

Relative Navigation in Asteroid Missions Using Dual Quaternion Filtering

Razgus, B.; Mooij, Erwin; Choukroun, Daniel

DOI

[10.2514/1.G002805](https://doi.org/10.2514/1.G002805)

Publication date

2017

Document Version

Accepted author manuscript

Published in

Journal of Guidance, Control, and Dynamics: devoted to the technology of dynamics and control

Citation (APA)

Razgus, B., Mooij, E., & Choukroun, D. (2017). Relative Navigation in Asteroid Missions Using Dual Quaternion Filtering. *Journal of Guidance, Control, and Dynamics: devoted to the technology of dynamics and control*, 40, 2151-2166. <https://doi.org/10.2514/1.G002805>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

Relative Navigation in Asteroid Missions using Dual Quaternion Filtering

B. Razgus^a and E. Mooij^b

Delft University of Technology, Faculty of Aerospace Engineering,

Kluyverweg 1, 2629 HS Delft, The Netherlands

D. Choukroun^c

Ben-Gurion University of the Negev, POB 653, 84105, Beer Sheva, Israel

This paper investigates the efficacy of dual quaternion filtering in the realm of asteroid missions. The main contribution is the development of a dual quaternion relative navigation filter applied to asteroid circumnavigation. The simulated target asteroid is Kleopatra, a dog-bone shaped asteroid featuring a low potential highly-perturbed gravity field. The spacecraft is equipped with a navigation camera and a laser ranger for position sensing, and a star tracker and rate gyroscope for attitude sensing. The paper innovates in the methods for landmarks identification within a camera field-of-view, true range and ranging errors determination, and spacecraft gravity-gradient torque modeling. For the sake of comparison, a navigation filter based on a conventional pose representation using Cartesian coordinates position and attitude quaternion is developed and tested under the same conditions. The dual quaternion filter succeeds in estimating the relative pose with high accuracy, as well as the gyroscope drift and the asteroid angular rates. The latter depends on the frequency and geometry of the landmarks lines-of-sight detected within the camera field-of-view. Significant gains in the transients of the estimation errors are achieved by the dual quaternion filter, when compared to the conventional filter. The errors feature similar steady-state levels in both filters.

^a MSc graduate, section Astrodynamics and Space Missions/ Trainee at ESA, ESTEC

^b Assistant Professor, section Astrodynamics and Space Missions, e.mooij@tudelft.nl, Associate Fellow AIAA.

^c Senior Lecturer, Mechanical Engineering Dept., Senior Member AIAA

I. Introduction

Asteroids are gaining increasing attention among space agencies and industries due to their preserved state since the beginning of the Solar System, the opportunity of mining rare materials, and the need to monitor potential “killer asteroids”. As of June 2016 three missions dedicated to small bodies were launched and completed. NASA’s NEAR visited and landed on the asteroid Eros in 2000 [1]. JAXA’s Hayabusa retrieved samples from the Itokawa asteroid [2], and ESA’s Rosetta successfully put a lander on a comet in 2014 [3] before being controlled-crashed in 2016 as part of its end-of-life strategy. For these missions’ success precise navigation relative to the asteroid/comet was critical. While Earth-based measurements can provide accurate navigation, they also introduce time delays that could be too large for proximity manoeuvres. Some level of autonomy in navigation is therefore required, which is typically enabled via relative sensing systems like navigation cameras [4] and laser rangars [5, 6]. It is common practice to develop navigation algorithms based on relative pose modeling, rather than inertial pose modeling. Spacecraft pose is conventionally represented by a position vector augmented by the attitude quaternion or matrix [7, 8]. An in-depth development of a two-spacecraft relative navigation filter using the conventional pose model is presented in [7]. Dual quaternions provide an alternate representation [9–11] and have spawned growing attention with applications ranging from image processing to spacecraft navigation [12–16]. Research shows that it could result in higher accuracy and faster convergence [14, 15], while redundancy in the position representation can be efficiently handled via order reduction [16], alike the approach used for multiplicative attitude quaternion EKF [17, 18].

Yet, there have been no attempts to our knowledge to develop dual quaternion based filters adapted to asteroids circumnavigation. In [16] the measurement model incorporates velocity and/or acceleration measurements only, which is not adapted to state-of-art sensors that equip nowadays spacecraft for asteroid circumnavigation. Strapdown inertial navigation systems [13] are not yet adapted to asteroid missions as they heavily rely on the asteroid’s gravity field knowledge - a dominant source of uncertainty. In [12] the pose process equation omits translational dynamics, the state vector does not include a full dual quaternion, and no dual quaternion linearized dynamics are developed.

A realistic description of the asteroid’s gravity field is sine-qua-non for the Truth model to which the navigation filter is applied. A vast body of research exists on the topic [19–22]. An accurate representation of the gravity field of an irregular body can be provided by a polyhedron model [19], used in this work. Further keys to a realistic simulation are: (a) the landmarks identification within a camera field-of-view, (b) the true range and ranging errors evaluation, (c) the spacecraft gravity-gradient Truth modeling, which is especially challenging because of the combination of a low potential very irregular gravity field with large spacecraft dimensions [23, 24]. Simplifying approaches like randomly appearing landmarks in the field of view [6], ellipsoid body shape [6], or limited expansions of the gravity-gradient torque [25] allow for analytical, sometime tedious, treatment yet impair the model realism.

In this paper we aim at investigating the efficacy of dual quaternion modeling in the realm of asteroid missions. The main contribution is the development of a dual quaternion relative navigation filter (DQEKF) applied to asteroid circumnavigation. The paper also innovates in the methods for (a) landmarks identification within a camera field-of-view, (b) true range and ranging errors determination, (c) spacecraft gravity-gradient torque Truth modeling. The target asteroid for simulations is Kleopatra, a dog-bone shaped asteroid featuring a low potential highly-perturbed gravity field. A navigation filter is developed using the conventional pose representation [7] (QVEKF) and tested under the same conditions for the sake of comparison. The simulations illustrate significant gains in the transients of the estimation errors when using the DQEKF. The reduction in the settling times of the DQEKF errors range from 1500 s for the velocity of the spacecraft relative to the asteroid, to 1000 s for the relative position, attitude, and asteroid rate. The polyhedron modeling properties are exploited as follows: 1/ to determine which surface point is actually illuminated by the laser under pointing control errors. This in turn enables a truthful, albeit numerical, determination of the actual ranging errors and therefore of their statistics - a key feature for the filter - capturing the traditional foreshortening effect while not being constrained to simplified assumptions of the asteroid body shape, 2/ to determine which landmarks are within the camera polyhedron-shaped field of view. This easy-to-implement approach does away with the arbitrary in picking the landmarks lines-of-sight. The spacecraft is here modeled as a mass-point grid. Our numerical findings suggest

that tuning coefficients can be evaluated in order to retrieve the gravity-gradient torque accuracy of a large grid model while carrying low grid model computations.

Section II provides an overview of the process Truth model. Section III presents the sensors hardware mathematical models for the navigation camera and the laser-ranger measurements. The QVEKF is developed in Sec. IV. Section V introduces dual quaternions for pose representation. Sec. VI presents the development of the DQEKF. Sec. VII includes simulation results and analysis. Conclusions are drawn in Sec. VIII.

II. Dynamics

Asteroids have one of the most perturbed environments in the Solar System. Weak and non-central gravity fields, irregular shapes, fast spinning and other disturbances make the dynamics in the vicinity of an asteroid very unpredictable, thus simulating it has to be discussed in a greater detail than the cases of conventional Earth-based missions.

A. Reference frames

Three reference frames are used throughout this paper: inertial, asteroid and body, denoted by \mathcal{F}_I , \mathcal{F}_A and \mathcal{F}_B , respectively. Inertial and asteroid frames have the same origin (the centre of mass of the asteroid), with the latter being fixed to the asteroid itself (rotating together with the asteroid around the Z-axis), and the body frame is fixed to the spacecraft.

B. Translational Motion

The dynamics are simulated in the inertial frame, and the equations of motion are written in the following form.

$$\dot{\mathbf{R}}_I = \mathbf{V}_I \tag{1}$$

$$\dot{\mathbf{V}}_I = \mathbf{g}_I + \mathbf{a}_{dist} \tag{2}$$

where \mathbf{R}_I , \mathbf{V}_I and \mathbf{g}_I are a position, velocity and gravity field vector in the \mathcal{F}_I frame, respectively. \mathbf{a}_{dist} is a perturbing acceleration, due to non-gravitational disturbances. It is assumed that no control is present, so no external propulsion force.

Polyhedron Gravity Field

Since the gravity field of asteroids is highly irregular, a central gravity-field approximation would be inaccurate. Having a 3D surface model (polyhedron) of an asteroid and assuming constant density, the most accurate way to simulate it is a constant-density polyhedron gravity model, first introduced by Werner and Scheeres [19]. The gradient of the potential is given as a sum over the edges and faces:

$$\mathbf{g}_A = \nabla U = -G\rho \sum_{e \in \text{edges}} \mathbf{E}_e \mathbf{r}_e L_e + G\rho \sum_{f \in \text{faces}} \mathbf{F}_f \mathbf{r}_f \omega_f \quad (3)$$

where G is the gravitational constant, ρ is the mean density of the asteroid (assumed to be constant), \mathbf{r}_e and \mathbf{r}_f are the distances from a field point (in this case the spacecraft) to an edge and a face respectively, \mathbf{E}_e is a dyadic matrix of an edge, \mathbf{F}_f is an outer product of a face normal, and L_e is a dimensionless per-edge factor, representing the potential of the edge. For a complete derivation and implementation, the interested reader is referred to Werner and Scheeres [19]. The sum through all faces of dimensionless per-face factor ω_f gives a solid angle, which vanishes if the field point is outside the polyhedron, or equals 4π , if it is inside the volume. This is a very useful property of polyhedron models, since it gives a mean to know, whether a spacecraft crashes into an asteroid. Moreover, this property will be exploited for hardware simulations later on.

Disturbance forces

Third-body perturbations for the orbits simulated in this paper are negligible, so only solar-radiation-pressure (SRP) force is simulated. It can be calculated as a sum over all (N) illuminated planes of the spacecraft [18]. The simulated spacecraft has solar panels with a reflectivity of $\epsilon_{SP} = 0.21$ and the body with $\epsilon_B = 0.5$.

C. Rotational Motion

Attitude Representation

There are many ways to represent the attitude of a spacecraft, each having its advantages and disadvantages. Commonly, quaternions proved to be the most efficient, singularity-free choice, so they will be used for this study. Unfortunately, quaternions are not uniquely defined, and since different authors use various conventions, we will use the one in Markley and Crassidis' book [18].

A quaternion is defined as a four-dimensional vector with the first three components being a vectorial part and the last one a scalar component. For attitude description only unit quaternions are used, thus their norm has to be one. Furthermore, two quaternion products are defined as follows:

$$[\mathbf{q} \otimes] = \begin{bmatrix} q_4 \mathbf{I}_3 - [\mathbf{q}_{1:3} \times] & \mathbf{q}_{1:3} \\ -\mathbf{q}_{1:3}^T & q_4 \end{bmatrix}, \quad [\mathbf{q} \odot] = \begin{bmatrix} q_4 \mathbf{I}_3 + [\mathbf{q}_{1:3} \times] & \mathbf{q}_{1:3} \\ -\mathbf{q}_{1:3}^T & q_4 \end{bmatrix} \quad (4)$$

and the following holds:

$$\mathbf{q}_1 \otimes \mathbf{q}_2 = \mathbf{q}_2 \odot \mathbf{q}_1 \quad (5)$$

The first product is more often used, since it represents the composition of rotations in a similar way as it is done with a direction cosine matrix (DCM). Let $\mathbf{q}_{B/A}$ denote the attitude quaternion from \mathcal{F}_A to \mathcal{F}_B , it is decomposed as follows:

$$\mathbf{q}_{B/A} = \mathbf{q}_{B/I} \otimes \mathbf{q}_{I/A} \quad (6)$$

where $\mathbf{q}_{I/A}$ denotes the quaternion from \mathcal{F}_A to \mathcal{F}_I and $\mathbf{q}_{B/I}$ denotes the quaternion from \mathcal{F}_I to \mathcal{F}_B . Let \mathbf{v}_A and \mathbf{v}_B denote the projections of the same physical vector onto the frames \mathcal{F}_A and \mathcal{F}_B , respectively, then

$$\mathbf{v}_B = \mathbf{q}_{B/A} \otimes \mathbf{v}_A \otimes \mathbf{q}_{A/B} \quad (7)$$

Kinematics and Dynamics

The most appealing property of using quaternions is the kinematic equation, which is linear with respect to the quaternion. For a spacecraft attitude relative to an inertial frame, it is written as:

$$\dot{\mathbf{q}}_{B/I} = \frac{1}{2} \boldsymbol{\omega}_{B/I}^B \otimes \mathbf{q}_{B/I} \quad (8)$$

where $\boldsymbol{\omega}_{B/I}^B$ is the angular velocity of \mathcal{F}_B with respect to \mathcal{F}_I , expressed in \mathcal{F}_B . Note that in this formulation real vectors, formulated as quaternions, have the fourth component (scalar part) equal to zero. According Eq. (4), the kinematic equation is then written as:

$$\dot{\mathbf{q}}_{B/I} = \frac{1}{2} \boldsymbol{\Omega}(\boldsymbol{\omega}_{B/I}^B) \mathbf{q}_{B/I}, \quad \text{where } \boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} -[\boldsymbol{\omega} \times] & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix} \quad (9)$$

Furthermore, the dynamics of a spacecraft is given by Euler's equation:

$$\dot{\boldsymbol{\omega}}_{B/I}^B = \mathbf{I}^{-1} \left(-\boldsymbol{\omega}_{B/I}^B \times \mathbf{I} \boldsymbol{\omega}_{B/I}^B + \mathbf{T}_{dist} \right) \quad (10)$$

where \mathbf{I} and \mathbf{T}_{dist} denote, respectively, the inertia tensor and a disturbance torque resultant acting on the spacecraft, expressed in \mathcal{F}_B .

