



# Journal of Applied Sciences

ISSN 1812-5654

**science**  
alert

**ANSI***net*  
an open access publisher  
<http://ansinet.com>

## Design and Implementation for a Non Linear State Filter for LEO Micro Satellite

S. Chouraqui and M. Benyettou

Simulations and Modelling of Industrial Systems Laboratory, Department of Computer Science,  
Faculty of Science, University of Science and Technology of Oran, Mohammed Boudiaf, USTO, Algeria

**Abstract:** This study preliminarily investigates the numerical application of both Extended Kalman Filter (EKF) (which has traditionally been used for non linear estimation) and a relatively new filter, Unscented Kalman Filter (UKF) to the nonlinear estimation problem. The new method can be applied to nonlinear systems without the linearization process necessary for the EKF and it does not demand a Gaussian distribution of noise and what's more, its ease of implementation and more accurate estimation features enables it to demonstrate its good performance. Present experimental results and analysis indicate that unscented Kalman filtering UKF have shown better performances in presence of the severe nonlinearity in state equations.

**Key words:** Extended Kalman filtering, unscented Kalman filtering, spacecraft, quaternion, nonlinear estimation

### INTRODUCTION

This study considers the problem of applying the Kalman Filter (KF) to nonlinear systems. Estimation in nonlinear systems is extremely important because almost all practical systems involve nonlinearities. Accurately estimating the state of such systems is extremely important for fault detection and control applications. However, estimation in nonlinear systems is extremely difficult. The optimal (Bayesian) solution to the problem requires the propagation of the description of the full probability density function (pdf) (Kushner and Budhiraja, 2000). This solution is extremely general and incorporates aspects such as multimodality, asymmetries and discontinuities. However, because the form of the pdf is not restricted, it cannot, in general, be described using a finite number of parameters. Therefore, any practical estimator must use an approximation of some kind. Many different types of approximations have been developed; unfortunately, most are either computationally unmanageable or require special assumptions about the form of the process and observation models that cannot be satisfied in practice. For these and other reasons, the KF remains the most widely used estimation algorithm. The most common application of the KF to nonlinear systems is in the form of the Extended KF (EKF) (Martin, 2001). Exploiting the assumption that all transformations are quasi-linear, the EKF simply linearizes all nonlinear transformations and substitutes Jacobian matrices for the linear transformations in the KF

equations. Although, the EKF maintains the elegant and computationally efficient recursive update form of the KF, it suffers a number of serious limitations.

Linearized transformations are only reliable if the error propagation can be well approximated by a linear function. If this condition does not hold, the linearized approximation can be extremely poor. At best, this undermines the performance of the filter. At worst, it causes its estimates to diverge altogether. However, determining the validity of this assumption is extremely difficult because it depends on the transformation, the current state estimate and the magnitude of the covariance. This problem is well documented in many applications such as the estimation of ballistic parameters of missiles (Costa, 1994; Chowdhary and Jategaonkar, 2006) and computer vision (Viéville *et al.*, 1993).

Linearization can be applied only if the Jacobian matrix exists. However, this is not always the case. Some systems contain discontinuities in which the parameters can change abruptly, others have singularities and in others the states themselves are inherently discrete.

Calculating Jacobian matrices can be a very difficult and error-prone process. The Jacobian equations frequently produce many pages of dense algebra that must be converted to code. This introduces numerous opportunities for human coding errors that may undermine the performance of the final system in a manner that cannot be easily identified and debugged-especially given the fact that it is difficult to know what quality of performance to expect. Regardless of whether the obscure



code associated with a linearized transformation is or is not correct, it presents a serious problem for subsequent users who must validate it for use in any high integrity system.

To address these limitations, Julier and Uhlmann (1997) developed the UKF. Therefore, it was extended to parameter estimation applications by Van der Merve (2004).

Instead of linearizing these non linear functions using Jacobian matrices, the UKF uses a deterministic sampling approach to calculate the mean and covariance estimates of Gaussian random state variables with a minimal set of sample points called as sigma point (Kandepu *et al.*, 2008) through the actual non linear functions. This approach yields more accurate results compared to ordinary functions linearization in the EKF.

### FILTERS ALGORITHMS

**The extended Kalman filter:** The Kalman Filter (KF) in its standard form is a popular choice for any stochastic estimation problem. However, in most real applications of interest; the system dynamics and observation equations are non linear and hence suitable modifications to the standard kalman filter are required. The Extended Kalman Filter (EKF) provides these modifications by linearizing the non linear process and observation models around the last state estimate. In this way, the EKF gives an approximation of the optimal estimate and hence can be considered as the most common and popular approach for both non linear state estimation and parameter estimation.

Let the process be estimated and the associated observation relationship be described by the following non linear state space model:

$$\begin{aligned} x_{k+1} &= f(x_k, u_k) + w_k \\ z_k &= h(x_k) + v_k \end{aligned} \tag{1}$$

where,  $x_k$  represents the unobserved state of system  $u_k$  is known exogenous input and  $z_k$  is the observed measurement output. The process noise  $w_k$  drives the dynamic system and the observation noise is given by  $v_k$ .

The EKF involves the following recursive estimation of the mean ( $\hat{x}$ ) and error covariance ( $P$ ) of the state estimate under a Gaussian assumption:

- Consider the last estimated state  $\hat{x}(k|k)$
- Linearize the non linear system dynamics  $x_{k+1} = f(x_k, u_k) + w_k$  around  $\hat{x}(k|k)$
- Apply the prediction cycle of the kalman filter algorithm to the linearized system dynamics in order to calculate a prior state estimate  $\hat{x}(k+1|k)$  and a prior estimate of the error covariance matrix  $P(k+1|k)$

- Linearize the non linear measurement dynamics  $y_k = h(x_k) + v_k$  around  $\hat{x}(k+1|k)$
- Apply filtering or update cycle of the kalman filter algorithm to the linearized measurement dynamics in order to calculate the posteriori state estimate  $\hat{x}(k+1|k)$  and the posteriori estimate of error covariance matrix  $P(k+1|k)$

Let  $\Phi(k)$  and  $H(k)$  be the Jacobian matrices of the non linear process  $f(\cdot)$  and observation  $h(\cdot)$  models around the estimated state, denoted by:

$$\begin{aligned} \Phi(k) &= \left. \frac{\partial f(x, u)}{\partial x} \right|_{\hat{x}(k|k), u_k} \\ H(k) &= \left. \frac{\partial h(x)}{\partial x} \right|_{\hat{x}(k+1|k)} \end{aligned} \tag{2}$$

The EKF works almost like a standard KF, except for  $F$  and  $H$ , which vary in time based on the estimated state  $\hat{x}$ . Its actual algorithm can be stated in terms of the following two distinct cycles:

**Predict cycle:** Predict next state, before measurements are taken:

$$\begin{aligned} \hat{x}(k+1|k) &= f(\hat{x}(k|k), u_k) \\ P(k+1|k) &= \Phi(k)P(k|k)\Phi^T(k) + Q(k) \end{aligned} \tag{3}$$

**Update cycle:** Update state, after measurements are taken:

$$\begin{aligned} \hat{x}(k+1|k+1) &= \hat{x}(k+1|k) + K(k+1)[y_{k+1} - h(\hat{x}(k+1|k))] \\ K(k+1) &= P(k+1|k)H^T(k+1)[H(k+1)P(k+1|k)H(k+1)^T + R(k+1)]^{-1} \\ P(k+1|k+1) &= [I - K(k+1)H(k+1)]P(k+1|k) \end{aligned} \tag{4}$$

where,  $Q$  and  $R$  are covariance matrices, describing the second order properties of the state and measurement noise (i.e.,  $Q = E(w w^T)$  and  $R = E(v v^T)$ , where  $E(\cdot)$  denotes the expected operator).  $K$  denotes the kalman gain matrix, used in the update observer.

