2013 International Conference on Unmanned Aircraft Systems (ICUAS)
May 28-31, 2013, Grand Hyatt Atlanta, Atlanta, GA

# Observer-control scheme for autonomous navigation: Flight tests validation in a quadrotor vehicle

L. E. Muñoz[1] and P. Castillo[2] and P. Garcia[3]

*Abstract*— **We present in this paper an improvement of a nonlinear control algorithm based on the Lyapunov analysis and the saturation functions, to realize autonomous navigation of a quadrotor vehicle. The algorithm is analyzed and the convergence of the states is ameliorated, in addition, the stability analysis in closed-loop system is proved with these new conditions. In order to locate the aerial vehicle a new position estimation algorithm is developed using the dead-reckoning technique with the Extended Kalman Filter (EKF). This algorithm is based in the data fusion of the classical sensors; an Inertial Measurement Unit (IMU), an ultrasonic sensor and a vision system. The algorithms are validated on-board in flight tests to realize autonomous navigation of the quadrotor vehicle. The most important results are depicted in some graphs.**

## I. Introduction

An Unmanned Aerial Vehicle - UAV, also called drone, is a self-descriptive term commonly used to describe military and civil applications of the latest generations of pilotless aircraft. UAVs are defined as aircrafts without the onboard presence of human pilots, used to perform intelligence, surveillance, and reconnaissance missions. The technological objective of UAVs is to serve across the full range of missions cited previously. UAVs present several basic advantages compared to manned systems that include better manoeuvrability, lower cost, smaller radar signatures, longer endurance, and minor risk to crew. The UAVs are also reliable and have the capacity to navigate autonomously in a prescribed trajectory avoiding obstacles. For safety reasons, the total weight of the UAV should be as low as possible. The achievement of this objective certainly requires innovations from both technological and scientific points of view.

For autonomous navigation an imperative need for UAV autonomy is the ability to self-localization in the environment. Indeed, precise localization is crucial in order to achieve high performance flight and to interact with the environment. The design of position controllers has been the focus of several groups in the research community, which has resulted in significant and interesting breakthroughs in this field. Nevertheless, existing position controllers require that all system states are accurately measured.

*This work was supported by CONACYT

[1]L. E. Muñoz is with the Laboratory Heudiasyc UMR 7253, University of Technology of Compiègne, Compiègne Cedex 60205, France. (lmunozhe@hds.utc.fr)

[2]P. Castillo are with the Laboratory LAFMIA UMI 3175, CINVESTAV - CONACYT - CNRS, Mexico. (castillo@hds.utc.fr)

[3]P. Garcia is with the ISA Department, Universitat Politècnica de València, E-46022 Valencia, Spain (pggil@isa.upv.es)

For some applications where the UAVs must often be able to hold a quasi-stationary flights independent of the atmospheric conditions, the Vertical Take-off Landing (VTOL) vehicles are more suitable. Many control algorithms to stabilize VTOLs aircraft are designed based on classical techniques that vary from conventional PD (Proportional-Derivative) control to more advances techniques depending on the nature of the problem [1], [2]. On the other hand, the development of quadrotors has generated great interest in the control community in the last few decades. This vehicle is based on a VTOL concept and it is extensively used to develop control laws. The particular interest of the research community for the quadrotor design can be linked to two main advantages over comparable VTOL UAVs, such as helicopters. First, quadrotors do not require complex mechanical control linkages for rotor actuation. This simplifies both the design and maintenance of the vehicle. Secondly, the use of four rotors ensures that individual rotors are smaller in diameter. The quadrotor configuration will therefore be easier to handle and possesses a higher maneuverability. Several control laws (linear, nonlinear, robust, etc) have been proposed in the literature to stabilize the quadcopter. Many of them were developed using a simplified nonlinear system or a linear model. Other works (including ourselves) have obtained the control algorithms representing the dynamics of the quadcopter by integrators in cascade.

The stabilization of chains of integrators have been extensively studied in the last decades. Relevant results are general, however, large amount of the studies on low amplitude designs is typically based on small gain considerations. They also require full state feedback, and in some cases only semiglobal results are provided [3] - [7]. From a structural point of view, for systems described by a generalized linear [4] - [9] or nonlinear [10], [11] chain of integrators, the control laws consist of a generalization of the nested saturations scheme of [9] or linear combinations of saturations [8]. These designs also make use of passivity, in the sense that, at each step of the procedure, the feedback consists in a function of the state for which it is a relative degree one output. See also [12], where Teel's nested saturation scheme is robustified against unmodeled dynamics.

However, the tendency of the increasing maneuverability, the nonlinearities and the unpredictable changes in the environment necessitate more sophisticated control systems. These issues have been studied in the context of linear control and more recently using nonlinear techniques [4]-[18].

978-1-4799-0817-2/13/$31.00 ©2013 IEEE

Even though there are so many modern control techniques that can be adopted for autopilot application, simple control algorithms remain a very interesting and effective approach for autopilot designs. Recently, in [19] we have proposed a control algorithm based on separated saturation functions. In this kind of controller, the saturation arguments generally depend on several state variables. We have compared and showed in simulation that the separated saturation controller in general is more efficient than Teel's results [9]. In addition, the gains and bounds of Teel's controller are relatively complex to tune. [20] and [21] present the stabilization problem for a similar class of nonlinear feedforward systems using bounded control. On the other hand, the main problem in this kind of controllers is the slow time of convergence, this fact is observed in real-time applications. The algorithm is excellent for hover applications, however, for navigation purposes is not so suitable.

The motivation of this work is to extend and improve the controller proposed in [19] to realize autonomous navigation reducing the time of convergence and validate it in flight tests. In addition, a new position estimation algorithm is also developed to estimate the position in the horizontal plane ($x$,$y$). This algorithm is proposed in order to validate in real time the control law.

The outline of the paper is the following: a background containing the equations of motion is given in Section II. The control strategy is developed in section III. The prototype is described in section IV. The estimation algorithm is presented in section V. The results from flight tests are reported in Section VI. And finally, conclusion and future work are discussed in section VII.

## II. Background: Equations of Motion

The quadrotor is a useful prototype to learn about aerodynamic phenomena in flying machines which can hover. This helicopter does not have a swashplate and has constant pitch blades. These vehicles have four electric motors that are controlled by varying its angular speed. In addition, the front and the rear motors rotate counterclockwise, while the other two rotate clockwise.

The dynamic models described in [22]-[27] have widely inspired the following dynamic equations for the quadrotor:

$$m_{X4}\dot{\upsilon} = R\left(\sum_{i=1}^{4} f_i + f_{d_k}\right) + f_g \tag{1a}$$

$$M(\eta)\ddot{\eta} = -C(\eta,\dot{\eta})\dot{\eta} + \sum_{i=1}^{4}(\tau_{M_i} + \tau_{r_i}) \tag{1b}$$

where $m_{X4}$ defines the mass of the vehicle, $\upsilon$ denotes the body speed relative to inertial frame $\mathscr{I}$, $f_g = -m_{X4}g\hat{\mathbf{k}}$ defines the gravitational force applied to the vehicle, $g$ denotes the acceleration due to gravity and $\hat{\mathbf{k}}$ represents the unit vector codirectional with the $z$-axis, $R$ is the rotation matrix,$f_{d_k}$ describes the vector of the drag forces in the body frame $\mathscr{B}$ and without loss of generality they can be approximated as

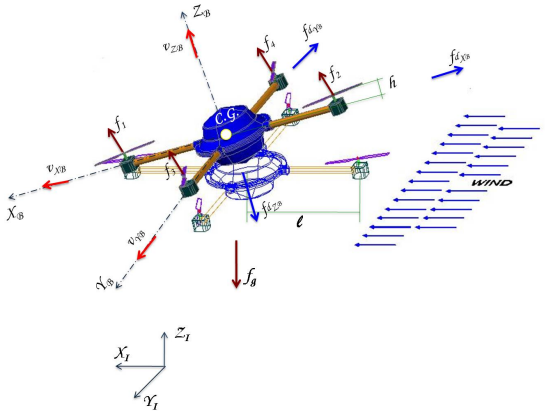


Fig. 1. The quadrotor in presence of lateral wind.

$$f_{d_k} \approx C_{d_k}\rho A_k(v_{w_k} - v_k)^2 \qquad k : X_B, Y_B, Z_B$$

where $C_{d_k}$ are the quadrotor drag coefficients, $A_k$ means the contact frontal area in the $k$-axis, $v_{w_k}$ defines the components of the wind velocity in $\mathscr{B}$ and $v_k$ represents the velocity of the quadrotor center of mass in the $k$-axis.

