Intelligent Fuzzy Controller of a Quadrotor

Matilde Santos, Victoria López
Facultad de Informática,
Universidad Complutense,
28040 Madrid, Spain
msantos@dacya.ucm.es, vlopez@fdi.ucm.es

Francisco Morata
Telefónica
Madrid, Spain
fran_morata@hotmail.com

Abstract—The aim of this work is to describe an intelligent system based on fuzzy logic that is developed to control a quadrotor. A quadrotor is a helicopter with four rotors, that make the vehicle more stable but more complex to model and to control. The quadrotor has been used as a testing platform in the last years for various universities and research centres. A quadrotor has six degrees of freedom, three of them regarding the position: height, horizontal and vertical motions; and the other three are related to the orientation: pitch, roll and yaw. A fuzzy control is designed and implemented to control a simulation model of the quadrotor. The inputs are the desired values of the height, roll, pitch and yaw. The outputs are the power of each of the four rotors that is necessary to reach the specifications. Simulation results prove the efficiency of this intelligent control strategy.

Keywords-intelligent control system, fuzzy control, quadrotor, UAVs

I. INTRODUCTION

Autonomous helicopters are one of the most used robotic platforms because their ability of vertical take-off and landing as well as stationary flight [1, 2, 3].

They can be very useful in natural disasters, rescues, watchfulness, fields or structures inspections, making geographic maps and so on. In many occasions the actions involve risks and require high mobility and small size aircrafts. This insists on the idea of unmanned vehicles [4, 5, 6].

This paper presents the design and simulation of an intelligent control of a quadrotor. The main advantage of these four planar-rotor aircrafts with respect to a standard helicopter is the stability of the vehicle. As a disadvantage it could be mentioned the difficulties of coordinating the four rotors at once.

The proposed intelligent control strategy is a PID-like fuzzy one. That is, it is based on fuzzy logic to take advantage of the facility to implement by rules the control actions.

Other works related to this topic can be found in the literature. On the one hand, there are some papers on identification and control of conventional helicopters, applying different techniques, such as [7], who applies neural network for identification, or [8, 9], where fuzzy logic is applied. On the other hand, different approaches to the modeling and control of quadrotor can be read in [10, 11, 12, 13, 14].

This paper is organized as follows. Section 2 briefly describes the quadrotor model and its characteristics. In Section 3, the design of the fuzzy controller is presented. Section 4 shows the simulation results for different testing trajectories. Finally in section 5 the conclusions are presented.

II. DESCRIPTION OF THE SYSTEM

In a quadrotor, the four rotors are symmetrically distributed respect to the central cabin (see Fig. 1). We are working with a real replica of the system. The main characteristics are: the mass of the cabin is 500 gr, mass of the rotor-gear-motor set is 50 gr, the rod mass can be neglected. The radio of the vehicle is 6 cm and the distance of any engine to the center of the cabin is 23 cm. Each span is 46 cm long.

Figure 1. Picture of the quadrotor

The control output is the sum of the forces produced by each engine [1, 4, 15], that is,

$$\mu = \sum_{i=1}^{4} f_i \qquad (1)$$

Where f_1 is the force applied by the front engine, f_2 the force applied by the left motor, f_3 the force applied by the back engine and f_4 the force applied by the right rotor, as it can be seen in Fig. 4.

978-1-4244-6793-8/10/$26.00 ©2010 IEEE

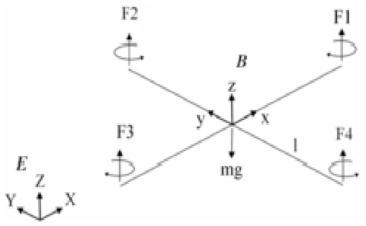


Figure 2. Quadrotor System Reference

A quadrotor is controlled by varying the angular velocity of the rotors. Front and back rotors rotate clockwise, meanwhile left and right rotors rotates anti-clockwise. This makes the aerodynamic effects and gyroscopic moments to be canceled in a stationary flight [1, 4].

Rigid body movement has six degrees of freedom: three of them define the reference position (usually the center of mass), and the other three define the orientation of the vehicle [15]

$\mathbf{x} = (x, y, z)$ position vector of the quadrotor.

$\mathbf{v} = (u, v, w)$ speed vector of the quadrotor.

$\boldsymbol{\alpha} = (\varphi, \theta, \psi)$ are the Euler angles (pitch, roll and yaw respectively).

