Inexpensive CubeSat Attitude Estimation Using Quaternions and Unscented Kalman Filtering

Vinther, Kasper; Fuglsang Jensen, Kasper; Larsen, Jesper Abildgaard; Wisniewski, Rafal

Published in:
Automatic Control in Aerospace

Publication date:
2011

Document Version
Accepted author manuscript, peer reviewed version

INEXPENSIVE CUBESAT ATTITUDE ESTIMATION USING QUATERNIONS AND UNSCENTED KALMAN FILTERING

Kasper Vinther, Kasper F. Jensen, Jesper A. Larsen, and Rafael Wisniewski

Department of Electronic Systems, Section of Automation and Control
Aalborg University, 9220, Denmark
{kasperv,kfjensen,jal,raf}@es.aau.dk

Abstract:
This paper describes a quaternion implementation of an Unscented Kalman Filter for attitude estimation on CubeSats using measurements of a sun vector, a magnetic field vector and angular velocity. For faster convergence of the attitude estimate, a SVD-method solving Wahba's problem has been proposed, which provides an initial attitude estimate. Using unit quaternions provides a singularity free attitude parameterization. However, the unity constraint requires a redesign of the Unscented Kalman Filter. Therefore, a quaternion error state is introduced. Emphasis has been put in making the implementation accessible to other CubeSat developers via pseudo code and simulations have shown that the extra computational cost of estimating bias in measurements is worthwhile. The simulations where performed in a simulation environment for the CubeSat AAUSAT3, where robustness has been an important factor during tuning of the attitude estimators. The results indicate that it is possible to achieve acceptable CubeSat attitude estimation, even during eclipse, on a limited budget without expensive high precision sensor setups.

Keywords:
Attitude estimation, Bias estimation, Unscented Kalman filtering, Quaternions, CubeSats, Magnetic sensors, Sun sensors, Gyroscopes, AAUSAT3.

1. INTRODUCTION

There has been an increasing interest in CubeSats since the standard was introduced in 1999 by California Polytechnic State University and Stanford University (Puig-Suari et al. (2001)). According to Chin et al. (2008), the CubeSat community consisted of more than 90 universities and 40 different companies and organizations around the world in 2008, and these numbers have undoubtedly increased in the past two years. The low cost and small size associated with CubeSats has made space accessible to universities. Aalborg University is currently developing their third student satellite called AAUSAT3 (visit `http://www.aausat3.space.aau.dk` for more information). This satellite is based on the CubeSat standard (CalPoly SLO), which entails a size of approximately 10x10x11 cm.

A satellite is often composed of many subsystems including experiment payloads, which might impose requirements on attitude stabilization. An Attitude Determination and Control System (ADCS) is one of these subsystems and estimating the satellites attitude introduces the possibility of using more advanced pointing control, compared to simple detumbling control. Knowing the attitude and possibly controlling it, benefit the experiment payloads as well as improve mission success rates, by, e.g., providing more stable communication links.

The Unscented Kalman Filter (UKF) approach for satellite attitude estimation has been used by e.g. Crassidis and Markley (2003); Cheon (2005); Pourtakdoust and Ghanbarpour (2007); Kraft (2003); VanDyke et al. (2004). This approach is characterized by a set of sample (sigma) points, which are used to approximate a Gaussian probability distribution; whereas, the well known Extended Kalman Filter (EKF) is distinguished by the fact that it uses a linearization of the nonlinear model equations around the current estimate. The computational load of the UKF is higher than that of the EKF, because all sigma points have to be propagated, on the other hand, the UKF does not rely on derivation of Jacobian matrices. Furthermore, the error of the UKF is expected to be lower than that of the EKF, since the sigma points provide a way of estimating the posterior mean and covariance accurately to the second order for any nonlinearity Haykin (2001).

This paper presents attitude estimation using UKF, for CubeSats with limited sensor accuracy. In this work, AAUSAT3 has been used as a test case for the developed algorithms. The sensors chosen for AAUSAT3 are low cost off-the-shelf sensors (below 200 USD). The sensor configuration consists of a 3-axis magnetometer, 3-axis driftless gyroscope, and photodiodes, acting as sun sensor, on each of the six sides of the satellite. If the performance requirements for the ADCS are strict, it might be necessary to include a star camera, earth sensors and/or fine sun sensors; however, fitting such sensors into the size and budget constraints of a CubeSat can be difficult.

The quaternion provides a singularity free attitude representation, which is convenient and as an example, Bak (1999);

Cheon (2005); Crassidis and Markley (2003); Kraft (2003); Lefferts et al. (1982) use quaternions as attitude parameters and they are also used in this paper. However, introducing the unit quaternion, representing pure rotations, requires an extension of the classical UKF. The authors are especially interested in making the quaternion UKF implementation accessible to other CubeSat builders and has, thus, provided the implementation for AAUSAT3 as a pseudo code. Utilizing the SVD-method to solve Wahba's problem makes it possible to provide the quaternion UKF implementation with an initial attitude estimate, which has shown to reduce the convergence time for the UKF, especially in cases where the sensor measurements are biased.

The structure of this paper is as follows. First, the satellite equations of motion are presented together with a description of reference coordinate systems and the quaternion parameterization. Then, an introduction to Wahba's problem, the SVD-method and the UKF is provided. Then, a description of the quaternion UKF implementation is provided, followed by a specification of the simulation environment and the test cases. Finally, the simulation results are presented with a discussion of the performance, robustness and computational load of the implementations.

2. SATELLITE EQUATIONS OF MOTION

Three reference frames are introduced in order to describe rotation of rigid bodies and to define vectors in $\mathbb{R}^3$. Reference frame is used as a descriptive term for a right handed three-dimensional Cartesian coordinate system, described by three mutually perpendicular unit vectors. Multiple reference frames carefully placed ease calculations.

The motion of a satellite (rigid body) is best described in an inertial reference frame (Newtonian reference frame). Therefore, this frame is placed in the center of the Earth with the x-axis going through the point where the vernal equinox and the equatorial plane crosses and the z-axis through the geographic north pole. This approximately creates a non-accelerating reference frame without fictitious forces (Serway (2004)). The y-axis is the cross product of the x- and z-axis, thus creating the right handed Cartesian coordinate system, which is called Earth centered inertial reference frame.

Another reference frame called satellite body reference frame is used to define orientation of ADCS hardware and attitude measurements. The origin of the frame is located in a corner of the satellite to ease measurement of component placement relative to the frame. The x-, y- and z-axes are chosen to be parallel to the satellite frame structure.

The last reference frame is called the controller reference frame, and it is located in the center of mass of the satellite with the x-axis defined as the minor axis of inertia and the z-axis defined as the major axis of inertia. The y-axis is the intermediate axis of inertia and also the cross product between the two other axes. This is a body fixed frame, and it is used for calculations involving the satellites dynamics, as all off-diagonal entries in the inertia matrix are eliminated, which is computationally convenient. The axes are also known as the principal axes (Wie (1998)).

After having defined reference frames for describing position and orientation of objects, it is obvious to discuss rotation of such reference frames, thus making it possible to express the orientation of the objects relative to different viewpoints. This could, e.g., be the orientation of the controller reference frame, which is a satellite body fixed frame, relative to the Earth centered inertial reference frame. It is important that such a rotation preserves distance and natural orientation of $\mathbb{R}^3$ and if $\underline{\mathbf{A}}$ is a transformation matrix, then this transformation must be orthogonal and comply with the following constraints (Wertz (1994)):

$$\begin{aligned} \underline{\mathbf{A}}^{\mathrm{T}}\underline{\mathbf{A}} &= \underline{\mathbf{1}} \\ \det(\underline{\mathbf{A}}) &= 1. \end{aligned} \tag{1}$$

The space spanned by all the transformation matrices satisfying the stated constraints are denoted *SO(3)* also called the special orthogonal group. The matrix $\underline{\mathbf{A}}$ has nine parameters, but only three are independent (i.e. six constraints, thus, three degrees of freedom). However, no three-parameter set can be both global and nonsingular. Even though rotation matrices have a more intuitive representation, quaternions are the preferred attitude representation because of the smaller amount of parameters. Quaternions also have no trigonometric functions in the kinematics and provide a convenient product rule for successive rotations, which makes them computational faster.

The quaternion $\mathbf{q}$ is a hyper complex number composed of a scalar part q_4 and a vector part $\mathbf{q}_{1:3}$, with components spanning $\mathbb{R}^3$:

$$\mathbf{q} = \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3 + q_4 \tag{2}$$

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1 \tag{3}$$

$$\mathbf{ij} = \mathbf{k},\ \mathbf{ji} = -\mathbf{k},\ \mathbf{jk} = \mathbf{i},\ \mathbf{kj} = -\mathbf{i},\ \mathbf{ki} = \mathbf{j},\ \mathbf{ik} = -\mathbf{j}. \tag{4}$$

For any unit quaternion, the operation

$$\mathbf{w} = \mathbf{q}^* \otimes \boldsymbol{\nu} \otimes \mathbf{q} \tag{5}$$

may be interpreted as a frame rotation relative to a fixed space of points or vectors $\boldsymbol{\nu} = \begin{bmatrix}\mathbf{v}^{\mathrm{T}} & 0\end{bmatrix}^{\mathrm{T}}$, or as taking the vector $\mathbf{v}$ and expressing it in another frame (Kuipers (2002)). The operator $\otimes$ denotes a quaternion multiplication, which is defined in e.g. Kuipers (2002). The notation in Eq. (6) is used throughout this paper, where the quaternion ${}^{s}_{i}\mathbf{q}$ represents the satellite attitude (rotation from the Earth centered inertial reference frame to the satellite body reference frame). This quaternion is the product of the rotation from the Earth centered inertial reference frame to the controller reference frame (${}^{c}_{i}\mathbf{q}$) with the inverse rotation from the satellite body reference frame to the controller reference frame (${}^{c}_{s}\mathbf{q}$):

$${}^{s}_{i}\mathbf{q} = {}^{c}_{i}\mathbf{q} \otimes {}^{c}_{s}\mathbf{q}^{-1}. \tag{6}$$