Disturbance Torques

Disturbance torques can either be internal or external. Internal torques are caused by fuel sloshing, mass imbalances in reaction wheels, etc., however, these are currently assumed to be absent. External torques are caused by sources, such as solar-radiation pressure, a gravity gradient, magnetic fields and an atmosphere. Since an asteroid does not have a significant magnetic field, nor an atmosphere, these can be discarded. Moreover, the simulated spacecraft is axi-symmetric, both in geometry and surface properties, so the SRP torque is neglected. Thus, only a gravity-gradient torque will be simulated. Many sources give the gravity-gradient torque for a central gravity field as [18, 23]:

$$\mathbf{T}_{gg} = 3 \frac{\mu}{R^3} \mathbf{a}_3 \times \mathbf{I} \mathbf{a}_3 \quad (11)$$

where μ is the gravitational parameter, R is the distance to the attracting body and \mathbf{a}_3 is the third component of Local Vertical Local Horizontal reference frame, which has nadir direction. Equation (11) only holds for central gravity fields, which is not true for an asteroid, so a new way to simulate this torque is needed. In general, the gravity-gradient torque arises from different gravity field strengths at different parts of the spacecraft body. We can assume the spacecraft to be made of N point-masses, as seen in Figure 1.

Then the torque is given by:

$$\mathbf{T}_{gg,B} = \sum_{i=1}^N \mathbf{R}_{B,i} \times m_i \mathbf{g}_{B,i} \quad (12)$$

where $\mathbf{R}_{B,i}$ is the distance of the point-mass from the centre of mass of the SC, $\mathbf{g}_{B,i}$ is gravity field strength at the point-mass (calculated with polyhedron model), and m_i is its mass. Obviously, the more point-masses are taken into account, the more accurate the torque will be. Figure 2 shows the comparison of GG torques computed with different number of point-masses. There is an obvious

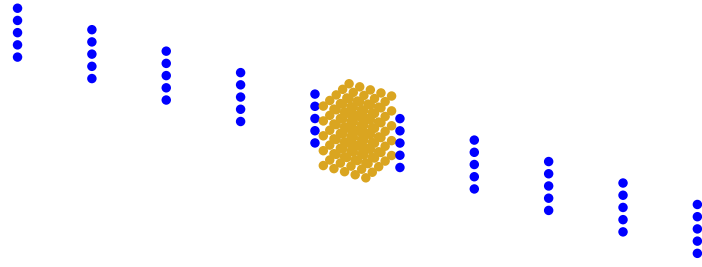


Fig. 1: Spacecraft made of $N=175$ point-masses.

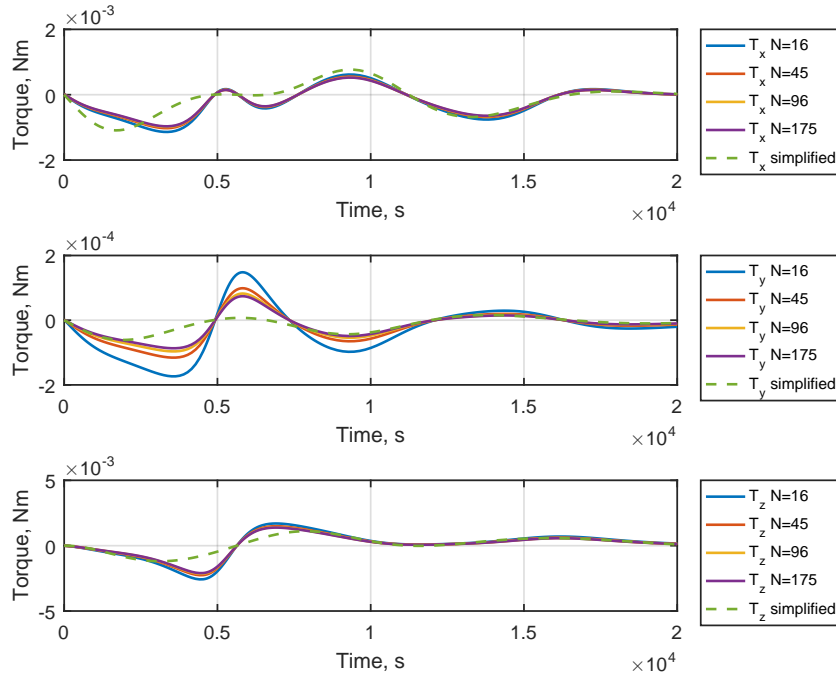


Fig. 2: Gravity gradient torques with different number of masses compared.

convergence trend seen, when the number of point-masses increases, whereas the simplified torque model is instead inaccurate. To understand the errors better, it is assumed that the model with 175 point masses is the ‘real’ one, since it is the most accurate from the given ones, and the relative errors are computed as $T = \frac{|T_{real} - T(N)|}{T_{real}} \times 100\%$. The results, plotted in Figure 3, are remarkable. All the relative errors stay constant. The constant is clearly a function of the number of point

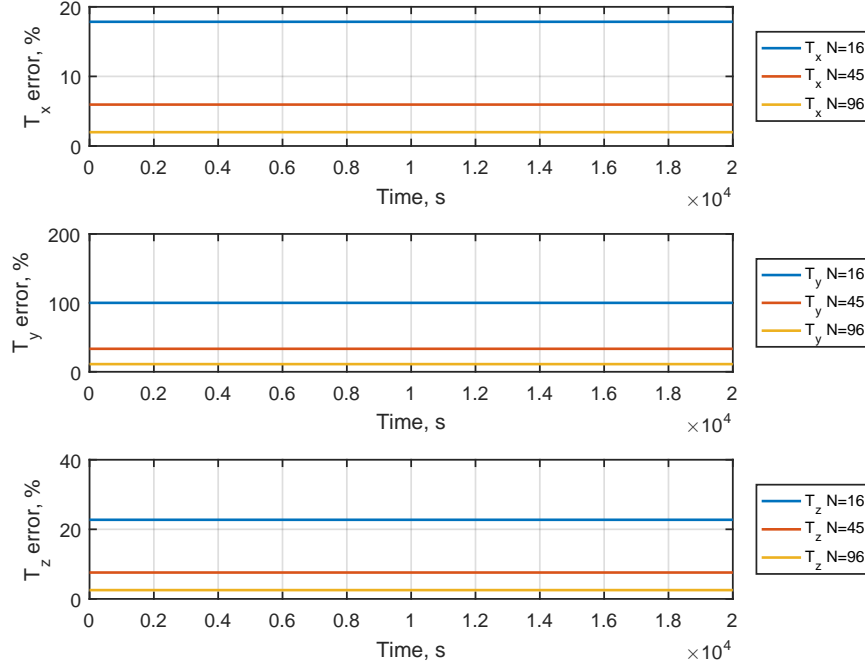


Fig. 3: Gravity Gradient torques with various number of point-masses compared to $N=175$ torque.

masses taken, N . As a result, it suggests a different way of calculating GG torque accurately. Since:

$$\frac{T_{real} - T(N)}{T_{real}} = \text{const}(N) \quad (13)$$

Then the real gravity gradient torque can be computed as:

$$T_{real} = \frac{T(N)}{1 + \text{const}(N)} \quad (14)$$

So we need to evaluate the constant for a large number of point-masses taken, then use the least number of point masses, e.g., 16 to simulate the GG torque, which is eventually corrected, using Eq. (14). However, this method is based on empirical data and lacks some background theory to explain why the error is constant.

D. Asteroid Dynamics

The orbit of an asteroid around the Sun is not simulated, since the simulation time is up to (only) three hours, therefore the movement around the Sun is negligible. Furthermore, the angular velocity of the asteroid is assumed to be constant in direction and magnitude, ignoring any nutation and/or precession:

$$\omega_{A/I}^A = \text{const} \quad (15)$$

This leaves only the attitude of the asteroid to be simulated, which, if expressed in quaternions, is given similarly to the SC kinematics, Eq. (9):

$$\dot{\mathbf{q}}_{A/I} = \frac{1}{2} \boldsymbol{\Omega} \left(\boldsymbol{\omega}_{A/I}^A \right) \mathbf{q}_{A/I} \quad (16)$$

In other words, if one knows the initial attitude of the asteroid and its angular velocity, he will also know it at any time in the future.

E. Relative States

Until now, the dynamics was expressed with respect to an inertial reference frame, but since the navigation filter will deal with relative states, these need to be expressed. Let \mathbf{R}_A denote the position vector from the center of frame \mathcal{F}_A to the center of frame \mathcal{F}_B , expressed in \mathcal{F}_A , a.k.a ‘the spacecraft relative position’. Let \mathbf{V}_A denote its time-derivative with respect to \mathcal{F}_A expressed in \mathcal{F}_A , a.k.a ‘the spacecraft relative velocity’. The states describing the spacecraft relative dynamics and kinematics are thus \mathbf{R}_A , \mathbf{V}_A , $\mathbf{q}_{B/A}$, and $\boldsymbol{\omega}_{B/A}^B$. Let us start with the position and velocity. Relative position is simply obtained by frame transformation:

$$\mathbf{R}_A = \mathbf{C}_{A/I} \mathbf{R}_I \quad (17)$$

where $\mathbf{C}_{A/I}$ is expressed with $\mathbf{q}_{A/I}$ as follows:

$$\mathbf{C}(\mathbf{q}) = \left(q_4^2 - \|\mathbf{q}_{1:3}\|^2 \right) \mathbf{I}_3 + 2\mathbf{q}_{1:3}\mathbf{q}_{1:3}^T - 2q_4 [\mathbf{q}_{1:3} \times] \quad (18)$$

Differentiating Eq. (17) yields the relative velocity:

$$\mathbf{V}_A = \mathbf{C}_{A/I} \mathbf{V}_I - \boldsymbol{\omega}_{A/I}^A \times \mathbf{R}_A \quad (19)$$

The relative attitude quaternion is:

$$\mathbf{q}_{B/A} = \mathbf{q}_{B/I} \otimes \mathbf{q}_{A/I}^{-1} \quad (20)$$

where $\mathbf{q}_{A/I}^{-1}$ denotes the quaternion conjugate, $\mathbf{q}_{A/I}^*$, and the angular rate:

$$\boldsymbol{\omega}_{B/A}^B = \boldsymbol{\omega}_{B/I}^B - \mathbf{C}_{B/A} \boldsymbol{\omega}_{A/I}^A \quad (21)$$

where $\mathbf{C}_{B/A}$ is obtained from $\mathbf{q}_{B/A}$.

III. Hardware

The simulated spacecraft is equipped with a star tracker and a gyroscope for inertial state measurements, a navigation camera and a laser ranger for relative state measurements.

A. Gyroscope

Gyroscopes measure the inertial angular velocity directly, but, as all sensors, they are susceptible to noise. Moreover, they experience a drift, which is an accumulated noise over time. Other errors, such as misalignment and scale errors are discarded for now, assuming that the knowledge of these parameters is absolute. In continuous time, the measured rate, $\tilde{\omega}_{B/I}^B$ (the tilde symbol denotes measured values), of a spacecraft can be written as:

$$\tilde{\omega}_{B/I}^B(t) = \omega_{B/I}^B(t) + \boldsymbol{\mu}(t) + \boldsymbol{\eta}_v(t) \quad (22)$$

$$\dot{\boldsymbol{\mu}}(t) = \boldsymbol{\eta}_u(t) \quad (23)$$

where $\boldsymbol{\mu}$ is the drift, and $\boldsymbol{\eta}_v, \boldsymbol{\eta}_u$ are zero-mean, white noise vectors with known variance. Gyroscopes output values in discrete time, thus it is more precise to simulate them in discrete time [18]:

$$\tilde{\omega}_{k+1} = \omega_{k+1} + \frac{1}{2} (\boldsymbol{\mu}_{k+1} + \boldsymbol{\mu}_k) + \left(\frac{\sigma_v^2}{\Delta t} + \frac{1}{12} \sigma_u^2 \Delta t \right)^{1/2} \boldsymbol{\eta}_{v_k} \quad (24)$$

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k + \sigma_u \Delta t^{1/2} \boldsymbol{\eta}_{u_k} \quad (25)$$

B. Star Tracker

A star tracker measures the position of the stars in its field-of-view and compares this to the known positions on the celestial sphere in the star map, stored on-board. Basically, it would output a single unit vector for each star, and when there are more than one star, one could estimate the attitude. Modern star trackers output a quaternion $\tilde{\mathbf{q}}_{B/I}$ directly, having a simple estimator inside the tracker itself. This will be the way the star tracker is simulated. The real quaternion $\mathbf{q}_{B/I}$ will be affected by a noise quaternion $\delta \mathbf{q}$.

$$\tilde{\mathbf{q}}_{B/I} = \delta \mathbf{q}^{-1} \otimes \mathbf{q}_{B/I} \quad (26)$$

The noise quaternion $\delta \mathbf{q}$ is constructed by normalizing the following vector:

$$\delta \mathbf{q} = \begin{pmatrix} \phi/2 \\ \theta/2 \\ \psi/2 \\ 1 \end{pmatrix} \quad (27)$$

where ϕ , θ , ψ denote mutually independent angular noises that are simulated as zero-mean white Gaussian sequences with known variances. The axis corresponding to the boresight has usually a higher variance than the other two axes. The scale-factors and misalignment errors are assumed to be perfectly known and are not accounted for in this simulation. Otherwise they would be estimated via standard techniques such as state augmentation.

C. Navigation Camera

The navigation camera detects landmarks on the asteroid, identifies them and outputs unit vectors to the corresponding landmarks, which are then fed to the navigation filter. The positions of the landmarks on the asteroid are assumed to be known.

