ISTITUTO DI ANALISI DEI SISTEMI ED INFORMATICA
"Antonio Ruberti"
CONSIGLIO NAZIONALE DELLE RICERCHE

T. Fiorenzani, C. Manes, G. Oriolo, P. Peliti

COMPARATIVE STUDY OF UNSCENTED KALMAN FILTER AND EXTENDED KALMAN FILTER FOR POSITION/ATTITUDE ESTIMATION IN UNMANNED AERIAL VEHICLES

R. 08-08 2008

Tiziano Fiorenzani – UTRI S.p.A, Viale del Lido 37, 00122 Roma, Italy, E-mail: tfiorenzani@utri.it.

Costanzo Manes – Dipartimento di Ingegneria Elettrica e dell'Informazione, Università degli Studi dell'Aquila, Monteluco di Roio, 67040 L'Aquila, Italy, E-mail: costanzo.manes@univaq.it.

Giuseppe Oriolo – Dipartimento di Informatica e Sistemistica, Università degli Studi di Roma "Sapienza", via Ariosto 25, 00185 Roma, Italy, E-mail: oriolo@dis.uniroma1.it.

Pietro Peliti – UTRI S.p.A.,Viale del Lido 37, 00122 Roma, Italy, E-mail: ppeliti@utri.it.

This work was partially supported by IASI-CNR

ISSN: 1128–3378

Collana dei Rapporti dell'Istituto di Analisi dei Sistemi ed Informatica "Antonio Ruberti", CNR

viale Manzoni 30, 00185 ROMA, Italy

tel. ++39-06-77161
fax ++39-06-7716461
email: iasi@iasi.rm.cnr.it
URL: http://www.iasi.rm.cnr.it

Abstract

State estimation is a very common task in many engineering applications involving dynamic systems. For linear systems corrupted by gaussian noises the problem of optimal state estimation is solved by the well known Kalman filter, which is a finite-dimensional system. In the general case of non-linear/non-gaussian stochastic systems, the optimal state estimator (the *Bayesian estimator*) is an infinite-dimensional system that provides the conditional probability density of the state given the measurements. Only finite-dimensional approximations of the Bayesian estimator can be implemented in practice, but accurate approximations are computationally prohibitive in most applications. For this reason, many non-optimal filtering techniques have been proposed in the literature, where optimality of the estimate is traded off against lower computational cost. One of the most used estimation algorithms is the so called Extended Kalman Filter (EKF), that is constructed by applying the simple equations of the Kalman filter to the first order Taylor approximation of the equations that define the nonlinear system, computed around the most recent state estimate. The main advantage of the EKF is the simplicity of its construction. However, the use of Jacobians in the EKF has been criticized by many authors, and Jacobian-free filtering methods have been proposed. Among these, the so called Unscented Kalman Filter (UKF) has recently received considerable attention. Advantages and disadvantages of the EKF w.r.t. the UKF are very difficult to be theoretically assessed. It is easier to make comparisons on some specific engineering application. In this report the performances of the two filters are compared when applied to the state estimation of a rotorcraft Unmanned Aerial Vehicle (UAV). The UAV considered is the MHELI, a small size Vertical Take-Off and Landing aircraft (VTOL) developed by the Unmanned Technologies Research Institute S.p.A., equipped with an Inertial Measurement Unit and a GPS receiver.

Key words: Nonlinear filtering, Kalman filter, UAV, INS/GPS fusion

1. Introduction

The issue of state estimation arises in many engineering applications involving dynamic systems, when the measurements are incomplete and affected by random noises. A filter is an algorithm that computes an estimate of the state of a given dynamic system by recursively processing the sequence of available noisy measurements. A filter is said to be optimal when it provides the best state estimate, according to some specific criterion. Bayesian filters are those that provide estimates with minimum error variance, i.e. the conditional expectation of the state given the measurements. It is known that for *linear systems* corrupted by *additive gaussian noises*, the problem of optimal state estimation is solved by the well known Kalman filter, that is a finite-dimensional system characterized by an elegant linear form that recursively provides the state estimate together with the error covariance matrix. It is interesting to note that when the noises are gaussian, the sequences of state estimates and of error covariances provided by the Kalman Filter coincide with the mean values and covariance matrices of the (gaussian) conditional distribution of the state. On the other hand, if the noises are not gaussian the Kalman Filter does not provide the optimal state estimate. However, it can be proved that the Kalman Filter is still the best among any other linear filtering algorithm, although in this case it does not provide the mean and covariance of the conditional distribution.

In the general case of non-linear/non-gaussian stochastic systems the Bayesian estimator is an infinite-dimensional system that provides the conditional probability density of the state given the measurements (differently from gaussian distributions, generic distributions are not simply characterized by the mean and the covariance of the random vector). As a consequence, only finite-dimensional approximations of the Bayesian estimator can be implemented in practice, but accurate approximations are computationally prohibitive in most applications. For this reason, many non-optimal filtering techniques have been proposed in the literature, where optimality of the estimate is traded off against lower computational cost.

One of the most used estimation algorithms is the so called Extended Kalman Filter (EKF), which is a quite natural extension of the Kalman Filter, and preserves its prediction-correction structure. The EKF accounts for nonlinearities by using the first order Taylor approximation (linearization) of both the transition and measurement equations of the nonlinear system. In the prediction step the state transition function is linearized around the last available state estimate, while in the measurement update step, the output function is linearized around the one-step state prediction. The underlying assumption is that the error incurred by neglecting the higher-order terms is small in comparison to the first-order terms. As a consequence The EKF preserves the simplicity of the standard Kalman Filter equations and is very easy to implement. However, if the first order Taylor approximation is too coarse, maybe because the estimation error on the last available or the additive noises are too large, the EKF may not give satisfactory results, and may even diverge. An interesting approach is to use higher order Taylor approximations of the system equations for the filter derivation (Polynomial Extended Kalman Filter [1]). The disadvantage of this approach is that the complexity increases with the approximation degree: whereas the EKF requires the computation of the Jacobians (first order derivatives) of the state transition and of the measurement equations, the PEKF of a given order ν requires the computation of higher order derivatives, up to the order ν.

Among the Jacobian-free methods aimed to overcome some of the shortcomings of the EKF, a considerable attention has received the so called Unscented Kalman Filter (UKF), developed by J. Julier and and J. Uhlmann [5]. The UKF is an extension of the Kalman Filter that implements the so called unscented transformation for the propagation of means and covariances.

The unscented transformation uses a set of samples, called sigma points, that approximate the a priori mean and covariance of the state. The sigma points are transformed by the true nonlinear transformation and allow the computation the posterior mean and covariance of the state. In many applications the UKF has shown better accuracy and convergence capability than the EKF.

In this paper we compare the performances of the EKF and UKF, both in terms of estimation accuracy and computational complexity, by applying them to the problem of state estimation of MHELI, a small size Vertical Take-Off and Landing (VTOL) Unmanned Aerial Vehicle (UAV) developed by the Unmanned Technologies Research Institute SpA. MHELI is equipped with a wide range of sensors. The two principal on-board measurement systems are the Inertial Navigation System (INS), constituted by three accelerometers, three gyros and three magnetometers, and the GPS receiver. A barometer is also included to measure the altitude of the UAV. The fusion of the outputs of these sensors by an estimation algorithm can considerably increase the accuracy of the attitude, position and velocity estimation of the UAV, and make the navigation system more dependable. GPS and INS have complementary capabilities: INS gives, with an high working frequency, position of the UAV integrating the accelerations and angular rate sensed by the inertial sensors. It doesn't require any external signal, making the vehicle totally autonomous. However the estimated position will eventually drift from the real position because of the accumulation of errors due to the sensors noise and numerically integration. The GPS receiver can measure the position of the vehicle to the required precision. However the GPS accuracy depends on external factors and a GPS solution could be unavailable for several seconds because of the occlusions of satellite signals.

The UAV state estimation problem is highly non linear because both the system and the sensors models present strong nonlinearities, and therefore it is an interesting application field for the comparison of the EKF and UKF.

This report is organized as follows: in section 2 the EKF and UKF are presented, in section 3 the process and measurement models describing the MHELI are explained, in section 4 the filters implementation issues and the numerical results are described, finally in section 5 some conclusions are discussed.

2. Background on EKF and UKF

2.1. Process and measurement models

The EKF and UKF are algorithms aimed to the state estimation of stochastic discrete-time systems described by equations of the type

$$\begin{aligned} x_{k+1} &= f(x_k, u_k, v_k, k), \\ y_k &= h(x_k, n_k, k), \end{aligned} \quad k = 0, 1, \ldots, \tag{2.1.1}$$
$$\begin{aligned} x_k &\in I\!R^{N_x}, \\ u_k &\in I\!R^{N_u}, \\ y_k &\in I\!R^{N_y}, \end{aligned} \quad \begin{aligned} v_k &\in I\!R^{N_v}, \\ n_k &\in I\!R^{N_n}, \end{aligned}$$

where f and h are non-linear functions, x_k is the system state, u_k is a known input, y_k is the measured output. v_k and n_k are the process noise and measurement noise sequences, respectively.

The two noise sequences are assumed to be independent and stationary, with known means and covariances

$$\begin{aligned} \bar{v} &= E\left\{v_k\right\}, \quad R_v = E\left\{(v_k - \bar{v})(v_k - \bar{v})^T\right\}, \\ \bar{n} &= E\left\{n_k\right\}, \quad R_n = E\left\{(n_k - \bar{n})(n_k - \bar{n})^T\right\}. \end{aligned} \tag{2.1.2}$$

The initial state x_0 is a stochastic vector with known mean and covariance, independent of v_k and n_k:

$$\bar{x}_0 = E\left\{x_0\right\}, \qquad P_{x_0} = E\left\{(x_0 - \bar{x}_0)(x_0 - \bar{x}_0)^T\right\}. \tag{2.1.3}$$

2.2. Extended Kalman Filter

The EKF is a quite natural extension of the standard Kalman Filter equations for dealing with the state estimation problem of nonlinear systems of the type (2.1.1). The EKF recursively estimate its state starting from the last time step estimation and from the actual measurements and inputs. It consists of two different phase: the Time-Update (prediction) and the Measurement-Update (correction). In the Time-Update the last available estimation and its covariance are propagated through the model equations to obtain a prior estimation of the actual state $\hat{x}_k^-$. In the Measurement-Update the posterior estimation $\hat{x}_k$ is obtained from the correction of the prior by including the informations obtained from the measurements.

The EKF needs the system to be linearized obtaining the Jacobian matrixes:

$$A_k = \left.\frac{\partial f(x, u_k, \bar{v}, k)}{\partial x}\right|_{\hat{x}_k} \tag{2.2.1}$$

$$F_k = \left.\frac{\partial f(\hat{x}_k, u_k, v, k)}{\partial v}\right|_{\bar{v}} \tag{2.2.2}$$

$$H_k = \left.\frac{\partial h(x, \bar{n}, k)}{\partial x}\right|_{\hat{x}_k^-} \tag{2.2.3}$$

$$G_k = \left.\frac{\partial h(\hat{x}_k^-, n, k)}{\partial n}\right|_{\bar{n}}. \tag{2.2.4}$$

The needs of linearization is the largest shortcoming of the EKF. In case of strong nonlinearities the approximation by a Taylor series expansion can lead to instability of the filter and in some case they may not exist. The EKF is summarized in algorithm 1. For more detail see [2].

