https://doi.org/10.3311/PPee.20361

Periodica Polytechnica Electrical Engineering and Computer Science, 66(3), pp. 215–226, 2022

Comparison of Adaptive Fuzzy EKF and Adaptive Fuzzy UKF for State Estimation of UAVs Using Sensor Fusion

Huda Naji Al-sudany[1*], Béla Lantos[1]

[1] Department of Control Engineering and Information Technology, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, H-1521 Budapest, P.O.B. 91, Hungary
[*] Corresponding author, e-mail: alsudany@iit.bme.hu

Received: 22 April 2021, Accepted: 27 May 2022, Published online: 30 June 2022

Abstract

Development of an Adaptive Fuzzy Extended Kalman Filter (AFEKF) and an Adaptive Fuzzy Unscented Kalman Filter (AFUKF) for the state estimation of unmanned aerial vehicles (UAVs) were presented in this paper based on real flight data of a fixed wing airplane. The Adaptive Neuro Fuzzy extension helps to estimate the values of the EKF's and UKF's R_k covariance matrix at each sampling instant when measurement update step is carried out. The ANFIS monitors the EKF's and UKF's performances attempt to eliminate the gap between theoretical and real innovation sequences' covariance. The investigations show that AFUKF can provide better performance in accuracy and less error than the AFEKF in case of real flight data for maneuvering fixed wing UAVs.

Keywords

Extended Kalman Filter (EKF), Fuzzy Inference System (FIS), Unscented Kalman filter (UKF), Adaptive Neuro-Fuzzy System (ANFIS)

1 Introduction

Estimating the state of an unmanned aerial vehicle (UAV) is an important intermediate step in modeling and control. Parameter and state estimation is essential for the design and detection of missiles, aircraft, and shells, and also for a range of related applications including object dynamics simulations and defect detection and diagnostics.

The classic Kalman filter formulation involves prior information of the process and measurement noise statistics [1, 2], as well as Q and R matrices. This paper concentrates on the state estimation using sensor fusion. The knowledge of the popular approaches EKF and UKF is assumed on a basic level typical in the control practice. Only their main notations will be repeated here necessary to understand the extensions enhanced by fuzzy inference system and adaptively in sensor signal filtering and state estimation.

The attitude (orientation) could be represented in various forms, like quaternions, Euler angles, Direction Cosine Matrix (DCM), The sensory information can be contained in the form of Accelerometers, Rate gyroscope, Magnetometers and GPS sensors. The main idea behind this is a fuzzy inference system that adapts R measurement covariance matrix filtering and estimating it to prevent the filter from diverging during the non-stationary flight phases. The tuning procedure is dependent in relation to the concept of Degree of Matching, which ought to trend to zero whatever of flight phase. The main fuzzy (EKF&UKF) outcomes could be summarized as follows [3]:

1. Estimation of unknown measurement noise covariance matrix.
2. Develop estimation precision by using IMU and magnetometer data (without GPS data).
3. Improve accuracy of estimation using outputs of IMU and magnetometer (with data GPS).
4. Accuracy of Gyro bias estimate has been improved.

By detecting gravity acceleration under statically or quasi-statically condition, the accelerometer helps estimate the body's pitch roll and yaw. Under the circumstance of no magnetic disturbance, the magnetometer can detect the orientation of the body by measuring the geomagnetic field based on the pitch and roll information supplied by the accelerometer. GPS navigation is performed relative to ECEF frame while the IMU (3D acceleration and 3D angular velocity) sensors measure relative to ECI frame.

The structure of the paper is as follows. Section 2 and Section 3 summarize the fundamental formulas of EKF and UKF, respectively. Section 4 formulates the concept of adaptive tuning. Section 5 describes the details of the

Cite this article as: Al-sudany, H. N., Lantos, B. "Comparison of Adaptive Fuzzy EKF and Adaptive Fuzzy UKF for State Estimation of UAVs Using Sensor Fusion", Periodica Polytechnica Electrical Engineering and Computer Science, 66(3), pp. 215–226, 2022. https://doi.org/10.3311/PPee.20361

FIS system and the control configuration. The derivation of AFEKF and AFUKF based on the used innovation technique is discussed in Section 6 together with the algorithm of covariance correction. Section 7 describes the implementation concept of AFEKF and AFUKF together with the degree of matching (DOM) of the input signals, the Gaussian membership functions, and the fuzzy rules for covariance matrix tuning. Section 8 contains the comparison of AFUKF and AFUKF for real flight data of a fixed wing UAV. The paper is finished with the Conclusions, Appendix, and References. The Appendix summarizes the fundamentals of navigation systems based on IMU, Magnetometer and GPS sensors used in the developed software. It is suggested to read Appendix first if the reader is not familiar with WGS-84 GPS standard and the state variables in the kinematic model of vehicles.

2 Extended Kalman Filter

An Extended Kalman Filter, or EKF, is a natural extension of the time variant Linear Kalman Filter (LKF) that linearize current mean and covariance. Taylor series, which uses the partial derivatives of the process and measurement functions to produce estimates in the presence of nonlinear interactions is the basis of EKF. It can estimate past, present, and possible future (predicted) states even if the exact details of the represented system are unclear. Equations (1) and (2) actually produce generic nonlinear system and measurement form [1]:

$$x_{k+1} = f\left(x_k, u_k, w_k\right) \tag{1}$$

$$z_k = h_k\left(x_k, v_k\right) \tag{2}$$

Extended Kalman Filter forecasts the system's future state [4]. There are two main groups of equations in the Extended Kalman Filter method: the prediction equations and the measurement updates equations:

1. Time updates equations:

$$\hat{x}_{k+1}^{-} = f\left(\hat{x}_k, u_k, 0\right) \tag{3}$$

$$P_{k+1}^{-} = \left(\nabla_x f_k\right) P_k \left(\nabla_x f_k\right)^T + \left(\nabla_w f_k\right) Q_k \left(\nabla_w f_k\right)^T \tag{4}$$

2. Measurement updates equations:

$$K_k = P_k^{-} H_k^T \left(H_k P_k^{-} H_k^T + R_k\right)^{-1} \tag{5}$$

$$\hat{x}_k = \hat{x}_k^{-} + K_k\left(z_k - h\left(\hat{x}_k^{-}\right)\right) \tag{6}$$

$$P_k = \left(I - K_k H_k\right) P_k^{-} \tag{7}$$

where I is the identity matrix and $H_k = \partial h_k / \partial x$.

3 UKF Algorithm for Nonlinear State Estimation

UKF operating principle is unscented transformation (UT) that is a technique for computing the statistics of a random variable that has undergone a nonlinear transformation. With the aid of UT, the sigma points are found. The distribution of sigma points around a particular state is determined by this parameter [5–7]. We use the notations in [5] but instead of χ and γ stand X and Z with subscripts, and $f_k(x_k, v_k)$ with index identifies also u_k.