Generating Landmarks

In general, a landmark can be a crater, boulder, ridge or any other distinguishable feature, however, here a landmark is taken as a point on the surface, as no identification algorithm is included. It will also be assumed that the landmarks are spread evenly on the surface of the asteroid. The asteroid's 3D model is given as a polyhedron file with defined vertices and triangular faces. So, first a face is randomly chosen after which a point is placed randomly on the triangle. A way to place a random point on a triangle was investigated by Osada et al [26] in his shape-recognition paper. The random point on a triangle equation is given as:

$$\mathbf{P} = (1 - \sqrt{r_1})\mathbf{A} + \sqrt{r_1}(1 - r_2)\mathbf{B} + \sqrt{r_1}r_2\mathbf{C} \quad (28)$$

where \mathbf{A} , \mathbf{B} and \mathbf{C} are the vertices of a triangle and r_1, r_2 are random numbers, $r_1, r_2 \in [0, 1]$. As a result, Asteroid Kleopatra is depicted with 2000 landmarks in Figure 4. The number 2000 is chosen, because it assures that the navigation camera will see sufficient number of landmarks most of the time, but is not correlated with the actual landmarks of the asteroid.

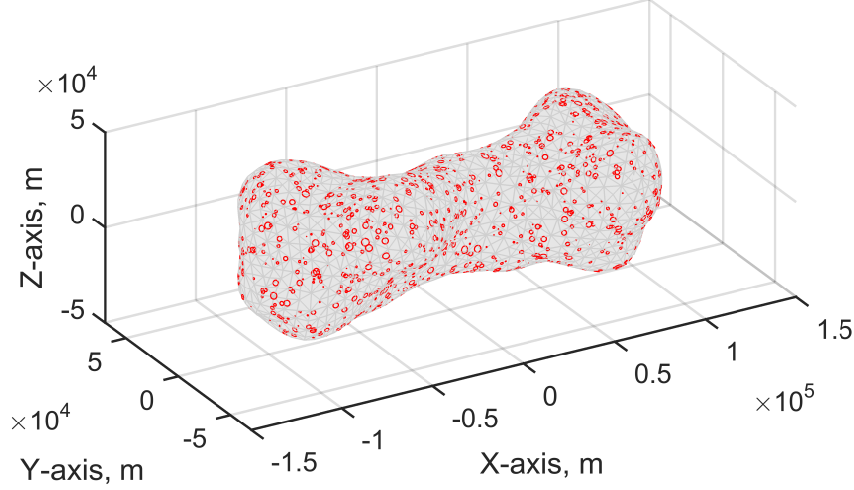


Fig. 4: Asteroid Kleopatra with 2000 landmarks

Each landmark gets an identifier and has its coordinates and normal vector associated to it in the asteroid frame \mathcal{F}_A .

Landmarks in the Field-Of-View

Having the landmark map generated, one needs to know, which ones are seen by the navigation camera (NAVCAM). The parameters of the NAVCAM are taken from the Rosetta mission. The field-of-view (FOV) is a $5^\circ \times 5^\circ$ square with 1024×1024 pixels and a focal length, f , of 152.5 mm. The FOV forms an imaginary pyramid with a square base. If one extends this pyramid until the base is inside the asteroid (if possible), then the landmarks, within the FOV, will be in this pyramid. Then, a polyhedron model is constructed for this pyramid. Now, we recall the useful property of polyhedron models that the sum of the solid angle ω_f through all faces indicates, whether a point lies inside or outside the body. Running through all the landmarks will give those, which are in the FOV of the camera.

Pixel Coordinates

When landmarks in the FOV are identified, their coordinates have to be expressed in the \mathcal{F}_B frame and then projected on the camera plane. For the sake of simplicity, which is standard practice, the focal point of the camera coincides with the origin of \mathcal{F}_B . Figure 5 shows the geometry of the process. So, first the landmark coordinates are transformed:

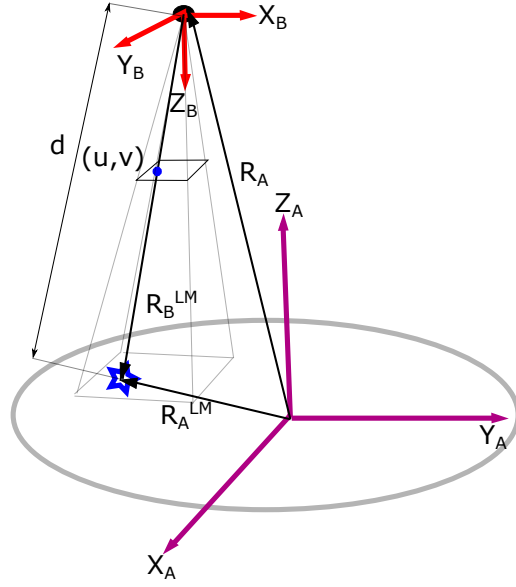


Fig. 5: A landmark projected on a sensor plane

$$\mathbf{R}_B^{LM} = \mathbf{C}_{B/A} (\mathbf{R}_A^{LM} - \mathbf{R}_A) \quad (29)$$

Then, using a pinhole-camera-model, they are projected on the sensor [7] by:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{1}{p} \frac{f}{Z_B^{LM}} \begin{pmatrix} X_B^{LM} \\ Y_B^{LM} \end{pmatrix} \quad (30)$$

where u, v are coordinates in pixels, p is the pixel size, which in this case (Rosetta mission) is $13 \mu\text{m}$.

Furthermore, since the actual landmark-recognition and centre-finding algorithm is not simulated, errors/noise are added to the coordinates:

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \end{pmatrix} = \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} \delta u \\ \delta v \end{pmatrix} \quad (31)$$

where $\delta u, \delta v$ are zero-mean, white noises with known variance, expressed in (sub-)pixels.

Measurement Vector

To implement Eq. (31) in the navigation filter, it is more convenient to use unit vectors to the landmarks and write them in the form:

$$\tilde{\mathbf{b}} = \mathbf{C}(\mathbf{q}) \mathbf{r} \quad (32)$$

where the measured unit vector $\tilde{\mathbf{b}}$ is:

$$\tilde{\mathbf{b}} = \frac{1}{\sqrt{(p\tilde{u})^2 + (p\tilde{v})^2 + f^2}} \begin{pmatrix} p\tilde{u} \\ p\tilde{v} \\ f \end{pmatrix} \quad (33)$$

and \mathbf{r} is the reference vector, which in this case is defined as:

$$\mathbf{r} = \frac{\mathbf{R}_{LM} - \mathbf{R}_A}{\|\mathbf{R}_{LM} - \mathbf{R}_A\|} \quad (34)$$

D. Laser Ranging

When landmarks are detected in the FOV, one of them will be chosen to measure the distance. It is assumed that the laser ranger has a gimbal or mirror system, which enables pointing it to the landmark. However, as reality is not perfect, this pointing introduces an error, which will propagate to the distance error.

Measuring the Distance

The distance, d , between the spacecraft and a landmark can be simply expressed by looking at Fig. 5:

$$d = \|\mathbf{R}_A^{LM} - \mathbf{R}_A\| \quad (35)$$

However, as mentioned before, if a pointing error is introduced, the measured distance can change significantly. Figure 6 (left) shows that when the angle between the laser beam and the local normal increases, the same angular deviation causes larger overshoots. This effect is increased even more for irregular shapes, which is mostly the case for asteroids. As a result, Eq. (35) is not valid any more for getting the measured distance, since the pointing-angle error changes the distance in an unpredictable way. There is no analytical function to solve this problem for an irregular body, thus a numerical solution must be found. The very same useful property of polyhedrons will be used.

Suppose the nominal pointing of the laser is $\mathbf{b}_B^{nom} = (0 \ 0 \ 1)^T$, then this vector is rotated around X- and Y-axis in the \mathcal{F}_B frame in an active (alibi) manner to point to the landmark:

$$\mathbf{b}_B^{LM} = \mathbf{R}_y(\tilde{\theta})\mathbf{R}_x(\tilde{\phi})\mathbf{b}_B^{nom} \quad (36)$$

where $\tilde{\phi}$ and $\tilde{\theta}$ are control angles, extracted from the landmark pixel-coordinates, with introduced

point, the vector $\mathbf{R}_A + \tilde{d}_3 \mathbf{b}_A^{LM}$ is outside the asteroid, so now we would extend it by a quarter of \tilde{d}_1 value and continue the process until a desired accuracy is reached. The measured distance \tilde{d} will be the value from the last iteration, \tilde{d}_i .

IV. Nominal EKF Development

There are many different variants of the Extended Kalman filter, especially concerning attitude with quaternions. A multiplicative quaternion error [17] is chosen for this paper. Furthermore, the nominal EKF will include a position vector, thus the relative pose will be represented by a quaternion-vector pair. This filter will be called QVEKF. The error model, measurement equations and the QVEKF algorithm will be briefly presented.

A. State Vector

First of all, we need to define the state vector for the filter. It should include the relative position, \mathbf{R}_A , and velocity, \mathbf{V}_A , relative attitude, $\mathbf{q}_{B/A}$, and the angular velocity $\boldsymbol{\omega}_{B/A}^B$. The common practice for inertial navigation filters is not to estimate the angular rate directly, but to estimate its drift $\boldsymbol{\mu}$, since the rate is measured by the gyroscope. We will do the same for the relative rate, because it is a function of the inertial angular velocity and the relative attitude, Eq. (21). However, estimating the drift accurately requires frequent attitude measurements, which cannot be achieved by the navigation camera, therefore star-tracker measurements will be used. This requires the inertial attitude, $\mathbf{q}_{B/I}$, to be added to the state vector as well. Finally, the rotation period of the asteroid will also be estimated, assuming that a coarse estimate was available before arrival to the asteroid. So, we have:

$$\mathbf{X}_{20} = \begin{pmatrix} \mathbf{R}_A \\ \mathbf{V}_A \\ \mathbf{q}_{B/A} \\ \mathbf{q}_{B/I} \\ \boldsymbol{\mu} \\ \boldsymbol{\omega}_{A/I}^A \end{pmatrix} \quad (38)$$

B. Process Equation

The navigation filter will supposedly run on-board the spacecraft, so it cannot deal with complex models, e.g., a polyhedron gravity field. As a result, we will use simplified models and introduce a process noise, \mathbf{w} . For the gravity field we will use the central-gravity field approximation with the noise, $\boldsymbol{\eta}_g$, which will be a tuning parameter for the filter. Furthermore, we do not know the real inertial angular velocity of the spacecraft, so the measured one, corrected for the drift, will be used. The errors of the gyroscope are therefore included in the process equation, and they are not treated as measurements by the filter. As a result, the non-linear process equation for the spacecraft around an asteroid is given as:

$$\dot{\mathbf{X}} = \mathbf{f}(\mathbf{X}) + \mathbf{G}\mathbf{w} = \quad (39)$$

$$= \begin{pmatrix} \mathbf{V}_A \\ -\frac{GM}{R_A^3}\mathbf{R}_A - 2\boldsymbol{\omega}_{A/I}^A \times \mathbf{V}_A - \boldsymbol{\omega}_{A/I}^A \times \boldsymbol{\omega}_{A/I}^A \times \mathbf{R}_A \\ \frac{1}{2}\boldsymbol{\Omega}(\tilde{\boldsymbol{\omega}}_{B/I}^B - \boldsymbol{\mu} - \mathbf{C}_{B/A}(\mathbf{q}_{B/A})\boldsymbol{\omega}_{A/I}^A)\mathbf{q}_{B/A} \\ \frac{1}{2}\boldsymbol{\Omega}(\tilde{\boldsymbol{\omega}}_{B/I}^B - \boldsymbol{\mu})\mathbf{q}_{B/I} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} + \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{4 \times 3} & -\frac{1}{2}\Xi(\mathbf{q}_{B/A}) & \mathbf{0}_{4 \times 3} & \mathbf{0}_{4 \times 3} \\ \mathbf{0}_{4 \times 3} & -\frac{1}{2}\Xi(\mathbf{q}_{B/I}) & \mathbf{0}_{4 \times 3} & \mathbf{0}_{4 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix} \begin{pmatrix} \boldsymbol{\eta}_g \\ \boldsymbol{\eta}_v \\ \boldsymbol{\eta}_u \\ \boldsymbol{\eta}_a \end{pmatrix}$$

where $\boldsymbol{\eta}_a$ represents the process noise of the asteroid rate, although it is noiseless, Eq. (15). The noise is only added for tuning the filter. Since the EKF deals with errors of the state, this process has to be written in a linear way for the perturbations:

$$\delta\dot{\mathbf{X}} = \mathbf{F}(\mathbf{X})\delta\mathbf{X} + \mathbf{G}\mathbf{w} \quad (40)$$

where $\mathbf{F}(\mathbf{X})$ is the gradient matrix of $\mathbf{f}(\mathbf{X})$, however, the attitude quaternions are constrained by the norm equal to one, and the Kalman filter is not meant for constrained optimization, so a linear attitude-error model has to be derived.