Algorithm 1 Extended Kalman Filter (EKF)

- *Initialization:*

$$\begin{aligned}
\hat{x}_0 &= E\{x_0\} && (2.2.5)\\
P_{x_0} &= E\left\{(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T\right\}\\
R_v &= E\left\{(v_k - \bar{v})(v_k - \bar{v})^T\right\}\\
R_n &= E\left\{(n_k - \bar{n})(n_k - \bar{n})^T\right\}
\end{aligned}$$

- `for` $k = 1, \ldots, \infty$

1. Computation of the Jacobian matrixes A_{k-1} and F_{k-1}:

$$\begin{aligned}
A_{k-1} &= \frac{\partial f(x, u_{k-1}, \bar{v}, k-1)}{\partial x}\Big|_{\hat{x}_{k-1}} && (2.2.6)\\
F_{k-1} &= \frac{\partial f(\hat{x}_{k-1}, u_{k-1}, v, k-1)}{\partial v}\Big|_{\bar{v}} && (2.2.7)\\
& && (2.2.8)
\end{aligned}$$

2. *Time-Update*: computation of the prediction and prediction error covariance

$$\begin{aligned}
\hat{x}_k^- &= f\left(\hat{x}_{k-1}, u_{k-1}, \bar{v}, k-1\right) && (2.2.9)\\
P_{x_k}^- &= A_{k-1} P_{x_{k-1}} A_{k-1}^T + F_{k-1} R_v F_{k-1}^T && (2.2.10)
\end{aligned}$$

3. Computation of the Jacobian matrixes H_k and G_k

$$\begin{aligned}
H_k &= \frac{\partial h(x, \bar{n})}{\partial x}\Big|_{\hat{x}_k^-} && (2.2.11)\\
G_k &= \frac{\partial h(\hat{x}_k^-, n)}{\partial n}\Big|_{\bar{n}} && (2.2.12)
\end{aligned}$$

4. *Measurement-Update*: prediction correction

$$\begin{aligned}
K_k &= P_{x_k}^- H_k^T \left(H_k P_{x_k}^- H_k^T + G_k R_n G_k\right)^{-1} && (2.2.13)\\
\hat{x}_k &= \hat{x}_k^- + K_k \left(y_k - h(\hat{x}_k^-, \bar{n}, k)\right) && (2.2.14)\\
P_{x_k} &= \left(I - K_k H_k\right) P_{x_k}^- && (2.2.15)
\end{aligned}$$

2.3. Unscented Kalman Filter

The Unscented Kalman Filter is a recursive method for state estimation in non linear systems, developed by J. Julier and and J. Uhlmann. It is based on a simple intuition: *it is easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function or transformation* [5]. Starting from this basic idea it is possible to find a parametrization which captures the mean and covariance information while at the same time permitting the direct propagation of the information through an arbitrary set of nonlinear equations. This can be done by sampling the prior distribution choosing a finite number of points so that their sample mean and sample covariance are the same of the original distribution. The points (called *sigma-points*) are then propagated through the non-linear transformation, obtaining a cloud of points in the transformed space, and their mean and covariance are computed. This method permits to avoid the linearization and takes the name of *Unscented Transformation* (UT).

It differs from a Monte Carlo Method in two relevant aspects:

1. in a Monte Carlo method the samples are randomly taken, instead in the UT they are selected following precise rules

2. the distribution is approximated by a small number of points, instead of the thousands of points usually needed by a Monte Carlo Method.

Let's consider the propagation of an N-dimensional gaussian variable x, of mean $\bar{x}$ and covariance P_{xx}, through an arbitrary nonlinear function f:

$$y = f(x). \tag{2.3.1}$$

To obtain the first and second order moments of y, $N+1$ sigma points $\mathcal{X}$ are selected and a weight W is assigned to each of them:

$$\begin{aligned}
\mathcal{X}_0 &= \bar{x} & W_0 &= \kappa/(N+\kappa) & \\
\mathcal{X}_i &= \bar{x} + \left(\sqrt{(N+\kappa)P_{xx}}\right)_i & W_i &= 1/2(N+\kappa) & i = 1, \ldots, N \\
\mathcal{X}_{i+N} &= \bar{x} - \left(\sqrt{(N+\kappa)P_{xx}}\right)_i & W_{i+N} &= 1/2(N+\kappa) & i = 1, \ldots, N
\end{aligned} \tag{2.3.2}$$

where $\sum_{i=0}^{2N} W_i = 1$, $\left(\sqrt{(N+\kappa)P_{xx}}\right)_i$ is the i-th column of the square root weighted covariance matrix $(N+\kappa)P_{xx}$ and κ is a scalar parameter. Since there are several square-roots of a matrix, any orthonormal rotation of a set of sigma-points is still a valid set.

The sigma points are then propagated trough the non linear function obtaining the points

$$\mathcal{Y}_i = f(\mathcal{X}_i)\ ,\ i = 0, \ldots 2N, \tag{2.3.3}$$

and the predicted mean and covariance are calculated:

$$\bar{y} = \sum_{i=0}^{2N} W_i \mathcal{Y}_i \tag{2.3.4}$$

$$P_{yy} = \sum_{i=0}^{2N} W_i [\mathcal{Y}_i - \bar{y}][\mathcal{Y}_i - \bar{y}]^T \tag{2.3.5}$$

$$\tag{2.3.6}$$

In [4] is shown that the mean and covariance of y, obtained by the Taylor series linearization, are respectively corrupted at the second and fourth order. Instead, using the UT, both mean and covariance have a fourth order precision. Further the higher order errors can be reduced by a proper choice of κ.

The dispersion of the sigma points from the mean increases proportionally to N because the distance between the generic point $\mathcal{X}_i$ and $\bar{x}$ is proportional to $\sqrt{N+\kappa}$. Although the mean and covariance of the prior are still correctly captured it can be a problem in presence of strong nonlinearties. In this case the large dispersion of the points can bring to sample nonlocal effects. To solve this problem the dispersion can be increased or reduced by changing the parameter κ: for $\kappa > 0$ the distance between the points and $\bar{x}$ increases, for $\kappa < 0$ it becomes smaller. However the choice of a negative value for κ can result in a non positive definiteness of the predicted covariance. To overcome this shortcoming a modified version of the UT, called *Scaled Unscented Transformation* (SUT), has been developed [6].

The SUT introduces other degrees of freedom in the choice of the sigma points dispersion and avoids the non-positive definiteness of the covariance, by adding two scaling parameter α and β. α controls the distance of the points from the mean and has to be chosen between 0 and 1. β is a non negative parameter that is used to add informations about the third and higher moments of the probability function, if they are known. For gaussian prior the optimal value is $\beta = 2$ [6].

The Unscented Kalman Filter is a Kalman Filter where the mean and covariance prediction of the system state are obtained by the SUT. To incorporate the effect of the process and measurements noise the state space is augmented with noise variables. The augmented state is:

$$x_k^a = \begin{bmatrix} x_k^x \\ x_k^v \\ x_k^n \end{bmatrix} = \begin{bmatrix} x_k \\ v_k \\ n_k \end{bmatrix} \tag{2.3.7}$$

and its dimension is $N = N_x + N_v + N_n$ where N_x is the system state dimension and N_v and N_n are respectively the process and measurement noise dimensions. They are supposed to be gaussian and zero-mean with covariance matrices R_v and R_n. The augmented state covariance is:

$$P^a = \begin{bmatrix} P_x & 0 & 0 \\ 0 & R_v & 0 \\ 0 & 0 & R_n \end{bmatrix}. \tag{2.3.8}$$

The UKF is summarized in algorithm 2.

The most computational burning operation is the calculation of the square-root of the covariance matrix in the sigma points extraction. To reduce the computational complexity is possible to use the square root form of the algorithm called the *Square-root Unscented Kalman Filter*. In this version the square root of the covariance matrix is directly propagated in the form of its Cholesky factorization, preventing its extraction at each iteration. The algorithm needs some algebra techniques like QR decomposition, Cholesky factor updating and efficient pivot-based least squares, for more details see [7] and [8]. It has the same accuracy of the standard UKF form and better numerical stability, for these reasons we implemented this form for our purpose. The SRUKF is fully described in algorithm 3.

Algorithm 2 Unscented Kalman Filter (UKF)

- *Initialization:*

$$\hat{x}_0 = E\{x_0\} \qquad P_{x_0} = E\{(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T\}$$

$$\hat{x}_0^a = E\{x_0^a\} = [\hat{x}_0 \quad 0 \quad 0]^T \qquad P_0^a = E\{(x_0^a - \hat{x}_0^a)(x_0^a - \hat{x}_0^a)^T\} = \begin{bmatrix} P_{x_0} & 0 & 0 \\ 0 & R_v & 0 \\ 0 & 0 & R_n \end{bmatrix}$$

- for $k = 1, \ldots, \infty$

1. Sigma-points extraction:

$$\mathcal{X}_{k-1}^a = \left[\hat{x}_{k-1}^a \quad \hat{x}_{k-1}^a + \sqrt{(N+\lambda)P_{k-1}^a} \quad \hat{x}_{k-1}^a - \sqrt{(N+\lambda)P_{k-1}^a}\right] \tag{2.3.9}$$

2. Time-Update:

$$\mathcal{X}_{k|k-1}^a = f\left(\mathcal{X}_{k-1}^x, u_{k-1}, \mathcal{X}_{k-1}^v, k-1\right) \tag{2.3.10}$$

$$\hat{x}_k^- = \sum_{i=0}^{2N} W_i^{(m)} \mathcal{X}_{i,k|k-1}^x \tag{2.3.11}$$

$$P_{x_k}^- = \sum_{i=0}^{2N} W_i^{(c)} [\mathcal{X}_{i,k|k-1}^x - \hat{x}_k^-][\mathcal{X}_{i,k|k-1}^x - \hat{x}_k^-]^T \tag{2.3.12}$$

