Modeling, Simulation and Implementation of a modified PID Controller for stabilizing a Quadcopter

Ernesto Paiva, Juan Soto, Julio Salinas and William Ipanaqué, *Member, IEEE*

***Abstract*— This paper presents the mathematical modeling of a quadcopter. A modified PID control algorithm in Simulink is developed, observing the system's behavior in a virtual environment. PID controller is implemented in an embedded system. The quadcopter is attached to a test bench for submission to external forces. Finally, real test on outdoor scenarios have been made to validate the proposed PID controller.**

***Index Terms*—Quadrotor, modelling, attitude, PID.**

I. Introduction

In [1], it explained that the significant progress in the last decade in detection technologies, energy storage, miniature sensors and data processing have made the development of unmanned aerial vehicles possible.

In 2004, research at the Swiss Federal Institute of Technology in Zurich led to publications such as Design and Control of an Indoor Micro Quadrotor [2] in which the project called Omnidirectional Stationary Flying Outstretched Robot (OS4) is presented, making a comparison between different types of UAVs, shows the dynamic model of the quadcopter and rotors. In [1], Bouabdallah and his team presented an article where a series of simulation and experimental tests of his OS4 is performed implementing the classical PID control algorithm and advanced control LQ. It is published in 2005 [3] and improvements in OS4 hardware and results obtained using Lyapunov control are presented. Furthermore, the simulation in three dimensions of a quadcopter in Webots software is presented.

The team led by Dr. Samir Bouabdallah is one of the pioneers in topics regarding the control of quadcopters. The vast majority of his scientific articles are based on his doctoral thesis [4]. The work of Dr. Bouabdallah is one of the main reference sources for this work. Moreover, Herisse [5] presents flight stabilization, and secondly an autoland by using an optical flow sensor as feedback information.

Another important work is published by Rubio [6], where a PID controller is tuned by using H-infinity controller standards. This work takes place in a simulation environment and shows robustness in stabilizing the UAV (Unmanned Aerial Vehicle) against external disturbances.

E. Paiva is with Universidad de Piura, Av. Ramón Mugica 131, Piura, Perú (corresponding author e-mail: ernesto.paiva@udep. pe).

J. Soto is with Universidad de Piura, Av. Ramón Mugica 131, Piura, Perú (e-mail: juan.soto@udep.pe).

J. Salinas is with Universidad de Piura, Av. Ramón Mugica 131, Piura, Perú (e-mail: julio.salinas@pregrado.udep.udep.pe)

W. Ipanaqué is with Universidad de Piura, Av. Ramón Mugica 131, Piura, Perú (e-mail: william.ipanaque@udep.pe).

9781509011476/16/$31.00©2016IEEE

This article describes the design and implementation of a modified PID control algorithm resistant to wind disturbances in a quadcopter, comparing the robustness in simulations with the results in [6]. In many articles such as [5], [6], [7], [8], [9], [10], [11], [12] and [13] the motor's dynamics is not considered. This paper studies the methodology described in Boaudballah's publications since the nature of the quadcopter's motors under study has a considerable time constant. In [14], the mathematical modeling of a quadcopter and the stabilization and position PID control algorithms are presented. Unlike the above publications, this article show flight tests performed outdoors.

The aerodynamic model of a quadcopter under the Newton-Euler formulation is explained in Section II. Section III shows the use of the experimental approach used to determine the aerodynamic constants of the quadcopter, and through SolidWorks software, its inertias are determined. Section IV explains the logic of orientation control. The dynamics of the motors is embedded on the equations governing the orientation - very few times seen in previous work. Section V shows the results of the PID controller modified through simulations, and the closed-loop process is validated with experimental data. To check its good performance, external disturbances for the quadcopter are added and flight tests are performed outdoors.

II. Mathematical Modelling of Quadcopter

A. Quadcopter Configuration

The quadcopter can be described as a UAV with four propellers in x-type or cross configuration. In the cross configuration the two pairs of propellers (1,3) and (2,4) rotate in opposite directions, which eliminates the need for a tail rotor [15], see Fig. 1. By varying the rotor speed, the thrust force can be changed and movement can be created [2], see Fig. 1.