**The unscented Kalman filter:** The UKF is based on the idea that it is easier to approximate a probability distribution than to approximate an arbitrary nonlinear transformation (Julier and Uhlmann, 1994). The actual algorithm is based on propagating carefully selected finite set of points, called sigma points, through the system nonlinear dynamics and then approximating the first two moments of the distribution (mean and covariance) through a suitable method; such as weighted sample mean and covariance calculations (Julier and Uhlmann, 1994; Julier *et al.*, 2000). The flaw in the EKF that results



from propagating the mean and covariance through linear approximations of the nonlinear transformation is thus eliminated in the UKF, leading to theoretically better performance of the UKF. Furthermore, the UKF implementation does not need the calculation of any Jacobian or Hessian matrices, which not only results in considerable simplicity in implementation, but also makes the UKF suitable for real-time applications and applications involving non-differentiable functions. The accuracy of the UKF can be compared to that of the second order EKF, the computational order is comparable to the EKF (Julier and Uhlmann, 1994; Julier *et al.*, 2000).

The UKF follows the same overall structure as the EKF. The UKF is fundamentally different in the way it estimates the noise distributions of the filter. As opposed to approximating the non-linear transformation of the noise distributions, the UKF approximates the transformation by applying the non-linear transformation to a number of selected points in the state space. These points are called sigma points and are calculated from the covariance of the estimation error. The approach is shown in Fig. 1. The sigma points, circled points in the Fig. 1, are selected such that they describe the distribution of the estimation error. The transformed sigma points are then used to calculate the approximation of the transformed distribution. This is known as the UT published in (Julier and Uhlmann, 1997). In the following, the equations of the UKF are presented.

As with the EKF, we present an algorithmic description of the UKF. Given the state vector at step  $k$  (we use the same state vector as in Eq. 1), we compute a collection of sigma points, stored in the columns of the  $L \times (2L+1)$  sigma point matrix  $\chi_k$  where  $L$  is the dimension of the state vector. In our case,  $L = 7$  so  $\chi_k$  is a  $7 \times 15$  matrix. The columns of  $\chi_k$  are computed by:

$$\begin{aligned} (\chi_k)_0 &= x_k \\ (\chi_k)_i &= x_k + (\sqrt{(L+\lambda)P_k})_i, \quad i=1...L \\ (\chi_k)_i &= x_k - (\sqrt{(L+\lambda)P_k})_i, \quad i=L+1...2L \end{aligned} \tag{5}$$

where,  $\sqrt{(L+\lambda)P_k}$  is the  $i$ th column of the matrix square root and  $\lambda$  is defined by:

$$\lambda = \alpha^2(L + \kappa - L) \tag{6}$$

where,  $\alpha$  is a scaling parameter which determines the spread of the sigma points and  $\kappa$  is a secondary scaling parameter.

Note that we assume  $\sqrt{(L+\lambda)P_k}$  is symmetric and positive definite which allows us to find the square root using Cholesky decomposition.

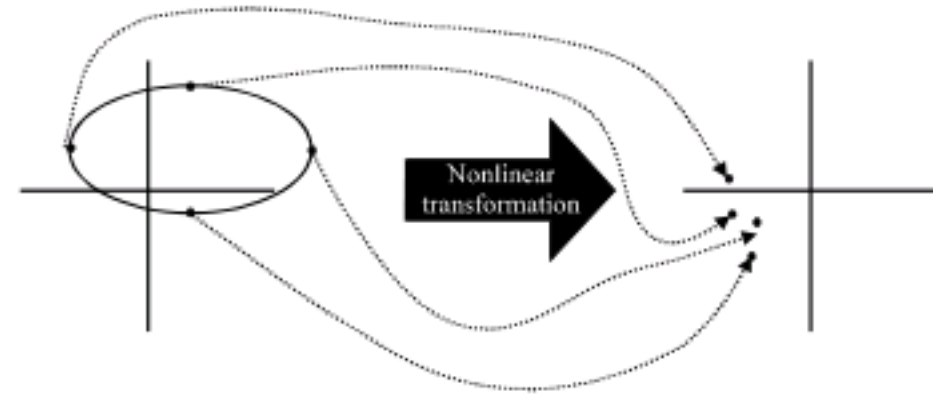


Fig. 1: The unscented transformation

**Prediction cycle:** Once  $\chi_k$  computed, we perform the prediction step by first propagating each column of  $\chi_k$  through time by  $\Delta t$  using nonlinear function as:

$$(\chi_{k+1})_i = f((\chi_k)_i), \quad i=0...2L \tag{7}$$

In present formulation, since  $L = 7$ , we perform 15 4th order Runge-Kutta integrations.

Calculate the a priori state estimate:

$$x_{k+1} = \sum_{i=0}^{2L} W_i^{(m)} (\chi_{k+1})_i \tag{8}$$

where,  $W_i^{(m)}$  are weights defined by:

$$\begin{aligned} W_0^{(m)} &= \lambda / (L + \lambda) \\ W_i^{(m)} &= 1 / 2(L + \lambda), \quad i=1...2L \end{aligned} \tag{9}$$

Calculate the a priori error covariance:

$$P_{k+1} = \sum_{i=0}^{2L} W_i^{(c)} [(\chi_{k+1})_i - x_{k+1}] [(\chi_{k+1})_i - x_{k+1}]^T + Q_k \tag{10}$$

where,  $Q_k$  is once again the process error covariance matrix and the weights are defined by:

$$\begin{aligned} W_0^{(c)} &= \lambda / (L + \lambda) + (1 + \alpha^2 + \beta) \\ W_i^{(c)} &= 1 / 2(L + \lambda), \quad i=1...2L \end{aligned} \tag{11}$$

**Correction cycle:** Transform the columns of  $\chi_k$  through the measurement function.

$$(Z_{k+1})_i = h((\chi_{k+1})_i), \quad i=0...2L \tag{12}$$

Calculate the predicted observation:

$$\hat{z}_{k+1} = \sum_{i=0}^{2L} W_i^{(m)} (Z_{k+1})_i \tag{13}$$

Calculate the a posteriori state estimate:

$$x_{k+1} = x_{k+1}^- + K_{k+1}(z_{k+1} - z_{k+1}^-) \quad (14)$$

where,  $K_k$  is once again Kalman gain. In the UKF formulation,  $K_k$  is defined by:

$$K_{k+1|k+1} = P_{k+1|k} P_{k+1|k}^{-1} \quad (15)$$

Where:

$$P_{k+1|k} = \sum_{i=0}^{2L} W_i^{(c)} [(Z_k)_i - z_k^-] [(Z_k)_i - \hat{z}_k^-]^T + R \quad (16)$$

$$P_{k+1|k} = \sum_{i=0}^{2L} W_i^{(c)} [(Z_k)_i - \hat{z}_k^-] [(Z_k)_i - \hat{z}_k^-]^T$$

Note that as with the EKF,  $R$  is the measurement noise covariance matrix.