$\boldsymbol{\omega} = (p, q, r)$ is the angular speed vector.

We can consider the quadrotor as a central sphere with mass M and radius R, four arms with length l and a punctual mass m at the edge of each arm, which represents the motor-rotor set. More detailed information on the equations of the motion of quadrotor vehicles can be found in [13, 16, 17].

The dynamic model has been developed with Matlab/Simulink [18]. There is a block, named Control, where the Fuzzy Controller is located, which corrects the errors by modifying the power of each one of the four engines.

There are two system limitations:

- Negative heights are not possible, obviously. When the height reaches a negative value, it is assigned to zero.
- Some angles restrictions are imposed for security reasons. For example, the roll and pitch can only takes values in the interval [-1.3, 1.3] radians if they act independently and between [-1, 1] radians if the tow motions are coupled. Otherwise the quadrotor can easily lose the stability.

III. Intelligent Fuzzy Control

The fuzzy control strategy implements the logic used to control this vehicle [19, 20, 21]. The controller decisions are taken based on the desired combination of the four motions: height, pitch, roll and yaw. These actions increase or reduce the power of each of the motors in order to get the specifications and to follow a trajectory.

The inputs of the controller are the error of the state variables, three for each vector $\mathbf{x}$, $\mathbf{v}$, $\boldsymbol{\alpha}$, and $\boldsymbol{\omega}$. The error is calculated as the difference between the desired state and the current state (Xd – X).

Four fuzzy PID-like controllers are implemented (see Fig. 3), for each of the motion that is going to be controlled: height (z), roll φ, pitch θ, and yaw, ψ.

The output is a vector of four components that gives the required engine power of each motor to get the desired position.

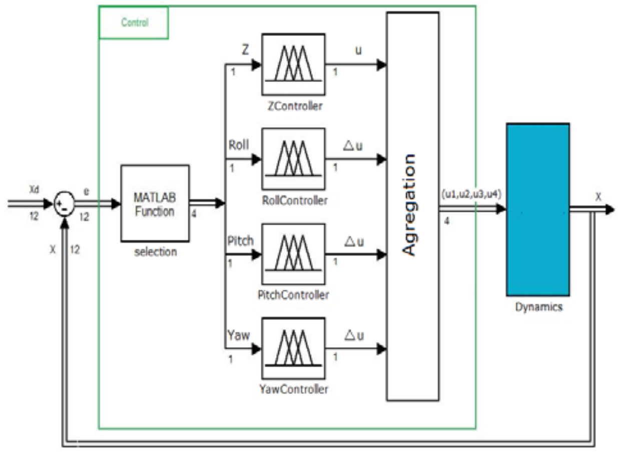


Figure 3. Control Diagram

A. *The aggregation block*

"Aggregation" (Fig. 3) is an intermediate block between the controllers and the dynamics of the system. Its functionality is to implement the coupling of the control actions of the four controllers.

Each controller gives four outputs, one for each motor. That is, z controller gives the power required of each engine, M1, M2, M3 and M4. And so does the pitch controller, the yaw and the roll ones.

The aggregation block calculates the final power of each of the four engines, combining the outputs of each controller for the specific rotor.

Z-controller

The Z-controller returns the power required by each engine to correct the error in height. The output of this controller must be the same for each rotor, either increasing or decreasing; otherwise it would produce a drift. That means that if we want to go up, the input for each engine must be the same and positive, and if the quadrotor has to go down, the power of the four rotors should be decremented the same quantity. The controller gives only one value as output, fz, and this value is applied to the four engines.

$$\mu_z = f_1 + f_2 + f_3 + f_4 = 4fz$$

The output of Z-controller is the base on which the outputs of the other controllers are added in the Aggregation block. The quadrotor has always to maintain a specific height so it needs this power. The other motions increase or decrease the power of some of the motors, but they are overlapped with the action of the z-controller.

$$\mu = \mu_z + \Delta\mu_{ROLL} + \Delta\mu_{PITCH} + \Delta\mu_{YAW}$$

Roll controller:

It acts on the lateral motors, M2 and M4. For rotation around x-axis, the front and back motors (M1 and M3) must be kept without changes. The increment in one of the lateral rotors must be compensate with the same decrement in the other one, to maintain the support force (μ) constant and in order not to destabilize the quadrotor.

Again this fuzzy controller only returns a single value, the absolute value $\Delta\mu_{ROLL}$ that should be added to M2 and be subtracted from M4 force.

