Ingeniería Investigación y Tecnología
volumen XXII (número 2), abril-junio 2021 1-18
ISSN 2594-0732 FI-UNAM artículo arbitrado
Información del artículo: Recibido: 16 de septiembre de 2020, reevaluado: 6 de enero de 2021, aceptado: 16 de febrero de 2021

https://doi.org/10.22201/fi.25940732e.2021.22.2.015

Ingeniería
Investigación y
Tecnología

Control and signal filtering system for a quadcopter; analysis, comparison and implementation via low-cost IMU and microcontroller

Sistema de control y filtrado de señales para un quadrotor; análisis, comparación e implementación mediante IMU de bajo costo y microcontrolador

Allende-Peña Jacobo Marcos
Tecnológico de Estudios Superiores de Tianguistenco
Estado de México - Instituto Politécnico Nacional, México
Sección de Estudios de Posgrado e Investigación de la
ESIME Azcapotzalco
E-mail: jacobmarcoap@gmail.com
https://orcid.org/0000-0003-1227-0449

Rodríguez-Paredes Salvador Antonio
Instituto Politécnico Nacional, México
Sección de Estudios de Posgrado e Investigación
de la ESIME U. Azcapotzalco
E-mail: sarodriguezp@ipn.mx
https://orcid.org/0000-0002-1725-9874

Salmerón-Quiroz Bernardino Benito
Instituto Politécnico Nacional, México
Sección de Estudios de Posgrado e Investigación de la
ESIME Azcapotzalco
E-mail: bsalmeron@ipn.mx
https://orcid.org/0000-0001-9028-8287

Abstract

This work presents the development, analysis and comparison of dynamic models, to implement in a simple way a filtering and control system for a quadcopter through the use of a microcontroller. A non-linear dynamic model obtained from the Euler Lagrange equations is approached; also a simplified non-linear model from the first model and a linear model are obtained. Subsequently, making use of each model, controllers are designed using the computed torque controller technique, then, their performance index is obtained through numerical simulations applied to the complete non-linear model to compare their response. Afterward, with the various models and through observers and Kalman filters, signal filtering systems are synthesized for a low-cost Inertial Measurement Unit (IMU), the filtering results are also compared using a performance index. Additionally, the proposed controller is compared in simulation with a fuzzy logic controller. Finally, the selection of the model, the controller and filtering are validated through experimentation with a quadcopter prototype developed by the authors, based on an experimental platform with four rotors, a fiberglass structure, a microcontroller and an IMU MPU 6050.
Keywords: Quadcopter, model analysis, computed torque controller, Kalman filter, observer, low-cost IMU, implementation, microcontroller.

Resumen

Se presenta el desarrollo, análisis y comparación de modelos dinámicos, para implementar de forma simple un sistema de filtrado y de control para una aeronave tipo quadrotor mediante el uso de microcontrolador. Se aborda un modelo dinámico no lineal obtenido de las ecuaciones de Euler Lagrange, así como un modelo no lineal simplificado a partir del primer modelo y, a su vez, un modelo lineal. Posteriormente, haciendo uso de cada modelo, se diseñan controladores mediante la técnica de par calculado, luego se obtienen sus índices de desempeño mediante simulaciones numéricas aplicadas al modelo no lineal completo, con el fin de comparar su respuesta. Además, con los diversos modelos y mediante observadores y filtros de Kalman, se sintetizan propuestas de filtrado de señales para una Unidad de Medición Inercial (IMU) de bajo costo, también se comparan los resultados del filtrado mediante un índice de desempeño. Adicionalmente se compara en simulación el controlador propuesto con un controlador por lógica difusa. Por último se valida la selección del modelo, el controlador y filtrado mediante experimentación con un prototipo de quadrotor desarrollado por los autores, basado en una plataforma experimental de cuatro rotores, una estructura de fibra de vidrio, un microcontrolador y una IMU MPU 6050.
Descriptores: Quadrotor, análisis de modelos, control par calculado, filtro de Kalman, observador de estado, IMU de bajo costo, implementación, microcontrolador.

https://doi.org/10.22201/fi.25940732e.2021.22.2.015

Introduction

In recent years, the rise of unmanned aerial vehicles (UAV) has had a great impact on society so the area of possible applications increases continuously, for example: in the field of the military, trade, the search, rescue, dangerous environment monitoring, product delivery, as well as others (Floreano & Wood, 2015; Gupta *et al*., 2013; Cárdenas, 2015; Gutiérrez *et al*., 2017.

One of the main problems, when is generating a control system for an aerial vehicle, is to develop a dynamic model that describes the real behavior of the aircraft in the most precise way possible and at the same time; keep the model simple to design, analyze and investigate various control strategies, signal filtering, as well as implement the controller and filtering system in a digital system in a simple way. The quadcopter mathematical model is a set of equations that allows representing in detail the dynamic behavior of the quadcopter in flight; one of the main ways to approximate the dynamic model is applying the Euler-Lagrange methodology**,** which consists in observing the angular velocities of the vehicle, from a reference frame, movable but aligned to an inertial frame (Sadr *et al.,* 2014; Rodríguez *et al.,* 2016; Balasubramanian & Vasantharaj, 2013; Nonami *et al.,* 2010; Beard & McLain, 2012; Garcia *et al.,* 2006; Castillo *et al.,* 2007; Carrillo *et al.,* 2012; Raffo, 2007), this methodology incorporates the inertial and gyroscopic effects of the aircraft structure into the model. On the other hand, there is a model developed using Newton-Euler, which observes the angular velocities of the quadrotor from a mobile frame defined in the center of rotation of the aircraft (Bouabdallah & Siegwart, 2007; Falconi & Melchiorri, 2012; Elruby *et al.,* 2012; de Jesus Rubio *et al.,* 2014; Paiva, 2016; Bouabdallah & Siegwart, 2004; Hossain *et al.,* 2010; Patete & Erazo, 2016; Agho, 2017), in this model, it is common to observe that the gyroscopic and inertial effects of the aircraft structure are despised, however, the gyroscopic effect of the rotation of the propellers is incorporated. Also, there are linear approximations, in which the flight dynamics around an operating point are frequently approximated (Sabatino, 2015; Roldan, 2016; Sevilla, 2014).

The implementation of the controller to stabilize the rotational movements of the quadrotor requires knowledge of the angular positions and speeds, these variables can be known through sensors or estimated through the model that describes the dynamics of the system. A methodology that allows fusing such data is the Kalman filter, this methodology has been used in countless areas of engineering since its proposal by Kalman (1960), particularly in the area of unmanned aerial vehicles; the main application is the estimation of the position and orientation of the aircraft (attitude) (You *et al.,* 2020;. Bauer & Bokor, 2008; Loianno *et al.,* 2016; Markley, 2003; Emran *et al.,* 2015; Sanz *et al.,* 2014; Sebesta & Boizot, 2013; Muñoz *et al.,* 2013; Gośliński *et al.,* 2013; Sarim *et al.,* 2015; Tailanián *et al.,* 2014; Amoozgar *et al.,* 2013; Xiong & Zheng, 2015). Likewise, there are proposals for filtering signals in unmanned aircraft based on state observers, with the advantage of reducing the computational cost, due not processing the calculation at each moment the profit matrix or Kalman matrix (Escobedo *et al.,* 2018; Macdonald *et al.,* 2014; Hanley & Bretl, 2016; Lendek *et al.,* 2011; Shi *et al.,* 2018; Yu *et al.,* 2015). Other proposals for filtering signals with a low computational cost include complementary filters (Noordin *et al.,* 2018; Jung & Tsiotras, 2007; Euston *et al.,* 2008; Yoo *et al.,* 2001), however; these proposals are less efficient than observers or the Kalman filter.

In this work, the Euler-Lagrange model of a quadcopter is analyzed, such as proposed in (Beard & McLain, 2012; Garcia *et al.,* 2006; Castillo *et al.,* 2007; Carrillo *et al.,* 2012; Raffo, 2007), but additionally incorporates the gyroscopic effects of the rotation of the helices, then approximations based on Taylor series are made, that simplify the model rotation dynamics which we called complete and allow us to represent the dynamics in a reduced non-linear model similar to the one proposed in (Bouabdallah & Siegwart, 2007; Falconi & Melchiorri, 2012; Elruby *et al.,* 2012; de Jesus Rubio *et al.,* 2014), additionally, a linear model is obtained.

Aircraft modeling

This section describes the model that defines the dynamic behavior of translation and rotation of the quadcopter. To develop this model, the Euler-Lagrange formalism is used.

Quadcopter speeds, forces and motions

Figure 1 shows the reference frames used for modeling, as well as the free-body diagram. The orientation of the aircraft is defined by the Euler angles: $\boldsymbol{\eta} = \{\phi, \theta, \psi\}$, pitch, roll, and yaw, respectively.

The total angular velocity of the aircraft is a sum of the rotational speeds in each reference axis of the system {c}. Therefore, the total angular velocity of the quadrotor represented by $(p, q, r) = (\boldsymbol{\Omega})$, is seen in the axes (X,Y,Z) of the reference frame {a} and is expressed according to the following relation (in the region where the Euler angles are valid), (Beard & McLain, 2012; Gar-

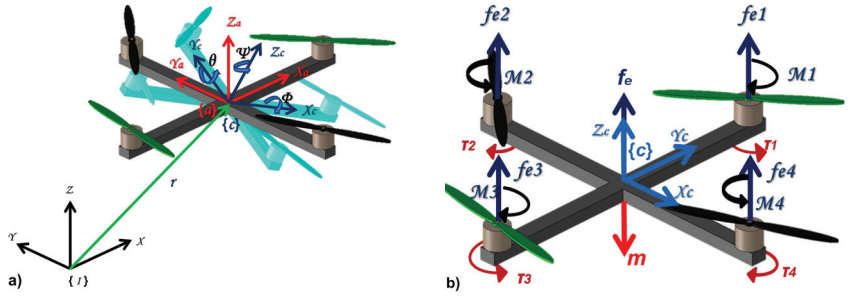

Figure 1. a) Reference frames, b) free-body diagram

cia *et al.*, 2006; Castillo *et al.*, 2007; Carrillo *et al.*, 2012; Raffo, 2007).