Calculate the a posteriori estimate of the mean state and the error covariance:

$$\hat{x}_k = x_k^- + K_k(z_k - \hat{z}_k^-) \quad (17)$$

$$P_k = P_k^- - K_k P_{k|k} K_k^T$$

In present study, we have used the 7 state vectors, for both EKF and UKF, which defines the attitude of the satellite and also the rates at which the attitude is changing (Matthew *et al.*, 2004). The state vector is comprised of the four-element quaternion attitude vector combined with the three-element body rates vector, with respect to the inertial frame ( $\omega^I_{By}$ ). Symbolically, this state vector can be represented as:

$$x = [q_1, q_2, q_3, q_4, \omega_x, \omega_y, \omega_z]^T \quad (18)$$

During the propagation cycle, the quaternion and angular rate components of the state vector are propagated separately. The quaternion are propagated forward in time utilizing the basic quaternion dynamic equation, specifically

$$\dot{q} = \frac{1}{2} \Omega q \quad (19)$$

with

$$\Omega = \begin{bmatrix} 0 & \omega_{oz} & -\omega_{oy} & \omega_{ox} \\ -\omega_{oz} & 0 & \omega_{ox} & \omega_{oy} \\ \omega_{oy} & -\omega_{ox} & 0 & \omega_{oz} \\ -\omega_{ox} & -\omega_{oy} & -\omega_{oz} & 0 \end{bmatrix} \quad (20)$$

and

$$\omega_B^O = [\omega_{ox} \quad \omega_{oy} \quad \omega_{oz}]^T$$

is the local orbital referenced body angular rate written in the body frame.

This vector can be found using the inertial referenced body rates and the orbital-referenced orbital rate vector by the following equations:

$$\omega_B^O = \omega_B^I - T \omega_0 \quad (21)$$

where,  $T$  is the quaternion transformation matrix from the orbital to body coordinates:

$$T = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1 q_2 + q_3 q_4) & 2(q_1 q_3 - q_2 q_4) \\ 2(q_1 q_2 - q_3 q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2 q_3 + q_1 q_4) \\ 2(q_1 q_3 + q_2 q_4) & 2(q_2 q_3 - q_1 q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \quad (22)$$

and  $\omega_0 = [0 \quad -\omega_0 \quad 0]$  is the inertial referenced orbital angular rate written in the orbital frame.

If we can assume that the satellite has a circular orbit, then we can find  $\omega_0$ , with the equation for mean motion:

$$\omega_0 = \sqrt{\frac{\mu_T}{a^3}} \quad (23)$$

where,  $\mu_T$  is the earth's gravitational parameter and  $a$  is the satellite's semi-major axis.

The body rates are propagated forward in time utilizing the dynamic equation of motion for orbiting satellites:

$$I \dot{\omega} = N - \omega \times I \omega \quad (24)$$

Where:

$I = \text{diag}(I_x, I_y, I_z)$ : the 3x3 moment of inertia tensor

$N_M =$  The magnetic torque vector

$N_D =$  The environmental disturbance torque vector

$N_T =$  The thrusters control torque vector

With these definitions in mind, the basic mathematical process for state Kalman Filtering can be detailed below.

The state transformation matrix  $\Phi$  is defined by:

$$\Phi_{k+1} = I_{7 \times 7} + \begin{bmatrix} \frac{\partial \dot{q}}{\partial q} & \frac{\partial \dot{q}}{\partial \omega_b^I} \\ \frac{\partial \dot{\omega}_b^I}{\partial q} & \frac{\partial \dot{\omega}_b^I}{\partial \omega_b^I} \end{bmatrix} \cdot (t_{k+1} - t_k) \quad (25)$$

$$= I_{7 \times 7} + \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix} \cdot \Delta t \quad (26)$$



The  $F_{11}$  is a (4×4) matrix and is calculated using Eq. 19, partial derivative of  $\dot{q}$  with respect to each  $q_i$  becomes:

$$\frac{\partial \dot{q}}{\partial q_1} = \frac{1}{2} \Omega \frac{\partial q}{\partial q_1} + \frac{1}{2} \frac{\partial \Omega}{\partial q_1} q \quad (27a)$$

$$\frac{\partial \dot{q}}{\partial q_1} = \frac{1}{2} \Omega \frac{\partial q}{\partial q_1} + \frac{1}{2} \Lambda \frac{\partial \omega_b^o}{\partial q_1} \quad (27b)$$

$$\frac{\partial \dot{q}}{\partial q_1} = \frac{1}{2} \Omega \frac{\partial q}{\partial q_1} + \frac{1}{2} \Lambda \frac{\partial}{\partial q_1} (\omega_b^l - T \omega_o) \quad (27c)$$

$$\frac{\partial \dot{q}}{\partial q_1} = \frac{1}{2} \Omega \frac{\partial q}{\partial q_1} - \frac{1}{2} \Lambda \frac{\partial T}{\partial q_1} \omega_o \quad (27d)$$

Thus the matrix is computed by evaluating:

$$F_{11} = \frac{1}{2} \Omega + [\tau_1 \quad \tau_2 \quad \tau_3 \quad \tau_4] \quad (28)$$

Where:

$$\tau_i = -\frac{1}{2} \Lambda \frac{\partial T}{\partial q_i} \omega_o \quad (29)$$

Finally

$$\tau_1 = \omega_b \begin{bmatrix} q_3 q_1 \\ q_3 q_2 \\ q_3^2 - 1 \\ q_3 q_4 \end{bmatrix} \quad (30a)$$

$$\tau_2 = \omega_b \begin{bmatrix} q_1 q_4 \\ q_2 q_4 \\ q_3 q_4 \\ q_4^2 - 1 \end{bmatrix} \quad (30b)$$

$$\tau_3 = \omega_b \begin{bmatrix} 1 - q_1^2 \\ -q_1 q_2 \\ -q_1 q_3 \\ -q_1 q_4 \end{bmatrix} \quad (30c)$$

$$\tau_4 = \omega_b \begin{bmatrix} -q_1 q_2 \\ 1 - q_2^2 \\ -q_2 q_3 \\ -q_2 q_4 \end{bmatrix} \quad (30d)$$

The (4×3)  $F_{12}$  matrix is computed using partial derivative of  $\dot{q}$  with respect to  $\omega_b^l$  and can be derived as follows. Again, it requires a substitution of Eq. 21.

$$\frac{\partial \dot{q}}{\partial \omega_b^l} = \frac{\partial}{\partial \omega_b^l} \left( \frac{1}{2} \Omega q \right) = \frac{\partial}{\partial \omega_b^l} \left( \frac{1}{2} \Lambda \omega_b^o \right) \quad (31a)$$

$$\frac{\partial \dot{q}}{\partial \omega_b^l} = \frac{1}{2} \Lambda \frac{\partial \omega_b^o}{\partial \omega_b^l} = \frac{1}{2} \Lambda \frac{\partial}{\partial \omega_b^l} (\omega_b^l - C^{1/2} \omega_o) \quad (31b)$$

$$\frac{\partial \dot{q}}{\partial \omega_b^l} = \frac{1}{2} \Lambda \frac{\partial \omega_b^l}{\partial \omega_b^l} = \frac{1}{2} \Lambda \quad (31c)$$

The partial derivative (3×4) matrix  $F_{21}$  can be found as follows:

$$F_{21} = [\xi_1 \quad \xi_2 \quad \xi_3 \quad \xi_4] \quad (32)$$

$$\xi_1 = \begin{bmatrix} 6\omega_b^2 \alpha (-T_{23} q_1 + T_{33} q_4) \\ 6\omega_b^2 \beta (+T_{33} q_3 - T_{13} q_1) \\ 6\omega_b^2 \gamma (+T_{13} q_4 + T_{23} q_3) \end{bmatrix} \quad (33a)$$

$$\xi_2 = \begin{bmatrix} 6\omega_b^2 \alpha (-T_{23} q_2 + T_{33} q_3) \\ 6\omega_b^2 \beta (-T_{33} q_4 - T_{13} q_2) \\ 6\omega_b^2 \gamma (+T_{13} q_3 - T_{23} q_4) \end{bmatrix} \quad (33b)$$

$$\xi_3 = \begin{bmatrix} 6\omega_b^2 \alpha (+T_{23} q_3 + T_{33} q_2) \\ 6\omega_b^2 \beta (+T_{33} q_1 + T_{13} q_3) \\ 6\omega_b^2 \gamma (+T_{13} q_2 + T_{23} q_1) \end{bmatrix} \quad (33c)$$

$$\xi_4 = \begin{bmatrix} 6\omega_b^2 \alpha (+T_{23} q_4 + T_{33} q_1) \\ 6\omega_b^2 \beta (-T_{33} q_2 + T_{13} q_4) \\ 6\omega_b^2 \gamma (+T_{13} q_1 - T_{23} q_2) \end{bmatrix} \quad (33d)$$