C. Linear Perturbation Model for Attitude

Inertial Attitude

Let us start with the inertial quaternion. An error between the estimated and the real quaternion

is defined as follows:

$$\delta \mathbf{q}_I = \mathbf{q}_{B/I} \otimes \hat{\mathbf{q}}_{B/I}^{-1} \quad (41)$$

where the hat symbol denotes an estimated value. The perturbation is then differentiated with respect to time:

$$\begin{aligned} \delta \dot{\mathbf{q}}_I &= \dot{\mathbf{q}}_{B/I} \otimes \hat{\mathbf{q}}_{B/I}^{-1} + \mathbf{q}_{B/I} \otimes \dot{\hat{\mathbf{q}}}_{B/I}^{-1} = \\ &= \frac{1}{2} \boldsymbol{\omega}_{B/I}^B \otimes \mathbf{q}_{B/I} \otimes \hat{\mathbf{q}}_{B/I}^* - \mathbf{q}_{B/I} \otimes \frac{1}{2} \hat{\mathbf{q}}_{B/I}^* \otimes \hat{\boldsymbol{\omega}}_{B/I}^B = \frac{1}{2} \boldsymbol{\omega}_{B/I}^B \otimes \delta \mathbf{q}_I - \frac{1}{2} \delta \mathbf{q}_I \otimes \hat{\boldsymbol{\omega}}_{B/I}^B \end{aligned} \quad (42)$$

The real angular rate is:

$$\boldsymbol{\omega}_{B/I}^B = \tilde{\boldsymbol{\omega}}_{B/I}^B - \boldsymbol{\mu} - \boldsymbol{\eta}_v \quad (43)$$

where $\tilde{\boldsymbol{\omega}}_{B/I}^B$ is the measured rate from the gyro. The estimated rate is defined as:

$$\hat{\boldsymbol{\omega}}_{B/I}^B = \tilde{\boldsymbol{\omega}}_{B/I}^B - \hat{\boldsymbol{\mu}} \quad (44)$$

Combining Eqs. (43) and (44) gives:

$$\boldsymbol{\omega}_{B/I}^B = \hat{\boldsymbol{\omega}}_{B/I}^B + \hat{\boldsymbol{\mu}} - \boldsymbol{\mu} - \boldsymbol{\eta}_v = \hat{\boldsymbol{\omega}}_{B/I}^B + \delta \boldsymbol{\omega} \quad (45)$$

where $\delta \boldsymbol{\omega} = \hat{\boldsymbol{\mu}} - \boldsymbol{\mu} - \boldsymbol{\eta}_v$. Substituting the expression above into Eq. (42), yields:

$$\begin{aligned} \delta \dot{\mathbf{q}}_I &= \frac{1}{2} \left(\hat{\boldsymbol{\omega}}_{B/I}^B + \delta \boldsymbol{\omega} \right) \otimes \delta \mathbf{q}_I - \frac{1}{2} \delta \mathbf{q}_I \otimes \hat{\boldsymbol{\omega}}_{B/I}^B = \\ &= \frac{1}{2} \hat{\boldsymbol{\omega}}_{B/I}^B \otimes \delta \mathbf{q}_I - \frac{1}{2} \delta \mathbf{q}_I \otimes \hat{\boldsymbol{\omega}}_{B/I}^B + \frac{1}{2} \delta \boldsymbol{\omega} \otimes \delta \mathbf{q}_I \end{aligned} \quad (46)$$

Then, since the angular rate, when expressed as a quaternion, has the fourth component equal to zero, Eq. (46) simplifies to:

$$\begin{aligned} \delta \dot{\mathbf{q}}_I &= \frac{1}{2} \hat{\boldsymbol{\omega}}_{B/I}^B \otimes \delta \mathbf{q}_I - \frac{1}{2} \hat{\boldsymbol{\omega}}_{B/I}^B \odot \delta \mathbf{q}_I + \frac{1}{2} \delta \boldsymbol{\omega} \otimes \delta \mathbf{q}_I \\ &= \frac{1}{2} \begin{bmatrix} -2 \left[\hat{\boldsymbol{\omega}}_{B/I}^B \times \right] & 0 \\ 0 & 0 \end{bmatrix} \delta \mathbf{q}_I + \frac{1}{2} \delta \boldsymbol{\omega} \otimes \delta \mathbf{q}_I = \end{aligned} \quad (47)$$

$$= \begin{bmatrix} -\left[\hat{\omega}_{B/I}^B \times\right] & 0 \\ \mathbf{0}_{1 \times 3} & 0 \end{bmatrix} \delta \mathbf{q}_I - \frac{1}{2} \delta \boldsymbol{\mu} - \frac{1}{2} \boldsymbol{\eta}_v$$

where $\hat{\omega}_{B/I}^B = \tilde{\omega}_{B/I}^B - \hat{\boldsymbol{\mu}}$ and $\delta \boldsymbol{\mu} = \boldsymbol{\mu} - \hat{\boldsymbol{\mu}}$. For small angle errors, a quaternion can be approximated by a rotation vector:

$$\delta \mathbf{q} \approx \begin{pmatrix} \frac{1}{2} \delta \boldsymbol{\vartheta} \\ 1 \end{pmatrix} \quad (48)$$

Thus, Eq. (47) can be written for the rotation vector:

$$\delta \dot{\boldsymbol{\vartheta}}_I = \left[-\hat{\omega}_{B/I}^B \times \right] \delta \boldsymbol{\vartheta}_I - \delta \boldsymbol{\mu} - \boldsymbol{\eta}_v \quad (49)$$

Relative Attitude

With the relative attitude quaternion, the same process is repeated. The error quaternion, $\delta \mathbf{q}_R$ (R denotes the relative state) is defined as:

$$\delta \mathbf{q}_R = \mathbf{q}_{B/A} \otimes \hat{\mathbf{q}}_{B/A}^{-1} \quad (50)$$

Then, differentiating and expressing the error in rotation-vector form, the following is obtained [7]:

$$\delta \dot{\boldsymbol{\vartheta}}_R = -\left[\hat{\omega}_{B/I}^B \times\right] \delta \boldsymbol{\vartheta}_R - \delta \boldsymbol{\mu} - \mathbf{C}(\mathbf{q}_{B/A}) \delta \boldsymbol{\omega}_A - \boldsymbol{\eta}_v \quad (51)$$

where $\delta \boldsymbol{\omega}_A = \boldsymbol{\omega}_{A/I}^A - \hat{\boldsymbol{\omega}}_{A/I}^A$. Notice that the relative attitude error rate depends on the inertial angular velocity of the spacecraft, rather than the relative angular velocity, and that this result would hold even if the asteroid angular rate was time-varying.

Error-state Vector

Since the attitude errors were expressed with three-dimensional rotation vectors, the error-state vector size decreased by two dimensions, compared to the state vector itself, and is written as follows:

$$\delta \mathbf{X}_{18} = \begin{pmatrix} \delta \mathbf{R}_A \\ \delta \mathbf{V}_A \\ \delta \boldsymbol{\vartheta}_R \\ \delta \boldsymbol{\vartheta}_I \\ \delta \boldsymbol{\mu} \\ \delta \boldsymbol{\omega}_A \end{pmatrix} \quad (52)$$

Transition Matrix

Combining Eqs. (39), (49) and (51) we can obtain the continuous time state transition matrix, \mathbf{F} , for perturbations, Eq. (40). It is written as:

$$\mathbf{F} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \left. \frac{d\mathbf{f}(\mathbf{X})_{4:6}}{d\mathbf{R}_A} \right|_{\mathbf{X}=\hat{\mathbf{X}}} & \left. \frac{d\mathbf{f}(\mathbf{X})_{4:6}}{d\mathbf{V}_A} \right|_{\mathbf{X}=\hat{\mathbf{X}}} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \left. \frac{d\mathbf{f}(\mathbf{X})_{4:6}}{d\boldsymbol{\omega}_{A/I}^A} \right|_{\mathbf{X}=\hat{\mathbf{X}}} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -\left[\hat{\boldsymbol{\omega}}_{B/I}^B \times\right] & \mathbf{0}_{3 \times 3} & -\mathbf{I}_{3 \times 3} & -\mathbf{C}(\mathbf{q}_{B/A}) \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -\left[\hat{\boldsymbol{\omega}}_{B/I}^B \times\right] & -\mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix} \quad (53)$$

where:

$$\frac{\partial \dot{\mathbf{V}}_A}{\partial \mathbf{R}_A} = \mu \frac{3\mathbf{R}\mathbf{R}^T - \mathbf{R}^T \mathbf{R} \mathbf{I}_3}{R^5} + \left(\boldsymbol{\omega}_{A/I}^T \boldsymbol{\omega}_{A/I} \mathbf{I}_3 - \boldsymbol{\omega}_{A/I} \boldsymbol{\omega}_{A/I}^T \right) \quad (54)$$

$$\frac{\partial \dot{\mathbf{V}}_A}{\partial \mathbf{V}_A} = -2 \left[\boldsymbol{\omega}_{A/I} \times \right] \quad (55)$$

$$\frac{\partial \dot{\mathbf{V}}_A}{\partial \boldsymbol{\omega}_{A/I}} = 2 \left[\mathbf{V}_A \times \right] - \boldsymbol{\omega}_{A/I}^T \mathbf{R}_A \mathbf{I}_3 - \boldsymbol{\omega}_{A/I} \mathbf{R}^T + 2\mathbf{R}_A \boldsymbol{\omega}_{A/I}^T \quad (56)$$

and its discrete time version, Φ :

$$\Phi = \exp^{\mathbf{F}\Delta t} \approx \mathbf{I}_{18 \times 18} + \mathbf{F}\Delta t \quad (57)$$

D. Measurement Equations

Since most of the measurements are nonlinear, the measurement matrices, \mathbf{H} , for the filter must be found.

Startracker

Since the star tracker measures the inertial attitude quaternion then a measurement of the filter state $\delta\boldsymbol{\vartheta}_R$ is obtained by computing twice the vector part of the quaternion measurement prediction error, and the associated measurement matrix is expressed as follows:

$$\mathbf{H}_{ST} = \begin{bmatrix} \mathbf{0}_{3 \times 9} & \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 6} \end{bmatrix} \quad (58)$$

Navigation Camera

The navigation camera gives unit vectors to the landmarks, so its measurement to the i^{th} landmark is:

$$\mathbf{z}_{NAVCAM,i} = \mathbf{h}(\mathbf{X}) = \mathbf{C}(\mathbf{q}_{B/A}) \frac{\mathbf{R}_{LM,i} - \mathbf{R}_A}{\|\mathbf{R}_{LM,i} - \mathbf{R}_A\|} \quad (59)$$

To adapt Eq. (59) for the filter, we express the error between the real and predicted measurement.

For this, we write the attitude matrix as follows [7]:

$$\mathbf{C}(\mathbf{q}_{B/A}) = (\mathbf{I}_3 - [\delta\boldsymbol{\vartheta}_R \times]) \mathbf{C}(\hat{\mathbf{q}}_{B/A}) \quad (60)$$

Substituting Eq. (60) into Eq. (59), and defining $\delta\mathbf{z} = \mathbf{z} - \hat{\mathbf{z}}$ yields:

$$\delta\mathbf{z} = \mathbf{C}_{B/A}(\hat{\mathbf{q}}_{B/A}) \frac{\mathbf{R}_{LM,i} - \mathbf{R}_A}{\|\mathbf{R}_{LM,i} - \mathbf{R}_A\|} \times \delta\boldsymbol{\vartheta}_R \quad (61)$$

The measurement error becomes linear with respect to the relative attitude error, but not to the position. As a result, the measurement matrix for a single landmark is:

$$\mathbf{H}_{NAVCAM} = \left[\left. \frac{d\mathbf{h}(\mathbf{X})}{d\mathbf{R}_A} \right|_{\mathbf{X}=\hat{\mathbf{X}}} \quad \mathbf{0}_{3 \times 3} \quad \left[\mathbf{C}_{B/A}(\hat{\mathbf{q}}_{B/A}) \frac{\mathbf{R}_{LM,i} - \hat{\mathbf{R}}_A}{\|\mathbf{R}_{LM,i} - \hat{\mathbf{R}}_A\|} \times \right] \quad \mathbf{0}_{3 \times 9} \right] \quad (62)$$

where:

$$\frac{\partial \mathbf{h}(\mathbf{X})}{\partial \mathbf{R}_A} = \frac{(\mathbf{R}_{LM} - \mathbf{R}_A)(\mathbf{R}_{LM} - \mathbf{R}_A)^T - (\mathbf{R}_{LM} - \mathbf{R}_A)^T(\mathbf{R}_{LM} - \mathbf{R}_A)\mathbf{I}_3}{\|\mathbf{R}_{LM} - \mathbf{R}_A\|^3} \quad (63)$$

Laser Ranger

The laser ranger measures distance to a selected landmark:

$$z_{LR} = h(\mathbf{X}) = \|\mathbf{R}_{LM,i} - \mathbf{R}_A\| \quad (64)$$

To obtain the measurement matrix, we just find a Jacobian of Eq. (64):

$$\mathbf{H}_{LR} = \left[\left. \frac{dh(\mathbf{X})}{d\mathbf{R}_A} \right|_{\mathbf{X}=\hat{\mathbf{X}}} \quad \mathbf{0}_{1 \times 15} \right] \quad (65)$$

where:

$$\frac{\partial h(\mathbf{X})}{\partial \mathbf{R}_A} = \frac{\mathbf{R}_{LM} - \mathbf{R}_A}{\|\mathbf{R}_{LM} - \mathbf{R}_A\|} \quad (66)$$

V. Dual Quaternions For Pose Representation

A dual quaternion is an eight-dimensional vector representing position and attitude in a coupled way. It was first introduced by William Kingdom Clifford in his paper about bi-quaternions [9]. To understand dual quaternions, it is first needed to understand dual numbers and dual algebra. Dual numbers (also called duplexes) are an extension to the real numbers. In the form they are written, dual numbers resemble complex numbers [10]:

$$\check{d} = a + \epsilon b \quad (67)$$

where \check{d} is a dual number (the symbol $\check{}$ denotes a dual quantity), a and b are real numbers, and ϵ has the following properties:

$$\epsilon \neq 0, \quad \epsilon^2 = 0 \quad (68)$$

The first part in Eq. (67) is called the primary (or real) part of the dual number and the second one represents the dual component. For further details about dual numbers and dual algebra the reader is referred to the book of Fischer [10].

