State Estimation for UAVs Using Sensor Fusion

Zsófia Bodó*, Béla Lantos†
*Budapest University of Technology and Economics, Department of Control Engineering and Information Technology,
†MTA-BME Control Engineering Research Group, Budapest, Hungary

***Abstract*—In this paper an improved approach is presented for the state estimation of unmanned aerial vehicles (UAVs). The three-loop technique is based on Extended Kalman Filters. From them EKF1 solve the quaternion based orientation (attitude) estimation using IMU and magnetometer. EKF2 improves the attitude estimation if GPS information is present. The third filter EKF3 determines the remaining state variables including the biases in an external loop if GPS measurement is available and tolerates the large difference between IMU and GPS frequencies. The method can be applied for state estimation of any type of vehicles. It is especially useful for vehicles having large (2-3 G) acceleration changes typical for UAVs. The results can latter be used for identification of the nonlinear dynamic model of vehicles (control surface, thrust and inertia effects) building the basis for advanced control.**

***Index Terms*—State estimation, EKF approach, Sensor fusion, UAV, Real flight data**

I. Introduction

State estimation for UAVs (Unmanned Aerial Vehicles) is an intermediate important step in modeling and control. The paper concentrates on the state estimation only. Popular approaches are based on EKF (Extended Kalman Filter) [1], sigma-point estimators among them UKF (Unscented Kalman Filter) [2] and symmetry-preserving observers (SPO) using Lie-Group technique [3]. A comparison of the methods can be found in [4]. Novel methods of tuning the error state coveriance matrix were suggested in [5] and [6]. State estimation for a sailplane was considered in [7] and [8].

In this paper an improved EKF method will be presented for an UAV based on the kinematic differential equations of navigation according to the WGS-84 standard of GPS.

The kinematic differential equations use the frames of ECI (Earth Centered Inertia), ECEF (Earth Centered Earth Fixed), NED (North, East, Down) and Body (vehicle body) coordinate systems. In the sequel they will be referred by the letters i, e, n and b, respectively.

ECEF is moving together with the earth. Its sidereal rotation rate relative to ECI is $\omega_E = 7.2921151467 \cdot 10^{-5}$rad/s. The earth can be approximated by a rotational ellipsoid.

In ECEF frame each point P can be characterized by a vector $r = (x, y, z)^T$ from the origin to the point. However, in ECEF the point can also be identified by the geodetic coordinates $p = (\varphi, \lambda, h)^T$, which are the latitude, longitude and hight, respectively, and $(x, y, z)^T$ can be determined from them. Conversion methods amongst them are available.

First the intersection of the plane through P and the z-axis with the rotational ellipsoid will be determined. The intersection is an exemplar of the rotated ellipse. The hight h of the point P is the shortest distance from the ellipse which defines the point Q on the ellipse. In point Q the tangent of the ellipse can be determined. The line through Q orthogonal to the tangent intersects the z-axis in the point $R = (0, -c)$. The QR section is the normal to the tangent, its length is N.

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.

II. INS navigation

The orientation of the K_n frame relative to K_e, i.e. the mapping from K_n to K_e, can be found by using two rotations:

$$R_n^e = Rot(z, \lambda) Rot(y, -(\frac{\pi}{2} + \varphi)) \tag{1}$$

The orientation matrix R_b^n (attitude, DCT) is the transformation from K_b to K_n. It can also be described by Euler (roll, pitch, yaw) angles ϕ, θ, ψ or unit quaternion $q = (s, w)$ where $s = q0 \in R^1$ and $w = (q_1, q_2, q_3)^T \in R^3$ yielding

$$R_b^n = Euler(\phi, \theta, \psi) = I_3 + 2s[w\times] + 2[w\times][w\times] \tag{2}$$

Here $[w\times]$ is the matrix of vector product. The quaternion product is $q_1 \star q_2 = (s_1 s_2 - w_1^T w_2,\ w_1 \times w_2 + s_1 w_2 + s_2 w_1)$.

There are different techniques for orientation characterization, however conversion between them is possible [9]. Especially, let R_b^n and q the orientation description of K_b relative to K_n and denote $\omega_{nb}^b = (P, Q, R)^T =: \omega$ the angular velocity of the vehicle relative to K_n.