The differential equations of the satellite dynamics relates the time derivative of the angular velocity and the torques applied to the satellite, as expressed in Eq. (7) (derivation can be found in e.g. Wertz (1994); Wie (1998))

$$\underline{\mathbf{I}}^{c}_{sat}\dot{\boldsymbol{\omega}}(t) = -\underline{\mathbf{S}}({}^{c}\boldsymbol{\omega}(t))\underline{\mathbf{I}}_{sat}\,{}^{c}\boldsymbol{\omega}(t) + {}^{c}\mathbf{N}_{dist}(t) + {}^{c}\mathbf{N}_{ctrl}(t), \tag{7}$$

where ${}^c\boldsymbol{\omega}(t)$ is the angular velocity of the controller reference frame with respect to the Earth centered inertial reference frame and the skew symmetric matrix $\underline{\mathbf{S}}({}^c\boldsymbol{\omega}(t))$ is defined in Eq. (8). The torque ${}^c\mathbf{N}_{ctrl}(t)$ is the applied control torque from e.g. magnetorquers and ${}^c\mathbf{N}_{dist}(t)$ is a disturbance torque originating from primarily gravity, solar radiation, atmospheric drag and residual magnetic fields onboard the satellite. In this paper, the disturbance torque is only modeled in the simulation environment and is not part of the model used for attitude estimation. However, the gravity gradient torque is often included in the attitude estimation model on larger satellites, but this is not necessary for CubeSats, which are small and symmetric satellites.

$$\mathbf{S}(\boldsymbol{a}) = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}, \; for \;\; \boldsymbol{a} \in \mathbb{R}^3 \tag{8}$$

The kinematic equations of motion can be expressed as a set of first order differential equations specifying the time evolution of the attitude parameters. Choosing the quaternion parameterization for the satellite kinematic analysis gives the expression in Eq. (9) (derivation can be found in e.g. Wertz (1994); Wie (1998)).

$${}^c_i\dot{\mathbf{q}}(t) = \frac{1}{2}\underline{\boldsymbol{\Omega}}({}^c\boldsymbol{\omega}(t))\,{}^c_i\mathbf{q}(t) \tag{9}$$

The matrix $\underline{\boldsymbol{\Omega}}({}^c\boldsymbol{\omega}(t))$ is defined as

$$\boldsymbol{\Omega}(\boldsymbol{a}) = \begin{bmatrix} \mathbf{S}(\boldsymbol{a}) & \boldsymbol{a} \\ -\boldsymbol{a}^{\mathrm{T}} & 0 \end{bmatrix}, \; for \;\; \boldsymbol{a} \in \mathbb{R}^3. \tag{10}$$

3. WAHBA'S PROBLEM AND THE SVD-METHOD

The attitude of a satellite can be determined by unit vector observations, coming from e.g. sun sensors or magnetic field sensors. If two or more vector observations are available in two different coordinate systems, then it is possible to find a special orthogonal matrix $\underline{\mathbf{A}}$, representing the rotation between the two coordinate systems, by solving what is known as Wahba's problem, see Wahba (1965). Wahba's problem is a least square optimization problem, where a cost function $J(\underline{\mathbf{A}})$ is minimized:

$$J(\underline{\mathbf{A}}) = \sum_{i=1}^{m} a_i \|\mathbf{b}_i - \underline{\mathbf{A}}\mathbf{r}_i\|^2. \tag{11}$$

Eq. (11) defines the cost function to be minimized, where $\mathbf{b}$ and $\mathbf{r}$ are vector observations in two different coordinate systems. The non-negative weight a can be chosen to be the inverse variance σ^{-2} of the measurement noise (Markley and Mortari (2000)). Eq. (11) can also be written in a more convenient form as shown in Markley and Mortari (2000); Shuster (2006):

$$J(\underline{\mathbf{A}}) = \sum_{i=1}^{m} a_i - trace\left(\underline{\mathbf{A}}\,\underline{\mathbf{B}}^{\mathrm{T}}\right), \tag{12}$$

where $\underline{\mathbf{B}} = \sum_{i=1}^{m} a_i \mathbf{b}_i \mathbf{r}_i^{\mathrm{T}}$. This reduces the problem to a question of maximizing $trace\left(\underline{\mathbf{A}}\,\underline{\mathbf{B}}^{\mathrm{T}}\right)$.

There are many ways of solving this problem. This paper focuses on the SVD-method, which is faster than the q-method for two vector observations and considered more robust than other faster methods, such as FOAM and ESOQ (Markley and Mortari (2000)).

The matrix $\underline{\mathbf{B}}$ has the singular value decomposition:

$$\underline{\mathbf{B}} = \underline{\mathbf{U}}\,\underline{\boldsymbol{\Sigma}}\,\underline{\mathbf{V}}^{\mathrm{T}} = \underline{\mathbf{U}}\,diag\left[\Sigma_{11} \;\; \Sigma_{22} \;\; \Sigma_{33}\right]\underline{\mathbf{V}}^{\mathrm{T}}, \tag{13}$$

where *diag* refers to a square matrix with zeros outside the main diagonal. The matrices $\underline{\mathbf{U}}$ and $\underline{\mathbf{V}}$ are orthogonal and the singular values obey $\Sigma_{11} \geq \Sigma_{22} \geq \Sigma_{33} \geq 0$ (Markley and Mortari (2000)). Then

$$trace\left(\underline{\mathbf{A}}\,\underline{\mathbf{B}}^{\mathrm{T}}\right) = trace\left(\underline{\mathbf{U}}^{\mathrm{T}}\,\underline{\mathbf{A}}\,\underline{\mathbf{V}}\,diag\left[\Sigma_{11} \;\; \Sigma_{22} \;\; \Sigma_{33}\right]\right). \tag{14}$$

Furthermore, according to Markley and Mortari (2000), the maximized trace is obtained by

$$\underline{\mathbf{U}}^{\mathrm{T}}\,\underline{\mathbf{A}}_{opt}\,\underline{\mathbf{V}} = diag\left[1 \;\; 1 \;\; det\left(\underline{\mathbf{U}}\right)det\left(\underline{\mathbf{V}}\right)\right], \tag{15}$$

which is consistent with the constraint $det(\underline{\mathbf{A}}) = 1$. This gives the optimal rotation matrix $\underline{\mathbf{A}}_{opt}$ defined as

$$\underline{\mathbf{A}}_{opt} = \underline{\mathbf{U}}\,diag\left[1 \;\; 1 \;\; det\left(\underline{\mathbf{U}}\right)det\left(\underline{\mathbf{V}}\right)\right]\underline{\mathbf{V}}^{\mathrm{T}}. \tag{16}$$

It is possible to examine how good the estimate $\underline{\mathbf{A}}_{opt}$ is, by calculating the covariance matrix $\underline{\mathbf{P}}$ (of the rotation angle error vector). According to Markley and Mortari (2000), $\underline{\mathbf{P}}$ can be calculated as

$$\underline{\mathbf{P}} = \underline{\mathbf{U}}\,diag\left[(s_2+s_3)^{-1} \;\; (s_3+s_1)^{-1} \;\; (s_1+s_2)^{-1}\right]\underline{\mathbf{U}}^{\mathrm{T}}, \tag{17}$$

with the three definitions $s_1 \equiv \Sigma_{11}$, $s_2 \equiv \Sigma_{22}$ and $s_3 \equiv det\left(\underline{\mathbf{U}}\right)det\left(\underline{\mathbf{V}}\right)\Sigma_{33}$. $\underline{\mathbf{P}}$ becomes infinite if the attitude is unobservable. If the satellite only has two sensors (e.g. sun and magnetic field sensor), then the SVD-method fails when the satellite is in eclipse and when the two observations are parallel. This makes the SVD-method unsuitable for continuous attitude estimation. However, it can be used as a sanity-check for filters like the UKF.

4. UNSCENTED KALMAN FILTER

The UKF is based on a sigma point sampling method called Unscented TransForm (UTF). Sigma points are a structured set of sample points selected in such a way, that the sigma points gives an adequate coverage of the input and output probability distribution. Grewal and Andrews (2008) describes different methods of calculating the sigma points, and the scaled symmetric sigma point method, providing extra adjustable scaling parameters, is used in this work. The sample size of the Symmetric UTF is $2L + 1$, where L is the number of states in the filter.

The UKF procedure is described in e.g. Haykin (2001); Grewal and Andrews (2008); Simon (2006) and is summarized in the following.

1) Use the error covariance matrix $\underline{\mathbf{P}}_{k-1|k-1}$ to calculate the sigma points $\left(\boldsymbol{\chi}_{k-1|k-1}\right)_i$:

$$\left(\boldsymbol{\chi}_{k-1|k-1}\right)_0 = \mathbf{x}_{k-1|k-1} \tag{18}$$

$$\left(\boldsymbol{\chi}_{k-1|k-1}\right)_i = \mathbf{x}_{k-1|k-1} + \left(\sqrt{(L+\lambda)\,\underline{\mathbf{P}}_{k-1|k-1}}\right)_i,\ i = 1, ..., L \tag{19}$$

$$\left(\boldsymbol{\chi}_{k-1|k-1}\right)_i = \mathbf{x}_{k-1|k-1} - \left(\sqrt{(L+\lambda)\,\underline{\mathbf{P}}_{k-1|k-1}}\right)_i,\ i = L+1, ..., 2L. \tag{20}$$

$\mathbf{x}_{k-1|k-1}$ is the state vector at time $k-1$ and $\lambda = \alpha^2(L+\kappa) - L$ is a scaling parameter. The constant α normally has a small positive value between $10^{-4} \leq \alpha \leq 1$ and it determines the spread of the sigma points, while the constant κ is normally set to $3-L$ (Haykin (2001)). However, for $\kappa < 0$ the resulting error covariance matrix, under the square root, cannot be guaranteed to be positive definite and κ is therefore often set to 0 (Grewal and Andrews (2008)). The property of $\underline{\mathbf{P}}_{k-1|k-1}$ being positive definite must be satisfied in order to solve $\sqrt{(L+\lambda)\,\underline{\mathbf{P}}_{k-1|k-1}}$ using Cholesky factorization (Simon (2006)).