On the other hand, a rotor in translational flight undergoes an effect known as blade flapping. This aerodynamic effect causes an imbalance in lift, inducing an up-and-down oscillation of the rotor blades. Then, sometimes the rotor plane is not aligned with the $X_B, Y_B$ plane of translation. Then $f_i$ is defined as

$$f_i = f_{M_i}\begin{pmatrix} -\sin(a_{1s_i}) \\ \cos(a_{1s_i})\sin(b_{1s_i}) \\ \cos(b_{1s_i})\cos(a_{1s_i}) \end{pmatrix}$$

where $f_{M_i}$ is the force produced by motor $i$, $a_{1s_i}$ and $b_{1s_i}$ describe the longitudinal and lateral harmonic flapping angles[1] of the rotor $i$.

From the rotational dynamics (equation (1b)); $\tau_{r_i}$ is the vector moments produced by the rotor flapping and defined as

$$\tau_{r_i} = k_\beta a_{1s_i} + \mathbf{r}_i \times f_i \tag{2}$$

where $k_\beta$ is the stiffness of the rotor blade and $\mathbf{r}_i$ denotes the vector from the C.G. to each rotor [28]. It is defined as

$$r_1 = \begin{pmatrix} 0 & \ell & h \end{pmatrix} \qquad r_3 = \begin{pmatrix} \ell & 0 & h \end{pmatrix}$$
$$r_2 = \begin{pmatrix} 0 & -\ell & h \end{pmatrix} \qquad r_4 = \begin{pmatrix} -\ell & 0 & h \end{pmatrix}$$

Due to the quadrotor bilateral symmetries, moments generated by lateral deflections of the rotor plane get cancelled. Therefore, the only flapping moment is created by the backward tilt of the rotor plane through a deflection angle $a_{1s_i}$ with the longitudinal thrust. In addition typically, the physical stiffness of a rotor is ignored in flyer analysis and the rotor stiffness is modelled purely as a centrifugal term. As a result (2) can be substantially simplified.

[1]The full analysis of blade flapping is beyond the scope of this paper, but is presented in more detail in [23], [24].

Likewise from (1b), $M(\eta)$ describes the inertia matrix for the full rotational kinetic energy of the vehicle expressed in terms of the generalized coordinates $\eta$, $C(\eta,\dot{\eta})$ represents the Coriolis matrix, $\eta = (\psi,\theta,\phi)$ defines the orientation vector of the vehicle and $\tau_{M_i}$ represents the torque produced by motor $i$. In order to preserve the sign of rotation for counter-rotating rotors, $\tau_{M_i}$ can described as

$$\tau_{M_i} = C_{Q_i}\rho A_p r^3 \omega_i |\omega_i| \hat{\mathbf{k}}$$

where $r$ represents the rotor radius, $\omega_i$ describes the rotor speed, $\rho$ denotes the density of air and $A_p$ defines the propeller disk area. $C_{Q_i}$ is the dimensionless rotor torque coefficient. These coefficient can be obtained using the blade element theory [22]-[24].

## III. Control algorithm

When the control algorithm is developed to control or navigate this flying vehicle, the complete nonlinear dynamic model is always simplified or linearized. In the last years, the linearized model of this vehicle is commonly represented by a chain of integrators in cascade. Although the algorithms are obtained using the linear model, these are validated, in the most cases, with the nonlinear model and in other cases implemented and tested in flight tests.

*Theorem 1 ([25]):* Let consider a system with the form

$$\begin{aligned} \dot{r}_1 &= r_2 \\ &\vdots \\ \dot{r}_n &= u_r \end{aligned} \tag{3}$$

and the saturation function

$$\sigma_{b_i}(s) = \begin{cases} -b_i & \text{for} \quad s < -b_i \\ s & \text{for} \quad -b_i \leq s \leq b_i \\ b_i & \text{for} \quad s > b_i \end{cases}$$

with $b_i > 0$ is constant. Then the following control law

$$u_r = -\sum_{i=1}^{n} \sigma_{b_i}(\bar{K}_i r_i) \tag{4}$$

stabilizes the system (3) for all $\bar{K}_i > 0$ constant and bounds every state $r_i$ using the nonlinearity $\sigma_{b_i}$.

Rewriting (4) for the quadrotor system we obtain

$$\bar{u} = -\sigma_{b_{2_z}}\left(\bar{K}_{2_z}\dot{z}\right) - \sigma_{b_{1_z}}\left(\bar{K}_{1_z}(z-z_d)\right) \tag{5a}$$

$$\tau_\psi = -\sigma_{b_{2_\psi}}\left(\bar{K}_{2_\psi}\dot{\psi}\right) - \sigma_{b_{1_\psi}}\left(\bar{K}_{1_\psi}(\psi-\psi_d)\right) \tag{5b}$$

$$\begin{aligned} \tau_\theta = &-\sigma_{b_{4_\theta}}\left(\bar{K}_{4_\theta}\dot{\theta}\right) - \sigma_{b_{3_\theta}}\left(\bar{K}_{3_\theta}(\theta-\theta_d)\right) \\ &+ \sigma_{b_{2_\theta}}\left(\bar{K}_{2_\theta}\dot{x}\right) + \sigma_{b_{1_\theta}}\left(\bar{K}_{1_\theta}(x-x_d)\right) \end{aligned} \tag{5c}$$

$$\begin{aligned} \tau_\phi = &-\sigma_{b_{4_\phi}}\left(\bar{K}_{4_\phi}\dot{\phi}\right) - \sigma_{b_{3_\phi}}\left(\bar{K}_{3_\phi}(\phi-\phi_d)\right) \\ &- \sigma_{b_{2_\phi}}\left(\bar{K}_{2_\phi}\dot{y}\right) - \sigma_{b_{1_\phi}}\left(\bar{K}_{1_\phi}(y-y_d)\right) \end{aligned} \tag{5d}$$

where $x$, $y$, and $z$ denote the position of the quadrotor, $\psi,\theta$, and $\phi$ are the yaw, pitch and roll angles respectively. The control inputs are described by $u = \sum f_i$, $\tau_\theta$, $\tau_\phi$ and $\tau_\psi$ that represent the main thrust and the pitching, the rolling and the yawing moment, respectively. In addition, $\bar{u} = u - g$ and $z_d$, $x_d$, $y_d$, $\psi_d$, $\theta_d$ and $\phi_d$ are the desired values for each state. It has been proved in [25] that the previous control law stabilize the quadrotor system (1).

Notice that (5) stabilizes the quadrotor even if the desired position is far. The main practical constraint is when realizing flight tests, the quadrotor moves so slowly (due to the bounds). Observe also that, when using the previous control strategy the states of the quadrotor go to the origin ($r_i \to 0$), or in other cases, to a constant desired value ($r_i \to r_{i_d}$). Some research teams have worked with (or with similar) control schemes proposing some improvements to the algorithm in order to ameliorate the convergence; someone add gains outside the saturation function while others prefers adaptive saturation functions. Nevertheless, the stability analysis has not been carried out.

