Trajectory Tracking of a Quadcopter UAV with Optimal Translational Control

E.G. Hernandez-Martinez * G. Fernandez-Anaya ** E.D. Ferreira *** J.J. Flores-Godoy **** A. Lopez-Gonzalez *

* Departamento de Ingenierías, Universidad Iberoamericana, Ciudad de México, 01219 México (e-mail: eduardo.gamaliel@ibero.mx)
** Departamento de Física y Matemáticas, Universidad Iberoamericana, Ciudad de México, 01219 México, (e-mail: guillermo.fernandez@ibero.mx)
*** Departamento de Ingeniería Eléctrica, Universidad Católica del Uruguay, Montevideo, 11600 Uruguay (e-mail: enferrei@ucu.edu.uy)
**** Departamento de Matemática, Universidad Católica del Uruguay, Montevideo, 11600 Uruguay (e-mail: jose.flores@ucu.edu.uy)

Abstract: This paper presents a two-level control strategy for the trajectory tracking of the quadcopter. The external loop is designed to move the translational dynamics of the robot to follow smooth parametric functions using an appropriate variables substitution. The outer loop generates the desired trajectories of the orientation angles, which are controlled in the inner loop. The parameter gains are calculated using linear optimal control in order to minimize the energy consumption of the rotors. The control architecture is designed to be compatible with standard electronic platforms of commercial quadcopters and incorporates the purpose of maximizing flight time, an important requirement in real applications of multi-rotor aerial vehicles. Some numerical simulations show the performance of the control approach.

© 2015, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.

Keywords: Quadcopter, Optimal Control, Mobile Robots, Trajectory Tracking.

1. INTRODUCTION

Recently, the multi-rotor aerial vehicles have found a wide field of applications in the areas of police and military surveillance, open field exploration, searching and rescue, aerial filming, packages delivering, etc. The related research areas involves aeronautics, mechatronics, dynamical systems, automatic control, signal processing, wireless communication, etc. The hobbyists market and the academic interest has promoted the costs reduction of sensor, actuators and the development of electronic embedded devices dedicated to these aerial vehicles (Lim, 2012). The main interest of the robot control community is to design control strategies to realize autonomous flight following pre-programmed trajectories in the space, converting the multi-rotor platform, driven initially by a radio control, in a fully Unmanned Autonomous Vehicle (UAV).

The quad-rotor setup is the main multi-rotor configuration studied in the control literature due to the simplicity of the modeling and control respect to other platforms and because has the capacity to realize aggressive maneuvers like flip behaviors. It consists on four electrical rotors with propellers mounted at the ends of a cross-shape structure. The dynamical configuration permits the take off and land maneuvers in reduced spaces, hover above targets and the omnidirectional motion in the space. However, the main disadvantage of the quadcopter is the continuous rotation of the actuators which causes a high electrical current spending reducing the practical flight time, commonly in a range of 12 to 18 minutes in commercial platforms (Aranda, 2014). Also, multi-rotor aircrafts are known to be unstable, subject to effects of vibration and noise all the time, and the dynamical control becomes complex if the effects of air turbulence, propellers deflection and other perturbations are incorporated into the dynamical underactuated system model.

In order to simplify the problem, the kinematics and dynamics of a quadcopter can be modeled, discarding environmental perturbations, using the approaches of Newton-Euler, Euler-Lagrangian and Quaternions Bertrand (2011). It results in a nonlinear dynamical system with four inputs and six degree of freedom (three cartesian coordinates and three orientation angles). Some works like Gonzalez (2013) deals the case of orientation control for hover flight. In Castillo (2004) is considered take-off landing strategies. As mentioned in Garcia (2013), if the orientation dynamics is faster respect to the translational dynamics, the posture and orientation control laws can be separated in two levels. Example of hierarchical scheme is proposed in Mahony (2012), where the Euler angles in the low level are linearized around zero in order to simplify the trajectory tracking in the high level. Similar focus is presented Aranda (2014) with real-time experiments. The works of Castillo et al (2005) and Escareno (2006) present the linearization of the translational dynamics to track trajectories considering saturated inputs and alternatively,
Pykin (2014) adds new auxiliar controls dependant of the Euler angles. A similar linealisation controls technique is given in Minh (2014) which choose a different rotation matrix and does not include the dynamics of the yaw angle. Other works includes the motion coordination of quadcopters teams like Sumano (2013), where the orientation control is approximated in the yaw angle and a consensus error terms are added to the control laws, or the approach of Mahmood (2015), where a change of coordinates is selected to apply a state feedback linearisation. Finally, a wide research about the common PID algorithms programmed in some available embedded controls for multi-rotors is presented in Lim (2012).