A positive value means a rotation clockwise (right) and a negative value means anti-clockwise (left). That is, for right rotation it is necessary that the left rotor (M2) had more power than the right one (M4). And vice versa.

$$f_2 = fz - \Delta_{ROLL} \qquad f_4 = fz - \Delta_{ROLL}$$

Demonstration:

- Turn right: if $\Delta\mu_{ROLL} > 0$ then $f_2 > f_4$
- Turn left: if $\Delta\mu_{ROLL} < 0$ then $f_4 > f_2$

Pitch controller

It operates on the front and back motors, M1 and M3. It turns around y-axis, so M1 and M3 must be modified and lateral rotors (M2 y M4) must be kept. The value calculated by the Pitch fuzzy control is $Äì_{PITCH}$, which is added or subtracted.

$$f_1 = fz + \Delta_{PITCH} \qquad f_3 = fz - \Delta_{PITCH}$$

Yaw controller

It acts on the four motors, M1, M2, M3 and M4. It turns around the z-axis. It is known that regarding the yaw movement, the rotors are grouped in pairs: lateral ones (M2 and M4) turn in one direction, and the front and back rotors (M1 and M3) change in the other. Furthermore, if the support force is kept constant, anything that increases/decremented the power engine in one pair should produce the opposite effect on the other pair.

This fuzzy controller gives the value $+\Delta\mu_{YAW}$ as output that is added to the output of the z-controller as an increment in M2 and M4 and subtracted from M1 and M3.

Yaw Controller returns a positive power if it rotates clockwise (right) and a negative power if it rotates anticlockwise (left). It acts like roll-controller.

$$f_1 = fz + \Delta_{YAW} \qquad f_2 = fz - \Delta_{YAW}$$

$$f_3 = fz + \Delta_{YAW} \qquad f_4 = fz - \Delta_{YAW}$$

With this strategy the number of fuzzy rules is reduced and the controller is simpler.

B. Intelligent Fuzzy controllers

Each of the controllers works with the error, its derivate and the integral (PID-like). These input variables are multiplied by gains, GE, GDE, GIE respectively. The output is the control action, u, and the corresponding gain is GU.

Fuzzy controllers have been developed and implemented with the Fuzzy Logic Toolbox of Matlab. All of them have these following commons characteristics:

- Mandami inference.
- The central membership function is triangular and the rest are trapezoidal.
- Fuzzy rules have followed this form:

  if (Ez is E) and (dEz is DE) and (iEz is IE) then (U is Uz), where E, DE, IE, Uz are fuzzy sets.
- Defuzzification method: centroid.
- AND operator implemented as the minimum.
- Implication: minimum function.

The tuning of the gains of the input and output variables have been done by trial and error.

C. Fuzzy Height controller

As an example, the fuzzy controller of the height is presented. The input is the error (difference between the desired height and the present one), its derivative and its integral. The output is the control value of the power to be applied to the four motors.

The three fuzzy sets of the input error are: up, stand, down. For the derivate and for the integral are negative, equal and positive (universe of discourse [-2, 2].

The output has five fuzzy sets (Fig. 4): Go Down Much (GDM), Go Down (GD), Stand (S), Go Up (GU) and Go Up Much (GUM).

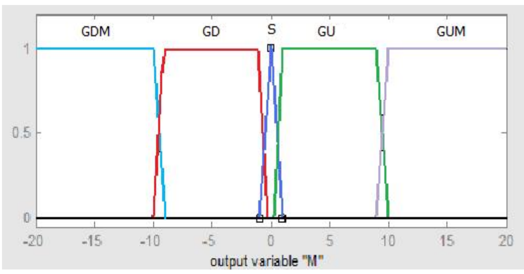


Figure 4. Fuzzy sets for output M of height

The tuning parametes of the fuzzy controller are:

GE = 1, GDE = 0.5, GIE = 10, GU = 1

Table 1 shows the control actions of the possible combinations of the three input values. Although the number of rules is 27, they have been reduced to 11, joining the ones that produce the same output.

TABLE I. INFERENCE RULES OF THE HEIGHT CONTROLLER

dZ - iZ \ Z	Up	Stand	Down
Negative-Negative	GDM	GD	S
Negative-Equal	GDM	GD	S
Negative-Positive	GDM	GD	S
Equal-Negative	GD	GU	GU
Equal-Equal	GD	S	GU
Equal-Positive	GD	S	GU
Positive-Negative	GU	GU	GUM
Positive-Equal	GU	GU	GUM
Positive-Positive	GU	GU	GUM

In the same way, the other fuzzy controllers are defined.