A. Differential equations of the orientation descriptions

$$\frac{d}{dt}\begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix} = \begin{bmatrix} 1 & T_\theta S_\phi & T_\theta C_\phi \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi / C_\theta & C_\phi / C_\theta \end{bmatrix} \begin{pmatrix} P \\ Q \\ R \end{pmatrix} \tag{3}$$

$$\begin{aligned} \dot{q} &= \frac{1}{2}\begin{bmatrix} 0 & -\omega^T \\ \omega & -[\omega\times] \end{bmatrix}\begin{pmatrix} s \\ w \end{pmatrix} = \frac{1}{2}\begin{bmatrix} -\omega^T w \\ s\omega - \omega \times w \end{bmatrix} \\ &= \frac{1}{2}\begin{bmatrix} -w^T \\ sI_3 + [w\times] \end{bmatrix}\omega =: T(q)\omega \end{aligned} \tag{4}$$

Here we used the short notations S, C, T for $\sin, \cos, \tan$.

B. Long distance INS navigation

In case of long distance INS navigation the NED frame cannot be considered fixed, i.e. it is a wandering NED frame. The orientation of the body frame will be considered relative to the moving NED.

Denote M and N the cross-directional curvatures and R their average value:

$$M = \frac{a(1-e^2)}{(1-e^2 S_\varphi^2)^{3/2}},\ N = \frac{a}{\sqrt{1-e^2 S_\varphi^2}},\ R = \sqrt{MN} \quad (5)$$

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 $f^n = R_b^n(\tilde{f}^b - b_a)$. Notice that f^n contains also the gravity effect $g^n = (0\,0\,\gamma)^T$ whose approximation by the simple Rogers model is $\gamma(h) = g_0(R)/(R+h))^2$ where h is the hight 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. Notice that the IMU measures the acceleration and the angular velocity relative to ECI. Since $\omega_{ib}^b = R_n^b(\omega_{ie}^n + \omega_{en}^n) + \omega_{nb}^b$ therefore for constant bias yields:

$$\omega := \omega_{nb}^b = \tilde{\omega}^b - b_\omega - R_n^b(\omega_{ie}^n + \omega_{en}^n)$$

$$\omega_{ie}^n + \omega_{en}^n = \begin{pmatrix} C_\varphi \\ 0 \\ -S_\varphi \end{pmatrix} \omega_e + \begin{pmatrix} v_E/(N+h) \\ -v_N/(M+h) \\ -v_E T_\varphi/(N+h) \end{pmatrix}$$

where φ is the geodetic latitude.

<u>Kinetic equations of GPS with IMU</u> [10]:

$$\begin{pmatrix} \dot{\varphi} \\ \dot{\lambda} \\ \dot{h} \end{pmatrix} = \begin{bmatrix} \frac{1}{M+h} & 0 & 0 \\ 0 & \frac{1}{(N+h)C_\varphi} & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{pmatrix} v_N \\ v_E \\ v_D \end{pmatrix} \quad (6)$$

$$\begin{pmatrix} \dot{v}_N \\ \dot{v}_E \\ \dot{v}_D \end{pmatrix} = f^n + g^n + \quad (7)$$

$$\begin{pmatrix} -v_E\left[2\omega_e + \frac{v_E}{(N+h)C_\varphi}\right] S_\varphi + \frac{v_N v_D}{M+h} \\ \left[\frac{v_E}{(N+h)C_\varphi} + 2\omega_e\right] v_N S_\varphi + v_D\left[2\omega_e + \frac{v_E}{(N+h)C_\varphi}\right] C_\varphi \\ -\frac{v_N^2}{M+h} - 2v_E\omega_e C_\varphi - \frac{v_E^2}{N+h} \end{pmatrix}$$

$$\dot{q} = \frac{1}{2}\begin{bmatrix} -w^T \\ sI_3 + [w\times] \end{bmatrix} \cdot \quad (8)$$

$$\left(\tilde{\omega}^b - b_\omega - [I - 2s[w\times] + 2[w\times]\,[w\times]]\,(\omega_{ie}^n + \omega_{en}^n)\right)$$

The derivatives of the constant biases are simply $\dot{b}_a = 0$ and $\dot{b}_\omega = 0$.

Notice that in the above kinematic equations the 'position' vector $p = (\varphi,\, \lambda,\, h)^T$ represents the real position vector $r = (x, y, z)^T$ from ECEF to BODY in abstract form eliminating the large order of the components of r. Its derivative $R_e^n \dot{r} = v = (v_N,\, v_E,\, v_D)^T$ is the relative velocity of the vehicle to the (rotating) ECEF frame expressed in the basis of the NED frame. The quaternion q represents the relative orientation of the BODY frame to the NED frame. However, the IMU measurements are relative to ECI but expressed in the BODY frame. In the presented concept the NED frame plays the central role hence the navigation is called INS navigation.