Inspired in Escareno (2006), the main contribution of this paper is to propose a separated orientation and posture control to design two input-output linearization control laws, avoiding the approximation of the orientational and translational dynamics, different to the previous works. The control laws clarify the programming of the two control levels in order to be compatible with the standard architecture of commercial embedded controls. Also, the energy saving is addressed using a linear optimal control strategy in in order to maximize the flight time of the quadcopter. Similar works as Ritz (2011); Argentin (2013) add optimal values in the gain parameters of classical controllers, but applied to a reduced linearized quadrotor model only. The performance of the control approach is shown in numerical simulations.

2. KINEMATIC AND DYNAMIC MODEL OF THE QUADCOPTER UAV

A quadcopter vehicle can be conceived as a rigid body with mass and inertia subjected to a gravitational force moving in a 3D space. The movement of the vehicle is generated by the combination of the angular velocities of four electrical motors with propellers. They produce one main thrust force applied in the mass center and three moments simultaneously.

According to the figure 1, the position of the body frame $\Gamma_b$ respect to fixed frame $\Gamma_e$ is given by $\xi = [x, y, z]^T$. The rotational coordinates of $\Gamma_b$ respect to $\Gamma_e$ are defined as $\eta = [\phi, \theta, \psi]^T$. These angles are commonly known as Euler angles (roll, pitch and yaw, respectively). The linear and angular velocities in $\Gamma_b$ are given by $V = [u, v, w]^T$ and $\Omega = [p, q, r]^T$, respectively. The kinematics of the rigid body describes the relationship of $\Gamma_e$ respect to $\Gamma_b$ between the translational and rotational forces. Thus, the translational velocities are given by

$$\dot{\xi} = R_b^e\dot{V}$$

where $R_b^e$ is the result of the multiplication of the three standard rotations matrices, given by

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi - \sin \phi & 0 \\ 0 & \sin \phi - \cos \phi & 0 \end{bmatrix},$$

$$R_y = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix},$$

$$R_z = \begin{bmatrix} \cos \psi - \sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$
where \( \tau = [\tau_\phi, \tau_\theta, \tau_\psi]^T \) constitutes the vector of auxiliary torques. Thus, substituting the rotational matrix \( R_0^e \) given (2) and the torques (7), the dynamical system (5) can be reduced to

\[
m\ddot{\xi} = \begin{bmatrix} F \sin \theta \\ -F \cos \theta \sin \phi \\ F \cos \theta \cos \phi - mg \end{bmatrix}
\]

(8)

\[
\ddot{y} = \ddot{x}
\]

(9)

Observe that the selection of \( R_0^e \) implies that the yaw angle \( \psi \) does not appear in the translational dynamics. This property will be exploited in the control strategy in the next section.

As expected, the main thrust force \( F = f_1 + f_2 + f_3 + f_4 \) where \( f_i, i = 1, \ldots, n \) are the individual thrust forces generated by the \( i \)-th motor as shown in Figure 1. In this paper we consider all rotors and propellers are similar. Note that the rotors 1 and 2 rotate clockwise producing changes in the angle \( \phi \) and an eventual displacement in the \( Y \) axis. Let \( \ell \) be the length of every arm of the quadcopter, then \( \tau_\phi = \ell(f_2 - f_1) \). Similarly, the rotors 3 and 4 rotate counterclockwise moving the angle \( \theta \) and the motion in the \( X \) axis. Thus, \( \tau_\theta = \ell(f_4 - f_3) \). The torque in the yaw angle is produced by the differences of the rotor pairs \( 1-2 \) and \( 3-4 \), and it is given by \( \tau_\psi = \frac{k}{2}[-f_1 - f_2 + f_3 + f_4] \), where \( k \) is the thrust constant of every individual array rotor-propeller, and \( \theta \) is the so-called drag constant of the rigid body (Mahony, 2012). Summarizing, the relation of individual thrust forces and the system inputs of (8)-(9) can be written in matrix form as

\[
\begin{bmatrix} F \\ \tau \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -\ell & \ell & 0 & 0 \\ 1 & 0 & -\ell & \ell \\ b & b & b & b \\ k & -k & -k & k \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}
\]