3. Measurement-Update:

$$\mathcal{Y}_{k|k-1}^a = h\left(\mathcal{X}_{k|k-1}^x, \mathcal{X}_{k-1}^n, k\right) \tag{2.3.13}$$

$$\hat{y}_k^- = \sum_{i=0}^{2N} W_i^{(m)} \mathcal{Y}_{i,k|k-1} \tag{2.3.14}$$

$$P_{y_k} = \sum_{i=0}^{2N} W_i^{(c)} [\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-][\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-]^T \tag{2.3.15}$$

$$P_{x_k y_k} = \sum_{i=0}^{2N} W_i^{(c)} [\mathcal{X}_{i,k|k-1}^x - \hat{x}_k^-][\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-]^T \tag{2.3.16}$$

$$K_k = P_{x_k y_k} P_{y_k}^{-1} \tag{2.3.17}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - \hat{y}_k^-) \tag{2.3.18}$$

$$P_{x_k} = P_{x_k}^- - K_k P_{y_k} K_k^T \tag{2.3.19}$$

- where:

$$\begin{array}{ll} \lambda = \alpha^2(N+\kappa) - N & \\ W_i^{(m)} = \frac{\lambda}{N+\lambda} & i = 0 \\ W_i^{(c)} = \frac{\lambda}{N+\lambda} + (1 - \alpha^2 + \beta) & i = 0 \\ W_i^{(m)} = W_i^{(c)} = \frac{1}{2(N+\lambda)} & i = 1, \ldots, 2N \end{array} \tag{2.3.20}$$

Algorithm 3 Square-Root Unscented Kalman Filter (SR-UKF)

- *Initialization:*

$$\hat{x}_0 = E\{x_0\} \qquad S_{x_0} = \mathsf{chol}\{E\{(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T\}\} \qquad S_v = \sqrt{R_v} \qquad S_n = \sqrt{R_n}$$

$$\hat{x}_0^a = E\{x_0^a\} = [\hat{x}_0 \quad 0 \quad 0]^T \qquad S_0^a = \mathsf{chol}\{E\{(x_0^a - \hat{x}_0^a)(x_0^a - \hat{x}_0^a)^T\}\} = \begin{bmatrix} S_{x_0} & 0 & 0 \\ 0 & S_v & 0 \\ 0 & 0 & S_n \end{bmatrix}$$

- For $k = 1, \ldots, \infty$

1. Sigma points extraction:

$$\mathcal{X}_{k-1}^a = \left[\hat{x}_{k-1}^a \quad \hat{x}_{k-1}^a + \sqrt{(N+\lambda)}S_{k-1}^a \quad \hat{x}_{k-1}^a - \sqrt{(N+\lambda)}S_{k-1}^a\right] \tag{2.3.21}$$

2. Time-Update:

$$\mathcal{X}_{k|k-1}^a = f\left(\mathcal{X}_{k-1}^x, u_{k-1}, \mathcal{X}_{k-1}^v, k-1\right) \tag{2.3.22}$$

$$\hat{x}_k^- = \sum_{i=0}^{2N} W_i^{(m)} \mathcal{X}_{i,k|k-1}^x \tag{2.3.23}$$

$$S_{x_k}^- = \mathsf{qr}\left\{\sqrt{W_1^{(c)}}\left(\mathcal{X}_{1:2N,k|k-1} - \hat{x}_k^-\right)\right\} \tag{2.3.24}$$

$$S_{x_k}^- = \mathsf{cholupdate}\left\{S_{x_k}^-, \mathcal{X}_{0,k|k-1} - \hat{x}_k^-, W_0^{(c)}\right\} \tag{2.3.25}$$

3. Measurement Update

$$\mathcal{Y}_{k|k-1}^a = h\left(\mathcal{X}_{k|k-1}^x, \mathcal{X}_{k-1}^n, k\right) \tag{2.3.26}$$

$$\hat{y}_k^- = \sum_{i=0}^{2N} W_i^{(m)} \mathcal{Y}_{i,k|k-1} \tag{2.3.27}$$

$$S_{y_k} = \mathsf{qr}\left\{\sqrt{W_1^{(c)}}\left(\mathcal{Y}_{1:2N,k|k-1} - \hat{y}_k^-\right)\right\} \tag{2.3.28}$$

$$S_{y_k} = \mathsf{cholupdate}\left\{S_{y_k}^-, \mathcal{Y}_{0,k|k-1} - \hat{y}_k^-, W_0^{(c)}\right\} \tag{2.3.29}$$

$$P_{x_k y_k} = \sum_{i=0}^{2N} W_i^{(c)} [\mathcal{X}_{i,k|k-1}^x - \hat{x}_k^-][\mathcal{Y}_{i,k|k-1} - \hat{y}_k^-]^T \tag{2.3.30}$$

$$K_k = \left(P_{x_k y_k} / S_{y_k}^T\right) / S_{y_k} \tag{2.3.31}$$

$$U = K_k S_{y_k} \tag{2.3.32}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - \hat{y}_k^-) \tag{2.3.33}$$

$$S_{x_k} = \mathsf{cholupdate}\left\{S_{x_k}^-, U, -1\right\} \tag{2.3.34}$$

3. UAV Modeling

3.1. Kinematic Model

To describe the attitude, position and velocity of the UAV we make use of different reference frames: the NED, the WGS84 and the Body-frame. The NED is the local inertial reference frame. It is fixed on a point on the ground, the N axis points to the geographic North, the E to the East and D (Down) is parallel with the gravitational field. The body-frame is fixed with respect to the UAV and has its origin in the its center of mass. The WGS84 is a geodetic reference frame, centered in the earth center of mass and fixed with respect to the earth rotation. The position in WGS84 are expressed in terms of latitude ϕ, longitude λ and altitude h (Fig 1).

The aircraft is described by its kinematic model. The state space is composed from the variables of position velocity and attitude. Other variables representing the sensor biases are added to permit them estimation. The global state space has 19 variables:

$$\begin{aligned} X &= [P^T \quad V^T \quad q^T \quad a_{bias}^T \quad \omega_{bias}^T \quad H_{bias}^T]^T \\ &= [\phi \quad \lambda \quad h \quad u \quad v \quad w \quad q_0 \quad q_1 \quad q_2 \quad q_3 \quad a_{x_b} \quad a_{y_b} \quad a_{z_b} \quad \omega_{p_b} \quad \omega_{q_b} \quad \omega_{r_b} \quad H_{x_b} \quad H_{y_b} \quad H_{z_b}]^T \end{aligned}$$

Variable	Description	Variable	Description
ϕ	latitude (WGS84)	a_{x_b}	accelerometer bias on x
λ	longitude (WGS84)	a_{y_b}	accelerometer bias on y
h	altitude (WGS84)	a_{z_b}	z accelerometer bias on z
u	x velocity (Body-frame)	ω_{p_b}	gyro bias on roll-rate
v	y velocity (Body-frame)	ω_{q_b}	gyro bias on pitch-rate
w	z velocity (Body-frame)	ω_{r_b}	gyro bias on yaw-rate
q_0	quaternion	H_{x_b}	magnetometer bias on x
q_1	quaternion	H_{x_b}	magnetometer bias on y
q_2	quaternion	H_{x_b}	magnetometer bias on z
q_3	quaternion		

The attitude of the UAV is described by the quaternions to avoid the typical singularities of

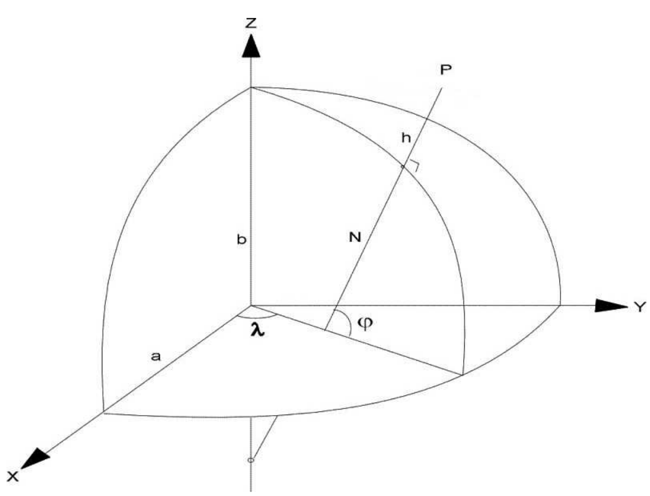


Figure 1: Geodetic reference frame.

the Euler angles representation. The quaternion dynamics is:

$$\dot{q} = -\frac{1}{2}\begin{bmatrix} 0 & \omega_p & \omega_q & \omega_r \\ -\omega_p & 0 & -\omega_r & \omega_q \\ -\omega_q & \omega_r & 0 & -\omega_p \\ -\omega_r & -\omega_q & \omega_p & 0 \end{bmatrix}\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{3.1.1}$$

where ω_p, ω_q and ω_r are respectively the roll-rate, pitch-rate and yaw-rate. The quaternion dynamic can be discretized by a closed form. Assuming constant angular rates during the sample time, the UAV rotations will be:

$$\begin{aligned} \Delta\phi &= \omega_p \cdot \Delta T \\ \Delta\theta &= \omega_q \cdot \Delta T \\ \Delta\psi &= \omega_r \cdot \Delta T, \end{aligned} \tag{3.1.2}$$

where ϕ, θ and ψ are the roll, pitch and yaw angles. Defining the skew matrix

$$\Phi_\Delta = \begin{bmatrix} 0 & \Delta\phi & \Delta\theta & \Delta\psi \\ -\Delta\phi & 0 & -\Delta\psi & \Delta\theta \\ -\Delta\theta & \Delta\psi & 0 & -\Delta\phi \\ -\Delta\psi & -\Delta\theta & \Delta\phi & 0 \end{bmatrix} \tag{3.1.3}$$

and the variable

$$s = \frac{1}{2}\sqrt{(\Delta\phi)^2 + (\Delta\theta)^2 + (\Delta\psi)^2} \tag{3.1.4}$$

the discrete time quaternion evolution will be:

$$q_{k+1} = \left[I(\cos(s) + \lambda \cdot j \cdot \Delta T) - \frac{1}{2}\Phi_\Delta \frac{\sin(s)}{s} \right] q_k. \tag{3.1.5}$$

In this equation $j = 1- \parallel q \parallel^2$ and λ is a Lagrange multiplier which forces the quaternion norm to the unit, avoiding the error due to numerical integration [15]. The numerical stability is ensured for $\lambda \cdot \Delta T < 1$ [12].