C. *Nonlinear state equations of long distance INS navigation*

In the presence of GPS sensor it is useful to formulate the state equation in the NED frame. The state equation was already presented for the choice of state $x = (p^T, v^T, q^T, b_a^T, b_\omega^T)^T$, input $u = (\tilde{a}^T, \tilde{\omega}^T)^T$ and output $y = (p^T, v^T)^T$. From the system noises $\tilde{a}_{noise}$ is additively contained in $\tilde{f}^b$ and $\tilde{\omega}_{noise}$ is additively contained in $\tilde{\omega}$, respectively. The outputs contain additive noises $\tilde{p}_{noise}$ and $\tilde{v}_{noise}$, respectively. The nonlinear state equations of INS navigation have the form

$$\dot{x} = f(x, u, n_x) \quad (9)$$

$$y = h(x, n_y) = \begin{bmatrix} I_3 & 0 & 0 & 0 & 0 \\ 0 & I_3 & 0 & 0 & 0 \end{bmatrix} = Cx + n_y \quad (10)$$

If the state estimation is performed in discrete time then the state equations have to be converted from continuous to discrete time by using Euler method or similar ones. If the state estimation uses EKF then the state equations should be linearized by the variables in every multiple of the sampling time. During linearizing the task is to find the derivatives of the right sides f_p, f_v, f_q of the state equation by their variables. During derivation the curvatures M, N, R can be assumed constants. The requested form is:

$$\begin{aligned} \delta\dot{x} &= A(t)\delta x + B(t)\delta u + B_n(t) n_x \\ \delta y &= C(t)\delta x + n_y \end{aligned} \quad (11)$$

In order to increase the precision the state estimator can internally contain the integration of the state equations. The estimated state can overwrite the computed state. The derivatives can be used in the EKF (inner loop) running with the high sampling frequency of the IMU.

If the motion is limited to the close neighborhood of the initial stationary place of the vehicle then the INS navigation can be simplified. In this case the motion is limited to the neighborhood if the initial NED frame (called NED_0) which will be considered to be a (quasi) inertial system (Flat Earth Navigation).

D. *Integrating magnetic sensor measurements*

Let us assume that the vehicle is standing and magnetometer and IMU measurements are available. The magnetic field measurement is $H^b = (H_x^b, H_y^b, H_z^b)^T$ which can be transformed in the NED frame by R_b^n that is:

$$R_b^n = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix}$$

$$\begin{aligned} H_x^n &= (C_\theta H_x^b + S_\theta S_\phi H_y^b + S_\theta C_\phi H_z^b) C_\psi \\ &\quad + (-C_\phi H_y^b + S_\phi H_z^b) S_\psi \\ &=: a_1 C_\psi + a_2 S_\psi \quad (12) \\ H_y^n &= (C_\phi H_y^b - S_\phi H_z^b) C_\psi \\ &\quad + (C_\theta H_x^b + S_\theta S_\phi H_y^b + S_\theta C_\phi H_z^b) S_\psi \\ &=: b_1 C_\psi + b_2 S_\psi \quad (13) \end{aligned}$$

In stationary situation the accelerometer measures the negative gravity acceleration which would be in NED frame $(0,\,0,\,-1)^T$ if it is scaled in G, hence, transforming in BODY we obtain

$$\tilde{a} = \tilde{f} = -\left[\begin{array}{ccc} -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{array}\right]^T \tag{14}$$

$$\phi = \mathrm{atan2}(a_y, a_z), \quad \theta = \mathrm{atan2}(-C_\phi a_x, a_z) \tag{15}$$

and therefore a_1, a_2, b_1, b_2 are cumputable.

On the other hand, the geometrical and magnetic North directions in NED frame are not coinciding, the difference between them is the magnetic declination angle δ, hence we can form an <u>electronic compass</u> as follows:

$$\begin{aligned} T_\delta &= \frac{H_y^n}{H_x^n} = \frac{b_1 C_\psi + b_2 S_\psi}{a_1 C_\psi + a_2 S_\psi} \Rightarrow \\ S_\psi(-T_\delta a_2 + b_2) &= C_\psi(T_\delta a_1 - b_1) \\ T_\psi &= \frac{T_\delta a_1 - b_1}{-T_\delta a_2 + b_2} \xrightarrow{\mathrm{atan}} \psi \end{aligned} \tag{16}$$

Specially, $\delta = 3.8202 * \pi/180$ [rad] at Budapest Ferihegy. Notice that a function *wrldmagm* is available for determining δ anywhere from GPS coordinates and the actual date, furthermore this function is updated in every fifth year.

