Relative Navigation in Asteroid Missions

A Dual Quaternion Approach

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

DOI
10.2514/6.2017-1521

Publication date
2017

Document Version
Accepted author manuscript

Published in
AIAA Guidance, Navigation, and Control Conference, 2017

This is an Accepted Manuscript of an article published in: AIAA Guidance, Navigation, and Control Conference
January 9-13, Grapevine, TX. ISBN: 978-1-62410-450-3
Available online: https://arc.aiaa.org/doi/10.2514/6.2017-1521

Relative Navigation in Asteroid Missions: A Dual Quaternion Approach

B. Razgus* and E. Mooij†
Delft University of Technology, Faculty of Aerospace Engineering,
Kluyverweg 1, 2629 HS Delft, The Netherlands

D. Choukroun‡
Ben-Gurion University of the Negev,
POB 653, 84105, Beer Sheva, Israel

This paper presents an Extended Kalman filter (EKF) for relative position and attitude (pose) estimation in a mission around an asteroid. It compares two different ways of representing the pose: a conventional one (Cartesian coordinates for position and quaternions for attitude) and a novel approach (dual quaternions, which comprise both attitude and position in one eight-dimensional vector). Moreover, this paper presents a 'realistic' modelling for dynamics and hardware simulation for missions around small bodies. It includes a polyhedron gravity field modelling, polyhedron gravity gradient torque, navigation camera and laser ranger measurements. The results of the filters show that it is not only possible to estimate the relative states with high accuracy, but also parameters such as gyroscope drift and asteroid angular rates can be estimated. This, however, can only be achieved when the navigation camera detects landmarks in its field-of-view. Finally, the dual quaternion representation does not give any noticeable advantages over the conventional one; in fact, the two filters are identical in the steady state.

I. Introduction

Asteroids are gaining more attention among space agencies, due to their preserved state since the beginning of the Solar System. In addition, they could be a potential hazard to our planet or oppositely, another source of resources in the future. Thus, the interest is expected to grow even further. As of June 2016 three missions dedicated to small bodies were accomplished. NEAR visited and landed on the asteroid Eros in 2000.[1] JAXA's mission Hayabusa retrieved samples from the Itokawa asteroid,[2] and ESA's mission Rosetta successfully put a lander on a comet in 2014[3] The spacecraft was controlled-crashed in 2016 as part of its end-of-life strategy. All of these missions required precise navigation relative to the asteroid/comet. Earth-based measurements provide accurate results, however, they introduce a delay, which could be too large for proximity manoeuvres, thus autonomy in navigation is required. As a result, most of the missions have navigation cameras,[4] which provide measurements relative to the asteroid. Furthermore it was shown that range measurements could improve the navigation and make it more robust, especially when only a few landmarks are visible.[5,6]

When position and attitude (pose) are required between two non-inertial frames, e.g., spacecraft docking, formation flying, an asteroid mapping/landing, it is common practice to develop navigation algorithms based on relative pose-dynamics modelling, as opposed to inertial pose-dynamics modelling. The case of two spacecraft is extensively covered by Kim et al.[7] In that work, an extended Kalman filter (EKF) estimates the pose vector modelled as Cartesian relative coordinates augmented with the relative quaternion of rotation.

*MSc Graduate, section Astrodynamics and Space Missions, bronius.razgus@gmail.com.
†Assistant Professor, section Astrodynamics and Space Missions, e.mooij@tudelft.nl, Associate Fellow AIAA.
‡Senior Lecturer, Mechanical Engineering Dept., Senior Member AIAA

An alternate modelling approach that has recently spawned growing attention, models the pose via a dual quaternion, which represents position and attitude in an eight-dimensional vector. Research shows that it could result in higher accuracy and faster convergence.[8,9] As of June 2016, there have been no attempts to implement dual quaternions for navigation in an asteroid mission, which is where this paper comes in. The main contribution of this paper is the investigation of dual quaternions as an efficient representation in relative-navigation algorithms for asteroid missions.

The main focus of the paper is the simulation of the environment in the vicinity of an asteroid and the appropriate relative-navigation-filter development. The idea is to compare two different position and attitude representations. The first, named as conventional, is a vector (Cartesian coordinates) for position and a quaternion for attitude, and a new approach based on a dual quaternion representation. The target asteroid for simulations is asteroid Kleopatra, which is a dog-bone shaped body, resulting in a highly-perturbed gravity field. Another contribution of this work is the introduction of a novel approach for simulation of a navigation camera and laser-ranger measurements.

Section II gives an overview of the way the dynamics is simulated. Reference frames, the translational and rotational motion, perturbations, and relative states are discussed. Section III presents the hardware (sensor) models, which includes novel modelling for the navigation camera and the laser-ranger measurements. In Sec. IV the nominal Extended Kalman filter is developed, based on the conventional pose representation. Sections V and VI introduce dual quaternions and show the development of the corresponding EKF. In Sec. VII the results are presented and discussed, and the paper is finished with conclusions and recommendations 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.

II.A. Reference frames

Three reference frames are used throughout this paper: inertial, asteroid and body, with indices $\mathcal{F}_{\mathcal{I}}$, $\mathcal{F}_{\mathcal{A}}$ and $\mathcal{F}_{\mathcal{B}}$ 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.

II.B. Translational Motion

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

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

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

where $\boldsymbol{R}_I$, $\boldsymbol{V}_I$ and $\boldsymbol{g_I}$ are a position, velocity and gravity field vector in the $\mathcal{F}_{\mathcal{I}}$, respectively. $\boldsymbol{a}_{dist}$ 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.[10] The gradient of the potential is given as a sum over the edges and faces:

$$\boldsymbol{g}_A = \nabla U = -G\rho \sum_{e \in edges} \boldsymbol{E}_e \boldsymbol{r}_e L_e + G\rho \sum_{f \in faces} \boldsymbol{F}_f \boldsymbol{r}_f \omega_f \tag{3}$$

where G is the gravitational constant, ρ is the mean density of the asteroid (assumed to be constant), $\boldsymbol{r}_e$ and $\boldsymbol{r}_f$ are the distances from a field point (in this case the spacecraft) to an edge and a face respectively, $\boldsymbol{E}_e$ is a dyadic matrix of an edge, $\boldsymbol{F}_f$ 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.[10] 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.[11] The simulated spacecraft has solar panels with a reflectivity of $\epsilon_{SP} = 0.21$ and the body with $\epsilon_B = 0.5$.

II.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.[11] 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:

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

and the following holds:

$$\boldsymbol{q}_1 \otimes \boldsymbol{q}_2 = \boldsymbol{q}_2 \odot \boldsymbol{q}_1 \tag{5}$$

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). For example, the relative attitude between a spacecraft and an asteroid would be expressed as:

$$\boldsymbol{q}_{B/A} = \boldsymbol{q}_{B/I} \otimes \boldsymbol{q}_{I/A} \tag{6}$$

Kinematics and Dynamics

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

$$\dot{\boldsymbol{q}}_{B/I} = \frac{1}{2}\boldsymbol{\omega}_{B/I}^B \otimes \boldsymbol{q}_{B/I} \tag{7}$$

where $\boldsymbol{\omega}_{B/I}^B$ is the angular velocity of a body with respect to inertial frame, expressed in body frame. 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:

$$\dot{\boldsymbol{q}}_{B/I} = \frac{1}{2}\boldsymbol{\Omega}\left(\boldsymbol{\omega}_{B/I}^B\right)\boldsymbol{q}_{B/I}, \quad \text{where} \quad \boldsymbol{\Omega}\left(\boldsymbol{\omega}_{B/I}^B\right) = \begin{bmatrix} -[\boldsymbol{\omega}\times] & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix} \tag{8}$$

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

$$\dot{\boldsymbol{\omega}}_{B/I}^B = \boldsymbol{I}^{-1}\left(-\boldsymbol{\omega}_{B/I}^B \times \boldsymbol{I}\boldsymbol{\omega}_{B/I}^B + \boldsymbol{T}_{dist}\right) \tag{9}$$

where $\boldsymbol{I}$ is the inertia tensor and $\boldsymbol{T}_{dist}$ is a sum of disturbance torques acting on a spacecraft, which will be detailed now.

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:[11,12]

$$\boldsymbol{T}_{gg} = 3\frac{\mu}{R^3}\boldsymbol{a}_3 \times \boldsymbol{I}\boldsymbol{a}_3 \tag{10}$$

where μ is the gravitational parameter, R is the distance to the attracting body and $\boldsymbol{a}_3$ is the third component of Local Vertical Local Horizontal reference frame, which has nadir direction. Equation (10) 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. If one assumes the SC to be made of N point-masses, the torque is then given by:

$$\boldsymbol{T}_{gg,B} = \sum_{i=1}^{N} \boldsymbol{R}_{B,i} \times m_i \boldsymbol{g}_{B,i} \tag{11}$$

where $\boldsymbol{R}_{B,i}$ is the distance of the point-mass from the centre of mass of the SC, $\boldsymbol{g}_{B,i}$ 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.

II.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:

$$\boldsymbol{\omega}^{A}_{A/I} = \text{const} \tag{12}$$

This leaves only the attitude of the asteroid to be simulated, which, if expressed in quaternions, is given similarly to the SC kinematics, Eq. (8):

$$\dot{\boldsymbol{q}}_{A/I} = \frac{1}{2}\boldsymbol{\Omega}\left(\boldsymbol{\omega}^{A}_{A/I}\right)\boldsymbol{q}_{A/I} \tag{13}$$

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.

