacecraft. However the rotational and translational configuration space $\mathsf{SE}(3)$ is topologically complex, so attempting to satisfy our constraints by removing a subset from this space results in a non-convex path planning problem.

In this paper, unlike the conventional parameterization used in the literature,[15,16,17] we approach such constraints through the methodology of unit dual quaternions. Such a parameterization has two main advantages: it is the minimally globally nonsingular representation of $\mathsf{SE}(3)$ and it can represent the general motion of a rigid body without dependency on the underlying coordinate frame. The unit dual quaternion parameterization has been widely used in the field of manipulator control for the synthesis and analysis of coordination strategies of multiple rigid bodies.[8,10] Notable references on the unit dual quaternion include refs.[7,9,19] This parameterization enables us to develop rotationally and translationally constrained zones in a unified fashion and as we show subsequently that they can be formulated as a convex optimization problem based on the application of the piece-wise affine model predictive control (PAMPC) framework. PAMPC is a form of model predictive control (MPC) that makes explicit use of a successively sampled model of the system to obtain the control signal by minimizing an objective function over a finite receding horizon, subject to constraints. In PAMPC, the sampled system model is used to predict the future plant behaviors based on the current and previous information in terms of feedback. General studies on PAMPC can be found in Refs.[21,22,27]

The paper is organized as follows. First we present fundamentals on unit quaternions and unit dual quaternions. Then the rotationally and translationally constrained zones such as field of view constraints are presented as convex subsets in the unit dual quaternion parameterzation. Based on relative dynamics in unit dual quaternions, subsequently formulate the piece-wise affine model predictive control problem. We then discuss simulations based on feasible mission conditions. Finally in the last section we present our conclusions and plans for future work.

UNIT QUATERNIONS AND UNIT DUAL QUATERNIONS

In this section, we provide a brief background on unit quaternions and unit dual quaternions. More detailed discussion on these representations can be found in refs.[7,8,9]

Unit Quaternions

The attitude of a rigid body, describing the relative orientation between a reference frame and a body-fixed frame, evolves on the special orthogonal group $\mathsf{SO}(3)$. The unit quaternion is a minimal parameterization of $\mathsf{SO}(3)$ free from singularities. A unit quaternion, as the three dimensional extension of a unit complex number, parameterizes a rotating axis with three perpendicular imaginary axes i, j, and k. Let us present a 4×1 vector notation for unit quaternions as

$$\mathbf{q} = \left[\begin{array}{cccc} n_x \sin\frac{\theta}{2} & n_y \sin\frac{\theta}{2} & n_z \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{array} \right]^T, \tag{1}$$

where n_x, n_y and n_z denote three perpendicular components of the eigen-rotation axis $\mathbf{n}$ and θ denotes a rotation angle around this axis. Unit quaternions are closed for the following operations:

Quaternion Multiplication: Using the matrix notation, we can find its multiplicative product in the following form as

$$\mathbf{q} \otimes \mathbf{p} = \left[\begin{array}{c} q_0\boldsymbol{p} + p_0\boldsymbol{q} + \boldsymbol{q} \times \boldsymbol{p} \\ q_0 p_0 - \boldsymbol{q}^T\boldsymbol{p} \end{array} \right], \tag{2}$$

where $\mathbf{q} = [\boldsymbol{q}^T,\ q_0]^T$ and $\mathbf{p} = [\boldsymbol{p}^T,\ p_0]^T$. Another quaternion operation is the "unit quaternion conjugation" given as $\mathbf{q}^* = [-\boldsymbol{q}^T\ \ q_0\ ]^T$ in the vector notation, which facilitates the judicious definition of the "attitude difference/error" of $\mathbf{p}$ with respect to $\mathbf{q}$ as

$$\mathbf{q}_e \stackrel{\text{def}}{=} \mathbf{q}^* \otimes \mathbf{p} = \begin{bmatrix} q_0\boldsymbol{p} - p_0\boldsymbol{q} - \boldsymbol{q} \times \boldsymbol{p} \\ q_0 p_0 + \boldsymbol{q}^T\boldsymbol{p} \end{bmatrix}. \tag{3}$$

Note that the identity quaternion is expressed by $\mathbf{q_I} = [0\ 0\ 0\ 1]^T$. Quaternion multiplication is analogous in many ways to vector cross products. Thus, quaternion multiplication can be expressed as the product of a skew symmetric matrix and a quaternion:

$$\mathbf{q} \otimes \mathbf{p} = \begin{bmatrix} [\boldsymbol{q}]_\times + q_0\mathbf{I_3} & \boldsymbol{q} \\ -\boldsymbol{q}^T & q_0 \end{bmatrix} \begin{bmatrix} \boldsymbol{p} \\ p_0 \end{bmatrix} \stackrel{\text{def}}{=} [\mathbf{q}]_\otimes\, \mathbf{p} \tag{4}$$

$$= \begin{bmatrix} [\boldsymbol{p}]_\times^T + p_0\mathbf{I_3} & \boldsymbol{p} \\ -\boldsymbol{p}^T & p_0 \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ q_0 \end{bmatrix} \stackrel{\text{def}}{=} [\mathbf{p}]_\otimes^*\, \mathbf{q}, \tag{5}$$

where $[\mathbf{q}]_\otimes$ and $[\mathbf{p}]_\otimes^*$ denote 4×4 skew symmetric matrices related to "$\mathbf{q}$" and "$\mathbf{p}$", respectively; $[\boldsymbol{a}]_\times$ denotes a cross product operator associated with a vector $\boldsymbol{a}$ defined by

$$[\boldsymbol{a}]_\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}. \tag{6}$$

Additionally we define the following operation.

Quaternion Cross Products: We define a "relaxed" quaternion operation in order to facilitate a dimension reduction. For instance, the quaternion multiplication Eq. (2) can be relaxed to the newly defined *quaternion cross product* as

$$\mathbf{q} \oslash \mathbf{p} = \begin{bmatrix} q_0\boldsymbol{p} + p_0\boldsymbol{q} + \boldsymbol{q} \times \boldsymbol{p} \\ 0 \end{bmatrix} = \begin{bmatrix} [\boldsymbol{q}]_\times + q_0\mathbf{I}_3 & \boldsymbol{q} \\ 0_{1\times 3} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{p} \\ p_0 \end{bmatrix} \stackrel{\text{def}}{=} [\mathbf{q}]_\oslash \mathbf{p}, \tag{7}$$

$$= \begin{bmatrix} [\boldsymbol{p}]_\times^T + p_0\mathbf{I}_3 & \boldsymbol{p} \\ 0_{1\times 3} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ q_0 \end{bmatrix} \stackrel{\text{def}}{=} [\mathbf{p}]_\oslash^* \mathbf{q} \tag{8}$$

where $\mathbf{q}$ and $\mathbf{p}$ denote quaternions and $[\cdot]_\oslash$ denotes a 4×4 linear operator matrix corresponding to the *quaternion cross product*.

When quaternions are represented as elements in $\mathbb{R}^4$, their algebraic properties are extended with vector-based products such as the inner product. The following are notable extended algebraic properties of quaternions:

$$\mathbf{a} \otimes (\mathbf{b} + \mathbf{c}) = \mathbf{a} \otimes \mathbf{b} + \mathbf{a} \otimes \mathbf{c} \tag{9}$$

$$(\mathbf{a} \otimes \mathbf{b})^* = \mathbf{b}^* \otimes \mathbf{a}^* \tag{10}$$

$$(\gamma\mathbf{a}) \otimes \mathbf{b} = \mathbf{a} \otimes (\gamma\mathbf{b}) = \gamma(\mathbf{a} \otimes \mathbf{b}) \tag{11}$$

$$\mathbf{a} \otimes (\mathbf{b} \otimes \mathbf{c}) = (\mathbf{a} \otimes \mathbf{b}) \otimes \mathbf{c} \tag{12}$$

$$\mathbf{a}^T(\mathbf{b} \otimes \mathbf{c}) = c^T(\mathbf{b}^* \otimes \mathbf{a}) = \mathbf{b}^T(\mathbf{a} \otimes \mathbf{c}^*), \tag{13}$$

where $\gamma \in \mathbb{R}$. Note that quaternions $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^4$ and all properties are valid with unit quaternions since $\mathcal{S}^3 \subseteq \mathbb{R}^4$. In particular, Eq. (13) is an interesting new identity. Now, let us present another interesting algebraic property only applicable to unit quaternions. This identity is inspired by the Binet-Cauchy identity.[20]

Theorem 1 (Unit Quaternion Triple Identity)**.** *Assume that* $\mathbf{t}$ *and* $\mathbf{v}$ *are quaternions, and* $\mathbf{q}$ *is a unit quaternion. Then, for the vector inner product between two quaternion products, the following identity holds:*

$$(\mathbf{t} \otimes \mathbf{q})^T (\mathbf{y} \otimes \mathbf{q}) = (\mathbf{q} \otimes \mathbf{t})^T (\mathbf{q} \otimes \mathbf{y}) = \mathbf{t}^T \mathbf{y}, \tag{14}$$

where $\mathbf{t}, \mathbf{y} \in \mathbb{R}^4$ *and* $\mathbf{q} \in \mathcal{S}^3$.