III. State estimation using EKFs

The most popular state estimator is the extended Kalman filter (EKF). However, its stability is not guaranteed and it needs the derivatives of the nonlinear functions in the discrete time model which is a hard problem with increasing dimension of the state vector.

EKF is a stochastic state estimation method based on the linearization of the discrete time state equations in every integer multiple of the sampling time. First the EKF algorithm will be summarized. Then the derivatives of the state equations will be given for INS navigation. Finally the application of EKF will be demonstrated in vehicle navigation.

A. The EKF algorithm

Consider the discrete time nonlinear system in the form

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

where w is the system noise and z is the measurement noise. It is assumed that w and z are not correlated and have zero mean, their covariance matrices $R_{v,k-1}$ and $R_{z,k}$ are known. The mean value of the input state x_0 and its covariance matrix S_0 should also be known. The EKF algorithm can be performed by introducing the following notations:

$$\begin{aligned} A_{k-1} &= \frac{\partial f(\hat{x}_{k-1}, u_{k-1}, 0)}{\partial x} \quad & B_{w,k-1} &= \frac{\partial f(\hat{x}_{k-1}, u_{k-1}, 0)}{\partial w}, \\ C_k &= \frac{\partial g(\bar{x}_k, 0)}{\partial x} & C_{z,k} &= \frac{\partial g(\bar{x}_k, 0)}{\partial z}, \end{aligned} \tag{17}$$

where $\hat{x}_k$ is the estimated and $\bar{x}_k$ the predicted value of x_k. The well-known two steps (Prediction, Time Update) can be found for example in [1] or [9]. Notice that S_k is the covariance matrix of $x_k - \hat{x}_k$ while M_k is the covariance matrix of $x_k - \bar{x}_k$. The differentiations in (17) are parts of the time update step. A collection of the necessary derivatives of A_{pp}, A_{pv} etc. for INS navigation can be found in [9].

B. Observation model

It will be assumed that for the state estimation IMU (3D accelerometer and 3D angular velocity sensor) and 3D magnetometer, both based on MEMS technology, and a GPS receiver are available. The state estimation is solved by their fusion taking into consideration the different sampling frequencies. We assume the measured or computed output observation in the form $y = g(x) + n_y$.

<u>The output mappings</u>

$$y_1 = g_1(x) + n_{y_1} = \left[\begin{array}{ccc} T_\delta & -1 & 0 \end{array}\right] R_b^n H^b + n_{y_1} \tag{18}$$

$$y_2 = g_2(x) + n_{y_2} = R_b^n(\tilde{a}^b - b_a) + n_{y_2} \tag{19}$$

$$y_3 = g_3(x) + n_{y_3} = R_b^n \left[\begin{array}{ccc} 1 & 0 & 0 \end{array}\right]^T + n_{y_3} \tag{20}$$

$$y_4 = g_4(x) + n_{y_4} = p + n_{y_4} \tag{21}$$

$$y_5 = g_5(x) + n_{y_5} = v + n_{y_5} \tag{22}$$

These mappings reflect the following concept:

- Equation (18) is equivalent to $T_\delta H_x^n = H_y^n$, see also (II-D).
- Equation (19) is the image of the acceleration sensor measurement removing its bias which is the acceleration minus the gravity acceleration $g^n = (0, 0, \gamma)^T$ in NED frame, i.e. f^n, see also (II-B).
- Equation (20) assumes that the direction of the x_B axes is equal to the direction of the velocity $v^b = (U, V, W)^T$ in body frame, i.e. $V = W = 0$. This assumption means that the kinematic angle of attack and the kinematic side slip angle are both zero. This is only an approximation whose error is removed to the noise n_{y_3}. Critical may be for quickly maneuvering unmanned vehicles (UAVs, UGVs etc.) that the zero mean assumption of EKF cannot be well satisfied. Notice that $g_3 = (R_{b,11}^n,\, R_{b,21}^n\, R_{b,31}^n)^T$ which is a <u>unit vector</u>.
- y_4 and y_5 are immediately the GPS measurements.