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{pmatrix} \dot{\phi} \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} 0 \\ \dot{\theta} \\ 0 \end{pmatrix} \tag{1}$$

$$+ \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \dot{\psi} \end{pmatrix}$$

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{pmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{pmatrix} \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \tag{2}$$

$$\boldsymbol{\Omega} = \boldsymbol{T}_{\omega c}\dot{\boldsymbol{\eta}} \tag{3}$$

According to Figure1b, the thrust force seen in the reference frame {c} is defined:

$$F_T = (f_{e1} + f_{e2} + f_{e3} + f_{e4}) \tag{4}$$

The thrust force f_{ei} for each engine and the aerodynamic drag force τ_i, which opposes the torque M_i generated by the rotation of the propellers, are expressed respectively, as:

$$f_{ei} = k_e\omega_i^2 \tag{5}$$

$$T_i = k_a\omega_i^2 \tag{6}$$

Where, ω_i is the angular velocity of each engine, k_a *and* k_e are positive definite constants that depend on the density of the air, the radius of rotation, the area and the shape of the propeller blades, as well as other factors. In conditions where the angular speed ω_i of the motor is constant or the rates of change of this speed are small, the aerodynamic drag torque is equivalent to the torque produced by the motor (Beard & McLain, 2012; Garcia *et al.*, 2006).

The thrust force acts on the "z" axis of the reference frame {c}, so the lift force in the inertial reference frame is defined:

$${}^{I}f = \begin{pmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ f_e \end{pmatrix} \tag{7}$$

$${}^{I}f = \begin{pmatrix} f_e(\sin\phi\sin\psi + \cos\phi\cos\psi\sin\theta) \\ f_e(\cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi) \\ f_e(\cos\theta\cos\phi) \end{pmatrix} \tag{8}$$

The rotational movements are proposed on the bisector of the aircraft arms, that is, a flight in the "x" configuration. Then the rotation of an angle Φ (*pitch*), is caused by a difference in the forces, such that:

$$\tau_\phi = d(f_{e1} + f_{e2}) - d(f_{e3} + f_{e4}) \tag{9}$$

The rotation of an angle θ (*roll*) , is achieved through a difference in the forces, which produces a torque:

$$\tau_\theta = d(f_{e2} + f_{e3}) - d(f_{e1} + f_{e4}) \tag{10}$$

To obtain a rotation of an angle ψ (*yaw*) it is achieved by a difference in the torques $(T_3 + T_1)$ and $(T_2 + T_4)$ which produces:

$$\tau_\psi = [(T_3 + T_1) - (T_2 + T_4)] \tag{11}$$

The aerodynamic drag torque T_i is expressed as:

$$T_i = lf_{ri} = lk_f f_{ei} \tag{12}$$

Where $\boldsymbol{f}_{ri}$ is the force perpendicular to length l and k_r relates f_{ei} and f_{ri}, so that equation (13) can be rewritten as:

$$\tau_{\psi} = l[(f_{r3} + f_{r1}) - (f_{r2} + f_{r4})] \tag{13}$$

Also, each engine-propeller pair is considered as a rigid disk rotating with a speed ω_i, around the "z", axis, of the reference frame {c}, so the following gyroscopic effects ($\tau_{G\phi}$, $\tau_{G\theta}$, $\tau_{G\psi}$), acts in the inertial reference frame, opposing to the control inputs. Where J is the moment of inertia of the engine-propeller pair and $\omega_T = \omega_2 + \omega_4 - \omega_1 - \omega_3$.

$$\begin{pmatrix} \tau_{G\phi} \\ \tau_{G\theta} \\ \tau_{G\psi} \end{pmatrix} = -\begin{pmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{pmatrix} \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ J_{\omega T} \end{pmatrix} = \tag{14}$$

$$\begin{pmatrix} -J\omega_T(\dot{\theta}\cos\phi + \dot{\psi}\sin\phi\cos\theta) \\ -J\omega_T(\dot{\psi}\sin\theta - \dot{\phi}) \\ 0 \end{pmatrix}$$

The equations that define the relation of the total thrust force, the torques and the thrust forces of each engine, are given by equations (9), (10), (11), (12) and (13), then the forces in each engine as a function of the control inputs are:

$$\begin{pmatrix} F_{e1} \\ F_{e2} \\ F_{e3} \\ F_{e4} \end{pmatrix} = \begin{pmatrix} 1/4 & -1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & -1/4 & 1/4 \\ -1/4 & 1/4 & 1/4 & 1/4 \\ -1/4 & -1/4 & -1/4 & 1/4 \end{pmatrix} \begin{pmatrix} \tau_{\phi}/d \\ \tau_{\theta}/d \\ \tau_{\psi}/lk_r \\ F_T \end{pmatrix} \tag{15}$$

$$= \begin{pmatrix} 1/4d\,\tau_{\phi} & -1/4d\,\tau_{\theta} & +1/4lk_r\,\tau_{\psi} & +1/4\,F_T \\ 1/4d\,\tau_{\phi} & +1/4d\,\tau_{\theta} & -1/4lk_r\,\tau_{\psi} & +1/4\,F_T \\ -1/4d\,\tau_{\phi} & +1/4d\,\tau_{\theta} & +1/4lk_r\,\tau_{\psi} & +1/4\,F_T \\ -1/4d\,\tau_{\phi} & -1/4d\,\tau_{\theta} & -1/4lk_r\,\tau_{\psi} & +1/4\,F_T \end{pmatrix}$$

Translation and rotation model

Due to the rotational kinetic energy and the translational kinetic energy do not have dependent terms; it is possible to separate the translation and rotation equations (Beard & McLain, 2012; Garcia *et al.*, 2006; Castillo *et al.*, 2007; Carrillo *et al.*, 2012; Raffo, 2007). According to the Euler-Lagrange equations, the translation model is stated as:

$$m\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} + mg\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{pmatrix} f_e(\sin\phi\sin\psi + \cos\phi\cos\psi\sin\theta) \\ f_e(\cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi) \\ f_e(\cos\theta\cos\phi) \end{pmatrix} \tag{16}$$

Likewise, the dynamic model that describes the rotation of the quadcopter is given as:

$$\boldsymbol{M}(\boldsymbol{\eta})\ddot{\boldsymbol{\eta}} + \boldsymbol{C}(\boldsymbol{n}, \dot{\boldsymbol{\eta}})\,\dot{\boldsymbol{\eta}} = \boldsymbol{\tau} + \boldsymbol{\tau}_G \tag{17}$$

Where $\boldsymbol{C}(\boldsymbol{n}, \dot{\boldsymbol{\eta}})$ is the Coriolis matrix, which contains the centrifugal forces and gyroscopic effects associated with the vector $\boldsymbol{\eta}$, $\boldsymbol{M}(\boldsymbol{\eta})$**:** represents the inertia matrix and $\boldsymbol{\tau}_G$ is the gyroscopic effects associated with rotation of the propellers. Also, due to the symmetric structure in the aircraft, the inertia matrix takes the form (Beard & McLain, 2012; Garcia *et al.*, 2006; Castillo *et al.*, 2007; Carrillo *et al.*, 2012; Raffo, 2007).

$$\boldsymbol{M}(\boldsymbol{\eta}) = \boldsymbol{T}_{\boldsymbol{\omega c}}^{T}\boldsymbol{I}\boldsymbol{T}_{\boldsymbol{\omega c}}$$
$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{pmatrix} \begin{pmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{pmatrix} \begin{pmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{pmatrix} \tag{18}$$

In turn, the Coriolis matrix is defined as (Castillo *et al.*, 2007; Raffo, 2007):

$$\boldsymbol{C}(\boldsymbol{n}, \dot{\boldsymbol{\eta}}) = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{pmatrix} \tag{19}$$

Where:

$c_{11} = 0$

$c_{12} = (I_y - I_z)(\dot{\theta}\cos\phi\sin\phi + \dot{\psi}\sin^2\phi\cos\theta) + (I_z - I_y)\,\dot{\psi}\cos^2\phi\cos\theta - I_x\dot{\psi}\cos\theta$

$c_{13} = (I_z - I_y)(\dot{\psi}\cos\phi\sin\phi\cos^2\theta)$

$c_{21} = (I_z - I_y)(\dot{\theta}\cos\phi\sin\phi + \dot{\psi}\sin^2\phi\cos\theta) + (I_y - I_z)(\dot{\psi}\cos^2\phi\cos\theta) + I_x\dot{\psi}\cos\theta$

$c_{22} = (I_z - I_y)(\dot{\phi}\cos\phi\sin\phi)$

$c_{23} = -I_x \dot{\psi} \sin\theta\cos\theta + I_y\dot{\psi} \sin^2\phi \cos\theta\sin\theta + I_z\dot{\psi}\cos^2 \phi \sin\theta \cos\theta$

$c_{31} = (I_y - I_z)(\dot{\psi}\cos^2\theta\sin\phi\cos\phi) - I_x\dot{\theta}\cos\theta$

$c_{32} = (I_z - I_y)(\dot{\theta}\cos\phi\sin\phi\sin\theta + \dot{\phi}\sin^2\phi\cos\theta) + (I_y - I_z)(\dot{\phi}\cos^2\phi\cos\theta) + I_x\dot{\psi}\sin\theta\cos\theta - I_y\dot{\psi}\sin^2\phi\cos\theta\sin\theta - I_z\dot{\psi}\cos^2\phi\sin\theta\cos\theta$

$c_{33} = (I_y - I_z)(\dot{\phi}\cos\phi\sin\phi\cos^2\theta - I_y\dot{\theta}\sin^2\phi\cos\theta\sin\theta - I_z\dot{\theta}\cos^2\phi\cos\theta\sin\theta + I_x\dot{\theta}\cos\theta\sin\theta)$

Reductions and simplifications to the complete rotation model

In this section, two approaches to the rotation model are presented. Initially, a classical approximate linearization is approached, then, an approximation is presented to obtain a reduced non-linear model.