The velocity of the UAV is described in body-frame. The temporal evolution is obtained by the acceleration measured by the accelerometers purified from the terms don't due to the linear motion: the centripetal acceleration and gravity. The centripetal acceleration is due to the rotation of the UAV and to the offset of the IMU from the center of mass r_{IMU}:

$$\begin{bmatrix} a_{c_x} \\ a_{c_y} \\ a_{c_z} \end{bmatrix} = \begin{bmatrix} \omega_p \\ \omega_q \\ \omega_r \end{bmatrix} \times \begin{bmatrix} u \\ v \\ w \end{bmatrix} + \begin{bmatrix} \omega_p \\ \omega_q \\ \omega_r \end{bmatrix} \times \left[\begin{bmatrix} \omega_p \\ \omega_q \\ \omega_r \end{bmatrix} \times \begin{bmatrix} r_{IMU_x} \\ r_{IMU_y} \\ r_{IMU_z} \end{bmatrix} \right] \tag{3.1.6}$$

where the component due to angular acceleration are neglected. The component of gravity are obtained projecting the gravity vector into the body frame by the T_{bi}, the rotational matrix which converts a vector from NED to body-frame:

$$\begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = T_{bi} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.1.7}$$

where

$$T_{bi} = \begin{bmatrix} 1-2(q_2^2+q_3^2) & 2(q_1q_2+q_0q_3) & 2(q_1q_3-q_0q_2) \\ 2(q_1q_2-q_0q_3) & 1-2(q_1^2+q_3^2) & 2(q_2q_3+q_0q_1) \\ 2(q_1q_3+q_0q_2) & 2(q_2q_3-q_0q_1) & 1-2(q_1^2+q_2^2) \end{bmatrix}. \tag{3.1.8}$$

Representing the vector products by using a skew-symmetric matrix, the global acceleration is then:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} - \Omega \begin{bmatrix} u \\ v \\ w \end{bmatrix} - \Omega^2 \begin{bmatrix} r_{IMU_x} \\ r_{IMU_y} \\ r_{IMU_z} \end{bmatrix} - T_{bi} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.1.9}$$

where a_x, a_y and a_z are the global acceleration in body axis accelerations and

$$\Omega = \begin{bmatrix} 0 & -\omega_r & -\omega_q \\ \omega_r & 0 & -\omega_p \\ \omega_q & \omega_p & 0 \end{bmatrix}. \tag{3.1.10}$$

The UAV position is expressed in WGS84 in order to simplify the comparison with the GPS measurements. To obtain the position from the velocity, that are expressed in body-frame, two rotations are required: from body-frame to NED and from NED to WGS84:

$$\begin{aligned} \begin{bmatrix} \dot{\phi} \\ \dot{\lambda} \\ \dot{h} \end{bmatrix} &= \underbrace{\begin{bmatrix} \frac{1}{R_{long}+h} & 0 & 0 \\ 0 & \frac{1}{(R_{lat}+h)\cos\phi} & 0 \\ 0 & 0 & -1 \end{bmatrix}}_{T_{NW}} \begin{bmatrix} V_N \\ V_E \\ V_D \end{bmatrix} \\ &= T_{NW} T_{bi}^T \begin{bmatrix} u \\ v \\ w \end{bmatrix} \end{aligned} \tag{3.1.11}$$

Position and velocity equation can be discretized by a first order Euler discretization

$$\begin{aligned} P_{k+1} &= P_k + \dot{P}_k \Delta T \\ V_{k+1} &= V_k + \dot{V}_k \Delta T, \end{aligned}$$

where $\dot{P}$ and $\dot{V}$ are given by (3.1.11) and (3.1.9).

3.2. Sensor Model

The UAV is equipped with a large number of sensors. It has an IMU, containing three accelerometers, three gyros and three magnetometers, a GPS receiver and a barometer. Accelerometers and gyros measure accelerations and angular rates in body-frame and drive the kinematic model. They are supposed to be afflicted by bias and gaussian noise. The biases are supposed to be constant or slow-varying, them evolution can be represented like a random-walk process.

Magnetometers, GPS and barometer are instead used in the measurement-update. Magnetometers sense the magnetic field components around the body-axis, giving an indirect measure of the attitude:

$$H^{mag} = T_{bi} \cdot H_{Wgs84} + H_b + n_H \tag{3.2.1}$$

where H_{Wgs84} is the magnetic field vector in the geodetic position of the UAV, given by the standard WGS84, H_b is the vector of magnetometers biases and n_H is a gaussian noise.

The accelerometers also can give a measurements of the attitude. While being used like inclinometers, they sense the gravity vector components along the body-axis:

$$a^{incl} = T_{bi} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + a_b + n_a. \tag{3.2.2}$$

The measurement is accurate only in case of stationary flight, otherwise the vehicle accelerations will be summed to the gravity components.

GPS measures the latitude, longitude and altitude of the UAV. It has a variable accuracy dependents from several factors, like position of satellites and multi-path of the signal. The actual accuracy of the GPS data are described by the signal PDOP [14]. We summarize all sources of error in a gaussian noise variable with covariance varying according to the PDOP. The position error due to the GPS receiver displacement with respect to the center of mass r_{GPS} is negligeable, whereas it has to be considered in the velocity computation. The measurement model is:

$$P_k^{GPS} = P_k + n_{p_k} \tag{3.2.3}$$

$$V_k^{GPS} = T_{bi} V_k + T_{bi} \cdot [\omega_k \times r_{GPS}] + n_{v_k}. \tag{3.2.4}$$

Barometer give a measure of the altitude, sensing the atmospheric pressure. They are related by a non-linear transformation given by the *International Standard Atmosphere* (ISA) model:

$$p^{baro} = p_0 \left(1 - \frac{\Gamma h}{T_0}\right)^{\frac{g}{R_d \Gamma}} \tag{3.2.5}$$

where

$$\begin{aligned} h &= \text{altitude } [m] \\ \Gamma &= \text{temperature gradient } [K/m] \\ R_d &= \text{gas constant in the air } [J/Kg\,K] \\ g &= \text{gravity } [N] \\ p_0 &= \text{atmospheric pressure at sea level } [Pa] \\ p^{baro} &= \text{sensed atmospheric pressure}[Pa]. \end{aligned}$$

The altitude obtained by the barometer is with respect to the mean sea level, described in the standard EGM96 [11], where the one given by the GPS is with respect to the WGS84. When comparing the two data is so necessary to take count of the difference by a bias term $bias_{BaroGps}$ which represent the theoretical gap from the two measurements.

3.3. The Complete Model

The complete process and measurements models are now presented. The process model consists of the kinematic model and of the sensor biases evolution equations. The accelerations and angular rates are purified by the bias and noise terms:

$$\bar{a} = a - a_{bias} - v_a \tag{3.3.1}$$

$$\bar{\omega} = \omega - \omega_{bias} - v_\omega. \tag{3.3.2}$$

Then the complete non linear process model will be:

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = -\frac{1}{2} \begin{bmatrix} 0 & \bar{\omega}_p & \bar{\omega}_q & \bar{\omega}_r \\ -\bar{\omega}_p & 0 & -\bar{\omega}_r & \bar{\omega}_q \\ -\bar{\omega}_q & \bar{\omega}_r & 0 & -\bar{\omega}_p \\ -\bar{\omega}_r & -\bar{\omega}_q & \bar{\omega}_p & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{3.3.3}$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\lambda} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} \frac{1}{R_{long}+h} & 0 & 0 \\ 0 & \frac{1}{(R_{lat}+h)\cos\phi} & 0 \\ 0 & 0 & -1 \end{bmatrix} T_{bi}^T \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{3.3.4}$$

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \bar{a}_x \\ \bar{a}_y \\ \bar{a}_z \end{bmatrix} - \bar{\Omega} \begin{bmatrix} u \\ v \\ w \end{bmatrix} - \bar{\Omega}^2 \begin{bmatrix} r_{IMU_x} \\ r_{IMU_y} \\ r_{IMU_z} \end{bmatrix} - T_{bi} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.3.5}$$

$$\dot{a}_b = v_{a_b} \tag{3.3.6}$$

$$\dot{\omega}_b = v_{\omega_b} \tag{3.3.7}$$

$$\dot{H}_b = v_{h_b}. \tag{3.3.8}$$

where

$$\bar{\Omega} = \begin{bmatrix} 0 & \bar{\omega}_r & -\bar{\omega}_q \\ -\bar{\omega}_r & 0 & \bar{\omega}_p \\ \bar{\omega}_q & -\bar{\omega}_p & 0 \end{bmatrix}. \tag{3.3.9}$$

We can see that the additive noise sensor terms, once integrated in the model, results in a non-linear process noise.

The complete non-linear measurement model is:

$$y = \begin{bmatrix} H^{mag} \\ a^{incl} \\ p^{baro} \\ P^{GPS} \\ V^{GPS} \end{bmatrix} = h(x,u,n) = \begin{bmatrix} T_{bi} \cdot H_{Wgs84} + H_b + n_H \\ T_{bi} \cdot \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + a_b + n_a \\ p_0 \left(1 - \frac{\Gamma(h + bias_{BaroGps})}{T_0}\right)^{\frac{g}{R_d \Gamma}} + n_p \\ P + n_P \\ T_{bi} V + T_{bi} \cdot [\omega \times r_{GPS}] + n_V \end{bmatrix} \tag{3.3.10}$$

4. Implementation and Results

4.1. Filters Implementation

To use the process and measurements models into a digital estimation algorithm, they have to be digitalized. Furthermore the EKF requires also them linearization. Some considerations have to be done about the order in which exploit these two operations. In the previous section we showed that position and velocity can be discretized by a first order Euler discretization and biases by a random walk. The quaternion dynamics instead has a non-linear digital closed form. Linearizing the velocity and position by a first order Taylor expansion, we will have the same result independently by the order of linearization and discretization. Instead the quaternion dynamic is linear in continuous time but non-linear in discrete time, the order of the two operations is so more relevant. To conserve the stability of the closed form solution we choose to linearize the non-linear digital form, instead of discretize the linear continuous time form. For more detail and for a comparison of the results obtained by a different choice see [9].

A critical phase of the filter implementation is the initialization. The sensor noise covariance has been obtained by the calibration of the real sensors. The artificial noise which drives the bias estimation can be tuned to help the convergence. The biases are supposed constant or slow-varying, so the artificial noise must have a small covariance. In this way it prevents rapid variation of the estimate and avoids, for example, that large accelerations could be interpreted from the filter like a picks of accelerometers bias.

Another key parameter is the initial covariance of state estimation error. For attitude and position relatively large values have been chosen, where for velocity and biases a very small one. In fact biases are unknown but very small, because the sensors are already calibrated.

A relevant aspect to take in count during filters implementation is the sensor signal timing. The IMU works at $100Hz$ and the GPS at $4Hz$, but, because of external factors, a GPS measure can also be unavailable for a long period. The filters work at $40Hz$, and so at each iteration a new IMU signal will be available but not always a new GPS data. It is signed by the GPS flag, so, first of include the GPS in the Measurements-Update, a test on it is necessary.

4.2. Vehicle Operating Mode

To ensure a better estimation, the flight has been divided in phases called Vehicle Operating Mode (VOM), and a different set of filters parameters have been chosen for each phase. The three phases are *Initialization*, *Warm-Up* and *Ready*.

During the Initialization the filter parameters and states are initialized. Moreover the $bias_{BaroGps}$ is calculated by an average of the measures of the two sensors.

The Warm-Up is performed while the UAV is still on the ground. Using the certain information about the zero velocity of the vehicle, is possible to obtain a better and faster estimation of the initial attitude and sensor biases. The GPS velocity noisy measurement is substituted in the Measurements-Update by an artificial null measure, with zero covariance.

Trying to estimate both the accelerometers and magnetometers bias while the vehicle doesn't move, brings to a slower convergence of that variables. This happen because both the sensors are used for attitude estimation and the effect that their bias have on it are not easily observable. For this reason the magnetometer bias estimation is stopped during the Warm-Up. To see the effect that brings a different choice see [9].

In the Ready phase the UAV is ready to fly. The GPS velocity signal is fed trough the filter and the magnetometer bias estimation is activate.