<u>The output measurements</u>

$$y_1 = 0 \tag{23}$$

$$y_2 = \begin{pmatrix} \dot{v}_N \\ \dot{v}_E \\ \dot{v}_D \end{pmatrix} - g^n - \tag{24}$$

$$\begin{pmatrix} -v_E\left[2\omega_e + \frac{v_E}{(N+h)C_\varphi}\right] S_\varphi + \frac{v_N v_D}{M+h} \\ \left[\frac{v_E}{(N+h)C_\varphi} + 2\omega_e\right] v_N S_\varphi + v_D\left[2\omega_e + \frac{v_E}{(N+h)C_\varphi}\right] C_\varphi \\ -\frac{v_N^2}{M+h} - 2v_E\omega_e C_\varphi - \frac{v_E^2}{N+h} \end{pmatrix}$$

$$y_3 = \left[\begin{array}{ccc} v_N/\|v\| & v_E/\|v\| & v_D/\|v\| \end{array}\right]^T \tag{25}$$

$$y_4 = p \tag{26}$$

$$y_5 = v \tag{27}$$

<u>The following noise covariance matrices can be suggested:</u>

$$\mathrm{cov}_{n1} = \left(\begin{bmatrix} T_\delta & -1 & 0 \end{bmatrix} R_b^n\right) \mathrm{cov}_{\tilde{H}^b} \left(\begin{bmatrix} T_\delta & -1 & 0 \end{bmatrix} R_b^n\right)^T$$
$$\mathrm{cov}_{n2} = R_b^n \, \mathrm{cov}_{\tilde{a}^b} \, (R_b^n)^T \tag{28}$$
$$\mathrm{cov}_{n3} = \|\mathrm{cov}_{\tilde{\omega}^b}\|_2 \begin{bmatrix} s_1 & 0 & 0 \\ 0 & s_2 & 0 \\ 0 & 0 & s_3 \end{bmatrix}$$

As can be seen, we suggested to derive the covariance of n_3 from the covariance of $\tilde{\omega}^b$ of the IMU. However, as we mentioned above, experimentally chosen $s_1,\, s_2,\, s_3$ scaling factors are needed in order to suppress the influence of not zero mean error.

IV. Implementation Concept of INS navigation

We suggest the following EKF based state estimation concept for long distance INS navigation using sensor fusion:

- The realization consists of the continuous integration of the kinematic equations driven by the IMU sensors and three EKFs (EKF1,EKF2, EKF3). Sampling time of IMU, magnetometer and GPS may be for example 20ms, 50ms and 500ms, respectively. The declination angle δ and the magnetic force H^n in NED can be computed from the GPS position using the `wrldmagm` function in MATLAB. Gravity acceleration can be determined from GPS position using Rogers or Schwartz&Wei model.
- Sensor data are filtered and partly differentiated real time before use. For filtering the Savitzky-Golay (polynomial) FIR smoothing filter can be applied which is supported in MATLAB function `sgolay`. Differentiation can be solved based on its outputs and realized in a function `diffsgolay`.
- Since the sampling time T_m of the 3D magnetometer is near to the sampling time T of the IMU ($T_m > T$) hence $\tilde{H}^b$ will be re-sampled with T which simplify the realization. Then we will have IMU and magnetometer data with the same frequency.
- The integration of the kinematic state equations in INS is performed for the whole state $x = (p^T, v^T, q^T, b_a^T, b_\omega^T)^T$ using Euler or Runge-Kutta methods (RK2, RK4).
- EKF1 is used for orientation estimation based on unit quaternion. It works with sampling time T, uses $x_1 = (q^T, b_\omega^T)^T$ and the sensor inputs of IMU and magnetometer. Output mappings of the observation model are $g_1,\, g_2$, output measurements are $y_1,\, y_2$ while output noises are $n_1,\, n_2$. The estimated state $\hat{x}_1$ of EKF1 overwrites the actual value of portion x_1 of the running kinematic equations.
- EKF2 is used for estimating the whole state. It works with sampling time T_{GPS} of GPS measurements, uses $x = (p^T, v^T, q^T, b_a^T, b_\omega^T)^T$ and the sensor inputs of IMU, magnetometer and GPS position and state. Output mappings of the observation model are $g_1,\, g_2,\, g_3$, output measurements are $y_1,\, y_2,\, y_3$ while output noises are $n_1,\, n_2,\, n_3$. The estimated state $\hat{x}$ of EKF2 overwrites the actual value of x of the running kinematic equations.
- EKF3 is used the correct the whole state in an external loop. Its work is divided into two steps.