(10)

In a more extended model, the individual forces \( f_i \) depend on the angular velocity of the rotor and the features of the propeller (length, pitch angle and deflection), the air turbulence and other environment disturbances. These aerodynamical considerations construct in general a nonlinear actuator dynamics. According to Escareno (2006) and discarding transient effects and disturbances, the forces \( f_i \) can be modeled as

\[
f_i = kw_i^2, i = 1, \ldots, 4
\]

(11)

where \( \omega_i \) is the angular velocity of the rotor \( i \), and \( k \) is the thrust constant mentioned before. This static relation is not very restrictive since the dynamics of the actuators is fast with respect to the motion of the vehicle. In the case of electrical three-phase rotors, commonly used in commercial quadcopters, the internal velocity loop and the power circuit is solved by the so-called Electronic Speed Control (ESC), and the manufacturers provide rotor’s datasheets where the force (and indirectly the constant \( k \)) is calculated by experimental setups using different propellers. It is clear that a low value of angular velocity requires a less current consumption.

3. CONTROL STRATEGY

As mentioned before, the quadcopter control can be decomposed in two hierarchical levels (posture control and orientation control) assuming that the rotational dynamics converges faster than the translational dynamics. Thus, the control scheme is depicted in Figure 2. The outer loop is related to the posture control designed to follow a desired trajectory in the space. This external controller generates the desired values of the orientation angles for the inner loop. Then, the orientation control moves the angles to desired values using an Inertial Measurement Unit (IMU) sensor, generating a translational motion of the vehicle. It can be measured using some positioning sensors, like GPS for outdoors or videocameras for indoors, closing the loop for the posture control. The inner control also calculates the desired individual forces and angular velocities, which are implemented for the ESC to the rotors.

The control strategy also is designed to be compatible with the commercial control electronics for the quadcopters. As mentioned in Lin (2012), different dedicated microprocessors, like Arducopter, Mikrocopter, Naze32, etc., are designed to implement the orientation control from a remote control communicated by radiofrequency. The radio control sends the desired values of \( \phi, \theta \) for a displacement in the axis \( XY \), the value of \( F \) to increase or decrease the power to all rotors and modifies indirectly the motion over \( Z \) axis, and the desired value of the yaw angle \( \psi \). Note that these four inputs for the inner loop are provided by the outer control, as shown in the Figure 2. It enables the possibility to switch an automatic posture control and a manual control of the angles via remote control in the same experimental platform using some commercial setups.

The next subsection describes the proposed posture and orientation control. The main contributions of the paper appear in the two control levels. For the posture control, we do not use the standard approximation for the translational dynamics, where the control is bounded in a neighborhood around small values of the Euler angles. Also, the control parameters are defined considering the concepts of optimal control, where it is possible to increase the settling time to avoid large values of the control inputs and consequently, a larger energy consumption. In the inner loop, the dynamics of the angles is also linearized and the optimal parameter gains of the posture control are used to design the convergence time of the orientation angles.

3.1 Posture control

In the translational dynamics given in (8), the altitude control is related to the dynamics of the coordinate \( z \), given by

\[
m\ddot{z} = F \cos \theta \cos \phi - mg.
\]

(12)

In order to linearize the dynamics of this coordinate, define the total thrust \( F \) as

\[
F = \frac{m(r_z + g)}{\cos \theta \cos \phi}.
\]

(13)

Note that substituting (13) in (12), the dynamics of \( z \) now is reduced to \( \ddot{z} = r_z \), where \( r_z \) is an auxiliary control defined below. The motion in the coordinate \( Y \) is governed, according to (8), by the equation

\[
m\ddot{y} = -F \cos \theta \sin \phi.
\]