Where:

$$\alpha = \frac{I_z - I_y}{I_x}, \beta = \frac{I_x - I_z}{I_y}, \gamma = \frac{I_y - I_x}{I_z} \quad (34)$$

The partial derivative (3×3)  $F_{22}$  matrix can be found as follows:

$$F_{22} = \begin{bmatrix} 0 & k\omega_z & k\omega_y \\ -k\omega_z & 0 & -k\omega_x \\ 0 & 0 & 0 \end{bmatrix} \quad (35)$$

$$k = 1 - \frac{I_z}{I_t} \quad (36)$$

The process noise covariance matrix  $Q$  is used to represent the uncertainty in the linearized model of the system and how that uncertainty is correlated between the states. Based on the design of a Kalman filter and probability definitions, the process noise covariance matrix  $Q$  for step  $k$  is defined as follows (Anderson *et al.*, 2004):

$$Q = Q_1 \Delta t + Q_2 \frac{(\Delta t)^2}{2} + Q_3 \frac{(\Delta t)^3}{3} \quad (37)$$

Where:

$$Q_1 = \begin{bmatrix} 0_{4 \times 4} & 0_{4 \times 3} \\ 0_{4 \times 3} & S \end{bmatrix} \quad (38a)$$

$$Q_2 = \begin{bmatrix} 0_{4 \times 4} & F_{12}S \\ SF_{12}^T & SF_{22}^T + F_{22}S \end{bmatrix} \quad (38b)$$

$$Q_3 = \begin{bmatrix} F_{12}SF_{12}^T & F_{12}SF_{22}^T \\ F_{22}SF_{12}^T & F_{22}SF_{22}^T \end{bmatrix} \quad (38c)$$

$$S = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{bmatrix} \quad (38d)$$

The variances  $\sigma_x^2 \dots \sigma_z^2$  are an estimated variance of the body rates. This is not a straightforward quantity and best determined by computer simulation.

The extended Kalman filter approximates the observation matrix H by:

$$H_k = \frac{\partial z}{\partial x} = \begin{bmatrix} \frac{\partial T}{\partial q_1} z_o & \frac{\partial T}{\partial q_2} z_o & \frac{\partial T}{\partial q_3} z_o & \frac{\partial T}{\partial q_4} z_o \dots \\ \frac{\partial T}{\partial \omega_x} z_o & \frac{\partial T}{\partial \omega_y} z_o & \frac{\partial T}{\partial \omega_z} z_o \end{bmatrix} \quad (39)$$

where, z is measurements of system state, either sun-sensor or magnetometer

This basic process has introduced several variables that merit further explanation. The covariance matrix (P) essentially is a time-referenced estimate of the accuracy of both the system model and the measurements. The correction cycle depends heavily upon these accuracies in order to determine how much to trust either the propagated state or the entered measurements. The state transformation matrix ( $\Phi$ ) is an approximation of the change that the state undergoes over the specified time interval. The process noise covariance matrix (Q) is derived from the expected error in the filtering process. The observation matrix (H) is a measure of how dependent the measurements are upon the state of the system. The measurement noise covariance matrix (R) entails the expected error in the states themselves, derived from the precision of the system model.

### ATTITUDE DYNAMICS

The dynamic equation of motion for an orbiting satellite is (Hashida, 2004):

$$I\dot{\omega} = N_{GG} + N_D + N_M - \omega \times (I\omega + h) - \dot{h} \quad (40)$$

Where:

$\omega_b^i = [\omega_x \ \omega_y \ \omega_z]^t$  = Inertially referenced body angular rate vector

$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}$  = Moment of inertia tensor of spacecraft (MOI)

$h = [h_x \ h_y \ h_z]^t$  = Reaction wheel angular momentum vector

$N_{GG} = [N_{GGx} \ N_{GGy} \ N_{GGz}]^t$  = Gravity-gradient torque vector

$N_D = [N_{dx} \ N_{dy} \ N_{dz}]^t$  = External disturbance torque vector such as aerodynamic torque and solar radiation pressure torque

$N_M = [N_{mx} \ N_{my} \ N_{mz}]^t$  = Applied torque vector by 3-axis magnetorquers

For an axially symmetric satellite with Y/Z wheels and the principal moment of inertia axes along the body axes, the off-diagonal products of inertia elements in the MOI tensor will be zero. The deployed boom along the Z-axis also increases the MOI elements Ixx and Iyy to a much larger and equal value. This value is called the transverse MOI,  $I_T$ .

The complete set of dynamic equations of motion can then be written as follows:

$$\begin{aligned} I_T \dot{\omega}_x &= N_{mx} + N_{dx} - 3\omega_0(I_T - I_{zz})T_{23}T_{33} - \omega_y(I_{zz}\omega_x + h_x) + \omega_z(I_T\omega_y + h_y) \\ I_T \dot{\omega}_y &= N_{my} + N_{dy} + 3\omega_0(I_T - I_{zz})T_{13}T_{33} + \omega_x(I_{zz}\omega_y + h_y) - \omega_z(I_T\omega_x) - \dot{h}_y \\ I_z \dot{\omega}_z &= N_{mz} + N_{dz} - \omega_x h_y - \dot{h}_z \end{aligned} \quad (41)$$

The spacecraft attitude is parameterized by quaternion denoted  $q = (q1 \ q2 \ q3 \ q4)$  which has a corresponding rotation matrix T in (22). The attitude kinematics, which couple  $\omega$  and q are given by Eq. 19 and 20.

### SIMULATION RESULTS

In order to examine its operability and performance, EKF and UKF were programmed in the computer language MATLAB/SIMULINK. These simulation results were all completed during sun-lit portions of orbit, when both the sun sensors and magnetometer were providing data to the filter.



In order to test the EKF before launch, a simulation of the operating environment of the satellite must first be developed. This simulation is required in order to generate realistic measurements for the EKF to utilize as an input. The simulation used to generate the input measurements for all analysis detailed herein is an Alsat-1 flight orbit code. It has been used on various satellites engineered by Surrey Satellite Technology Ltd. This orbit propagator generates simulated magnetometer and sun sensor inputs of two types. First, the orbit propagator generates these magnetometer and sun sensor inputs in the local orbital frame. These filter inputs are what is considered the orbit propagator predicted values. This input from the simulation will be necessary during actual satellite operations. The second set of input data generated by the orbit propagator is a set of measured values. These values simulate measurement input from the magnetometer and sun sensor and so this set of input is in the body frame. From this second set of inputs, the measurement inputs, are only simulating values that will come from onboard sensors during the actual operation of the satellite, this data will not be generated by the simulator during actual satellite operations.