4.3. Simulator

To validate the filters they have been tested in simulation using an high fidelity simulator developed in U.T.R.I.. The simulator implements accurately the dynamic model of the UAV and realistic models of the sensors. Particular attention has been devoted to a realistic simulation of the GPS signals. The dynamic model of the UAV is driven by the commands coming from a joystick.

4.4. Numerical Result

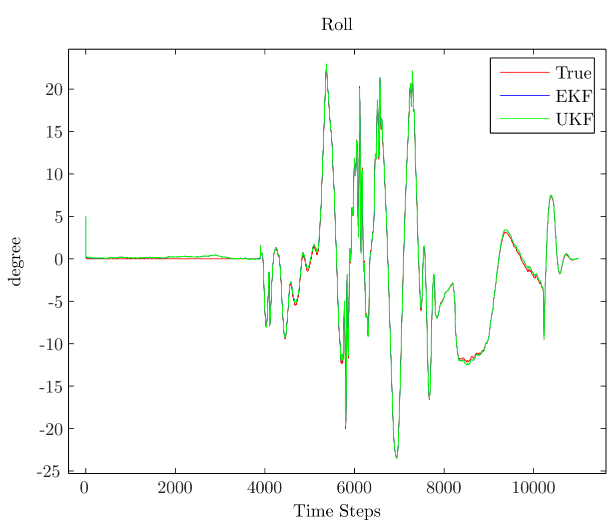
A set of different test have been developed in order to have a complete comparisons between EKF and SRUKF

The first simulation takes in count the typical condition of a real mission, considering an initial attitude estimation error of about 5 degrees on each attitude angle. The initial attitude for an UAV VTOL is really important because the trust direction depends on it. To have a precise vertical take off the UAV has to be well oriented with respect to the ground (often a stabilized platform is used for this task), and so the initial attitude is well known and usually the initial error will not be larger than 5 degrees. The initial position error is about 10 meters and the velocity error is null.

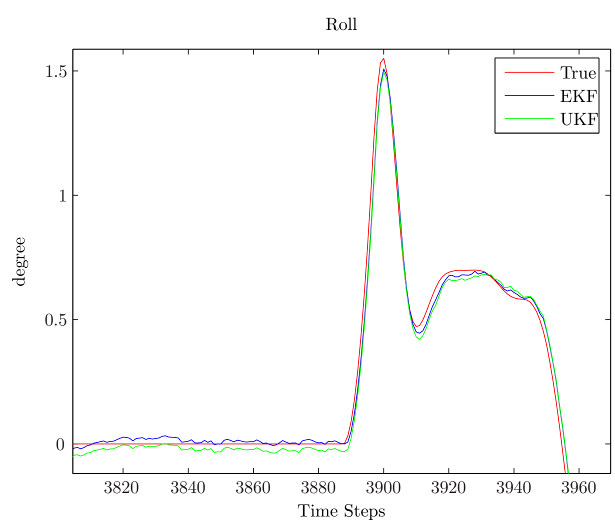
To help the results readability the attitude is showed in Euler angles and the position is converted from degree to meters, passing from geodetic position to NED position.

To test the filters in the worst working conditions we chose a really aggressive trajectory.

The results of the first simulation are showed in figures 2-12. As we can see the two filters performance are almost identical and both give a very precise estimation of the state of the UAV. The initial error on pitch and roll is instantly reduced, and it further decreases after the take off. The yaw error estimation remains large during the warm-up and decreases only after the take-off. The yaw angle is mainly observable by the magnetometers measurements and so it converges after the estimation of them biases. As we can see in figure 12 (a) the gyro biases are estimated very quickly, this because during the Warm-Up, using the certain information of the null velocity of the vehicle, the gyro output consist only of noise and bias, and so they are easily observable. Also the y and z axes accelerometers biases have a rapid convergence. Instead a_{b_x}, not observable while the UAV is on the ground, converges after the take off, when the acceleration components of the UAV are summed to the gravity vector. The magnetometers bias estimation is activated in the ready phase and rapidly converges. From figures 6 7 is possible to note that position and velocity estimation is more precise than GPS measurements. These are represented like a square signal because of the lower working frequency respect to the filter. The altitude estimation error is smaller than others thanks to the fusion of barometer and GPS data. In figure 5 is shown how the quaternion norm, thanks to the lagrange term λ of equation, is close to the unit.

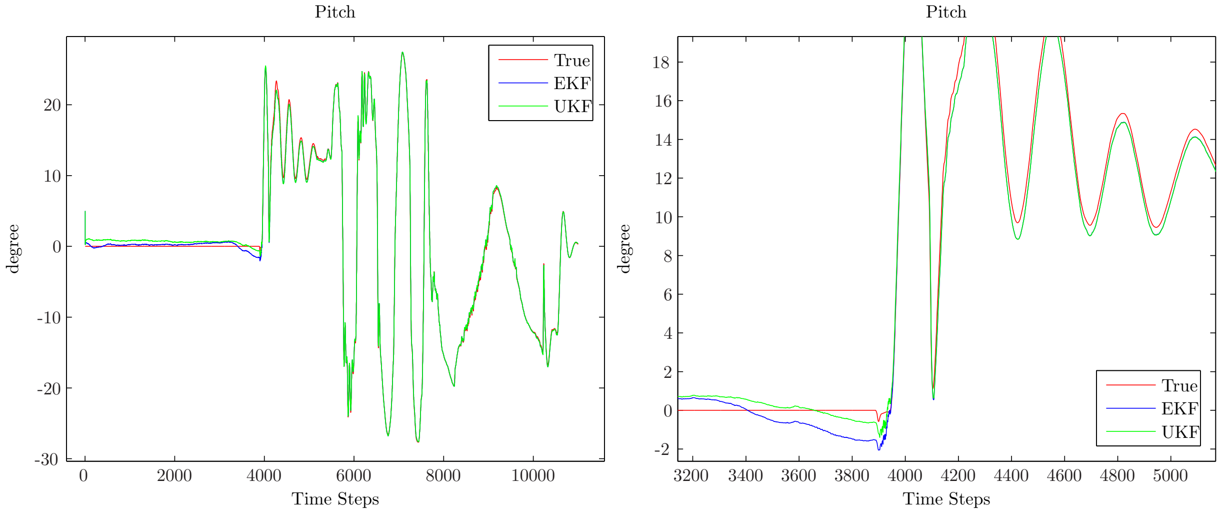


(a) Roll



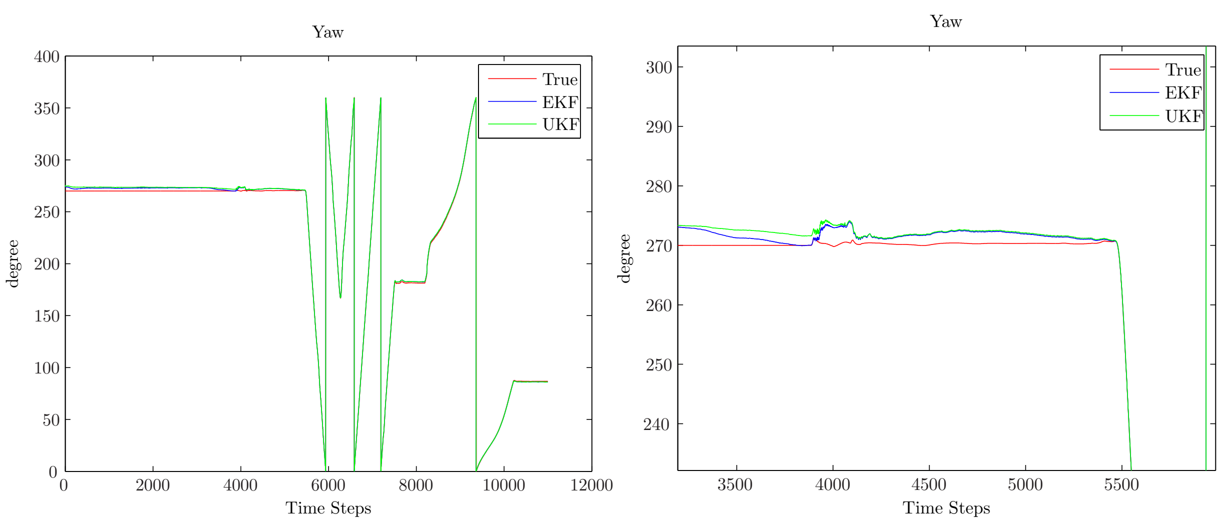
(b) Roll: particular

Figure 2: EKF and UKF comparison: Roll estimation.



(a) Pitch (b) Pitch: particular

Figure 3: EKF and UKF comparison: Pitch estimation.



(a) Yaw (b) Yaw: particular

Figure 4: EKF and UKF comparison: Yaw estimation.

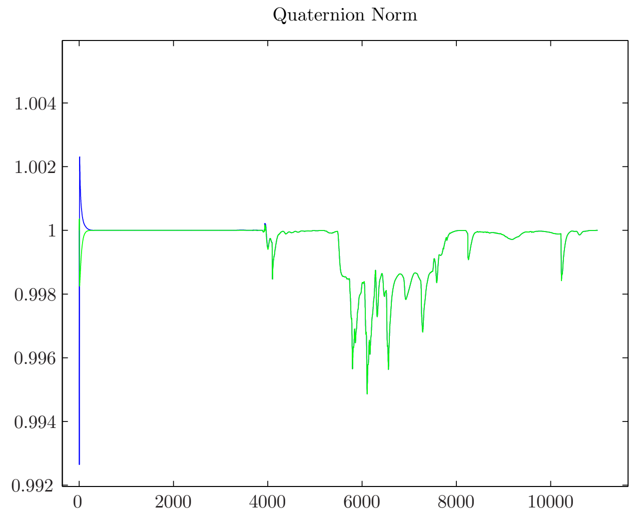
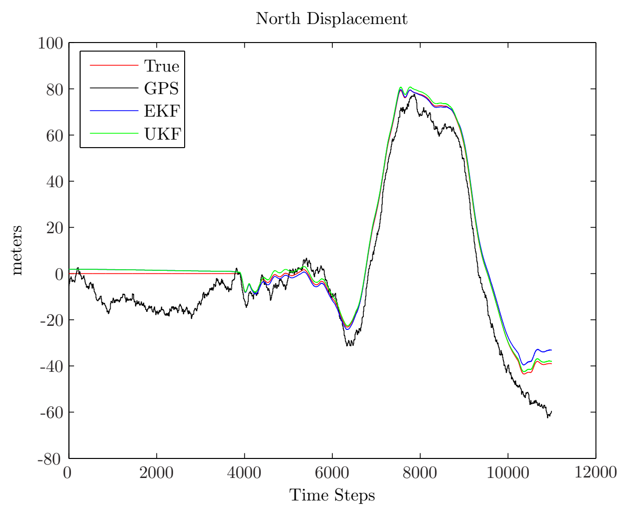
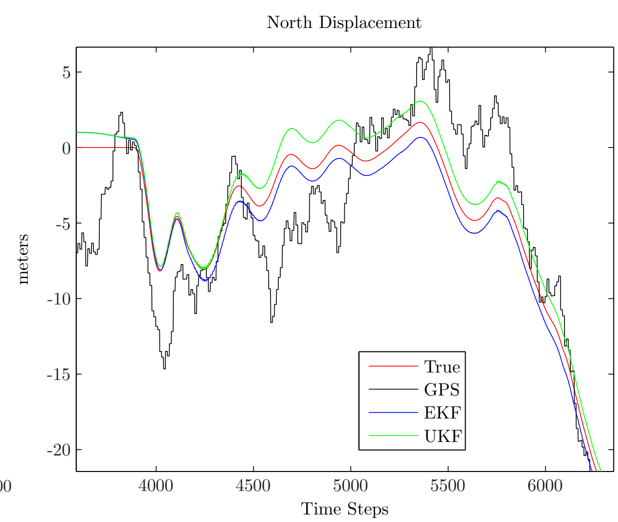


Figure 5: Quaternion norm.