II.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. The relative spacecraft states are position, $\boldsymbol{R}_A$, and velocity, $\boldsymbol{V}_A$, expressed in frame $\mathcal{F}_\mathcal{A}$, attitude, $\boldsymbol{q}_{B/A}$, and angular velocity $\boldsymbol{\omega}^{B}_{B/A}$. Let us start with the position and velocity. Relative position is simply obtained by frame transformation:

$$\boldsymbol{R}_A = \boldsymbol{C}_{A/I}\boldsymbol{R}_I \tag{14}$$

where $\boldsymbol{C}_{A/I}$ is expressed with $\boldsymbol{q}_{A/I}$. Differentiating Eq. (14), gives the relative velocity:

$$\boldsymbol{V}_A = \boldsymbol{C}_{A/I}\boldsymbol{V}_I - \boldsymbol{\omega}^{A}_{A/I} \times \boldsymbol{R}_A \tag{15}$$

The relative attitude, expressed in quaternions is:

$$\boldsymbol{q}_{B/A} = \boldsymbol{q}_{B/I} \otimes \boldsymbol{q}_{A/I}^{-1} \tag{16}$$

and the angular rate:

$$\boldsymbol{\omega}_{B/A}^{B} = \boldsymbol{\omega}_{B/I}^{B} - \boldsymbol{C}_{B/A}\boldsymbol{\omega}_{A/I}^{A} \tag{17}$$

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

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.

III.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, $\tilde{\boldsymbol{\omega}}_{B/I}^{B}$ (the tilde symbol denotes measured values), of a spacecraft can be written as:

$$\tilde{\boldsymbol{\omega}}_{B/I}^{B}(t) = \boldsymbol{\omega}_{B/I}^{B}(t) + \boldsymbol{\mu}(t) + \boldsymbol{\eta}_v(t) \tag{18}$$

$$\dot{\boldsymbol{\mu}}(t) = \boldsymbol{\eta}_u(t) \tag{19}$$

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:[11]

$$\tilde{\boldsymbol{\omega}}_{k+1} = \omega_{k+1} + \frac{1}{2}\left(\boldsymbol{\mu}_{k+1} + \boldsymbol{\mu}_k\right) + \left(\frac{\sigma_v^2}{\Delta t} + \frac{1}{12}\sigma_u^2\Delta t\right)^{1/2}\boldsymbol{\eta}_{v_k} \tag{20}$$

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k + \sigma_u \Delta t^{1/2}\boldsymbol{\eta}_{u_k} \tag{21}$$

III.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 $\tilde{\boldsymbol{q}}_{B/I}$ directly, having a simple estimator inside the tracker itself. This will be the way the star tracker is simulated. The real quaternion $\boldsymbol{q}_{B/I}$ will be affected by a noise quaternion $\delta\boldsymbol{q}$.

$$\tilde{\boldsymbol{q}}_{B/I} = \delta\boldsymbol{q}^{-1} \otimes \boldsymbol{q}_{B/I} \tag{22}$$

where $\delta\boldsymbol{q}$ is approximated by a small-angle rotation vector:

$$\delta\boldsymbol{q} = \begin{pmatrix} \phi/2 \\ \theta/2 \\ \psi/2 \\ 1 \end{pmatrix} \tag{23}$$

The noise angles ϕ, θ, ψ are simulated as zero-mean, white noise vectors with known variance. The axis, corresponding to the boresight axis has usually a higher variance of the noise. Note that the noise quaternion is no longer a unit quaternion, so it has to be normalised. Also, no scale-factor or misalignment errors are simulated, assuming perfect knowledge of them. The inclusion of this kind of instrument errors remains to be done as future work.

III.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 identification part is left as future work; and it is thus assumed that the position of the landmarks on the asteroid is known with infinite accuracy (the landmark map is perfect).

Generating Landmarks

In general, a landmark can be a crater, bolder, 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[13] in his shape-recognition paper. The random point on a triangle equation is given as:

$$\boldsymbol{P} = (1 - \sqrt{r_1})\boldsymbol{A} + \sqrt{r_1}(1 - r_2)\boldsymbol{B} + \sqrt{r_1}r_2\boldsymbol{C} \tag{24}$$

where $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ 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 1. 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.

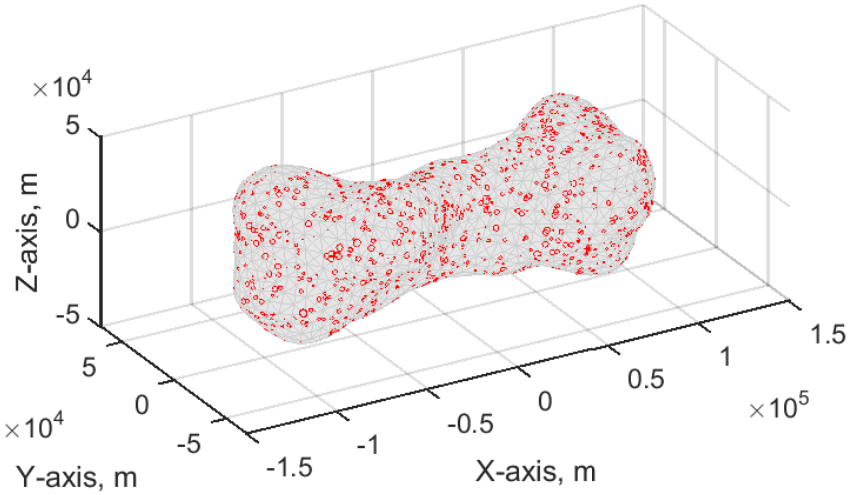

Figure 1: Asteroid Kleopatra with 2000 landmarks

Each landmark gets an identifier and has its coordinates and normal vector associated to it in $\mathcal{F}_{\mathcal{A}}$.

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[a]. 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 $\boldsymbol{\omega}_f$ 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.

[a] ftp://ssols01.esac.esa.int/pub/data/SPICE/ROSETTA/kernels/ik/ROS_NAVCAM_V01.TI, date accessed: 02-06-16

Pixel Coordinates

When landmarks in the FOV are identified, their coordinates have to be expressed in the $\mathcal{F}_{\mathcal{B}}$ and then projected on the camera plane. Figure 2 shows the geometry of the process.

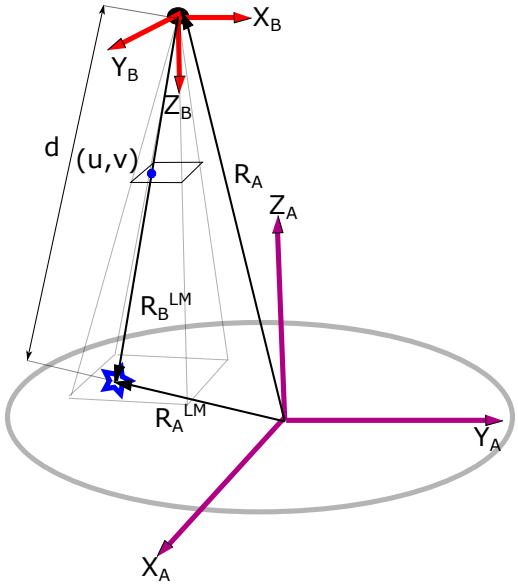

Figure 2: A landmark projected on a sensor plane

So, first the landmark coordinates are transformed:

$$\boldsymbol{R}_B^{LM} = \boldsymbol{C}_{B/A}\left(\boldsymbol{R}_A^{LM} - \boldsymbol{R}_A\right) \tag{25}$$

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

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{1}{p}\frac{f}{Z_B^{LM}}\begin{pmatrix} X_B^{LM} \\ Y_B^{LM} \end{pmatrix} \tag{26}$$

where u, v are coordinates in pixels, p is the pixel size, which in this case (Rosetta mission) is 13 μm. Furthermore, since the actual landmark-recognition and centre-finding algorithm is not simulated, errors/noise are added to the coordinates:

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \end{pmatrix} = \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} \delta u \\ \delta v \end{pmatrix} \tag{27}$$

where δu, δv are zero-mean, white noises with known variance, expressed in (sub-)pixels.

Measurement Vector

To implement Eq. (27) in the navigation filter, it is more convenient to use unit vectors to the landmarks and write them in the form:

$$\tilde{\boldsymbol{b}} = \boldsymbol{C}\left(\boldsymbol{q}\right)\boldsymbol{r} \tag{28}$$

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

$$\tilde{\boldsymbol{b}} = \frac{1}{\sqrt{(p\tilde{u})^2 + (p\tilde{v})^2 + f^2}}\begin{pmatrix} p\tilde{u} \\ p\tilde{v} \\ f \end{pmatrix} \tag{29}$$

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

$$\boldsymbol{r} = \frac{\boldsymbol{R}_{LM} - \boldsymbol{R}_A}{\|\boldsymbol{R}_{LM} - \boldsymbol{R}_A\|} \tag{30}$$

III.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. 2:

$$d = \left\| \boldsymbol{R}_A^{LM} - \boldsymbol{R}_A \right\| \tag{31}$$

However, as mentioned before, if a pointing error is introduced, the measured distance can change significantly. Figure 3 (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. (31) 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.