B. Dynamic Configuration of Quadcopter

Fig. 1 shows the forces of each of the rotors that produce the roll (φ angle), pitch (θ angle) and yaw (ψ angle) movements. Also shown is the quadcopter's scheme with a fixed coordinate system and an inertial reference coordinate system on its mass center $\vec{\boldsymbol{B}} = [\overrightarrow{X_L}, \overrightarrow{Y_L}, \overrightarrow{Z_L}]$ and an inertial reference coordinate system $\vec{\boldsymbol{I}} = [\vec{X}, \vec{Y}, \vec{Z}]$ which is considered fixed to the ground, where $\vec{X}$ is the tangent axis to the magnetic meridian and is positive when pointing to the magnetic north pole, $\vec{Z}$ is the local vertical axis pointing up (decreasing direction of gravity g), $\vec{Y}$ is the axis defined such that it forms a direct trihedron with $\vec{X}$ and $\vec{Z}$. [16].

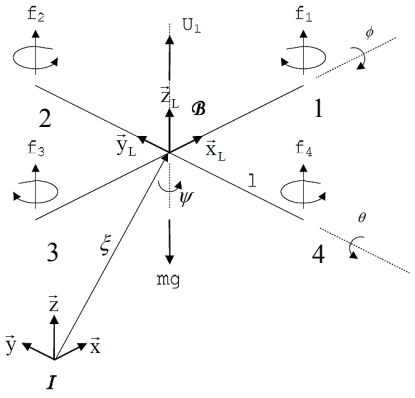

Fig. 1 Position and Orientation in a quadcopter. Source: [17]

The vehicle's orientation is given by an orthonormal rotation matrix $\boldsymbol{R}$, which transforms the orientation changes on $\vec{\boldsymbol{B}}$ System to the $\vec{\boldsymbol{I}}$ System [1]. Rotation of a rigid body can be obtained by using Euler angles.

$$R = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi c\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \quad (1)$$

The sine and cosine of the angle is represented by *s* and *c* respectively. The derivatives of Tait-Bryan angles $\dot{\Theta} = [\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$ are distinct from the angular velocities on the coordinate system of the rigid body $= [p, q, r]^T$, which can be measured by gyroscopes. The relationship between $\dot{\Theta}$ and ω is, [18]:

$$\dot{\Theta} = \boldsymbol{T}\omega \quad (2)$$

$\boldsymbol{T}$ is called Euler Matrix and is given by [18].

$$\boldsymbol{T} = \begin{bmatrix} 1 & sen\phi tan\theta & cos\phi tan\theta \\ 0 & cos\phi & -sen\phi \\ 0 & sen\phi sec\theta & cos\phi sec\theta \end{bmatrix} \quad (3)$$

When the angles φ, θ and ψ are small, the equation may approximate as shown in the equation (4).

$$\begin{aligned} p &= \dot{\phi} \\ q &= \dot{\theta} \\ r &= \dot{\psi} \end{aligned} \quad (4)$$

The kinematics of a rigid body with 6 degrees of freedom is given as [19], [15]:

$$\dot{\xi} = J_\Theta \nu \quad (5)$$

Where ξ is comprised of a vector of linear and angular position relative to the inertial coordinate system $\xi = [\Gamma \quad \Theta]^T = [X \quad Y \quad Z \quad \phi \quad \theta \quad \psi]^T$. Similarly, ν is comprised of a vector of linear velocity and angular velocity relative to the coordinate system fixed on the rigid body $\nu = [V \quad \omega]^T = [u \quad v \quad w \quad p \quad q \quad r]^T$. Thus, J_Θ is the conversión matrix:

$$J_\Theta = \begin{bmatrix} R & 0_{3x3} \\ 0_{3x3} & T \end{bmatrix} \quad (6)$$

The dynamics of a rigid body under external forces applied to the center of mass and expressed in the system of coordinates linked to the body can be obtained through the Newton-Euler equations [18]:

$$\begin{bmatrix} mI_{3x3} & 0_{3x3} \\ 0_{3x3} & I \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} \omega x m V \\ \omega x I \omega \end{bmatrix} = \begin{bmatrix} F_B \\ \tau_B \end{bmatrix} \quad (7)$$