2) Propagate all sigma points with the nonlinear system model and the input vector $\mathbf{u}_{k-1}$:

$$\left(\boldsymbol{\chi}_{k|k-1}\right)_i = f\left(\left(\boldsymbol{\chi}_{k-1|k-1}\right)_i, \mathbf{u}_{k-1}\right),\ i = 0, ..., 2L \tag{21}$$

3) Calculate the a priori state estimate $\mathbf{x}_{k|k-1}$ and the a priori error covariance matrix $\underline{\mathbf{P}}_{k|k-1}$:

$$\mathbf{x}_{k|k-1} = \sum_{i=0}^{2L} W_i^{(m)} \left(\boldsymbol{\chi}_{k|k-1}\right)_i \tag{22}$$

$$\underline{\mathbf{P}}_{k|k-1} = \sum_{i=0}^{2L} W_i^{(c)} \left(\left(\boldsymbol{\chi}_{k|k-1}\right)_i - \mathbf{x}_{k|k-1}\right)\left(\left(\boldsymbol{\chi}_{k|k-1}\right)_i - \mathbf{x}_{k|k-1}\right)^{\mathrm{T}} + \underline{\mathbf{Q}}_k. \tag{23}$$

The matrix $\underline{\mathbf{Q}}_k$ is the process noise covariance. The weights $W_i^{(m)}$ and $W_i^{(c)}$, used to calculated the a priori state estimate and the a priori error covariance matrix, are defined in Eq. (24), (25) and (26).

$$W_0^{(m)} = \frac{\lambda}{L+\lambda} \tag{24}$$

$$W_0^{(c)} = \frac{\lambda}{L+\lambda} + \left(1 - \alpha^2 + \beta\right) \tag{25}$$

$$W_i^{(m)} = W_i^{(c)} = \frac{1}{2\,(L+\lambda)},\ i = 1, ..., 2L \tag{26}$$

The scaling parameter β incorporates prior knowledge of the distribution of the state vector $\mathbf{x}_k$. According to Haykin (2001), the optimal value of β for a Gaussian distribution is 2.

4) Propagate the sigma points $\left(\boldsymbol{\chi}_{k|k-1}\right)_i$ through the sensor model in order to obtain the transformed sigma points $(\mathbf{Z}_{k|k-1})_i$ (predicted measurements):

$$(\mathbf{Z}_{k|k-1})_i = h\left(\left(\boldsymbol{\chi}_{k|k-1}\right)_i\right),\ i = 0, ..., 2L. \tag{27}$$

The transformed measurement vector $\mathbf{z}_{k|k-1}$ is then calculated as

$$\mathbf{z}_{k|k-1} = \sum_{i=0}^{2L} W_i^{(m)} \left(\mathbf{Z}_{k|k-1}\right)_i. \tag{28}$$

5) Calculate the a posteriori state estimate $\mathbf{x}_{k|k}$ using the measurement vector $\mathbf{z}_k$, containing measurements obtained at time k:

$$\mathbf{x}_{k|k} = \mathbf{x}_{k|k-1} + \underline{\mathbf{K}}_k \left(\mathbf{z}_k - \mathbf{z}_{k|k-1}\right). \tag{29}$$

The Kalman gain $\underline{\mathbf{K}}_k$ is calculated as

$$\underline{\mathbf{K}}_k = \underline{\mathbf{P}}_{x_k z_k} \underline{\mathbf{P}}_{z_k z_k}^{-1}, \tag{30}$$

where the measurement covariance matrix $\underline{\mathbf{P}}_{z_k z_k}$ and the state-measurement cross-covariance matrix $\underline{\mathbf{P}}_{x_k z_k}$ are calculated as in Eq. (31) and (32) respectively, where the matrix $\mathbf{R}_k$ is the measurement noise covariance matrix.

$$\underline{\mathbf{P}}_{z_k z_k} = \sum_{i=0}^{2L} W_i^{(c)} \left(\left(\mathbf{Z}_{k|k-1}\right)_i - \mathbf{z}_{k|k-1}\right) \left(\left(\mathbf{Z}_{k|k-1}\right)_i - \mathbf{z}_{k|k-1}\right)^{\mathrm{T}} + \underline{\mathbf{R}}_k \tag{31}$$

$$\underline{\mathbf{P}}_{x_k z_k} = \sum_{i=0}^{2L} W_i^{(c)} \left(\left(\boldsymbol{\chi}_{k|k-1}\right)_i - \mathbf{x}_{k|k-1}\right) \left(\left(\mathbf{Z}_{k|k-1}\right)_i - \mathbf{z}_{k|k-1}\right)^{\mathrm{T}} \tag{32}$$

6) Calculate the a posteriori error covariance $\underline{\mathbf{P}}_{k|k}$:

$$\underline{\mathbf{P}}_{k|k} = \underline{\mathbf{P}}_{k|k-1} - \underline{\mathbf{K}}_k \underline{\mathbf{P}}_{z_k z_k} \underline{\mathbf{K}}_k^{\mathrm{T}}. \tag{33}$$

The a posteriori error covariance is used in the next filter iteration to calculate the sigma points.

Points 1) to 3) are the predict steps, where the states are propagated based on the system equations and points 4) to 6) are the update steps, where measurement are used to correct the prediction.

5. ATTITUDE ESTIMATION FOR AAUSAT3

Four attitude estimators have been designed and tested in a Matlab Simulink simulation environment made for the AAUSAT3 CubeSat. The first attitude estimator is a deterministic single point SVD-method implementation solving Wahba's problem, which was addressed in Section 3. The second attitude estimator is a quaternion UKF implementation without bias estimation having the state vector

$$\mathbf{x} = \begin{bmatrix} {}^{c}_{i}\mathbf{q}^{\mathrm{T}} & {}^{c}\boldsymbol{\omega}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \tag{34}$$

and the third and fourth are UKF implementations that also estimate the bias in a 3-axis magnetometer ($\mathbf{b}_{mag}$) and a 3-axis gyroscope ($\mathbf{b}_{gyro}$) and a similar implementation that does not include the quaternion ${}^{c}_{s}\mathbf{q}$, which reduces the amount of computations, thus taking advantage of the fact that one unit CubeSats are close to symmetric. The third and fourth UKF implementation have the state vector

$$\mathbf{x} = \begin{bmatrix} {}^{c}_{i}\mathbf{q}^{\mathrm{T}} & {}^{c}\boldsymbol{\omega}^{\mathrm{T}} & {}^{c}\mathbf{b}_{mag}^{\mathrm{T}} & {}^{c}\mathbf{b}_{gyro}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \tag{35}$$

The quaternion has the advantage that it is singularity free, but there is a unity constraint on the four parameters. This means that the unit quaternion does not belong to a vector space, but belongs to a sphere in $\mathbb{R}^4$, where it is not possible to calculate

the mean of a set of unit quaternions by a weighted sum, since it is not closed for addition and scalar multiplication. The resulting quaternion can not be guaranteed to be a unit quaternion.

One way of solving this problem is to introduce an error quaternion $\delta\mathbf{q}$ defined by Eq. (36) and (37) as in e.g. Bak (1999); Cheon (2005); Kraft (2003); Lefferts et al. (1982); VanDyke et al. (2004).

$$\delta\bar{\mathbf{q}}_{k|k} = \mathbf{q}_{k|k} \otimes \mathbf{q}_{k|k-1}^{-1} \tag{36}$$

$$\delta\mathbf{q}_{k|k} = \begin{bmatrix}\delta q_1 & \delta q_2 & \delta q_3\end{bmatrix}^{\mathrm{T}} \tag{37}$$

The error quaternion $\delta\mathbf{q}_{k|k}$ is the update to the predicted quaternion $\mathbf{q}_{k|k-1}$, giving the estimated quaternion $\mathbf{q}_{k|k}$. By using a multiplicative update step, with the quaternion error state, it is possible to maintain the unity constraint, as in Eq. (38) and (39).

$$\delta\mathbf{q}_{k|k} = \underline{\mathbf{K}}_k \left(\mathbf{z}_k - \mathbf{z}_{k|k-1}\right) \tag{38}$$

$$\mathbf{q}_{k|k} = \begin{bmatrix}\delta\mathbf{q}_{k|k}^{\mathrm{T}} & \sqrt{1 - \delta\mathbf{q}_{k|k}^{\mathrm{T}} \cdot \delta\mathbf{q}_{k|k}}\end{bmatrix}^{\mathrm{T}} \otimes \mathbf{q}_{k|k-1} \tag{39}$$

The three-element error quaternion is expanded to a four-element quaternion in Eq. (39) as in e.g. Bak (1999); VanDyke et al. (2004).