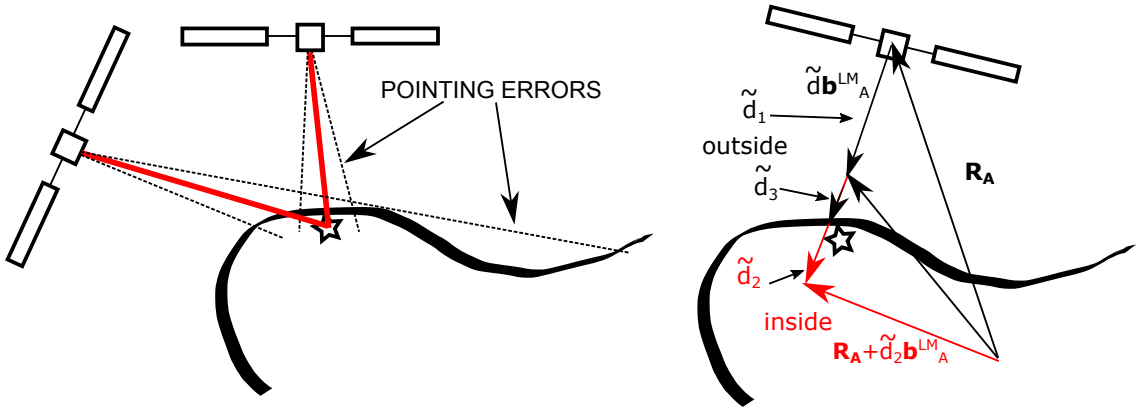

Figure 3: Laser ranger errors (left) and measurement simulation process (right)

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

$$\boldsymbol{b}_B^{LM} = \boldsymbol{R}_y(\tilde{\theta})\boldsymbol{R}_x(\tilde{\phi})\boldsymbol{b}_B^{nom} \tag{32}$$

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

$$\tilde{\phi} = \phi + \delta\phi \qquad \tilde{\theta} = \theta + \delta\theta \tag{33}$$

where $\delta\phi$, $\delta\theta$ are zero-mean, white noise representing the pointing error. This can be seen as the actual gimbal/mirror system, rotating the laser beam to the selected landmark. The actual geometry is depicted in Fig. 4.

If the pointing vector $\boldsymbol{b}_A^{LM}$ (expressed in $\mathcal{F}_\mathcal{A}$) is extended in the same direction by $\tilde{d}$, Figure 3, at some point the vector $\boldsymbol{R}_A + \tilde{d}\boldsymbol{b}_A^{LM}$ will be inside the asteroid (the sum of solid angle ω_f will be 4π). Then, in a similar way the bisection method works, the boundary between inside and outside can be found, which will represent the measured distance $\tilde{d}$. For example, suppose we extend the vector by $\tilde{d}_1$, Fig. 3. The vector $\boldsymbol{R}_A + \tilde{d}_1\boldsymbol{b}_A^{LM}$ is still outside the asteroid, so we extend it again by the same distance, which results in $\tilde{d}_2\boldsymbol{b}_A^{LM}$. Now the vector $\boldsymbol{R}_A + \tilde{d}_2\boldsymbol{b}_A^{LM}$ is inside the asteroid, so we reduce the distance by half of $\tilde{d}_1$ value, which represents the vector $\tilde{d}_3\boldsymbol{b}_A^{LM}$. At this point, the vector $\boldsymbol{R}_A + \tilde{d}_3\boldsymbol{b}_A^{LM}$ 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, $\tilde{d}_i$.

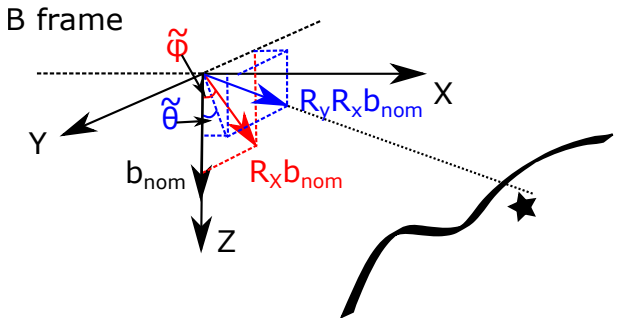

Figure 4: Laser pointing

IV. Nominal EKF Development

There are many different variants of the Extended Kalman filter, especially concerning attitude with quaternions. A multiplicative quaternion error[14] 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.

IV.A. State Vector

First of all, we need to define the state vector for the filter. It should include the relative position, $\boldsymbol{R}_A$, and velocity, $\boldsymbol{V}_A$, relative attitude, $\boldsymbol{q}_{B/A}$, and the angular velocity $\boldsymbol{\omega}^B_{B/A}$. 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. (17). 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, $\boldsymbol{q}_{B/I}$, 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:

$$\boldsymbol{X}_{20} = \begin{pmatrix} \boldsymbol{R}_A \\ \boldsymbol{V}_A \\ \boldsymbol{q}_{B/A} \\ \boldsymbol{q}_{B/I} \\ \boldsymbol{\mu} \\ \boldsymbol{\omega}^A_{A/I} \end{pmatrix} \tag{34}$$

IV.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, $\boldsymbol{w}$. 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:

$$\dot{\boldsymbol{X}} = \boldsymbol{f}(\boldsymbol{X}) + \boldsymbol{G}\boldsymbol{w} = \tag{35}$$

$$= \begin{pmatrix} \boldsymbol{V}_A \\ -\frac{GM}{R_A^3}\boldsymbol{R}_A - 2\boldsymbol{\omega}^A_{A/I} \times \boldsymbol{V}_A - \boldsymbol{\omega}^A_{A/I} \times \boldsymbol{\omega}^A_{A/I} \times \boldsymbol{R}_A \\ \frac{1}{2}\boldsymbol{\Omega}\left(\tilde{\boldsymbol{\omega}}^B_{B/I} - \boldsymbol{\mu} - \boldsymbol{C}_{B/A}(\boldsymbol{q}_{B/A})\boldsymbol{\omega}^A_{A/I}\right)\boldsymbol{q}_{B/A} \\ \frac{1}{2}\boldsymbol{\Omega}(\tilde{\boldsymbol{\omega}}^B_{B/I} - \boldsymbol{\mu})\boldsymbol{q}_{B/I} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{pmatrix} + \begin{bmatrix} \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{4\times3} & -\frac{1}{2}\Xi(\boldsymbol{q}_{B/A}) & \boldsymbol{0}_{4\times3} & \boldsymbol{0}_{4\times3} \\ \boldsymbol{0}_{4\times3} & -\frac{1}{2}\Xi(\boldsymbol{q_{B/I}}) & \boldsymbol{0}_{4\times3} & \boldsymbol{0}_{4\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix} \begin{pmatrix} \boldsymbol{\eta}_g \\ \boldsymbol{\eta}_v \\ \boldsymbol{\eta}_u \\ \boldsymbol{\eta}_a \end{pmatrix}$$

where $\boldsymbol{\eta}_a$ represents the process noise of the asteroid rate, although it is noiseless, Eq. (12). 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:

$$\delta\dot{\boldsymbol{X}} = \boldsymbol{F}(\boldsymbol{X})\delta\boldsymbol{X} + \boldsymbol{G}\boldsymbol{w} \tag{36}$$

where $\boldsymbol{F}(\boldsymbol{X})$ is the gradient matrix of $\boldsymbol{f}(\boldsymbol{X})$, 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.

IV.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:

$$\delta\boldsymbol{q}_I = \boldsymbol{q}_{B/I} \otimes \hat{\boldsymbol{q}}^{-1}_{B/I} \tag{37}$$

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

$$\delta\dot{\boldsymbol{q}}_I = \dot{\boldsymbol{q}}_{B/I} \otimes \hat{\boldsymbol{q}}^{-1}_{B/I} + \boldsymbol{q}_{B/I} \otimes \dot{\hat{\boldsymbol{q}}}^{-1}_{B/I} \tag{38}$$

After some mathematical manipulation, the error kinematics can be written as:

$$\delta\dot{\boldsymbol{q}}_I = \begin{bmatrix} -\left[\hat{\boldsymbol{\omega}}^B_{B/I}\times\right] & 0 \\ \boldsymbol{0}_{1\times3} & 0 \end{bmatrix}\delta\boldsymbol{q}_I - \frac{1}{2}\delta\boldsymbol{\mu} - \frac{1}{2}\boldsymbol{\eta}_v \tag{39}$$

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

$$\delta\boldsymbol{q} \approx \begin{pmatrix} \frac{1}{2}\delta\boldsymbol{\vartheta} \\ 1 \end{pmatrix} \tag{40}$$

Thus, Eq. (39) can be written for the rotation vector:

$$\delta\dot{\boldsymbol{\vartheta}}_I = \left[-\hat{\boldsymbol{\omega}}^B_{B/I}\times\right]\delta\boldsymbol{\vartheta}_I - \delta\boldsymbol{\mu} - \boldsymbol{\eta}_v \tag{41}$$

Relative Attitude

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

$$\delta\boldsymbol{q}_R = \boldsymbol{q}_{B/A} \otimes \hat{\boldsymbol{q}}^{-1}_{B/A} \tag{42}$$

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

$$\delta\dot{\boldsymbol{\vartheta}}_R = -\left[\hat{\boldsymbol{\omega}}^B_{B/I}\times\right]\delta\boldsymbol{\vartheta}_R - \delta\boldsymbol{\mu} - \boldsymbol{C}(\boldsymbol{q}_{B/A})\delta\boldsymbol{\omega}_A - \boldsymbol{\eta}_v \tag{43}$$

where $\delta\boldsymbol{\omega}_A = \boldsymbol{\omega}^A_{A/I} - \hat{\boldsymbol{\omega}}^A_{A/I}$. This is already an interesting result, since the relative attitude error depends only on the inertial angular velocity of the spacecraft.