In order to improve the convergence of the states to the desired positions preserving the bounds inputs, we have linked the desired angular values with the position errors, i.e.

$$\theta_d = \sigma_{\theta_d}(k_{\theta_d}(\hat{x} - x_d)) \tag{6}$$

$$\phi_d = \sigma_{\phi_d}(k_{\phi_d}(\hat{y} - y_d)) \tag{7}$$

where $|\theta_d| \leq b_{\theta_d}$ and $|\phi_d| \leq b_{\phi_d}$, $k_i > 0$ is a constant gain to improve the convergence and $b_i > 0$ with $i : \theta_d, \phi_d$ the maximum desired angle. Remark that when using previous equations the convergence is realized very quickly keeping the states bounded and the stability of the closed-loop system.

For paper length the stability analysis will be given only for the lateral and longitudinal dynamics (four integrators in cascade).

*Theorem 2:* Consider the system (3) with $n = 4$, then, the following control law

$$u_r = -\sigma_{b_4}(\zeta_4) - \sigma_{b_3}(\zeta_3) - \sigma_{b_2}(\zeta_2) - \sigma_{b_1}(\zeta_1) \tag{8}$$

with

$$\begin{aligned} \zeta_4 &= \bar{K}_4 r_4 \\ \zeta_3 &= \bar{K}_3(r_3 - r_{d_3}); \quad r_{d_3} = f(\zeta_1) \ \text{ with } |r_{d_3}| \leqslant b_{d_3} \\ \zeta_2 &= \bar{K}_2 r_2 \\ \zeta_1 &= \bar{K}_1(r_1 - r_{d_1}); \quad r_{d_1} = \text{constant} \end{aligned}$$

makes the closed-loop system stable and improves the convergence of the states.

**Remark:** In Theorem 2, $r_{d_3} = f(r_1, r_{d_1})$ is a bounded function and is related directly with the first state.

*Proof:* To simplify the analysis, a recursive methodology is proposed. Let us assume that

$$\begin{aligned} \xi_i &= \sigma_{b_i}(\zeta_i) + \xi_{i-1} \text{ and } |\xi_i| \leqslant b_{\xi_i} \ i : n, ..., 2 \\ \xi_1 &= \sigma_{b_1}(\zeta_1) \end{aligned}$$

and

$$\dot{\zeta}_n = \bar{K}_n u_r = -\bar{K}_n \xi_n \tag{9}$$

Define the following positive definite function

$$V_n = \frac{1}{2}\zeta_n^2$$

whose derivative with respect to time is

$$\dot{V}_n = -\bar{K}_n\zeta_n\left(\sigma_{b_n}(\zeta_n)+\xi_{n-1}\right)$$

Proposing $b_n > b_{\xi_{n-1}}$ and using definition of the saturation function, this signifies that $|\zeta_n| > |\xi_{n-1}|$ and implies that $\dot{V}_n < 0$. This means that, $\exists$ a time $T_n$, such that, $|\zeta_n| \leqslant b_{\xi_{n-1}}$ and $u_r = -\zeta_n - \sigma_{b_{n-1}}(\zeta_{n-1}) - \ldots - \sigma_{b_1}(\zeta_1),\ \forall\ t > T_n$.

Define $\beta_{n-1} = \frac{\zeta_n}{\bar{K}_n} + \frac{\bar{K}_n\zeta_{n-1}}{\bar{K}_{n-1}}$, thus, $\dot{\beta}_{n-1} = -\xi_{n-1}$. Similarly that for $n$, the following positive definite function is proposed $V_{n-1} = \frac{1}{2}\beta_{n-1}^2$. Differentiating with respect to time, we have

$$\begin{aligned}\dot{V}_{n-1} &= -\left(\frac{\zeta_n}{\bar{K}_n} + \frac{\bar{K}_n\zeta_{n-1}}{\bar{K}_{n-1}}\right)\left(\sigma_{n-1}(\zeta_{n-1})+\xi_{n-2}\right)\\ &\leqslant -\left(\kappa_{n-1} + \frac{\bar{K}_n\zeta_{n-1}}{\bar{K}_{n-1}}\right)\left(\sigma_{n-1}(\zeta_{n-1})+\xi_{n-2}\right)\quad \forall\ t > T_n\end{aligned}$$

with $\kappa_{n-1} = \frac{b_{\xi_{n-1}}}{\bar{K}_n}$. Notice that $b_{\xi_{n-1}} = b_{n-1} + b_{\xi_{n-2}}$. Propose $\bar{K}_n^2 > \bar{K}_{n-1}$ then $sign(\kappa_{n-1} + \frac{\bar{K}_n\zeta_{n-1}}{\bar{K}_{n-1}}) = sign(\zeta_{n-1})$. Define $b_{n-1} \geqslant b_{\xi_{n-2}}$, therefore the previous inequalities implies that $\dot{V}_{n-1} \leqslant 0$.

The base case of the recursion occurs when the $V_1$ function is analyzed and this case will be treated a little different. Propose $V_1 = \frac{1}{2}\beta_1^2$; with $\beta_1 \leqslant \kappa_1 + \frac{\bar{K}_n\zeta_1}{\bar{K}_{n-1}}$ and $\dot{\beta}_1 = -\xi_1$. Thus, its derivative with respect to time is

$$\dot{V}_1 \leqslant -\left(\kappa_1 + \frac{\bar{K}_n\zeta_1}{\bar{K}_{n-1}}\right)\sigma_1(\zeta_1)$$

then if $|\frac{\bar{K}_n\zeta_1}{\bar{K}_{n-1}}| > \kappa_1$ then $\dot{V}_1 \leqslant 0$. Then $b_1$ need to be chosen to satisfy the previous inequality.

Observe from the previous analysis that all the states go to zero (or a constant desired value). We take the case when the state goes to a desired function (it is the case for the $r_3$). For this case, $r_{d_3}$ is defined as a bounded function of the first state, i.e.

$$r_{d_3} = \sigma_{r_{d_3}}(k_{r_{d_3}}(r_1 - r_{d_1})) \tag{10}$$

with $k_{r_{d_3}}$ is a constant gain. From the definition of the saturation function this implies that $r_{d_3}$ is bounded and from its construction is also decreasing, i.e. $|r_{d_3}| \leqslant b_{r_{d_3}}$ and $r_{d_3} \to 0$ when $(r_1 - r_{d_1}) \to 0$. Observe from (10) that $r_{d_3}$ has a maximum value, thus, if $b_{r_{d_3}} = \max|r_{d_3}|$ is chosen for the analysis of convergence then, the previous analysis yields. ■

## IV. Experimental Platform

The proposed controller has been validated in flight tests. Our experimental platform consists of a quadrotor flying vehicle, an embedded navigation system and a ground station, see Figure 2. The vehicle is based on a Mikrokopter structure and has four brushless motors controlled by drivers i2c BlCtrl. The total mass of the prototype is $1100g$. A battery LiPO $11.1V$ 6000 $mAh$ is employed to energize all the electronic systems. The electronic onboard is equipped with an IGEPv2 board based on the Texas Instruments system on Chip OMAP3530 with an ARM CortexA8 processor running at 720 Mhz and a Digital Signal Processor (DSP) C64x+ running at 520Mhz. In addition, the quadrotor is equipped with an IMU 3DMGX3-25 to measure the vehicle attitude, an ultrasonic range finder SRF10 for the altitude measurement and a PS3eye camera used to compute the optical flow. The electronic board, the sensors and the control algorithms compose the embedded navigation system.