Linear model

To linearize the system, the rotation model is expressed in state variables and the zero equilibrium state is considered. The dynamics of rotation in state variables from (17) is expressed as:

$$\begin{pmatrix}\dot{\boldsymbol{\eta}}\\ \ddot{\boldsymbol{\eta}}\end{pmatrix} = \begin{pmatrix}\dot{\boldsymbol{\eta}}\\ -\boldsymbol{M}(\boldsymbol{\eta})^{-1}\boldsymbol{C}(n,\dot{\boldsymbol{\eta}})\dot{\boldsymbol{\eta}} + \boldsymbol{M}(\boldsymbol{\eta})^{-1}\tau_G + \boldsymbol{M}(\boldsymbol{\eta})^{-1}\boldsymbol{\tau}\end{pmatrix} \tag{20}$$

The state vector and the equilibrium input vector, corresponding to:

$$\bar{\boldsymbol{X}} = \begin{pmatrix}\bar{\boldsymbol{\eta}}\\ \dot{\bar{\boldsymbol{\eta}}}\end{pmatrix} = \begin{pmatrix}0\\ 0\end{pmatrix} \tag{21}$$

Defining the incremental states and incremental inputs as:

$$\begin{aligned}\mathbf{x}_\delta &= (\mathbf{x} - \bar{\mathbf{x}})\\ \mathbf{u}_\delta &= (\mathbf{u} - \bar{\mathbf{u}})\end{aligned} \tag{22}$$

Then the linearized system around an equilibrium point takes the form:

$$\dot{\mathbf{x}}_\delta = \mathbf{A}\mathbf{x}_\delta + \mathbf{B}\mathbf{u}_\delta \tag{23}$$

Where:

$$\boldsymbol{A} = \frac{\partial \boldsymbol{f}(\bar{\boldsymbol{x}},\bar{\boldsymbol{u}})}{\partial \boldsymbol{x}} = \frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})\,|_{(\bar{x},\bar{u})}$$

$$\boldsymbol{B} = \frac{\partial \boldsymbol{f}(\bar{\boldsymbol{x}},\bar{\boldsymbol{u}})}{\partial \boldsymbol{u}} = \frac{\partial}{\partial \boldsymbol{u}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})\,|_{(\bar{x},\bar{u})}$$

Therefore, the approximate linear system is defined by the following expression:

$$\begin{pmatrix}\dot{\boldsymbol{\eta}}\\ \ddot{\boldsymbol{\eta}}\end{pmatrix} = \begin{pmatrix}\boldsymbol{0}_{3x3} & \boldsymbol{I}_{3x3}\\ \boldsymbol{0}_{3x3} & \boldsymbol{Mb}(\boldsymbol{\eta})^{-1}\tau_{bG}\end{pmatrix}\begin{pmatrix}\boldsymbol{\eta}\\ \dot{\boldsymbol{\eta}}\end{pmatrix} + \begin{pmatrix}\boldsymbol{0}_{3x3}\\ \boldsymbol{M}_b(\boldsymbol{\eta})^{-1}\end{pmatrix}(\boldsymbol{\tau}) \tag{24}$$

$$\tau_{bG} = \begin{pmatrix}0 & -J\omega_T & 0\\ J\omega_T & 0 & 0\\ 0 & 0 & 0\end{pmatrix}, \boldsymbol{M}_b(\boldsymbol{\eta}) = \begin{pmatrix}I_x & 0 & 0\\ 0 & I_y & 0\\ 0 & 0 & I_z\end{pmatrix}, \boldsymbol{\eta} = \begin{pmatrix}\phi\\ \theta\\ \psi\end{pmatrix}, \dot{\boldsymbol{\eta}} = \begin{pmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{pmatrix}$$

Simplifying the previous expressions, we obtain:

$$\begin{pmatrix}\ddot{\phi}\\ \ddot{\theta}\\ \ddot{\psi}\end{pmatrix} = \begin{pmatrix}\frac{1}{I_x} & 0 & 0\\ 0 & \frac{1}{I_y} & 0\\ 0 & 0 & \frac{1}{I_z}\end{pmatrix}\begin{pmatrix}0 & -J\omega_T & 0\\ J\omega_T & 0 & 0\\ 0 & 0 & 0\end{pmatrix}\begin{pmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{pmatrix} + \begin{pmatrix}\frac{1}{I_x} & 0 & 0\\ 0 & \frac{1}{I_y} & 0\\ 0 & 0 & \frac{1}{I_z}\end{pmatrix}\begin{pmatrix}\tau_\phi\\ \tau_\theta\\ \tau_\psi\end{pmatrix} \tag{25}$$

Reduced non-linear model

Another way to reduce the complexity of the model is to consider small movements, that is, the transformation matrix $\boldsymbol{T}_{\omega c} \rightarrow \boldsymbol{I}$, in this way $\Omega \approx \dot{\boldsymbol{\eta}}$. Also we can approximate:

$$\cos\phi \approx \cos\theta \approx \cos\psi \approx 1; \sin\phi, \approx \sin\theta \approx \theta, \sin\psi \approx \psi \tag{26}$$

Then, applying these approximations to the inertia matrix (18) and Coriolis matrix (19), we obtain:

$$\boldsymbol{M}_c(\boldsymbol{\eta}) = \boldsymbol{T}_{\omega c}^T \boldsymbol{I}\boldsymbol{T}_{\omega c} = \tag{27}$$

$$\begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}I_x & 0 & 0\\ 0 & I_y & 0\\ 0 & 0 & I_z\end{pmatrix}\begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix} = \begin{pmatrix}I_x & 0 & 0\\ 0 & I_y & 0\\ 0 & 0 & I_z\end{pmatrix}$$

$$\boldsymbol{C}_c(\boldsymbol{n},\dot{\boldsymbol{\eta}}) = \begin{pmatrix} c_{c11} & c_{c12} & c_{c13} \\ c_{c21} & c_{c22} & c_{c23} \\ c_{c31} & c_{c32} & c_{c33} \end{pmatrix}$$

$$\begin{aligned} c_{c11} &= 0 \\ c_{c12} &= \dot{\psi}\left(I_z - I_y - I_x\right) + \dot{\theta}\phi\left(I_y - I_z\right) \\ c_{c13} &= \dot{\psi}\phi\left(I_z - I_y\right) \\ c_{c21} &= \dot{\psi}(I_y + I_x - I_z) + \dot{\theta}\phi\left(I_z - I_y\right) \\ c_{c22} &= \dot{\phi}\phi(I_y - I_z) \\ c_{c23} &= \dot{\psi}\theta(I_z - I_x) \\ c_{c31} &= \dot{\psi}\phi\left(I_y - I_z\right) - \dot{\theta}I_x \\ c_{c32} &= \dot{\theta}\phi\theta\left(I_z - I_y\right) + \dot{\phi}\left(I_y - I_z\right) + \dot{\psi}\theta(I_x - I_z) \\ c_{c33} &= \dot{\phi}\phi(I_y - I_z) + \dot{\theta}\theta(I_x - I_z) \end{aligned} \tag{28}$$

$$\tau_{cG} = \begin{pmatrix} \tau_{G\phi} \\ \tau_{G\theta} \\ \tau_{G\psi} \end{pmatrix} = -\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ J\omega_T \end{pmatrix} = \begin{pmatrix} -\dot{\theta}J\omega_T \\ \dot{\phi}J\omega_T \\ 0 \end{pmatrix} \tag{29}$$

Then replacing (27), (28), (29) in (17) and rearranging, we obtain the reduced non-linear model for the rotation dynamics:

$$\begin{pmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{pmatrix} = -\begin{pmatrix} \frac{1}{I_x} & 0 & 0 \\ 0 & \frac{1}{I_y} & 0 \\ 0 & 0 & \frac{1}{I_z} \end{pmatrix}\begin{pmatrix} c_{c11} & c_{c12} & c_{c13} \\ c_{c21} & c_{c22} & c_{c23} \\ c_{c31} & c_{c32} & c_{c33} \end{pmatrix}\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} + \begin{pmatrix} 1/I_x & 0 & 0 \\ 0 & 1/I_y & 0 \\ 0 & 0 & 1/I_z \end{pmatrix}\begin{pmatrix} -\dot{\theta}J\omega_T \\ \dot{\phi}J\omega_T \\ 0 \end{pmatrix} + \begin{pmatrix} 1/I_x & 0 & 0 \\ 0 & 1/I_y & 0 \\ 0 & 0 & 1/I_z \end{pmatrix}\begin{pmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{pmatrix} \tag{30}$$

Computer torque controller

The open-loop dynamics of the quadcopter is unstable, so it is necessary to apply a closed-loop control system. In this section, we propose a control system based on the computed torque technique (Reyes, 2011; Sira *et al.*, 2005).

Control based on the full model

The controller designed using the full model is:

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{\eta})\left(\ddot{\boldsymbol{\eta}}^* - \boldsymbol{K}_a\dot{\boldsymbol{\eta}}_e - \boldsymbol{K}_b\boldsymbol{\eta}_e - \boldsymbol{K}_i\int_0^t \boldsymbol{\eta}_e dt\right) + \boldsymbol{C}(\boldsymbol{n},\dot{\boldsymbol{\eta}})\dot{\boldsymbol{\eta}} - \tau_G \tag{31}$$

Where:

$\boldsymbol{\eta}_e = \boldsymbol{\eta} - \boldsymbol{\eta}^*; \dot{\boldsymbol{\eta}}_e = \dot{\boldsymbol{\eta}} - \dot{\boldsymbol{\eta}}^*; \ddot{\boldsymbol{\eta}}_e = \ddot{\boldsymbol{\eta}} - \ddot{\boldsymbol{\eta}}^*$, the superscript (*) represent the desired value.

Introducing the control in (17) and simplifying is obtained:

$$\ddot{\boldsymbol{\eta}} = -\boldsymbol{M}(\boldsymbol{\eta})^{-1}\boldsymbol{C}(\boldsymbol{n},\dot{\boldsymbol{\eta}})\dot{\boldsymbol{\eta}} + \boldsymbol{M}(\boldsymbol{\eta})^{-1}\tau_G + \boldsymbol{M}(\boldsymbol{\eta})^{-1}\boldsymbol{\tau} + \xi \tag{32}$$

$$\begin{aligned} \ddot{\boldsymbol{\eta}} &= -\boldsymbol{M}(\boldsymbol{\eta})^{-1}\boldsymbol{C}(\boldsymbol{n},\dot{\boldsymbol{\eta}})\dot{\boldsymbol{\eta}} + \boldsymbol{M}(\boldsymbol{\eta})^{-1}\tau_G \\ &+\boldsymbol{M}(\boldsymbol{\eta})^{-1}\Big[\boldsymbol{M}(\boldsymbol{\eta})\left(\ddot{\boldsymbol{\eta}}^* - \boldsymbol{K}_a\dot{\boldsymbol{\eta}}_e - \boldsymbol{K}_b\boldsymbol{\eta}_e - \boldsymbol{K}_i\int_0^t \boldsymbol{\eta}_\delta dt\right) \\ &+\boldsymbol{C}(\boldsymbol{n},\dot{\boldsymbol{\eta}})\dot{\boldsymbol{\eta}} - \tau_G\Big] + \xi \end{aligned} \tag{33}$$

The term ξ, are constant disturbances. Then the closed-loop dynamics that define the behavior of the quadcopter is:

$$\ddot{\boldsymbol{\eta}} = \ddot{\boldsymbol{\eta}}^* - \boldsymbol{K}_a\dot{\boldsymbol{\eta}}_e - \boldsymbol{K}_b\boldsymbol{\eta}_e - \boldsymbol{K}_i\int_0^t \boldsymbol{\eta}_e dt + \xi \tag{34}$$

$$\boldsymbol{K}_a = \begin{pmatrix} k_2 & 0 & 0 \\ 0 & k_4 & 0 \\ 0 & 0 & k_6 \end{pmatrix}, \boldsymbol{K}_b = \begin{pmatrix} k_1 & 0 & 0 \\ 0 & k_3 & 0 \\ 0 & 0 & k_5 \end{pmatrix}, \boldsymbol{K}_i = \begin{pmatrix} k_7 & 0 & 0 \\ 0 & k_8 & 0 \\ 0 & 0 & k_9 \end{pmatrix} \tag{35}$$

$$\begin{pmatrix} \ddot{\phi}_e \\ \ddot{\theta}_e \\ \ddot{\psi}_e \end{pmatrix} = \begin{pmatrix} -k_1\phi_e - k_2\dot{\phi}_e - k_7\int_0^t \phi_e dt + \xi_\phi \\ -k_3\theta_e - k_4\dot{\theta}_e - k_8\int_0^t \theta_e dt + \xi_\phi \\ -k_5\psi_e - k_6\dot{\psi}_e - k_9\int_0^t \psi_e dt + \xi_\phi^\theta \end{pmatrix} \tag{36}$$

ψ

When each of the previous expressions is derived, we can observe that the perturbations considered constant become zero, then the closed-loop stability can be analyzed through the characteristic polynomial, choosing positive definite values for $\boldsymbol{K}_a$, $\boldsymbol{K}_b$ and $\boldsymbol{k}_i$ such that $\boldsymbol{K}_a$, $\boldsymbol{K}_b > \boldsymbol{k}_i$; the system is stable.