The observations given as input to the attitude estimators are measurements given in the satellite body reference frame, which are a sun vector ${}^{s}\mathbf{v}_{sun,k}$, a magnetic field vector ${}^{s}\mathbf{v}_{mag,k}$ and angular velocity ${}^{s}\boldsymbol{\omega}_k$ and predicted measurements in the Earth centered inertial frame, which are a sun vector ${}^{i}\mathbf{v}_{sun,k|k-1}$ and a magnetic field vector ${}^{i}\mathbf{v}_{mag,k|k-1}$. The UKF implementation also incorporates an eclipse indication $E_{ecl,k}$. This boolean variable is calculated by an ephemeris model and an orbit propagator. Table 1 presents the implemented UKF in pseudo code, where rotation of vectors $\mathbf{v}$ with quaternions are written as $\underline{\mathbf{A}}(\mathbf{q})\mathbf{v}$.

The SVD-method is utilized in the initialization of the UKF, where an initial quaternion estimate ${}^{s}_{i}\mathbf{q}_0$ is calculated. This ensures faster convergence of the UKF, but it should be noted that this is only possible when two or more non-parallel vector observations are available. The singular value decomposition algorithm is denoted *svd* in Table 1 and numerical implementations can easily be found in the literature and on the Internet, e.g. in the GNU scientific library.

It is only necessary to calculate the weights once and this is done in the initialization. These weights are determined by the scaling parameters α, β and κ, which was set to $\sqrt{3}$, 2 and 0, respectively, which gives weights with a total sum of 1 $\left(\sum_{i=0}^{2L} W_i^{(m)} = 1,\ \sum_{i=0}^{2L} W_i^{(c)} = 1\right)$. The scaling parameters are tuning parameters and the values chosen are based on filter performance results and the guidelines in Section 4.

Calculating sigma points involves finding the matrix square root and an efficient way of calculating this is done by use of a Cholesky decomposition, as described in e.g. Cheon (2005); Crassidis and Markley (2003); LaViola (2003). In the implementation a lower triangular Cholesky factorization is used and this is denoted *ch* in Table 1. The parameter $L = 6$ is the number of error states (three quaternion parameters and three angular velocities). This gives a total of 13 scaled symmetric error sigma points distributed around zero, where the first one is zero, as presented in Table 1. These error sigma points are then expanded with quaternion multiplication and angular velocity addition, using the previous attitude estimate $\mathbf{x}_{k-1|k-1}$.

In the predict step, the state is propagated using the non-linear satellite equations of motion defined in Eq. (7) and (9). This can be done by numerically integrating the continuous time functions over a period of *Ts* using a fourth order Runge Kutta implementation as in e.g. LaViola (2003). Since using a Runge Kutta method (denoted RK4 in step 1.4 in Table 1) involves addition in each sub-step, a normalization of the quaternion is required. If the time step *Ts* is large and the angular velocity is high, it might be necessary to use more sub-steps within the Runge Kutta implementation and/or normalize after each sub-step rather than only after the last sub-step. Ten sub-steps was used with normalization after each sub-step in the implementation. The predict step has to be repeated for all the sigma points. The a priori full state estimate is then calculated as a weighted sum of all the propagated sigma points, with quaternion normalization, as defined in step 1.5. This approximation of the mean quaternion is also performed in e.g. Cheon (2005).

A normalization of the sun and the magnetic field vectors are performed, since the difference in length of these vectors is very large, leading to numerical problems in matrix computations. Normalizing of vector information is suggested, since it is only the direction of the vectors and not the magnitude that is important for attitude estimation. Sun vector measurements can only be considered valid when the satellite is not in eclipse. The sun vector measurement and prediction is therefore set equal to each other during eclipse, which is equivalent to stating that there is no discrepancy between the two, and thus no update is performed on the state in the update step. The same is the case if there is no new measurements available, which happens when one sensor is sampled slower than each time step *Ts*. The state should only be updated by the particular measurement if it has changed since last filter iteration. The normalization of vectors, eclipse check and new measurement check is performed in steps 2.1 - 2.6, where the measurements and predicted measurements are also rotated to the controller reference frame. The predicted measurements are gathered with a weighted sum in step 2.5 and this result is used in step 2.10.

The UKF implementation with bias estimation is very similar to the pseudo code in Table 1. The only difference is that the state vector is expanded by six states (see Eq. (35)) and the bias is approximated as being constant during propagation. Furthermore, simulations have shown that the magnetometer bias should only be updated in the update step when both sun and magnetic field vector measurements are available. This means that the UKF implementation should be started outside

Table 1. Pseudo code for quaternion UKF implementation without bias estimation.

Init.:	0.1	Solve Wahba (SVD):	$\underline{\mathbf{B}} = \sigma_{sun}^{-2}\, {}^{s}\mathbf{v}_{sun,k}\, {}^{i}\mathbf{v}_{sun,k|k-1}^{\mathrm{T}} + \sigma_{mag}^{-2}\, {}^{s}\mathbf{v}_{mag,k}\, {}^{i}\mathbf{v}_{mag,k|k-1}^{\mathrm{T}}$ $svd\left(\underline{\mathbf{B}}\right) = \left[\underline{\mathbf{U}}, \underline{\mathbf{S}}, \underline{\mathbf{V}}\right],\ {}_{i}^{s}\underline{\mathbf{A}}_{opt} = \underline{\mathbf{U}}\, diag\left[1 \quad 1 \quad det\left(\underline{\mathbf{U}}\right) det\left(\underline{\mathbf{V}}\right)\right] \underline{\mathbf{V}}^{\mathrm{T}}$
	0.2	Calculate ${}_{i}^{s}\mathbf{q}_0$:	${}_{i}^{s}\underline{\mathbf{A}}_{opt}$ to ${}_{i}^{s}\mathbf{q}_0$ conversion
	0.3	Initialize state:	$\mathbf{x}_{k|k} = \left[\left({}_{i}^{s}\mathbf{q}_0 \otimes {}_{s}^{c}\mathbf{q}\right)^{\mathrm{T}} \quad \left(\underline{\mathbf{A}}\left({}_{s}^{c}\mathbf{q}\right){}^{s}\boldsymbol{\omega}_0\right)^{\mathrm{T}}\right]^{\mathrm{T}}$
	0.4	Calculate weights:	$W_0^{(m)} = \frac{\lambda}{L+\lambda},\ W_0^{(c)} = \frac{\lambda}{L+\lambda} + \left(1 - \alpha^2 + \beta\right),\ W_i^{(m)} = W_i^{(c)} = \frac{1}{2(L+\lambda)},\ i = 1, ..., 2L$
	0.5	Save:	${}^{s}\mathbf{v}_{sun,k}, {}^{s}\mathbf{v}_{mag,k}, {}^{s}\boldsymbol{\omega}_k, {}^{s}\mathbf{N}_{ctrl,k}$ wait Ts seconds and go to 1.1
Predict:	1.1	Error sigma points:	$\delta\underline{\boldsymbol{\chi}}_{k-1|k-1} = \left[\underline{\mathbf{0}}_{6x1} \quad -\left(ch\left(\left(L+\lambda\right)\underline{\mathbf{P}}_{k-1|k-1}\right)\right)^{\mathrm{T}} \quad \left(ch\left(\left(L+\lambda\right)\underline{\mathbf{P}}_{k-1|k-1}\right)\right)^{\mathrm{T}}\right]$
	1.2	Full sigma points:	$\left(\boldsymbol{\chi}_{k-1|k-1}\right)_i = \left[\left[\left[\left(\delta\underline{\boldsymbol{\chi}}_{k-1|k-1}\right)_{1-3,i}^{\mathrm{T}} \quad \sqrt{1 - \left(\delta\underline{\boldsymbol{\chi}}_{k-1|k-1}\right)_{1-3,i}^{\mathrm{T}} \left(\delta\underline{\boldsymbol{\chi}}_{k-1|k-1}\right)_{1-3,i}}\right]^{\mathrm{T}}\right.\right.$
		$(\cdot)_{1-3,i}$ = row 1-3, col. i	$\left.\left.\otimes\, {}_{i}^{c}\mathbf{q}_{k-1|k-1}\right)^{\mathrm{T}} \quad \left({}^{c}\boldsymbol{\omega}_{k-1|k-1} + \left(\delta\underline{\boldsymbol{\chi}}_{k-1|k-1}\right)_{4-6,i}\right)^{\mathrm{T}}\right]^{\mathrm{T}},\ i = 1, ..., 2L+1$
	1.3	Rotate control torque:	${}^{c}\mathbf{N}_{ctrl,k-1} = \underline{\mathbf{A}}\left({}_{s}^{c}\mathbf{q}\right){}^{s}\mathbf{N}_{ctrl,k-1}$
	1.4	Numerical propagation:	$\left(\boldsymbol{\chi}_{k|k-1}\right)_i = \mathrm{RK4}\left(\left(\boldsymbol{\chi}_{k-1|k-1}\right)_i, {}^{c}\mathbf{N}_{ctrl,k-1}, Ts, steps\right),\ i = 1, ..., 2L+1$
	1.5	A priori state estimate:	$\mathbf{x}_{k|k-1} = \left[\left(\frac{\sum_{i=0}^{2L} W_i^{(m)} \left(\boldsymbol{\chi}_{k|k-1}\right)_{1-4,i+1}}{\left\|\sum_{i=0}^{2L} W_i^{(m)} \left(\boldsymbol{\chi}_{k|k-1}\right)_{1-4,i+1}\right\|}\right)^{\mathrm{T}} \quad \left(\sum_{i=0}^{2L} W_i^{(m)} \left(\boldsymbol{\chi}_{k|k-1}\right)_{5-7,i+1}\right)^{\mathrm{T}}\right]^{\mathrm{T}}$
	1.6	Full to error state:	$\left(\delta\boldsymbol{\chi}_{k|k-1}\right)_i = \left[\left(\left(\boldsymbol{\chi}_{k|k-1}\right)_{1-4,i} \otimes {}_{i}^{c}\mathbf{q}_{k|k-1}^{-1}\right)_{1-3,i}^{\mathrm{T}} \quad \left(\left(\boldsymbol{\chi}_{k|k-1}\right)_{5-7,i} - {}^{c}\boldsymbol{\omega}_{k|k-1}\right)^{\mathrm{T}}\right]^{\mathrm{T}},\ i = 1, ..., 2L+1$
	1.7	Mean error state:	$\delta\mathbf{x}_{k|k-1} = \sum_{i=0}^{2L} W_i^{(m)} \left(\delta\boldsymbol{\chi}_{k|k-1}\right)_{i+1}$
	1.8	A priori covariance:	$\underline{\mathbf{P}}_{k|k-1} = \sum_{i=0}^{2L} W_i^{(c)} \left(\left(\delta\boldsymbol{\chi}_{k|k-1}\right)_{i+1} - \delta\mathbf{x}_{k|k-1}\right)\left(\left(\delta\boldsymbol{\chi}_{k|k-1}\right)_{i+1} - \delta\mathbf{x}_{k|k-1}\right)^{\mathrm{T}} + \underline{\mathbf{Q}}$
Update:	2.1	Save:	${}^{s}\mathbf{v}_{sun,k}, {}^{s}\mathbf{v}_{mag,k}, {}^{s}\boldsymbol{\omega}_k, {}^{s}\mathbf{N}_{ctrl,k}$
	2.2	Eclipse check:	If in eclipse go to 2.6, else go to 2.3
	2.3	New measurement?:	If new vector measurement go to 2.4, else go to 2.6
	2.4	Normalize and rotate:	${}^{c}\mathbf{v}_{sun,k} = \underline{\mathbf{A}}\left({}_{s}^{c}\mathbf{q}\right) \frac{{}^{s}\mathbf{v}_{sun,k}}{\|{}^{s}\mathbf{v}_{sun,k}\|}$
	2.5	Estimate measurement:	$\left({}^{c}\mathbf{v}_{sun,k|k-1}\right)_i = \underline{\mathbf{A}}\left(\left(\boldsymbol{\chi}_{k|k-1}\right)_{1-4,i}\right) \frac{{}^{i}\mathbf{v}_{sun,k|k-1}}{\|{}^{i}\mathbf{v}_{sun,k|k-1}\|},\ i = 1, ..., 2L+1$
			${}^{c}\mathbf{v}_{sun,k|k-1} = \sum_{i=0}^{2L} W_i^{(m)} \left({}^{c}\mathbf{v}_{sun,k|k-1}\right)_{i+1}$ go to 2.7
	2.6	Hardcode vector:	${}^{c}\mathbf{v}_{sun,k} = {}^{c}\mathbf{v}_{sun,k|k-1} = [0 \quad 0 \quad 0]^{\mathrm{T}}$
	2.7	Repeat step 2.3 - 2.6:	For magnetic field vectors (${}^{c}\mathbf{v}_{mag,k}$, ${}^{c}\mathbf{v}_{mag,k|k-1}$) and angular velocities (${}^{c}\boldsymbol{\omega}_k$, ${}^{c}\boldsymbol{\omega}_{k|k-1}$)
	2.8	Calculate covariances:	$\underline{\mathbf{P}}_{x_k z_k}$ and $\underline{\mathbf{P}}_{z_k z_k}$ according to Eq. (32) and (31)
	2.9	Calculate Kalman gain:	$\underline{\mathbf{K}}_k = \underline{\mathbf{P}}_{x_k z_k} \underline{\mathbf{P}}_{z_k z_k}^{-1}$
	2.10	Calculate error state:	$\delta\mathbf{x}_{k|k} = \underline{\mathbf{K}}_k \left(\mathbf{z}_k - \mathbf{z}_{k|k-1}\right)$
	2.11	Expand quaternion:	${}_{i}^{c}\mathbf{q}_{k|k} = \left[\delta\mathbf{q}_{k|k}^{\mathrm{T}} \quad \sqrt{1 - \delta\mathbf{q}_{k|k}^{\mathrm{T}} \cdot \delta\mathbf{q}_{k|k}}\right]^{\mathrm{T}} \otimes {}_{i}^{c}\mathbf{q}_{k|k-1}$
	2.12	Calculate full state:	$\mathbf{x}_{k|k} = \left[{}_{i}^{c}\mathbf{q}_{k|k}^{\mathrm{T}} \quad \left({}^{c}\boldsymbol{\omega}_{k|k-1} + \delta\boldsymbol{\omega}_{k|k}\right)^{\mathrm{T}}\right]^{\mathrm{T}}$
	2.13	A posteori covariance:	$\underline{\mathbf{P}}_{k|k} = \underline{\mathbf{P}}_{k|k-1} - \underline{\mathbf{K}}_k \underline{\mathbf{P}}_{z_k z_k} \underline{\mathbf{K}}_k^{\mathrm{T}}$
	2.14	Rotate and output:	Output = $\left[\left({}_{i}^{c}\mathbf{q}_{k|k} \otimes {}_{s}^{c}\mathbf{q}^{-1}\right)^{\mathrm{T}} \quad \left(\underline{\mathbf{A}}\left({}_{s}^{c}\mathbf{q}^{-1}\right){}^{c}\boldsymbol{\omega}_{k|k}\right)^{\mathrm{T}}\right]^{\mathrm{T}}$
	2.15	Repeat:	Go to 1.1 and iterate filter every time step Ts