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:

$$\delta \boldsymbol{X}_{18} = \begin{pmatrix} \delta \boldsymbol{R}_A \\ \delta \boldsymbol{V}_A \\ \delta \boldsymbol{\vartheta}_R \\ \delta \boldsymbol{\vartheta}_I \\ \delta \boldsymbol{\mu} \\ \delta \boldsymbol{\omega}_A \end{pmatrix} \tag{44}$$

Transition Matrix

Combining Eqs. (35), (41) and (43) we can obtain the continuous time state transition matrix, $\boldsymbol{F}$, for perturbations, Eq. (36). It is written as:

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \frac{d\boldsymbol{f}(\boldsymbol{X})_{4:6}}{d\boldsymbol{R}_A}\Big|_{\boldsymbol{X}=\hat{\boldsymbol{X}}} & \frac{d\boldsymbol{f}(\boldsymbol{X})_{4:6}}{d\boldsymbol{V}_A}\Big|_{\boldsymbol{X}=\hat{\boldsymbol{X}}} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \frac{d\boldsymbol{f}(\boldsymbol{X})_{4:6}}{d\boldsymbol{\omega}^A_{A/I}}\Big|_{\boldsymbol{X}=\hat{\boldsymbol{X}}} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & -\left[\hat{\boldsymbol{\omega}}^B_{B/I}\times\right] & \boldsymbol{0}_{3\times3} & -\boldsymbol{I}_{3\times3} & -\boldsymbol{C}(\boldsymbol{q}_{B/A}) \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & -\left[\hat{\boldsymbol{\omega}}^B_{B/I}\times\right] & -\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \end{bmatrix} \tag{45}$$

and its discrete time version, $\boldsymbol{\Phi}$:

$$\boldsymbol{\Phi} = \exp^{\boldsymbol{F}\Delta t} \approx \boldsymbol{I}_{18\times18} + \boldsymbol{F}\Delta t \tag{46}$$

IV.D. Measurement Equations

Since most of the measurements are non linear, the measurement matrices, $\boldsymbol{H}$, for the filter must be found.

Startracker

The star tracker measures the inertial attitude directly, thus it can be written in the form:

$$\boldsymbol{z}_{ST} = \tilde{\boldsymbol{q}}_{B/I} = \boldsymbol{H}_{ST}\boldsymbol{X} \tag{47}$$

$$\boldsymbol{H}_{ST} = \begin{bmatrix} \boldsymbol{0}_{3\times9} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times6} \end{bmatrix} \tag{48}$$

Navigation Camera

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

$$\boldsymbol{z}_{NAVCAM,i} = \boldsymbol{h}(\boldsymbol{X}) = \boldsymbol{C}(\boldsymbol{q}_{B/A})\frac{\boldsymbol{R}_{LM,i} - \boldsymbol{R}_A}{\|\boldsymbol{R}_{LM,i} - \boldsymbol{R}_A\|} \tag{49}$$

To adapt Eq. (49) for the filter, we express the error between the real and predicted measurement. For this, we write the attitude matrix as follows:[7]

$$\boldsymbol{C}\left(\boldsymbol{q}_{B/A}\right) = \left(\boldsymbol{I}_3 - [\delta\boldsymbol{\vartheta}_R\times]\right)\boldsymbol{C}\left(\hat{\boldsymbol{q}}_{B/A}\right) \tag{50}$$

Substituting Eq. (50) into Eq. (49), and defining $\delta\boldsymbol{z} = \boldsymbol{z} - \hat{\boldsymbol{z}}$ yields:

$$\delta\boldsymbol{z} = \boldsymbol{C}_{B/A}(\hat{\boldsymbol{q}}_{B/A})\frac{\boldsymbol{R}_{LM,i} - \boldsymbol{R}_A}{\|\boldsymbol{R}_{LM,i} - \boldsymbol{R}_A\|} \times \delta\boldsymbol{\vartheta}_R \tag{51}$$

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:

$$\boldsymbol{H}_{NAVCAM} = \left[\left. \frac{d\boldsymbol{h}(\boldsymbol{X})}{d\boldsymbol{R}_A} \right|_{\boldsymbol{X}=\hat{\boldsymbol{X}}} \quad \boldsymbol{0}_{3\times 3} \quad \left[\boldsymbol{C}_{B/A}(\hat{\boldsymbol{q}}_{B/A}) \frac{\boldsymbol{R}_{LM,i}-\hat{\boldsymbol{R}}_A}{\|\boldsymbol{R}_{LM,i}-\hat{\boldsymbol{R}}_A\|} \times \right] \quad \boldsymbol{0}_{3\times 9} \right] \tag{52}$$

Laser Ranger

The laser ranger measures distance to a selected landmark:

$$z_{LR} = h(\boldsymbol{X}) = \|\boldsymbol{R}_{LM,i} - \boldsymbol{R}_A\| \tag{53}$$

To obtain the measurement matrix, we just find a Jacobian of Eq. (53):

$$\boldsymbol{H}_{LR} = \left[\left. \frac{d\boldsymbol{h}(\boldsymbol{X})}{d\boldsymbol{R}_A} \right|_{\boldsymbol{X}=\hat{\boldsymbol{X}}} \quad \boldsymbol{0}_{1\times 15} \right] \tag{54}$$

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.[15] 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:[17]

$$\check{d} = a + \epsilon b \tag{55}$$

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:

$$\epsilon \neq 0, \qquad \epsilon^2 = 0 \tag{56}$$

There is no trivial explanation of these properties of ϵ, thus it has to be taken as it is given, and accepted as truth. The first part in Eq. (55) 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.[17]

V.A. Dual Quaternion

Similarly to how a dual number is constructed, a dual quaternion (DQ), $\check{\boldsymbol{q}}$, can be written as:[16]

$$\check{\boldsymbol{q}} = \boldsymbol{q}_r + \epsilon \boldsymbol{q}_d \tag{57}$$

where $\boldsymbol{q}_r$ is the real and $\boldsymbol{q}_d$ 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:

$$\check{\boldsymbol{q}} = q_{r1}i + q_{r2}j + q_{r3}k + q_{r4} + \epsilon\left(q_{d1}i + q_{d2}j + q_{d3}k + q_{d4}\right) \tag{58}$$

where i, j and k are imaginary numbers. It is cumbersome to manipulate a dual quaternion algebraically, as in Eq. (58). Thus, in this paper a dual quaternion is seen as an 8-tuple vector, defined as follows:

$$\check{\boldsymbol{q}} = \begin{pmatrix} \boldsymbol{q}_r \\ \boldsymbol{q}_d \end{pmatrix} = \begin{pmatrix} q_{r1} \\ q_{r2} \\ q_{r3} \\ q_{r4} \\ q_{d1} \\ q_{d2} \\ q_{d3} \\ q_{d4} \end{pmatrix} \tag{59}$$

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

$$\check{\boldsymbol{q}}_1 \check{\otimes} \check{\boldsymbol{q}}_2 = \boldsymbol{q}_{r1} \otimes \boldsymbol{q}_{r2} + \epsilon \left(\boldsymbol{q}_{r1} \otimes \boldsymbol{q}_{d2} + \boldsymbol{q}_{d1} \otimes \boldsymbol{q}_{r2} \right) \tag{60}$$

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

$$\left[\check{\boldsymbol{q}}_1 \check{\otimes} \right] \check{\boldsymbol{q}}_2 = \begin{bmatrix} [\boldsymbol{q}_{r1} \otimes] & \mathbf{0}_{4\times 4} \\ [\boldsymbol{q}_{d1} \otimes] & [\boldsymbol{q}_{r1} \otimes] \end{bmatrix} \begin{pmatrix} \boldsymbol{q}_{r2} \\ \boldsymbol{q}_{d2} \end{pmatrix} \tag{61}$$

V.B. Conjugates

A dual quaternion can have three different conjugates:[16]

$$\check{\boldsymbol{q}}^{\diamond} = \boldsymbol{q}_r - \epsilon \boldsymbol{q}_d \tag{62}$$

$$\check{\boldsymbol{q}}^{*} = \boldsymbol{q}_r^{*} + \epsilon \boldsymbol{q}_d^{*} \tag{63}$$

$$\check{\boldsymbol{q}}^{\circ} = \boldsymbol{q}_r^{*} - \epsilon \boldsymbol{q}_d^{*} \tag{64}$$

The first and the third conjugates are not much useful, while the second one is. A multiplication of DQ with this conjugate gives:

$$\check{\boldsymbol{q}} \check{\otimes} \check{\boldsymbol{q}}^{*} = \left(\boldsymbol{q}_r + \epsilon \boldsymbol{q}_d \right) \left(\boldsymbol{q}_r^{*} + \epsilon \boldsymbol{q}_d^{*} \right) = \boldsymbol{q}_r \otimes \boldsymbol{q}_r^{*} + \epsilon \left(\boldsymbol{q}_r \otimes \boldsymbol{q}_d^{*} + \boldsymbol{q}_d \otimes \boldsymbol{q}_r^{*} \right) \tag{65}$$

A quaternion multiplication with its conjugate ($\boldsymbol{q}_r \otimes \boldsymbol{q}_r^{*}$) gives a scalar, and the dual part, after some mathematical treatment, turns out to be also a scalar. Thus, the result of the second conjugate product is given as follows:

$$\check{\boldsymbol{q}} \check{\otimes} \check{\boldsymbol{q}}^{*} = \left\| \boldsymbol{q}_r \right\|^2 + 2\epsilon \left(q_{r1} q_{d1} + q_{r2} q_{d2} + q_{r3} q_{d3} + q_{r4} q_{d4} \right) \tag{66}$$

The result in a general case is a dual number ($a + \epsilon b$), but if the real part is orthogonal to the dual one, the dual part vanishes, and the product becomes a real number.