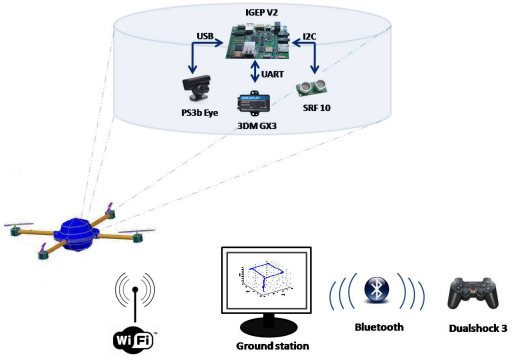


Fig. 2. Experimental Platform

All the information collected by the microprocessor is sent to a ground station using a Wi-Fi connection. Its objective is to graph the states to supervise the system responses, to tune the control parameters and to redefine tasks or missions, all in real time. The base station is written with the QT library, making it a multi-platform. For flight manual phases a Playstation 3 joystick is used. The ground station and the joystick are connected using bluetooth communication.

Remark that only some states can be measured $(z, \psi, \theta, \phi, \dot{\psi}, \dot{\theta}, \dot{\phi}, \dot{x}, \dot{y})$ with the sensors of the vehicle. Observe also that to apply the controller is necessary all the states, and in our case we do not have a sensor to measure the vertical velocity $(\dot{z})$, and the planar position $(x, y)$. Therefore, these states need to be estimated to realize hover, tracking trajectory or navigation missions with this prototype.

## V. Estimation Algorithm using an EKF

Fusing data from different sensors improves the performance of the overall sensing system. For example for aerial navigation outdoors, fusion of GPS (Global Position System) with INS (Inertial Navigation Systems) outputs by means of filtering techniques increases the localization precision required by UAV missions. Some works about indoor position control for flying objects based on localization systems or Simultaneous Localization and Mapping (SLAM) have been already reported in the literature. However, the most of them either depend on a highly accurate and expensive localization system that needs to be deployed manually [29], [30] or they need a high amount of (mostly off-board) computing processing [31]. Other works fuse information from two or more sensors. For example; in [32], the authors use a low-cost video system with a Kalman Filter (KF) for data-fusion algorithm that includes measures from inertial sensors. In addition, vision algorithms + KF with on board computation have been proposed using landmarks [33], [34].

Similar works but with off-board computation have been also presented in the literature [35], [36].

Notice that, the techniques for data fusion are numerous and most of them are mainly based on the fusion of the position (obtained by GPS, laser or vision) and inertial sensors, with the objective to estimate the translational velocities. Remark that for our platform these approaches can not been used because our prototype is not equipped with a position sensor. Thus, other challenge that we are addressing in this paper is the development of a 3D indoor localization system that can be an efficient and viable solution for UAV applications with denied position measurement. Our originality lies in estimating, with a good precision, onboard and in real time the *xy*-position based mainly in the translational velocities. In our best knowledge, this result is novel and it can be extended to outdoor applications. Similar works are based in dead reckoning, however, dead reckoning is subject to cumulative errors.

Therefore in this part and in order to improve the dead-reckoning technique, a position estimation algorithm based on the Extended Kalman Filter (EKF) is presented.The algorithm fuses the measures coming from a camera onboard, an ultrasound and an IMU to estimate onboard and in real time the quadrotor relative position $(\hat{x},\hat{y})$ and the vertical velocity, $\hat{\dot{z}}$. Another characteristic of this work is that all the image processing is computed on board simultaneously with the control law.

### A. The EKF

The Kalman Filter is acknowledged as the most popular estimation algorithm probably because it is easy to implement (linear system). When either the system state dynamics or the observation dynamics are nonlinear, the conditional probability density functions that provide the minimum mean-square estimate are no longer Gaussian and its evaluation represents an high computational burden. A non optimal approach to solve the problem is the EKF. The EKF algorithm is basically composed of a prediction step, which involves developing a state and the covariance estimate of the next time step based on the current estimatation and system dynamics model; and an update step or filtered cycle, where the new measurement processing and the prediction update are computed using the new information.

Let consider the following nonlinear dynamics with external inputs

$$\begin{aligned}\bar{\mathbf{x}}_{k+1} &= \mathbf{f}(\bar{\mathbf{x}}(k),\mathbf{u}(k))+\alpha(k)\\ \mathbf{y}_k &= \mathbf{g}(\bar{\mathbf{x}}_k)+\beta(k)\end{aligned}$$

where $\bar{\mathbf{x}}_{k+1}$ represents the states vector, $\mathbf{f}(\bar{\mathbf{x}}(k),\mathbf{u}(k))$ denotes the nonlinear dynamics of the vehicle, $\mathbf{y}_k$ signifies the output equation, $\mathbf{g}(\bar{\mathbf{x}}_k)$ is the desired vector output. Likewise $\alpha(k)$ and $\beta(k)$ are assumed to be Gaussian noises with covariance matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$ respectively. Then, the equations for the EKF are given by

$$\hat{\bar{\mathbf{x}}}_{k+1|k} = \mathbf{f}(\hat{\bar{\mathbf{x}}}_{k|k},\mathbf{u}_{k|k}) \tag{11a}$$
$$\hat{\mathbf{y}}_{k+1|k} = \mathbf{g}(\bar{\mathbf{x}}_{k+1|k}) \tag{11b}$$
$$\mathbf{P}_{k+1|k} = \mathbf{A}\mathbf{P}_{k|k}\mathbf{A}^T+\mathbf{Q}_k \tag{11c}$$
$$\mathbf{K}_{k+1|k} = \mathbf{P}_{k+1|k}\mathbf{G}^T(\mathbf{G}\mathbf{P}_{k+1|k}\mathbf{G}^T+\mathbf{R}_{k+1})^{-1} \tag{11d}$$
$$\hat{\bar{\mathbf{x}}}_{k+1|k+1} = \hat{\bar{\mathbf{x}}}_{k+1|k}+\mathbf{K}_{k+1}(\mathbf{y}_{k+1}-\hat{\mathbf{y}}_{k+1|k}) \tag{11e}$$
$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k}-\mathbf{K}_{k+1|k}\mathbf{G}_{k+1}+\mathbf{P}_{k+1|k} \tag{11f}$$

where $\mathbf{A}=\left[\frac{\partial \mathbf{f}}{\partial \bar{\mathbf{x}}}(\hat{\bar{\mathbf{x}}},\mathbf{u})\right]$ and $\mathbf{G}=\left[\frac{\partial \mathbf{g}}{\partial \bar{\mathbf{x}}}(\hat{\bar{\mathbf{x}}})\right]$. Similarly, $\mathbf{K}_{k+1|k}$ defines the gain filter and the initial state $\hat{\bar{\mathbf{x}}}(0)$ and the initial covariance $\mathbf{P}(0)$ are assumed to be known.

Remark that contrary to the Kalman Filter, the EKF may diverge, if the consecutive linearizations are not a good approximation of the linear model in all the associated uncertainty domain.

### B. Model adaption

Usually the quadrotor operates in a attitude range within $\pm 30^\circ$ and therefore, the equations of motion are approximately decoupled about each attitude axis. Its dominant dynamics are associated with the longitudinal/lateral dynamics of the vehicle. For quasi-stationary maneuvers the flapping angles are so small and then, the longitudinal and lateral rotor thrust can be neglected. On the other hand, usually the drag force is neglected in computing the drag moment [22], [24]. This force was found to cause a negligible disturbance on the total moment over the flight regime of interest, relative to blade flapping torques.