$I_{3x3} \in \Re^{3x3}$ is the identity matrix and m is the total mass of body. $I \in \Re^{3x3}$ is the diagonal matrix of inertia. F_B and $\tau_B \in \vec{B}$ denote the forces and external torques applied to the rigid body expressed in equation (8) and (9) respectively [17] [3].

$$RF_B = -mge_3 + R_{e_3}\left(b\sum_{i=1}^{4}{\Omega_i}^2\right) \quad (8)$$

$$\tau_B = -\sum_{i=1}^{4} J_R(\omega x e_3).\Omega_i + \tau_a \quad (9)$$

The acceleration of gravity is represented by g, e_3 is a basic component of $\Re^3$, J_R is the moment of rotational of the rotor around its axis, b is the thrust coefficient and Ω_i is the angular velocity of the ith rotor.

The main force applied to the helicopter U_1 and responsible for the thrust is modeled as the sum of the thrust forces generated by each rotor. The torque applied to the three axes $\tau_a$is expressed in equation (10) [19]:

$$\begin{bmatrix} l(f_4 - f_2) = lU_2 \\ l(f_3 - f_1) = lU_3 \\ \sum_{i=1}^{4} \tau_{M_i} = lU_4 \end{bmatrix} = \begin{bmatrix} lb({\Omega_4}^2 - {\Omega_2}^2) \\ lb({\Omega_3}^2 - {\Omega_1}^2) \\ d({\Omega_2}^2 + {\Omega_4}^2 - {\Omega_1}^2 - {\Omega_3}^2) \end{bmatrix} \quad (10)$$

The distance between the quadcopter's motors and its center of gravity is given by *l*. The constant *d* is the drag coefficient of the propellers. Substituting in equation (7) all equations previously presented, equation (11) is obtained, where a quadcopter's nonlinear model is described, [17], [19], [3]and [15].

$$\begin{aligned} \ddot{X} &= (sen\psi sen\phi + cos\,\psi\, sen\theta cos\phi)\frac{U_1}{m} \\ \ddot{Y} &= (-cos\,\psi\, sen\phi + sen\psi sen\theta cos\phi)\frac{U_1}{m} \\ \ddot{Z} &= -g + (cos\theta cos\phi)\frac{U_1}{m} \\ \ddot{\phi} &= \frac{I_{YY} - I_{ZZ}}{I_{XX}} qr - \frac{J_{TP}}{I_{XX}} q\Omega + \frac{lU_2}{I_{XX}} \\ \ddot{\theta} &= \frac{I_{ZZ} - I_{XX}}{I_{YY}} pr + \frac{J_{TP}}{I_{YY}} p\Omega + \frac{lU_3}{I_{YY}} \\ \ddot{\psi} &= \frac{I_{XX} - I_{YY}}{I_{ZZ}} pq + \frac{U_4}{I_{ZZ}} \end{aligned} \quad (11)$$

Ω is defined as the overall speed using equation (12) where opposing directions of rotation are to be subtracted.

$$\Omega = \Omega_1 + \Omega_2 + \Omega_3 + \Omega_4 \quad (12)$$

The quadcopter dynamics should be simplified to provide an easy inverse model that can be implemented in the control algorithms. The system of equations (11) can be rearranged considering that near the stationary flight condition, the effects produced by the angles can be neglected. Under this consideration, the system of equations is obtained. (13). [15]

$$\ddot{X} = (sen\psi sen\phi + cos\,\psi\, sen\theta cos\phi)\frac{U_1}{m}$$

$$\ddot{Y} = (-cos\,\psi\, sen\phi + sen\psi sen\theta cos\phi)\frac{U_1}{m}$$

$$\ddot{Z} = -g + (cos\theta cos\phi)\frac{U_1}{m}$$

$$\ddot{\phi} = \frac{lU_2}{I_{XX}}$$

$$\ddot{\theta} = \frac{lU_3}{I_{YY}}$$

$$\ddot{\psi} = \frac{U_4}{I_{ZZ}}$$

In the X-type configuration, two front motors and two rear motors are considered. Mathematically, the X-type configuration is expressed by equation (14).