V.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[16] 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:

$$\boldsymbol{q}_d = \frac{1}{2} \boldsymbol{q}_r \otimes \boldsymbol{R} \tag{67}$$

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

$$\check{\boldsymbol{q}} \check{\otimes} \check{\boldsymbol{q}}^{*} = \left(\boldsymbol{q}_r + \frac{1}{2} \epsilon \boldsymbol{q}_r \otimes \boldsymbol{R} \right) \check{\otimes} \left(\boldsymbol{q}_r + \frac{1}{2} \epsilon \boldsymbol{q}_r \otimes \boldsymbol{R} \right)^{*} = \left(\boldsymbol{q}_r + \frac{1}{2} \epsilon \boldsymbol{q}_r \otimes \boldsymbol{R} \right) \check{\otimes} \left(\boldsymbol{q}_r^{*} + \frac{1}{2} \epsilon \boldsymbol{R}^{*} \otimes \boldsymbol{q}_r^{*} \right) = \tag{68}$$

$$= \boldsymbol{q}_r \otimes \boldsymbol{q}_r^{*} + \frac{1}{2} \epsilon \left(\boldsymbol{q}_r \otimes \boldsymbol{R} \otimes \boldsymbol{q}_r^{*} + \boldsymbol{q}_r \otimes \boldsymbol{R}^{*} \otimes \boldsymbol{q}_r^{*} \right)$$

The quaternion conjugate of a pure vector is simply $\boldsymbol{R}^{*} = -\boldsymbol{R}$, so Eq. (68) reduces to a scalar 1, 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:

$$\check{\boldsymbol{q}} \check{\otimes} \check{\boldsymbol{q}}^{*} = 1 \tag{69}$$

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:

$$\left\| \boldsymbol{q}_r \right\|^2 = 1 \tag{70}$$

Furthermore, a second constraint is introduced, which is derived from the dual part of the dual quaternion product in Eq. (66).

$$(q_{r1}q_{d1} + q_{r2}q_{d2} + q_{r3}q_{d3} + q_{r4}q_{d4}) = \boldsymbol{q}_r \cdot \boldsymbol{q}_d = 0 \tag{71}$$

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 $\mathcal{F}_\mathcal{A}$ and $\mathcal{F}_\mathcal{B}$. The attitude of $\mathcal{F}_\mathcal{B}$ with respect to $\mathcal{F}_\mathcal{A}$ is $\boldsymbol{q}_{B/A}$. Then we would write the position vector in $\mathcal{F}_\mathcal{A}$ and the resulting dual quaternion, representing the pose of $\mathcal{F}_\mathcal{B}$ with respect to $\mathcal{F}_\mathcal{A}$, would be given as follows:

$$\check{\boldsymbol{q}}_{B/A} = \boldsymbol{q}_{B/A} + \frac{\epsilon}{2}\boldsymbol{q}_{B/A} \otimes \boldsymbol{R}_A \tag{72}$$

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

$$\check{\boldsymbol{q}}_{B/A} = \boldsymbol{q}_{B/A} + \frac{\epsilon}{2}\boldsymbol{q}_{B/A} \otimes \boldsymbol{R}_A \otimes \boldsymbol{q}^*_{B/A} \otimes \boldsymbol{q}_{B/A} \tag{73}$$

where $\boldsymbol{q}_{B/A} \otimes \boldsymbol{R}_A \otimes \boldsymbol{q}^*_{B/A}$ is a quaternion frame transformation for vectors in frame $\mathcal{F}_\mathcal{A}$ to be expressed in frame $\mathcal{F}_\mathcal{B}$. As a result, Eq. (73) becomes:

$$\check{\boldsymbol{q}}_{B/A} = \boldsymbol{q}_{B/A} + \frac{\epsilon}{2}\boldsymbol{R}_B \otimes \boldsymbol{q}_{B/A} \tag{74}$$

Comparing Eqs. (72) and (74), 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:

$$\boldsymbol{q}_{B/A} = \boldsymbol{q}_r \tag{75}$$

$$\boldsymbol{R}_A = 2\boldsymbol{q}^*_r \otimes \boldsymbol{q}_d \tag{76}$$

V.D. Screw Displacement

Similarly to Euler's theorem, which allows us to visualise a quaternion, there is a Chasle's theorem, which 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. This can be seen as a screw motion (rotation and translation at the same time) with parameters as pitch angle, pitch distance and the screw axis, derived from a dual quaternion.[16]

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.

VI.A. Relative Pose

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

$$\check{\boldsymbol{q}}_{A/I} = \boldsymbol{q}_{A/I} + \epsilon\boldsymbol{0} \tag{77}$$

$$\check{\boldsymbol{q}}_{B/I} = \boldsymbol{q}_{B/I} + \frac{\epsilon}{2}\boldsymbol{q}_{B/I} \otimes \boldsymbol{R}_I \tag{78}$$

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

$$\check{\boldsymbol{q}}_{B/A} = \check{\boldsymbol{q}}_{B/I} \check{\otimes} \check{\boldsymbol{q}}^*_{A/I} = \left(\boldsymbol{q}_{B/I} + \frac{\epsilon}{2}\boldsymbol{q}_{B/I} \otimes \boldsymbol{R}_I\right)\left(\boldsymbol{q}^*_{A/I} + \epsilon\mathbf{0}\right) = \boldsymbol{q}_{B/I} \otimes \boldsymbol{q}^*_{A/I} + \frac{\epsilon}{2}\boldsymbol{q}_{B/I} \otimes \boldsymbol{R}_I \otimes \boldsymbol{q}^*_{A/I} \tag{79}$$

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

$$\boldsymbol{R}_I = \boldsymbol{q}^*_{A/I} \otimes \boldsymbol{R}_A \otimes \boldsymbol{q}_{A/I} \tag{80}$$

Substituting Eq. (80) into Eq. (79) yields:

$$\check{\boldsymbol{q}}_{B/A} = \boldsymbol{q}_{B/A} + \frac{\epsilon}{2}\boldsymbol{q}_{B/I} \otimes \boldsymbol{q}^*_{A/I} \otimes \boldsymbol{R}_A \otimes \boldsymbol{q}_{A/I} \otimes \boldsymbol{q}^*_{A/I} = \boldsymbol{q}_{B/A} + \frac{\epsilon}{2}\boldsymbol{q}_{B/A} \otimes \boldsymbol{R}_A \tag{81}$$

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

VI.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:

$$\boldsymbol{X}_{21} = \begin{pmatrix} \check{\boldsymbol{q}}_{B/A} \\ \boldsymbol{V}_A \\ \boldsymbol{q}_{B/I} \\ \boldsymbol{\mu} \\ \boldsymbol{\omega}^A_{A/I} \end{pmatrix} \tag{82}$$

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

VI.C. Dual Quaternion Kinematic Equation

By taking Eq.(81) and differentiating it with respect to time, the following is obtained:

$$\dot{\check{\boldsymbol{q}}}_{B/A} = \dot{\boldsymbol{q}}_{B/A} + \frac{\epsilon}{2}\left(\dot{\boldsymbol{q}}_{B/A} \otimes \boldsymbol{R}_A + \boldsymbol{q}_{B/A} \otimes \dot{\boldsymbol{R}}_A\right) = \tag{83}$$

$$= \frac{1}{2}\boldsymbol{\omega}^B_{B/A} \otimes \boldsymbol{q}_{B/A} + \frac{\epsilon}{2}\left(\frac{1}{2}\boldsymbol{\omega}^B_{B/A} \otimes \boldsymbol{q}_{B/A} \otimes \boldsymbol{R}_A + \boldsymbol{q}_{B/A} \otimes \boldsymbol{V}_A\right)$$

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

$$\dot{\check{\boldsymbol{q}}}_{B/A} = \frac{1}{2}\begin{bmatrix} \left[\boldsymbol{\omega}^B_{B/A}\otimes\right] & \mathbf{0}_{4\times4} \\ \left[\boldsymbol{V}_A\odot\right] & \left[\boldsymbol{\omega}^B_{B/A}\otimes\right] \end{bmatrix}\begin{pmatrix} \boldsymbol{q}_r \\ \boldsymbol{q}_d \end{pmatrix} \tag{84}$$

where $\boldsymbol{q}_r = \boldsymbol{q}_{B/A}$ and $\boldsymbol{q}_d = \frac{\epsilon}{2}\boldsymbol{q}_{B/A} \otimes \boldsymbol{R}_A$ are the real and the dual parts of the dual quaternion, respectively. Furthermore the term $\boldsymbol{q}_{B/A} \otimes \boldsymbol{V}_A$ in Eq. (83) can be rewritten as:

$$\boldsymbol{q}_{B/A} \otimes \boldsymbol{V}_A = \boldsymbol{q}_{B/A} \otimes \boldsymbol{V}_A \otimes \boldsymbol{q}^*_{B/A}\boldsymbol{q}_{B/A} = \boldsymbol{V}_B \otimes \boldsymbol{q}_{B/A} \tag{85}$$

Then, substituting Eq. (85) into Eq. (83) and writing it in matrix form, yields:

$$\dot{\check{\boldsymbol{q}}}_{B/A} = \frac{1}{2}\begin{bmatrix} \left[\boldsymbol{\omega}^B_{B/A}\otimes\right] & \mathbf{0}_{4\times4} \\ \left[\boldsymbol{V}_B\otimes\right] & \left[\boldsymbol{\omega}^B_{B/A}\otimes\right] \end{bmatrix}\begin{pmatrix} \boldsymbol{q}_r \\ \boldsymbol{q}_d \end{pmatrix} \tag{86}$$

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

$$\check{\boldsymbol{\omega}}_B = \boldsymbol{\omega}^B_{B/A} + \epsilon\boldsymbol{V}_B \tag{87}$$