Select sigma point

Denote the mean of the initial state $\bar{x}(0) = \hat{x}_{0/0}$ and let the augmented state and covariance matrix be respectively $\hat{x}_{k/k}^a = \left(\hat{x}_{k\backslash k}^T, \bar{w}_{k\backslash k}^T, \bar{v}_{k\backslash k}^T\right)^T$ and $P_{k\backslash k}^a = \text{blkdiag}\left(P_{k\backslash k}, R_{w,k\backslash k}, R_{v,k\backslash k}\right)$, and define:

$$X_0 = \hat{x}_{k/k} \tag{8}$$

$$X_{i,k} = \hat{x}_{k/k} + \left(\sqrt{(N+\lambda) P_{k/k}}\right)_i, \quad i = 1, \ldots., N \tag{9}$$

$$X_{i,k} = \hat{x}_{k/k} - \left(\sqrt{(N+\lambda) P_{k/k}}\right)_i, \quad i = N+1, \ldots., 2N \tag{10}$$

where $\left(\sqrt{A}\right)_i$ is the i-th column of the matrix square root.

Sigma Points' Weights:

$$W_0^{(m)} = \lambda / (N + \lambda) \tag{11}$$

$$W_0^{(c)} = \lambda / (N + \lambda) + \left(1 - \alpha^2 + \beta\right) \tag{12}$$

The Scaled Unscented Transformation's Parameters:
$\lambda = \alpha^2(N + k) - N$, where N is the state dimension;
$\beta : \beta = 2$ for Gaussian distributions.

Algorithm of UKF

Model of State Space:

$$x_{k+1} = f_k\left(x_k\right) + w_k \tag{13}$$

$$z_k = h_k\left(x_k\right) + v_k \tag{14}$$

Filter initialization

$$\hat{x} = E\left(x_0\right) \tag{15}$$

$$P_0 = E\left(\left(x_o - \hat{x}_o\right)\left(x_o - \hat{x}_o\right)^T\right) \tag{16}$$

Compute the sigma points

$$X_k = \left(\hat{x}_k, \hat{x}_k + \gamma\sqrt{P_k}, \hat{x}_k - \gamma\sqrt{P_k}\right) \tag{17}$$

Time Update

$$\hat{X}_{i,k+1/k} = f_k\left(X_{i,k}\right) \tag{18}$$

$$\hat{x}_{k+1/k} = \sum_{i=0}^{2N} W_i^{(m)}\left(\hat{X}_{i,k+1/k}\right) \tag{19}$$

$$P_{k+1/k}=\sum_{i=0}^{2N}W_i^{(c)}\left(\hat{X}_{i,k+1/k}-\hat{x}_{k+1/k}\right)\left(\hat{X}_{i,k+1/k}-\hat{x}_{k+1/k}\right)^T+Q_k \quad (20)$$

Measurement Prediction

$$Z_{i,k+1/k}=h_k\left(X_{i,k+1/k}\right) \quad (21)$$

$$\hat{z}_{k+1}=\sum_{i=0}^{2N}W_i^{(m)}Z_{i,k+1/k} \quad (22)$$

Kalman Gain

$$P_{z_{k+1}z_{k+1}}=\sum_{i=o}^{2N}W_i^{(c)}\left(Z_{i,k+1/k}-\hat{z}_{k+1}\right)\left(Z_{i,k+1/k}-\hat{z}_{k+1}\right)^T+R_k \quad (23)$$

$$P_{x_{k+1}z_{k+1}}=\sum_{i=o}^{2N}W_i^{(c)}\left(X_{i,k+1/k}-\hat{x}_{k+1}\right)\left(Z_{i,k+1/k}-\hat{z}_{k+1}\right)^T \quad (24)$$

$$K_{k+1}=P_{x_{k+1}y_{k+1}}P^{-1}_{y_{k+1}y_{k+1}} \quad (25)$$

Measurement update

$$\hat{x}_{k+1/k+1}=\hat{x}_{k+1/k}+K_{k+1}\left(z_{k+1}-\hat{z}_{k+1}\right) \quad (26)$$

$$P_{k+1/k+1}=P_{k+1/k}-K_{k+1}P_{z_{k+1}z_{k+1}}K_{k+1}^{\ T} \quad (27)$$

4 Adaptive Fuzzy (AFEKF&AFUKF) Filter

Suggested filter consists of the primary filters EKF & UKF and a secondary block (adaptation component), where the portion of adaptation is FIS system. AFEKF & AFUKF could be seen in Fig. 1. Degree of matching (DOM) of fuzzy inference system is calculated using updated measurements vector and output UKF, EKF at each AFEKF & AFUKF iteration. The suggested filter made up of the primary filter and FIS block (part of adaptation) where FIS system is the adaptation part, see Fig. 1.

There are two matrices needed for calculation illustrated in Table 1.

5 Fuzzy Inference System (FIS)

Fuzzy inference systems, sometimes called fuzzy-rule-based systems, use fuzzy rules to make decisions or when

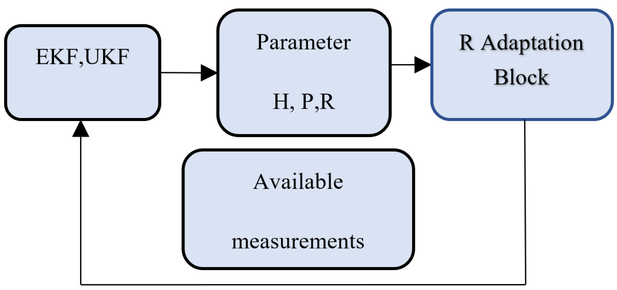

Fig. 1 Structure of our improvement block

Table 1 Covariance matrices needed in adaptation

State error covariance	$P_k=E\left\{\left(x_k-\hat{x}_k\right)\left(x_k-\hat{x}_k\right)^T\right\}$
Diagonal covariance matrix	R_k
	H_k the linearization of the measurements nonlinear vector function

it employed as controllers called fuzzy controllers or fuzzy models.

System with fuzzy rules (FIS) is a fuzzy logic based computing framework principles, fuzzy inference system offers systematic approach that describing operating and executing a heuristic used by humans understanding of how to operate a system [8]. FIS fundamental structure consists of five functional elements, the functions of which are as follows:

- Fuzzy rules: a set of IF-THEN rules.
- Membership functions: is a set of curves that define how each point in the input space gets a membership value (or degree of membership) ranging from 0 to 1.
- Decision-making unit that uses rules to execute inference operations.
- Fuzzification interface: transforms incoming data into appropriate linguistic values.
- Defuzzification interface: as the name indicates, is the inverse of fuzzification, converts the inference's fuzzy outcomes into a crisp output [9]. The process of adaption is carried out on the statistical filter R matrix of data, that is modified by using a fuzzy inference system (FIS) which is focused on a filter innovation sequence utilizing a covariance matching strategy.

The aerodynamics propulsion forces are also part of the flight data under real control but not used here in the paper. They include the contributions due to the control deflections that is, the elevator, aileron and rudder denoted by δa, δe and δr respectively for a standard aircraft. Their values together with the identified states are the basis for the future dynamic model identification of maneuvering vehicles.