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} b(\Omega_1{}^2 + \Omega_2{}^2 + \Omega_3{}^2 + \Omega_4{}^2) \\ b(\Omega_4{}^2 + \Omega_3{}^2 - \Omega_1{}^2 - \Omega_2{}^2) \\ b(\Omega_2{}^2 + \Omega_3{}^2 - \Omega_1{}^2 - \Omega_4{}^2) \\ d(\Omega_1{}^2 + \Omega_3{}^2 - \Omega_2{}^2 - \Omega_4{}^2) \end{bmatrix} \quad (14)$$

The model can consider perturbations on the axis $\vec{X}$, $\vec{Y}$, and $\vec{Z}$, called A_p, A_q y A_r respectively. In equation (15) this model considering the latter variables is shown as Guilherme presented in [17]

$$\dot{p} = \frac{I_{YY} - I_{ZZ}}{I_{XX}}qr - \frac{J_{TP}}{I_{XX}}q\Omega + \frac{lU_2}{I_{XX}} + \frac{A_p}{I_{XX}}$$

$$\dot{q} = \frac{I_{ZZ} - I_{XX}}{I_{YY}}pr + \frac{J_{TP}}{I_{YY}}p\Omega + \frac{lU_3}{I_{YY}} + \frac{A_q}{I_{XX}} \quad (15)$$

$$\dot{r} = \frac{I_{XX} - I_{YY}}{I_{ZZ}}pq + \frac{U_4}{I_{ZZ}} + \frac{A_r}{I_{ZZ}}$$

III. Estimation of Parameters

The structure of the quadcopter used is 3DR ArduCopter Quad-C. The motors used are of the 850Kv AC2830-358 model. The brushless motors are connected to an ESC (Electronic speed control).

A. Identification of Propellers

The velocity test determines the linear relationship between the PWM with the angular velocity squared. The data approximates a straight line, see equation (16).

$$\Omega^2 = a_\Omega * PWM - b_\Omega \quad (16)$$

Where Ω^2 is the angular velocity squared, a_Ω y b_Ω are the constants defining the approximate straight line.

The thrust produced by the rotor for a given duty cycle, which enters the engine driver, approaches a straight line and it is given by equation (17).

$$E = a_e * PWM - b_e \quad (17)$$

E is the thrust in kilograms, a_e y b_e are the constants that define the approximate straight line. Finally the thrust coefficient b is determined by the equation (18).

$$b = \frac{a_e * g}{a_\Omega} \quad (18)$$

The drag produced by the rotor for a given duty cycle entered in the motor driver has been calculated experimentally. The data approximate a straight line, see equation (19).

$$A = a_a * PWM - b_a \quad (19)$$

A is the thrust in kilograms, a_a y b_a are the constants that define the approximate straight line. Finally the thrust coefficient is determined by the equation(20).

$$d = \frac{a_a * g * l}{a_\Omega} \quad (20)$$

B. Identification of the brushless motors

A study of the dynamics of driver-engine nonlinear system for different ranges of duty cycle is performed. 10 first-order transfer functions for the various ranges in Simulink are built. To tune the controllers, a transfer function in the motor's operating point is determined, see equation(21).

$$\frac{\Omega_{(S)}}{PWM_{(S)}} = \frac{K}{\tau_m s + 1} \quad (21)$$

C. Inertia Estimation

The quadcopter of the Lab of Automatic Control Systems is a modified structure of the 3DR ArduCopter Quad-C, being this the reason why inertias and weights of the system are assessed using SolidWorks software. Fig. 2 shows the designed model in which the metal structure, the embedded system, accessories, engines, propellers, nuts and bolts and spacers shown are considered.

Fig. 2 Quadcopter designed in SolidWorks

IV. Orientation Control Algorithms

Considering a PID with derivative part in the output, the control algorithm receives, as inputs, the sensor data and the reference. The algorithm output determines the PWM signal for the four motors.

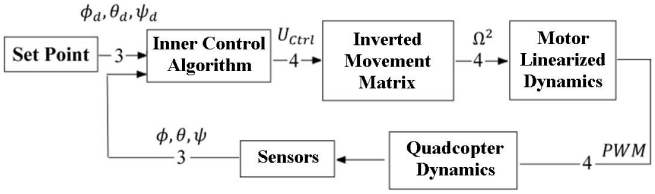

Fig. 3 Diagram of Control Block **[15]**

Fig. 3 shows a schematic for the algorithm of the quadcopter orientation control. The Inner Control Algorithm block contains the proposed PID controllers. The Inverted Movement Matrix block calculates the velocity squared of the propellers of the four rotors by equation (14). The Motor

Linearized Dynamics block calculates the duty cycle of the PWM signal of the four motors using equation (16).