Proof. The following proof utilizes the properties of quaternions. Using Eq. (13), we have

$$\begin{aligned}(\mathbf{t} \otimes \mathbf{q})^T (\mathbf{y} \otimes \mathbf{q}) &= \mathbf{y}^T ((\mathbf{t} \otimes \mathbf{q}) \otimes \mathbf{q}^*) && (15)\\ &= \mathbf{y}^T (\mathbf{t} \otimes (\mathbf{q} \otimes \mathbf{q}^*)) && (16)\\ &= \mathbf{y}^T \mathbf{t}, && (17)\end{aligned}$$

and

$$\begin{aligned}(\mathbf{q} \otimes \mathbf{t})^T (\mathbf{q} \otimes \mathbf{y}) &= \mathbf{y}^T (\mathbf{q}^* \otimes (\mathbf{q} \otimes \mathbf{t})) && (18)\\ &= \mathbf{y}^T (\mathbf{q}^* \otimes \mathbf{q}) \otimes \mathbf{t}) && (19)\\ &= \mathbf{y}^T \mathbf{t}, && (20)\end{aligned}$$

which concludes the proof. Note that this identity is still valid when $\mathbf{t}, \mathbf{y}$ are pure quaternions, namely, when $\mathbf{t} = [\, \boldsymbol{t}^T \; 0\,]$, $\mathbf{v} = [\, \boldsymbol{v}^T \; 0\,]$ with $\boldsymbol{t}, \boldsymbol{v} \in \mathbb{R}^3$. □

Unit Dual Quaternions

The concept of the "dual quaternion" was invented by W. Clifford to accommodate the additional information–relative position, encoded in a "regular" quaternion. The dual quaternion $\tilde{\mathbf{q}}$ is comprised of two quaternions: $\mathbf{q}_1$ and $\mathbf{q}_2$, denoting the real and dual parts of $\tilde{\mathbf{q}}$, respectively, combined as

$$\tilde{\mathbf{q}} = \mathbf{q}_1 + \epsilon\, \mathbf{q}_2, \tag{21}$$

where ϵ denotes the "dual unit" satisfying the "unconventional" properties

$$\epsilon^2 = 0 \quad \text{and} \quad \epsilon \neq 0. \tag{22}$$

Let us now present convenient notations for unit dual quaternions. Since all operations are linear, unit dual quaternions can be embedded in Euclidean space $\mathbb{R}^8$ as

$$\tilde{\mathbf{q}} = \mathbf{q}_1 + \epsilon\, \mathbf{q}_2, \text{ or } \tilde{\mathbf{q}} = \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \end{bmatrix}_{8\times 1}, \tag{23}$$

where $\mathbf{q}_1$ and $\mathbf{q}_2$ are quaternions. For a *unit dual quaternion*, which is a subset of dual quaternions, the real part $\mathbf{q}_1$ is restricted to be a unit norm vector $\|\mathbf{q}_1\|_2 = 1$.

The unit dual quaternion is closed under multiplication, given the property defined in Eq. (22). This linear operation is similarly defined as

$$\tilde{\mathbf{q}}\tilde{\otimes}\tilde{\mathbf{p}} = \mathbf{q}_1 \otimes \mathbf{p}_1 + \epsilon\left(\mathbf{q}_1 \otimes \mathbf{p}_2 + \mathbf{q}_2 \otimes \mathbf{p}_1\right) = \begin{bmatrix} \mathbf{q}_1 \otimes \mathbf{p}_1 \\ \mathbf{q}_1 \otimes \mathbf{p}_2 + \mathbf{q}_2 \otimes \mathbf{p}_1 \end{bmatrix}. \tag{24}$$

Note that since dual quaternion multiplication, denoted by $\tilde{\otimes}$, is a linear operation, we can represent it as a multiplication between a matrix and a vector (dual quaternion) in conformance with quaternion multiplication as described in Eq. (4). This is shown below:

$$\begin{aligned} \tilde{\mathbf{q}}\tilde{\otimes}\tilde{\mathbf{p}} &= \begin{bmatrix} [\mathbf{q}_1]_{\otimes} & 0_{4\times4} \\ [\mathbf{q}_2]_{\otimes} & [\mathbf{q}_1]_{\otimes} \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \end{bmatrix} \stackrel{\text{def}}{=} [\tilde{\mathbf{q}}]_{\tilde{\otimes}}\,\tilde{\mathbf{p}}, \\ &= \begin{bmatrix} [\mathbf{p}_1]^*_{\otimes} & 0_{4\times4} \\ [\mathbf{p}_2]^2_{\otimes} & [\mathbf{p}_1]^*_{\otimes} \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \end{bmatrix} \stackrel{\text{def}}{=} [\tilde{\mathbf{p}}]^*_{\tilde{\otimes}}\,\tilde{\mathbf{q}}, \end{aligned} \tag{25}$$

As an extension of the *quaternion cross product*, we also define

$$\begin{aligned} \tilde{\mathbf{q}}\tilde{\oslash}\tilde{\mathbf{p}} &= \begin{bmatrix} \mathbf{q}_1 \oslash \mathbf{p}_1 \\ \mathbf{q}_1 \oslash \mathbf{p}_2 + \mathbf{q}_2 \oslash \mathbf{p}_1 \end{bmatrix} = \begin{bmatrix} [\mathbf{q}_1]_{\oslash} & 0_{4\times4} \\ [\mathbf{q}_2]_{\oslash} & [\mathbf{q}_1]_{\oslash} \end{bmatrix} \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \end{bmatrix} \stackrel{\text{def}}{=} [\tilde{\mathbf{q}}]_{\tilde{\oslash}}\,\tilde{\mathbf{p}} \\ &= \begin{bmatrix} [\mathbf{p}_1]^*_{\oslash} & 0_{4\times4} \\ [\mathbf{p}_2]^*_{\oslash} & [\mathbf{p}_1]^*_{\oslash} \end{bmatrix} \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \end{bmatrix} \stackrel{\text{def}}{=} [\tilde{\mathbf{p}}]^*_{\tilde{\oslash}}\,\tilde{\mathbf{q}} \end{aligned} \tag{26}$$

When the dual quaternion has a nonzero real part, its inverse can be obtained as $\tilde{\mathbf{q}}^{-1} = \|\tilde{\mathbf{q}}\|^{-1}\tilde{\mathbf{q}}^*$; on the other hand, its conjugate is defined as

$$\tilde{\mathbf{q}}^* = \begin{bmatrix} \mathbf{q}_1^* \\ \mathbf{q}_2^* \end{bmatrix}. \tag{27}$$

The norm of a dual quaternion is defined as $\|\tilde{\mathbf{q}}\|_{dq} = \tilde{\mathbf{q}}^* \otimes \tilde{\mathbf{q}} = \tilde{\mathbf{q}} \otimes \tilde{\mathbf{q}}^*$. In the case of unit dual quaternions, the norm of the dual quaternion yields

$$\|\tilde{\mathbf{q}}\|_{dq} = \begin{bmatrix} \mathbf{q_I} \\ 0_{4\times1} \end{bmatrix}, \tag{28}$$

where $\mathbf{q_I}$ denotes an identity unit quaternion. Note that the norm of a dual quaternion in this sense is a dual quaternion, as distinct from the case of quaternions (the norm of a quaternion is a scalar).

Unit Dual Quaternions and $\mathsf{SE}(3)$

Suppose that the relationship between the inertially fixed frame $\mathcal{O}$ and the body frame $\mathcal{B}$ is as shown in Fig. 1. At each instance, the configuration space for position and orientation of the rigid body is described by a 4×4 homogeneous transformation matrix. $\mathsf{SE}(3)$ is the set of all rigid body transformations in three dimensional space:

$$\mathsf{SE}(3) = \left\{ T \in \mathbb{R}^{4\times4} \,|\, T = \begin{bmatrix} R & t_o \\ 0 & 1 \end{bmatrix},\, R \in \mathsf{SO}(3),\, t_o \in \mathbb{R}^3 \right\}, \tag{29}$$