Control based on linear model

Following the same strategy applied to the complete model, the control designed through the linear system is:

$$\tau_b = \boldsymbol{M}_b(\boldsymbol{\eta})\left(\ddot{\boldsymbol{\eta}}^* - \boldsymbol{K}_a\dot{\boldsymbol{\eta}}_e - \boldsymbol{K}_b\boldsymbol{\eta}_e - \boldsymbol{K}_i\int_0^t \boldsymbol{\eta}_e dt\right) - \boldsymbol{\tau}_{bG\boldsymbol{\eta}} \tag{37}$$

$$\tau_b = \begin{pmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{pmatrix} = \begin{pmatrix} I_x\left(\ddot{\phi}^* - k_1\phi_e - k_2\dot{\phi}_e - k_7\int_0^t \phi_e dt\right) + \dot{\theta}J\omega_T \\ I_y\left(\ddot{\theta}^* - k_3\theta_e - k_4\dot{\theta}_e - k_8\int_0^t \theta_e dt\right) - \dot{\phi}J\omega_T \\ I_z\left(\ddot{\psi}^* - k_5\psi_e - k_6\dot{\psi}_e - k_9\int_0^t \psi_e dt\right) \end{pmatrix} \tag{38}$$

Control based on reduced nonlinear model

In turn, the control system designed from the reduced system is:

$$\tau_c = \boldsymbol{M}_c(\boldsymbol{\eta})\left(\ddot{\boldsymbol{\eta}}^* - \boldsymbol{K}_a\dot{\boldsymbol{\eta}}_e - \boldsymbol{K}_b\boldsymbol{\eta}_e - \boldsymbol{K}_i\int_0^t \boldsymbol{\eta}_e dt\right) - \boldsymbol{\tau}_{cG} + \boldsymbol{C}_c(\boldsymbol{n},\dot{\boldsymbol{\eta}})\dot{\boldsymbol{\eta}} \quad (39)$$

$$\tau_c = \begin{pmatrix}\tau_\phi\\ \tau_\theta\\ \tau_\psi\end{pmatrix} = \begin{pmatrix} I_x\left(\ddot{\phi}^* - k_1\phi_e - k_2\dot{\phi}_e - k_7\int_0^t \phi_e dt\right) + \dot{\theta}J\omega_T \\ I_y\left(\ddot{\theta}^* - k_3\theta_e - k_4\dot{\theta}_e - k_8\int_0^t \theta_e dt\right) - \dot{\phi}J\omega_T \\ I_z\left(\ddot{\psi}^* - k_5\psi_e - k_6\dot{\psi}_e - k_9\int_0^t \psi_e dt\right)\end{pmatrix} + C_c(n,\dot{\eta})\begin{pmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{pmatrix} \quad (40)$$

Signal filtering

Figure 2a, shows the signal of one of the rotation angles of the quadcopter prototype, the signal is obtained from the projection of gravity on the axes of a low-cost Inertial Measurement Unit of 6 degrees of freedom (IMU 6 DOF, MPU6050), in addition, the angular positions and speeds of the drone can be accessed. Figure 2b shows one of the duty cycle signals (1000 us to 2000 us) with a frequency of 60 Hz, sent to the electronic speed controllers (ESC). From these graphs, we can observe the impact of mechanical vibration (noise) on the acquisition of the rotation signal, also, the noise increases as the speed of the motors increases.

For this reason, in this section, systems are developed and synthesized for filtering the quadcopter rotation signals.

Standard Kalman filter

The standard Kalman filter is an optimal state estimator and is applicable to a linear system, described by the following equation of state represented in discrete form; where: w_k and v_k are the process and measurement noise respectively.

$$x_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k + w_k \quad (41)$$

$$y_k = \boldsymbol{C}\boldsymbol{x}_k + v_k \quad (42)$$

The equations that describe the standard Kalman filter are defined as (Kalman, 1960; Simon, 2006):

$$\boldsymbol{K}_k = \boldsymbol{P}_k\boldsymbol{C}^T(\boldsymbol{C}\boldsymbol{P}_k\boldsymbol{C}^T + \boldsymbol{R})^{-1} \quad (43)$$

$$\boldsymbol{P}_{k+1} = \boldsymbol{A}(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C})\,\boldsymbol{P}_k\boldsymbol{A}^T + \boldsymbol{Q} \quad (44)$$

$$\hat{\boldsymbol{x}}_{k+1} = \boldsymbol{A}\hat{\boldsymbol{x}}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{K}_k[\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k] \quad (45)$$

$$\hat{\boldsymbol{x}}_k = \begin{pmatrix}\hat{\boldsymbol{\eta}}_k\\ \hat{\dot{\boldsymbol{\eta}}}_k\end{pmatrix}, \hat{\boldsymbol{\eta}}_k = \begin{pmatrix}\boldsymbol{\phi}_k\\ \boldsymbol{\theta}_k\\ \boldsymbol{\psi}_k\end{pmatrix}, \hat{\dot{\boldsymbol{\eta}}}_k = \begin{pmatrix}\dot{\boldsymbol{\phi}}_k\\ \dot{\boldsymbol{\theta}}_k\\ \dot{\boldsymbol{\psi}}_k\end{pmatrix}, \boldsymbol{u}_k = \boldsymbol{\tau}_k \quad (46)$$

Where:

$\hat{\boldsymbol{x}}_k$ = vector of estimated states
$\boldsymbol{y}_k$ = measurement matrix
$\boldsymbol{K}_k$ = filter gain matrix
$\boldsymbol{Q}$ = covariance of the noise process w_k
$\boldsymbol{R}$ = covariance of the noise in the measurement v_k
$\boldsymbol{I}$ = identity matrix

Also and according to the linear system (25), matrix $\boldsymbol{A}$, matrix $\boldsymbol{B}$, matrix $\boldsymbol{C}$ and the vector of states to be estimated are defined as:

$$\boldsymbol{A} = \begin{pmatrix}\boldsymbol{0}_{3x3} & \boldsymbol{I}_{3x3}\\ \boldsymbol{0}_{3x3} & \boldsymbol{M}_b(\boldsymbol{\eta}_k)^{-1}\boldsymbol{\tau}_{bG}\end{pmatrix}, B = \begin{pmatrix}\boldsymbol{0}_{3x3}\\ \boldsymbol{M}_b(\boldsymbol{\eta}_k)^{-1}\end{pmatrix}, \boldsymbol{C} = \begin{pmatrix}\boldsymbol{I}_{3x3} & \boldsymbol{0}_{3x3}\\ \boldsymbol{0}_{3x3} & \boldsymbol{I}_{3x3}\end{pmatrix} \quad (47)$$

Extended Kalman filter

The extended Kalman filter, unlike the standard KF, uses the non-linear model to estimate the system states. That is, the equation of state is represented as:

$$x_{k+1} = f(x_k, u_k) + w_k \quad (48)$$

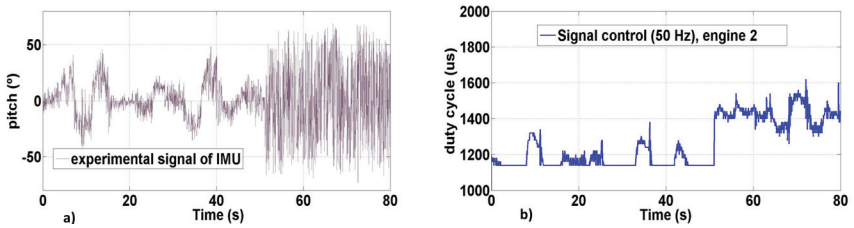

Figure 2. a) Angle of rotation contaminated in noise, b) duty cycle of one of the engines

$$y_k = h(x_k) + v_k \tag{49}$$

The equations that define the extended Kalman filter are (Simon, 2006):

$$K_k = P_k C^T_k (C_k P_k C^T_k + R)^{-1} \tag{50}$$

$$P_{k+1} = A_k(I - K_k C_k)P_k A^T_k + Q \tag{51}$$

$$\hat{x}_{k+1} = f(\hat{x}_k, u_k) + K_k[y_k - h(\hat{x}_k)] \tag{52}$$

The application of the previous equations requires the calculation of the matrices A_k, and C_k at each instant of time, these matrices are calculated trough the partial derivative of the equation of state and the output equation, then the results are evaluated in the vector of updated estimated states. According to the reduced nonlinear model (30), the matrix A_k and the matrix C_k are defined as:

$$A_k = f'(\hat{x}_k, u_k) = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ A_{k41} & 0 & 0 & 0 & A_{k45} & A_{k46} \\ A_{k51} & A_{k52} & 0 & A_{k54} & A_{k55} & A_{k56} \\ A_{k61} & A_{k62} & 0 & A_{k64} & A_{k65} & A_{k66} \end{pmatrix} \tag{53}$$