  The main step is performed if GPS measurement is obtained. It takes the difference of the computed position and velocity (determined through integration of the kinematic equations) and the measured GPS position and velocity. The difference is called $\delta y = (\delta p^T,\, \delta v^T)^T$ and used as output measurement to produce $\delta\hat{x}$. This correction will be added to the actual value of the whole state of the running kinematic equations: $x := x + \delta\hat{x}$. After the correction was made then $\delta\hat{x}$ is zeroed for the next GPS step.

  Since T_{GPS} is large relative to $T = T_{IMU}$ hence the linear behavior of EKF3 would be inaccurate for application. Hence a finer step $T_c = T_{GPS}/n_c$ will be chosen such that T_c is integer multiple of T. For example, T_c =100ms can be chosen in the above example where $n_c = 5$. In every integer multiple of T_c a correction step will be performed in order to compensate the difference between linear approximation and nonlinear behavior.

  The goal of the correction is to update the state and the covariances in EKF3. The base of the correction is the rule for time varying linear systems $\dot{x} = A(t)x + B(t)n$, in our case with noise input, by which

  $$x_{k+1} = \Phi_k \, x_k + \Gamma_k \, n_k$$
  $$\Phi_k = e^{A_k T_c}, \quad \Gamma_k = \int_0^{T_c} e^{A_k \sigma} d\sigma B_k \approx \Phi_k T_c B_k$$

  Hence, if the state and noises are not correlated and the noises have zero mean then the resulting state covariance matrix (S) and the noise covariance matrix (Q) can be computed recursively:

  $$S_{k+1} = \Phi_k S_k, \; Q_{k+1} = Q_k + (\Phi_k B_k) \, T_c^2 \, \mathrm{cov}_n \, (\Phi_k B_k)^T$$

  If the GPS measurement arrives the EKF3 performs the corrected prediction and time update steps for the outer loop. After the correction a re-initialization follows according to $S_0 := I_n$ and $Q_0 := 0_{n\times n}$ where $n = \dim x$.

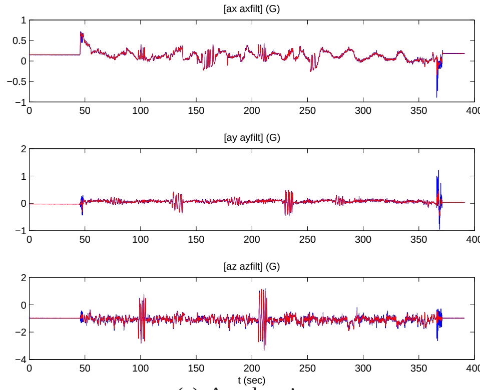
V. Experimental results using UAV flight data

The concept of state estimation using three level Extended Kalman Filters were tested for real flight data of a moving UAV. The sensor was of type mNAV100CA integrating IMU, magnetometer and GPS using MEMS technology. The UAV was remotely controlled from the ground and the commands were performed by the control system of the UAV. We obtained the logged flight data from MTA SZTAKI Systems and Control Laboratory in the frame of a cooperation.

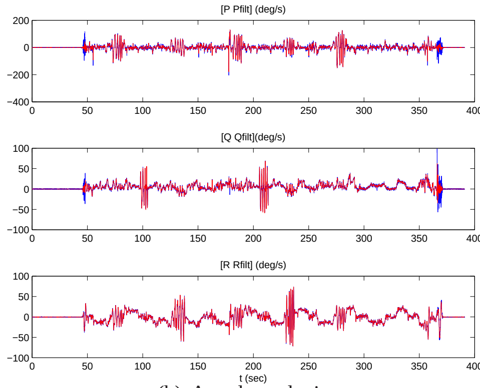
As a consequence of the on-board control system the UAV is moving dominantly in the direction of the body x_B axes so that the angle of attack is approximately zero and the error of output mapping $g_3(x)$ is relatively small which can positively influence the quality of state estimation.

On the other hand, because of quickly varying remote commands, i.e. reference signals, the control signals are also quickly varying and the same is valid also for the IMU sensor signals. The acceleration can be 2G or higher during the transients which makes the state estimation more difficult.

Some parts of the sensor signals are shown in Fig. 1-2 where the above effects (quick changes, oscillations) can be well detected. Beside the original sensor signals the filtered ones using `sgolay` are also shown. In case of the GPS velocity the derivatives can also be determined and used latter for computing f^n, the image of $\tilde{a}^b$ in the NED frame. Notice that this signal can be computed from the GPS signals as well. If the orientation R_b^n will be determined using the estimated quaternion then the transformation can be performed and the difference of the two values (`fn` $-$ `fncomp`) has to be small if the estimated orientation is acceptable. Small error can prove good orientation estimation.