6 Derivation of AFEKF & AFUKF

Describe the structure of innovation or (adaptive estimation based on innovation, *Inn*) as equation below [10]:

$$Inn_k=\delta z_k^-=\tilde{z}_k-h\left(\hat{x}_k^-\right) \quad (28)$$

It is known as the difference between measurement vector and it's a priori estimate, Inn_k technique is focused on improving existing filter performance by adaptive estimate of Q and/or R filter matrices (in this article only R). Inn_k indicates extra information that the filter can use. The occurrence of faulty data initially presents itself in Inn_k. The disparity between projected and actual measurements is described in this way. Compute the covariance S_k of the result of the sum of two stochastic procedures:

$$Inn_k = \delta z_k^- = H_k \delta x_{r,k} + V_k$$

It will be assumed that both the state error and the measurement error are white noises having zero mean and independent each to other in the sense that they are unrelated to a priori calculation of the mixed effects (covariance), this then refers to [10]:

$$S_k = \operatorname{cov}\left(\delta z_k^-\right) = \operatorname{cov}\left(H_k \delta x_{r,k}\right) + \operatorname{cov}\left(V_k\right) \tag{29}$$

$$S_k = H_k \operatorname{cov}\left(\delta x_{r,k}\right)\left(H_k\right)^T + R_k \tag{30}$$

$$S_k = H_k P_k^- H_k^T + R_k \tag{31}$$

with:

$$\operatorname{cov}(\lambda) = E\left[(\lambda - E(\lambda)) \cdot (\lambda - E(\lambda))^T\right] \tag{32}$$

S_k can be analytically computed in real-time processing using Eq. (30). It is going to be known as C_{Inn_k}.

$$C_{Inn_k} = \frac{1}{M} \sum_{i=k-M+1}^{k} \left(\tilde{z}_i - h\left(\hat{x}_i^-\right)\right)^T \tag{33}$$

$$C_{Inn_k} \approx \operatorname{cov}\left(\delta z_k^-\right) \tag{34}$$

Because the δz_k^- process is considered to be ergodic, probabilistic and arithmetic covariance are equivalent. The major applications of arithmetic covariance are to identify quick changes in variance. Because maneuvering is too quick, arithmetic covariance is estimated inside of the window (with time duration M), and an appropriate M value and (FIS) membership function variables will be described below. M ought to be less than duration of maneuvering. Employing a large period M will not detect maneuvers. Due to this, the matrix of covariance of measurement error estimation is represented on the left side of previously, so we define:

$$DoM = C_{Inn_k} - S_k \tag{35}$$

The expression DoM represents degree of matching among two values $\left(C_{Inn_k},\ S_k\right)$ and theoretically it should be zero. As a result, the adaptation is predicated on continually setting a factor to maintain this term as near to zero as feasible. As Eq. (34), it is well acknowledged that measurement covariance of error is immediately appeared in the DoM value.

$$DoM = C_{Inn_k} - \left(H_k P_k^- H_k^T + R_k\right) \tag{36}$$

At each iteration R_k is adjusted to remain as near to zero as possible.

The algorithm of the fuzzy correction of the covariance matrix is as follows:

$$R_k = R_{k-1}^+ \tag{37}$$

$$\rightarrow DoM = C_{Inn_k} - \left(H_k P_k^- H_k^T + R_k\right) \tag{38}$$

$$\rightarrow DoM \boxed{FIS} \rightarrow \Delta R_k \tag{39}$$

$$\rightarrow R_k^+ = R_k + \Delta R_k \tag{40}$$

Here R_k is the adjusted measurement covariance matrix at the k-th sample. R_k is continually calculated using a fuzzy inference approach. FIS receives DOM as an input, which is computed during each iteration of the Kalman filter [11, 3].

7 Implementation concept of AFEKF and AFUKF

Using MATLAB, numeric computations were done to compare the results of the AFEKF method to AFUKF using all flight data sensors (IMU, Magnetometer and GPS). For both degree of matching and , FIS includes 3 sets (N: Negative, P: Positive, Z: Zero). For both degree of matching and ΔR_k, mean and sigma with Gaussian membership functions are used to represent membership functions and the rules. In order to simplify notations, only three membership functions (P, Z, N) will be considered here, however, the method can easily be generalized for more than three (PL, PM, Z, NM, NL etc.) membership functions:

1. if (Degree of matching is N) then ΔR_k is P),
2. if (Degree of matching is Z) then ΔR_k is Z),
3. if (Degree of matching is P) then ΔR_k is N).

FIS parameters, like sigma and mean of Gaussian membership functions are selected carefully depending on the essential requirements:

- Considering (R_k) relationship between the input degree of matching (DOM) and output (R_k).
- Input and output parameters range.
- Aspect of adjustment: additive.

We have the measurement in the body frame, mapping to n- frame (NED), we get:

$$z_1 = (o, o, -g)^T$$

$$z_2 = Mag_{angle}(H_x, H_y, H_z)^T$$

The value of z_2 can be determined by the MATLAB function **wrldmagm.** Denote R_n^b the orientation transformation from NED to BODY then:

$$Inn_k = z_k - R_n^b (z_1^T, z_2^T)^T = R_b^n z_k - (z_1^T, z_2^T)^T$$

where $R_n^b = (R_b^n)^{-1} = (R_b^n)^T$.

Magnetometer measurement might be used as row measurements, but they are converted to another term in the code, it is a declination angle in Budapest. FIS has three sets (N: Negative, P: Positive, Z: Zero) for both degree of matching and ΔR, see Figs. 2–4.

Window size is M = 5; and the first test is shown in Fig. 5, which depicts the path of this simulated trip. The whole above results are achieved in the case of gyros output; these measurements in Fig. 5 reach 100 degree per seconds in some periods; the whole methods (UKF&EKF with fuzzy).

8 Comparison of AFUKF and AFEKF for maneuvering UAVs based on real flight data

Steps might be considered to improve results:
has the ability to get useful data and predict some variables