Where:

$$A_{k41} = \frac{1}{I_x}\dot{\theta}_k^2(I_y - I_z) + \frac{1}{I_x}\dot{\psi}_k^2(-I_y + I_z)$$

$$A_{k45} = \frac{1}{I_x}\dot{\psi}_k(-I_x - I_y + I_z) + \frac{1}{I_x}2\dot{\theta}_k\phi_k(I_y - I_z) - \frac{J\omega_T}{I_x}$$

$$A_{k46} = \frac{1}{I_x}\dot{\theta}_k(-I_x - I_y + I_z) + \frac{1}{I_x}2\dot{\psi}_k\phi_k(-I_y + I_z)$$

$$A_{k51} = \frac{1}{I_y}2\dot{\phi}_k\dot{\theta}_k(-I_y + I_z)$$

$$A_{k52} = \frac{1}{I_y}\dot{\psi}_k^2(-I_x + I_z)$$

$$A_{k54} = \frac{1}{I_y}\dot{\psi}_k(I_x + I_y - I_z) + \frac{1}{I_y}2\dot{\theta}_k\phi_k(-I_y + I_z) + \frac{J\omega_T}{I_y}$$

$$A_{k55} = \frac{1}{I_y}2\dot{\phi}_k\phi_k(-I_y + I_z)$$

$$A_{k56} = \frac{1}{I_y}\dot{\phi}_k(I_x + I_y - I_z) + \frac{1}{I_y}2\dot{\psi}_k\theta_k(-I_x + I_z)$$

$$A_{k61} = \frac{1}{I_z}\dot{\theta}_k^2\theta_k(-I_y + I_z) + \frac{1}{I_z}2\dot{\phi}_k\dot{\psi}_k(I_y - I_z)$$

$$A_{k62} = \frac{1}{I_z}\dot{\theta}_k^2\phi_k(-I_y + I_z) + \frac{1}{I_z}2\dot{\theta}_k\dot{\psi}_k(I_x - I_z)$$

$$A_{k64} = \frac{1}{I_z}\dot{\theta}_k(-I_x + I_y - I_z) + \frac{1}{I_z}2\dot{\psi}_k\phi_k(I_y - I_z)$$

$$A_{k65} = \frac{1}{I_z}\dot{\phi}_k(-I_x + I_y - I_z) + \frac{1}{I_z}2\dot{\theta}_k\theta_k\phi_k(-I_y + I_z) + \frac{1}{I_z}2\dot{\psi}_k\theta_k(I_x - I_z)$$

$$A_{k66} = \frac{1}{I_z}\dot{\theta}_k\theta_k(I_x - I_z) + \frac{1}{I_z}2\dot{\phi}_k\phi_k(I_y - I_z)$$

$$C_k = h'(\hat{x}_k) = \begin{pmatrix} I_{3x3} & 0_{3x3} \\ 0_{3x3} & I_{3x3} \end{pmatrix}, \hat{x}_k = \begin{pmatrix} \hat{\eta}_k \\ \hat{\dot{\eta}}_k \end{pmatrix}, u_k = \tau_k \tag{54}$$

Filter based on observer

The observability matrix of (A, C), for the linear system (25), is full-range, so the state observer is established as:

$$\dot{\hat{x}} = A\hat{x} + Bu + K_o[y - \hat{y}] \tag{55}$$

$$\hat{y} = C\hat{x}, \hat{x} = \begin{pmatrix} \hat{\eta} \\ \dot{\hat{\eta}} \end{pmatrix} \tag{56}$$

The equation that defines the dynamics of the error that relates to the estimated states and the real states is:

$$\dot{e}_O = \dot{x} - \dot{\hat{x}} = Ax + Bu - A\hat{x} - Bu - K_O[Cx - C\hat{x}]) \tag{57}$$

$$\dot{e}_O = (A - k_O C)e_O \tag{58}$$

We can verify the stability of the system through the eigenvalues of the matrix ($A - k_O C$), where the values can be chosen for the k_O, matrix, such that, the observer behaves in a stable dynamic.

Filter based on extended observer

We can estimate the linear representation of the dynamics of the aircraft through a states observer, so the estimate states can be extrapolated to the non-linear model, that is, an extended Luenberger-type observer is proposed, now the estimation of the states is given by:

$$\dot{\hat{x}} = f(\hat{x}, u) + k_e[y - \hat{y}] \tag{59}$$

Now the dynamics of the error between the estimated states with the real states is defined as:

$$\dot{e} = \dot{x} - \dot{\hat{x}} = f(x, u) - f(\hat{x}, u) + K_e(h(x) - h(\hat{x})) \tag{60}$$

As the movements of the system are controlled around the equilibrium states $x = 0$ and $\dot{x} = 0$, the dynamics of the error can be studied in a linear way, taking the first term of the decomposition by means of Taylor series, around the value $e = 0$.

$$\dot{e} = f(x, u) - f(x + e, u) + K_e(h(x) - h(x + e)) \tag{61}$$

$$\dot{e} = \left[\frac{\delta}{\delta x} f(\bar{x}, \bar{u}) + K_e \frac{\delta}{\delta x} h(\bar{x}) \right] e + 0(e^2) \tag{62}$$

$$\dot{e} = (A - k_e C)e \tag{63}$$

Simulation

In this section, numerical simulations in open-loop, closed-loop and filtering system are presented with the different models, the simulation was executed using MatLab software in the Simulink environment, the numerical method used was Runge-Kutta of fourth-order to a simulation step of 0.001s. The parameters that define the complete aircraft model, as well as the parameters used in the control and filtering system, are shown in Table 1.

Table 1. Model parameters

Parameter	Value in model	Value in controller and filtering system	Unit
l	0.23	0.21	m
d	0.16	0.15	m
m	1.1	1.1	Kg
g	9.81	9.81	m/s^2
I_x	$15x10^{-3}$	$45x10^{-3}$	$Kg\ m^2$
I_y	$15x10^{-3}$	$45x10^{-3}$	$Kg\ m^2$
I_z	$30x10^{-3}$	$90x10^{-3}$	$Kg\ m^2$
J	$1x10^{-4}$	$3x10^{-4}$	$Kg\ m^2$
k_r	1/12	1/10	-
k_e	$1.5x10^{-6}$	$1.5x10^{-6}$	-

Open loop simulation

In open-loop simulations, the initial conditions are established as: (ϕ, θ, ψ) = (0.0, 0.0, 0.0) *rad* and ($\dot{\phi}$, $\dot{\theta}$, $\dot{\psi}$) = (0, 0, 0)*rad/s*, the following torques are proposed as constants; (τ_ϕ, τ_θ, τ_ψ) = (0.002, – 0.002, 0.002)Nm. Figures 3a, 3b and 3c correspond to the comparison of the rotation models, we can observe that the models have similar behavior in a small region of operation, then in aggressive flight conditions, for example beyond 1 rad, the divergence of the linear model is evident.

Control system simulation

In the closed-loop simulation, each of the controllers designed through the different models is applied to the complete non-linear dynamics of the rotation system, equation (17).

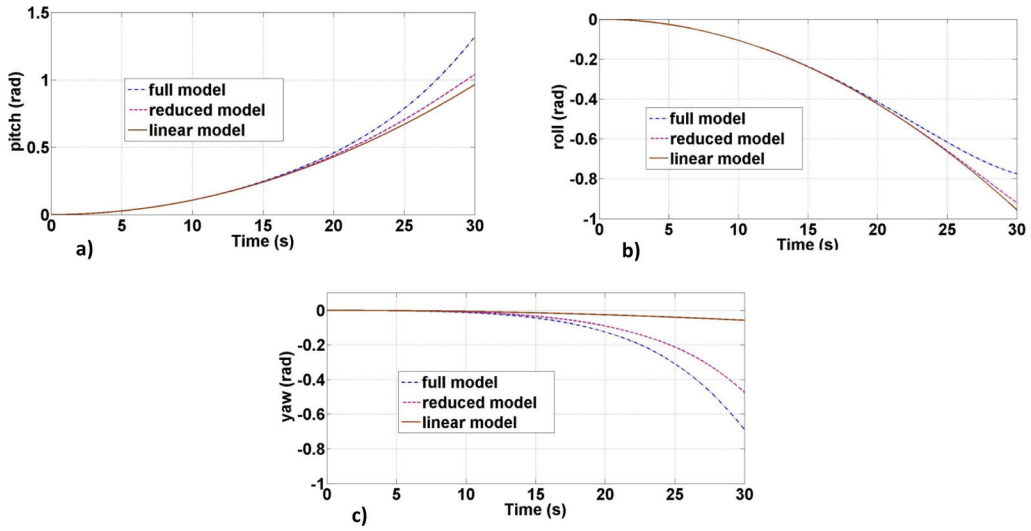

Figure 3. Open-loop comparison of the behavior of each model, a) Pitch axis, b) roll axis, c) yaw axis

To compare the response of the different controllers; the performance index is calculated under the $\mathcal{L}_2$ (Reyes, 2011) norm, equation (64). The initial conditions are established as: $(\phi, \theta, \psi) = (0.3, 0.3, 0.3)$ *rad* and $(\dot{\phi}, \dot{\theta}, \dot{\psi}) = (0.3, 0.3, 0.3)$*rad/s* and , the gains of the controllers, are detailed in Table 2.

$$\mathcal{L}_2 = [\tilde{q}] = \sqrt{\frac{1}{T}\int_0^T \left\|\tilde{q}(t)\right\|^2 dt} \tag{64}$$

Figure 4 corresponds to the comparison of the response of the different controllers in the simultaneous movements in each axis of the aircraft. The overlapping responses of each controller are observed, only with the performance index presented in 4d, we can distinguish that the control designed with the reduced model offers the best performance.

Simulation of the control system and filtering system

Figure 5 illustrates through a block diagram the methodology used in the simulation of the noise process and filtering system. Table 2 defines the controller and observer gains.

Figures 6b to 6d, show the behavior with the reduced model in the filtering and control system for simultaneous movements in the axes of rotation of the aircraft, when is adding the white noise signal to the states of the system defined in 6a. The behavior of the reduced model offers the best performance for the controller and the filtering system as seen in Figure 7a and 7b. To obtain and compare the performance index of the control system, the different models are applied to the control and the filtering stage, then to obtain the performance index of the filtering stage, we proceed similarly.

We can see en Figure 7, that filter system with observers or Kalman filters is very similar, this can be explained because the operation of the aircraft is considered in a small region, even if there are combined movements in the rotations. Due to the implementation of the observer is simpler than the Kalman filter. Figure 8 shows the comparison of the controller and filtering through observers with the different models for combined or simultaneous movements in each axis of rotation.

It has shown that the performance offered by the control system and filtering system with the linear model is similar to that obtained with a Kalman filter or an observer using the full or reduced model. Due that it is simpler to implement a filtering and control system based on the linear model, we chose to implement the experimental part with this model.

Also, in order to compare the proposed control that bases its design on the approximate linear model of the aircraft with a control technique that does not require a dynamic model for its design, a comparison with a fuzzy logic controller is proposed.