eclipse, the first time it is run, and the bias should be saved as a starting guess if the filter goes through on/off cycles. In theory, it is possible to estimate bias in the magnetometer during eclipse if the satellite is in motion, since the changing direction of the magnetic field measurement will make the constant bias term detectable. However, having less vector measurements to update the estimate with, requires a different tuning of the UKF, where the designer trusts the model more than the measurement, which again, according to simulations, gives poorer overall performance, since the model is subject to disturbance torques and parameter uncertainty.

6. SIMULATION ENVIRONMENT

The simulation environment made for AAUSAT3 has been implemented in Matlab as a Simulink library. A "truth model" simulates the position of the Earth, the Sun, the Moon and the satellite. The position of the satellite is simulated with a Two Line Element (TLE) and a Simplified General Perturbations satellite orbit model 4 (SGP4). The TLE used in the simulations is for the AAUSAT-II satellite, which is in a polar orbit with an inclination of approximately 98 degrees and an orbit height of approximately 630 km. This low Earth orbit

is considered typical for CubeSats. The "truth model" also simulates disturbance torques from atmospheric drag, solar radiation, Earth gravity and magnetic residuals. The satellite equations of motion, presented in Section 2, provides the simulated attitude and angular velocity of the satellite and an International Geomagnetic Reference model generation 10 (IGRF10), implemented with order 13, simulates the Earth's magnetic field.

The simulation environment also provides what is considered satellite onboard software. This software consists of another IGRF10 magnetic field model, implemented with order 8, that, together with a SGP4 orbit propagator, provide the predicted magnetic field vector in the Earth centered inertial reference frame. It also provides a sun vector and eclipse indication with ephemeris models and the SGP4. Using an 8th order IGRF10 is also done in Bak (1999) and gives a small discrepancy of up to approximately ±100 nT between the "truth model" and the onboard software. The onboard software does not estimate any of the disturbance torques; however, the torque from a permanent magnet is simulated and given as control input torque to the UKF. This permanent magnet has a magnetic moment of 0.0030 Am^2 and will be part of the ADCS for AAUSAT3.

The sensor setup chosen for AAUSAT3 was simulated by taking the output from the "truth model" and adding Gaussian noise. The sun vector, the magnetic field vector and the angular velocity was added with Gaussian noises with standard deviations of 3.33 degrees, 3 degrees and 0.2 degrees/s respectively. The size of these standard deviations are pessimistically choosen based on datasheets, since no real measurements have yet been obtained from the choosen hardware. A standard deviation of 3.33 degree corresponds to having approximately 99% of the measurements within ±10 degree. Consideration to power, size and cost of sensors was a key issue when choosing sensors, which is why the performance of the sensors are not in the high end. The chosen gyroscope for AAUSAT3 measures the angular velocity directly using the Coriolis effect and therefore has no simulated drift term. Furthermore, temperatures are measured either in or at each sensor to be able to compensate for any temperature drift.

The model noise covariance matrix $\underline{\mathbf{Q}}$, measurement noise covariance matrix $\underline{\mathbf{R}}$ and initial error covariance matrix $\underline{\mathbf{P}}_0$ was set to the following based on sensor noise and iterative testing (*diag* refers to a diagonal matrix):

$$\underline{\mathbf{Q}} = diag\,[1\ \ 1\ \ 1\ \ 10\ \ 10\ \ 10\ \ 1\ \ 1\ \ 1\ \ 0.01\ \ 0.01\ \ 0.01]\,10^{-6}$$
$$\underline{\mathbf{R}} = diag\,[3.4\ \ 3.4\ \ 3.4\ \ 2.7\ \ 2.7\ \ 2.7\ \ 0.012\ \ 0.012\ \ 0.012]\,10^{-3}$$
$$\underline{\mathbf{P}}_0 = diag\,[1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 0.1\ \ 0.1\ \ 0.1\ \ 1\ \ 1\ \ 1]\,10^{-3}.$$

The last 6 diagonal elements in $\underline{\mathbf{Q}}$ and $\underline{\mathbf{P}}_0$ are only necessary in the UKF implementation with bias estimation.

Attitude estimation is often performed after the satellite has been detumbled and the initial angular velocities in the tests were therefore set to (0.02,0.02,0.02) rad/s, which is within a realistic bound of what can be expected. The sampling frequency was set to 1 Hz.