In addition, a standard of comparison is necessary in order to determine the veracity of the EKF results. Another well-tested program developed by Surrey Satellite Technology Ltd. was utilized for this purpose. This simulation is an attitude propagator, which models the actual satellite dynamics and outputs Euler angles and Euler rates over a specified period of time. This simulation simply uses Euler's Moment Equations and the quaternion dynamic equation to propagate expected angles and rates. Due to the construction of this model, it is quite simple to simplify the model to discount all disturbances torques etc. or to precisely model gravity gradient (including boom deployment), magnetic (including commanded magnetic moment), drag, reaction wheel effects and other disturbance torques.

We consider our satellite is operating in a  $98^\circ$  inclined circular LEO at an altitude of 686 km. The spacecraft is estimated to have a mass of 90 kg and essentially be configured as a rectangular shape  $0.65 \times 0.65 \times 0.6$  m<sup>3</sup> with a 6 m (to the centre of gravity) gravity gradient boom and 3 kg tip mass. The microsatellite is Nadir pointing. These requirements specify several operating constants for the EKF; a summary of operating constants is presented.

A 2006 IGRF B-field model was used to obtain the geomagnetic field values.

Note that the assumed integration time step ( $\Delta t_{int}$ ) is 1 S unless otherwise specified. The inertia tensor is boom deployed.

Several other practical consideration merit mention. The integration in the state propagation is accomplished by numerical integration. This cyclic process is computationally intensive. Therefore, to reduce the computational demand without sacrificing significant accuracy, an Adam's 2nd order numerical integrator will be implemented rather than a more complex and accurate integrator.

Another practical consideration concerns the nature of the computer code. Since, the code will be implemented on a satellite, the code must be extremely robust. Most importantly, the ADCS computer code must not crash the satellite's onboard computer at all expenses. Therefore, error checks are conducted many times throughout each cycle of the EKF. If an error is detected, the ADCS algorithm is immediately exited and a message relayed to satellite operators to allow human operators to deal with the problem rather than the onboard computer crashing as a result of the error. Finally, the nature of the assumptions underlying the development of the EKF permits a small amount of error to creep into the quaternion calculations. In order to negate this error, the quaternion must be normalized after every instance where they are calculated. This includes both the quaternion calculation in the state propagation step and in the state update step.

Finally, several other initialization parameters deserve an explanation. The covariance matrix (P) embodies an approximate error associated with attitude estimates. The first four diagonal elements of the covariance matrix represent the estimated error of the quaternion, while the next three diagonal elements give the estimated error of the angular rates. It is important to note that the covariance matrix is merely an initialization; over time, the matrix changes as the EKF converges. The process noise covariance matrix (Q) is another initialization parameter that deserves mention. The process noise covariance matrix contains information relating an estimate for the error associated with the system equations. From a strict mathematical standpoint, the process noise covariance matrix also changes with time. However, these changes can be ignored due to their small magnitude. The first four diagonal elements of the process noise covariance matrix represent the error associated with the quaternion dynamic equation, while the next three diagonal elements are the estimated error associated with Euler's moment equations. The process noise covariance matrix error estimates is  $1e^{-5}$  for both quaternion and rates. Lastly, the measurement noise covariance matrix (R) contains the error expected to be associated with measurements. Note that the measurement noise covariance will change as the sun sensor is switched on and off. However, the measurement noise covariance is otherwise constant over time.



Table 1: Orbit characteristic

Orbits	Circular
Inclination (°)	98
Altitude (km)	680
Sampling period (sec)	10

Table 2: Initial uncertainty of quaternions and angular rate (Initial covariance matrix P)

Characteristics	Values
Initial quaternion $q_1$	$(0.1)^2$
Initial quaternion $q_2$	$(0.1)^2$
Initial quaternion $q_3$	$(0.1)^2$
Initial quaternion $q_4$	$(0.1)^2$
Initial angular rate $w_x$ (rad) <sup>2</sup>	$\left( (0.1)^2 \cdot \frac{\pi}{180^\circ} \right)$
Initial angular rate $w_y$ (rad) <sup>2</sup>	$\left( (0.1)^2 \cdot \frac{\pi}{180^\circ} \right)$
Initial angular rate $w_z$ (rad) <sup>2</sup>	$\left( (0.1)^2 \cdot \frac{\pi}{180^\circ} \right)$

Table 3: Process noise intensity of quaternions and angular rate (System noise covariance matrix Q)

Characteristics	Values
Intensity for $q_1$ term $\left( \frac{s^3}{kg \cdot m^2} \right)^2$	$\frac{1e^{-4} \cdot \Delta T^3}{12 \cdot I_x^2}$
Intensity for $q_2$ term $\left( \frac{s^3}{kg \cdot m^2} \right)^2$	$\frac{1e^{-4} \cdot \Delta T^3}{12 \cdot I_y^2}$
Intensity for $q_3$ term $\left( \frac{s^3}{kg \cdot m^2} \right)^2$	$\frac{1e^{-4} \cdot \Delta T^3}{12 \cdot I_z^2}$
Intensity for $q_4$ term $\left( \frac{s^3}{kg \cdot m^2} \right)^2$	$\left( q_1^2 \cdot \frac{1e^{-4}}{I_x^2} + q_2^2 \cdot \frac{1e^{-4}}{I_y^2} + q_3^2 \cdot \frac{1e^{-4}}{I_z^2} \right) \cdot \Delta T^3$ 12
Intensity for $w_x$ (rad) <sup>2</sup> $\left( \frac{s}{(kg \cdot m^2)^2} \right)$	$\frac{1e^{-4} \cdot \Delta T}{I_x^2}$
Initial angular rate $w_y$ $\left( \frac{s}{(kg \cdot m^2)^2} \right)$	$\frac{1e^{-4} \cdot \Delta T}{I_y^2}$
Initial angular rate $w_z$ $\left( \frac{s}{(kg \cdot m^2)^2} \right)$	$\frac{1e^{-4} \cdot \Delta T}{I_z^2}$

Table 4: Inertia tensor (Satellite configuration I)

Inertia tensor	Values
$I = \text{diag} (I_x, I_y, I_z)^T$ (kg.m <sup>2</sup> )	(158, 158, 5) <sup>T</sup>

Table 5: Measurement error variance (Measurement noise covariance matrix R)

Error variance	Values
Magnetometer measurement error variance X/Y/Z axis (μT) <sup>2</sup>	(0.3) <sup>2</sup>
Sun sensor measurement error variance X/Y/Z axis (°) <sup>2</sup>	(0.1) <sup>2</sup>

The measurement and system noise covariance matrices were chosen as diagonal and adjusted through simulation to give a kalman filter with the best filtering properties smallest tracking error. In all simulations the following values were used throughout. These measurements are shown in Table 1-5.