A. Dual Quaternion

Similarly to how a dual number is constructed, a dual quaternion (DQ), $\check{\mathbf{q}}$, can be written as [11]:

$$\check{\mathbf{q}} = \mathbf{q}_r + \epsilon \mathbf{q}_d \quad (69)$$

where \mathbf{q}_r is the real and \mathbf{q}_d is the dual part of the dual quaternion, and both of them are quaternions (not necessarily unit quaternions). Furthermore, a dual quaternion can be seen as a dual-hyper-complex vector and can be written in the following form:

$$\check{\mathbf{q}} = q_{r1}i + q_{r2}j + q_{r3}k + q_{r4} + \epsilon (q_{d1}i + q_{d2}j + q_{d3}k + q_{d4}) \quad (70)$$

where i , j and k are imaginary numbers. It is cumbersome to manipulate a dual quaternion algebraically, as in Eq. (70). Thus, in this paper a dual quaternion is seen as an 8-tuple vector, defined

as follows:

$$\tilde{\mathbf{q}} = \begin{pmatrix} \mathbf{q}_r \\ \mathbf{q}_d \end{pmatrix} = \begin{pmatrix} q_{r1} \\ q_{r2} \\ q_{r3} \\ q_{r4} \\ q_{d1} \\ q_{d2} \\ q_{d3} \\ q_{d4} \end{pmatrix} \quad (71)$$

The multiplication ($\check{\otimes}$) of two dual quaternions follows the rules of dual numbers and quaternions.

It reads as follows:

$$\tilde{\mathbf{q}}_1 \check{\otimes} \tilde{\mathbf{q}}_2 = \mathbf{q}_{r1} \otimes \mathbf{q}_{r2} + \epsilon (\mathbf{q}_{r1} \otimes \mathbf{q}_{d2} + \mathbf{q}_{d1} \otimes \mathbf{q}_{r2}) \quad (72)$$

where $\check{\otimes}$ is the dual quaternion multiplication, which in matrix form is written as:

$$[\tilde{\mathbf{q}}_1 \check{\otimes}] \tilde{\mathbf{q}}_2 = \begin{bmatrix} [\mathbf{q}_{r1} \otimes] & \mathbf{0}_{4 \times 4} \\ [\mathbf{q}_{d1} \otimes] & [\mathbf{q}_{r1} \otimes] \end{bmatrix} \begin{pmatrix} \mathbf{q}_{r2} \\ \mathbf{q}_{d2} \end{pmatrix} \quad (73)$$

B. Conjugates

A dual quaternion can have three different conjugates [11]:

$$\tilde{\mathbf{q}}^\diamond = \mathbf{q}_r - \epsilon \mathbf{q}_d \quad (74)$$

$$\tilde{\mathbf{q}}^* = \mathbf{q}_r^* + \epsilon \mathbf{q}_d^* \quad (75)$$

$$\tilde{\mathbf{q}}^\circ = \mathbf{q}_r^* - \epsilon \mathbf{q}_d^* \quad (76)$$

A multiplication of DQ with its second conjugate gives:

$$\tilde{\mathbf{q}} \check{\otimes} \tilde{\mathbf{q}}^* = (\mathbf{q}_r + \epsilon \mathbf{q}_d) (\mathbf{q}_r^* + \epsilon \mathbf{q}_d^*) = \mathbf{q}_r \otimes \mathbf{q}_r^* + \epsilon (\mathbf{q}_r \otimes \mathbf{q}_d^* + \mathbf{q}_d \otimes \mathbf{q}_r^*) \quad (77)$$

A quaternion multiplication with its conjugate ($\mathbf{q}_r \otimes \mathbf{q}_r^*$) gives a quaternion with zero vector part, and the dual part, after some mathematical treatment, turns out to be also a quaternion with zero

vector part. Thus, the result of the second conjugate product is given as follows:

$$\check{\mathbf{q}} \check{\otimes} \check{\mathbf{q}}^* = \|\mathbf{q}_r\|^2 + 2\epsilon (q_{r1}q_{d1} + q_{r2}q_{d2} + q_{r3}q_{d3} + q_{r4}q_{d4}) \quad (78)$$

The result in a general case is a dual quaternion with scalar real and dual parts, but if the real part is orthogonal to the dual one, the dual part vanishes, and the product becomes a dual quaternion with real scalar part.

C. Pose Representation

Up to this point, we have discussed the general properties of dual numbers and dual quaternions, but we have not introduced a way to represent both, the position and the attitude (pose of a Cartesian coordinates frame with respect to another frame) with a dual quaternion. To begin with, the real part of a dual quaternion will be a unit quaternion, thus representing the attitude. Furthermore, the position has to be incorporated as well. Jia [11] gives the dual part of the dual quaternion as a multiplication of the real part with a position vector from the origin frame to the frame the pose is described for:

$$\mathbf{q}_d = \frac{1}{2}\mathbf{q}_r \otimes \mathbf{R} \quad (79)$$

where \mathbf{R} is treated as a quaternion (fourth component is zero). Then we calculate the dual quaternion product with its second conjugate:

$$\begin{aligned} \check{\mathbf{q}} \check{\otimes} \check{\mathbf{q}}^* &= \left(\mathbf{q}_r + \frac{1}{2}\epsilon \mathbf{q}_r \otimes \mathbf{R} \right) \check{\otimes} \left(\mathbf{q}_r + \frac{1}{2}\epsilon \mathbf{q}_r \otimes \mathbf{R} \right)^* = \left(\mathbf{q}_r + \frac{1}{2}\epsilon \mathbf{q}_r \otimes \mathbf{R} \right) \check{\otimes} \left(\mathbf{q}_r^* + \frac{1}{2}\epsilon \mathbf{R}^* \otimes \mathbf{q}_r^* \right) = \\ &= \mathbf{q}_r \otimes \mathbf{q}_r^* + \frac{1}{2}\epsilon (\mathbf{q}_r \otimes \mathbf{R} \otimes \mathbf{q}_r^* + \mathbf{q}_r \otimes \mathbf{R}^* \otimes \mathbf{q}_r^*) \end{aligned} \quad (80)$$

The quaternion conjugate of a pure vector is simply $\mathbf{R}^* = -\mathbf{R}$, so Eq. (80) reduces to a unit dual quaternion, which, means that the dual quaternion has the norm of one. Thus, in a similar way the quaternion of rotation must have a norm of one, a dual quaternion to represent a pose must also have a norm (real number) of one:

$$\check{\mathbf{q}} \check{\otimes} \check{\mathbf{q}}^* = 1 \quad (81)$$

The equation above requires two constraints to be met. The real part of a dual quaternion has to have a norm of one, which is equivalent to the quaternion being a quaternion of rotation, and it is written as follows:

$$\|\mathbf{q}_r\|^2 = 1 \quad (82)$$

Furthermore, a second constraint is introduced, which is derived from the dual part of the dual quaternion product in Eq. (78).

$$(q_{r1}q_{d1} + q_{r2}q_{d2} + q_{r3}q_{d3} + q_{r4}q_{d4}) = \mathbf{q}_r \cdot \mathbf{q}_d = 0 \quad (83)$$

which requires that the dot product between the real and the dual parts is equal to zero. In other words, the two quaternions have to be orthogonal.

A dual-quaternion is an eight-dimensional vector, so these constraints remove two parameters, and thus only six are left, which is the number required to represent the pose.

Now, it is essential to define in which reference frame the position vector is expressed. Suppose having two reference frames \mathcal{F}_A and \mathcal{F}_B . The attitude of the frame \mathcal{F}_B with respect to the frame \mathcal{F}_A is $\mathbf{q}_{B/A}$. Then we would write the position vector in frame \mathcal{F}_A and the resulting dual quaternion, representing the pose of frame \mathcal{F}_B with respect to frame \mathcal{F}_A , would be given as follows:

$$\check{\mathbf{q}}_{B/A} = \mathbf{q}_{B/A} + \frac{\epsilon}{2} \mathbf{q}_{B/A} \otimes \mathbf{R}_A \quad (84)$$

If one wants to use the vector in frame \mathcal{F}_B , then a simple right multiplication of the dual part with a quaternion unity $\mathbf{I}_q = \mathbf{q}^* \otimes \mathbf{q}$ has to be done.

$$\check{\mathbf{q}}_{B/A} = \mathbf{q}_{B/A} + \frac{\epsilon}{2} \mathbf{q}_{B/A} \otimes \mathbf{R}_A \otimes \mathbf{q}_{B/A}^* \otimes \mathbf{q}_{B/A} \quad (85)$$

where $\mathbf{q}_{B/A} \otimes \mathbf{R}_A \otimes \mathbf{q}_{B/A}^*$ is a quaternion frame transformation for vectors in the frame \mathcal{F}_A to be expressed in frame \mathcal{F}_B . As a result, Eq. (85) becomes:

$$\check{\mathbf{q}}_{B/A} = \mathbf{q}_{B/A} + \frac{\epsilon}{2} \mathbf{R}_B \otimes \mathbf{q}_{B/A} \quad (86)$$

Comparing Eqs. (84) and (86), we can see that the same dual parts can be expressed in two different frames by just switching the places of the multiplicands.

To retrieve the attitude quaternion and the position vector from a dual quaternion, one has to do the following steps:

$$\mathbf{q}_{B/A} = \mathbf{q}_r \quad (87)$$

$$\mathbf{R}_A = 2\mathbf{q}_r^* \otimes \mathbf{q}_d \quad (88)$$

D. Screw Displacement

Chasle's theorem states that any rigid displacement is equivalent to a rotation around a line, called the screw axis, followed by a translation in the direction of the line [11]. This allows us to visualize a dual quaternion in a similar way a quaternion can be visualised (Euler theorem).

VI. Dual Quaternion Extended Kalman Filter

This section explains the development of the dual-quaternion Extended Kalman Filter (DQEKF). The main difference between DQEKF and QVEKF is that the relative pose in the latter is expressed by a quaternion-vector pair and in the DQEKF by a dual quaternion. All the other states remain the same. Also, the relative attitude is represented identically in both filters.

A. Relative Pose

Suppose having reference frames \mathcal{F}_A and \mathcal{F}_B defined relative to \mathcal{F}_I . We recall that the \mathcal{F}_I has the same origin as the asteroid frame, so we express their pose in dual quaternions as:

$$\check{\mathbf{q}}_{A/I} = \mathbf{q}_{A/I} + \epsilon \mathbf{0} \quad (89)$$

$$\check{\mathbf{q}}_{B/I} = \mathbf{q}_{B/I} + \frac{\epsilon}{2} \mathbf{q}_{B/I} \otimes \mathbf{R}_I \quad (90)$$

where \mathbf{R}_I is the position vector of the spacecraft, expressed in \mathcal{F}_I . The relative pose is then:

$$\check{\mathbf{q}}_{B/A} = \check{\mathbf{q}}_{B/I} \check{\otimes} \check{\mathbf{q}}_{A/I}^* = \left(\mathbf{q}_{B/I} + \frac{\epsilon}{2} \mathbf{q}_{B/I} \otimes \mathbf{R}_I \right) \left(\mathbf{q}_{A/I}^* + \epsilon \mathbf{0} \right) = \mathbf{q}_{B/I} \otimes \mathbf{q}_{A/I}^* + \frac{\epsilon}{2} \mathbf{q}_{B/I} \otimes \mathbf{R}_I \otimes \mathbf{q}_{A/I}^* \quad (91)$$

The position quaternion can be expressed in the asteroid frame using the quaternion frame transformation:

$$\mathbf{R}_I = \mathbf{q}_{A/I}^* \otimes \mathbf{R}_A \otimes \mathbf{q}_{A/I} \quad (92)$$

Substituting Eq. (92) into Eq. (91) yields:

$$\check{\mathbf{q}}_{B/A} = \mathbf{q}_{B/A} + \frac{\epsilon}{2} \mathbf{q}_{B/I} \otimes \mathbf{q}_{A/I}^* \otimes \mathbf{R}_A \otimes \mathbf{q}_{A/I} \otimes \mathbf{q}_{A/I}^* = \mathbf{q}_{B/A} + \frac{\epsilon}{2} \mathbf{q}_{B/A} \otimes \mathbf{R}_A \quad (93)$$

which is a logical outcome, showing a translation, \mathbf{R}_A , in \mathcal{F}_A , followed by a rotation.

B. DQEKF State Vector

In DQEKF the relative pose is expressed in a dual-quaternion form. We change the state representation accordingly, and the full state vector reads as follows:

$$\mathbf{X}_{21} = \begin{pmatrix} \check{\mathbf{q}}_{B/A} \\ \mathbf{V}_A \\ \mathbf{q}_{B/I} \\ \boldsymbol{\mu} \\ \boldsymbol{\omega}_{A/I}^A \end{pmatrix} \quad (94)$$

We can see that the state representation increased by one dimension, because the position is now expressed in a four-dimensional quaternion form.