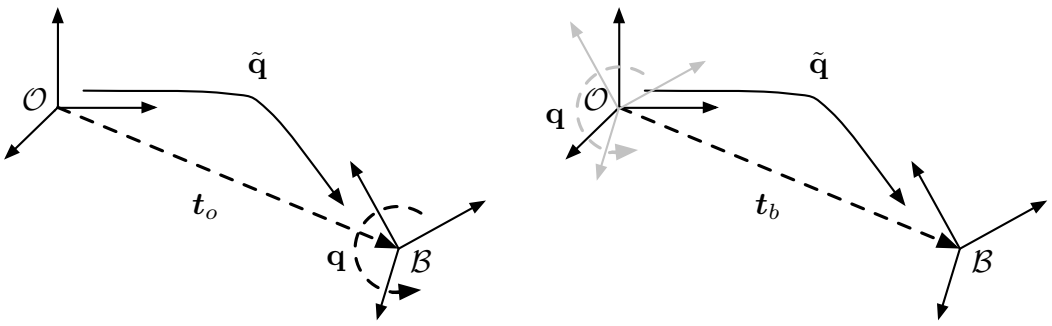

Figure 1. The geometric difference between the frame $\mathcal{B}$ and $\mathcal{O}$ can be expressed in two ways: a translation t_o followed by a rotation q [left figure] and a rotation q followed by a translation t_b [right figure], which are represented by the unit dual quaternion $\tilde{\mathrm{q}}$

where t_o denotes a position vector to the body frame $\mathcal{B}$ with inertial frame components. As unit quaternions parameterize $\mathsf{SO}(3)$, a unit dual quaternion can be used to define a rigid body rotation ($\tilde{\mathbf{q}} \in \mathbf{S}^3 \times \mathbb{R}^3 \mapsto \mathsf{SE}(3)$). The geometric difference between the frame $\mathcal{B}$ with respect to the inertially fixed frame $\mathcal{O}$ can be expressed in two ways. A translation $\boldsymbol{t}_o$ (represented in frame $\mathcal{O}$) followed by a rotation $\mathbf{q}_1$ and a rotation $\mathbf{q}_1$ followed by a translation $\boldsymbol{t}_b$ (represented in frame $\mathcal{B}$), which are represented in the form of *unit dual quaternions* as

$$\tilde{\mathbf{q}} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{q} \\ \frac{1}{2}\boldsymbol{t}_o \otimes \mathbf{q} \end{bmatrix} = \begin{bmatrix} \mathbf{q} \\ \frac{1}{2}\mathbf{q} \otimes \boldsymbol{t}_b \end{bmatrix}, \tag{30}$$

where $\boldsymbol{t}_o$ and $\boldsymbol{t}_b$ represent the translation vector $\boldsymbol{t}$ with respect to the frames $\mathcal{O}$ and $\mathcal{B}$, respectively. Note that it is easy to check that Eq. (30) satisfies Eq. (28). Converting between $\boldsymbol{t}_o$ and $\boldsymbol{t}_b$ is governed by the quaternion rotation operator as follows:

$$\boldsymbol{t}_o = \mathbf{q} \otimes \boldsymbol{t}_b \otimes \mathbf{q}^*, \tag{31}$$

$$\boldsymbol{t}_b = \mathbf{q}^* \otimes \boldsymbol{t}_o \otimes \mathbf{q}. \tag{32}$$

Remark 1. When using position information extracted from the unit dual quaternion, it should be noted that $\boldsymbol{t}_b$ and $\boldsymbol{t}_o$ are represented in two different frames but the two vectors are rooted in the inertially fixed frame; see Fig. (1). As a result, the position vector to the inertially fixed frame $\mathcal{O}$ from the body-fixed frame $\mathcal{B}$ is actually given as $-\boldsymbol{t}_b$.

ROTATIONALLY AND TRANSLATIONALLY CONSTRAINED ZONES

In this section, we parameterize two constrained zones for rendezvous and proximity operations; field of view and glide slope.

First we assume that the target object is tumbling. This requires that the spacecraft's vision sensor continuously receives relative attitude and position information for the duration of the rendezvous maneuver. Using the unit dual quaternion parameterization, we can show the position and orientation dynamics satisfying the field of view and glide slope constraints are in fact convex sets.

Field of View Constraint

The field of view constraint can be defined as a cone around a boresight vector in the body frame $\mathcal{B}$ (see Fig. 2). We assume that the frame $\mathcal{O}$ lies on the asteroid. Consider that the body frame $\mathcal{B}$

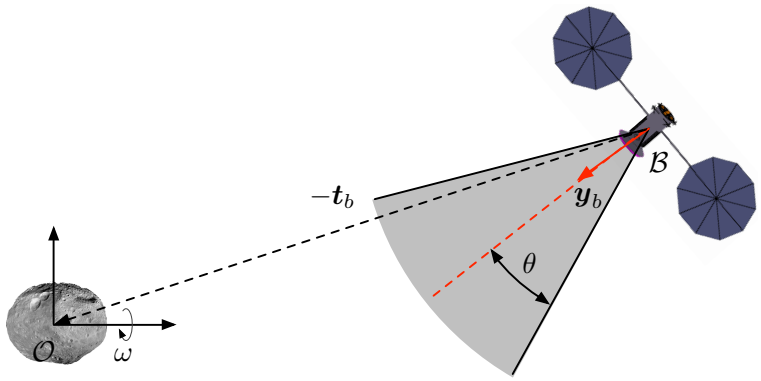

Figure 2. Illustration of rotationally and translationally constrained field of view constraint. It is defined as a cone around the fixed boresight vector y_b in the body frame.

is initially aligned to the frame $\mathcal{O}$, and subject to translation and rotation by $\tilde{\mathbf{q}}$. Thus, at a specific time, we can determine if the sensor's boresight vector $\boldsymbol{y}_b$, in the frame $\mathcal{B}$, stays within an angle θ around the direction to the target $\boldsymbol{t}_b$, in the frame $\mathcal{O}$, with the inner product

$$-\boldsymbol{t}_b \cdot \boldsymbol{y}_b \geq \|\boldsymbol{t}_b\| \cos\theta, \tag{33}$$

where $\boldsymbol{t}_b$ denotes a translation vector represented in the frame $\mathcal{B}$. Note that Eq. (33) holds for $-\pi \leq \theta \leq \pi$.

Then one can show that the left side of the above equation can be expressed by a quadratic function as

$$\begin{aligned}
\boldsymbol{t}_b \cdot \boldsymbol{y}_b &= (\mathbf{q}_1 \otimes \mathbf{t}_b)^T (\mathbf{q}_1 \otimes \mathbf{y}_b) && (34)\\
&= (\mathbf{q}_1 \otimes \mathbf{t}_b)^T [\mathbf{y}_b]^*_{\otimes}\, \mathbf{q}_1 && (35)\\
&= \begin{bmatrix} \mathbf{q}_1 \\ \frac{1}{2}\mathbf{q}_1 \otimes \boldsymbol{t}_b \end{bmatrix}^T \begin{bmatrix} 0_{4\times4} & [\mathbf{y}_b]^{*\,T}_{\otimes} \\ [\mathbf{y}_b]^*_{\otimes} & 0_{4\times4} \end{bmatrix} \begin{bmatrix} \mathbf{q}_1 \\ \frac{1}{2}\mathbf{q}_1 \otimes \boldsymbol{t}_b \end{bmatrix} \\
&= \tilde{\mathbf{q}}^T M_H \tilde{\mathbf{q}}, && (36)
\end{aligned}$$

where $\tilde{\mathbf{q}}$ denotes a unit dual quaternion and the quaternion properties Eqs. (5) and (14) have been applied. We note that M_H forms an indefinite symmetric matrix. In addition, inspired by Eq. (14), we can find the following useful identities:

$$\begin{aligned}
&\tilde{\mathbf{q}}^T \tilde{\mathbf{q}} = \|\tilde{\mathbf{q}}\|^2 = 1 + \frac{1}{4}\|\boldsymbol{t}_b\|^2, && (37)\\
&\tilde{\mathbf{q}}^T E_u \tilde{\mathbf{q}} = 1, \quad 4\,\tilde{\mathbf{q}}^T E_d \tilde{\mathbf{q}} = \|\boldsymbol{t}_b\|^2, \text{ and} && (38)\\
&\|\boldsymbol{t}_b\| = \|2\,E_d \tilde{\mathbf{q}}\| && (39)
\end{aligned}$$

where

$$E_u = \begin{bmatrix} \mathbf{I}_4 & 0_{4\times4} \\ 0_{4\times4} & 0_{4\times4} \end{bmatrix}, \quad E_d = \begin{bmatrix} 0_{4\times4} & 0_{4\times4} \\ 0_{4\times4} & \mathbf{I}_4 \end{bmatrix}. \tag{40}$$

Rewriting Eq. (33) with the above identities, we obtain

$$f_1(\tilde{\mathbf{q}}) = \tilde{\mathbf{q}}^T M_H \tilde{\mathbf{q}} + \|2\, E_d \tilde{\mathbf{q}}\| \cos\theta \leq 0, \tag{41}$$

where we note that the unit dual quaternion $\tilde{\mathbf{q}}$ is defined as an open set over $\mathbb{R}^3$ since $\boldsymbol{t}_b \in \mathbb{R}^3$. Consider a closed subset of $\mathbb{R}^3$ as

$$\|\boldsymbol{t}_b\| \leq \delta. \tag{42}$$

Correspondingly, we define the identical set in $\tilde{\mathbf{q}}$ as

$$\tilde{\mathbf{q}}^T \tilde{\mathbf{q}} \;\leq\; 1 + \frac{1}{4}\delta^2. \tag{43}$$

Then, one can show that the left side of Eq. (41) is convex over the defined closed subset.

Proposition 2. *Suppose the quadratic function $f : \tilde{\mathbf{q}} \to \mathbb{R}$ is given by*

$$f_1(\tilde{\mathbf{q}}) = \tilde{\mathbf{q}}^T M_H \tilde{\mathbf{q}} + \|2\, E_d \tilde{\mathbf{q}}\| \cos\theta \leq 0 \tag{41}$$

with

$$\mathbf{dom}\, f_1 = \left\{ \tilde{\mathbf{q}} \in (\mathbf{S}^3 \times \mathbb{R}^3) \mid \tilde{\mathbf{q}}^T \tilde{\mathbf{q}} \;\leq\; 1 + \frac{1}{4}\delta^2 \right\}, \tag{44}$$

where $\delta \in \mathbb{R}$, M_H and E_d are defined in Eqs. (36) and (40), respectively. Then f_1 is convex regardless of θ.