The kinematics equation is then:

$$\dot{\check{\boldsymbol{q}}}_{B/A} = \frac{1}{2}\check{\boldsymbol{\omega}}_B \check{\otimes} \check{\boldsymbol{q}}_{B/A} \tag{88}$$

where $\check{\otimes}$ is a dual quaternion product and, according to Eq. (86), it is defined as follows:

$$\left[\check{\boldsymbol{\omega}}^B_{B/I}\check{\otimes}\right] = \begin{bmatrix} \left[\boldsymbol{\omega}^B_{B/A}\otimes\right] & \mathbf{0}_{4\times4} \\ [\boldsymbol{V}_B\otimes] & \left[\boldsymbol{\omega}^B_{B/A}\otimes\right] \end{bmatrix} \tag{89}$$

Full process model

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

$$\boldsymbol{f}(\boldsymbol{X}) = \begin{pmatrix} \frac{1}{2}\boldsymbol{\Omega}\left(\tilde{\boldsymbol{\omega}}^B_{B/I} - \boldsymbol{\mu} - \boldsymbol{C}(\boldsymbol{q}_r)\boldsymbol{\omega}^A_{A/I}\right)\boldsymbol{q}_r \\ \frac{1}{2}\boldsymbol{\omega}^B_{B/A}\otimes\boldsymbol{q}_d + \frac{1}{2}\boldsymbol{q}_r\otimes\boldsymbol{V}_A \\ -\frac{GM}{R_A^3}\boldsymbol{R}_A - 2\boldsymbol{\omega}_{A/I}\times\boldsymbol{V}_A - \boldsymbol{\omega}_{A/I}\times\boldsymbol{\omega}_{A/I}\times\boldsymbol{R}_A \\ \frac{1}{2}\boldsymbol{\Omega}(\tilde{\boldsymbol{\omega}}^B_{B/I} - \boldsymbol{\mu})\boldsymbol{q}_{B/I} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \tag{90}$$

VI.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:

$$\delta\check{\boldsymbol{q}} = \check{\boldsymbol{q}}_{B/A}\otimes\widehat{\check{\boldsymbol{q}}}^*_{B/A} \tag{91}$$

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

$$\check{\boldsymbol{q}}_{B/A} = \boldsymbol{q}_{B/A} + \frac{\epsilon}{2}\boldsymbol{q}_{B/A}\otimes\boldsymbol{R}_A \tag{92}$$

$$\widehat{\check{\boldsymbol{q}}}_{B/A} = \hat{\boldsymbol{q}}_{B/A} + \frac{\epsilon}{2}\hat{\boldsymbol{q}}_{B/A}\otimes\hat{\boldsymbol{R}}_A \tag{93}$$

Substituting Eqs. (92) and (93) to (91) and using $(\boldsymbol{q}_1\otimes\boldsymbol{q}_2)^* = \boldsymbol{q}_2^*\otimes\boldsymbol{q}_1^*$ yields:

$$\delta\check{\boldsymbol{q}} = \left(\boldsymbol{q}_{B/A} + \frac{\epsilon}{2}\boldsymbol{q}_{B/A}\otimes\boldsymbol{R}_A\right)\left(\hat{\boldsymbol{q}}^*_{B/A} + \frac{\epsilon}{2}\hat{\boldsymbol{R}}^*_A\otimes\hat{\boldsymbol{q}}^*_{B/A}\right) = \tag{94}$$

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

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

$$\delta\check{\boldsymbol{q}} = \delta\boldsymbol{q}_R + \frac{\epsilon}{2}\boldsymbol{q}_{B/A}\otimes\left(\boldsymbol{R}_A - \hat{\boldsymbol{R}}_A\right)\otimes\hat{\boldsymbol{q}}^*_{B/A} = \tag{95}$$

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

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

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

$$\delta\check{\boldsymbol{q}} \approx \delta\boldsymbol{q}_R + \frac{\epsilon}{2}\delta\boldsymbol{R}_B \tag{96}$$

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

Error kinematics

We differentiate the error, Eq. (95), 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. (43), therefore it is not repeated again. The dual part (all terms with ϵ) derivative is as follows:

$$\delta\dot{\boldsymbol{q}}_d = \left(\hat{\boldsymbol{V}}_B + \delta\boldsymbol{V}_B\right) \otimes \delta\boldsymbol{q}_R + \frac{1}{4}\left(\hat{\boldsymbol{\omega}}^B_{B/A} + \delta\boldsymbol{\omega} - \boldsymbol{C}\left(\hat{\boldsymbol{q}}_{B/A}\right)\hat{\boldsymbol{\omega}}^A_{A/I} \times \delta\boldsymbol{\vartheta}_R - \boldsymbol{C}\left(\hat{\boldsymbol{q}}_{B/A}\right)\delta\boldsymbol{\omega}_A\right) \otimes \delta\boldsymbol{R}_B + \tag{97}$$

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

Neglecting the second order terms yields:

$$\delta\dot{\boldsymbol{q}}_d = \frac{1}{2}\hat{\boldsymbol{V}}_B \otimes \delta\boldsymbol{q}_R + \frac{1}{2}\delta\boldsymbol{V}_B + \frac{1}{4}\hat{\boldsymbol{\omega}}^B_{B/A} \otimes \delta\boldsymbol{R}_B - \frac{1}{2}\delta\boldsymbol{q}_R \otimes \hat{\boldsymbol{V}}_B - \frac{1}{4}\delta\boldsymbol{R}_B \otimes \hat{\boldsymbol{\omega}}^B_{B/A} = \tag{98}$$

$$= -\left[\hat{\boldsymbol{V}}_B\times\right]\delta\boldsymbol{q}_R + \frac{1}{2}\delta\boldsymbol{V}_B - \frac{1}{2}\left[\hat{\boldsymbol{\omega}}^B_{B/A}\times\right]\delta\boldsymbol{R}_B =$$

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

Since the dual part error can be represented by $\delta\boldsymbol{R}_B$, Eq. (96), and the real part by a small rotation vector, $\boldsymbol{\vartheta}_R$, the whole dual quaternion error is written as:

$$\delta\check{\boldsymbol{q}} = \begin{pmatrix} \frac{1}{2}\delta\boldsymbol{\vartheta}_R \\ 1 \\ \frac{1}{2}\delta\boldsymbol{R}_B \\ 0 \end{pmatrix} \tag{99}$$

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

$$\delta\check{\boldsymbol{q}}_{1:6} = \begin{pmatrix} \delta\boldsymbol{\vartheta}_R \\ \delta\boldsymbol{R}_B \end{pmatrix} \tag{100}$$

Then, Eq. (98) becomes:

$$\delta\dot{\boldsymbol{R}}_B = -\left[\boldsymbol{C}\left(\hat{\boldsymbol{q}}_{B/A}\right)\hat{\boldsymbol{V}}_A\times\right]\delta\boldsymbol{\vartheta}_R + \boldsymbol{C}\left(\hat{\boldsymbol{q}}_{B/A}\right)\delta\boldsymbol{V}_A - \left[\hat{\boldsymbol{\omega}}^B_{B/A}\times\right]\delta\boldsymbol{R}_B \tag{101}$$

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

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:

$$\delta\boldsymbol{X}_{18} = \begin{pmatrix} \delta\boldsymbol{\vartheta}_R \\ \delta\boldsymbol{R}_B \\ \delta\boldsymbol{V}_A \\ \delta\boldsymbol{\vartheta}_I \\ \delta\boldsymbol{\mu} \\ \delta\boldsymbol{\omega}_A \end{pmatrix} \tag{102}$$

Full Linear-perturbation Model

The linear perturbation model would basically be different only by the position error kinematics, expressed in Eq. (101). However, since the error, $\delta \boldsymbol{R}_B$ is now expressed in $\mathcal{F}_{\mathcal{B}}$ instead of $\mathcal{F}_{\mathcal{A}}$, as it was in QVEKF, the Jacobian for the velocity has to be modified accordingly. Without loss of generality, we can write:

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{R}_B} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{R}_A} \frac{\partial \boldsymbol{R}_A}{\partial \boldsymbol{R}_B} \tag{103}$$

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

$$\frac{\partial \boldsymbol{R}_A}{\partial \boldsymbol{R}_B} = \boldsymbol{C}_{A/B} = \boldsymbol{C}\left(\boldsymbol{q}_{B/A}\right)^T \tag{104}$$

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

$$\frac{\partial \boldsymbol{V}_A}{\partial \boldsymbol{R}_B} = \frac{\partial \boldsymbol{V}_A}{\partial \boldsymbol{R}_A} \boldsymbol{C}\left(\boldsymbol{q}_{B/A}\right)^T \tag{105}$$

$$\boldsymbol{F} = \begin{bmatrix} -\left[\hat{\boldsymbol{\omega}}^B_{B/I}\times\right] & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & -\boldsymbol{I}_{3\times3} & -\boldsymbol{C}(\boldsymbol{q}_{B/A}) \\ -\left[\boldsymbol{C}(\boldsymbol{q}_{B/A})\hat{\boldsymbol{V}}_A\times\right] & -\left[\hat{\boldsymbol{\omega}}^B_{B/A}\times\right] & \boldsymbol{C}(\boldsymbol{q}_{B/A}) & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \frac{d\boldsymbol{f}(\boldsymbol{X})_{7:9}}{d\boldsymbol{R}_A}\boldsymbol{C}\left(\boldsymbol{q}_{B/A}\right)^T\Big|_{\boldsymbol{X}=\hat{\boldsymbol{X}}} & \frac{d\boldsymbol{f}(\boldsymbol{X})_{7:9}}{d\boldsymbol{V}_A}\Big|_{\boldsymbol{X}=\hat{\boldsymbol{X}}} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \frac{d\boldsymbol{f}(\boldsymbol{X})_{7:9}}{d\boldsymbol{\omega}^A_{A/I}}\Big|_{\boldsymbol{X}=\hat{\boldsymbol{X}}} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & -\left[\hat{\boldsymbol{\omega}}^B_{B/I}\times\right] & -\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \end{bmatrix} \tag{106}$$

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

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{0}_{3\times3} & -\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & -\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix} \tag{107}$$

VI.E. Measurement Equations

Since the position error in DQEKF is represented by $\delta \boldsymbol{R}_B$, and it was $\delta \boldsymbol{R}_A$ 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:

$$\boldsymbol{H}_{ST} = \begin{bmatrix} \boldsymbol{0}_{3\times9} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times6} \end{bmatrix} \tag{108}$$

Note that the order of the state variables has changed, so the matrix has changed accordingly, see Eq. (102).