The three fuzzy sets for the Roll error are Clockwise, Stand and Anti-clockwise. It refers to the sense of the spin around the x-axis, looking from the back. For the derivative and the integral are the same than for the height.

The output has five fuzzy sets with labels: TLM (Turn Left Much), TL (Turn Left), S (Stand), TR (Turn Right), TRM (Turn Right Much). The range is between [-0.1, 0.1].

The gains for the variables of this controller are:

$$GE = 1, GDE = 3, GIE = 5, GU = 1$$

For the pitch and the yaw PID-like controllers, the definition of the membership functions of the input and output variables are the same. The rules are quite similar.

IV. SIMULATION RESULTS

A small but significant set of tests have been carried out to prove the efficiency of the intelligent control system. Some of the motions have been simulated independently, and then coupled. For example, Fig. 5 shows the height gets the reference for a 2 m step. The curve is quite smooth, as it should be.

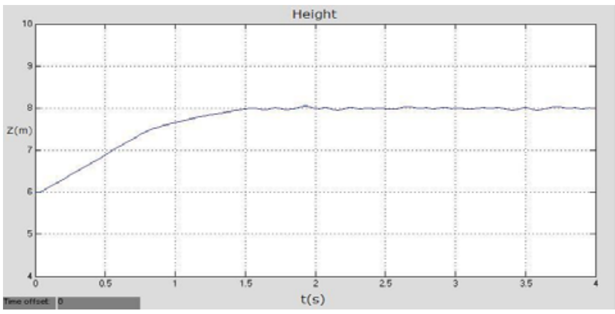
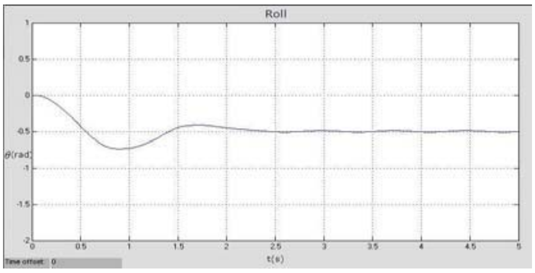


Figure 5. Fuzzy control of the height for a step from 6 to 8 m

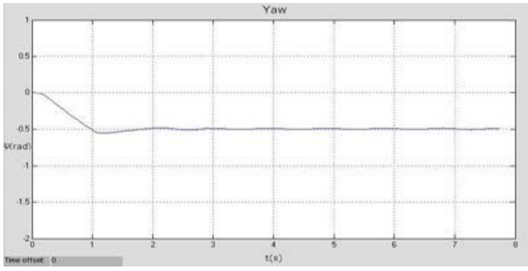
The roll has been tested with a change of 0.5 radians (28º) in both senses. It is stabilized fast and the transitory phase is smooth, and it never exceeds 1 rad.

Figure 6. Roll simulation from 0 to -0.5 radian



The same happens with the pitch and yaw motions. The controllers give a fast and good response for a step of 0.5 rad. The response for the yaw is more stable (Fig. 7).

Figure 7. Yaw simulation from 0 to -0.5 radian



A. *Coupled motions of the quadrotor*

Different flight tests that simulate the control of all variables (height, roll, pitch, yaw) simultaneously have been carried out.

The simulator incorporates an x-y map in which the quadrotor route is displayed (Fig. 8). It can be appreciated the lateral movements caused by yaw or roll movements, and the advance of the vehicle produced by the pitch.

In the initial position, quadrotor is on the ground and has all the angles set to zero.

The quadrotor takes off and goes up 20 meters high with no spin. During 4 seconds a pitch movement of -0.8 radians (Fig. 8) is the cause of the vehicle forward movement.

Ten seconds from the beginning, the roll turns to the left a little producing a great inclination in the quadrotor until it reaches the roll limit, 1.3 radians (74.5º) at 57 seconds. The quadrotor continues advancing until 87 seconds and then the pitch will be the only motion during a period of about 100 seconds. In that moment, roll (-0.5 radians) and yaw (-1 radian), make the vehicle turned right.

That continues until 160 seconds. Just then, the quadrotor keeps on moving forward but reducing the velocity (lower pitch) and at 200 seconds it will go into reverse. Finally it lands with no inclination and finishes the execution.