(14)
Substituting the value of $F$ given in (13) in (14), the dynamics of $y$ is reduced to

$$m\ddot{y} = -m(r_z + g)\tan\phi.$$  \hspace{1cm} (15)

Similar to the previous case, the dynamics of $y$ can be reduced to

$$\phi = \arctan\left(-\frac{r_y}{r_z + g}\right),$$  \hspace{1cm} (16)

and the dynamics of $y$ is reduced to $\ddot{y} = r_y$, where $r_y$ becomes its auxiliary control. Finally, a similar strategy is designed for the motion in the $x$ axis given by

$$m\ddot{x} = F \sin \theta,$$  \hspace{1cm} (17)

that can be reduced, substituting (13) in (17), to

$$m\ddot{x} = m(r_z + g)\tan\theta \cos\phi.$$  \hspace{1cm} (18)

Note that the $x$ dynamics can be reduced to $\ddot{x} = r_x$ when the value of $\theta$ is given by

$$\theta = \arctan\left(\frac{r_x \cos \phi}{r_z + g}\right).$$  \hspace{1cm} (19)

In order to obtain an optimal control law in the outer loop, define the position errors as

$$e_x = x - x_d,$$
$$e_y = y - y_d,$$
$$e_z = z - z_d$$  \hspace{1cm} (20)

where $x_d(t)$, $y_d(t)$ and $z_d(t)$ are the desired twice differentiable trajectories for the coordinates $x$, $y$ and $z$, respectively. Recalling the reduced translational dynamics using (13), (16) and (19), the accelerations of these errors are given by

$$\ddot{e}_x = r_x - \ddot{x}_d,$$
$$\ddot{e}_y = r_y - \ddot{y}_d,$$
$$\ddot{e}_z = r_z - \ddot{z}_d.$$  \hspace{1cm} (21)

Defining new auxiliary inputs $\dot{\hat{x}} = r_x - \ddot{x}_d$, $\dot{\hat{y}} = r_y - \ddot{y}_d$ and $\dot{\hat{z}} = r_z - \ddot{z}_d$, the dynamics of the errors in matrix form are given by

$$\dot{e} = Ae + B\hat{\dot{\hat{r}}}$$  \hspace{1cm} (22)

where $e = [e_x, \dot{e}_x, e_y, \dot{e}_y, e_z, \dot{e}_z]^T$, $\hat{\dot{r}} = [\dot{\hat{x}}, \dot{\hat{y}}, \dot{\hat{z}}]^T$, $A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$, $B = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$.

Then, a static state feedback $\dot{\hat{r}} = -K\hat{e}$ can be defined, where the matrix $K$ is calculated to minimize the cost function

$$J = \int_{0}^{\infty} (e^T Q e + \hat{\dot{r}}^T R \hat{\dot{r}}) \, dt,$$  \hspace{1cm} (23)

where $Q = I_{6 \times 6}$ and

$$R = \begin{bmatrix} \mu_x & 0 & 0 & 0 \\ 0 & \mu_y & 0 & 0 \\ 0 & 0 & \mu_z \end{bmatrix},$$  \hspace{1cm} (24)

where the parameters $\mu_x, \mu_y, \mu_z > 1$ can be adjusted to highlight the effect of the input values respect to the time convergence of the position errors. Using the standard optimal approach of Linear Quadratic Regulator (LQR) given in Argentim (2013), the matrix $K = R^{-1}B^TP$, where $P$ is the solution of the Ricatti equation $A^TP + PA - PBR^{-1}B^TP + Q = 0$. The gain matrix $K$ has the form

$$K = \begin{bmatrix} k_{1x} & k_{2x} & 0 & 0 & 0 & 0 \\ 0 & 0 & k_{1y} & k_{2y} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & k_{1z} \end{bmatrix},$$  \hspace{1cm} (25)

therefore, the control inputs result in

$$r_x = \ddot{x}_d + \ddot{r}_x = \ddot{x}_d - k_{2x}\dot{e}_x - k_{1x}e_x,$$
$$r_y = \ddot{y}_d + \ddot{r}_y = \ddot{y}_d - k_{2y}\dot{e}_y - k_{1y}e_y,$$
$$r_z = \ddot{z}_d + \ddot{r}_z = \ddot{z}_d - k_{2z}\dot{e}_z - k_{1z}e_z.$$  \hspace{1cm} (26)