Navigation Camera

For the Jacobians with respect to the position vector, the same method is applied as in Eq. (104):

$$\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{R}_B} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{R}_A} \boldsymbol{C}\left(\boldsymbol{q}_{B/A}\right)^T \tag{109}$$

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

$$\boldsymbol{H}_{NAVCAM} = \left[\begin{array}{cc} \left[\boldsymbol{C}_{B/A}(\hat{\boldsymbol{q}}_{B/A}) \frac{\boldsymbol{R}_{LM,i}-\hat{\boldsymbol{R}}_A}{\left\|\boldsymbol{R}_{LM,i}-\hat{\boldsymbol{R}}_A\right\|}\times\right] & \frac{d\boldsymbol{h}(\boldsymbol{X})}{d\boldsymbol{R}_A}\boldsymbol{C}\left(\boldsymbol{q}_{B/A}\right)^T \quad \boldsymbol{0}_{3\times 12} \end{array} \right] \Bigg|_{\boldsymbol{X}=\hat{\boldsymbol{X}}} \tag{110}$$

A different approach for dual quaternions is possible, where the landmarks are treated not as points, but rather as lines connecting two landmarks. This method was presented by Goddard,[?] however, for the sake of equal comparison the DQEKF will process the landmarks as points. The dual-line representation is left as a future work.

Laser Ranger

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

$$\boldsymbol{H}_{LR} = \left[\begin{array}{ccc} \boldsymbol{0}_{1\times 3} & \frac{d\boldsymbol{h}(\boldsymbol{X})}{d\boldsymbol{R}_A}\boldsymbol{C}\left(\boldsymbol{q}_{B/A}\right)^T & \boldsymbol{0}_{1\times 12} \end{array} \right] \Bigg|_{\boldsymbol{X}=\hat{\boldsymbol{X}}} \tag{111}$$

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[b]. A 'polar' orbit around asteroid Kleopatra is simulated for 10,000 s with the initial position and velocity:

$$\boldsymbol{R}_A = \begin{pmatrix} 0 \\ 0 \\ 200 \end{pmatrix} \text{ km}, \quad \boldsymbol{V}_A = \begin{pmatrix} 0 \\ -35.35 \\ 0 \end{pmatrix} \text{ m/s} \tag{112}$$

The initial inertial and relative attitudes are the same:

$$\boldsymbol{q}_{B/I} = \boldsymbol{q}_{B/A} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{113}$$

which means that at $t = 0$ frames $\mathcal{F}_{\mathcal{I}}$ and $\mathcal{F}_{\mathcal{A}}$ 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:

$$\boldsymbol{\omega}_{B/I}^B = \begin{pmatrix} 1.711 \times 10^{-4} \\ 0 \\ 0 \end{pmatrix} \text{ rad/s} \tag{114}$$

The trajectory is shown in Figure 5, where the light-blue pyramids show the FOV pyramids of the spacecraft.

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 6 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.

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

[b] http://www.esa.int/Our_Activities/Space_Science/Rosetta/The_Rosetta_orbiter

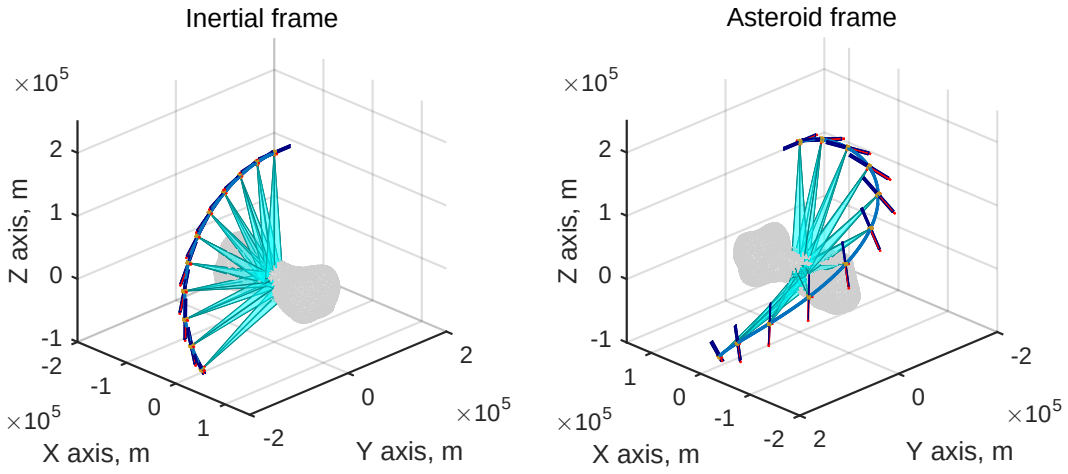

Figure 5: Simulated trajectory around the Kleopatra asteroid (the spacecraft is not in scale)

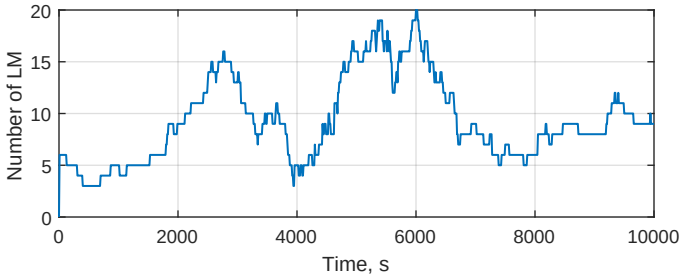

Figure 6: Number of landmarks, seen by the navigation camera per frame

between the pointing vector and the local (landmark) normal vector, Fig. 3. 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 dependant, so there is no analytical function to express them. Figure 7 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 ≈ 25 m^2. The second has an angle from 20° to 40° and the variance of ≈ 169 m^2. The third has an angle from 40° to 60° and the variance of ≈ 900 m^2, and the last one is 60° and above, and the error variance ≈ 2500 m^2.

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:

$$R_{LR1} = 25 \text{ m}^2, \quad R_{LR2} = 169 \text{ m}^2, \quad R_{LR3} = 900 \text{ m}^2 \text{ and } \quad R_{LR4} = 2500 \text{ m}^2 \tag{115}$$

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

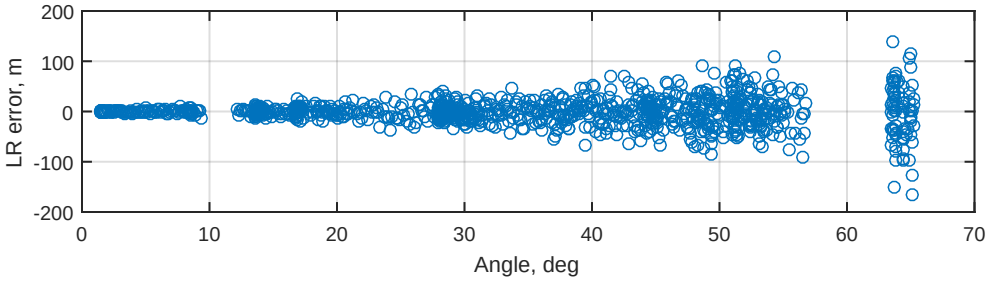

Figure 7: Laser-ranger angles between the pointing and landmark normal vectors, and corresponding distance errors

VII.A. Filter Initialisation

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

State vector

Position and velocity are initialised with errors of 1000 m and 1 m/s on each axis, respectively. Relative and inertial quaternions are selected as:

$$\hat{\boldsymbol{q}}_{B/A,0} = \begin{pmatrix} 0.905 \\ -0.272 \\ 0.181 \\ -0.272 \end{pmatrix} \quad \text{and} \quad \hat{\boldsymbol{q}}_{B/I,0} \begin{pmatrix} 0.945 \\ -0.189 \\ 0.189 \\ -0.189 \end{pmatrix} \tag{116}$$

which represents $\Phi_R = 50.35°$ and $\Phi_I = 38.18°$ 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:

$$\hat{\boldsymbol{\omega}}^A_{A/I,0} = \begin{pmatrix} 0.02 \\ 0.02 \\ 3.565 \end{pmatrix} \times 10^{-4} \text{ rad/s} \tag{117}$$

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:

$$\boldsymbol{P}_{0/0} = \begin{bmatrix} 10^7\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & 10\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & 10^{-8}\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & 10^{-4}\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & 10^{-10}\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & 10^{-14}\boldsymbol{I}_{3\times3} \end{bmatrix} \tag{118}$$

The process noise covariance matrix after tuning was set to:

$$\boldsymbol{Q} = \begin{bmatrix} \sigma_g^2\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \sigma_v^2\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \sigma_u^2\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & 10^{-16}\boldsymbol{I}_{3\times3} \end{bmatrix} \tag{119}$$

where $\sigma_g = 0.005$ m/s^2 is the uncertainty in the gravity field, $\sigma_v = 5.8 \times 10^{-7}$ rad/s$^{1/2}$ and $\sigma_u = 5.8 \times 10^{-8}$ rad/s$^{3/2}$ are the standard deviations of gyroscope noises. Some of the values in the error and process noise covariance matrices are set really small (close to zero, e.g., 10^{-16}), because it proved to give the best performance. Increasing them, could result in a divergence of the filter. The star tracker has the measurement covariance matrix:

$$\boldsymbol{R}_{ST} = 0.1 \times \begin{bmatrix} 4,76 \times 10^{-9} & 0 & 0 \\ 0 & 5.88 \times 10^{-11} & 0 \\ 0 & 0 & 5.88 \times 10^{-11} \end{bmatrix} \tag{120}$$

The covariance matrix of the navigation camera is:

$$\boldsymbol{R}_{NAVC} = \begin{bmatrix} 1.69 \times 10^{-11} & 0 & 0 \\ 0 & 1.69 \times 10^{-11} & 0 \\ 0 & 0 & 1.69 \times 10^{-13} \end{bmatrix} \tag{121}$$

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. (29). Finally, the laser-ranger covariance is given by Eq. (115).