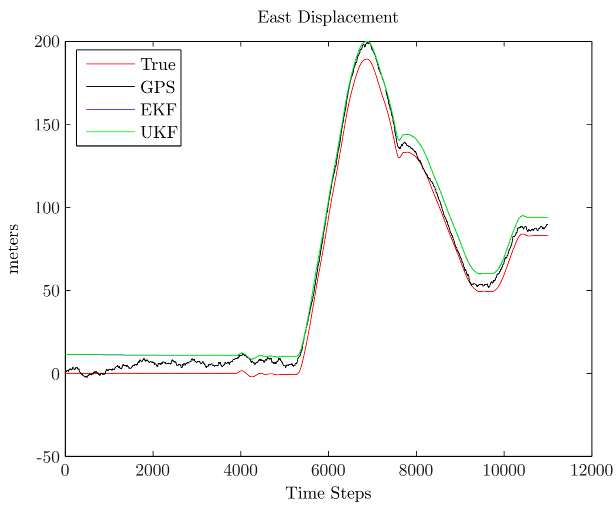


(a) North displacement

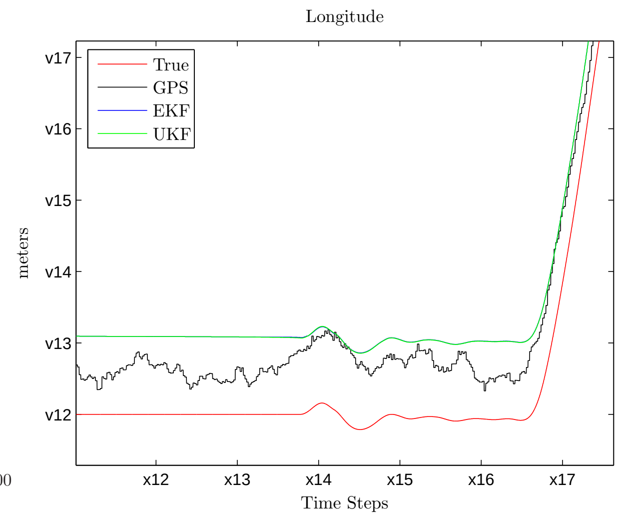


(b) North displacement: particular

Figure 6: EKF and UKF comparison: North displacement estimation.

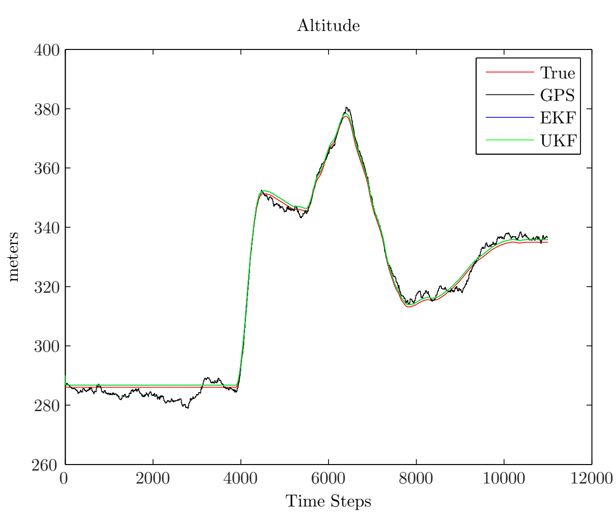


(a) East displacement

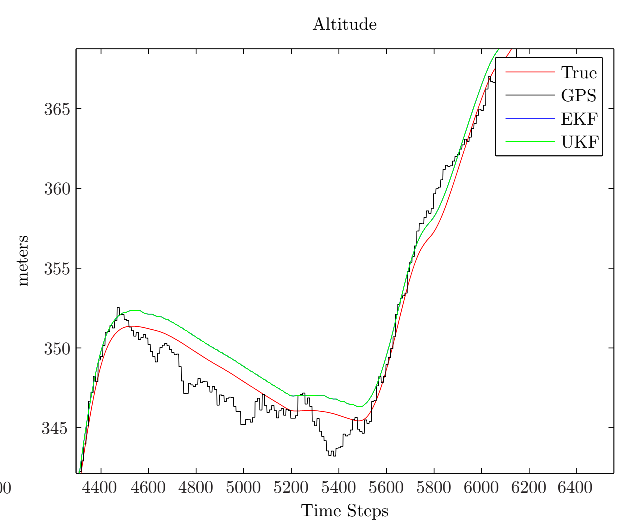


(b) East displacement: particular

Figure 7: EKF and UKF comparison: East displacement estimation.

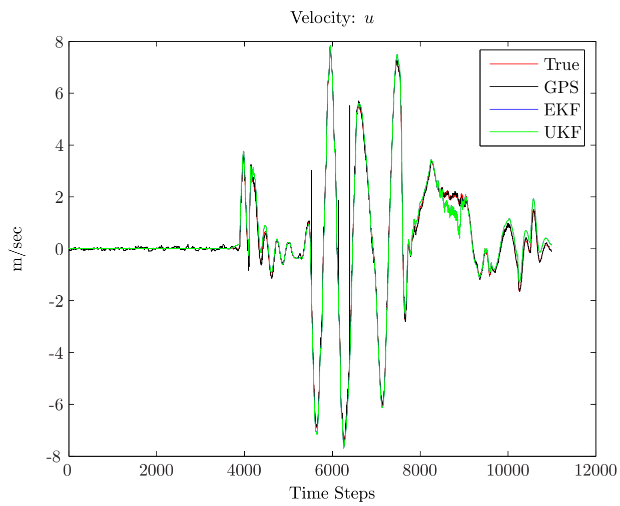


(a) Altitude

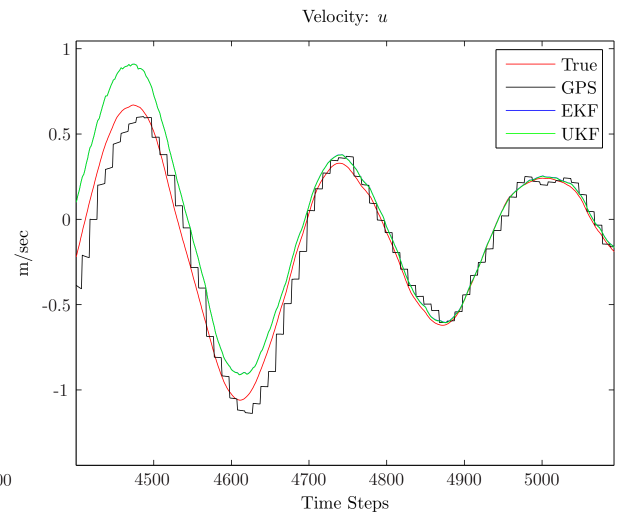


(b) Altitude: particular

Figure 8: EKF and UKF comparison: altitude estimation.

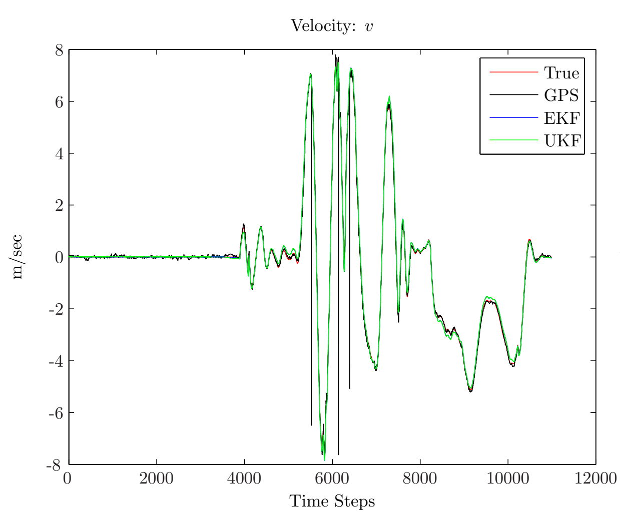


(a) u

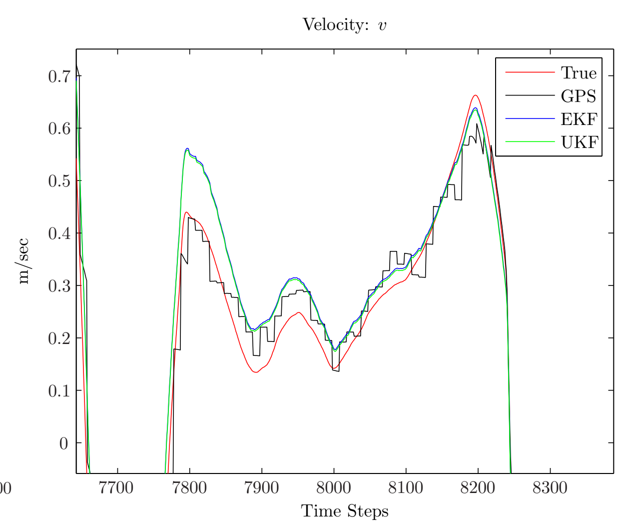


(b) u: particular

Figure 9: EKF and UKF comparison: u estimation.