Fuzzy logic control

The fuzzy-logic controller design is based, on producing changes in the control signal through the system error and its change reason, i.e. a PD system. The membership functions are presented in Figure 9. Where the following variables are defined for the error, its deriva-

https://doi.org/10.22201/fi.25940732e.2021.22.2.015

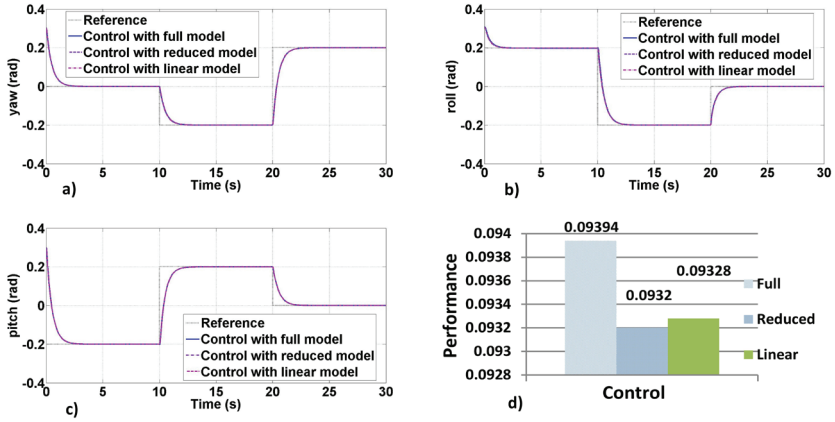

Figure 4. Response of the controllers acting on the complete model, a) Pitch angle, b) roll angle, c) yaw angle, d) performance index

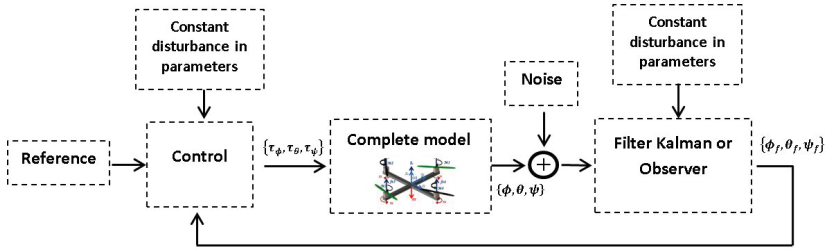

Figure 5. Block diagram of the simulation, control system and filtering system

Table 2. Gains of the control system and filtering system

Gains
$K_u = \begin{pmatrix} 30 & 0 & 0 \\ 0 & 30 & 0 \\ 0 & 0 & 30 \end{pmatrix}$ $K_i = \begin{pmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{pmatrix}$
$K_b = \begin{pmatrix} 20 & 0 & 0 \\ 0 & 20 & 0 \\ 0 & 0 & 20 \end{pmatrix}$ $K_o = k_c = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$

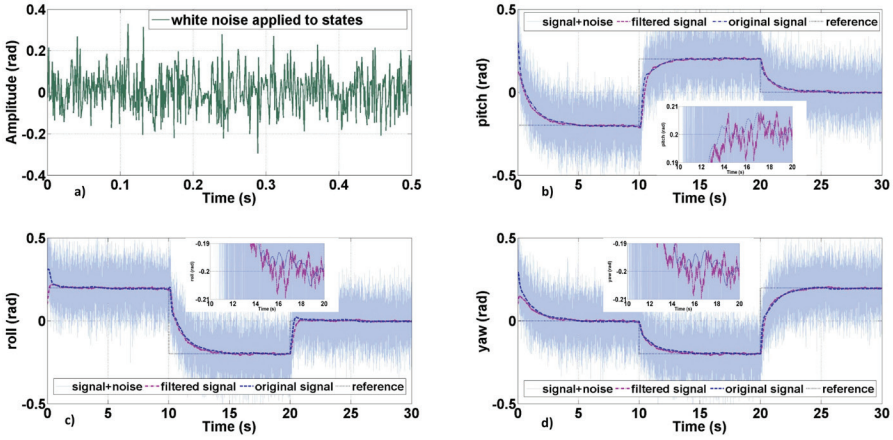

Figure 6. Response to the noise of the control system and filtering system with a reduced model, a) Noise added to the system states, b) pitch, c) roll, d) yaw

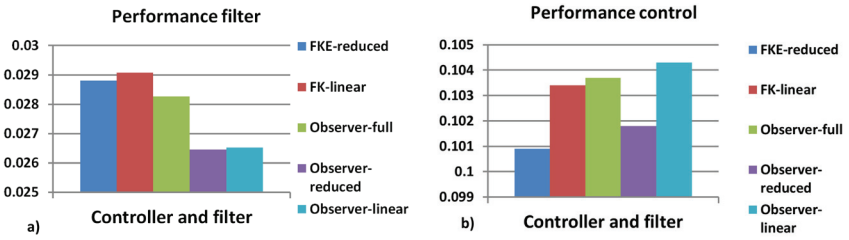

Figure 7. a) Control performance with different models and filters, b) filtering performance with different models and filters

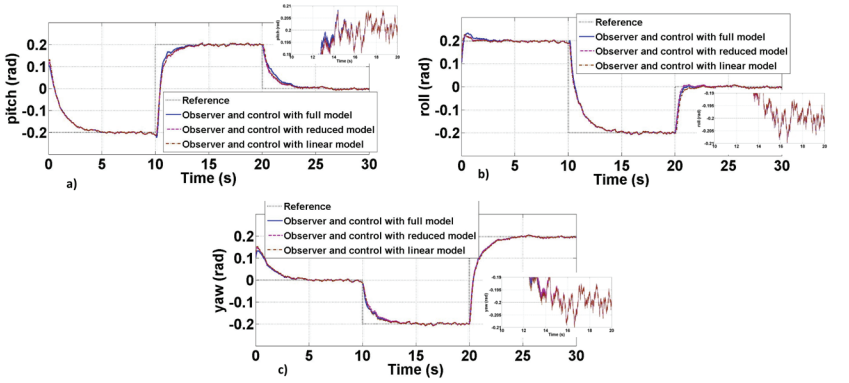

Figure 8. Comparison of the controller and filtering using observers with the different models for combined or simultaneous movements in each axis of rotation, a) Pitch angle, b) roll angle, c) yaw angle

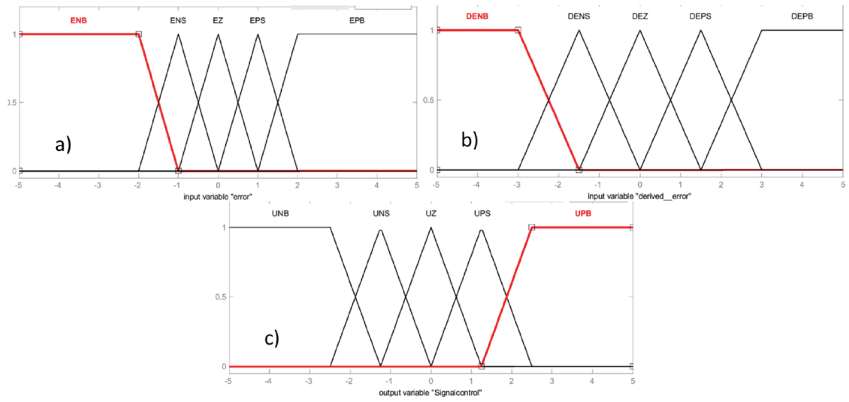

Figure 9. Membership functions, a) Error, b) derived error, c) signal control

tive, as well as the control or output signal; Negative Big (NB), Negative Small (NS), Zero (Z), Positive Small (PS) and Positive Big (PB).

The inference rules are directly related to the linguistic variables defined in the membership functions and are presented in Table 3. Also, for the controller design and the defuzzification signals, the Fuzzy Logic Design toolbox from Matlab is used, establishing Sugeno's methodology and centroid defuzzification.

The comparison in a simulation of the proposed controller with the fuzzy controller is shown in Figure 10, as can be seen, any of the controllers achieves the objective of stabilizing the system with similar behavior, however, although the fuzzy controller offers adequate system behavior and not require a model for its design has the disadvantage; requires precise knowledge of the experimental behavior of the system to propose the membership functions and inference rules, likewise, the implementation in a microcontroller requires more computational resources for its execution, mainly in the defuzzification stage, therefore and due to that the behavior offered by the controller proposed as well as the relative simplicity with which it can be implemented in a microcontroller, it is chosen to implement the experimental part with this controller.

Now the next section of the work shows the experimental results of this implementation.

Table 3. Inference rules

$e(t)$ / $d\,e(t)/dt$	NB	NS	Z	PS	PB
NB	NB	NS	NS	NS	Z
NS	NS	NS	NS	Z	PS
Z	NS	NS	Z	PS	PS
PS	NS	Z	PS	PS	PS
PB	Z	PS	PS	PS	PB

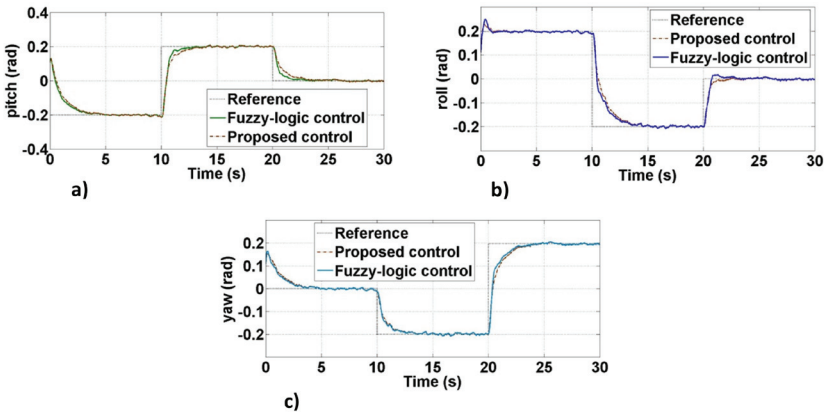

Figure 10. Comparison of the proposed controller and fuzzy-logic controller, a) Pitch angle, b) roll angle, c) yaw angle

Microcontroller implementation and experimental results

The implementation is done according to the block diagram of Figure 11. The signals of the IMU are interpreted and filtered in the microcontroller, also the control signals are calculated using a linear model, the equations that define the linear observer are solved by the Runge Kutta numerical method at an operation step of 0.001s. The experimentation is divided into two stages. The first one, through a test bench that allows rotations in the aircraft's pitch, roll and yaw axes but restricts translational movements, the second stage in a quasi-static flight in which only the rotation movements are controlled around the origin and the translational movements of the aircraft are left free. Figure 12 shows both stages of experimentation.

Experimental results on the test bench

The experimental results of the implementation of the controller and filtering stage with the test bench are shown in Figure 13.