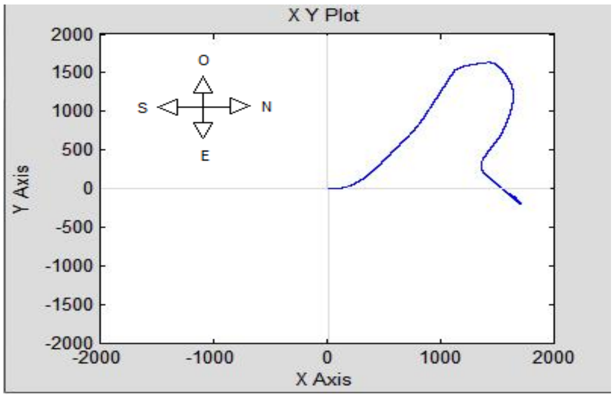


Figure 8. Trajectory of the quadrotor

Regarding the trajectory, as it is shown in Fig. 8, the first trace represents roll and pitch motions simultaneously. The quadrotor takes off, reaches twenty meters high and moves a few meters forwards. As it was said, the roll only produces a lateral movement, in this case to the left. That is the reason that after a while, the quadrotor has reached a position at the coordinates (1000, 1500) in the x-y plane. The displacement in the x-axis is due to the pitch, which in this case is lower than roll. The orientation is so far constant.

The next part of the trajectory shows a curve, the orientation changes because the roll movement has finished and the quadrotor is only moving along the x-axis.

Then the yaw starts and it produces a close curve to the right. That produces the quadrotor to go in a north-east direction. The roll needs more time than the yaw to show its effects because it has less inclination, but when it acts it produces a new curve to the right. When the roll finishes, there is not any lateral movement and the vehicle goes to the left. The quadrotor only moves forward due to the pitch and with a 57º north-east orientation because of the yaw.

After a while, the vehicle starts the landing. It reduces the speed, goes into reverse and then lands. Then the simulation ends.

A control panel (Fig. 9) shows the response for each variable simultaneously. The six graphics show the position variables *x*, *y* and *z*, and the three turns (roll, pitch and yaw).

For example, z variable (height) is at the bottom. It reaches quickly 20 m high and keeps that position until it lands. It is not possible to see in the plot the small oscillations that it presents, as it is almost impossible to completely stabilize the quadrotor at a high.

In the graphics of the pitch, roll and yaw motion is possible to see how they reach the specifications, and at the same time, they fulfill the constraints and never exceed the limit of 1.5 radians that would destabilize the vehicle.

The control panel also includes the *x* and *y* variables of the position, that represent the trajectory in the plane. The effect of the yaw signal can be noticed in the *y* plot, at around 100 seconds. The effects of the pitch in the *x* and *y* variables can be also seen, especially when the reverse movement of the quadrotor for the landing.

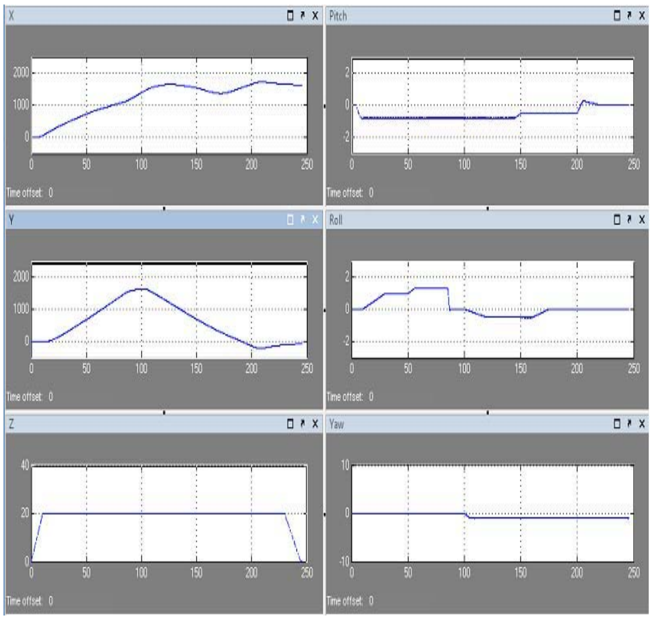


Figure 9. Control Panel

V. CONCLUSION

The application of intelligent strategies such as fuzzy logic in the design of control systems allows flexibility and efficiency. The application of expert knowledge about the behavior is sometimes the only way to deal with complex systems.