Proof. From the property of unit dual quaternions, Eq. (38), we have

$$\delta^2(\tilde{\mathbf{q}}^T E_u \tilde{\mathbf{q}}) - \delta^2 = 0. \tag{45}$$

Adding it into Eq. (41) yields

$$f_1 = \tilde{\mathbf{q}}^T M_H \tilde{\mathbf{q}} + \|2\, E_d \tilde{\mathbf{q}}\| \cos\theta + \delta^2(\tilde{\mathbf{q}}^T E_u \tilde{\mathbf{q}}) - \delta^2. \tag{46}$$

Note that f_1 is twice differentiable. Using the fact that

$$\tilde{\mathbf{q}}^T \left(\frac{d^2}{d\,\tilde{\mathbf{q}}^2} \|2 E_d \tilde{\mathbf{q}}\| \right) \tilde{\mathbf{q}} = 0, \tag{47}$$

the quadratic form of the *Hessian* of f_1 with $\tilde{\mathbf{q}}$ now has the simple form of

$$\tilde{\mathbf{q}}^T (\nabla^2 f_1) \tilde{\mathbf{q}} = 2(\tilde{\mathbf{q}}^T M_H \tilde{\mathbf{q}}) + 2\delta^2(\tilde{\mathbf{q}}^T E_u \tilde{\mathbf{q}}) \tag{48}$$

$$= 2(\boldsymbol{t}_b \cdot \boldsymbol{y}_b + \delta^2). \tag{49}$$

It is simple to show that $\boldsymbol{t}_b \cdot \boldsymbol{y}_b + \delta^2 \geq 0$ on $\mathbf{dom}\, f_1$. Since $\boldsymbol{y}_b$ is a unit vector, we know

$$-\|\boldsymbol{t}_b\|^2 \leq \boldsymbol{t}_b \cdot \boldsymbol{y}_b \leq \|\boldsymbol{t}_b\|^2. \tag{50}$$

Due to Eq. (42), we conclude that $\nabla^2 f_1(\tilde{\mathbf{q}})$ is positive semidefinite and thus, f_1 is convex on $\mathbf{dom}\, f$. □

Remark 2. Similarly, we can show that in the case where the sensor's boresight vector $\boldsymbol{y}_b$ stays *outside of* θ around the direction to the target $\boldsymbol{t}_b$ with

$$\boldsymbol{t}_b \cdot \boldsymbol{y}_b \leq \|\boldsymbol{t}_b\| \cos\theta \tag{51}$$

is convex on the same given domain. This can be done because in Eq. (44), we can find that $\tilde{\mathbf{q}}^T \tilde{\mathbf{q}}$ is naturally lower bounded by 1 such that $1 \leq \tilde{\mathbf{q}}^T \tilde{\mathbf{q}} \leq 1 + \delta^2/4$, which allows the *Hessian* to remain positive semidefinite on its domain.

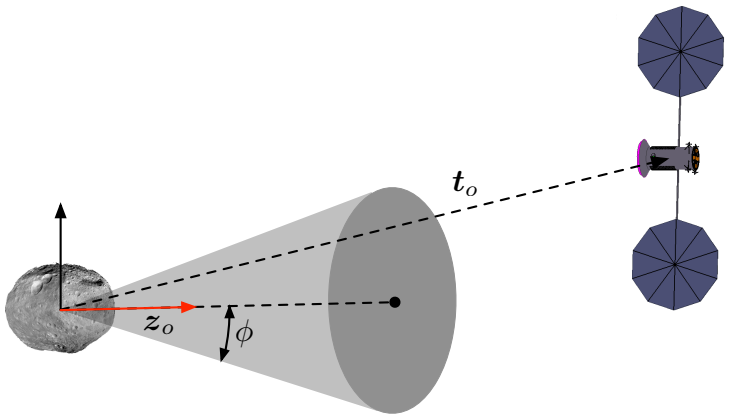

Figure 3. Illustration of the glide slope constraint. The constrained zone is defined as a cone around the inertially fixed vector z_o and angle ϕ.

Glide Slope Constraint

The glide slope constraint is defined as a cone around the fixed vector $\boldsymbol{z}_o$ that lies in the frame $\mathcal{O}$. See the illustration in Fig. 3. Note that the glide slope constraint only depends on the position of the frame $\mathcal{B}$. Similar to the line of sight constraint, we find the condition with which the glide slope constraint is satisfied as

$$\boldsymbol{t}_o \cdot \boldsymbol{z}_o \geq \|\boldsymbol{t}_o\| \cos\phi, \tag{52}$$

where $\boldsymbol{z}_o$ and $\boldsymbol{t}_o$ denote a unitized y axis vector and a position vector to frame $\mathcal{B}$ respectively. Note that they are represented with respect to frame $\mathcal{O}$. We reasonably assume that $0 < \phi \leq \frac{1}{2}\pi$.

Proposition 3. *The glide slope constraint can be represented in terms of the unit dual quaternion* $\tilde{\mathbf{q}}$ *as*

$$f_2(\tilde{\mathbf{q}}) = -\tilde{\mathbf{q}}^T M_G \tilde{\mathbf{q}} + \|2\, E_d \tilde{\mathbf{q}}\| \cos\phi \leq 0 \tag{53}$$

where

$$M_G = \begin{bmatrix} 0_{4\times4} & [\mathbf{z}_o]_\otimes{}^T \\ [\mathbf{z}_o]_\otimes & 0_{4\times4} \end{bmatrix}, \quad E_d = \begin{bmatrix} 0_{4\times4} & 0_{4\times4} \\ 0_{4\times4} & \mathbf{I}_4 \end{bmatrix} \tag{54}$$

with $\mathbf{z}_o = [\,\boldsymbol{z}_o^T \;\; 0\,]^T = [\,0 \;\; 0 \;\; 1 \;\; 0\,]^T$. *Then* $f_2 : \tilde{\mathbf{q}} \to \mathbb{R}$ *is convex on* $\mathbf{dom}\, f_2 = \{\tilde{\mathbf{q}} \in (\mathbf{S}^3 \times \mathbb{R}^3) \mid \tilde{\mathbf{q}}^T\tilde{\mathbf{q}} \leq 1 + \frac{1}{4}\delta^2\}$.

Proof. The left side of Eq. (52) is rewritten as follows

$$\begin{aligned} \boldsymbol{t}_o \cdot \boldsymbol{z}_o &= (\boldsymbol{t}_o \otimes \mathbf{q}_1)^T (\boldsymbol{z}_o \otimes \mathbf{q}_1) && (55)\\ &= (\boldsymbol{t}_o \otimes \mathbf{q}_1)^T [\mathbf{z}_o]_\otimes \mathbf{q}_1 && (56)\\ &= \begin{bmatrix} \mathbf{q}_1 \\ \frac{1}{2}\boldsymbol{t}_o \otimes \mathbf{q}_1 \end{bmatrix}^T \begin{bmatrix} 0_{4\times4} & [\mathbf{z}_o]_\otimes{}^T \\ [\mathbf{z}_o]_\otimes & 0_{4\times4} \end{bmatrix} \begin{bmatrix} \mathbf{q}_1 \\ \frac{1}{2}\boldsymbol{t}_o \otimes \mathbf{q}_1 \end{bmatrix} \\ &= \tilde{\mathbf{q}}^T M_G \tilde{\mathbf{q}}, && (57) \end{aligned}$$

where M_G denotes an indefinite symmetric matrix. Thus we obtain Eq. (53). The rest of the proof is analogous to the proof in *Proposition 2*. In $\mathbf{dom}\, f_2$, we can rewrite Eq. (53) as

$$f_2(\tilde{\mathbf{q}}) = \tilde{\mathbf{q}}^T M_G \tilde{\mathbf{q}} + \|2\, E_d \tilde{\mathbf{q}}\| \cos\phi + \delta^2 (\tilde{\mathbf{q}}^T E_u \tilde{\mathbf{q}}) - \delta^2. \tag{58}$$

The quadratic form of the *Hessian* of f_2 yields

$$\tilde{\mathbf{q}}^T (\nabla^2 f_2) \tilde{\mathbf{q}} = 2(-\boldsymbol{t}_o \cdot \boldsymbol{y}_o + \delta^2). \tag{59}$$

Since $\|t_o\| = \|t_b\|$, we have

$$-\boldsymbol{t}_o \cdot \boldsymbol{y}_o + \delta^2 \geq 0 \tag{60}$$

on $\mathbf{dom}\, f_2$ from Eq. (42). Note that $\boldsymbol{y}_o$ is a unit vector. Thus,

$$\nabla^2 f_2(\tilde{\mathbf{q}}) \succeq 0, \tag{61}$$

and $f_2(\tilde{\mathbf{q}})$ is convex on $\mathbf{dom}\, f_2$, which concludes the proof. □

Remark 3. The aforementioned constraint forms an analogous formulation in terms of unit dual quaternions. This is because a unit dual quaternion can be represented either with the inertial frame component or with the body frame component, and both yield the same quantity, as we observe from Eq. (30).