In recent times, a simplified nonlinear model - taken from (1) - has increased its popularity in the UAVs and control communities [26], [37], [38]. It is because this model retains the main features that must be considered when designing control algorithms for flights tests. The discrete-time form, considering a sampling period $T_s$ small enough, of this simplified nonlinear model is the following:

$$\bar{\mathbf{x}}_{k+1} = \begin{bmatrix} \dot{x}_k T_s + x_k \\ -(\frac{u_k}{m_{X_4}}\sin(\theta_k))T_s + \dot{x}_k \\ \dot{y}_k T_s + y_k \\ (\frac{u_k}{m_{X_4}}\cos(\theta_k)\sin(\phi_k))T_s + \dot{y}_k \\ \dot{z_k} T_s + z_k \\ (\frac{u_k}{m_{X_4}}\cos(\theta_k)\cos(\phi_k) - g)T_s + \dot{z}_k \\ \dot{\theta}_k T_s + \theta_k \\ \frac{\tau_{\theta_k}}{M_x}T_s + \dot{\theta}_k \\ \dot{\phi}_k T_s + \phi_k \\ \frac{\tau_{\phi_k}}{M_y}T_s + \dot{\phi}_k \\ \dot{\psi}_k T_s + \psi_k \\ \frac{\tau_{\psi_k}}{M_z}T_s + \dot{\psi}_k \end{bmatrix} \tag{12}$$

where $M_j$ with $j:x,y,z$, represents the constant inertia matrix in the $j$-axis and $T_s = t_{k+1}-t_k$; with $t_k$ is the time $t$ at instant $k$.

When the EKF filter is implemented numerically the most common problem is to choose the covariance matrices. If

these matrices are not well conditioned in some cases the filter diverges. However from our experience, the system matrices, **A** and **G**, need to be adjusted off-line or on-line to improve the performance. This fact could be obvious if observing equation (12). Notice from this equation that some constant parameters need to be considered for each vehicle (mass, inertia matrix, etc.). A fatal and common error, when using estimation algorithms, is to "normalize" these variables.

Our numerical implementation is given in three parts. Firstly, equations (11) and (12) are executed and validated off-line in Matlab code. This was a crucial part where the system matrices were re-adjusted when using real data collected by the ground station. In addition, a small identification part was carried out to describe the altitude dynamic. On the other hand, the state transition and measurement noises have been generated from Gaussian distributions. The second part involved the Kalman algorithm implementation in the microprocessor of the helicopter in C language. The goal in this part was to validate on-board, on-line and in open-loop system the algorithm. The most difficult part here was to compute the matrices operations. In this way, several flight tests were carried out in manual form. The behavior of the aerial vehicle was depicted in the ground station to be analyzed on-line, nevertheless some small adjustments were realized to improve the convergence of the EKF. In the last part, the EKF and the control algorithm were validated in closed loop.

### C. Optical flow - OF

As previously discussed, one contribution of this work lies in the estimation of the horizontal position $(x,y)$ using the translational velocities (dead-reckoning technique). This methodology is improved using an EKF and the estimation is corrected with the angle and angular rate measurements coming from inertial sensors. The translational velocities are derived from the optical flow. However, the heart of this estimation come from the computation of the OF. In the following paragraphs, a brief description and the main equations to calculate the OF are described.

It is well known that OF is used to calculate the motion between two images frames. The techniques to determine the optical flow can be categorized as differential methods, correlation, energy and frequency based methods. In this work, the Lucas and Kanade algorithm is employed. This algorithm is a differential method that takes the advantage of spatiotemporal derivatives of image sequences [39].

Let us define $I(x_{im},y_{im},t)$ as the grayscale density with the form

$$I(x_{im},y_{im},t) = I(x_{im}+\partial x_{im}, y_{im}+\partial y_{im}, t+\partial t)$$

where $I$ denotes the intensity and $(x_{im},y_{im})$ represents the position of a point in the image. Here after, we will consider that $I = I(x_{im},y_{im},t)$, $I_x = \frac{\partial I(x_{im},y_{im},t)}{\partial x_{im}}$, $I_y = \frac{\partial I(x_{im},y_{im},t)}{\partial y_{im}}$, $I_t = \frac{\partial I(x_{im},y_{im},t)}{\partial t}$. Then, using the Taylor series

$$I_x v_{x_{im}} + I_y v_{y_{im}} + I_t = 0$$

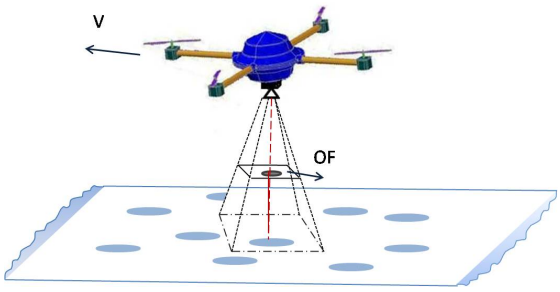


Fig. 3. Optical flow scheme

with $v_{x_{im}}, v_{y_{im}}$ define the $x_{im}, y_{im}$ components of the optical flow. Define $\Delta I = [I_x, I_y]^T$ and $V_{OF} = [v_{x_{im}}, v_{y_{im}}]^T$, thus it follows that

$$\Delta I \cdot V_{OF} + I_t = 0 \tag{13}$$

Observe from previous equation that there are two unknowns and only one equation, therefore other condition is necessary to find a solution. The algorithm of Lucas and Kanade also suggests that the optical flow $(v_{x_{im}}, v_{y_{im}})$ is constant over a neighborhood (a window of $p \times p$, with $p > 1$) centered on the pixel that we want to calculate the displacement. Then from the previous, it can be computed $p^2$ equations with form

$$V = (A^T A)^{-1} A^T b \tag{14}$$

where

$$A = \begin{bmatrix} I_{x_1} & I_{y_1} \\ I_{x_2} & I_{y_2} \\ \vdots & \vdots \\ I_{x_n} & I_{y_n} \end{bmatrix}, \qquad b = \begin{bmatrix} -I_{t_1} \\ -I_{t_2} \\ \vdots \\ -I_{t_n} \end{bmatrix}$$

and $n = p^2$. And finally using (13) and (14) the optical flow $V_{OF}$ can be computed.

It has been demonstrated in several works that if a camera is fixed in a vehicle in a way that they share the same movements, the OF computed by the camera is directly related with the translational velocity of the vehicle with the following relation (see Figure 3)

$$V_{OF_x} = -\frac{F\dot{x}}{z}$$
$$V_{OF_y} = -\frac{F\dot{y}}{z}$$

where $\dot{x}$ and $\dot{y}$ are the velocity vector of the vehicle in the plane $x-y$, $z$ represents the altitude and $F$ defines the focal distance. From previous equation, it is easy to compute the translational velocities.

## VI. Results from flight tests

Several flight tests were carried out to validate the observer-control scheme. All the algorithms are computed on-board, on-line with two cases: in manual and autonomous mode. For each mode, we performed several flights following

different trajectories; a line in the $x$ or $y$-axis, a diagonal line in the $x-y$ plane, a square, etc. All the estimated positions were manually verified, i.e., we placed in the horizontal workspace (ground) landmarks each one meter.

### A. Manual flight tests

Before realizing autonomous control of the quadrotor vehicle, we conducted a number of flight tests in manual form in order to validate the EKF implementation and to verify the accurate of the estimation of the states.

For paper length, we present in the following figures the results for the rectangle trajectory with $\ell_1 = 5$m in the $\hat{x}$-axis and $\ell_2 = 6$m in the $\hat{y}$-axis. This result is presented in Figures 4–6. The $\hat{x}$ and $\hat{y}$ estimation is presented in Figure 4. In same figure we can observe small oscillations, this results is normal and it is due to the pilot flight skills.

In Figure 5 the $z$ and $\dot{\hat{z}}$ performances are introduced. In Figure 6 some variations in the altitude can be observed, these behavior are normal when an amateurish pilot flies the helicopter with a rectangle trajectory.

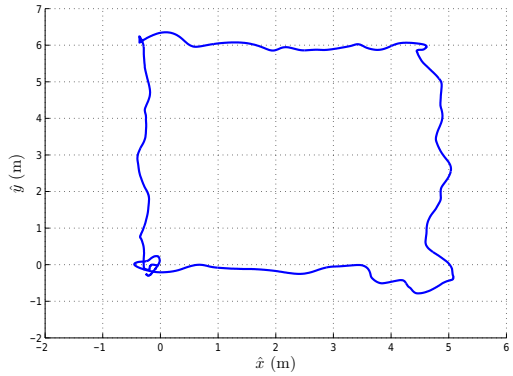


Fig. 4. $\hat{x},\hat{y}$ position estimation in manual mode and with rectangle trajectory.