The simulation results are represented in Fig. 2-9, the black graph shows the true values of quaternion, the

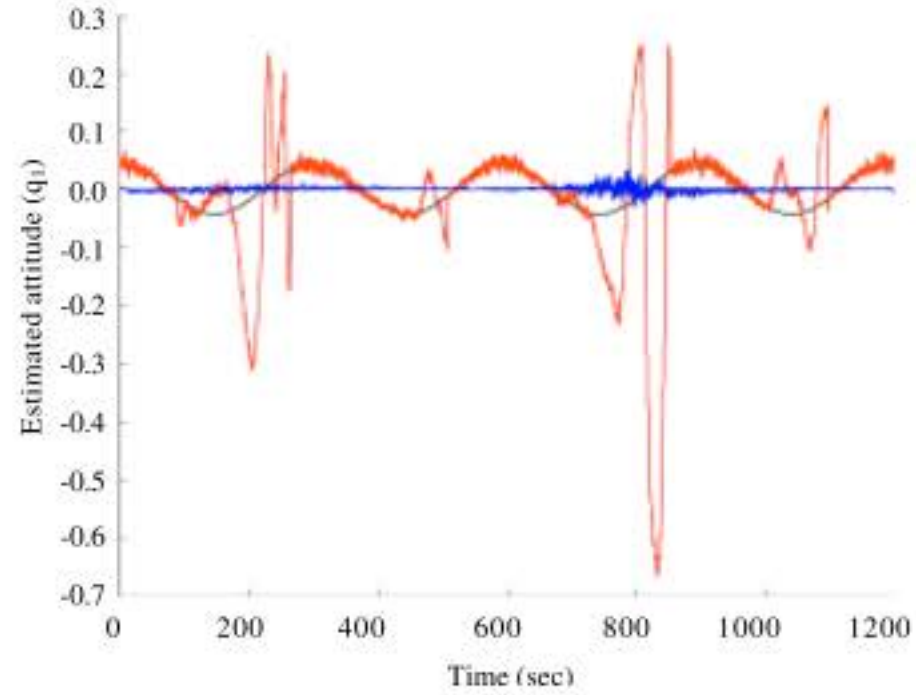


Fig. 2: Comparison between UKF and EKF for estimated state  $q_1$

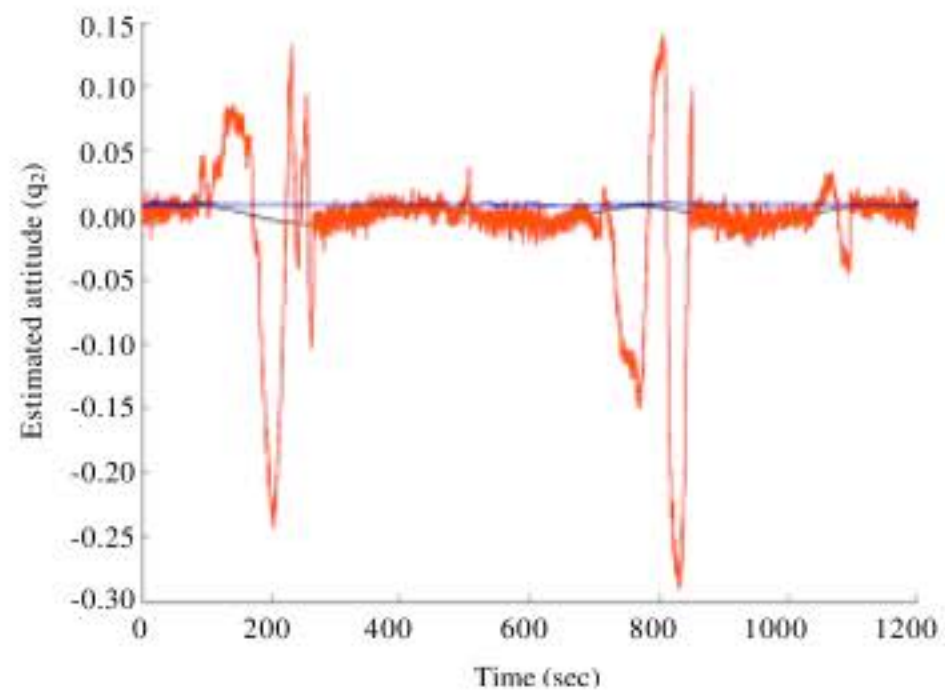


Fig. 3: Comparison between UKF and EKF for estimated state  $q_2$

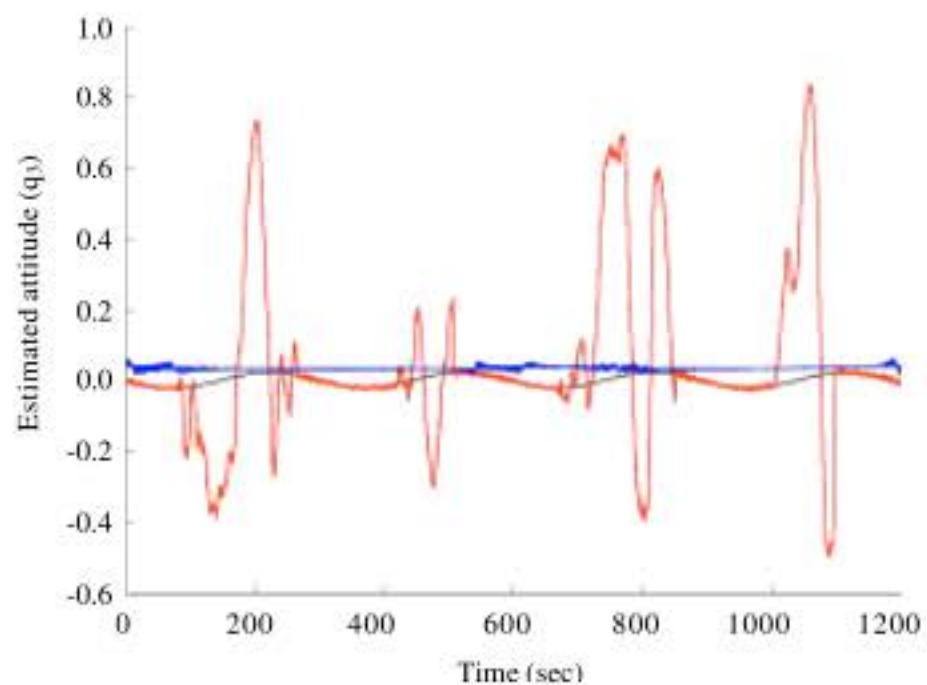


Fig. 4: Comparison between UKF and EKF for estimated state  $q_3$



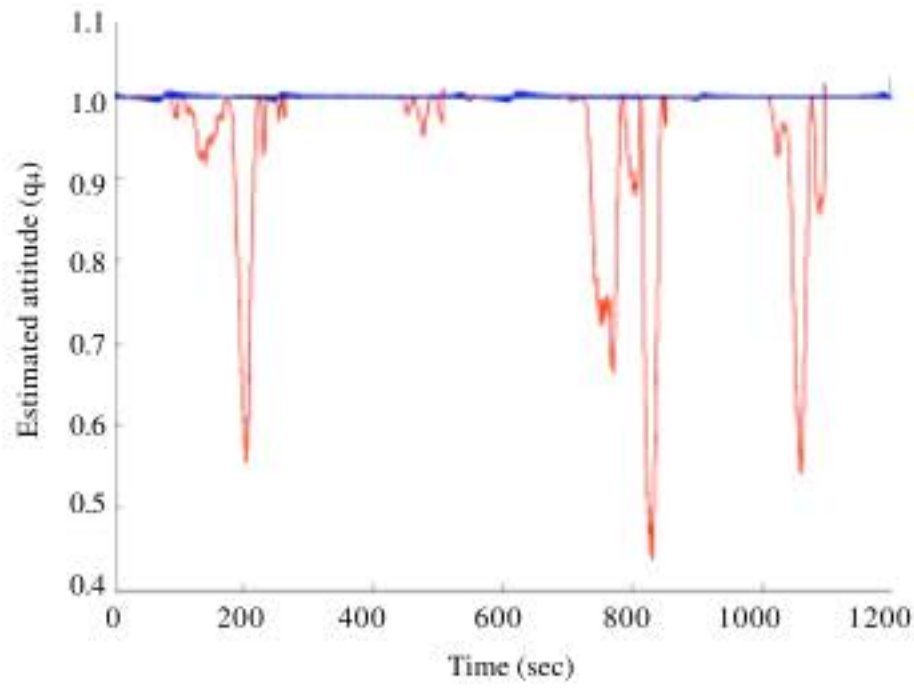


Fig. 5: Comparison between UKF and EKF for estimated state  $q_4$

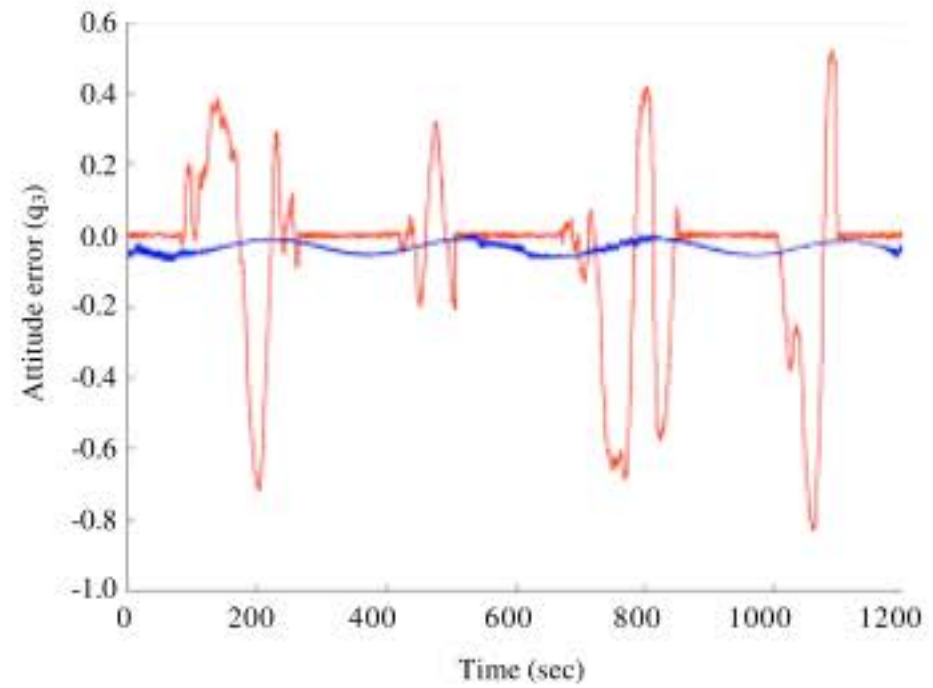


Fig. 8: Attitude error estimated state  $q_3$

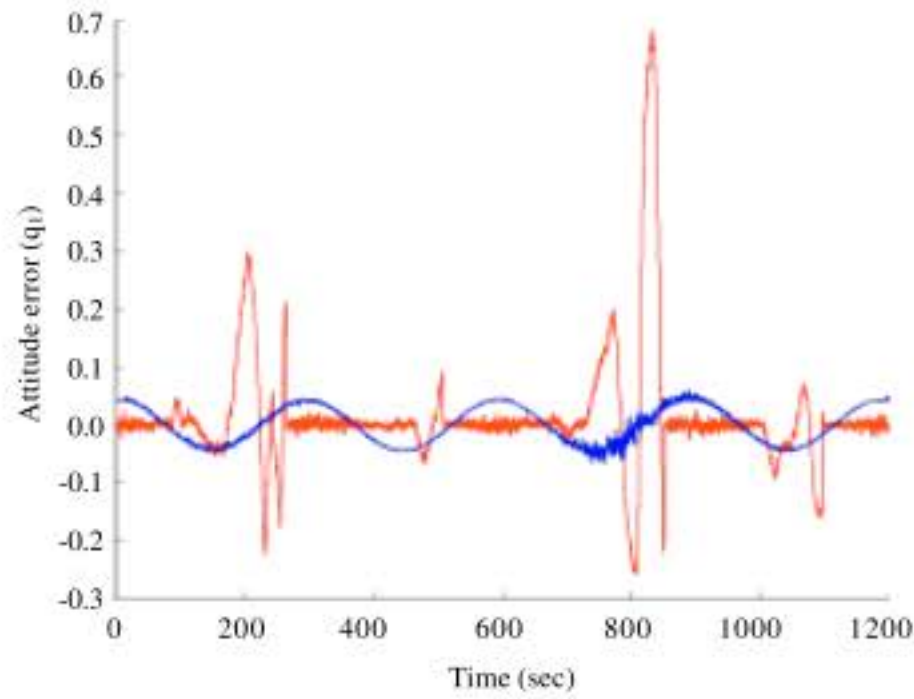


Fig. 6: Attitude error estimated state  $q_1$

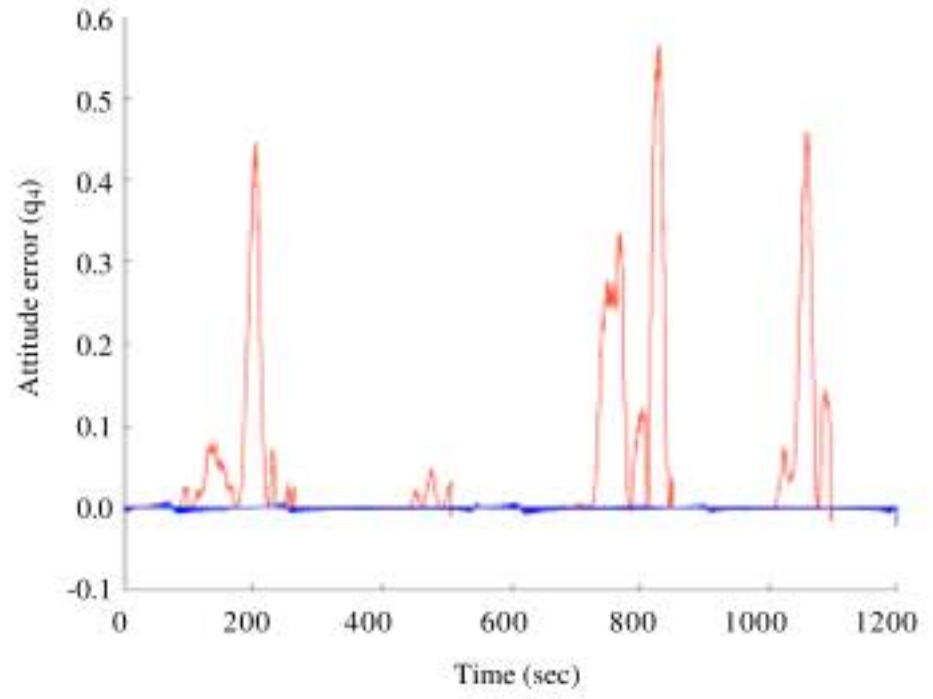


Fig. 9: Attitude error estimated state  $q_4$

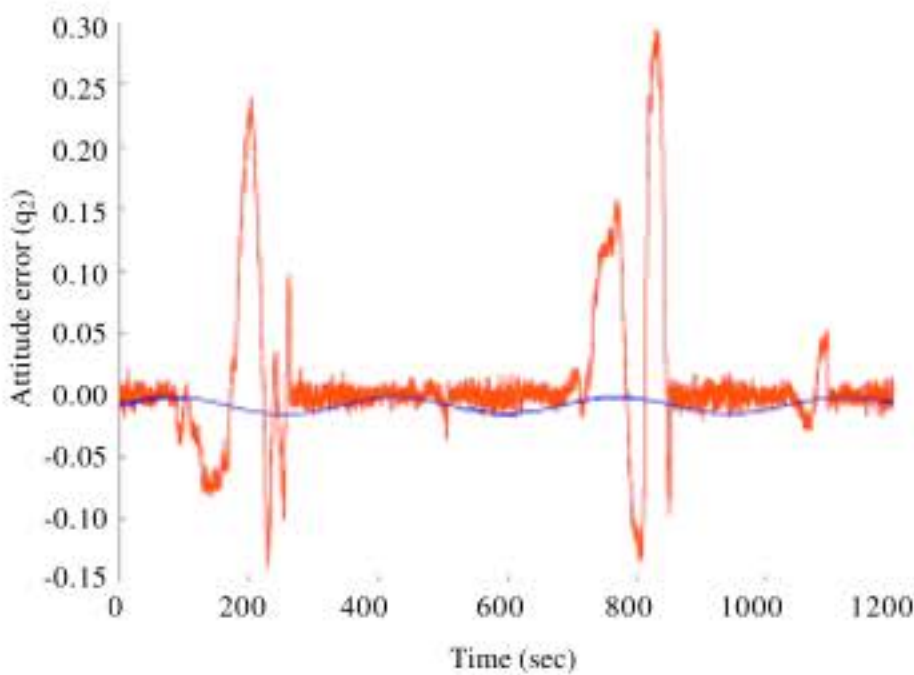


Fig. 7: Attitude error estimated state  $q_2$

red graph represents the EKF outputs and finally the blue one gives the UKF outputs. The EKF and UKF outputs for estimating quaternion ( $q_1$ ,  $q_2$ ,  $q_3$  and  $q_4$ ) are shown in

Fig. 2 for  $q_1$ , Fig. 3 for  $q_2$ , Fig. 4 for  $q_3$  and Fig. 5 for  $q_4$ . The quantities obtained from the both filters are compared with the real states in black line as showed in the Fig. 2-5. It can be seen that the performance of the EKF and the UKF is identical; this is to be expected since they both assume additive white noise. However, the UKF have shown better performances in presence of the severe nonlinearity in state equations.

The attitude error calculated between the real quantity and estimated one are illustrated in Fig. 6 for  $q_1$ , Fig. 7 for  $q_2$ , Fig. 8 for  $q_3$  and Fig. 9 for  $q_4$ , it is clear that the attitude converge to zero quickly. We can see from Fig. 6-9 that the more severe the nonlinearity of the state dynamics is the more error occurs in states estimation obtained by EKF. The new Kalman filter UKF has also shown better performances in presence of the severe nonlinearity, as seen in the blue graph in Fig. 6-9 for error estimation.



## DISCUSSION

The UKF as a tool for state estimation has been compared to the standard method for nonlinear state estimation the EKF. The state estimation methods have been compared using the same tuning parameters to make the comparative study as credible as possible.