Note that substituting (26) in (21), result three second order linear functions of the errors ensuring their convergence to zero. The settling time and damping coefficient for the three error are given respectively by

$$t_{sx} = \frac{10}{k_{2x}}, t_{sy} = \frac{10}{k_{2y}}, t_{sz} = \frac{10}{k_{2z}},$$  \hspace{1cm} (27)

$$\xi_x = \frac{k_{2x}}{2\sqrt{k_{1x}}}, \xi_y = \frac{k_{2y}}{2\sqrt{k_{1y}}}, \xi_z = \frac{k_{2z}}{2\sqrt{k_{1z}}}. $$  \hspace{1cm} (28)
As expected, a great value of $\mu_x$, $\mu_y$ and $\mu_z$ increases the settling time and the time convergence. However, it generates low values of the control inputs which is translated in less angular velocities of the rotors, saving energy and maximizing the flight time. The parameters $\mu_x$, $\mu_y$ and $\mu_z$ must be adjusted according to the rate change of the desired trajectories to achieve the gathering between and acceptable convergence time and the energy consumption.

Observe that the posture control generates the desired values of $F$, $\phi$, and $\theta$, which becomes in the desired setpoints for the orientation control (inner loop). In the computational order of these control laws, the value of $\phi$ (depending of $r_y$ only) is calculated in a first step according to (16). It is substituting in (19) to calculate $\theta$, and finally the values of the two previous angles are substituting in (13) to find the value of $F$. Note that the posture control requires only the coordinate $\xi$ and the information of the desired trajectories $x_d$, $y_d$ and $z_d$. Remember that the yaw angle $\psi$ do not appear in the translational dynamics as mentioned before. It is addressed in the next section.

### 3.2 Orientation control

Define as $\eta_d = [\phi_d, \theta_d, \psi_d]$ the desired twice differentiable trajectories for the orientation angles $\eta$. As show in the figure 2, the desired values of $\phi_d$, $\theta_d$ are provided by the posture control through (16) and (19). The desired trajectory $\psi_d$ for the yaw angle can be defined by the user because remains independent of the posture control. A possible selection of $\psi_d$ could be define to achieve a yaw angle according to the velocity vector generated by $x_d$ and $y_d$, i.e. $\psi_d = \arctan \frac{x_d}{y_d}$, and the vehicle will point to the forward direction of the path in the XY plane.

Define the auxiliary control inputs given in (9) as

$$
\begin{align*}
\tilde{\tau}_\phi &= \ddot{\phi}_d - k_{2\phi}(\phi - \phi_d) - k_{1\phi}(\phi - \phi_d) \\
\tilde{\tau}_\theta &= \ddot{\theta}_d - k_{2\theta}(\theta - \theta_d) - k_{1\theta}(\theta - \theta_d) \\
\tilde{\tau}_\psi &= \ddot{\psi}_d - k_{2\psi}(\psi - \psi_d) - k_{1\psi}(\psi - \psi_d)
\end{align*}
$$

(29)

Note that substituting (29) in (9), it is constructed a linear equation of the angle errors given by $e_\phi = \phi - \phi_d$, $e_\theta = \theta - \theta_d$ and $e_\psi = \psi - \psi_d$. To ensure a fast convergence of the orientation angles without overdamped responses, it is suggested to define

$$
\begin{align*}
k_{2\phi} &= k_{2\theta} = k_{2\psi} = \frac{10}{\rho (\min \{t_{ssx}, t_{ssy}, t_{ssz}\})}, \\
\sqrt{k_{2\phi}} &= k_{1\phi}, \sqrt{k_{2\theta}} = k_{1\theta}, \sqrt{k_{2\psi}} = k_{1\psi}
\end{align*}
$$

(30)