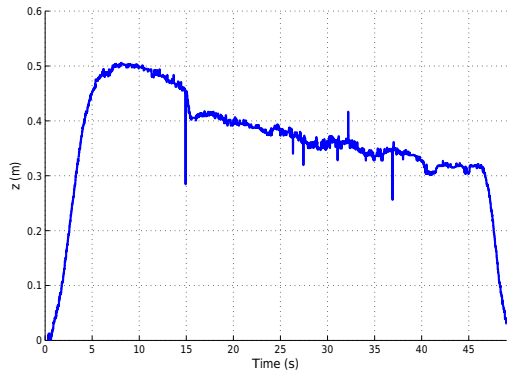


Fig. 5. Altitude $z$ performance in open-loop and with rectangle trajectory.

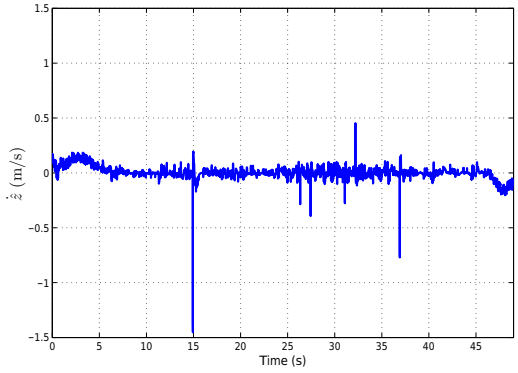


Fig. 6. Vertical velocity $\dot{\hat{z}}$ performance in open-loop and with rectangle trajectory.

### B. Autonomous mode

Since the experiments demonstrated that the algorithm maintains accurate state estimates, we proceeded to testing the algorithms and the control law in closed loop. We have carried out several flights tests in closed loop to validate the performance of the flying vehicle. Only two cases (square trajectory and user-desired path) are depicted in following graphs to illustrate these results.

One mission was to realize a square with 2 m in each side. Figures 7–11 introduce these results. In Figure 7 a 3D position response is shown when the aerial vehicle follows a trajectory with coordinates $(x_0,y_0,z_0) = (0,0,0.6)$, $(x_1,y_1,z_1) = (2,0,0.6)$, $(x_2,y_2,z_2) = (2,2,0.6)$ and $(x_3,y_3,z_3) = (0,2,0.6)$ all in meters. Observe that even if the flying vehicle is not equipped with a position sensor, the EKF is capable to estimate the $\hat{x}$ and $\hat{y}$ position with a excellent precision using essentially the translational velocities in this plane (see Figure 9) and the attitude of the vehicle.

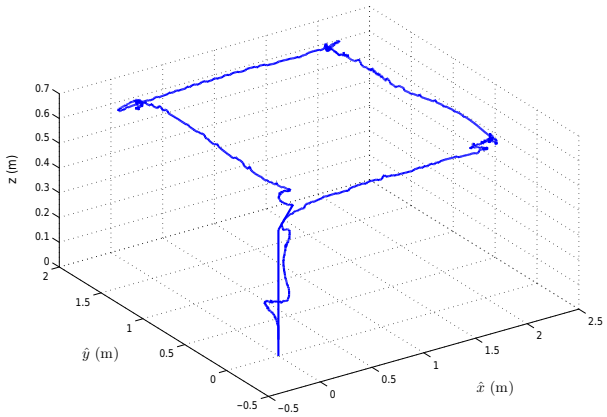


Fig. 7. $\hat{x},\hat{y},z$ responses when the flying vehicle realize a square trajectory autonomously.

In Figure 8 we introduce the performance of the vertical velocity. Notice from Figure 7 that no significant variations in the altitude are presented and consequently, its velocity response is small. In this experiment, the helicopter moves in $x$ and $y$ axis and it can be corroborated in Figure 9 where the horizontal velocities are depicted. In this figure, we can remark the four movements that the helicopter performs. These movements produce changes manly in the pitch and roll angles, see Figure 10.

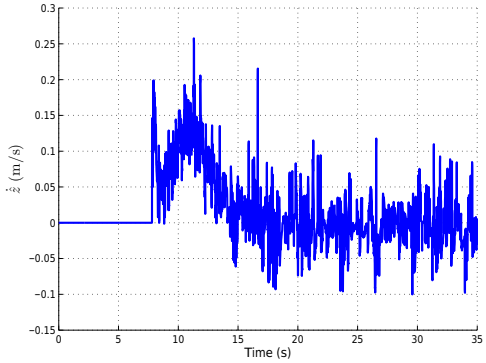


Fig. 8. $\dot{\hat{z}}$ estimation when the quadrotor follows a square autonomously

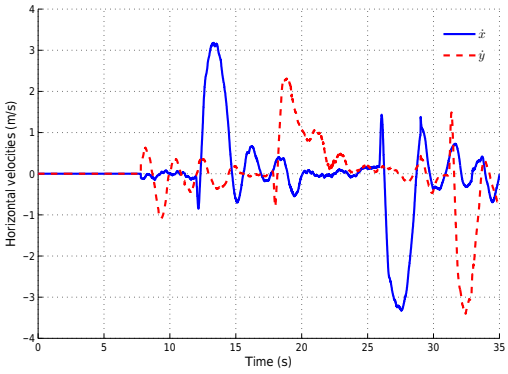


Fig. 9. $\dot{x}$ and $\dot{y}$ behavior obtained from OF technique.

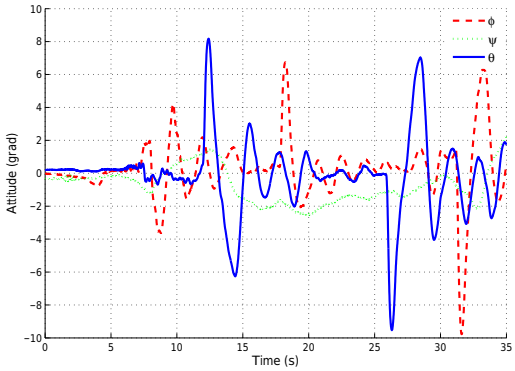


Fig. 10. Attitude response of the flying vehicle.

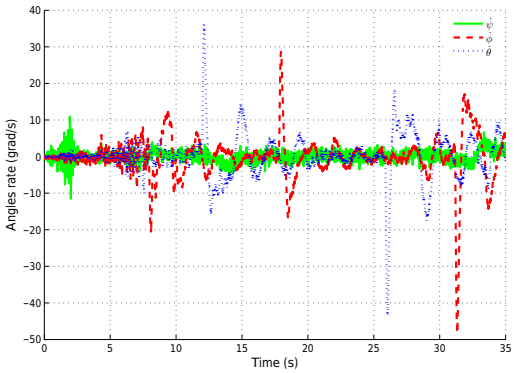


Fig. 11. Angles rate performance

These flight tests are also illustrated with some videos at: http://www.hds.utc.fr/~lmunozhe/dokuwiki/doku.php?id=en:videos or http://www.dailymotion.com/video/xtxrth_position-control-of-a-quadrotor-using-data-fusion_tech.

The previous experiments have proved the good performance and good precision of the proposed control-observer scheme. In order to prove the behavior in difficult trajectories and the no accumulative errors in the estimation we have realized some user-free paths (videos corresponding to these flight test can be found at: http://www.hds.utc.fr/~lmunozhe/dokuwiki/doku.php?id=en:videos or http://www.dailymotion.com/video/xu6db5_video-paths).