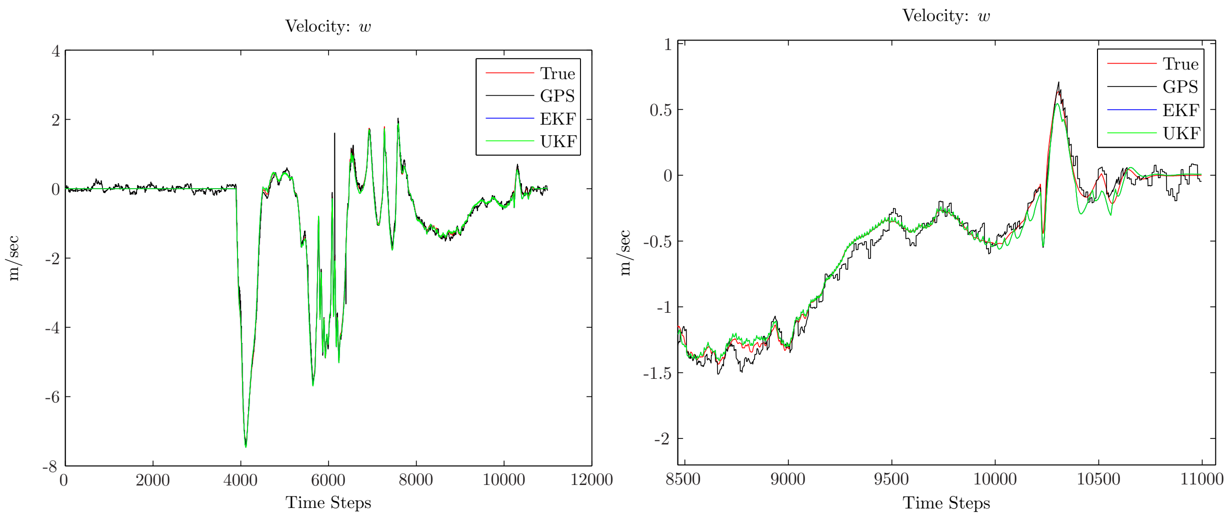


(a) v



(b) v: particular

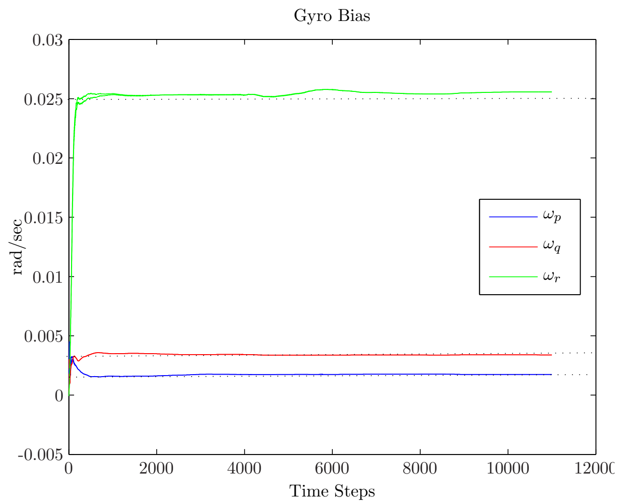
Figure 10: EKF and UKF comparison: v estimation.



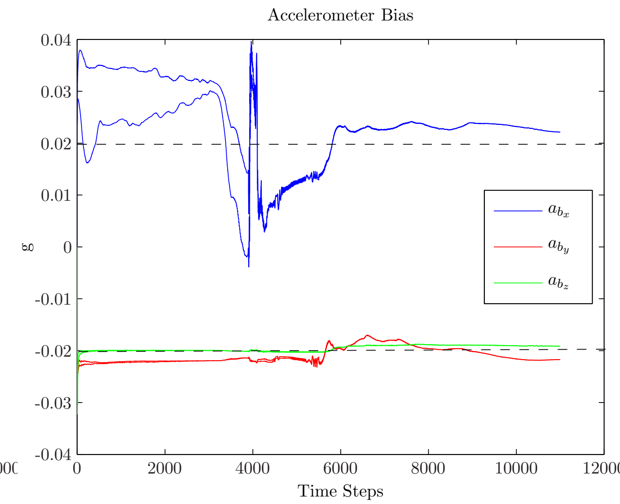
(a) w

(b) w: particular

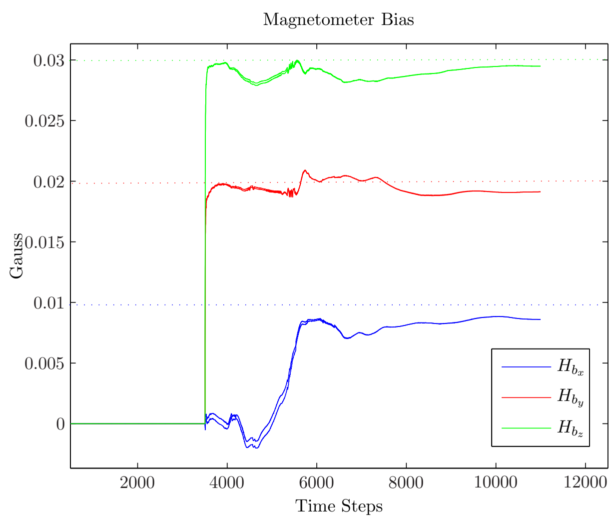
Figure 11: EKF and UKF comparison: w estimation.



(a) Gyros bias



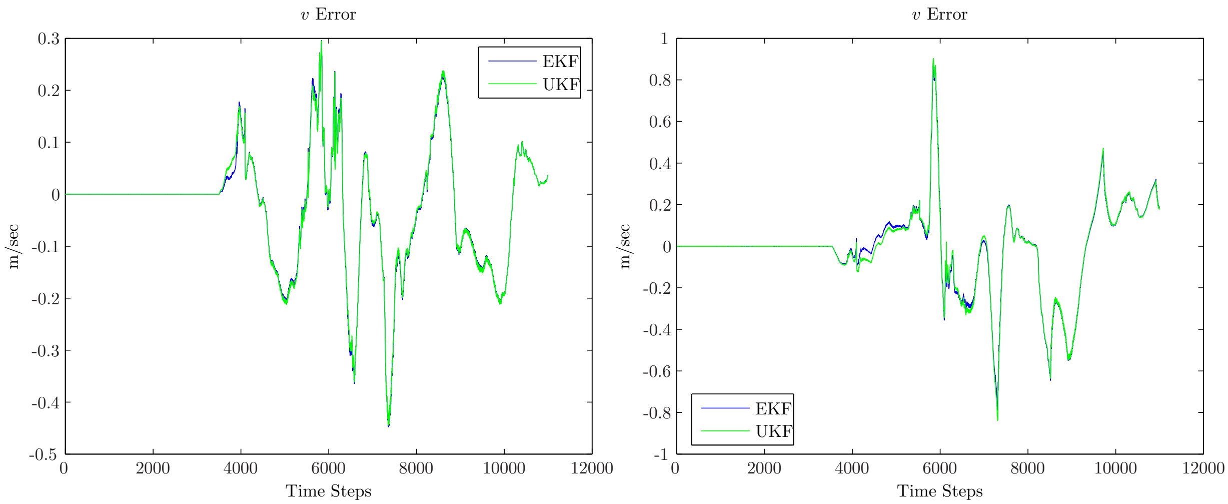
(b) Accelerometers bias



(c) Magnetometer bias

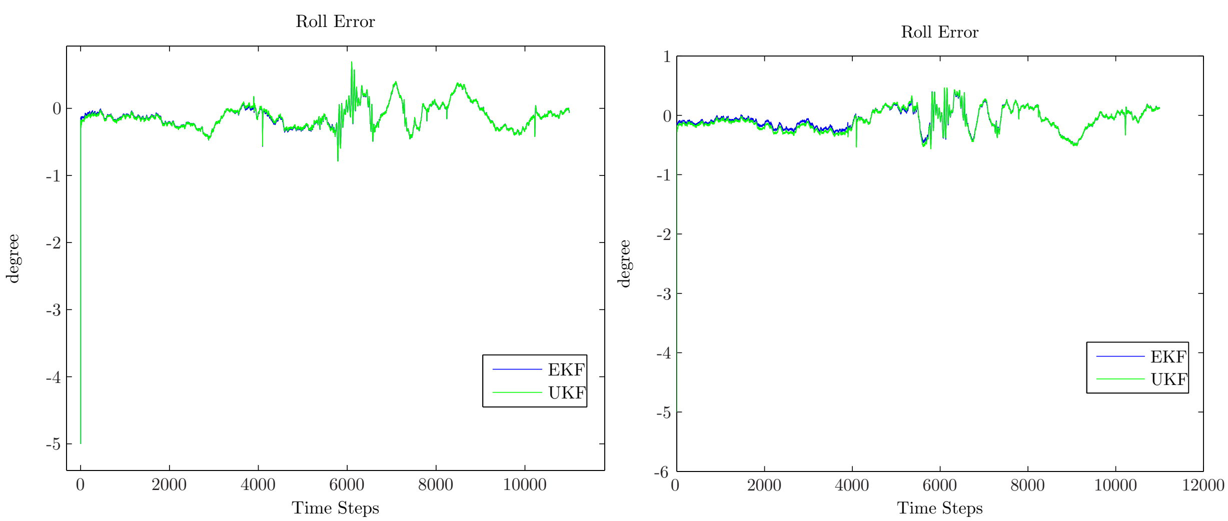
Figure 12: Sensor biases estimation. The true value is represented by the dashed line.

In a real mission could happen to loose the GPS signal. To consider this case we repeated the same simulation showed above but the GPS signal has been switched off for period of forty seconds. The comparison with the previous case is shown in figures 13 and 14. As example the roll and v variables have been considered, but the trends of other variables are similar. The lost of GPS brings to a little decreasing on velocity estimation quality, but it doesn't affect the attitude estimation. This underline the robustness of the developed filters.



(a) (b)

Figure 13: v stimation errors in the with the GPS signal always available (a) and in case of loss of GPS signal (b).

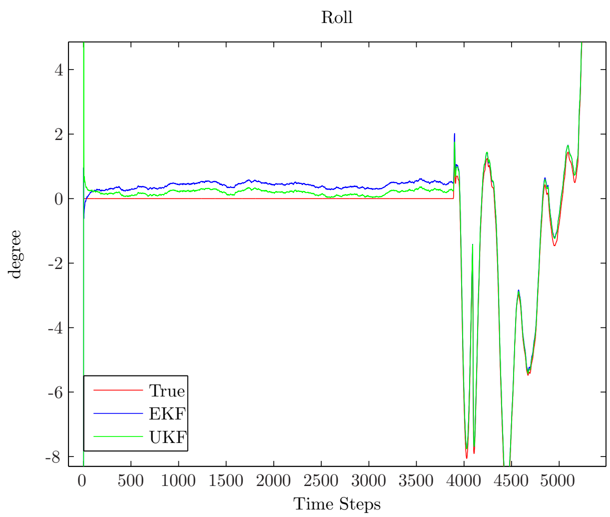


(a) (b)

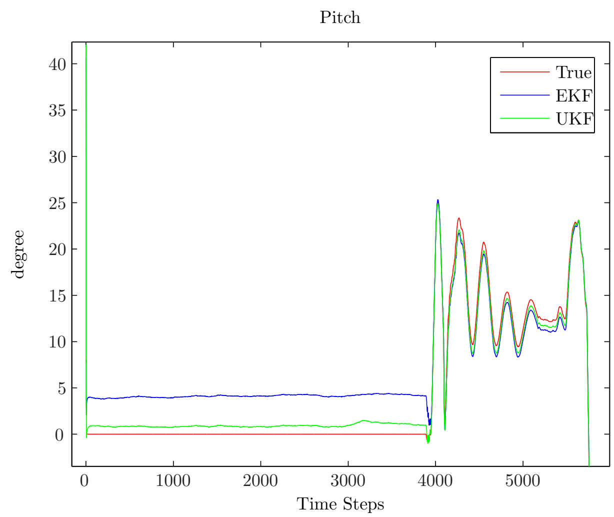
Figure 14: Roll stimation errors in the with the GPS signal always available (a) and in case of loss of GPS signal (b).