The controllers' outputs are U_{Ctrl2}, U_{Ctrl3} and U_{Ctrl4}, which are dynamically different to the forces and moments on the quadcopter represented by U_2, U_3 and U_4. Therefore, the dynamics of the motors is included as shown in equation (22).

$$P_\phi = \frac{\ddot{\phi}}{U_{Ctrl2}} = \frac{l}{I_{XX}s^2(\tau_m s+1)}$$
$$P_\theta = \frac{\ddot{\theta}}{U_{Ctrl3}} = \frac{l}{I_{YY}s^2(\tau_m s+1)} \quad (22)$$
$$P_\psi = \frac{\ddot{\psi}}{U_{Ctrl4}} = \frac{1}{I_{ZZ}s^2(\tau_m s+1)}$$

Where τ_m is the characteristic time of the motor. The orientation system considers a modified closed-loop PID controller as shown in equation (23).

$$PID_{(S)} = E_{(s)}K_P + \frac{E_{(s)}K_I}{S} - Y_{(s)}K_d S - Y_{(s)}K_{d_2}S^2 \quad (23)$$

Where the subscript (S) indicates the variables over time taken to the Laplace plane and S is the Laplace variable.

V. PID Tuning

A iteration method that minimizes a specific cost function is used. Considering an external disturbance, besides the energy of the manipulable variable is added to avoid oscillations; the values obtained may vary as determined by the designer.

Cost functions for roll, pitch and yaw are respectively, J_ϕ, J_θ y J_ψ.

$$J_\phi = (\phi-1)' * (\phi-1) * R_\phi + (timeu_2 .* U_2)' * (U_2)$$
$$J_\theta = (\theta-1)' * (\theta-1) * R_\theta + (timeu_3 .* U_3)' * (U_3) \quad (24)$$
$$J_\psi = (\psi-1)' * (\psi-1) + (timeu_3 .* U_3)' * (U_3) * R_\psi$$

In which R_ϕ, R_θ and R_ψ are the weights for each cost function.

The tuning is done by iterative simulation, looking for behavior that the designer considered optimal.

VI. Simulation And Validation

A 3-D simulation has been constructed to improve visualization of the behavior of the control algorithm proposed on the mathematical model, see Fig. 4.

Fig. 4 Virtual world developed in Simulink.

TABLE I. Summarizes the Parameters Used in this Project

Name	Symbol	Value
Drag Factor	d	$2.7668 * 10^{-6}\ N.m.s^2$
Thrust Factor	b	$160 * 10^{-6}\ N.s^2$
Quadcopter Mass	m	$1.295\ kg$
Gravity	g	$9.81\ ms^{-2}$
Inertia on the Axis x	I_{XX}	$15.19 * 10^{-3}\ Nms^2$
Inertia on the Axis y	I_{YY}	$28.37 * 10^{-3}\ Nms^2$
Inertia on the Axis z	I_{ZZ}	$15.06 * 10^{-3}\ Nms^2$
Distance between the quadcopter center and the motors	l	$0.267\ m$
Time constant of motor	τ_m	$0.26\ s$

Data capture for validation is performed by using an Arduino UNO board, an IMU MPU6050 sensor, the Arduino 1.6.3 and Visual Studio 2012 software. The logic is as follows, the IMU MPU6050 sensor is composed of 3 accelerometers, 3 gyroscopes and 3 magnetometers. The Arduino board has two functions: the first is to collect data from the MPU6050 and estimate Euler angles. The second function is to send data to the computer via USB. The Visual Studio software is used to capture data, in addition to organizing and saving in a .csv file.

A validation process should indicate numerically the degree of approximation between the model and reality. In this paper, the coefficient of determination R^2 is used, which takes values between 0 and 1. In addition, the RMSE (*Error Root Mean Squared*) is used, a measure commonly used to indicate the difference between an estimated model and the actually observed values.