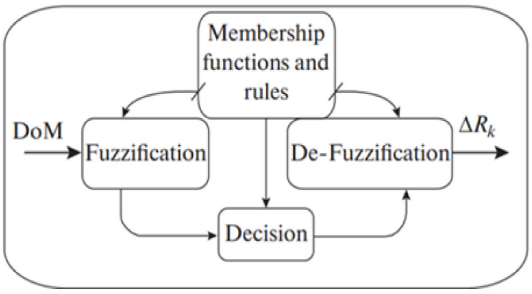

Fig. 2 FIS for estimate R_k

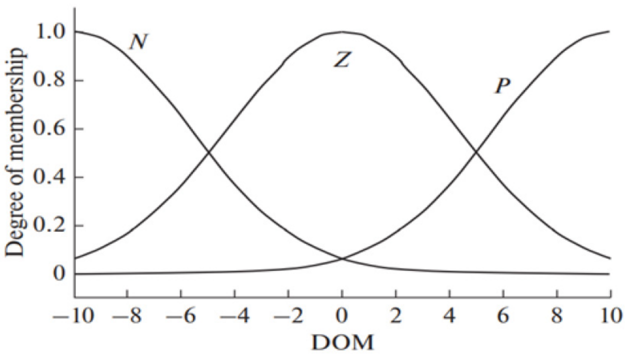

Fig. 3 FIS for DoM

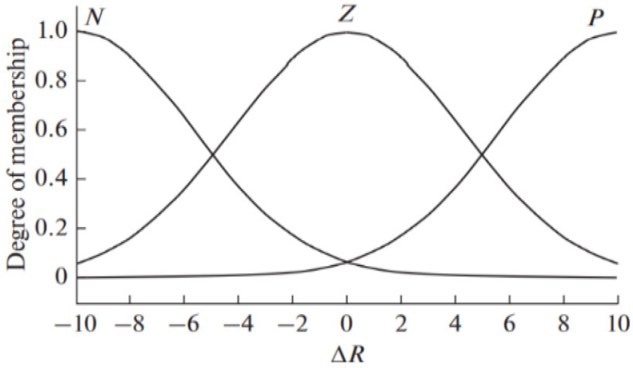

Fig. 4 FIS for ΔR

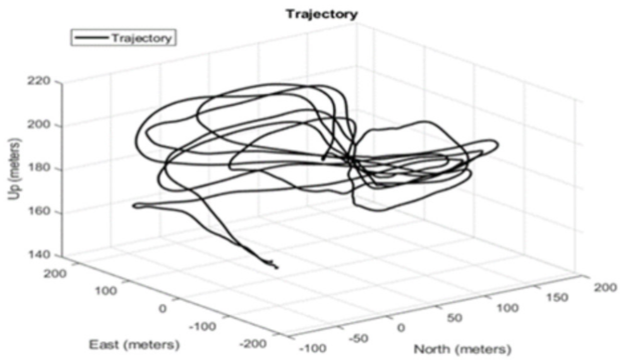

Fig. 5 Trajectory of simulated trip (3D)

- Changing fuzzy sets in Fig. 2 and Fig. 3.
- Changing the range of FIS input and FIS output, the input is the error or the term $\boldsymbol{DoM}$ this term should be watched to understand the range of changes of this term. The output range should be in the same grade of Matrix R.
- Changing the window size of the term $\boldsymbol{Inn}$.

The calculations of variance, and standard deviation values are executed in three coordinates (X, Y, Z) according to:

$$\bar{x} = (x_{i-M} + x_{i-M+1} + \ldots + x_i) / M \tag{41}$$

$$d = x_i - \bar{x} \tag{42}$$

$$\sigma = \sqrt{\sum |d|^2 / M} \tag{43}$$

$$v = \sigma^2 \tag{44}$$

- This process is implemented similarly for EKF: EKF1 (no GPS) and EKF2 (GPS is present) and for UKF: UKF1 (no GPS) and UKF2 (GPS is present).

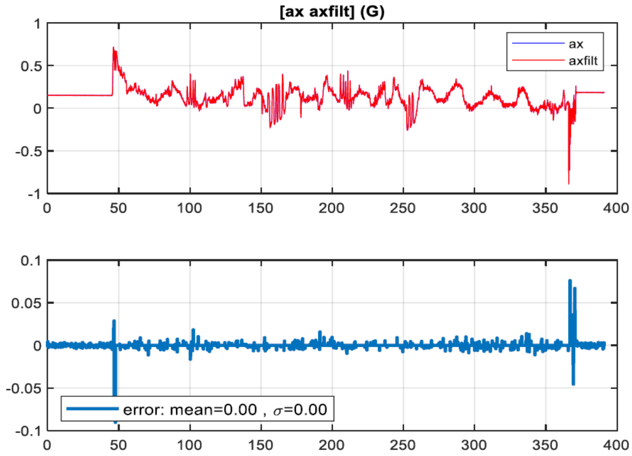

Fig. 6 (IMU acceleration x) true vs predicted (AFUKF)

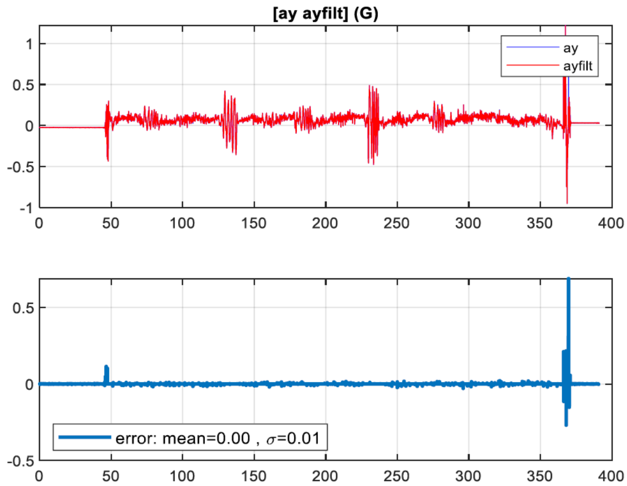

Fig. 7 (IMU acceleration y) true vs predicted (AFUKF)

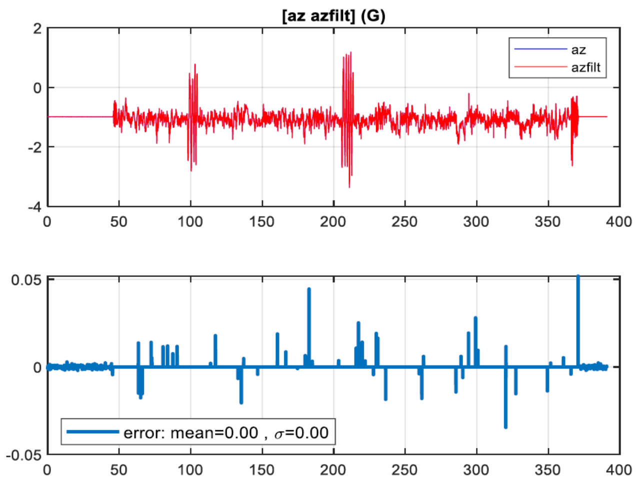

Fig. 8 (acceleration z) true vs predicted (AFUKF)

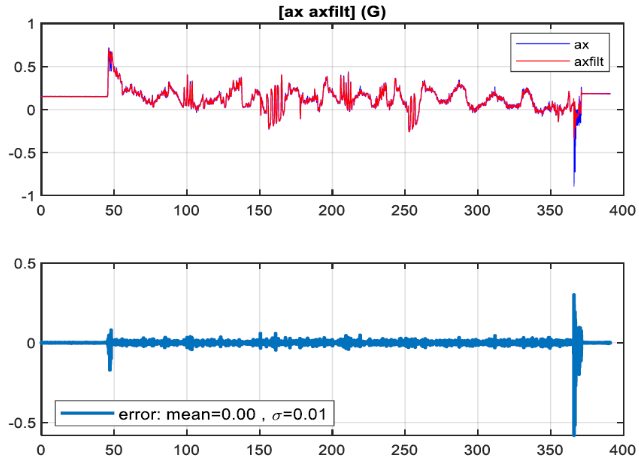

Fig. 9 (IMU acceleration x) true vs predicted (AFEKF)

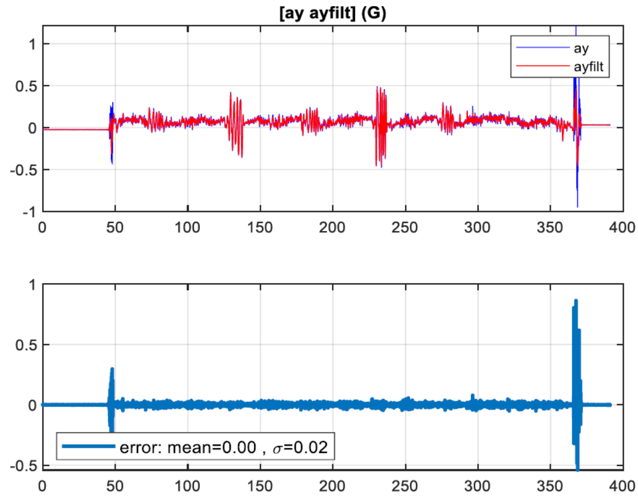

Fig. 10 (IMU acceleration y) true vs predicted (AFEKF)