C. Dual Quaternion Kinematic Equation

By taking Eq. (93) and differentiating it with respect to time, the following is obtained:

$$\begin{aligned} \dot{\check{\mathbf{q}}}_{B/A} &= \dot{\mathbf{q}}_{B/A} + \frac{\epsilon}{2} \left(\dot{\mathbf{q}}_{B/A} \otimes \mathbf{R}_A + \mathbf{q}_{B/A} \otimes \dot{\mathbf{R}}_A \right) = \\ &= \frac{1}{2} \boldsymbol{\omega}_{B/A}^B \otimes \mathbf{q}_{B/A} + \frac{\epsilon}{2} \left(\frac{1}{2} \boldsymbol{\omega}_{B/A}^B \otimes \mathbf{q}_{B/A} \otimes \mathbf{R}_A + \mathbf{q}_{B/A} \otimes \mathbf{V}_A \right) \end{aligned} \quad (95)$$

This result, after some rearrangement, can be written in a matrix form:

$$\dot{\check{\mathbf{q}}}_{B/A} = \frac{1}{2} \begin{bmatrix} \left[\boldsymbol{\omega}_{B/A}^B \otimes \right] & \mathbf{0}_{4 \times 4} \\ [\mathbf{V}_A \odot] & \left[\boldsymbol{\omega}_{B/A}^B \otimes \right] \end{bmatrix} \begin{pmatrix} \mathbf{q}_r \\ \mathbf{q}_d \end{pmatrix} \quad (96)$$

where $\mathbf{q}_r = \mathbf{q}_{B/A}$ and $\mathbf{q}_d = \frac{\epsilon}{2} \mathbf{q}_{B/A} \otimes \mathbf{R}_A$ are the real and the dual parts of the dual quaternion, respectively. Furthermore the term $\mathbf{q}_{B/A} \otimes \mathbf{V}_A$ in Eq. (95) can be rewritten as:

$$\mathbf{q}_{B/A} \otimes \mathbf{V}_A = \mathbf{q}_{B/A} \otimes \mathbf{V}_A \otimes \mathbf{q}_{B/A}^* \mathbf{q}_{B/A} = \mathbf{V}_B \otimes \mathbf{q}_{B/A} \quad (97)$$

Then, substituting Eq. (97) into Eq. (95) and writing it in matrix form, yields:

$$\dot{\mathbf{q}}_{B/A} = \frac{1}{2} \begin{bmatrix} [\boldsymbol{\omega}_{B/A}^B \otimes] & \mathbf{0}_{4 \times 4} \\ [\mathbf{V}_B \otimes] & [\boldsymbol{\omega}_{B/A}^B \otimes] \end{bmatrix} \begin{pmatrix} \mathbf{q}_r \\ \mathbf{q}_d \end{pmatrix} \quad (98)$$

Here, we introduce a dual velocity, which is defined in \mathcal{F}_B as:

$$\check{\boldsymbol{\omega}}_B = \boldsymbol{\omega}_{B/A}^B + \epsilon \mathbf{V}_B \quad (99)$$

The kinematics equation is then:

$$\dot{\mathbf{q}}_{B/A} = \frac{1}{2} \check{\boldsymbol{\omega}}_B \check{\otimes} \check{\mathbf{q}}_{B/A} \quad (100)$$

where $\check{\otimes}$ is a dual quaternion product and, according to Eq. (98), it is defined as follows:

$$[\check{\boldsymbol{\omega}}_{B/I}^B \check{\otimes}] = \begin{bmatrix} [\boldsymbol{\omega}_{B/A}^B \otimes] & \mathbf{0}_{4 \times 4} \\ [\mathbf{V}_B \otimes] & [\boldsymbol{\omega}_{B/A}^B \otimes] \end{bmatrix} \quad (101)$$

Full process model

Only the relative position representation has changed, so the process function \mathbf{f} changes accordingly:

$$\mathbf{f}(\mathbf{X}) = \begin{pmatrix} \frac{1}{2} \boldsymbol{\Omega} \left(\check{\boldsymbol{\omega}}_{B/I}^B - \boldsymbol{\mu} - \mathbf{C}(\mathbf{q}_r) \boldsymbol{\omega}_{A/I}^A \right) \mathbf{q}_r \\ \frac{1}{2} \boldsymbol{\omega}_{B/A}^B \otimes \mathbf{q}_d + \frac{1}{2} \mathbf{q}_r \otimes \mathbf{V}_A \\ -\frac{GM}{R_A^3} \mathbf{R}_A - 2\boldsymbol{\omega}_{A/I} \times \mathbf{V}_A - \boldsymbol{\omega}_{A/I} \times \boldsymbol{\omega}_{A/I} \times \mathbf{R}_A \\ \frac{1}{2} \boldsymbol{\Omega} (\check{\boldsymbol{\omega}}_{B/I}^B - \boldsymbol{\mu}) \mathbf{q}_{B/I} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \quad (102)$$

D. Linear Perturbation Model for Dual Quaternion

As was done for the QVEKF, a linear perturbation model for dual quaternions is developed first.

Dual quaternion error

The dual quaternion error is defined as:

$$\delta \check{\mathbf{q}} = \check{\mathbf{q}}_{B/A} \otimes \widehat{\check{\mathbf{q}}_{B/A}}^* \quad (103)$$

where $\check{\mathbf{q}}_{B/A}$ is the real dual quaternion and $\widehat{\check{\mathbf{q}}_{B/A}}$ is the estimated one. They are expressed as follows:

$$\check{\mathbf{q}}_{B/A} = \mathbf{q}_{B/A} + \frac{\epsilon}{2} \mathbf{q}_{B/A} \otimes \mathbf{R}_A \quad (104)$$

$$\hat{\mathbf{q}}_{B/A} = \hat{\mathbf{q}}_{B/A} + \frac{\epsilon}{2} \hat{\mathbf{q}}_{B/A} \otimes \hat{\mathbf{R}}_A \quad (105)$$

Substituting Eqs. (104) and (105) to (103) and using $(\mathbf{q}_1 \otimes \mathbf{q}_2)^* = \mathbf{q}_2^* \otimes \mathbf{q}_1^*$ yields:

$$\delta \check{\mathbf{q}} = \left(\mathbf{q}_{B/A} + \frac{\epsilon}{2} \mathbf{q}_{B/A} \otimes \mathbf{R}_A \right) \left(\hat{\mathbf{q}}_{B/A}^* + \frac{\epsilon}{2} \hat{\mathbf{R}}_A^* \otimes \hat{\mathbf{q}}_{B/A}^* \right) = \quad (106)$$

$$= \mathbf{q}_{B/A} \otimes \hat{\mathbf{q}}_{B/A}^* + \frac{\epsilon}{2} \mathbf{q}_{B/A} \otimes \mathbf{R}_A \otimes \hat{\mathbf{q}}_{B/A}^* + \frac{\epsilon}{2} \mathbf{q}_{B/A} \hat{\mathbf{R}}_A^* \otimes \hat{\mathbf{q}}_{B/A}^*$$

The quaternion error $\mathbf{q}_{B/A} \otimes \hat{\mathbf{q}}_{B/A}^*$ is $\delta \mathbf{q}_R$, and $\hat{\mathbf{R}}_A^* = -\hat{\mathbf{R}}_A$, then the error expression is simplified:

$$\delta \check{\mathbf{q}} = \delta \mathbf{q}_R + \frac{\epsilon}{2} \mathbf{q}_{B/A} \otimes \left(\mathbf{R}_A - \hat{\mathbf{R}}_A \right) \otimes \hat{\mathbf{q}}_{B/A}^* = \quad (107)$$

$$= \delta \mathbf{q}_R + \frac{\epsilon}{2} \mathbf{q}_{B/A} \otimes \hat{\mathbf{q}}_{B/A}^* \otimes \hat{\mathbf{q}}_{B/A} \otimes \left(\mathbf{R}_A - \hat{\mathbf{R}}_A \right) \otimes \hat{\mathbf{q}}_{B/A}^* =$$

$$= \delta \mathbf{q}_R + \frac{\epsilon}{2} \delta \mathbf{q}_R \otimes \delta \mathbf{R}_B$$

With a first order approximation $\delta \mathbf{q}_R \otimes \delta \mathbf{R}_B \approx \delta \mathbf{R}_B$, the dual quaternion error becomes:

$$\delta \check{\mathbf{q}} \approx \delta \mathbf{q}_R + \frac{\epsilon}{2} \delta \mathbf{R}_B \quad (108)$$

which shows a useful result, since the dual part of the error is approximately the position error itself, however, expressed in \mathcal{F}_B .

Error kinematics

We differentiate the error, Eq. (107), with respect to time, and since the real part of the dual quaternion error derivative is essentially the same as was derived for the relative attitude quaternion, Eq. (51), therefore it is not repeated again. The dual part (all terms with ϵ) derivative is as follows:

$$\delta \dot{\mathbf{q}}_d = \left(\hat{\mathbf{V}}_B + \delta \mathbf{V}_B \right) \otimes \delta \mathbf{q}_R + \frac{1}{4} \left(\hat{\omega}_{B/A}^B + \delta \omega - \mathbf{C}(\hat{\mathbf{q}}_{B/A}) \hat{\omega}_{A/I}^A \times \delta \boldsymbol{\vartheta}_R - \mathbf{C}(\hat{\mathbf{q}}_{B/A}) \delta \omega_A \right) \otimes \delta \mathbf{R}_B + \quad (109)$$

$$- \frac{1}{2} \delta \mathbf{q}_R \otimes \hat{\mathbf{V}}_B - \frac{1}{4} \delta \mathbf{R}_B \otimes \hat{\omega}_{B/A}^B$$

Neglecting the second order terms yields:

$$\delta \dot{\mathbf{q}}_d = \frac{1}{2} \hat{\mathbf{V}}_B \otimes \delta \mathbf{q}_R + \frac{1}{2} \delta \mathbf{V}_B + \frac{1}{4} \hat{\omega}_{B/A}^B \otimes \delta \mathbf{R}_B - \frac{1}{2} \delta \mathbf{q}_R \otimes \hat{\mathbf{V}}_B - \frac{1}{4} \delta \mathbf{R}_B \otimes \hat{\omega}_{B/A}^B = \quad (110)$$

$$\begin{aligned}
&= - \left[\hat{\mathbf{V}}_B \times \right] \delta \mathbf{q}_R + \frac{1}{2} \delta \mathbf{V}_B - \frac{1}{2} \left[\hat{\boldsymbol{\omega}}_{B/A}^B \times \right] \delta \mathbf{R}_B = \\
&= - \left[\mathbf{C}(\hat{\mathbf{q}}_{B/A}) \hat{\mathbf{V}}_A \times \right] \delta \mathbf{q}_R + \frac{1}{2} \mathbf{C}(\hat{\mathbf{q}}_{B/A}) \delta \mathbf{V}_A - \left[\hat{\boldsymbol{\omega}}_{B/A}^B \times \right] \delta \mathbf{q}_d
\end{aligned}$$

Since the dual part error can be represented by $\delta \mathbf{R}_B$, Eq. (108), and the real part by a small rotation vector, $\boldsymbol{\vartheta}_R$, the whole dual quaternion error is written as:

$$\delta \tilde{\mathbf{q}} = \begin{pmatrix} \frac{1}{2} \delta \boldsymbol{\vartheta}_R \\ 1 \\ \frac{1}{2} \delta \mathbf{R}_B \\ 0 \end{pmatrix} \quad (111)$$

Therefore, we reduce the pose vector dimension from eight to six:

$$\delta \tilde{\mathbf{q}}_{1:6} = \begin{pmatrix} \delta \boldsymbol{\vartheta}_R \\ \delta \mathbf{R}_B \end{pmatrix} \quad (112)$$

Then, Eq. (110) becomes:

$$\delta \dot{\mathbf{R}}_B = - \left[\mathbf{C}(\hat{\mathbf{q}}_{B/A}) \hat{\mathbf{V}}_A \times \right] \delta \boldsymbol{\vartheta}_R + \mathbf{C}(\hat{\mathbf{q}}_{B/A}) \delta \mathbf{V}_A - \left[\hat{\boldsymbol{\omega}}_{B/A}^B \times \right] \delta \mathbf{R}_B \quad (113)$$

As opposed to the QV filter the derivative of $\delta \mathbf{R}_A$ is not $\delta \mathbf{V}_A$.

Full State-error Vector

Since the dual quaternion error is now expressed as a six-dimensional number, the full state-error vector has the same size as in QVEKF case:

$$\delta \mathbf{X}_{18} = \begin{pmatrix} \delta \boldsymbol{\vartheta}_R \\ \delta \mathbf{R}_B \\ \delta \mathbf{V}_A \\ \delta \boldsymbol{\vartheta}_I \\ \delta \boldsymbol{\mu} \\ \delta \boldsymbol{\omega}_A \end{pmatrix} \quad (114)$$

Full Linear-perturbation Model

The linear perturbation model would basically be different only by the position error kinematics, expressed in Eq. (113). However, since the error, $\delta \mathbf{R}_B$ is now expressed in \mathcal{F}_B instead of \mathcal{F}_A , as

it was in QVEKF, the Jacobian for the velocity has to be modified accordingly. Without loss of generality, we can write:

$$\frac{\partial \mathbf{f}}{\partial \mathbf{R}_B} = \frac{\partial \mathbf{f}}{\partial \mathbf{R}_A} \frac{\partial \mathbf{R}_A}{\partial \mathbf{R}_B} \quad (115)$$

where $\frac{\partial \mathbf{R}_A}{\partial \mathbf{R}_B}$ is simply an attitude matrix:

$$\frac{\partial \mathbf{R}_A}{\partial \mathbf{R}_B} = \mathbf{C}_{A/B} = \mathbf{C}(\mathbf{q}_{B/A})^T \quad (116)$$

As a result, the velocity Jacobian is expressed as follows:

$$\frac{\partial \mathbf{V}_A}{\partial \mathbf{R}_B} = \frac{\partial \mathbf{V}_A}{\partial \mathbf{R}_A} \mathbf{C}(\mathbf{q}_{B/A})^T \quad (117)$$

$$\mathbf{F} = \begin{bmatrix} -[\hat{\omega}_{B/I}^B \times] & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -\mathbf{I}_{3 \times 3} & -\mathbf{C}(\hat{\mathbf{q}}_{B/A}) \\ -[\mathbf{C}(\hat{\mathbf{q}}_{B/A}) \hat{\mathbf{V}}_A \times] & -[\hat{\omega}_{B/A}^B \times] & \mathbf{C}(\hat{\mathbf{q}}_{B/A}) & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \left. \frac{d\mathbf{f}(\mathbf{X})_{7:9}}{d\mathbf{R}_A} \mathbf{C}(\hat{\mathbf{q}}_{B/A})^T \right|_{\mathbf{X}=\hat{\mathbf{X}}} & \left. \frac{d\mathbf{f}(\mathbf{X})_{7:9}}{d\mathbf{V}_A} \right|_{\mathbf{X}=\hat{\mathbf{X}}} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \left. \frac{d\mathbf{f}(\mathbf{X})_{7:9}}{d\omega_{A/I}^A} \right|_{\mathbf{X}=\hat{\mathbf{X}}} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -[\hat{\omega}_{B/I}^B \times] & -\mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix} \quad (118)$$

Since the state vector has changed, the noise mapping matrix has changed accordingly:

$$\mathbf{G} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & -\mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & -\mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix} \quad (119)$$

E. Measurement Equations

Since the position error in DQEKF is represented by $\delta \mathbf{R}_B$, and it was $\delta \mathbf{R}_A$ for QVEKF, then the measurement matrices are essentially the same.