The experiments should be performed in a controlled and safe environment. In order to do so, a test bench was constructed inspired on work by [2]and [20]The system consists of an articulated base with 3 degrees of freedom which is presented in [21], as shown in Fig. 5

Fig. 5 Test Bench for the quadcopter with 3 degrees of freedom.

Figures 6, 7 and 8 show comparative graphs for the controlled three angles between experimental data (in blue) and simulation data (in red).It is observed in Fig. 6 and Fig. 7 that curves are quite similar. In Fig. 8 a greater difference between the data is observed, which could be explained because the testing bench has a friction which is considerable considering the low drag coefficient, d. In addition, the center of gravity of the quadcopter does not match the center of rotation thereof when it is fixed to the testing bench.

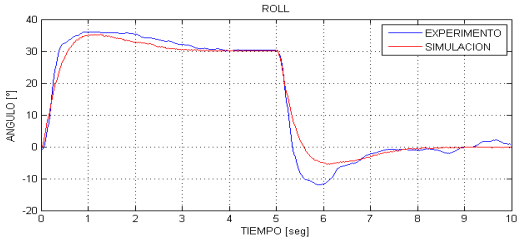

Fig. 6 Comparison of experimental data and simulation of roll angle

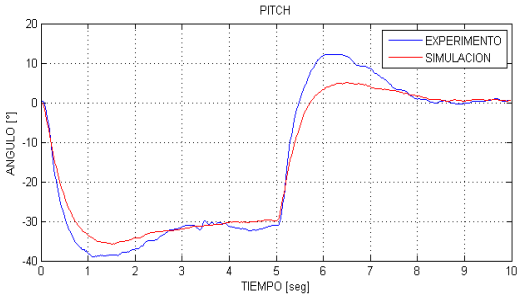

Fig. 7 Comparison of experimental data and simulation of pitch angle.

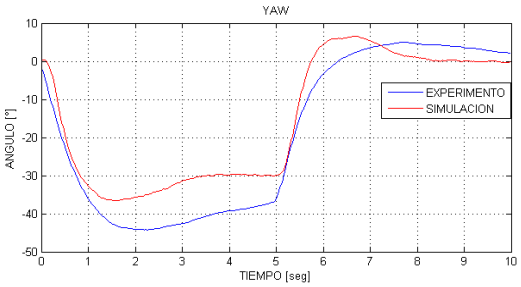

Fig. 8 Comparison of experimental data and simulation of yaw angle.

Table II summarizes the validation parameters used. Overall, satisfactory values are obtained, although less favorable for the yaw. However it is concluded that the mathematical model is validated.

TABLE II. SUMMARY CHART OF MODEL'S VALIDATION

Roll		Pitch		Yaw	
R^2	$RMSE$	R^2	$RMSE$	R^2	$RMSE$
0.9652	3.37	0.9662	3.44	0.8847	6.67

To verify our results in reality, experiments on the test bench are performed, see Fig. 9

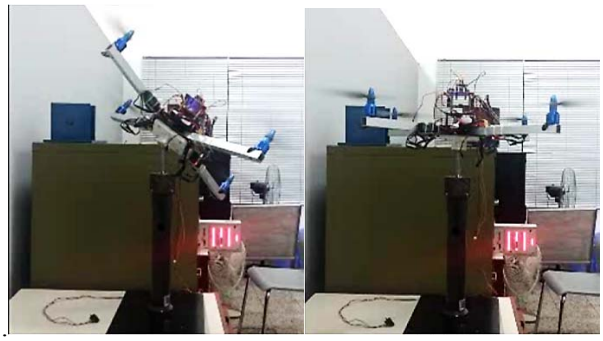

Fig. 9 Controller tests under an adverse initialization

During the tests performed, it was observed that the quadcopter, in spite of the quadcopter's start in unfavorable conditions, the algorithm is able to get over, see Fig. 9; besides, other disturbances were added in the form of human strength and it was observed that the quadcopter rapidly regains stability, see Fig. 10.