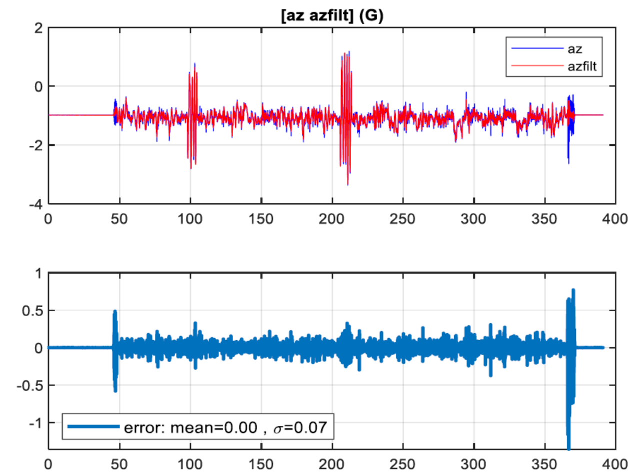

Fig. 11 (acceleration z) true vs predicted (AFEKF)

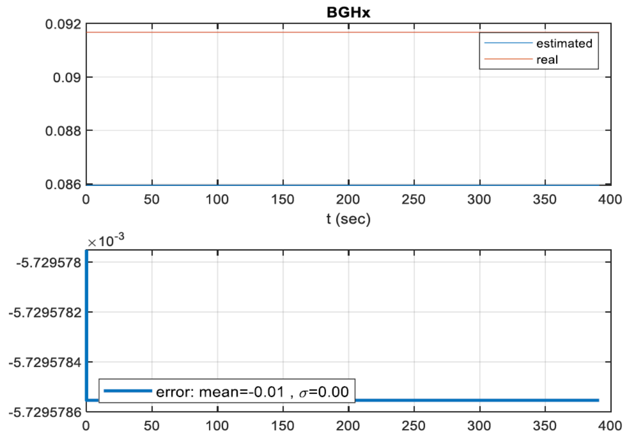

Fig. 12 Gyro bias (x)-axis true vs predicted (AFUKF)

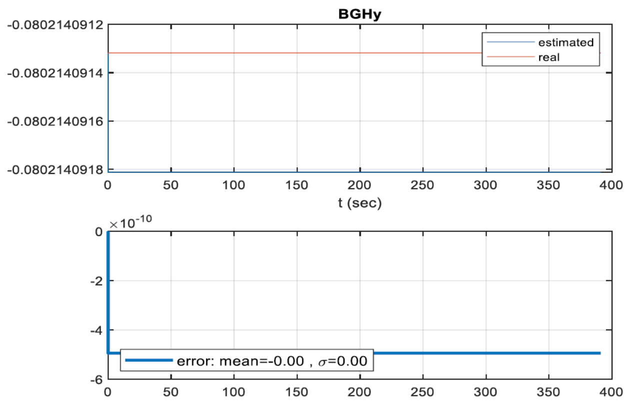

Fig. 13 Gyro bias (y)-axis true vs predicted (AFUKF)

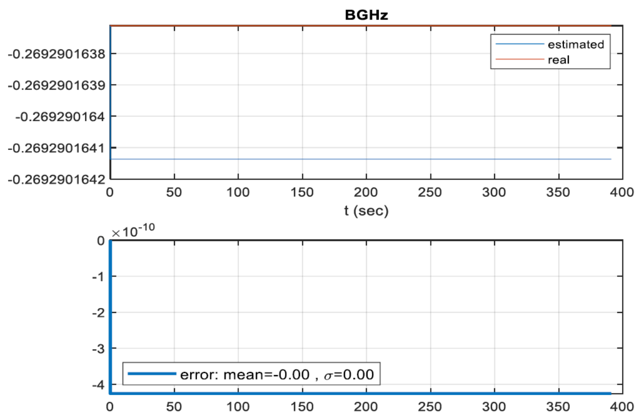

Fig. 14 Gyro bias (z)-axis true vs predicted (AFUKF)

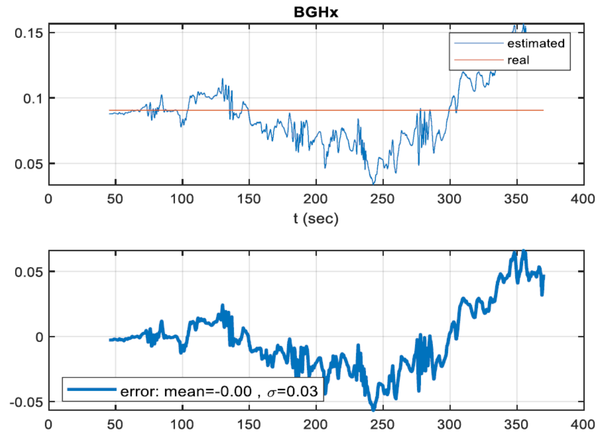

Fig. 15 Gyro bias (x)-axis true vs predicted (AFEKF)

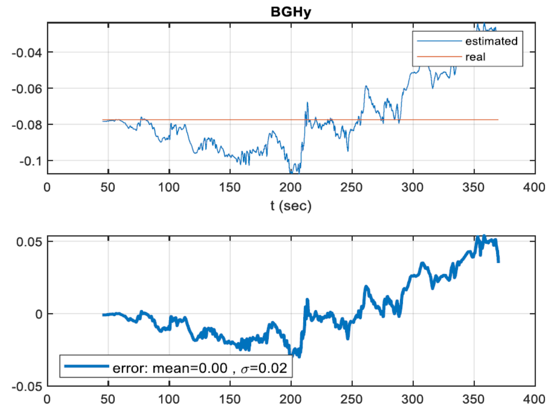

Fig. 16 Gyro bias (y)-axis true vs predicted (AFEKF)

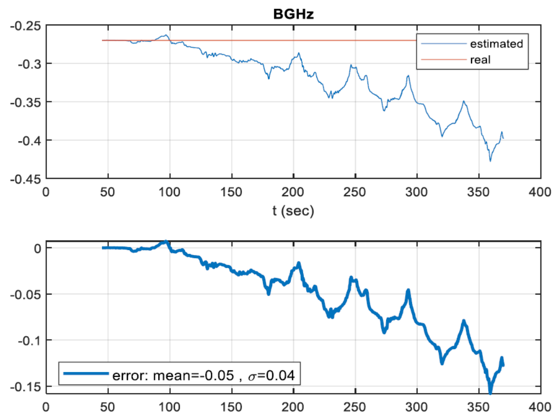

Fig. 17 Gyro bias (z)-axis true vs predicted (AFEKF)