The simulations showed that UKF and EKF, independently from loss of GPS, give very similar results while having a small initial estimation error. We now consider the case of a large initial error. In the following simulation the initial attitude error is of about 40 degrees on each angle, and also the initial state covariance is larger (Fig 15). In this case the UKF has a faster convergence with respect to the EKF , but after a settling time the performance become identical.

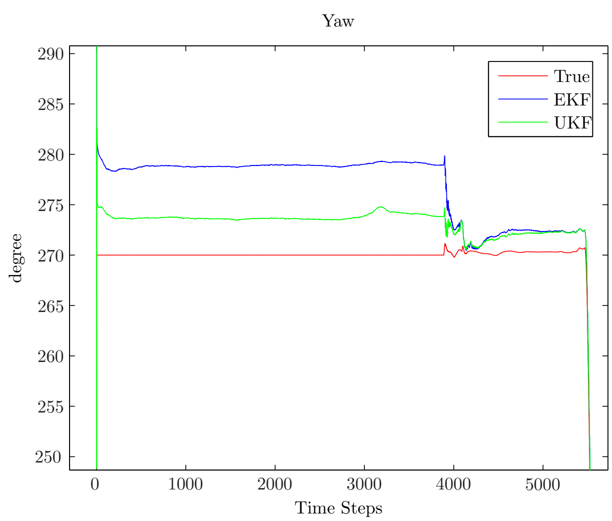
An interesting case is the one which sees a wrong setting of initial covariance. If the initial estimation error is very large, but the initial covariance is not increased according to it, we have the results showed in figure 16. The EKF diverges while the UKF converges. The divergence of attitude angles brings also the divergence of all the other state variables.



(a) Roll

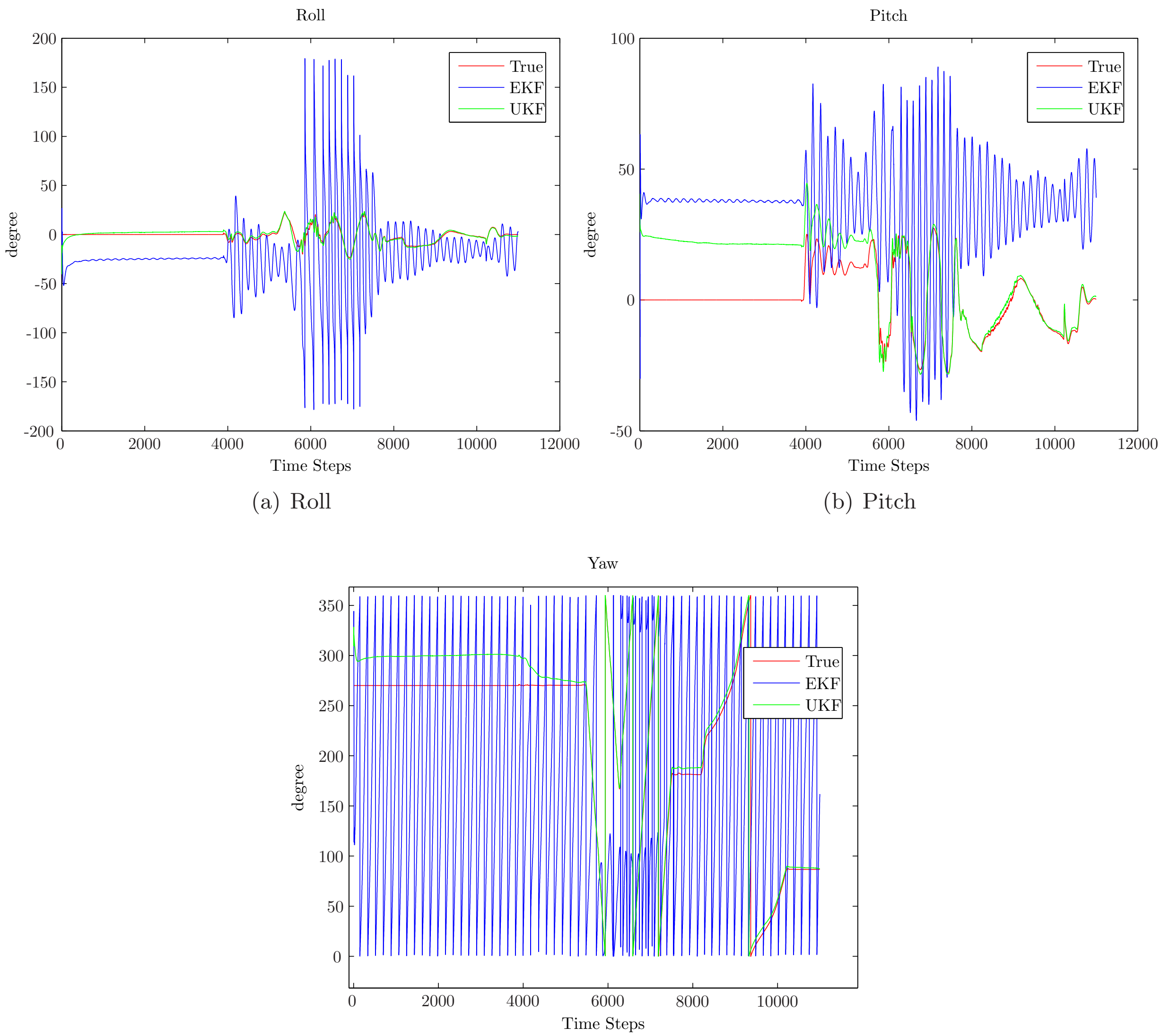


(b) Pitch



(c) Yaw

Figure 15: Attitude estimation in case of large initial error.



(a) Roll

(b) Pitch

(c) Yaw

Figure 16: Attitude estimation in case of large initial error but a small initial covariance.

5. Conclusions

In this report we presented the comparison of EKF and UKF applied to UAVs state estimation. The filters fuse the data of gyros, accelerometers, magnetometers, GPS and barometer, to estimate the attitude, position and velocity of the UAV and the biases of the sensors. The tests have been made simulating real working conditions, including the loss of the GPS signal.

The results show that the two filters give almost identical performance in case of good knowledge of the initial state. On the contrary, if the initial estimation error is large the UKF exhibits a faster convergence if compared with the EKF. Moreover, the UKF appears to be more robust with respect to the choice of the initial parameters of the filter, whereas a wrong choice of the initial estimation error covariance brings the EKF to instability. This difference depends on the nature of the two filters. The EKF obtains the posterior exploiting the linear approximation of the system equations; whereas the UKF, instead of approximating the non-linear equations, approximates the prior by a limited number of points (sigma-points). The posterior is then obtained propagating these points through the original non-linear function. The two approaches give similar results if the initial uncertainty is very close to the mean, so that the sigma-points are very close to each other. In this case the evaluation of the state transition function and of the output function on the sigma points is in accordance with the first order Taylor approximation, and therefore the propagation of the mean and covariance gives the same results given using the Riccati equations, which exploit the Jacobians. If the prior tail is wider around the mean, then the propagation of mean and covariance by means of a linear model, as in the EKF, does not give consistent results. In this case, the sigma-points in the UKF correctly sample a wider region of the state space and provide a more reliable mean and covariance propagation.

In general, the performances of the UKF and of the EKF may differ more or less according to the specific application considered. The main factors that can make the UKF perform better than the EKF are the type of non-linearity and the level of uncertainty that characterize the considered application.

Some considerations must be done about the computational burden of the two filters. In several papers [3] [7] the UKF is presented as less demanding, from a computational point of view, because it doesn't require the computation of Jacobians at each time step. However, in our simulation the SRUKF resulted more time consuming than the EKF, although it has been easier to implement. Similar results have been obtained in [16] [13] [10].

Considering that in a real mission the initial state of the UAV is quite accurately known, and that the computational resources are restricted, the use of the EKF seems more appropriate in the considered application.

References

[1] A. Germani, C. Manes, P. Palumbo, "Polynomial Extended Kalman Filter", IEEE Trans. on Automatic Control, Vol. 50, No. 12, pp. 2059- 2064, 2005.

[2] M. Dalla Mora, A. Germani, C. Manes *Introduzione alla Teoria dell'Identificazione dei Sistemi*. EURoma-La Goliardica, 1997.

[3] S. Julier and J. Uhlmann, "A new approch for filtering nonlinear systems," in *Proceedings of the American Control Conference*, 1995.

[4] S. Julier and J. Uhlmann, "A general method for approximating nonlinear transformations of probability distributions," 1996.

[5] S. Julier and J. Uhlmann, "A new extension of the kalman filter to nonlinear systems," in *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*, 1997.

[6] S. Julier and J. Uhlmann, "The scaled unscented transformation.," in *Proceedings of the IEEE American Control Conference*, 2002.

[7] R. van der Merwe, *Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*. PhD thesis, Oregon Health and Science University, 2004.

[8] R. van der Merwe and E. Wan, "The square-root unscented kalman filter for state and parameter-estimation," 2001.

[9] P. Peliti, “Tecniche di filtraggio applicate alla stima dello stato di un UAV dotato di INS e GPS,” Master’s thesis, Sapienza Università di Roma, 2008.

[10] S. Anavatti, A. Kallapur and M. Garratt, “Extended and unscented kalman filters for attitude estimation of an unmanned aerial vehicle,” in *Proceedings of Modelling, Identification, and Control*, 2008.

[11] F. Lemoine, S. Kenyon, J. Factor, R. Trimmer, N. Pavlis, D. Chinn, C. Cox, S. Klosko, S. Luthcke, M. Torrence, Y. Wang, R. Williamson, E. Pavlis, R. Rapp and T. Olson, “The development of the joint NASA gsfc and nima geopotential model EGM96,” tech. rep., NASA Goddard Space Flight Center, Greenbelt, Maryland, 20771 USA,, July 1998.

[12] V. Gavrilets, *Autonomous Acrobatic Maneuvring of Miniature Helicopters: Modeling and Control*. PhD thesis, Massachussets Institute of Technology, 2003.

[13] D. G. Greer, T. S. Bruggemann, and R. A. Walker, “Sigma point Kalman filters for GPS navigation with integrity in aviation,” in *International Global Navigation Satellite Systems Society Symposium*, 2007.

[14] M. S. Grewal, L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration 2nd Edition*. Wiley-Interscience, 2007.

[15] J. M. Rolfe and K. J. Staples, *Flight Simulation*. Cambridge University Press.

[16] Wendel J., Metzger J., Moenikes R., Maier A. and Trommer G., “A performance comparison of tightly coupled GPS/INS navigation systems based on extended and sigma point kalman filters,” in *Journal of The Institute of Navigation*, 2006.