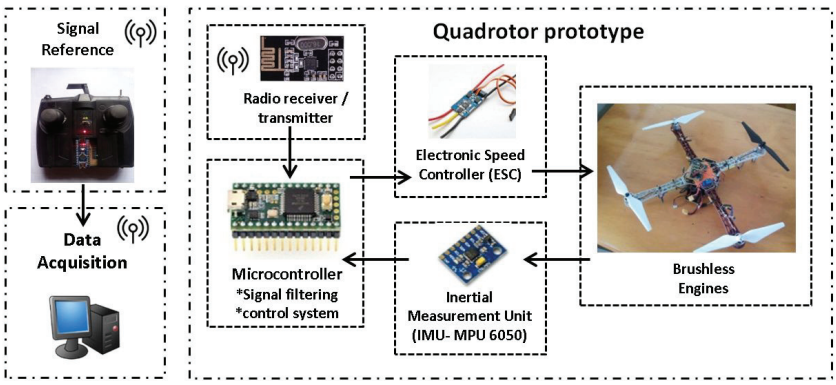

Figure 11. Block diagram of the implementation

Experimental results in quasi-static flight

The experimental results of the implementation of the controller and filtering stage in a quasi-static flight are shown in Figure 14.

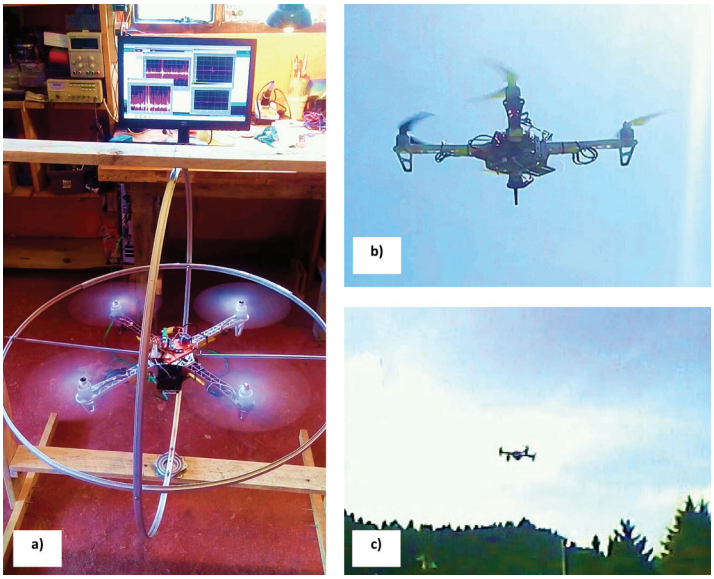

Figure 12. Experimentation, a) test bench, b) and c) quasi-static flight

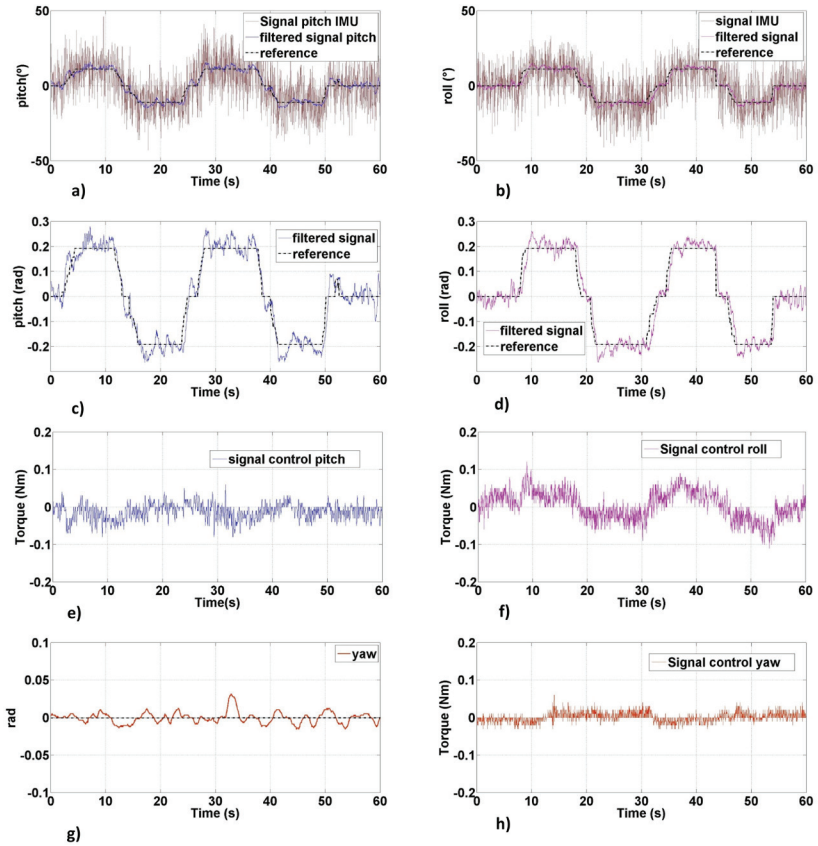

Figure 13. Results of the experimental stage on the test bench, a) Filtered signal for the pitch angle, b) filtered signal for the roll angle, c) trajectory tracking on the pitch axis, d) trajectory tracking on the roll axis, e) control signal for the pitch movement, f) control signal for the roll movement, g) yaw angle stabilization, h) control signal for the yaw movement

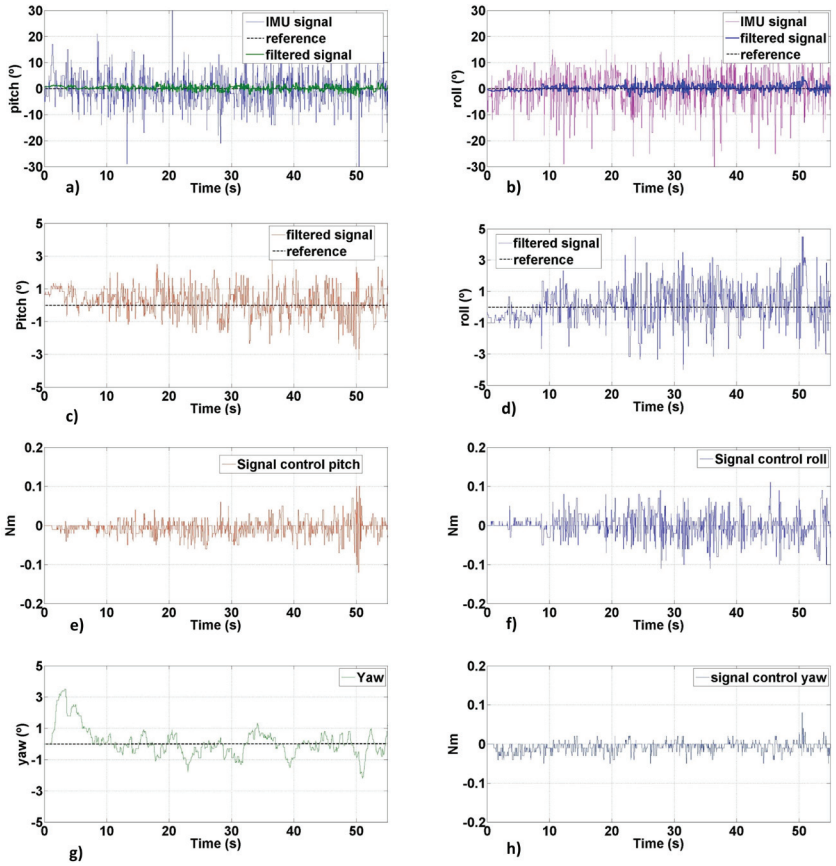

Figure 14. Experimental results of the quasi-static flight stage. a) Signal filtering for pitch angle, b) signal filtering for roll angle, c) pitch axis stabilization, d) roll axis stabilization, e) control signal for pitch movement, f) control signal for roll movement, g) yaw angle stabilization, h) control signal for the yaw movement

Conclusions

In this work, a dynamic model was developed using Euler-Lagrange for a quadcopter type aircraft, then a reduced model was obtained from the complete model, as well as a linear model. Afterward, its open-loop response was compared, observing that for an operating region less than 1 rad, the models behave similarly, later, controllers were designed using the computed torque technique based on the various models and a PID control, the controllers were compared through their performance index, it was determined that anyone controller achieves the objective of stabilizing or tracking the trajectory, later filtering systems were analyzed using observers and Kalman filters, where for each case the three analyzed models were applied, then it was compared the response of the controller and filtering system through a performance index when noise is added to the states and constant disturbances in the parameters, with this it was determined that a linear model is sufficient to design a controller and filtering system, which implies simple implementation and reduction of computational resources for operation with a microcontroller. Additionally, the proposed controller was compared in simulation with a fuzzy logic controller, it was observed that both controllers have similar behavior, however, the proposed controller is easy to implement in a microcontroller, and the stability of the system and filtering is justified with the mathematical analysis presented, against the fuzzy logic controller, for this reason, it was determined to implement the experimental part with said controller. Subsequently, the control and filtering system was implemented for a low-cost IMU and a 32-bit ARM microcontroller, the experimental results showed a behavior similar to the simulation, which validates the use of a linear model to stabilize and track trajectories in the operation of a non-aggressive or aerobatic flight of the aircraft. However, in the experimental part, it was noted that a linear model is sufficient to design the filter and controller, but a control strategy is necessary to give the system robustness to the disturbances caused by the wind or by the disturbances caused by no to know exactly the thrust force of each motor-propeller pair. It is also recommended to design an algorithm that allows improving the performance of the controller estimating the thrust forces of each motor.

Annexed

Glossary

ϕ: is pitch angle.
θ: is roll angle.
ψ: is yaw angle.
$\{c\}$: is the reference axis of the system.
$\{a\}$: is frame with same orientation of inertial frame.
Ω: is the total angular velocity of the quadrotor.
$\begin{pmatrix} p \\ q \\ r \end{pmatrix}$: are the state of the vector Ω.
F_T: is the thrust force of the aircraft.
f_{ei}: is the aerodynamic drag force.
ω_i: is the angular velocity of each engine.
k_a, k_e: are positive definite constants that depend on the density of the air, the radius of rotation, the area and the shape of the propeller blades, as well as other factors.
${}^{I}f$: is defined like to the lift force in the inertial reference frame.
J: is the moment of inertia of the engine-propeller pair.

$\begin{pmatrix} \tau_{G\phi} \\ \tau_{G\theta} \\ \tau_{G\psi} \end{pmatrix}$: are gyroscopic effects.

$\begin{pmatrix} F_{e1} \\ F_{e2} \\ F_{e3} \\ F_{e4} \end{pmatrix}$: these are the forces in each engine.

$C(n,\dot{\eta})$: is definded as the Coriolis matrix.
$M(\eta)$: represents the inertia matrix.
τ_G: is the gyroscopic effects associated with rotation of the propellers.