A number of simulations were performed on the implementations and these are listed in Table 2. Realistic sensor biases were applied in some of the simulations. These biases were randomly chosen and different from the ones used during tuning of the filter. The mass property and thus the inertia was changed in the second last simulation so that the center of mass was located approximately 20 mm from the geometric center of the satellite, which was just within the CubeSat requirement (center of mass moved from (-0.9,-1.1,-12.0) mm to (-0.7,-0.8,20.0) mm relative to the geometric center). The quaternion ${}^{c}_{s}\mathbf{q}$, that represents the rotation from the satellite body reference frame to the controller reference frame, is removed from the implementation in the last simulation.

A final implementation of the UKF is based on these simulations and followed up with a Monte Carlo simulation. 1000 simulations have been run, while the following was varied randomly with uniform distribution:

- The time used in the onboard software was varied between ±5 seconds giving a realistic satellite position error.
- A random initial attitude was choosen and the initial angular velocity was varied between ±(0.02,0.02,0.02) rad/s.
- Bias in the magnetometer measurement was varied between ±10000 nT (possible bias after on ground calibration).
- Bias in the gyroscope was varied between ±0.4 rad/s (possible bias after on ground calibration).
- All elements in the inertia matrix were varied up to 10% and the center of mass was located within a sphere with a radius of 5 mm from the calculated value of (-0.9,-1.1,-12.0) mm from the geometric center. The variations represents the uncertainty associated with simple calculations of the inertia and center of mass of the satellite. The mass of the satellite was constant in the simulations, since it is easy to measure it precisely.

7. RESULTS

Fig. 1 shows the results of a simulation of a stand-alone implementation of the SVD-method for approximately one orbit. The SVD-method failed during eclipse (from 2100 to 4090 s), which was expected, because the sun vector is not measured, thus making the attitude unobservable. The sum of elements in the covariance matrix was also simulated and this graph has been limited to 0.3, but it was infinite during eclipse.

Fig. 2 shows the results of a simulation of the implementation of the quaternion UKF with (Graph 2 and 4) and without (Graph 1 and 3) bias estimation for approximately one orbit. There was no measurement bias in the simulations shown on Graph 1 and 2 and a small bias, specified in Section 6, was applied in the simulations shown on Graph 3 and 4.

Fig. 3 shows the results of a simulation of the implementation of the quaternion UKF with bias estimation for approximately one orbit. Graph 1 is a situation where the inertia in the UKF had a small deviation compared to the one used in

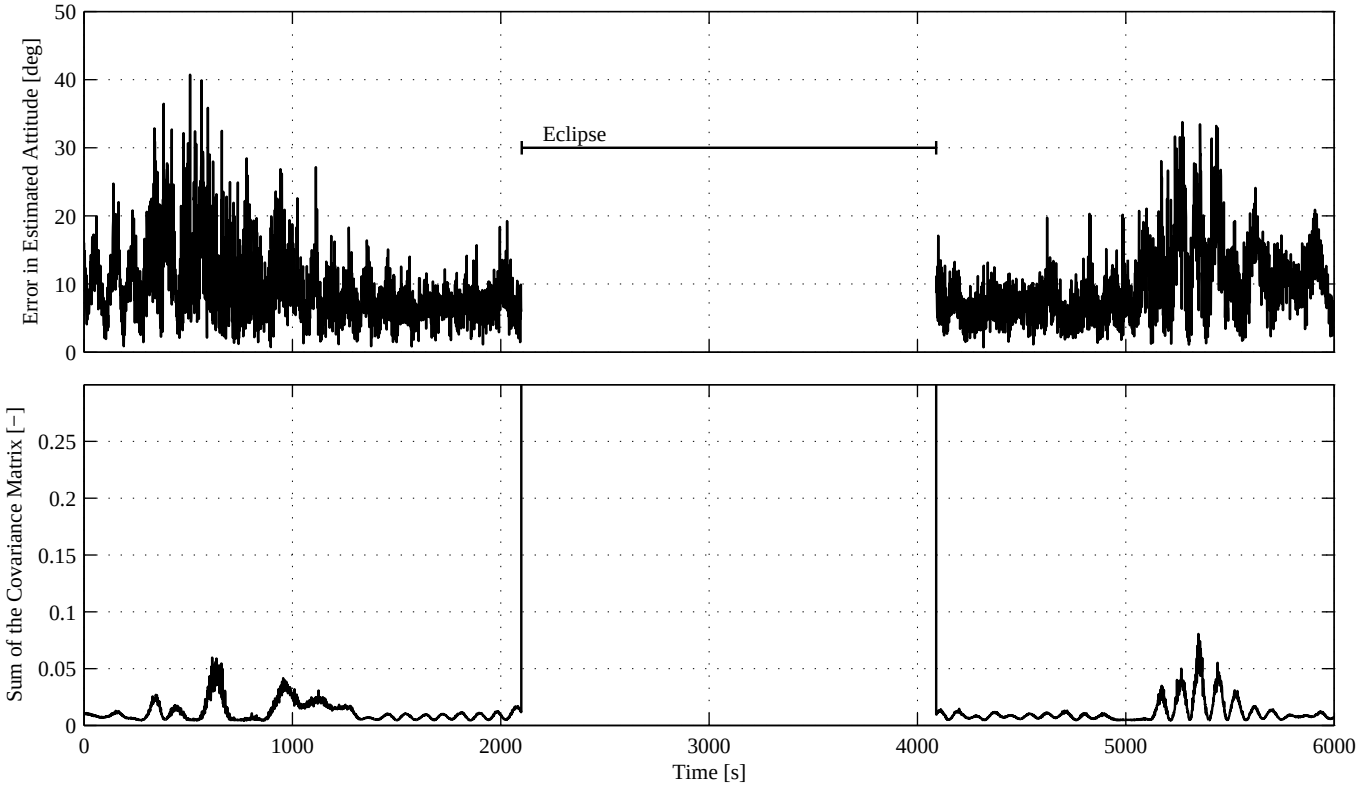

Fig. 1. Graph 1: Difference between the estimated attitude from the SVD-method and the attitude from the "truth model". Graph 2: Sum of elements in covariance matrix calculated according to Eq. (17).

the "truth model", and Graph 2 was a situation where all rotations between the controller reference frame and satellite body reference frame, represented by the quaternion ${}^{c}_{s}\mathbf{q}$, were removed from the UKF implementation.

The largest angular deviation has been identified for all simulations and classified into time regions. The first region from 0 to 1000 s is defined as the convergence phase. The second region is from 1001 s to 2100 (until eclipse), the third region is from 2101 to 4095, which is inside eclipse and 5 seconds after and the last region is after eclipse from 4096 to 6000 s. Table 2 presents these angular deviations.

Average computation times was recorded with the `tic-toc` commands in Matlab for the four different implementations and the results are presented in Table 2.

Finally, Fig. 3 shows the results of the Monte Carlo simulation of the quaternion UKF with bias estimation and only one Runge Kutta sub-step, which is the implementation proposed for AAUSAT3.

8. DISCUSSION

The stand-alone implementation of the SVD-method does provide an attitude estimate that stays within approximately 40 degrees during the simulated orbit. However, this performance is far from the performance of the quaternion UKF implementation and it does not work during eclipse. The error covariance matrix $\underline{\mathbf{P}}$ can be used to evaluate the performance of the SVD-method, where a large value of $\underline{\mathbf{P}}$ indicates a poor estimate of the attitude. Graph 2 in Fig. 1 shows a very large $\underline{\mathbf{P}}$ during eclipse and also relatively large values when the satellite passes the poles, which is where the vector observations are expected to be close to parallel. More precisely, a large $\underline{\mathbf{P}}$ will not occur over the poles, but just before and just after the poles, where the magnetic field lines are approximately tangent to the polar orbit and approximately parallel/anti-parallel to the sun vector. A large $\underline{\mathbf{P}}$ value is e.g. observed at 665 seconds, where the angle between the magnetic field vector and the sun vector, both given in the Earth centered inertial reference frame, was 22.53 degrees. This is the smallest observed angle between these vectors during the simulation. Calculating $\underline{\mathbf{P}}$ thus gives a measure of the confidence in the attitude estimate. Simulations have additionally shown that the quaternion UKF implementation with bias estimation has trouble converging if it is not supplied with a fairly accurate initial quaternion from e.g. the SVD-method. This is due to the chosen parameter tuning of the UKF, which is not too aggressive for better performance after the filter has converged and found a good sensor bias estimate. The tuning could be varied during simulation; however, it can be difficult to know when the filter has converged properly and providing the UKF implementation with an initial quaternion from the SVD-method gives a more robust implementation. Implementing the SVD-method also introduces the possibility of making sanity checks on the UKF output. Furthermore, additional increase in performance could possibly be obtained, by using Monte Carlo simulation to tune the filter parameters, instead of manual tuning as done in this paper.