To overwhelm the mission, we place four obstacles like pillars. Remember that the vehicle is not equipped with others sensors to avoid obstacles, then, a bad estimation and a worse tuning of the control parameters will result in the crash of the helicopter against the columns. The objective of the mission was to follow a trajectory given by a set of desired coordinates. In this mission, the aerial vehicle cross the four pillars, turns one of them, goes to the center of the pillars and lands, all in autonomous mode, see Figure 12.

The desired coordinates were: $(x_0, y_0, z_0) = (0,0,0)$m, $(x_1, y_1, z_1) = (0,0,0.6)$m, $(x_2, y_2, z_2) = (2,0,0.6)$m, $(x_3, y_3, z_3) = (2,4,0.6)$m, $(x_4, y_4, z_4) = (4,4,0.6)$m, $(x_5, y_5, z_5) = (4,2,0.6)$m, $(x_6, y_6, z_6) = (2,2,0)$m, and

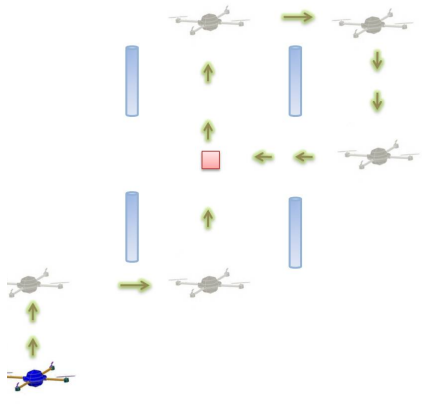

Fig. 12. The desired mission with obstacles.

$(x_f, y_f, z_f) = (2,2,0)$m. In Figures 13–17 we illustrate the states responses of this difficult mission.

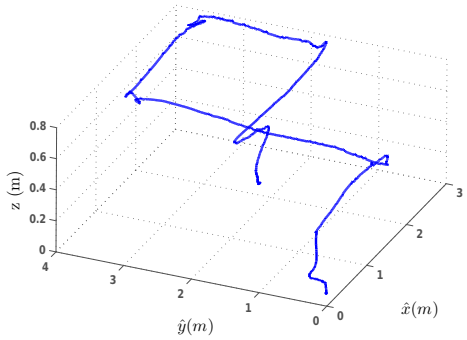


Fig. 13. $\hat{x}, \hat{y}, z$ response of the helicopter when it realizes the trajectory.

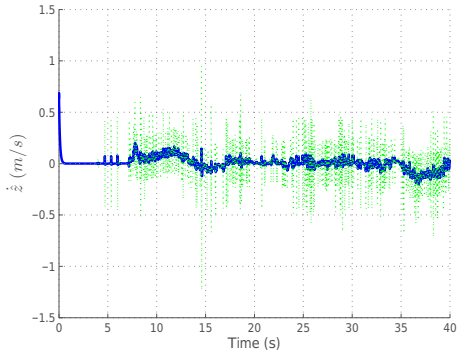


Fig. 14. $\dot{\hat{z}}$ response when using the EKF and a classical Euler derivation.

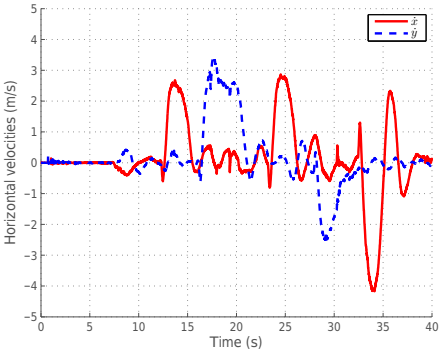


Fig. 15. $\dot{x}, \dot{y}$ response.

A 3D view of the $\hat{x}, \hat{y}, z$ response is introduced in Figure 13. Notice in this figure that the quadrotor realizes very well the desired mission. The $\dot{\hat{z}}$ performance is showed in Figure 14. Two performances of this state using different approaches are showed in this figure. In the figure the dashed line (green) represents the behavior when using the classical Euler derivation while the solid line (blue) illustrates the performance the EKF is used. Observe in this figure that the EKF has a better behavior than the classical approach. In Figures 6 and 8 we only present the $\dot{\hat{z}}$ behavior with the EKF. In closed-loop system the EKF approach was the only employed.

Note in Figure 12 that the quadrotor needs to realize the following displacements to complete the mission: first, in the positive longitudinal axis, secondly in the positive lateral axis, after that in the positive longitudinal axis, fourthly in the negative lateral axis and finally in the negative longitudinal axis. This can be corroborated in Figure 15. These movements imply changes in the pitch and roll angles and in the angular rate of the vehicle, see Figures 16 and 17.

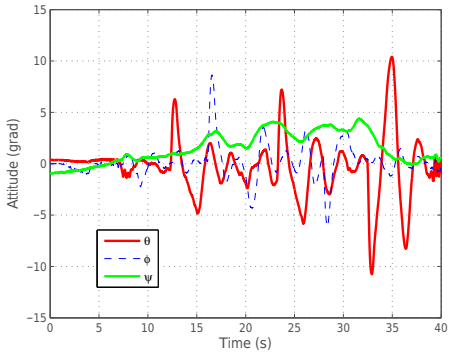


Fig. 16. Attitude behavior.

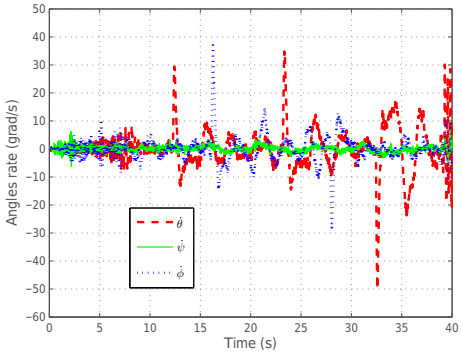


Fig. 17. Angular rate behavior.

## VII. Conclusion and Future Work

A navigation algorithm improvement has been presented in this paper. The control algorithm is simple, nonlinear and easy to implement and tuning in real time. The improvement in the control law is such that the desired values in the angles are in relation with the position error, with this, the convergence to the desired navigation values is faster. On the other hand, after several approaches adopted to estimate the position of an aerial vehicle using vision information, in this paper, we have developed this position estimation using an Extended Kalman Filter for a quadrotor. One characteristic of this algorithm is that it fuses only measures from an IMU, an ultrasonic sensor and a camera.

Some research teams use external measurement systems to locate the vehicle but the main drawback of these solutions appears when the aerial vehicle is not in the workspace of the external measurement system (VICON, GPS,...). In these cases, those systems become impractical or unusable. This methodology could be adopted to locate an aerial vehicle (plane or helicopter) in indoor our outdoor environments. Future work consists in realizing flight outdoor using others devices to measure the translational velocities and the altitude $(\dot{x}, \dot{y}, z)$. To improve the autonomous navigation the observer-control scheme will be fused with a laser sensor (Hokuyo) to avoid obstacle in unknown environments.

## REFERENCES


[1] Abdessameud, A. and Tayebi, A., "Global trayectory tracking control of VTOL-UAVs without linear velocity measurements", *Automatica*, Vol. 46, No. 6, June 2010, pp. 1053 – 1059. doi: 10.1016/j.automatica.2010.03.010

[2] Dierks, T. and Jagannathan, S., "Output Feedback Control of a Quadrotor UAV Using Neural Networks", *IEEE Transactions on Neural Networks*, Vol.21, No.1, Jan. 2010, pp. 50 – 66. doi: 10.1109/TNN.2009.2034145

[3] Garcia, G. and Tarbouriech, S., "Stabilization with eigenvalues placement of a norm bounded uncertain system by bounded inputs", *Int. J. Robust Nonlinear Control*, vol. 9, pp. 599-615, 1999.

[4] Lin, Z. and Saberi, A., "Semi-global exponential stabilization of linear systems subject to input saturation via linear feedbacks", *Systems & Control Letters*, 21, pp. 225-239, 1993.