(31)

where $0 < \rho < 1$ to establish a less settling time respect than the transational convergence. Finally, using the values of $F$ obtained in (13) and the three main torques of (29), the angular velocities of the rotors send to the ESC’s, using the equation (10) and substituting the values of (11), become in

$$
\begin{bmatrix}
\omega^2_z \\
\omega^2_y \\
\omega^2_x
\end{bmatrix} = \frac{1}{k} \begin{bmatrix}
1 & 1 & 1 \\
-\ell & \ell & 0 \\
1 & 0 & -\ell \\
0 & b & b \\
-k & -k & k
\end{bmatrix}^{-1} \begin{bmatrix}
F \\
\tau
\end{bmatrix}
$$

(32)

Fig. 3. Trajectory of the quadcopter in the space where $\tau$ is calculated from (7) and (29).

### 4. NUMERICAL SIMULATION

Figures 3-6 show a numerical simulation of the control approach using Matlab-Simulink©. The desired trajectory is a circular path for the coordinates $x$ and $y$, i.e. $m_{x_d} = -2 + 2\cos(\frac{\pi}{6}t)$, $m_{y_d} = 2\sin(\frac{\pi}{6}t)$ (in meters), and a smooth polynomial function for $z$, i.e. $m_{z_d} = \frac{3}{30}t^2 - \frac{2}{30}t^3$ for $t < 30$ and $m_{z_d} = 1$, for $t \geq 30$. Also, is defined $\psi_d = 0$, $t > 0$. The initial conditions are $\xi(0) = [0, 0, 0]^T$ and $\eta(0) = [0, 0, \frac{\pi}{4}]$.

For the posture control, $\mu_x = \mu_y = \mu_z = 2$, producing the optimal gains $k_{1x} = k_{1y} = k_{1z} = 0.7071$ and $k_{2x} = k_{2y} = k_{2z} = 1.3836$. Thus, the settling time and damping are fixed respectively by $t_{ssx} = t_{ssy} = t_{ssz} = 7.2278$ and $\xi_x = \xi_y = \xi_z = 0.8227$. Then, for the orientation control, the parameter gains are selected with $\rho = 0.2$ as $k_{2\phi} = k_{2\theta} = k_{2\psi} = \frac{10}{\rho(7.2278)} = 6.9177$ and $k_{1\phi} = k_{1\theta} = k_{1\psi} = \sqrt{\frac{6.9177}{2}} = 1.8597$.

As shown in the figure 3 and figure 4 the quadcopter achieves the desired trajectory and the orientation angles follow the desired values imposed by the posture control. Also, the convergence to the desired values is shown in the figure 5 where the errors converge to zero. Finally, the figure 6 shows the control inputs generated by the two control levels.

### 5. CONCLUSION

This paper presents a hierarchical two-level control scheme for the trajectory tracking of the quadcopter UAV. In the posture control, a change of variables is designed to generate the desired values of the angles and the main thrust force for the orientation control. The control laws do not use some approximation of the original nonlinear dynamics of the vehicle. The parameter gains are selected according to the LQR method in order to increase the settling time and reduce the magnitude of inputs in order to save energy. On the other hand, the orientation control defines its gain parameters according to the optimal gains of the outer loop. It ensures that the orientation dynamics
energy and maximizing the flight time. The parameters it generates low values of the control inputs which is the settling time and the time convergence. However, it is suggested to define trajectories for the orientation angles $\phi$.

In (16), it is substituting in $\phi$.

Also, the convergence to the desired values is shown in figure 3 and figure 4. The quadcopter becomes in using the equation (10) and substituting the values of (11), (29), the angular velocities of the rotors send to the ESC’s, obtained in (13) and the three main torques of $\tau$.

Thus, the settling time and damping appear in the translational dynamics as $k$. Note that the posture control, a change of variables is designed to obtain the optimal gains $k$.

As expected, a great value of $\eta$ user because remains independent of the posture control. Also, the posture control, a change of variables is designed to obtain the optimal gains $k$.

The design of the control laws can be compatible with the commercial platforms for the construction of UAV’s. As further research, the control scheme will be tested in a real quadcopter.

ACKNOWLEDGEMENTS

The author acknowledges the financial support from CONACyT by the project grant 216763, and the Universidad Iberoamericana, Mexico City.

REFERENCES