The UKF has been shown to be superior to the EKF in many state estimation, dual estimation and parameter estimation problems (Van der Merwe *et al.*, 2004). It has also been used in many applications, such as state estimation for road vehicles (Julier *et al.*, 1995), induction motors (Akin, 2003), quaternion motion (Joseph and Viola, 2003), visual contour tracking (Li and Zhang, 2002) and parameter estimation for time series modelling (Wan *et al.*, 2000) and neural network training.

In present study, we have used the new alternative filter to space application and the results obtained show the performance of UKF over EKF. The UKF was able to converge with poor initial estimates of the parameters, while the EKF was shown to have a greater tendency to diverge due to poor initial estimates of the parameters. In general, it can be expected that the performance of the UKF is better than that of the EKF, since it propagates the state noise more accurately.

## CONCLUSION

In this study, we have argued the principle difficulty for applying the Kalman filter to non linear systems is the need to consistently predict the new state and observation of the system. We have introduced another approach of filtering, called the unscented filter, which eliminates the need explicitly derive Jacobian matrices, permitting the implementation of highly complex nonlinear system models previously considered intractable. The benefits of the algorithm were demonstrated in realistic example and we anticipate that the technique described should find widespread application in control and estimation applications.

## REFERENCES

Akin, B., U. Orguner and A. Ersak, 2003. State estimation of Induction motor using Unscented Kalman filter. Proc. IEEE Conf. Control Appl., 2: 915-919.  
Anderson, A.D., J.J. Sellers and Y. Hashida, 2004. Attitude determination and control system simulation and analysis for Low-cost Micro-satellites. Proceedings of the IEEE Aerospace Conference, Mar. 6-13, USAF Acad., USA., pp: 2935-2949.

Chowdhary, G. and R. Jategaonkar, 2006. Aerodynamic parameter estimation from flight data applying extended and unscented Kalman filter. Proceedings of the AIAA Atmospheric Flight Mechanics Conference and Exhibit, Aug. 21-24, DLR Institute for Flight System, Braunschweig, Germany, pp: 1-18.  
Costa, P.J., 1994. Adaptive model architecture and extended Kalman-bucy filters. IEEE Trans. Aerospace Elect. Syst., 30: 525-533.  
Hashida, Y., 2004. ADCS for Future UoSat Standard Platform: Surrey Satellite Technology Limited Internal Technical Note. Surrey Satellite Technology Ltd., GUILFORD, UK.  
Joseph, J. and L. Viola Jr. 2003. A comparison of unscented and extended kalman filtering for estimating quaternion motion. Proc. Am. Control Conf., 3: 2435-2440.  