EQUATIONS OF MOTION IN UNIT DUAL QUATERNIONS

Dual Quaternion Kinematics

The unit dual quaternion kinematic equation[7] is then given as

$$\dot{\tilde{\mathbf{q}}} = \frac{1}{2} \tilde{\mathbf{q}} \tilde{\otimes} \tilde{\boldsymbol{\omega}} \quad \text{or} \quad \dot{\tilde{\mathbf{q}}} = \frac{1}{2} [\tilde{\boldsymbol{\omega}}]^*_{\tilde{\otimes}} \tilde{\mathbf{q}} \tag{62}$$

with

$$\tilde{\boldsymbol{\omega}} = \begin{bmatrix} \boldsymbol{\omega}_b \\ \boldsymbol{v}_b \end{bmatrix}_{8\times 1}, \tag{63}$$

where $[\tilde{\boldsymbol{\omega}}]^*_{\tilde{\otimes}} \in \mathbb{R}^{8\times 8}$ denotes a dual quaternion multiplication matrix defined in Eq. (25). $\boldsymbol{\omega}_b = [\omega_b^T\ 0]^T$ denotes the angular velocity of the rigid body and $\boldsymbol{v}_b = [v_b^T\ 0]^T$ denotes the translational velocity defined by $v_b = \dot{t}_b + \omega_b \times t_b$.

Dual Quaternion Dynamics

In this section, we consider rotational and translational dynamics in unit dual quaternions. As illustrated in Fig. (4), the target object is assumed to be positioned in the fixed frame $\mathcal{O}$ with fixed attitude. The relative translational motion of the spacecraft is modeled by a double integrator and it is given in body frame components by

$$F_b = \left[\frac{d}{dt}(mv)\right]_{\mathcal{B}} = m\dot{v}_b + \omega_b \times m v_b, \tag{64}$$

and the rotational motions for a fully actuated rigid body are expressed by Euler equations:[18]

$$T_b = \left[\frac{d}{dt}(J\omega)\right]_{\mathcal{B}} = J\dot{\omega}_b + \omega_b \times J\omega_b, \tag{65}$$

where $[d(\cdot)/dt]_{\mathcal{B}}$ denotes the time derivative represented in the body frame $\mathcal{B}$ and $m \in \mathbb{R}$ denotes the mass of the rigid body, $J \in \mathbb{R}^{3\times 3} = \texttt{diag}(J_1,\ J_2,\ J_3)$ denotes its inertia matrix along the body frame, and $\omega_b, v_b \in \mathbb{R}^3$ denote, respectively, the angular and translational velocities of the rigid body represented in the body frame. Moreover, F and T denote the external translational force and torque acting on the rigid body, written in the body frame. Note that in order to focus on the feasibility of the dual quaternion-based algorithm, we assume that uncertainties and all external disturbances on the rigid body are negligible.

Lemma 1. *Let $\boldsymbol{J}$ be a real block matrix of the form*

$$\boldsymbol{J} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \tag{66}$$

where A, B, C, D are $n \times n$ matrices and all commute with each other. Then, det $(\boldsymbol{J}) =$ det$(AD - BC)$.

Proof. See ref.[25] □

Proposition 4. *The rotational and translational motion dynamics Eqs. (65) and (64) can be represented in dual quaternions as*

$$\boldsymbol{J}\dot{\tilde{\boldsymbol{\omega}}} + \tilde{\boldsymbol{\omega}}\tilde{\oslash}\boldsymbol{J}\tilde{\boldsymbol{\omega}} = \tilde{\mathbf{F}} \tag{67}$$

where

$$\boldsymbol{J} = \left[\begin{array}{cc|cc} 0_{3\times 3} & 0 & m\mathbf{I}_3 & 0 \\ 0 & 0 & 0 & 1 \\ \hline J & 0 & 0_{3\times 3} & 0 \\ 0 & 1 & 0 & 0 \end{array}\right]_{8\times 8} \quad \textit{and} \quad \tilde{\mathbf{F}} = \begin{bmatrix} F \\ 0 \\ T \\ 0 \end{bmatrix}. \tag{68}$$

Moreover, $\boldsymbol{J} \in \mathbb{R}^{8\times 8}$ forms a block anti-diagonal matrix and is always invertible.

Proof. Rewriting Eq. (67) with Eqs. (63) and (68) yields

$$\begin{bmatrix} m\dot{\boldsymbol{v}}_b \\ J\dot{\boldsymbol{\omega}}_b \end{bmatrix} + \begin{bmatrix} \boldsymbol{\omega}_b \\ \boldsymbol{v}_b \end{bmatrix} \tilde{\oslash} \begin{bmatrix} m\boldsymbol{v}_b \\ J\boldsymbol{\omega}_b \end{bmatrix} = \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{T} \end{bmatrix} \tag{69}$$

where $\boldsymbol{F} = [\, F^T, 0\,]^T$ and $\boldsymbol{T} = [\, T^T, 0\,]^T$. Noting that

$$\boldsymbol{v}_b \oslash m\boldsymbol{v}_b = 0,$$

we have

$$\begin{bmatrix} m\dot{v}_b \\ J\dot{\omega}_b \end{bmatrix} + \begin{bmatrix} \omega_b \times m\boldsymbol{v}_b \\ \omega_b \times J\omega_b \end{bmatrix} = \begin{bmatrix} F \\ T \end{bmatrix}, \tag{70}$$

which is identical to Eqs. (64) and (65). Now, assume that $\boldsymbol{J}$ in Eq. (68) has the same form of matrix as Eq. (66). Since all 4×4 block matrices are commutative, from *Lemma 1* we have

$$\det(\boldsymbol{J}) = \det(-BC) \tag{71}$$
$$= \det(B)\,\det(C), \tag{72}$$

where B, C denote the right upper and left lower block matrices in $\boldsymbol{J}$ whose determinants are not zero. Thus, $\boldsymbol{J}$ is always invertible. □

Discretized Prediction Model

This section describes the discretization of the aforementioned nonlinear dynamic models written in dual quaternions for the development of the PAMPC controller. The continuous time differential equations Eqs. (62) and (67) are discretized in time with sampling period Δt. The equations are calculated as

$$\boldsymbol{J}\tilde{\boldsymbol{\omega}}(t+1) = \boldsymbol{J}\tilde{\boldsymbol{\omega}}(t) - \Delta t\,[\tilde{\boldsymbol{\omega}}(t)]_{\tilde{\oslash}}(\boldsymbol{J}\tilde{\boldsymbol{\omega}}(t)) + \Delta t\,\tilde{\mathbf{F}}(t) \tag{67d}$$
$$-\frac{\Delta t}{2}[\tilde{\mathbf{q}}(t)]_{\tilde{\otimes}}\tilde{\boldsymbol{\omega}}(t+1) + \tilde{\mathbf{q}}(t+1) = \tilde{\mathbf{q}}(t), \tag{62d}$$

where $\boldsymbol{J}$ denotes an inertia matrix in the body frame, $[\cdot]_{\tilde{\times}}$ and $[\cdot]_{\tilde{\otimes}}$ are defined in Eq. Note that we have assumed that the time sequence of the discretized equations is

$$\tilde{\mathbf{F}}(t) \implies \tilde{\boldsymbol{\omega}}(t+1) \implies \tilde{\mathbf{q}}(t+1). \tag{73}$$

By defining a new state variables as

$$x(t) = \begin{bmatrix} \tilde{\boldsymbol{\omega}}(t) \\ \tilde{\mathbf{q}}(t) \end{bmatrix}_{16\times 1} \quad \text{and} \quad u(t) = \tilde{\mathbf{F}}(t), \tag{74}$$

the set of equation yields

$$M(t)x(t+1) = A'(t)x(t) + B'u(t) \tag{75}$$

with

$$M(t) = \begin{bmatrix} \boldsymbol{J} & \mathbf{0}_{8\times 8} \\ -\frac{\Delta t}{2}[\tilde{\mathbf{q}}(t)]_{\tilde{\otimes}} & \mathbb{I}_8 \end{bmatrix}, \quad A'(t) = \begin{bmatrix} \boldsymbol{J} - \Delta t\,[\boldsymbol{J}\tilde{\boldsymbol{\omega}}(t)]^*_{\tilde{\oslash}} & \mathbf{0}_{8\times 8} \\ \mathbf{0}_{8\times 8} & \mathbb{I}_8 \end{bmatrix}, \quad B' = \begin{bmatrix} \Delta t\mathbb{I}_8 \\ \mathbf{0}_{8\times 8} \end{bmatrix}. \tag{76}$$

Then Eq. (75) can be formulated as a standard pice-wise affine system as

$$\begin{aligned} x(t+1) &= M(t)^{-1}A'(t)x(t) + M(t)^{-1}B'u(t) \\ &= A_t x(t) + B_t u(t) \end{aligned} \tag{77}$$

Note that M is always invertible by **Lemma 1** and the above system is stabilizable.

ADD here that a lemma ? shows the above system is controllable like

Proposition 5. *The above state space model sampled at t is stabilizable if M is ivertible and A and B is stabilizable.*

Proof. add proof here □

CONTROLLER DESIGN

In this section, based on the spacecraft dynamics and constraint parameterizations in unit dual quaternions presented in the previous section, we design a piece-wise affine model predictive controller and briefly present sufficient conditions for the system stability. The piece-wise affine model predictive control computes a sub-optimal trajectory to minimize a given cost function and satisfy constraints over a finite and receding horizon based on sampled linear system. Once an sub-optimal state and control trajectory has been carried out, the control corresponding to the first discrete time interval is implemented. The optimization horizon then recedes by one time step and the calculation process is repeated at each time step with the current state as the initial conditions. This loop generates a feedback action that can effectively compensate the uncertainties and disturbances. Furthermore, at each time step, constraints and even system model can be time-varying.