Star tracker

The star tracker matrix is very alike to the QVEKF one, and it reads as follows:

$$\mathbf{H}_{ST} = \begin{bmatrix} \mathbf{0}_{3 \times 9} & \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 6} \end{bmatrix} \quad (120)$$

Note that the order of the state variables has changed, so the matrix has changed accordingly, see Eq. (114).

Navigation Camera

For the Jacobians with respect to the position vector, the same method is applied as in Eq. (116):

$$\frac{\partial \mathbf{h}}{\partial \mathbf{R}_B} = \frac{\partial \mathbf{h}}{\partial \mathbf{R}_A} \mathbf{C}(\mathbf{q}_{B/A})^T \quad (121)$$

This allows us to use the same measurement matrices as for QVEKF by multiplying them with $\mathbf{C}_{B/A}^T$. The measurement matrix for the navigation camera is then:

$$\mathbf{H}_{NAVCAM} = \left[\left[\mathbf{C}_{B/A}(\hat{\mathbf{q}}_{B/A}) \frac{\mathbf{R}_{LM,i} - \hat{\mathbf{R}}_A}{\|\mathbf{R}_{LM,i} - \hat{\mathbf{R}}_A\|} \times \right] \frac{d\mathbf{h}(\mathbf{X})}{d\mathbf{R}_A} \mathbf{C}(\mathbf{q}_{B/A})^T \mathbf{0}_{3 \times 12} \right] \Big|_{\mathbf{X}=\hat{\mathbf{X}}} \quad (122)$$

Laser Ranger

Similarly to the NAVCAM measurement matrix, the laser ranger one is obtained:

$$\mathbf{H}_{LR} = \left[\mathbf{0}_{1 \times 3} \quad \frac{d\mathbf{h}(\mathbf{X})}{d\mathbf{R}_A} \mathbf{C}(\mathbf{q}_{B/A})^T \mathbf{0}_{1 \times 12} \right] \Big|_{\mathbf{X}=\hat{\mathbf{X}}} \quad (123)$$

VII. Simulation and Results

This section presents the results of the nominal EKF (QVEKF) and the dual quaternion counterpart (DQEKF). The spacecraft size and mass parameters are taken from Rosetta mission. A 'polar' orbit around asteroid Kleopatra is simulated for 10,000 s with the initial position and velocity:

$$\mathbf{R}_A = \begin{pmatrix} 0 \\ 0 \\ 200 \end{pmatrix} \text{ km}, \quad \mathbf{V}_A = \begin{pmatrix} 0 \\ -35.35 \\ 0 \end{pmatrix} \text{ m/s} \quad (124)$$

The initial inertial and relative attitudes are the same:

$$\mathbf{q}_{B/I} = \mathbf{q}_{B/A} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (125)$$

which means that at $t = 0$ frames \mathcal{F}_I and \mathcal{F}_A coincide. The asteroid is spinning around its Z-axis at the rate of $\omega_{A/I} = 3.241 \times 10^{-4}$ rad/s and the initial spacecraft angular velocity is:

$$\boldsymbol{\omega}_{B/I}^B = \begin{pmatrix} 1.711 \times 10^{-4} \\ 0 \\ 0 \end{pmatrix} \text{ rad/s} \quad (126)$$

The trajectory is shown in Figure 8, where the light-blue pyramids show the FOV pyramids of the spacecraft.

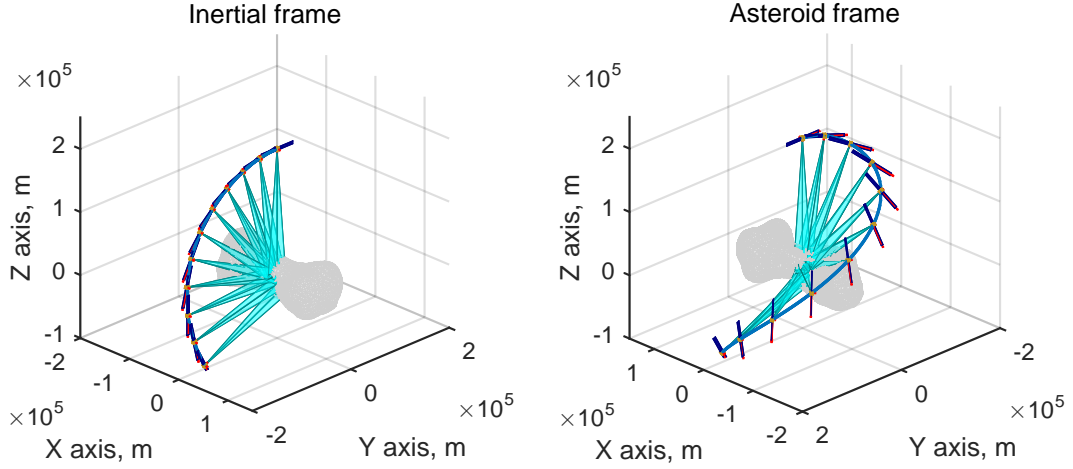


Fig. 8: Simulated trajectory around the Kleopatra asteroid (the spacecraft is not in scale)

The initial angular velocity of the SC is chosen to match the mean motion of its orbit, which in an ideal case would mean that it would always point nadir. However, in the current spacecraft model neither position nor attitude control is considered. This complicates the navigation, because

the navigation camera experiences unfavourable pointing relative to the asteroid. As a result, due to perturbations it drifts from its nominal trajectory. Figure 9 shows the evolution of the number of landmarks in the FOV per each frame. The number varies from 3 to 20, which will give significantly different situations for the filter.

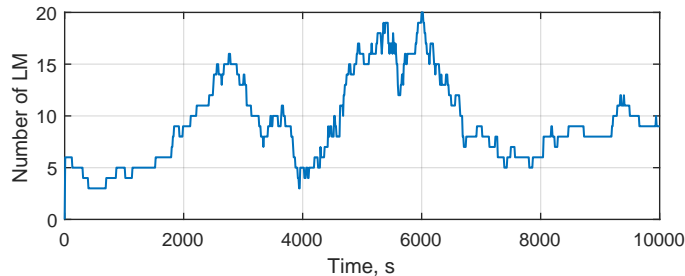


Fig. 9: Number of landmarks, seen by the navigation camera per frame

Target for Laser-Ranger Selection

When the landmarks in the FOV are identified, one of them is chosen for the range measurement. The choice can be done randomly, however, we recall that the error in measured distance depends on the angle between the pointing vector and the local (landmark) normal vector, Fig. 6. The landmark-normal vector is included in the landmark map database, thus assumed to be available on-board the spacecraft. With this knowledge we can choose a landmark with the smallest angle and thus reduce the possible distance errors.

Laser-Ranger Errors

Until now, we only discussed the laser ranger errors, but did not characterise them. The Kalman filter algorithm needs to know the measurement covariance R_{LR} , however the errors are state and asteroid surface dependent, so there is no analytical function to express them. Figure 10 gives the laser-ranger errors as a function of the angle between the landmark normal and the laser pointing vector. There is a clear correlation between the angle and the distance error, and a few cases can be distinguished. The first is between 0° to 20° , and the distance error has variance of $\approx 25 \text{ m}^2$. The second has an angle from 20° to 40° and the variance of $\approx 169 \text{ m}^2$. The third has an angle from 40° to 60° and the variance of $\approx 900 \text{ m}^2$, and the last one is 60° and above, and the error variance

$\approx 2500 \text{ m}^2$.

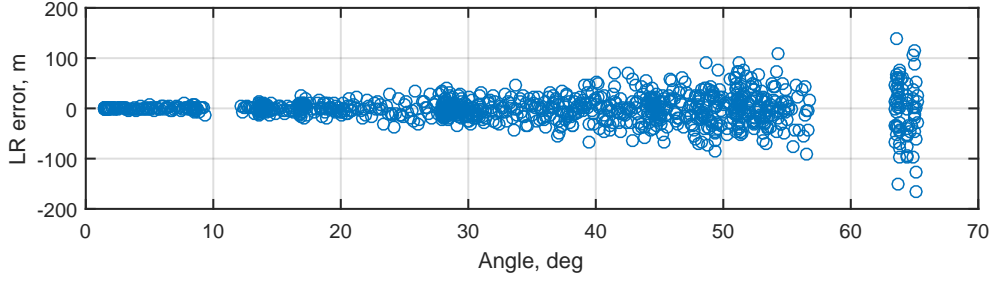


Fig. 10: Laser-ranger angles between the pointing and landmark normal vectors, and corresponding distance errors

Since we defined that the landmark map has normal vectors associated to each landmark, we can use this property in the filter on-board the spacecraft. For each aforementioned case we define the laser ranger covariance as:

$$R_{LR1} = 25 \text{ m}^2, \quad R_{LR2} = 169 \text{ m}^2, \quad R_{LR3} = 900 \text{ m}^2 \quad \text{and} \quad R_{LR4} = 2500 \text{ m}^2 \quad (127)$$

and switch them dynamically, according to the laser incidence angle.

A. Filter Initialisation

Both filters are initialised with the same state vector $\hat{\mathbf{X}}_{0/0}$, error covariance matrix $\mathbf{P}_{0/0}$, process covariance matrix \mathbf{Q} and measurement covariance matrices \mathbf{R}_{ST} , \mathbf{R}_{NAVC} and \mathbf{R}_{LR} , only the order of variables is different.

State vector

Position and velocity are initialised with errors of 100 m and 1 m/s on each axis, respectively.

Relative and inertial quaternions are selected as:

$$\hat{\mathbf{q}}_{B/A,0} = \begin{pmatrix} 0.985 \\ -0.099 \\ 0.099 \\ -0.099 \end{pmatrix} \quad \text{and} \quad \hat{\mathbf{q}}_{B/I,0} = \begin{pmatrix} 0.999 \\ -0.001 \\ 0.001 \\ -0.001 \end{pmatrix} \quad (128)$$

which represents $\Phi_R = 19.6^\circ$ and $\Phi_I = 1.98^\circ$ attitude errors ($2 \cos^{-1} \delta q_4$), respectively. The gyroscope drift is initialised with zeros and the asteroid rate is chosen such that it would represent 10% error for the Z-axis and small errors (2×10^{-6} rad/s) for X- and Y-axis:

$$\hat{\omega}_{A/I,0}^A = \begin{pmatrix} 0.02 \\ 0.02 \\ 3.565 \end{pmatrix} \times 10^{-4} \text{ rad/s} \quad (129)$$

Covariance Matrices

The covariance matrices are the main parameters for tuning the filter. Since it is not a linear Kalman filter, no offline analysis can be done and each parameter has to be tuned empirically. So, the state error covariance matrix is initialised as:

$$\mathbf{P}_{0/0} = \begin{bmatrix} 10^7 \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & 10 \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & 10^{-1} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & 10^{-1} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & 10^{-10} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & 10^{-8} \mathbf{I}_{3 \times 3} \end{bmatrix} \quad (130)$$

The process noise covariance matrix after tuning was set to:

$$\mathbf{Q} = \begin{bmatrix} \sigma_g^2 \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \sigma_v^2 \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \sigma_u^2 \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \sigma_a^2 \mathbf{I}_{3 \times 3} \end{bmatrix} \quad (131)$$

where $\sigma_g = 0.005 \text{ m/s}^2$ denotes the uncertainty in the gravity field, $\sigma_v = 5.8 \times 10^{-7} \text{ rad/s}^{1/2}$ and $\sigma_u = 5.8 \times 10^{-8} \text{ rad/s}^{3/2}$ denote the gyroscope noises standard deviations, and $\sigma_a = 3.2^{-8} \text{ rad/s}^{1/2}$ denotes the uncertainty in the asteroid rate, which is added in the filter model for the sake of

numerical performances. The star tracker has the measurement covariance matrix:

$$\mathbf{R}_{ST} = 0.1 \times \begin{bmatrix} 4,76 \times 10^{-8} & 0 & 0 \\ 0 & 5.88 \times 10^{-10} & 0 \\ 0 & 0 & 5.88 \times 10^{-10} \end{bmatrix} \quad (132)$$

The covariance matrix of the navigation camera is:

$$\mathbf{R}_{NAVC} = \begin{bmatrix} 1.69 \times 10^{-11} & 0 & 0 \\ 0 & 1.69 \times 10^{-11} & 0 \\ 0 & 0 & 1.69 \times 10^{-13} \end{bmatrix} \quad (133)$$

The Z-axis has a far smaller variance set, because the focal length is constant and the noise comes only from the norm of the vector, Eq. (33). Finally, the laser-ranger covariance is given by Eq. (127).

B. Results

This subsection presents the results of the QVEKF and DQEKF. Only the magnitude of vectorial errors between the real and estimated values are presented, together with $3\text{-}\sigma$ boundaries. The filter is run with a 0.1 s time step (the sampling time of the gyroscope), the star tracker is set to 1 s and the navigation camera (also the laser ranger) to 10-s sampling times. Let us start with the estimates of the gyro drift and the asteroid rate, since they affect the process equation (39). Figure 11 shows the estimate errors of the drift, $\boldsymbol{\mu}$ (top), and the the asteroid rate, $\boldsymbol{\omega}_{A/I}^A$ (bottom).