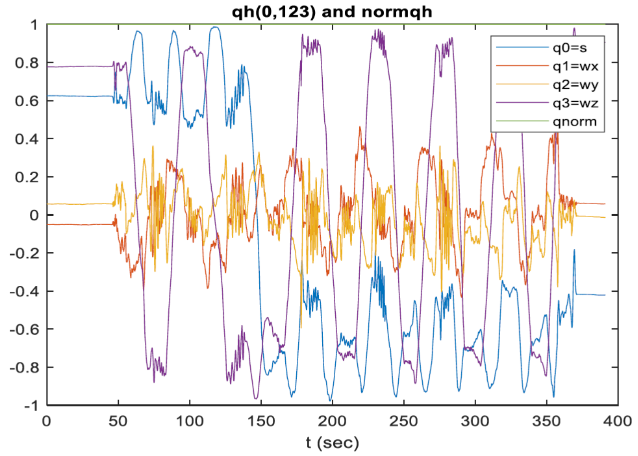

Fig. 18 Estimated quaternion (AFUKF)

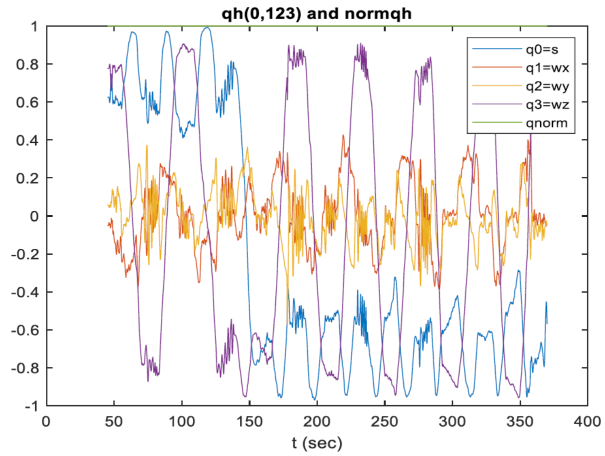

Fig. 21 Estimated quaternion (AFEKF)

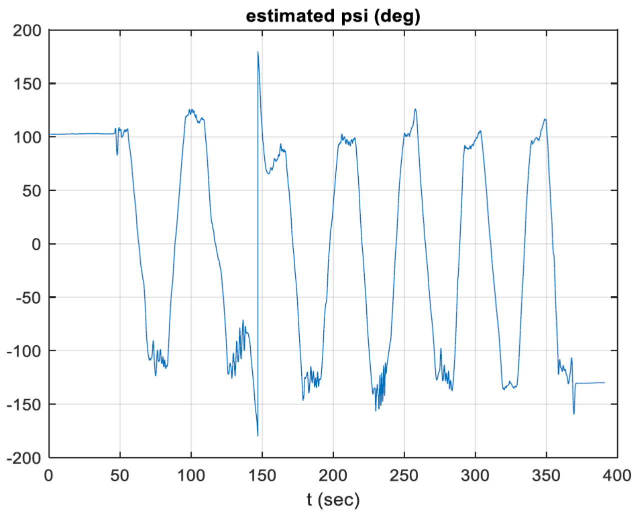

Fig. 19 Estimated psi (AFUKF)

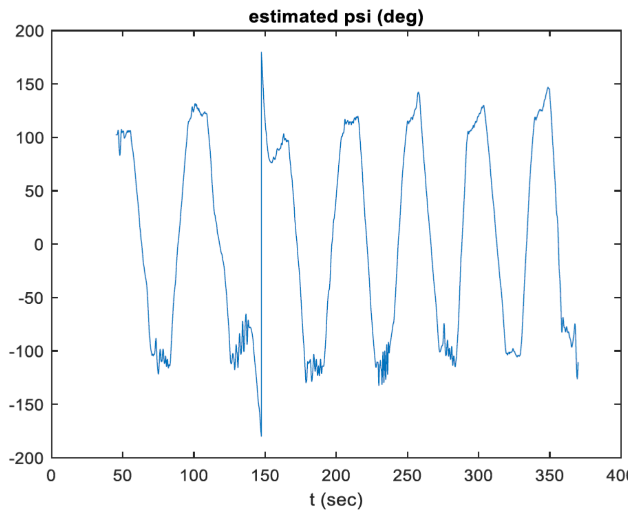

Fig. 22 Estimated psi (AFEKF)

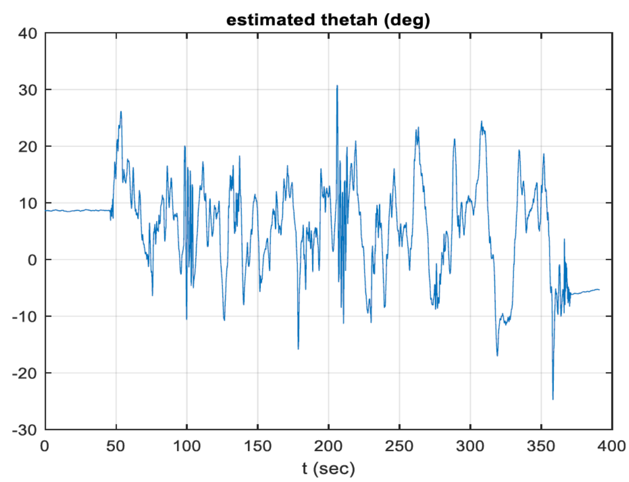

Fig. 20 Estimated theta (AFUKF)

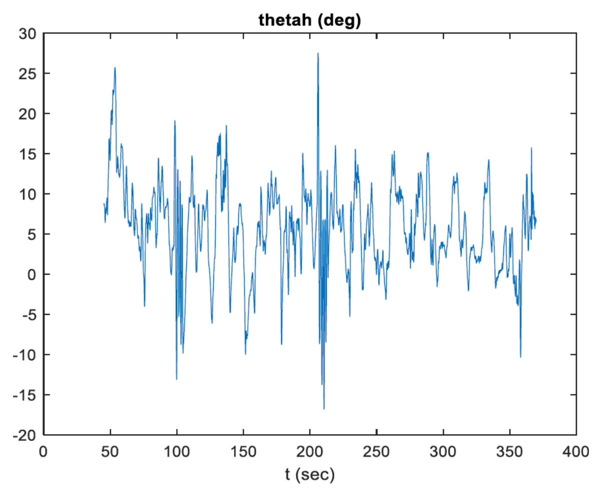

Fig. 23 Estimated theta (AFEKF)

Table 2 Demnostrate the comparsion between AFEKF & AFUKF of flight data (Acceleration & Gyro bias)

	bghx	
	Mean	Sigma
AFEKF	-0.0018	0.03
AFUKF	-0.01	0.00
	bghy	
	Mean	Sigma
AFEKF	0.00	0.020
AFUKF	-0.00	0.00
	bghz	
	Mean	Sigma
AFEKF	-0.05	0.0410
AFUKF	-0.00	0.00
	ax	
	Mean	Sigma
AFEKF	0	0.01
AFUKF	0.00	0.00
	ay	
	Mean	Sigma
AFEKF	0	0.02
AFUKF	0.00	0.01
	az	
	Mean	Sigma
AFEKF	0	0.07
AFUKF	0.00	0.00

Table 3 Demonstrate the comparsion between AFEKF & AFUKF of flight data (fnGPS and fnIMU, Angular velocity)

	*f*n(GPS, IMU) xn	
	Mean	sigma
AFEKF	0.086	1.267
AFUKF	0.00	0.00
	*f*n(GPS ,IMU) yn	
	Mean	sigma
AFEKF	0.088	0.9199
AFUKF	0.00	0.00
	*f*n(GPS ,IMU) zn	
	Mean	sigma
AFEKF	0.0286	2.699
AFUKF	0.00	0.01
	P	
	Mean	sigma
AFEKF	0.00	5.31
AFUKF	0.00	1.14
	Q	
	Mean	sigma
AFEKF	0.00	2.73
AFUKF	-0.00	0.28
	R	
	Mean	sigma
AFEKF	0.00	0.73
AFUKF	-0.00	0.12

We use in the paper standard SI units (m, s, m/s. rad) and a different dimension will be given only if it differs from SI.