Piece-wise Affine Model Predictive Control

At each time instance t, given a sampled linear system, we consider a model predictive control to minimize the cost function $J_t : \mathbb{R}^n \times \mathbb{R}^{Nm} \to \mathbb{R}_+$ defined by

$$J_N = \sum_{k=1}^{N-1} \left(x_{(k|t)}^T Q x_{(k|t)} + u_{(k|t)}^T R u_{(k|t)} \right) + x_{(N|t)}^T P x_{(N|t)}, \tag{78}$$

where $x \in \mathbb{R}^n$ denotes the state vector, $u \in \mathbb{R}^m$ denotes the control input, $N \in \mathbb{Z}_+$ denotes the finite horizon and $Q, R, P \in \mathbb{S}_+^n$ denote weighting matrices on state, control input and final time state, respectively. Then given the state measurements at t, the MPC solves the following optimization problem:

$$\begin{aligned}
\min_{U_t} \quad & J_N(x_{(1|t)}, U_t) && (79)\\
\text{subject to} \quad & x_{(k+1|t)} = A_t x_{(k|t)} + B_t u_{(k|t)}, \quad k = 1, ..., N-1 && (80)\\
& x_{(k|t)} \in \mathcal{X}, \quad k = 1, ..., N && (81)\\
& u_{(k|t)} \in \mathcal{U}, \quad k = 1, ..., N-1 && (82)\\
& x_{(1|t)} = x(t), && (83)
\end{aligned}$$

where $x_{(1|t)} = x(t)$ denotes the first predicted state fed-back by a measurement at time instance t, $U_t = [u_{1,t}, ..., u_{N-1,t}]^T$ denotes optimal controls with the finite horizon N, and $X_t = [x_{1,t}, ..., x_{N,t}]^T$ denotes the corresponding state at time t by applying $U_t = [u_{1,t}, ..., u_{N-1,t}]^T$. $\mathcal{X}$ and $\mathcal{U}$ denote the given compact sets for states and control, respectively. The matrices A_t and B_t are assumed to bounded and sampled at each time instance t. Note that the cost function Eq. (78) is convex. When state constraints Eq. (81) and control constraints Eq. (82) are convex, the optimization problem is convex and can efficiently be solved by available convex programming solvers.[14]

Stability Analysis

In this section, we investigate the conditions for the uniformly asymptotical stability of the origin of the system with the presented PAMPC laws in the previous section. We focus on the stability conditions and assume feasibility of the MPC control laws. We denote by $J^*(X_t, t)$ the value of the cost function $J(X_t, U_t)$ in Eq. (78) at time instance t when optimal control laws U^* and initial

states at t have applied to Eq. (80). Then, the rate of change $\Delta J^*(X_t, t)$ of the function $J^*(X_t, t)$ along the solutions of the difference equations of the system is defined by

$$\Delta J^*(X_t, t) = J^*(X_{t+1}, t+1) - J^*(X_t, t), \tag{84}$$

where $X_t = [x_{1,t}, ..., x_{N,t}]^T$ denotes the state at time t corresponding to the optimal control U^*.

Consider a closed-loop piece-wise affine system of Eq. (80) as

$$x_{t+1} = f_{cl}(x_t, t), \quad x_t = x(t), \tag{85}$$

where $k \in \mathbb{Z}_+$, $x_k \in \mathbb{R}^n$, and $f : \mathbb{R}^n \times \mathbb{Z} \to \mathbb{R}^n$.

Theorem 6 (Lyapunov Uniform Asymptotic Stability for Discrete-Time Systems[24,26])**.** *If in a neighborhood domain $\mathcal{D}$ of the equilibrium state $x = 0$ there exists a Lyapunov function $J^*(x(t), t) : \mathbb{R}^n \times \mathbb{Z}_+ \to R_+$ such that*

1. *$J^*(X_t, t)$ is positive definite and decrescent.*
2. *The rate of change, $\Delta J^*(X_t, t)$ is negative definite in $\mathcal{D}$, then the equilibrium state is uniformly asymptotically stable.*

Proposition 7. *Consider the optimal cost function $J^*(x(t), t)$ at t from the optimization problem presented in the previous section. Then, such cost function fulfills the requirements in Theorem 5 to be a Lyapunov function and thus the origin of the closed-loop system Eq. (85) is uniformly asymptotically stable.*

Proof. First we show that $J^*(X_t, t)$ is decrescent. Consider $U_t^* = [u_{1,t}^*, ..., u_{N-1,t}^*]^T$ as the solution which minimizes the cost function of $J(X_t, t)$. Moreover, A_t and B_t in Eq. (80) are bounded. Then, under the assumption that $x_k \in \mathcal{X}$ and $u_k \in \mathcal{U}$ where $\mathcal{X}$,$\mathcal{U}$ are compact sets lead to

$$J_N^*(X_t, t) \leq \sum_{k=1}^{N-1} \left(\|Qx_{(k+1|t)}^*\|^2 + \|Ru_{(k+1|t)}^*\|^2 \right) + \|Px_{(N|t)}\|^2 \tag{86}$$

for all x_k and t. Therefore, $J^*(X_t, t)$ is decrescent. In order to find the condition such that the rate of change $\Delta J^*(X_t, t)$ is negative definite in $\mathcal{D}$, recall the discretized Hamilton-Jacobi-Bellman equation,[23]

$$J^{**}(X_t, U_t) = \min_{u_t} \left(x_t^T Q x_t + u_t^T R u_t + J^{**}(X_{t+1}, U_{t+1}) \right). \tag{87}$$

Then sub-optimal PAMPC control cost can be written as

$$\begin{aligned} J^*(X_t, U_t) &= \sum_{k=1}^{N-1} \left(x_k^T Q x_k + u_k^T R u_k \right) && (88) \\ &= x_{(1|t)}^T Q x_{(1|t)} + u_{(1|t)}^T R u_{(1|t)} + \sum_{k=2}^{N} \left(x_{(k|t)}^T Q x_{(k|t)} + u_{(k|t)}^T R u_{(k|t)} \right) \\ &\quad - x_{(N|t)}^T Q x_{(N|t)} - u_{(N|t)}^T R u_{(N|t)} && (89) \\ &\geq x_{(1|t)}^T Q x_{(1|t)} + u_{(1|t)}^T R u_{(1|t)} + J^*(X_{t+1}, U_{t+1}) \\ &\quad - x_{(N|t+1)}^T Q x_{(N|t+1)} - u_{(N|t+1)}^T R u_{(N|t+1)}. && (90) \end{aligned}$$

The rate of change $\Delta J^*(X_t, t)$ is bounded by

$$J^*(X_{t+1}, U_{t+1}) - J^*(X_t, U_t) \leq l_{(N|t+1)} - l_{(1|t)} \tag{91}$$

with

$$l_{(N|t+1)} = x^T_{(N|t+1)} Q x_{(N|t+1)} + u^T_{(N|t+1)} R u_{(N|t+1)} \tag{92}$$

$$l_{(1|t)} = x^T_{(1|t)} Q x_{(1|t)} + u^T_{(1|t)} R u_{(1|t)} \tag{93}$$

where $x_{(1|t)}$ denotes state at time t with the predicted state at $k = 1$ while $x_{(N|t+1)}$ denotes the state at time $t + 1$ with the predicted state at $k = N$. Then, from the second condition in **Theorem 5**, we obtain the following inequality for stability:

$$l_{(N|t+1)} \leq l_{(1|t)}. \tag{94}$$

Thus, as long as the PAMPC controller satisfies the above inequality, uniform asymptotic stability is assured.

□

Spacecraft Rendezvous with Rotational and Translational Constraints

In this section, we formulate the spacecraft rendezvous problem using the design technique described in the previous section. The optimization problem to solve PAMPC is defined as follows:

$$\min_{U_t} \quad J_N(x_{(1|t)}, U_t) = \sum_{k=1}^{N-1} \left(\|x_d - x_{(k+1|t)}\|^2 + \rho \|R u_{(k|t)}\|^2 \right) \tag{95}$$

$$\text{subject to} \quad x_{(k+1|t)} = A_t x_{(k|t)} + B_t u_{(k|t)}, \quad k = 1, ..., N-1 \tag{77}$$

$$x^T_{(k+1|t)} M'_H x_{(k+1|t)} + \|2\, E'_d x_{(k+1|t)}\| \cos\theta \leq 0, \tag{41}$$

$$- x^T_{(k+1|t)} M'_G x_{(k+1|t)} + \|2\, E'_d x_{(k+1|t)}\| \cos\phi \leq 0, \tag{53}$$

$$\|x_{(k+1|t)}\|^2 \leq \zeta_x \tag{96}$$

$$\|u_{(k|t)}\| \leq \zeta_u, \tag{97}$$

$$x_{(1|t)} = x(t), \tag{98}$$

with

$$x_{(k|t)} = \begin{bmatrix} \tilde{\boldsymbol{\omega}}_{(k|t)} \\ \tilde{\mathbf{q}}_{(k|t)} \end{bmatrix}_{16\times 1} \quad \text{and} \quad u_{(k|t)} = \tilde{\mathbf{F}}_{(k|t)}, \tag{99}$$

where N denotes the prediction and the control horizon, x_d denotes a desired position and attitude, ρ denotes a weight on the control and ζ_x, ζ_u denote state and control bounds respectively. Since two types of constraints are acting on the position, related matrices are re-defined by

$$M'_H = \begin{bmatrix} 0 & 0 \\ 0 & M_H \end{bmatrix}, \quad M'_G = \begin{bmatrix} 0 & 0 \\ 0 & M_G \end{bmatrix}, \quad E'_d = \begin{bmatrix} 0 & 0 \\ 0 & E_d \end{bmatrix}. \tag{100}$$

We denote $U^*_t = [u^*_{(1|t)}, ..., u^*_{(N-1|t)}]$ the sequence of optimal control laws over finite horizon with a predicted model. Then, the first control law $u^*_{(1|t)}$ is applied to the actual system. At the next time step $t + 1$, the optimization problem is solved again over a shifted finite horizon based on updated system model A_{t+1}, b_{t+1} and initial condition x_{t+1}.