In this work, an intelligent system based on fuzzy logic has been designed and implemented in order to control a quadrotor. This vehicle has a complex dynamics because of the coupling of the different variables that represent the motion.

The simulations results obtained for different tests are quite promising. The control variables have been controlled, following the inevitable compromise between stability, precision and speed of the system response.

Due to the inter-dependence of the variables, the tuning of the controller parameters plays an important role in this application. Future works will be focused on this issue.

On the other hand, as a real replica of the quadrotor is available, the next step would be to prove the controller in that real environment. Problems such as noise, uncertainty from the sensors, etc, will help to show the robustness and efficiency of this strategy.

REFERENCES

[1] B. L. Stevens, and F.L. Lewis, Aircraft control and Simulation. John Wiley & Sons Inc., 2nd edition, 2003.

[2] A. Ollero, A. and I.Maza, Multiple Heterogeneous Unmaned Aerial Vehicles, vol.37 Springer Berlin/Heidelberg, 2007.

[3] T. R. Teo, J. S. Jang, C. J. Tomlin, "Automated Multiple UAV Flight – the Stanford DragonFly UAV Program," 43rd IEEE Conference on Decision and Control, 2004.

[4] S. Bouabdallah. Design and control of quadrotors with application to autonomous flying . PhD Thesis. Ecole Polytechnique Federale de Lausanne, 2007.

[5] P. Fabiani et al. Autonomous flight navigation of VTOL UAVs: from autonomy demonstrations to out-of-sight flights, Australian National University, Canberra, 2007.

[6] M. B. Srikanth, Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, “A Robust Environment for Simulation and Testing of Adaptive Control for Mini-UAVs,” American Control Conference, St Louis, USA June 10-12, 2009.

[7] R. San Martin, A. Barrientes, P. Gutierrez and J. del Cerro, “Neural networks training architecture for uav modelling,” World Automation Congress, WAC'06, 2006.

[8] C. Phillips, C. L. Karr, and G. Walker, “Helicopter Flight Control with Fuzzy Logic and Genetic Algoritms,” Eng. Appl. Artif. Intell., vol. 9 (2), pp. 175-184, 1996.

[9] Sugeno et al., “Intelligent control of an unmanned helicopter based on fuzzy logic,” Proc of American Helicopter Society 51th Annual Forum, Texas, 1995.

[10] S. Saripalli, J. F. Montgomery and G. S. Sukhtatme, “Vision-based Landing of an Unmanned Aerial Vehicle,” In IEEE Trans. on Robotics and Automation, vol. 19(3), pp. 371-381, 2003.

[11] E.N. Sanchez, H. M. Becerra, and C.M. Velez, “Combining fuzzy, PID and regulation control for an autonomous minihelicopter,” Information Sciences, 177, 1999-2022, 2007.

[12] T. Madani and A. Benallegue, “Control of a quadrotor mini-helicopter via full state backstepping technique,” International Journal of Machine Studies, pp. 1515-1520, 2006.

[13] P. Pounds, R. Mahony, P. Corke, “Modelling and Control of a Quad-Rotor Robot,” Aerospace Science and Technology, vol. 1l, pp. 183-193, 2007.

[14] M. Labadille, Non-linear Control of a Quadrotor. PhD dissertation, Cranfield University, 2007.

[15] H. Golstein, Classic Mechanics. Aguilar, 2nd ed., 1966.

[16] R. Lozano, P. Castillo and A. E. Dzul. Modelling and control of mini-flying machines. Springer, Londres, 2005.

[17] V. Martinez Martinez. Modelling of the Flight Dynamics of a Quadrotor Helicopter. PhD dissertation, Cranfield University, 2007.

[18] D. Sánchez Benítez, J.M. Cruz, G. Pajares. Uso de Visión Artificial Para Localización y Control, dos ejemplos prácticos. XXX Jornadas de Automática, Valladolid, Spain, 2009.

[19] M. Sugeno, Industrial applications of fuzzy control. Ed. Elsevier. North Holland, Amsterdam. 1985.

[20] K. M. Passino, S. Yurkovich, Fuzzy Control. Addison Wesley Longman, Inc., 1998.

[21] E. M. Mandami, “Applications of fuzzy algorithms for control of simple dynamics plant,” Proc. IEEE, vol. 121, 1585-1588, 1974.