Julier, S. and J.K. Uhlmann, 1994. A general method for approximating nonlinear transformations of probability distributions. Technical Report, Robotics Research Group, Department of Engineering Science, University of Oxford.  
Julier, S., J. Uhlmann and H.F. Durant-Whyte, 1995. Process models for the high-speed navigation of road vehicles. IEEE Int. Robotics Automation Conf., 1: 101-105.  
Julier, S.J. and J.K. Uhlmann, 1997. A new extension of the kalman filter to nonlinear systems. Proceedings of the AeroSense: The 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls, Apr. 21-24, Orlando, Fl., pp: 1-12.  
Julier, S., J. Uhlmann and H.F. Durrant-Whyte, 2000. A new method for the nonlinear transformation of means and covariances in filters and estimators. IEEE Trans. Auto. Control, 45: 477-482.  
Kandepu, R., L. Imsland and B.A. Foss, 2008. Constrained state estimation using the unscented Kalman filter. Proceedings of the 16th Mediterranean Conference on Control and Automation Congress Centre, Jun. 25-27, Ajaccio, France, pp: 1453-1458.  
Kushner, H.J. and A.S. Budhiraja, 2000. A nonlinear filtering algorithm based on an approximation of the conditional distribution. IEEE Trans. Automatic Control, 45: 580-585.  
Li, P. and T. Zhang, 2002. Unscented Kalman filter for visual curve tracking. Proceedings of the Workshop Statistical Methods Video Processing, Jun. 1-2, Copenhagen, Denmark, pp: 13-18.  
Martin, R., 2001. Noise power spectral density estimation based on optimal smoothing and minimum statistics. IEEE Trans. Speech Audio Process., 9: 504-512.



- Matthew, C.C., J. Hale, P. Vergez, M.J. Meerman and Y. Hashida, 2004. Kalman filtering and the attitude determination and control task. Proceedings of the Space 2004 Conference and Exhibit, Sept. 28-30, San Diego, California, pp: 1-1.
- Van der Merwe, R., E. Wan and S.J. Julier, 2004. Sigma-point Kalman filters for nonlinear estimation and Sensor-fusion: Applications to integrated navigation. Proceedings of the AIAA Guidance, Navigation and Control Conference, Aug. 2004, Providence, RI., pp: 1-14.
- Viéville, T., E. Clergue and P.E.D.S. Facao, 1993. Computation of ego-motion and structure from visual and internal sensors using the vertical cue. Proceedings of IEEE International Conference on Computer Vision, May 1993, Berlin, pp: 591-598.
- Wan, E.A., R. van der Merwe and A.T. Nelson, 2000. Dual estimation and the unscented. Neural Inform. Process. Syst., 12: 666-672.