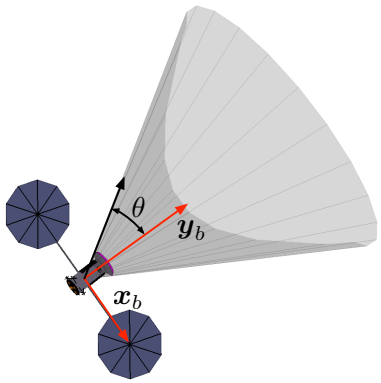

Figure 4. Spacecraft configuration with the field of view constraint $\boldsymbol{y}_b$

Δt	10 seconds
No. of iterations	400
Initial Attitude $\mathbf{q}(0)$	[0.2665, 0.3391, -0.4054, 0.8060]
Initial Position $\boldsymbol{t}_o(0)$	[380 820 -240] m
Initial Velocity $\boldsymbol{v}_b(0)$	[0.1, 0.2, 0] m/s
Initial Angular Velocity $\boldsymbol{\omega}_b(0)$	[0, 0, 0] rad/s
Desired Attitude $\mathbf{q}(t_f)$	[0.5, -0.5, 0.5, 0.5]
Desired Position $\mathbf{q}(t_f)$	[0, 50, 0] m
Desired Velocity $\boldsymbol{v}_b(t_f)$	[0, 0, 0] m/s
Desired Angular Velocity $\boldsymbol{\omega}_b(t_f)$	[0, 0, 0] rad/s

Table 1. Simulation parameters for a constrained proximity operation

SIMULATION RESULT

In this section, we present a simulation result to show the advantages and viability of the presented algorithm. The objective of the control is to rendezvous with a spinning object while staying inside of a glide slope and keeping the vision system continuously exposed to the object. The rendezvous direction is assumed to be along the z_o axis in the fixed frame attached on the object. All initial conditions including translational and angular velocities are randomly generated within reasonable bounds, whereas initial attitude and position are manually selected close to the boundaries of the glide slope and field of view constraints in order to check the validity of the proposed algorithm. The spacecraft's physical properties are given as

$$\text{Moment of inertia: } J = \texttt{diag}[\,3420, \quad 4160, \quad 2790\,]\, kg \cdot m^2, \tag{101}$$

$$\text{Mass: } m = 3270kg. \tag{102}$$

The illustration of a spacecraft with a field of view boresight vector is depicted in Fig. 4. The simulation initial conditions and desired conditions are given in Table 1 and detailed parameters of two types of constraints are presented in Table 2.

Based on the information from these tables, corresponding initial and desired conditions in unit

line of sight	$\boldsymbol{y}_b$ = [0, 1, 0],	$\theta = 50^\circ$
Glide slope	$\boldsymbol{z}_o$ = [0, 0, 1],	$\phi = 30^\circ$

Table 2. Parameters for Field of View and Glide Slope constraints

dual quaternions are calculated by

$$\tilde{\mathbf{q}}(0) = \begin{bmatrix} \mathbf{q}(0) \\ \frac{1}{2}\boldsymbol{t}_o(0) \otimes \mathbf{q}(0) \end{bmatrix} \tag{103}$$

$$= [0.2665,\ 0.3391,\ -0.4054,\ 0.8060,\ 0.2760,\ 3.7551,\ -1.4155,\ -2.3832]^T$$

$$\tilde{\mathbf{q}}(t_f) = [0.5000,\ -0.5000,\ 0.5000,\ 0.5000,\ 0.1250,\ 0.1250,\ -0.1250,\ 0.1250]^T, \tag{104}$$

In Fig. (5), overall trajectory of the spacecraft is depicted along with the representation of field of view angle cone (gray) and glide slope cone (red). We note that the shape of overall trajectory is affected by the initial and desired attitudes, which is not the case in the point mass model. This is due to the fact that the rotational and translational motions are combined with dual quaternions and the shortest path between two points on the unit dual quaternion manifold is exhibited as a screw motion in $\mathsf{SE}(3)$. By weighting the second part of the dual quaternion which is related to position, we can easily modify the convergence rate between rotational and translational motions.

Fig. 6 shows the time histories for the position and attitude in unit quaternions of the spacecraft which is observed with respect to an inertially fixed frame while Fig. 7 depicts the time histories for translational and angular velocities in the body-fixed frame. The translational forces and torques acting along the body-fixed axes are represented in Fig. 8. Fig. 9 shows the trace of the deviation angle over time from the field of view vector $\boldsymbol{y}_b$ to the object. As we can expect from the spacecraft configuration in Fig. 4, at rendezvous this value converges to zero. Additionally the glide slope angle constraint is defined by 30° and Fig. 10 represents the glide slope angle and the trajectory of position over time. All constraints are well regulated.

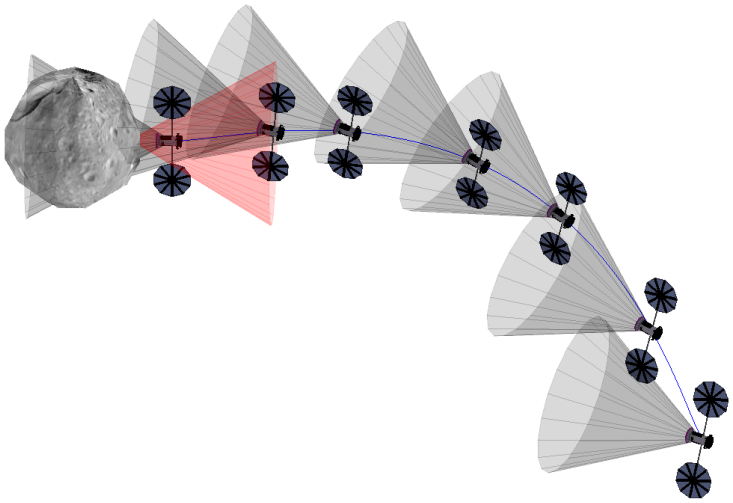

Figure 5. Position and orientation time history of a chasing spacecraft during rendezvous with an tumbling object

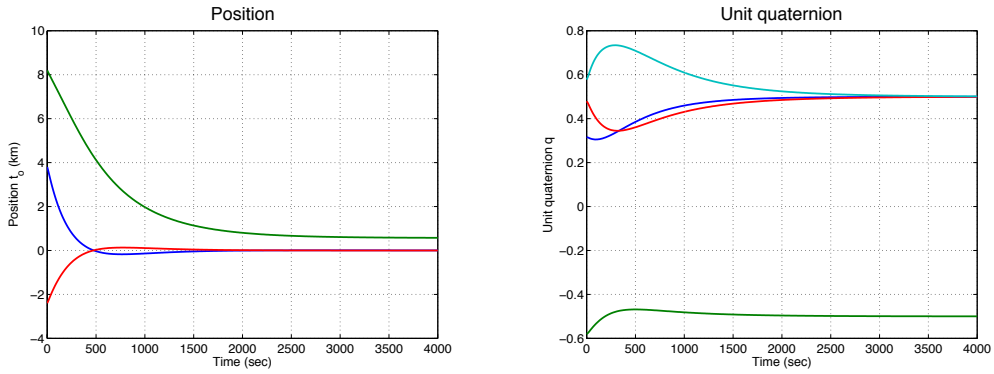

Figure 6. Position and orientation time history of a chasing spacecraft during rendezvous with an tumbling object

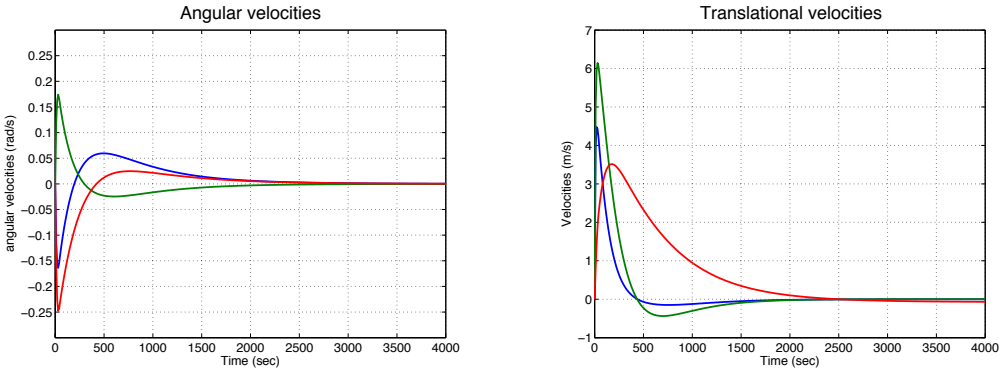

Figure 7. Position and orientation time history of a chasing spacecraft during rendezvous with an tumbling object