[5] Saberi, A. and Han, J. and Stoorvogel A., "Constrained stablization problems for linear plants", *Automatica*, vol. 38, pp. 639-654, 2002.

[6] Stoorvogel, A. and Saberi, A., "Output regulation for linear plants with actuators subject to amplitude and rate constraints", *Int. J. Robust Nonlinear Control*, vol. 9, pp. 631-657, 1999.

[7] Sznaier, M. and Suarez, R. and Miani, S. and Alvarez-Ramirez, J., "Optimal disturbance attenuation and global stabilization of linear systems with bounded control", *Int. J. Robust Nonlinear Control*, vol. 9, pp. 659-675, 1999.

[8] Sussmann, H.J. and Sontag, E.D. and Yang, Y., "A general result on the stabilization of linear systems using bounded controls", *IEEE Transactions on Automatic Control*, 93(12) pp. 2411-2425, 1994.

[9] Teel, A.R., "Global stabilization and restricted tracking for multiple integrators with bounded controls", *Systems and Control Letters*, vol. 18, pp. 165-171, 1992.

[10] Marconi, L. and Isidori, A., " Robust global stabilization of a class of uncertain feedforward nonlinear systems", *Syst. Control Lett.*, vol. 41, no. 4, pp. 281-290, 2000.

[11] Grognard, F. and Sepulchre, R. and Bastin, G., " Global stabilization of feedforward systems with exponentially unstable Jacobian linearization", *Syst. Control Lett.*, vol. 37, no. 2, pp. 107-115, 1999.

[12] Arcak, M. and Teel, A. R. and Kokotovic, P. V., " Robust nonlinear control of feedforward systems with unmodeled dynamics", *Automatica*, vol. 37, pp. 265-272, 2001.

[13] Danapalasingam, K.A. and Leth, J-J. and La Cour-Harbo, A. and Bisgaard, M., "Robust helicopter stabilization in the face of wind disturbance", in Proceeding of IEEE Conference on Decision and Control (CDC), pp.3832–3837, Dec. 2010

[14] Yu-Qing, H. and Jian-Da, H., "Acceleration Feedback Enhanced $H_\infty$ Disturbance Attenuation Control for a Class of Nonlinear Underactuated Vehicle Systems", *Acta Automatica Sinica*, Vol. 34 No. 5, May 2008, pp. 558 – 564.

[15] Raffo, G. V. and Ortega, M. G. and Rubio, F. R., "An integral predictive/nonlinear $H_\infty$ control structure for a quadrotor helicoper", *Automatica*, Vol. 46, No. 1, Jan 2010, pp. 29 – 39.

[16] Alexis, K. and Nikolakopoulos, G. and Tzes, A., "Constrained optimal attitude control of a quadrotor helicopter subject to wind-gusts: Experimental studies", in Proceeding of American Control Conference (ACC), 2010 , vol., no., pp.4451-4455, June 30 2010-July 2, 2010.

[17] Das, A. and Subbarao, K. and Lewis, F., "Dynamic inversion with zero-dynamics stabilisation for quadrotor control", *Control Theory and Applications, IET*, vol.3, no.3, pp.303-314, March 2009.

[18] Besnard, L. Shtessel, Y.B. and Landrum, B., "Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer", J. Franklin Inst. (2011).

[19] Munoz, L. E. and Castillo, P. and Sanahuja, G. and Santos, O., "Embedded robust nonlinear control for a four-rotor rotorcraft: Validation in real-time with wind disturbances", in Proceeding of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), San Francisco, USA, October, 2011.

[20] Kaliora, G. and Astolfi, A., "Nonlinear control of feedforward systems with bounded signals", *IEEE Trans. Automatic Control*, 49(11), pp. 1975–1990, 2004.

[21] Kaliora, G. and Astolfi, A., "On the stabilization of feedforward systems with bounded control", *Systems and Control Letters*, vol. 54, pp. 263-270, 2005.

[22] Pounds, P. and Gresham, J. and Mahony, R. and Robert, J. and Corke, P., "Towards dynamically favourable quadrotor aerial robots", in Proceedings of the Australasian Conference on Robotics and Automation, Canberra, Australia, December 2004.

[23] Prouty, R. W., *Helicopter Performance, Stability, and Control*. Krieger Publishing Company, 2002, reprint with additions, original edition 1986.

[24] Hoffmann, G. and Huang, H. and Waslander, S. and Tomlin, C., "Quadrotor helicopter flight dynamics and control: Theory and Experiment", in Proceedings of the AIAA Guidane, Navigation, and Control Conference, August 2007.

[25] Sanahuja, G. and Castillo, P. and Sanchez, A. "Stabilization of $n$ integrators in cascade with bounded input with experimental application to a VTOL laboratory system", *International Journal of Robust and Nonlinear Control*, Published online 28 July 2009; Vol. 20, No.10, 2009, pp. 1129 – 1139.

[26] Lozano, R. *Unmanned Aerial Vehicles Embedded Control*, John Wiley-ISTE Ltd, 2010. ISBN: 978-1-84821-127-8

[27] Hoffmann, G. M. and Huang, H. and Waslander, S. L. and Tomlin, C. J.,"Precision flight control for a multi-vehicle quadrotor helicopter testbed", *IEEE Control Engineering Practice*, Vol.19, Issue 9, September 2011, pp. 10231036

[28] Pounds, P. and Mahony, R. and Corke, P. "Modelling and control of a quadrotor robot", in Proceedings of the Australasian Conference on Robotics and Automation, Auckland, New-Zealand, December 2006.

[29] Ducard, G. and D'Andrea R., "Autonomous Quadrotor Flight Using a Vision System And Accommodating Frames Misalignment", in Proceedings of IEEE International Symposium on Industrial Embedded Systems (SIES), 2009, pp 261-264.

[30] Zhang, R. and Wang, X. and Cai, K-Y., "Quadrotor aircraft control without velocity measurements", in Proceedings of IEEE Conference on Decision and Control. Shangai, P.R. China, December, 2009.

[31] Eckert, J. and German, R. and Dressler, F., "On Autonomous Indoor Flights: High-Quality Real-Time Localization using Low-Cost Sensors", in Proceedings of IEEE International Conference on Communications (ICC), vol., no., pp.7093,7098, 10-15 June 2012.

[32] Bonak, M. and Matko, D. and Blazic, S.,"Quadrocopter control using an on-board video system with off-board processing". *Robotics and Autonomous Systems*, vol. 60, pp 657667, 2012.

[33] Eberli, D. and Scaramuzza, D. and Weiss, S. and Siegwart, R., "Vision Based Position Control for MAVs Using One Single Circular Landmark". *International Robot Systems*, num. 61, pp. 495-512, 2011.

[34] Martnez, C. and Mondragn, I. F. and Olivares-Mndez, M. A. and Campoy, P., "On-board and Ground Visual Pose Estimation Techniques for UAV Control", *Intelligent and Robotic Systems*, vol. 61, num.1-4, 2001, pp. 301-320.

[35] Altug, E. and Taylor, C., "Vision-based pose estimation and control of a model helicopter," in Proceedings of the IEEE International Conference on Mechatronics , no., 2004, pp. 316- 321.

[36] Teuliere, C. and Eck, L. and Marchand, E. and and Guenard, N., "3D model-based tracking for UAV position control", in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010, pp. 1084-1089.

[37] Castillo, P. and Lozano, R. and Dzul, A., *Modelling and control of mini flying machines*, Springer-Verlag, 2005. ISBN: 1-85233-957-8.

[38] Rondon, E., "Navigation d'un Vehicule Aerien par Flux Optique", PhD Thesis, Université de Tehcnologié de Compiègne, France, 2010.

[39] Eresen, A. and Imamoglu, N. and Onder Efe M., "Autonomous quadrotor flight with vision-based obstacle avoidance in virtual enviroment". *Expert Systems with Applications*. Vol 39, 2012, pp. 894-905.