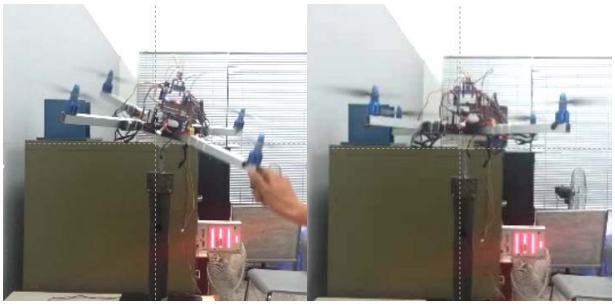

Fig. 10 Controller tests under external disturbances

Finally a field trial is performed to check the good results as shown in Fig. 11.

Fig. 11 Final flight test at the Campus of Universidad de Piura, Perú.

VII. CONCLUSION

This work has allowed a nonlinear model of good performance and a robust flight controller with real-time experiments.

The implementation of a virtual world has been very important to observe the operation of the quadcopter under controlled conditions.

Trials performed with the proposed PID controller have reflected smooth flight conditions, always under control, and stable under external perturbations.

As a final test, flights were carried out over a soccer field at the University of Piura, at nighttime to get the worst external conditions, showing great robustness against the strong wind currents.

REFERENCES

[1] S. Bouabdallah, A. Noth y R. Siegwart, «PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor,» *Proceedings. 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems,* vol. 3, pp. 2451-2456, 2004.

[2] S. Bouabdallah, P. Murrieri y R. Siegwart, «Design and Control of an Indoor Micro Quadrotor,» *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on,* vol. 5, pp. 4393 - 4398, 2004.

[3] S. Bouabdallah, P. Murrieri y R. Siegwart, «Towards Autonomous Indoor Micro VTOL,» *Autonomous Robots,* vol. 18(2), pp. 171-183, 2005.

[4] S. Bouabdallah, «Design and control of quadrotors with application to autonomous flying,» Tesis doctoral, Laboratoire de systèmes autonomes, ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE, LAUSANNE, 2007.

[5] B. Herisse, F.-X. Russotto, T. Hamel y R. Mahony, «Hovering flight and vertical landing control of a VTOL Unmanned Aerial Vehicle using Optical Flow,» *IEEE/RSJ International Conference on Intelligent Robots and Systems,* pp. 801 - 806, 2008.

[6] R. A. García, F. R. Rubio y M. G. Ortega, «Robust PID Control of the Quadrotor Helicopter,» *IFAC Conference on Advances in PID Control,* vol. 2(5), pp. 229-334, 2012.

[7] N. Guenard, T. Hamel y R. Mahony, «A Practical Visual Servo Control for an Unmanned Aerial Vehicle,» *IEEE Transactions on Robotics,* vol. 4(2), pp. 331 - 340, 2008.

[8] O. Bourquardez, R. Mahony, N. Guenard, F. Chaumette, T. Hamel y L. Eck, «Image-based Visual Servo Control of the Translation Kinematics of a Quadrotor Aerial Vehicle,» *IEEE Transactions on Robotics,* vol. 25(3), pp. 743 - 749, 2009.

[9] B. Hérissé, T. Hamel, R. Mahony y F.-X. Russotto, «Landing a VTOL Unmanned Aerial Vehicle on a moving platform using optical flow,» *IEEE Transactions on Robotics,* vol. 28(1), pp. 77 - 89, 2012.

[10] E. Altug, J. P. Ostrowski y M. Robert, «Control of a Quadrotor Helicopter Using Visual Feedback,» *IEEE International Conference on Robotics and Automation,* vol. 1, pp. 72 - 77, 2002.

[11] E. Altug, J. P. Ostrowski y C. J. Taylor, «Control of a Quadrotor Helicopter Using Dual Camera Visual Feedback,» *IEEE International Conference on Robotics and Automation,* vol. 24(5), pp. 329 - 341, 2003.

[12] D. Mellinger y V. Kumar, «Minimum Snap Trajectory Generation and Control for Quadrotors,» *IEEE International Conference on Robotics and Automation,* pp. 2520 - 2525, 2011.

[13] D. Mellinger, N. Michael y V. Kumar, «Trajectory Generation and Control for Precise Aggressive Maneuvers with Quadrotors,» *Experimental Robotics. The 12th International Symposium on Experimental Robotics,* vol. 79, pp. 361-373, 2014.