(a) Acceleration



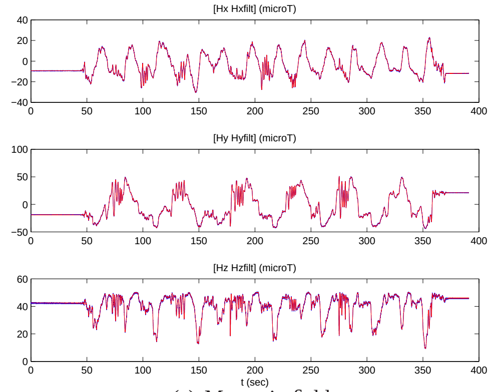
(b) Angular velocity

Fig. 1. IMU acceleration and angular velocity

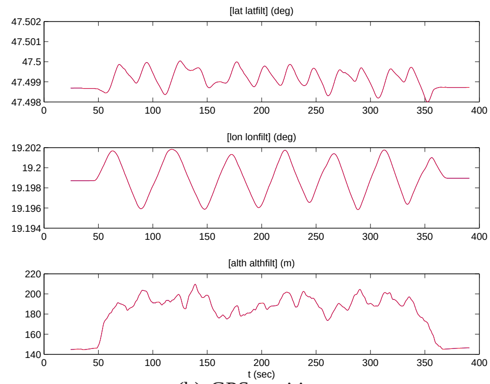
Some parts of the results of the state estimation using EKF1, EKF2 and EKF3 with inner integration of the long distance kinematic equations in INS assumption are shown in Figs. 3-5. Estimated variables are denoted by 'h' in the name.

Experiments using real vehicle's data show that real time implementation of the state estimation methods is possible on low-cost architectures.

The results in Fig. 4 show that the measured and the estimated velocities are similar except on places where a, ω or H are quickly varying. Other type direct computation of R_b^n shows that not the estimation but the GPS sensor itself is responsible for the differences, namely the applied GPS version on the UAV cannot follow quick changes. Except of



(a) Magnetic field



(b) GPS position

Fig. 2. Magnetic field and GPS position

these places, `fnGPSh` and `fnIMU` are also similar, see Fig. 5. Moreover, the computed U, V, W behave also according to $g3$ in (20), hence the chosen measurement covariances can well tolerate the nonzero-mean character of noises.

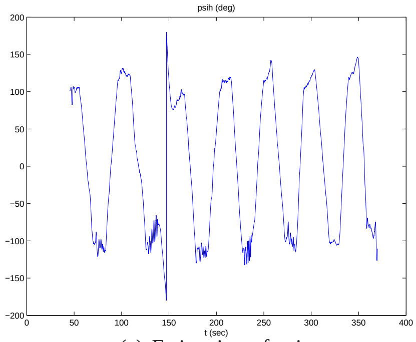
VI. Conclusion

In this paper a three-loop technique has been elaborated for the state estimation using Extended Kalman Filters. EKF1 solves the quaternion based attitude estimation using IMU and magnetometer. EKF2 improves the attitude estimation if GPS information is present. EKF3 determines the remaining state variables including the biases in an external loop if GPS measurement is available. The method can well tolerate the large difference between IMU and GPS frequencies and can be applied for any type of outdoor vehicles. The efficiency of the method was demonstrated for real flight data of an UAV.

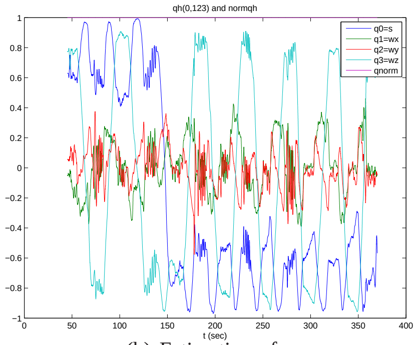
Future research will concentrate on the identification of the dynamic model of UAVs. Developments for the comparison of EKF and UKF are in progress.

Acknowledgements

The research of B. Lantos was supported by the MTA-BME Control Engineering Research Group.