The main results of the comparison of the two methods are illustrated in Fig. 6–23 (left column AFUKF and right column AFEKF) and Table 2–4. The UAV flight data is described in [2], sampling time 20 ms for IMU and magnetometer, and 500 ms for GPS. The first level of AF(U/E) KF uses z_1, z_2 for the innovations completed with second level z_3 if GPS measurements are present (notice the approximation character of z_3), and the extra level of (z_4, z_5) also for GPS, see the external loop for EKF [2].

Pre-filtering methods (hampel, sgolay in MATLAB) are used for the sensor measurements because of the large (2-3 G) and quick accelerations for maneuvering UAVs, and the differentiation needed for $f^n \rightarrow a^n$ conversion.

Transients in Fig. 6–17 illustrate the effect of sensor signal filtering based on innovations while Fig. 18–23 show some selected results from the state estimations. The long distance kinetic differential equations are numerically integrated and the new estimated states overwrite the integrated ones. Notice that different methods can give different state results while the etalon system is unknown.

9 Conclusion

In this paper two types of estimations methods have been developed and compared: a novel adaptive fuzzy filter (AFEKF) and another adaptive fuzzy filter (AFUKF) based on FIS Extended Kalman filtering, and FIS Unscented Kalman filtering for state estimation of maneuvering UAVs using sensor fusion for two loops:

1. The first filters UKF1 and EKF1 are used for orientation estimation based on unit quaternion, which works with sampling time T, they are executed when the GPS signal is absent and the measurements are acceleration, angular velocity (gyroscope signal), and magnetic field.
2. The second UKF2 and EKF2 filter with an improvement using GPS measurements together with acceleration, gyroscope and magnetic field signals. In this situation both work with sampling time TGPS of GPS measurements. The results of both AFEKF and AFUKF are accepted. AFEKF is adopted to improve the covariance matrix of measurement errors. In typical EKF, this matrix is used as constant matrix, AFEKF is well estimated this matrix.

Table 4 Demonstrate the comparison between AFEKF & AFUKF of flight data (Velocity, GPS & Magnetic field)

	VN	
	Mean	Sigma
AFEKF	-0.01	0.190
AFUKF	-0.10	0.28
	VE	
	Mean	Sigma
AFEKF	-0.01	0.141
AFUKF	-0.09	0.29
	VD	
	Mean	Sigma
AFEKF	-0.0044	0.349
AFUKF	0.04	0.37
	Lat	
	Mean	Sigma
AFEKF	0	0
AFUKF	-0.00	0.00
	Lon	
	Mean	Sigma
AFEKF	0	0
AFUKF	-0.00	0.00
	Alt	
	Mean	Sigma
AFEKF	0	0.12
AFUKF	-0.00	0.11
	Hx	
	Mean	Sigma
AFEKF	0.00	0.21
AFUKF	-0.00	0.06
	Hy	
	Mean	Sigma
AFEKF	0.00	0.26
AFUKF	0.02	0.12
	Hz	
	Mean	Sigma
AFEKF	0.00	0.26
AFUKF	-0.03	0.14

AFUKF is also used fuzzy technique to adapt the covariance matrix of measurement errors. An adaptation method for state error matrix R_k is also used same procedure in UKF, standard deviation and variance are calculated for each figures and the comparison between AFUKF and AFEKF computed and the latest results demonstrate that AFUKF consistently achieves better level of accuracy to estimate UAV data than AFEKF and the errors have been decreased, the basic concern is the error sigma & mean. The identification of the dynamic model of UAVs will be the focus of future study.

Appendix

A.1 Navigation frames

The navigation of vehicles is based on the WGS-84 standard of GPS. It uses the frames ECI, ECEF, NED and Body (vehicle body) denoted by i, e, n and b respectively. ECEF is moving together with the earth approximated by a rotational ellipsoid. Its sidereal rotation relative to the inertial frame ECI is $\omega_e = 7.2921151467 \cdot 10^{-5}$ rad / s.

In ECEF each point P can be characterized by a vector $r = (x, y, z)^T$ from the origin to the point. Because of the large distances the point can also be identified by the geodetic coordinates $p = (\varphi, \lambda, h)^T$, which are the latitude, longitude, and height, respectively, and $(x, y, z)^T$ can be determined from them. The intersection of the plane through P and the z-axis of the rotational ellipsoid can be determined. The height h of the point P is the shortest distance from the ellipse which defines the point Q. In Q the tangent of the ellipse can be determined. The line through Q orthogonal to the tangent intersects the z-axis in a point R. The QR section is normal to the tangent, its length is N. The NED frame has origin Q and axes x_N, y_E, and z_D in north, east and down direction, respectively.

GPS navigation is performed relative to ECEF while the IMU (3D acceleration and 3D angular velocity) sensors measure relative to ECI. We prefer the use of INS navigation.

A.2 Inertial (IMU) and Magnetometer sensors

Denote M and N the cross-directional curvatures of the rotational ellipsoid at Q and $R = \sqrt{MN}$ their geometrical average value.

Denote $\tilde{f}^b$ the 3D measurement of the acceleration sensor of the IMU and b_a its bias then its value in the NED frame is $f^n = R_b^n \left(\tilde{f}^b - b_a\right)$. Notice that f^n contains also the gravity effect $g^n = (0, 0, \gamma)^T$ whose approximation by the Rogers model is $\gamma(h) = g_0(R/(R+h))^2$ where h is the height and $g_0 = 9.81425$ is an average value for $h = 0$. Similarly, denote $\tilde{\omega}^b$ the 3D measurement of the angular velocity sensor of the IMU and b_ω its bias. Hence, for constant bias yields $\omega_{nb}^b = \tilde{\omega} - b_\omega - R_n^b \left(\omega_{ie}^n + \omega_{en}^n\right)$.

The Magnetometer sensor measures the 3D magnetic force H^b which can be transformed to $H^n = R_b^n H^b$. The geometric north and the magnetic north directions differ, the angle between them is the declination angle δ whose tangent is $T_\delta = H_y^n / H_x^n$, and δ can be determined by the function *wrldmagm* from the GPS coordinates and the actual date. Hence the magnetic force measurement supports the orientation (attitude) estimation.