[14] E. Paiva, J. Soto, J. Salinas y W. Ipanaqué, «Modelado y Control PID en cascada de un Cuadricóptero para seguimiento de trayectorias,» de *2015 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, Santiago, 2015.

[15] T. Bresciani, «Modelling, Identification and Control of a Quadrotor Helicopter,» Tesis M.S., Dept. of Automatic Control, Lund University, Lund, 2008.

[16] Universidad de Alcalá, «Servidor de Teoria de la Señal,» [En línea]. Available: http://agamenon.tsc.uah.es/Asignaturas/it/rd/apuntes/handout4.pdf.

[17] G. Vianna Raffo, «Modelo y control de un helicoptero Quadrotor,» Sevilla, 2017.

[18] R. Olfati-Saber, «Nonlinear control of underactuated mechanical Systems with Application to Robotics and Aerospace Vehicles,» Tesis M.S., Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Massachusetts, 2001.

[19] H. Fernando, A. De Silva, M. De Soysa, S. Munasinghe y K. Dilshan, «Modelling, Simulation and Implementation of a Quadrotor UAV,» *IEEE International Conference on Industrial and Information Systems,* pp. 207 - 212, 2013.

[20] M. Bayrakceken y A. Arisoy, «An educational setup for nonlinear control systems: Enhancing the motivation and learning in a targeted curriculum by experimental practices,» *Control Systems,* vol. 33, nº 2, pp. 64-81, 2013.

[21] J. Salinas, M. Villegas, R. Bocanegra, J. C. Soto, E. Paiva y W. Ipanaque, «Modelación, Simulación y Control de un Módulo Experimental para Control de ángulos de Orientación de un Quadcopter,» de *XXII Congreso Internacional de Ingeniería Eléctrica, Electrónica, Telecomunicaciones y Computación – INTERCON 2015*, Huancayo, 2015.

Ernesto Alonso Paiva Peredo recibió el título de Ingeniero Mecánico - Eléctrica de la Universidad de Piura, Perú, en el año 2013; ha realizado una Maestria en Ingeniería Mecánico Eléctrica con mención en Automática y Optimización en la Universidad de Piura financiada por CONCYTEC, 2016; Asistente de investigación del Laboratorio de Sistemas Automáticos de Control desde el año 2016; ha realizado pasantía en el Departamento de Ingeniería de Sistemas y Automática de la Escuela Superior de Ingeniería de la Universidad De Sevilla en el año 2015.

Juan Carlos Soto Bohórquez recibió el título de Ingeniero Mecánico - Eléctrica de la Universidad de Piura, Perú, en el año 2013; ha realizado una Maestría con mención en Eficiencia Energética en la Universidad de Piura, 2014; Investigador del Laboratorio de Sistemas Automáticos de Control desde el año 2012; ha trabajado en proyectos I+D+i con el FINCyT y CONCYTEC; posee registros de propiedad intelectual, derechos de autor y patente; ha realizado publicaciones en áreas de Sistemas Embebidos, Control Automático y Procesamiento de Imágenes Hiperespectrales, Algoritmos de Procesamiento de Imágenes.

Julio Antonio Salinas Castro estudiante de ingeniería Mecánica – Eléctrica de la Universidad de Piura, Perú; asistente en el Laboratorio del Sistemas automáticos de control de la Universidad de Piura desde el año 2014; ha participado en el desarrollo de proyectos I+D+i con el FINCyT y CONCYTEC; ha realizado publicaciones en áreas de Sistemas Unmanned Aerial Vehicle y Control; perteneció al comité organizador del XXIII Congreso Internacional de Ingeniería Electrónica, Eléctrica y Computación, 2016.

William Ipanaqué Alama recibió el título de Ingeniero Industrial en la Universidad de Piura, Perú, en 1984; PhD en Ingeniería Informática y Automática en el Politécnico de Milán, Italia; actualmente es docente e investigador de la Universidad de Piura; autor de papers sobre Modelación, Simulación y Control Avanzado (Predictivo) de Procesos Industriales. Además es director de varios proyectos de investigación en el Perú, y del Doctorado en Automatización, Control y Optimización de la Universidad de Piura.