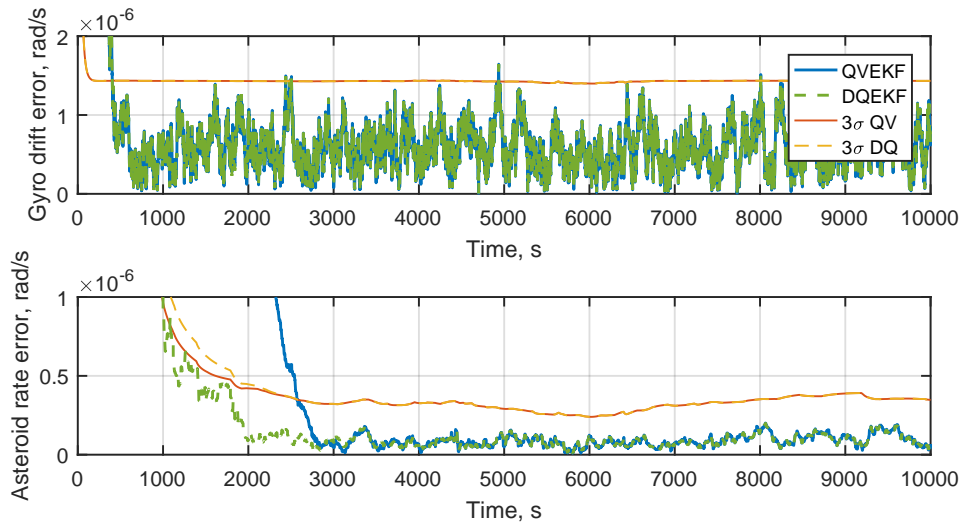


Fig. 11: Gyroscope drift and asteroid rate errors

Surprisingly, both filters behave identically in steady state. The gyroscope-drift estimates show a fast transient of about 500 s and their steady-state regime is not fluctuating in time. These estimates are indeed mainly influenced by star-tracker measurements, which are independent from the relative states and are always available. The errors stay within the $3\text{-}\sigma$ boundaries. The asteroid rates convergence are slower ($\approx 2000 - 3000$ s) and here we start seeing a difference between the filters. DQEKF converges around 1000 s faster than QVEKF. Also, its $3\text{-}\sigma$ boundaries follow the estimate more accurately.

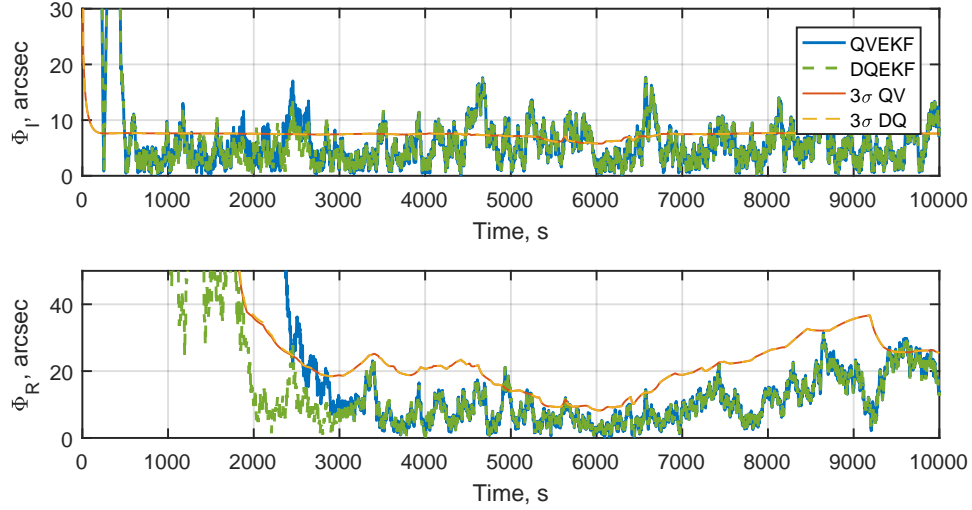


Fig. 12: Inertial and relative attitude errors

Furthermore, inertial and relative attitude estimates are presented in Fig. 12 by the angle $\Phi = 2 \cos^{-1} q_4$. In this case the QVEKF and DQEKF behave the same in the steady state again. Inertial attitude estimates converge in 500 s. After convergence the error does not change and stays within about 10 arcsec $3 - \sigma$ value. This is already a good result, since we recall having the errors of the star tracker with standard deviations 45, 5, 5 arcsec on the X-, Y- and Z-axis, respectively. So the estimates show an error reduction of more than an order of magnitude. The relative attitude estimates are different for the two filters in the transient phase, with DQEKF being faster than QVEKF to converge, 2000 s and 3000 s, respectively. The errors for both filters stay within 20-25 arcsec $3 - \sigma$ values on average, and reach minima of 10 arcsec $3 - \sigma$ at 6000 s (the maximum landmark number in the FOV).

Finally, relative position and velocity estimates are shown in Fig 13. Both filters still have an

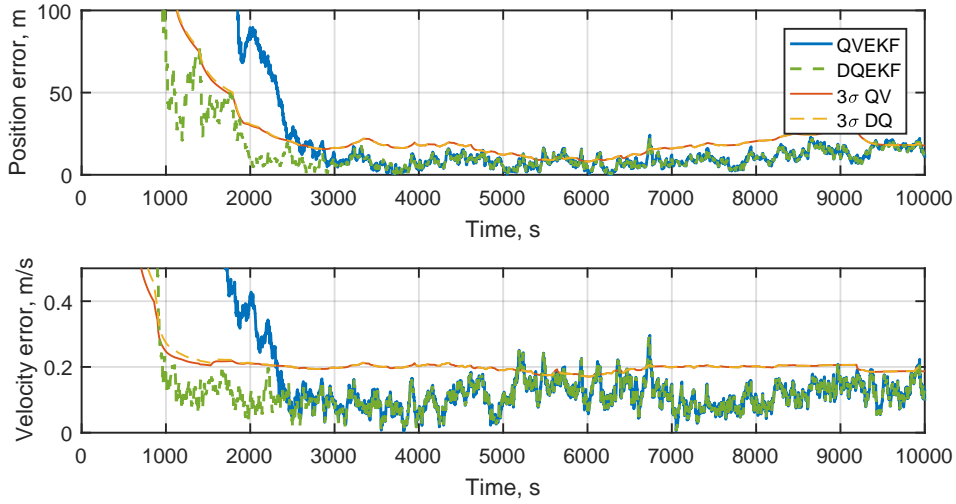


Fig. 13: Position and velocity estimates' errors

identical performance in the steady state, however, as seen in previous figures, DQEKF converges faster. DQEKF position and velocity estimates converge 1000 s and 1500 s, respectively, faster than QVEKF. Position and velocity converge once the asteroid rate converges. After convergence, the estimates stay within the 3σ boundaries, which is around 15 m for position and 0.2 m/s for the velocity. Position estimates reach a minimum at about $t = 6000$ s, which corresponds to the maximum number of the visible landmarks in the FOV, Fig. 9.

VIII. Conclusion

This paper presented a novel dual quaternion EKF applied to asteroid circumnavigation. It also introduces novel methods for landmarks identification within a camera field-of-view, true range and ranging errors determination, and spacecraft gravity-gradient torque Truth modeling. Realistic simulations of the environment based on typical sensing accuracies yielded, in steady-state, 5 m and 0.06 m/s standard deviations for the position and velocity, respectively, and standard deviations of around 3 arcsec and 5 arcsec for the inertial and relative attitude error, respectively. Significant gains in the estimation errors transients are achieved by the DQEKF filter when compared with the QVEKF filter. The settling times of the DQEKF filter errors range from 1000 s to 1500 s less than those of the QVEKF filter: 1000 s for the asteroid rate, the relative attitude, and the relative position and 1500 s for the relative velocity. It is conjectured that the latter stems from the coupling

between position and velocity that exists by definition of the dual part of the dual quaternion error. The similarity in the errors steady states in both filters might be partially explained by noting that both design models feature similar expressions for the linearized position error vector. The novel method in landmarks identification within the navigation camera field of view leverages the gravity field polyhedron model, exploiting the very useful property of the solid angles seen by a point lying inside/outside a polyhedron. The same property was exploited for the novel method in determining whether a landmark lies inside the polyhedron of the camera field of view. Furthermore, a by product of this enhanced realistic method is the distribution of the range error as a function of the pointing laser-finder angle for arbitrary shaped asteroids.

References

- [1] Cheng, A. F., “Near Earth Asteroid Rendezvous: Mission Summary”, *Asteroid III*, University of Arizona Press, Tucson, 2002, p.351-366.
- [2] Kawaguchi, J., Fujiwara, A., Uesugi, T., “Hayabusa - its technology and science accomplishment summary and Haybusa-2”, *Acta Astronautica*, Vol. 62, Issue 10-11, 2008.
- [3] Zuiani, F., Castellini, F., Kielbassa, S., Bielsa, C. and Garcia, J.M., “Rosetta command and monitoring operations for Philae landing”, *Proceedings 25th International Symposium on Space Flight Dynamics*, 2015.
- [4] Pardo, R. de Santayana and Lauer, M., “Optical measurements for Rosetta navigation near the comet”, *Proceedings 25th International Symposium on Space Flight Dynamics*, 2015.
- [5] Yang, H., Y., Vetrignano, M., Vasile, M. and Zhang, W., “Autonomous navigation of spacecraft formation in the proximity of minor bodies”, *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol 230, No. 1, 2016, pp. 189-204.
- [6] Dionne, K., “Improving Autonomous Navigation for Small Body Exploration Using Range Measurements”, AIAA 2009-6106. *AIAA Guidance, Navigation, and Control Conference*, 2009.
- [7] Kim, S. G., Crassidis, J. L., Cheng, Y. and Fosbury, A. M. , “Kalman Filtering for Relative Spacecraft Attitude and Position Estimation”, *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 1, Jan.-Feb. 2007, pp. 133-143.
- [8] Mourikis, A. I., Trawny, N., Roumeliotis, S. I., Johnson, A. E., Ansar, A., and Matthies, L., “Vision-Aided Inertial Navigation for Spacecraft Entry, Descent, and Landing”, *IEEE Transactions on Robotics*,

- Vol. 25, No. 2, April 2009, pp. 264–280.
- [9] Clifford, W. K., “Preliminary sketch of biquaternions”, *Proceedings of the London Mathematical Society*, Vol. 4, 1873, pp. 381-195.
 - [10] Fischer, I. S., *Dual-number methods in kinematics, statics, and dynamics*, Boca Raton, CRC Press, 1999.
 - [11] Jia, Y. B., “Dual quaternion”, Com S 477/577, Fall 2015, Course handouts, 2013. URL <http://web.cs.iastate.edu/~cs577/>. Date accessed: January 30, 2017.
 - [12] Goddard, J. S., and Abidi, M. A. “Pose and motion estimation using dual quaternion-based extended Kalman filtering”, Proc. SPIE 3313, Three-Dimensional Image Capture and Applications, 189 (March 6, 1998); doi:10.1117/12.302453; <http://dx.doi.org/10.1117/12.302453>.
 - [13] Wu, Yuanxin, et al. “Strapdown inertial navigation system algorithms based on dual quaternions.” *IEEE transactions on aerospace and electronic systems* 41.1 (2005): 110-132.
 - [14] Qiao, B., Tang, S., Ma, K. and Liu, Z., “Relative position and attitude estimation of spacecrafts based on dual quaternion for rendezvous and docking”, *Acta Astronautica*, Vol 91, 2013, pp. 237-244.
 - [15] Jun S., Shijie Z., Xiande W., Fengzhi G., Yaen X., ”Relative Status Determination for Spacecraft Relative Motion Based on Dual Quaternion”, *Mathematical Problems in Engineering*, Vol. 2014, No. 5, 2014, pp. 1-7.
 - [16] Filipe, N., Kontitsis, M., and Tsiotras, P., “Extended Kalman filter for spacecraft pose estimation using dual quaternions.” *Journal of Guidance, Control, and Dynamics* 38.9 (2015): 1625-1641.
 - [17] Markley, F. L., “Attitude Error Representations for Kalman filtering”, *Journal of Guidance, Control and Dynamics*, Vol. 26, No. 2, 2003.
 - [18] Markley, F.L. and Crassidis, J.L., *Fundamentals of Spacecraft Attitude Determination and Control*, Vol 2, 2013, ISBN 978-1-4939-0801-1, Springer.
 - [19] Werner, R. and Scheeres, D.J., “Exterior gravitation of a polyhedron derived and compared with harmonic and mascon gravitation representations of asteroid 4769 Castalia”, *Celestial Mechanics and Dynamical Astronomy*, Vol. 65, 1997, pp. 313-344.
 - [20] Scheeres, D. J., et al. “The dynamical environment about asteroid 25143 Itokawa, target of the Hayabusa mission.” AIAA/AAS Astrodynamics Specialist Conference and Exhibit. 2004.
 - [21] Scheeres, D. J., et al. “The actual dynamical environment about Itokawa.” AIAA/AAS Astrodynamics Specialist Conference and Exhibit. 2006.
 - [22] Yang, Yu, and Baoyin, H., “Orbital dynamics in the vicinity of asteroid 216 Kleopatra.” *The Astronomical Journal* 143.3 (2012): 62.

- [23] Wie, B, *Space Vehicle Dynamics and Control*, Second Edition, 2006, ISBN 978-1-56347-953-3, American Institute of Aeronautics and Astronautics, Inc.
- [24] Kumar, K.D. (2008), “Attitude dynamics and control of satellites orbiting rotating asteroids”, *Acta Mechanica*, Vol. 198, pp. 99-118.
- [25] Wang, Y. and Xu, S., “Gravitational orbit-rotation coupling of a rigid satellite around a spheroid planet”, *Journal of Aerospace Engineering*, 27(1):140-150, January 2014, DOI 10.1061/(ASCE)AS.1943-5525.0000222.
- [26] Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D., “Shape distributions”, *ACM Transactions on Graphics*, Vol. 21, No. 4, 2002, pp. 807-832.