A.3 Long distance kinetic differential equations

For long distance navigation it is useful to formulate the state equation in NED frame. The state, input and output are respectively:

$$x=\left(p^T,v^T,q^T,b_a^T,b_\omega^T\right)^T,\ u=\left(\tilde{a}^T,\tilde{\omega}^T\right)^T,\ y=\left(p^T,v^T\right)^T$$

The relative orientation between NED frame and Body frame can be characterized by the quaternion $q=(s,w)\in R^1\times R^3$ and $R_b^n=I_3+2s[w\times]+2[w\times]^2$. The derivatives of the constant biases are zero plus noise. The remaining kinematic state equations are as follows:

$$\begin{pmatrix}\dot{\varphi}\\ \dot{\lambda}\\ \dot{h}\end{pmatrix}=\begin{bmatrix}\frac{1}{M+h} & 0 & 0\\ 0 & \frac{1}{(N+h)C_\varphi} & 0\\ 0 & 0 & 1\end{bmatrix}\begin{pmatrix}v_N\\ v_E\\ v_D\end{pmatrix} \tag{A.1}$$

$$\begin{pmatrix}\dot{v}_N\\ \dot{v}_E\\ \dot{v}_D\end{pmatrix}=f^n+g^n+f_v\left(v_N,v_E,v_D,\varphi,h,M,N\right) \tag{A.2}$$

$$\dot{q}=\frac{1}{2}\begin{bmatrix}w^T\\ sI_3+[w\times]\end{bmatrix}\left(\tilde{\omega}^b-b_\omega-R_n^b\left(\omega_{ie}^n+\omega_{en}^n\right)\right) \tag{A.3}$$

A.4 Observation model

The measured or computed observation is assumed in the form $y = h(x) + n_y$.

<u>The output mappings:</u>

$$y_1=0 \tag{A.4}$$

$$y_2=\left(\dot{v}_N,\dot{v}_E,\dot{v}_D\right)^T-g^n-f_v\left(v_N,v_E,v_D,\varphi,h,M,N\right) \tag{A.5}$$

$$y_3=\left(v_N/\|v\|,v_E/\|v\|,v_D/\|v\|\right)^T \tag{A.6}$$

$$y_4=p \tag{A.7}$$

$$y_5=v \tag{A.8}$$

<u>The output measurements:</u>

$$z_1=\left[T_\delta,1,0\right]R_b^nH^b+n_{y1} \tag{A.9}$$

$$z_2=R_b^n\left(\tilde{a}^b-b_a\right)+n_{y2} \tag{A.10}$$

$$z_3=R_b^n\left[T_\delta,1,0\right]^T+n_{y3} \tag{A.11}$$

$$z_4=p+n_{y4} \tag{A.12}$$

$$z_5=v+n_{y5} \tag{A.13}$$

- Notice that Eq. (A.9) is equivalent to $T_\delta H_x^n=H_y^n$.
- Eq. (A.10) is the image of the acceleration sensor removing its bias which is the acceleration minus $g_n = (0, 0, \gamma)^T$, i.e. f^n.
- Eq. (A.11) assumes that the direction of the x_B axis is equal to the direction of $v^b = (U, V, W)$ in the Body frame, i.e. $V = W = 0$ that means that the kinematic angle and attack and the side slip angle are both zero which is only an approximation whose error is removed to n_{y3}. Critical may be for quickly maneuvering vehicles (UAV, UGV etc.) that the zero mean assumption of EKF cannot be satisfied. Notice that $\left(R_{b,11}^n,R_{b,21}^n,R_{b,31}^n\right)^T$ is a unit vector.
- Eqs. (A.12) and (A.13 are immediately the GPS measurements.

A.5 Short distance Flat Earth navigation

For short distance navigation we can assume that NED_0 is fixed and $R_{en_0}^{-1}R_{en}\approx I_3$ (i.e. latitude and longitude are approximately constant). Denoting the position difference vector by:

$$p:=\left(x,y,z\right)^T:=R_{e,n_0}^T\left(p_{eb}-p_{eb_0}\right)=p_{n_0b}-p_{n_0b_0}$$

and taking into consideration that $v = (v_N, v_E, v_D)^T$ is approximately its derivative, then state equations will be:

$$\begin{aligned}&\dot{p}=v,\quad \dot{v}=R_b^n\tilde{a}+\left(0,0,1\right)^Tg_0\\ &\dot{q}=\frac{1}{2}\begin{bmatrix}w^T\\ sI_3+[w\times]\end{bmatrix}\left(\tilde{\omega}^b-b_\omega\right)\end{aligned} \tag{A.14}$$

References

[1] Havangi, R., Teshnehlab, M., Nekoui, M. A. "A novel adaptive neuro-fuzzy unscented Kalman filter for SLAM", International Journal of Humanoid Robotics, 8(1), pp. 223–243, 2011. https://doi.org/10.1142/S0219843611002411

[2] Bodó, Z., Lantos, B. "State estimation for UAVs using sensor fusion", In: 2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY), Subotica, Serbia, 2017, pp. 000111–000116, 2017. ISBN 978-1-5386-3855-2 https://doi.org/10.1109/SISY.2017.8080535

[3] Assad, A., Khalaf, W., Chouaib, I. "Novel Adaptive Fuzzy Extended Kalman Filter for Attitude Estimation in GPS-Denied Environment", Gyroscopy and Navigation, 10(3), pp. 131–146. https://doi.org/10.1134/S2075108719030027

[4 Grewal, M. S., and Andrews, A. P. "Kalman filtering: Theory and practice", Prentice-HALL, 1993. ISBN-13: 978-0132113359

[5] Shi, Y., Han, C., Liang, Y. "Adaptive UKF for target tracking with unknown process noise statistics", In: 2009 12th International Conference on Information Fusion, Seattle, WA, USA, 2009, pp. 1815–1820. ISBN: 978-0-9824-4380-4

[6] Wan, E. A., Van Der Merwe, R. "The unscented Kalman filter for nonlinear estimation", In: Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373), Lake Louise, AB, Canada, 2000, pp. 153–158. ISBN: 0-7803-5800-7 https://doi.org/10.1109/ASSPCC.2000.882463

[7] Ge, B., Zhang, H., Jiang, L., Li, Z., Butt, M. M. "Adaptive Unscented Kalman Filter for Target Tracking with Unknown Time-Varying Noise Covariance", Sensors, 19(6), 1371, 2019. https://doi.org/10.3390/s19061371

[8] Jang, J. S. R. "ANFIS: adaptive-network-based fuzzy inference system", IEEE Transactions on Systems, Man, and Cybernetics, 23(3), pp. 665–685, 1993. https://doi.org/10.1109/21.256541

[9] Bai, S. M., Chen, S. M. "Automatically constructing concept maps based on fuzzy rules for adapting learning systems", Expert Systems with Applications, 35(1–2), pp. 41–49, 2008. https://doi.org/10.1016/j.eswa.2007.06.013

[10] Wang, J. J., Ding, W., Wang, J. "Improving Adaptive Kalman Filter in GPS/SDINS Integration with Neural Network", In: Proceedings of the 20th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2007), Fort Worth, TX, USA, 2007, pp. 571–578.

[11] Escamilla-Ambrosio, P. J., Mort, N. "Adaptive Kalman filtering through fuzzy logic", In: Proceedings of the 7th UK Workshop On Fuzzy Systems, Recent Advances and Practical Applications of Fuzzy, Neuro-Fuzzy, and Genetic Algorithm-Based Fuzzy Systems, Sheffield, UK, 2000, pp. 26–27.