VII.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-σ 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 (35). Figure 8 shows the estimate errors of the drift, $\boldsymbol{\mu}$ (top), and the the asteroid rate, $\boldsymbol{\omega}^A_{A/I}$ (bottom).

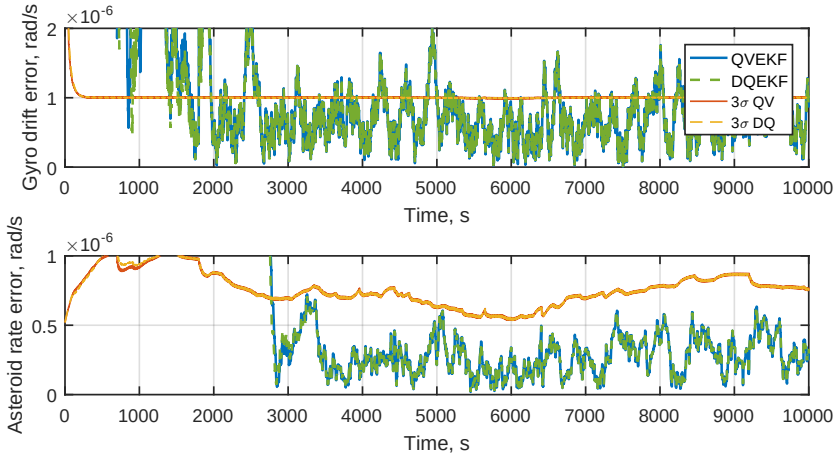

Figure 8: Gyroscope drift and asteroid rate errors

Surprisingly, both filters behave identically in steady state. Gyroscope-drift estimate shows a fast convergence in around 1500 s, and the estimate does not change in time, since it is mainly influenced by star-tracker measurements, which are independent from relative states and are always available. The errors are concentrated within 3-σ boundaries, although not perfectly. The asteroid rates converge slower ($\approx$ 3000 s) and they stay within 3-σ boundaries.

Furthermore, inertial and relative attitude estimates are presented in Fig. 9 by the angle $\Phi = 2\cos^{-1} q_4$. In this case the QVEKF and DQEKF behave the same again. Inertial attitude estimates converge in less than 3000 s. After convergence the error does not change and stays with about 8 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 arsec on the X-,Y- and Z-axis, respectively. So the estimates show an error reduction more then an order of magnitude. The relative attitude estimate converges at about $t = 3000$ s, just after the asteroid rates converge. The error stays of the same order as the inertial attitude error, but reaches a minimum of 10 arcsec $3-\sigma$ at 6000 s (the maximum landmark number in the FOV).

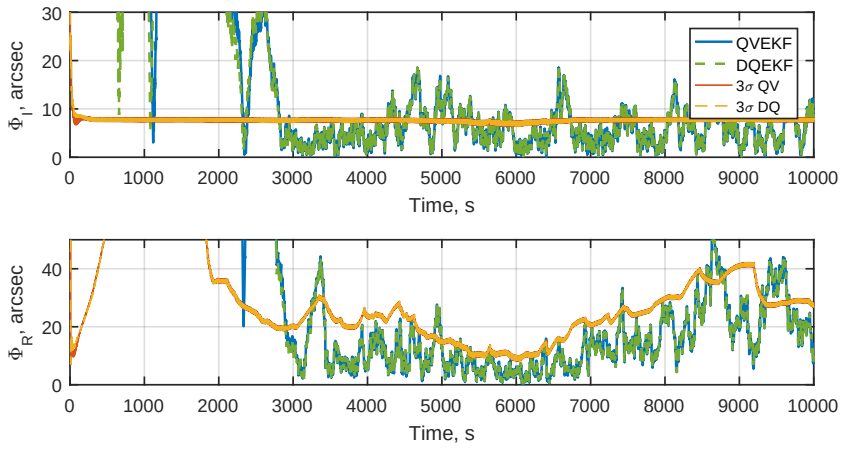

Figure 9: Inertial and relative attitude errors

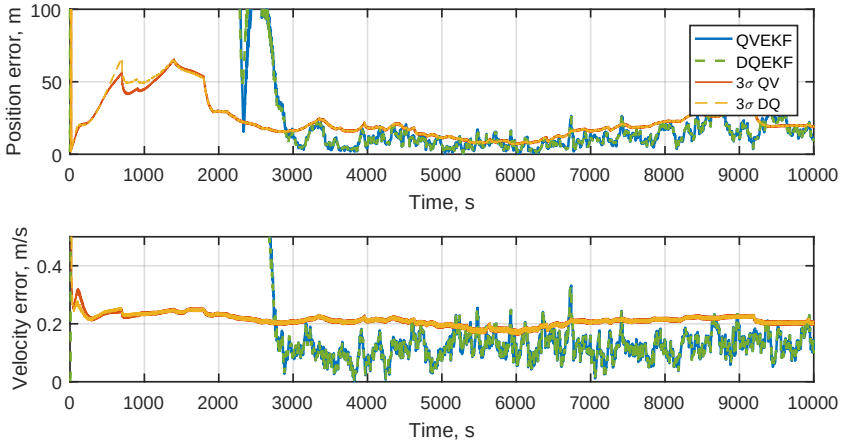

Figure 10: Position and velocity estimates' errors

Finally, relative position and velocity estimates are shown in Fig 10. Both filters still have an identical performance, which at this point can be safely said that the filters are the same. We recall the linear perturbation model developed for dual quaternions, which turned out to be expressed with a position vector in $\mathcal{F}_\mathcal{B}$, so in the end, both filters have linear models that are very alike, and thus the performance in steady state is the same. Nevertheless, position and velocity converge at the same time the asteroid rates converge, since the velocity (and therefore position) depend strictly on the asteroid rate. After convergence, the estimates stay within the 3-σ 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. 6.

VIII. Conclusions and Recommendations

This paper presented the work-flow of simulating the dynamics around an asteroid, modelling navigation sensors and developing a navigation filter. It contributes to all of these fields. Firstly, the gravity-gradient torque was simulated, assuming the spacecraft as a cloud of point-masses and calculating polyhedron gravity field values at each of them, which resulted in a more accurate gravity-gradient torque than the central-gravity field approximation. Secondly, the modelling of a navigation camera and a laser-ranger was accomplished by a novel method, exploiting the very useful property of polyhedron models, which allows to know, whether a point lies inside or outside it. This brings the simulation closer to reality and thus more representative.

Finally, two versions of Extended Kalman filters were developed: one with conventional pose representation (QVEKF), and the other one with dual quaternions (DQEKF). Surprisingly, the filters turned out to perform identically, which might be explained by the linear perturbation model for DQEKF. The dual part of the dual quaternion error can, in fact, be represented by a position vector (in $\mathcal{F}_\mathcal{B}$), which makes the two estimators very alike. However, they still give already satisfying results. The relative position and velocity can be estimated with errors having 5 m and 0.06 m/s standard deviations respectively, inertial and relative attitude with ≈ 3 and ≈ 5 arcsec standard deviations, respectively (when converged). However, these results can only be achieved when there are landmarks in the FOV of the navigation camera, because the dynamics model is not accurate enough and the estimates rely on these measurements. A more accurate model for the gravity field, e.g. spherical harmonic, could be used on-board.

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 25th ISSFD, 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 25th ISSFD, 2015.

[5]Yang, H., Y., Vetrisano, 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]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.

[9]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.

[10]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.

[11]Markley, F.L. and Crassidis, J.L., *Fundamentals of Spacecraft Attitude Determination and Control*, Vol 2, 2013, ISBN 978-1-4939-0801-1, pub. Springer.

[12]Wie, B, *Space Vehicle Dynamics and Control*, Second Edition, 2006, ISBN 978-1-56347-953-3, pub. American Institute of Aeronautics and Astronautics, Inc.

[13]Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D., "Shape distributions", *ACM Transactions on Graphics*, Vol. 21, No. 4, 2002, pp. 807-832.

[14]Markley, F. L., "Attitude Error Representations for Kalman filtering", *Journal of Guidance, Control and Dynamics*, Vol. 26, No. 2, 2003.

[15]Clifford, W. K., "Preliminary sketch of biquaternions", *Proceedings of the London Mathematical Society*, Vol. 4, 1873, pp. 381-195.

[16]Jia, Y. B., Dual quaternion, Com S 477/577, Fall 2015, Course handouts, 2013. URL http://web.cs.iastate.edu/~cs577/.

[17]Fischer, I. S., *Dual-number methods in kinematics, statics, and dynamics*, Boca Raton, CRC Press, 1999.