$\begin{pmatrix} \dot{\eta} \\ \ddot{\eta} \end{pmatrix}$: is the dynamics of rotation in state variables.

$\bar{x}$: is the state vector and the equilibrium input vector.
x_δ: is defined as the incremental states.
$\boldsymbol{u}_\delta$: is defined as the incremental input.
$\dot{\boldsymbol{x}}_\delta$: is the linearized system around an equilibrium point.
wk, *vk*: are the process and measurement noise respectively.
$\hat{x}_k$: is the vector of estimated states.
y_k: is the measurement matrix.
K_k: is the filter gain matrix.
Q: is the covariance of the noise process w_k.
R: is the covariance of the noise in the measurement v_k.
I: is the identity matrix.

References

Agho, C. (2017). Dynamic model and control of quadrotor in the presence of uncertainties.

Amoozgar, M. H., Chamseddine, A. & Zhang, Y. (2013). Experimental test of a two-stage Kalman filter for actuator fault detection and diagnosis of an unmanned quadrotor helicopter. *Journal of Intelligent & Robotic Systems*, 70(1-4), 107-117.

Balasubramanian E. & Vasantharaj R. (2013). Dynamic modeling and control of quadrotor. *International Journal of Engineering and Technology (IJET)*, 5(1), 01-07.

Bauer, P. & Bokor, J. (2008). LQ Servo control design with Kalman filter for a quadrotor UAV. *Periodica Polytechnica Transportation Engineering*, 36(1-2), 9-14.

Beard, R. W. & McLain, T. W. (2012). *Small unmanned aircraft: Theory and practice*. Princeton university press.

Bouabdallah, S., Noth, A. & Siegwart, R. (2004). PID vs LQ control techniques applied to an indoor micro quadrotor. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566) 3, 2451-2456. IEEE.

Cárdenas G. (2015). Drones Ciencia al Vuelo. ¿Cómo vez?, UNAM, México, 2015.

Carrillo, L. R. G., López, A. E. D., Lozano, R. & Pégard, C. (2012). *Quad rotorcraft control: vision-based hovering and navigation*. Springer Science & Business Media.

Castillo, P., García, P., Lozano, R. & Albertos, P. (2007). Modelado y estabilización de un helicóptero con cuatro rotores. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 4(1), 41-57. https://doi.org/10.1016/S1697-7912(07)70191-7

De Jesús-Rubio, J., Cruz, J. H. P., Zamudio, Z. & Salinas, A. J. (2014). Comparison of two quadrotor dynamic models. *IEEE Latin America Transactions*, 12(4), 531-537.

Elruby, A. Y., El-Khatib, M. M., El-Amary, N. H. & modeling Hashad, A. I. (2012). Dynamic and control of quadrotor vehicle. In The International Conference on Applied Mechanics and Mechanical Engineering 15(5). International Conference on Applied Mechanics and Mechanical Engineering. 1-10. Military Technical College.

Emran, B. J., Al-Omari, M., Abdel-Hafez, M. F. & Jaradat, M. A. (2015). Hybrid low-cost approach for quadrotor attitude estimation. *Journal of Computational and Nonlinear Dynamics*, 10(3). https://doi.org/10.1115/1.4028524

Escobedo, J. O., Garcia, E. C., Páramo, L. A., Meda, J. A. & Tapia, R. (2018). Theoretical application of a hybrid observer on altitude tracking of quadrotor losing GPS signal. *IEEE Access*, 6, 76900-76908.

Euston, M., Coote, P., Mahony, R., Kim, J. & Hamel, T. (2008). A complementary filter for attitude estimation of a fixed-wing

UAV. IEEE/RSJ international conference on intelligent robots and systems 340-345.

Falconi, R. & Melchiorri, C. (2012). Dynamic model and control of an over-actuated quadrotor uav. *IFAC Proceedings*, 45(22), 192-197.

Floreano, D. & Wood, R. J. (2015). Science, technology and the future of small autonomous drones. *Nature*, 521(7553), 460-466.

Garcia, P. C., Lozano, R. & Dzul, A. E. (2006). *Modelling and control of mini-flying machines.* Springer Science & Business Media.

Gautam, D. & Ha, C. (2013). Control of a quadrotor using a smart self-tuning fuzzy PID controller. *International Journal of Advanced Robotic Systems*, 10(11), 380.

Gośliński, J., Giernacki, W. & Gardecki, S. (2013, June). Unscented Kalman Filter for an orientation module of a quadrotor mathematical model. 9th Asian Control Conference (ASCC) 1-6. IEEE.

Gupta, S. G., Ghonge, D. & Jawandhiya, P. M. (2013). Review of unmanned aircraft system (UAS). *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, 2.

Gutiérrez, A. R. H., Duarte, M. A. T. & France, R. G. (2017). El uso de drones en ciencias de la tierra.

Hanley, D. J. & Bretl, T. W. (2016). An improved model-based observer for inertial navigation for quadrotors with low cost imus. In AIAA Guidance, Navigation, and Control Conference 0105.

Hossain R., Rideout D. & Krouglicof N. (2010). Bond graph dynamic modeling and stabilization of a quadrotor helicopter. *ResearchGate.*

Jung, D. & Tsiotras, P. (2007). Inertial attitude and position reference system development for a small UAV. In AIAA Infotech@ Aerospace 2007 Conference and Exhibit 2763.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.

Lendek, Z., Berna, A., Guzmán-Giménez, J., Sala, A. & García, P. (2011). Application of Takagi-Sugeno observers for state estimation in a quadrotor. 50th IEEE Conference on Decision and Control and European Control Conference 7530-7535. IEEE.

Loianno, G., Brunner, C., McGrath, G. & Kumar, V. (2016). Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and IMU. *IEEE Robotics and Automation Letters*, 2(2), 404-411.

Macdonald, J., Leishman, R., Beard, R. & McLain, T. (2014). Analysis of an improved IMU-based observer for multirotor helicopters. *Journal of Intelligent & Robotic Systems*, 74(3-4), 1049-1061.

Markley, F. L. (2003). Attitude error representations for Kalman filtering. *Journal of guidance, control, and dynamics*, 26(2), 311-317. https://doi.org/10.2514/2.5048

Muñoz, L. E., Castillo, P. & García, P. (2013). Observer-control scheme for autonomous navigation: Flight tests validation in a quadrotor vehicle. International Conference on Unmanned Aircraft Systems (ICUAS) 95-804. IEEE.

Nonami, K., Kendoul, F., Suzuki, S., Wang, W. & Nakazawa, D. (2010). *Introduction. In Autonomous Flying Robots*. Japan: Springer. 1-29

Noordin, A., Basri, M. A. M. & Mohamed, Z. (2018). Sensor fusion algorithm by complementary filter for attitude estimation of quadrotor with low-cost IMU. *Telkomnika*, 16(2), 868-875.

Paiva-Peredo, E. A. (2016). *Modelado y control de un cuadricóptero.*

Patete, A. & Erazo, A. (2016). Linear and nonlinear control for a micro unmanned aerial vehicle with quadrotor configuration. *Revista Ingeniería UC*, 23(3), 247-259.

Raffo, G. V. (2007). *Modelado y Control de un Helicóptero Quad-Rotor*. MS, disertación, Dept. telemática, automática y robótica, Univ. de Sevilla. https://www.researchgate.net/derefhttp%3A%2F%2Fdx.doi.org%2F10.1109%2FIROS.2007.4399042

Reyes, F. (2011). *Robótica-control de robots manipuladores*. Alfaomega grupo editor.

Roldan-Caballero A. (2016). *Modelado y monitoreo para un drone tipo cuadrirotor* (Tesis de Maestría en Ingeniería de Manufactura). Instituto Politécnico Nacional. Ciudad de México.

Rodríguez R., Ipanaqué W., Carrera E. (2016).*Comparación de modelos matemáticos y controladores PID vs LQR para un cuadricóptero*. Universidad de Piura, Facultad de Ingeniería.

Sabatino, F. (2015). Quadrotor control: modeling, nonlinearcontrol design, and simulation.

Sadr, S., Moosavian, S. A. A. & Zarafshan, P. (2014). Dynamics modeling and control of a quadrotor with swing load. *Journal of Robotics*, 2014. https://doi.org/10.1155/2014/265897

Sanz, R., Ródenas, L., García, P. & Castillo, P. (2014). Improving attitude estimation using inertial sensors for quadrotor control systems. International Conference on Unmanned Aircraft Systems (ICUAS) 895-901. IEEE.

Sarim, M., Nemati, A., Kumar, M. & Cohen, K. (2015). Extended kalman filter based quadrotor state estimation based on asynchronous multisensor data. In Dynamic Systems and Control Conference 57243. American Society of Mechanical Engineers. https://doi.org/10.1115/DSCC2015-9929

Sebesta, K. D. & Boizot, N. (2013). A real-time adaptive high-gain EKF, applied to a quadcopter inertial navigation system. *IEEE Transactions on Industrial Electronics*, 61(1), 495-503.

Sevilla, L. (2014). *Modelado y control de un cuadricóptero*. Universidad Pontificia de Comillas.

Shi, D., Wu, Z. & Chou, W. (2018). Generalized extended state observer based high precision attitude control of quadrotor vehicles subject to wind disturbance. *IEEE Access*, 6, 32349-32359.

Simon, D. (2006). Using nonlinear Kalman filtering to estimate signals. *Embedded Systems Design*, 19(7), 38.

Sira-Ramírez, H. E. B. E. R. T. T., Márquez, R., Rivas-Echeverria, F. & Llanes-Santiago, O. (2005). *Control de Sistemas no lineales.* Automática & Robótica. Pearson Educación SA.

Tailanián, M., Paternain, S., Rosa, R. & Canetti, R. (2014). Design and implementation of sensor data fusion for an autonomous quadrotor. IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings 1431-1436. IEEE.

Xiong, J. J. & Zheng, E. H. (2015). Optimal kalman filter for state estimation of a quadrotor UAV. *Optik*, 126(21), 2862-2868.

You, W., Li, F., Liao, L. & Huang, M. (2020). Data Fusion of UWB and IMU Based on Unscented Kalman Filter for Indoor Localization of Quadrotor UAV. *IEEE Access*, 8, 64971-64981.

Yu, J., Shi, P., Dong, W. & Yu, H. (2015). Observer and command-filter-based adaptive fuzzy output feedback control of uncertain nonlinear systems. *IEEE Transactions on Industrial Electronics*, 62(9), 5962-5970.

Yoo, T. S., Hong, S. K., Yoon, H. M. & Park, S. (2011). Gain-scheduled complementary filter design for a MEMS based attitude and heading reference system. *Sensors,* 11(4), 3816-3830.