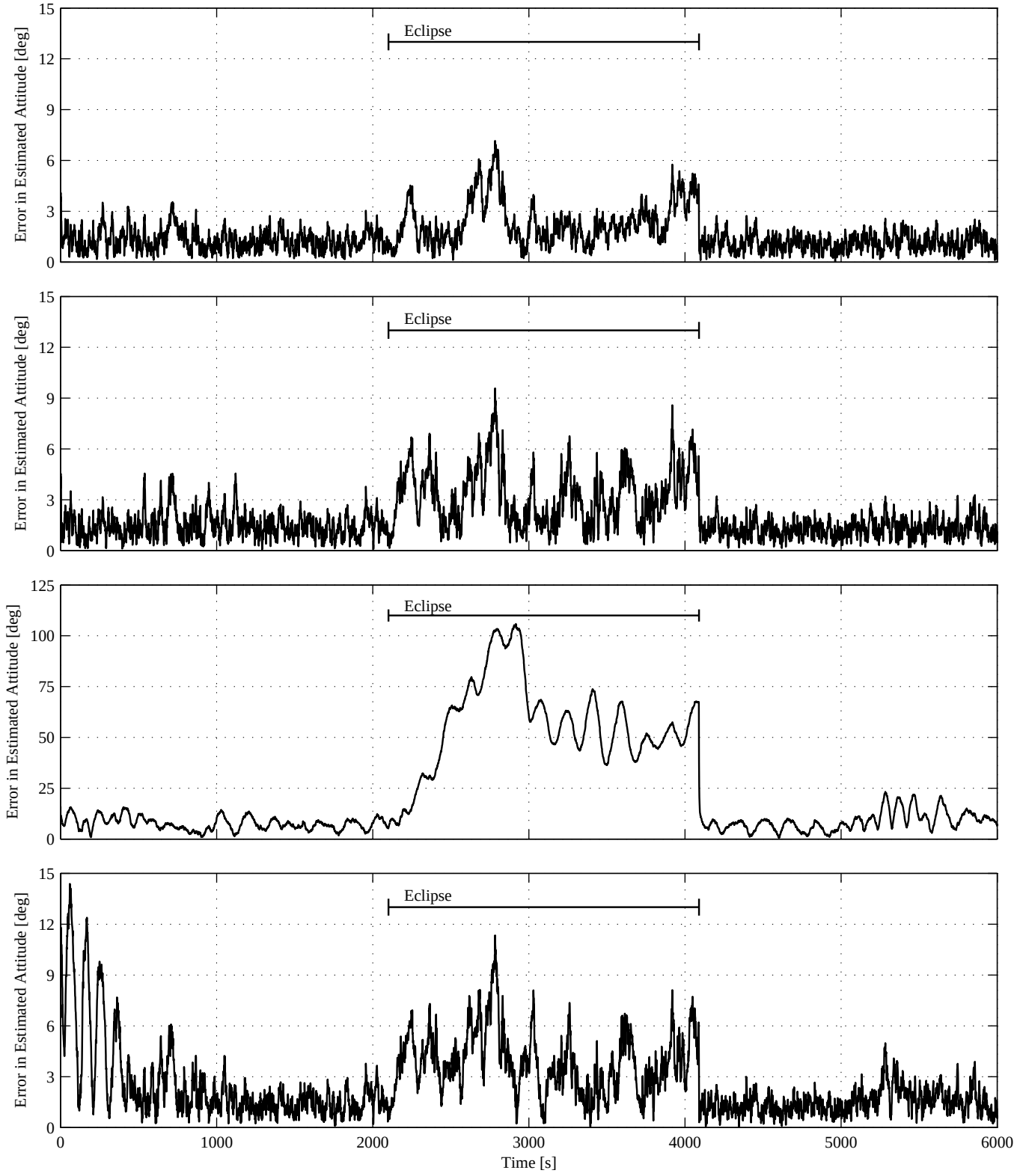

Fig. 2. Difference between the estimated attitude and the attitude from the "truth model". Graph 1: UKF without bias estimation and no bias on measurements. Graph 2: UKF with bias estimation and no bias on measurements. Graph 3: UKF without bias estimation and with bias on measurements. Graph 4: UKF with bias estimation and with bias on measurements.

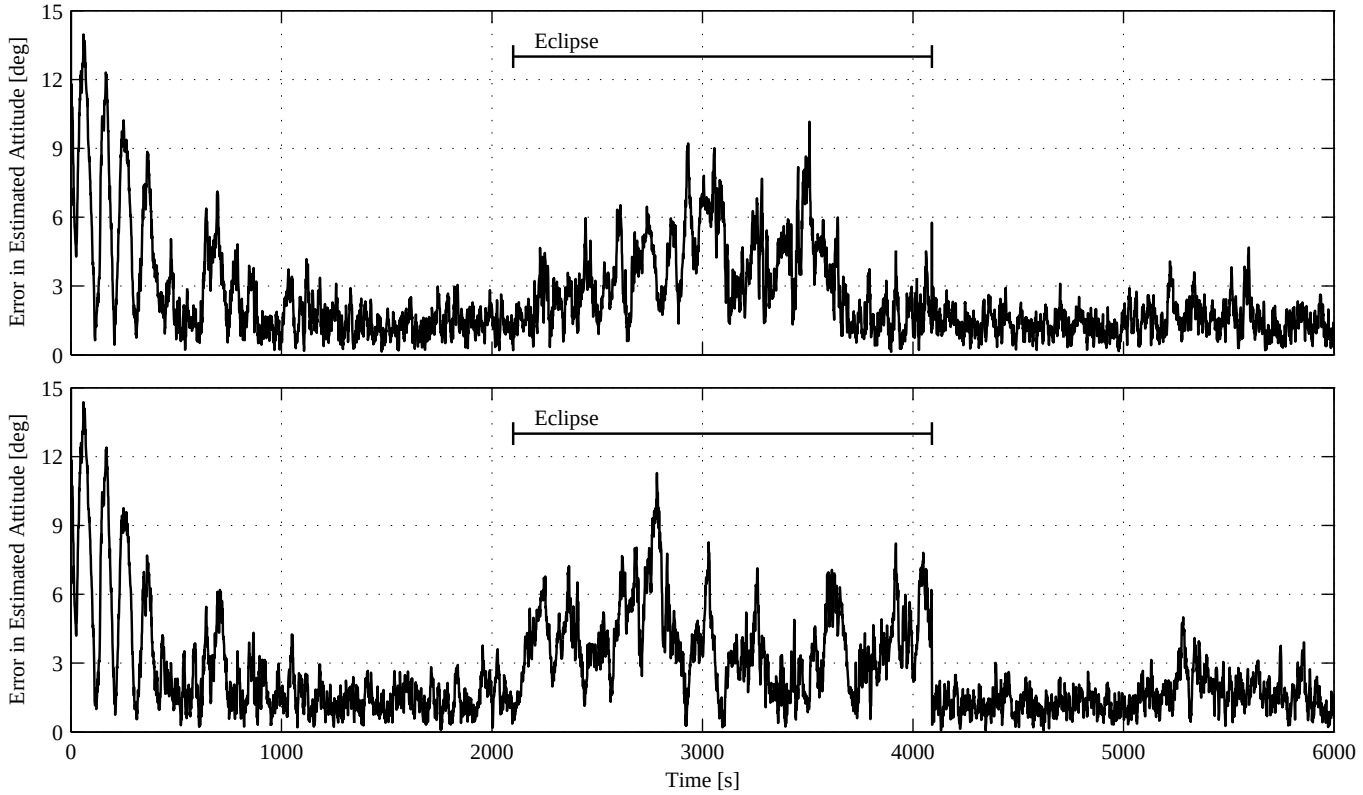

Fig. 3. Difference between the estimated attitude and the attitude from the "truth model". Graph 1: UKF with bias estimation, with bias on measurements and with a small inertia difference. Graph 2: UKF with bias estimation, without ${}^{c}_{s}\mathbf{q}$ and with bias on measurements.

Table 2. Simulation results for one orbit. Column 1-5: Simulated case. Column 6-9: Largest angular difference between the estimated attitude and the attitude from the "truth model" (Euler axis and angle representation). Err. 1 is between 0-1000 s, Err. 2 is between 1001-2100 s, Err. 3 is between 2101-4095 s and Err. 4 is between 4096-6000 s. Column 10: Average computation times per iteration for the implemented attitude estimators in Matlab on a single core Pentium 4 (2.8 GHz, 1 GB RAM).

Implementation	**Ref. [Fig.,Graph]**	**Mag. Bias [nT]**	**Gyro. Bias. [deg/s]**	**Inertia**	**Err. 1 [deg]**	**Err. 2 [deg]**	**Err. 3 [deg]**	**Err. 4 [deg]**	**Time [ms]**
Stand-alone SVD-method	(1,1)	$(5,1,-3)\cdot 10^3$	N/A	N/A	40.72	27.16	N/A*	33.78	0.22
UKF w.o. bias est.	(2,1)	(0,0,0)	(0,0,0)	Correct	4.10	3.03	7.15	2.74	61.67
UKF w. bias est.	(2,2)	(0,0,0)	(0,0,0)	Correct	4.55	4.56	9.58	3.28	125.02
UKF w.o. bias est.	(2,3)	$(5,1,-3)\cdot 10^3$	(0.2,0.2,0.2)	Correct	15.69	14.31	105.82†	23.19	-
UKF w. bias est.	(2,4)	$(5,1,-3)\cdot 10^3$	(0.2,0.2,0.2)	Correct	14.37	4.23	11.33	4.97	-
UKF w. bias est.	(3,1)	$(5,1,-3)\cdot 10^3$	(0.2,0.2,0.2)	Incorrect	13.97	4.18	10.16	4.68	-
UKF w. bias est. (No ${}^{c}_{s}\mathbf{q}$)	(3,2)	$(5,1,-3)\cdot 10^3$	(0.2,0.2,0.2)	Correct	14.37	4.25	11.28	5.00	123.84

*The SVD-method does not work during eclipse.

†The UKF without bias estimation works very poorly during eclipse, if there is bias in the measurements.