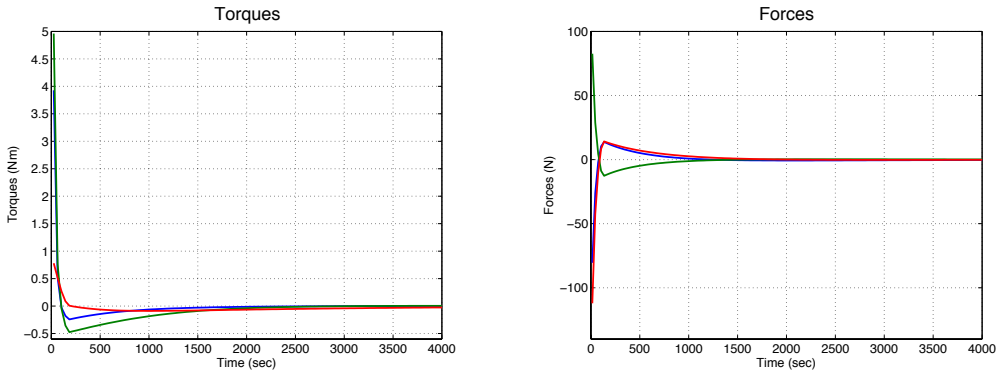

Figure 8. Position and orientation time history of a chasing spacecraft during rendezvous with an tumbling object

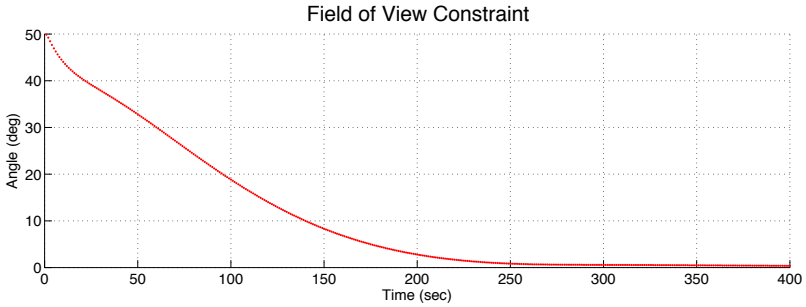

Figure 9. Trace of deviating angle over time from Field of View vector y_b to the object.

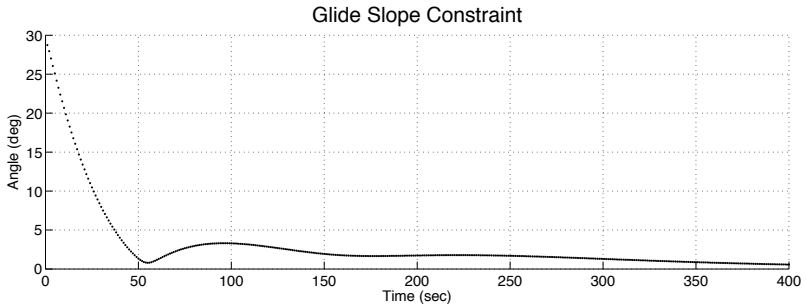

Figure 10. Trace of deviating angle over time from Glide slop angle

CONCLUSION

In this paper, we studied rendezvous and proximity operations between a spacecraft and a target object, such as an asteroid, in the presence of required motion constraints. The first constraint ensures the vision sensor is continuously exposed to the target object. The second constraint forces the spacecraft to maintain a reasonably shallow glide slope while approaching for rendezvous. We developed a new approach to this problem based on the the dual quaternion parameterization. As a potential onboard autonomous guidance system, piece-wise affine model predictive control has been implemented for suboptimal control laws. The simulation results conducted by MATLAB has shown that the proposed MPC controller was able to stabilize the spacecraft to the rendezvous position in good performance without constraint violation.

REFERENCES

[1] Brophy, et al, *"Asteroid Retrieval Feasibility Study,"* Keck Institute for Space Studies, California Institute of Technology, Jet Propulsion Laboratory, 2012.

[2] L. D. Berkovitz, *Convexity and Optimization in* $\mathbb{R}^n$*,* John Wiley & Sons, Inc., 2002.

[3] S. Boyd and L. Vandenberghe, *Convex Optimization,* Cambridge University Press, 2004.

[4] Keck Institute for Space Studies, *Asteroid Retrieval feasibility Study,* 2012.

[5] T. Griebel, *Solar Electric Propulsion Benefits for NASA and On-Orbit Satellite Servicing,* NASA Glenn Research Center, 2010.

[6] J. T. Wen, and K. Kreutz-Delgado, "The Attitude Control Problem," *IEEE Transaction on Automatic Control*, 1991, 36(10): 1148-1162.

[7] A. T. Yang, "Application of Quaternion Algebra and Dual Numbers to the Analysis of Spatial Mechanisms," Ph.D dissertation, Columbia University, 1963.

[8] J. M. McCarthy, *An Introduction to Theoretical Kinematics*, MIT Press, 1990.

[9] Y. Wu, X. Hu, D. Hu, and J. Lian, "Strapdown Inertial Navigation System Algorithms Based on Dual Quaternions," *IEEE Transaction on Aerospace and Electronic Systems*, vol. 41, no. 1. pp. 110-132, 2005.

[10] H. L. Pham, V. Perdereau, B. V. Adorno, and P. Fraisse, "Position and orientation control of robot manipulators using dual quaternion feedback," *in Proceeding of IEEE/RSJ International Conference of Intelligence Robots System,* Taipei, Taiwan, 2010, pp. 658663.

[11] W. H. Clohessy, and R. S. Wiltshire, "Terminal Guidance for Satellite Rendezvous," *Journal of Aerospace Sciences,* vol. 27, pp. 653, 1960.

[12] B. Wie, *Spacecraft Dynamics and Control,* 2nd Edition, AIAA 2010.

[13] J. R. Wertz, *Spacecraft Attitude Determination and Control,* Kluwer, Norwell, MA, 1978.

[14] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[15] S. D. Cairano, H. Park, and I. Kolmanovsky, "Model predictive control approach to guidance of spacecraft rendezvous and proximity maneuvering,"
it International Journal of Robust and Nonlinear Control, vol. 22, no. 12, pp. 13981427, 2012.

[16] E. Hartley, P. Trodden, A. Richards, and J. Maciejowski, "Model predictive control system design and implementation for spacecraft rendezvous," *Control Engineering Practice,* vol. 20, no. 7, pp. 695713, 2012.

[17] M. Saponara, V. Barrena, A. Bemporad, E. Hatley, J. Maciejowski, A. Richards, A. Tramutola, and P. Trodden, "Model Predictive Control application to spacecraft rendezvous in Mars return scenario," *Proceedings of 4th European Conference for Aerospace Sciences (EUCASS),* St. Petersburg, Russia, July, 2011.

[18] R. F. Stengel, *"Flight Dynamics"*, Princeton University Press, 2004.

[19] A. Perez and J. M. McCarthy, "Dual Quaternion Synthesis of Constrained Robotic Systems," *Journal of Mechanical Design*, vol. 126, no. 3, pp. 425-435, 2004.

[20] S. Liu, and G. Trenkler, " "Hadamard, Khatri-Rao, Kronecker and other Matrix Products," *International Journal of Information and Systems Sciences,* vol. 4, no. 1, pp. 160-177, 2008.

[21] J. B. Rawlings and D. Q. Mayne, *Model predictive control: Theory and design,* Nob Hill Publishing, 2009.

[22] E. F. Camacho and C. Bordons, *Model predictive control,* Springer-Verlag, London, 2004.

[23] A. Al-Tamimi, F. L Lewis, and M. Abu-Khalaf, "Discrete-Time Nonlinear HJB Solution Using Approximate Dynamic Programming: Convergence Proof," *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics,* vol. 38, no. 4, 2008.

[24] R.E. Kalman and J.E. Bertram, "Control system analysis and design via the second method of Lyapunov I: Continuous-time systems," *Transactions of the ASME. Series D, Journal of Basic Engineering,* vol. 82, pp. 371393, 1960.

[25] J. R. Silvester, "Determinants of Block Matrices," *The Mathematical Gazette,* Vol. 84, No. 501 , pp. 460-467, Nov., 2000.

[26] R.E. Kalman and J.E. Bertram, "Control system analysis and design via the second method of Lyapunov II: Discrete-time systems," *Transactions of the ASME. Series D, Journal of Basic Engineering,* vol. 82, pp. 94400, 1960.

[27] M. Lazar, W. P. M. H. Heemels, S. Weiland, and A. Bemporad, "Stabilizing Model Predictive Control of Hybrid Systems," *IEEE Transactions on Automatic Control,* vol. 51, no. 11, 2006.