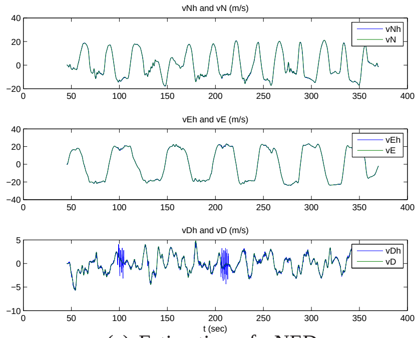


(a) Estimation of psi

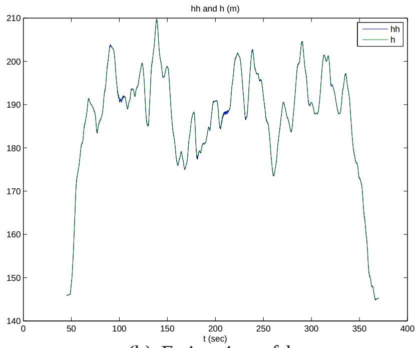


(b) Estimation of q

Fig. 3. Estimation of psi and quaternion q

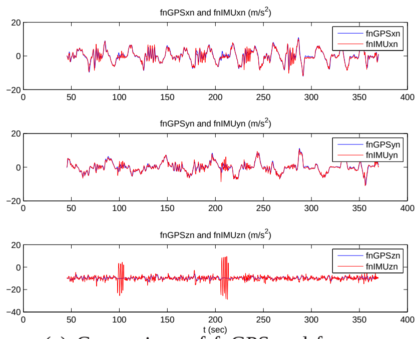


(a) Estimation of vNED

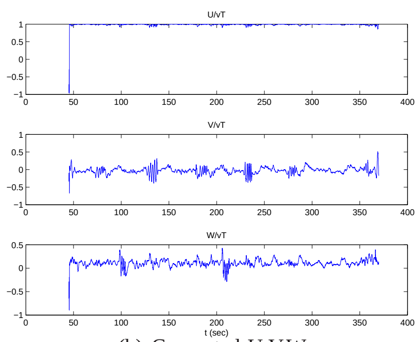


(b) Estimation of h

Fig. 4. Estimation of velocity vNED and hight h



(a) Comparison of fnGPS and fncomp



(b) Computed U,V,W

Fig. 5. Comparison of fn (GPS) and fn (computed) resulting in small error in accordance with U,V,W (computed)

REFERENCES

[1] B. Lantos, *Theory and Design of Control Systems II. (in Hungarian)*. Budapest: Akademiai Kiado, 2003.

[2] R. van der Merwe, E. Wan, and S. Julier, "Sigma-point kalman filters for nonlinear estimation and sensor fusion. applications to integrated navigation," in *Collection of Technical Papers. AIAA Guidance, Navigation and Control Conference*, Providence, USA, August 2004, pp. 1735–1764.

[3] S. Bonnabel, P. Martin, and P. Rouchon, "Symmetry-preserving observers," *IEEE Trans. on Automatic Control*, vol. 53, no. 11, pp. 2514–2526, 2008.

[4] G. Max and B. Lantos, "A comparison of advanced non-linear state estimation techniques for ground vehicles," *Hungarian Journal of Industry and Chemistry*, vol. 42, no. 2, pp. 57–64, 2014.

[5] J. Cheng, D. Chen, R. Landry, L. Zhao, and D. Guan, "An adaptive unscented kalman filtering algorithm for mems/gps integrated navigation systems." *Hindawi Publishing Corporation, Journal of Applied Mathematics, Article ID 451939*, pp. 1–8, 2014.

[6] J. Kim, B. Kiss, and D. Lee, "An adaptive unscented kalman filtering approach using selective scaling," in *IEEE Conference SMC2016*, Budapest, Hungary, October 2016, pp. 1–6, Conference CD.

[7] L. Lukács, *Advanced design methods for unmanned aerial vehicle control. PhD Thesis*. Budapest University of Technology and Economics, Dept. Control Engineering and Informatics, 2017.

[8] L. Kis and B. Lantos, "Development of state estimation system with INS, Magnetometer and Carrier Phase GPS for Vehicle Navigation," *Gyroscopy and Navigation*, vol. 5, no. 3, pp. 153–161, 2014.

[9] B. Lantos, *Efficient state estimation methods for orientation initialization and complete state estimation besed on sensor fusion. Task 2/4, Research Report*. Győr, Hungary: Széchenyi University, TAMOP-4.2.2.A-11/1/KONV-2012-0012, 2013.

[10] J. Farrell and M. Barth, *The Global Positioning System & Inertial Navigation*. New York: McGraw-Hill, 1999.