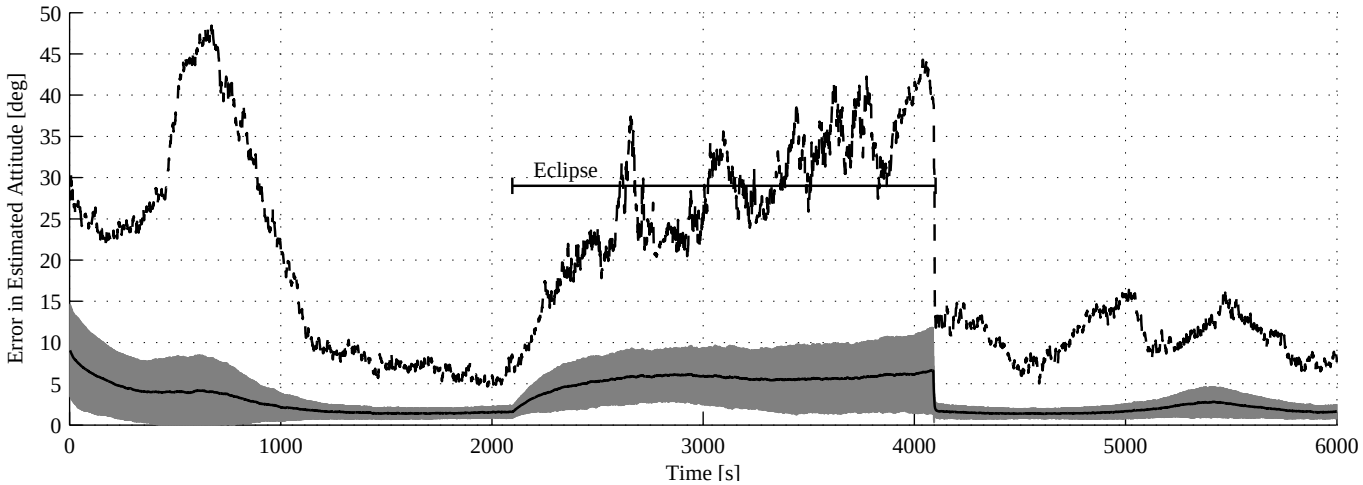

Fig. 4. Difference between the estimated attitude and the attitude from the "truth model" in the Monte Carlo simulation. The black line shows the sample mean error, the grey area indicates the sample standard deviation and the dashed black line shows the maximum errors over the 1000 simulations.

As Graph 3 and 4 in Fig. 2 indicate, bias in the sensor measurements greatly reduces the performance of the UKF unless the bias is estimated along with the attitude. However, it should be noted that expanding the state vector from 7 to 13 almost doubles the computation time of the filter (see Table 2).

As seen in Graph 1 in Fig. 3, small deviations between the estimated and actual inertia do not affect the performance of the UKF significantly. Larger deviations will affect the performance of the UKF; however, the CubeSat standard prescribes that the center of mass must not be further away than 2 cm from the geometric center for one unit CubeSats. It has been experienced that the maximum corresponding deviation in the inertia do not affect the performance of the UKF, and hence inertia estimation is unnecessary for one unit CubeSats.

The settings of the simulated UKF are chosen to rely mostly on the obtained measurements as the torques from disturbances have not been included in the satellite equations of motion in the onboard software. Furthermore, it is not expected that the control torque generated by e.g. permanent magnets and magnetorquers is precisely known, which introduces additional uncertainties. This means that the filter property of the UKF is toned down to improve robustness especially during eclipse at the expense of performance outside eclipse.

The main disadvantage of the UKF is the computation time. Taking advantage of the fact that one unit CubeSats are close to symmetric, it is possible to refrain from the rotation between the satellite body reference frame and the controller reference frame, as indicated in Graph 2 in Fig. 3. However, this do not lower the computation time significantly (see Table 2), as approximately 95% of the computation time is used to compute the prediction step 1.4 (see Table 1). Hence, reducing the number of Runge Kutta sub-steps may be the most effective way to reduce the computational time. Simulations have shown almost identical performance, with only one Runge Kutta sub-step. The computation time using this setup is approximately 1/9 of the 125.02 ms stated in Table 2. The reason for the almost equivalent performance is expected to be due to the low angular velocity after detumbling.

The final implementation for AAUSAT3 therefore uses only one Runge Kutta sub-step and sensor bias estimation. This implementation has been run through a Monte Carlo simulation and the mean attitude error and sample standard deviation is shown in Fig. 4. It takes approximately 1000 seconds before the mean error has converged and the UKF has found a good estimate of the sensor bias. The mean error increases, when the sun vector measurement disappears in eclipse and goes back to approximately 2 degrees when the satellite comes out of eclipse. A slightly larger error is encountered again when the vector measurements are close to parallel. Lastly, the worst case error stays bounded during all 1000 simulations and it is believed that even better attitude estimation could be obtained, if the UKF filter is tuned using the Monte Carlo simulation instead of the simple manual tuning performed in this paper. A similar Monte Carlo simulation, where the inertia and center of mass are the same both in the truth model and in the UKF, gives similar results as those shown in Fig. 4, which again indicates that it is not necessary to estimate the inertia and center of mass on CubeSats. Furthermore, mass balancing of the CubeSat to place the center of mass in the geometric center will give smaller disturbance torques and better attitude estimation performance.

The implementations proposed in this paper are targeted for CubeSats; however, the results are extendable to larger satellites. Doing so might require the incorporation of some of the disturbance models, such as calculation of the gravity gradient torque, in the onboard satellite equations of motion. This can also be done on CubeSats, but might not be appropriate with the limited computational power offered on CubeSats.

9. CONCLUSION

The focus of this paper has been to present a robust attitude estimator for low budget CubeSats. To summarize, it has been

proven important to estimate bias in the sensors, while inertia estimation is unnecessary for small symmetrically shaped satellites, since the diagonal elements in the inertia matrix are almost equal. Monte Carlo simulations of the implemented quaternion UKF with bias estimation have additionally shown, that the attitude estimate is improved with more non parallel vector measurements and that reasonable attitude estimates can be obtained even during eclipse, with the low cost off-the-shelf sensor setup chosen for AAUSAT3.

REFERENCES

T. Bak. *Spacecraft Attitude Determination - A Magnetometer Approach*. PhD thesis, Department of Control Engineering - Aalborg University, 1999. 2nd ed., pp. 9-69.

CalPoly SLO. Cubesat design specification rev.12. URL: `http://www.cubesat.org/images/developers/cds_rev12.pdf` [cited 7 Marts 2010].

Yee-Jin Cheon. Unscented Filtering in a Unit Quaternion Space for Spacecraft Attitude Estimation. *International Conference on Control, Automation and Systems*, June 2005.

A. Chin, R. Coelho, L. Brooks, R. Nugent, and J. Puig-Suari. Standardization Promotes Flexibility; A Review of CubeSats's Success. *AIAA/6th Responsive Space Conference*, (RS6-2008-4006):1–9, April-May 2008. Los Angeles, CA.

J. L. Crassidis and F. Landis Markley. Unscented Filtering for Spacecraft Attitude Estimation. *AIAA Journal of Guidance, Control, and Dynamics*, 26:536–542, August 2003.

Mohinder S. Grewal and Augus P. Andrews. *Kalman Filtering - Theory and Practice Using MATLAB*. John Wiley and Sons, Inc., 3rd edition, 2008. pp. 330-350.

Simon Haykin. *Kalman Filtering and Neural Networks*. John Wiley and Sons, Inc., 1st edition, 2001. pp. 221-267.

E. Kraft. A Quaternion-based Unscented Kalman Filter for Orientation Tracking. *Proceedings of the 6th International Conference on Information Fusion*, pages 47–54, July 2003. Cairns, Queensland.

J. B. Kuipers. *Quaternions and Rotation Sequences*. Princeton University Press, 1st edition, 2002. pp. 103-139.

J. J. LaViola. A Comparison of Unscented and Extended Kalman Filtering for Estimating Quaternion Motion. *Proceedings of the American Control Conference*, pages 2435–2440, June 2003. Denver, CO.

E. J. Lefferts, F. L. Markley, and M. D. Shuster. Kalman Filtering for Spacecraft Attitude Determination. *Journal of Guidance and Control*, 5(5):417–429, 1982.

F. L. Markley and D. Mortari. Quaternion Attitude Estimation Using Vector Observations. *The Journal of Astronautical Sciences*, 48(2 and 3):359–380, April-September 2000.

S.H. Pourtakdoust and H. Ghanbarpour. An Adaptive Unscented Kalman Filter for Quaternion-based Orientation estimation in Low-cost AHRS. *Aircraft Engineering and Aerospace Technology: An international Journal*, 79(5): 485–493, 2007.

J. Puig-Suari, C. Turner, and R. J. Twiggs. CubeSat: The Development and Launch Support Infrastructure for Eighteen Different Satellite Customers on One Launch. *In Proceedings of the 15th Annual AIAA/USU Conference on Small Satellites*, 2001. Logan, UT.

R. A. Serway. *Physics for Scientists and Engineers with Modern Physics*. Thomson Brooks/Cole, 6th edition, 2004. pp. 114-115.

M. D. Shuster. The Generelized Wahba Problem. *The Journal of Astronautical Sciences*, 54(2):245–259, April-June 2006.

Dan Simon. *Optimal State Estimation - Kalman, H∞ and Nonlinear Approaches*. John Wiley and Sons, Inc., 1st edition, 2006. pp. 433-461.

M. C. VanDyke, J. L. Schwartz, and C. D. Hall. Unscented Kalman Filtering for Spacecraft Attitude State and Parameter Estimation. *Proceedings of the 14th AAS/AIAA Space Flight Mechanics Conference*, 115(4), February 2004. Maui, Hawaii.

Grace Wahba. Problem 65-1: A Least Squares Estimate of Satellite Attitude. *Siam Review*, 7(3):409, 1965.

J. R. Wertz. *Spacecraft Attitude Determination and Control*. Kluwer Academic Publishers, 1st edition, 1994. pp. 510-524.

B. Wie. *Space Vehicle Dynamics and Control*. American Institute of Aeronautics and Astronautics, Inc., 1st edition, 1998. pp. 307-342.