Revue des Sciences et Sciences de l'ingénieur

Journal of sciences and engineering sciences

ISSN 2170-0737/EISSN: 2600-7029

https://www.asjp.cerist.dz/en/PresentationRevue/303

Fuzzy Logic Controller Using the Nonholonomic Constraints for Quadrotor Trajectory Tracking

***Ali Medjghou[a], Noureddine Slimane[b] and Kheireddine Chafaa[a]**

[a]University of Batna 2 Mostefa BenBoulaid: Electronics Department, Laboratory of Advanced Automatics and Systems Analysis (LAAAS)
[b]University of Batna 2 Mostefa BenBoulaid: Electronics Department, Laboratory of Advanced Electronic (LEA)

Article history
Submitted date: 2018-04-13
Acceptance date: 2018-09-03

Abstract

In this paper, an intelligent control approach for an unmanned aerial vehicle (UAV) using the nonholonomic constraints, is presented. The UAV is a mini drone with four rotors called quadrotor. It is a nonlinear coupled and unstable system. To properly control this robot and mitigate the disadvantages, a fuzzy logic controller (FLC) based on Takagi-Sugeno approach (TKS) for the altitude, the position and the attitude tracking of a quadrotor, in the presence of external disturbances is proposed, taking into account the nonholonomic constraints of the model. The desired roll and pitch angles are deduced from nonholonomic constraints. This adopted control strategy is summarized in the control of two subsystems. The first relates to the orientation (attitude) control, taking into account the position control along (x; y) axes. The second is that of the altitude control along z axis. For the concretization of this work, the matlab/simulink environment is used and the obtained results prove the efficiency of this fuzzy logic control strategy.

Key-words: Fuzzy Logic Control; Nonholonomic Constraints; Quadrotor; Trajectory tracking.

1. Introduction

The UAV type quadrotor has become more commonly used for many applications e.g. research, surveillance and reconnaissance in specific regions or in dangerous regions which are inaccessible or hard to reach for other types of vehicles [1]. Moreover, the field of UAV involves many engineering challenges in the areas of electrical, mechanical and control engineering.

In practical applications, the position in space of the UAVs is generally controlled by an operator through a remote-control system, while the attitude can be automatically stabilized via an onboard controller. The attitude controller is an important feature since it allows the vehicle to maintain a desired orientation and, hence, prevents the vehicle from flipping over and crashing when the pilot performs the desired maneuvers [2]. A quadrotor is a dynamic vehicle with four input forces, six output coordinators, highly coupled and unstable dynamics [3, 4].

Research into the quadrotor control problem has led to many potential solutions, several of which have been implemented successfully on hardware test beds. For example, in order to stabilize the quadrotor, authors in [5] propose a control law based on the choice of a stabilizing Lyapunov function ensuring the desired tracking trajectories along (*X, Z*) axes and roll angle. For the same problem*, of proportional integral derivative (PID)* control is used in [6]. However, they do not take into account nonholonomic constraints. Another popular nonlinear control techniques, as sliding mode and backstepping procedure, have used by [7] for the stabilization problem; the authors do not take into account frictions due to the aerodynamic torques nor drag forces or nonholonomic constraints. In [8], authors propose a control algorithm based on sliding mode using backstepping approach allowed the tracking of the various desired trajectories expressed in term of the mass center coordinates along (*X,Y,Z*) axes and yaw angle. A feedback linearization approach was used by [9]. Additionally, another controller design based on backstepping approach is used by [10]. Furthermore, the control approach proposed by [11] is based on the idea that the UAV model is constituted of two subsystems, then the attitude control is proposed using an integral sliding mode control, and a simple PID is used for the positioning control (motion in the *x-y* plane). Moreover, in [12], authors propose a control law based on linear algebra theory. In [13] a nonlinear decentralized model predictive control was proposed to control a formation of unmanned vehicles, the author's takes into account saturation constraints for the control actions, together with collision avoidance constraints.

In this paper, a particular interest is attributed principally to the quadrotor robot modeling taking into account various parameters which affect the dynamics of a flying structure such as frictions due to the aerodynamic torques, drag forces along (*X, Y, Z*) axes and gyroscopic effects which are identified in [4]. Thus, all these parameters taking part in the installation of a new configuration of the model and in the development of a new state representation of the system, much more complete, allow the synthesis of adequate control laws. Then, a new design of a fuzzy logic controller based on Takagi-Sugeno approach is presented, for ensuring the trajectory tracking along (*X, Y, Z*) axes and roll, pitch and yaw angles, while the desired roll and pitch angles are deduced from nonholonomic constraints.

2. Notation

The notation used throughout the paper is stated below.

Nomenclatures:

m	*total mass of the quadrotor*
J	*inertia matrix*
l	*distance between the mass centre of the quadrotor and the rotation axes of propeller*
δ	*disturbances applied to the quadrotor*
K_{fa}	*aerodynamics frictions factors*
K_{fd}	*translation drag coefficients*
J_r	*total rotational moment of inertia around the propeller axis*
C_p	*lift coefficient*
C_d	*drag factor of rotation*

Greek Symbols:

φ,ϑ,ψ	*pitch, roll, yaw angles*
Ω	*angular velocity*
ω	*angular speed of the rotor*

3. Quadrotor dynamic modeling

Quadrotor has six DOF and four actuators placed in a cross configuration. Using a symmetrical design of the quadrotor allows for a centralization of the control systems and the payload. Each one of the four rotors is connected to a propeller and all the propellers' axes of rotation are parallel to each other. Also, all the propellers have fixed-pitch blades and their airflow downwards to get an upward lift. The left and the right propellers rotate clock wise, while the front and the rear one rotate counter-clockwise. Using an opposite pair's directions will balance the quadrotor and remove the need for a tail rotor. Consequently, the movements of the quadrotor are directly related to the propellers velocities.

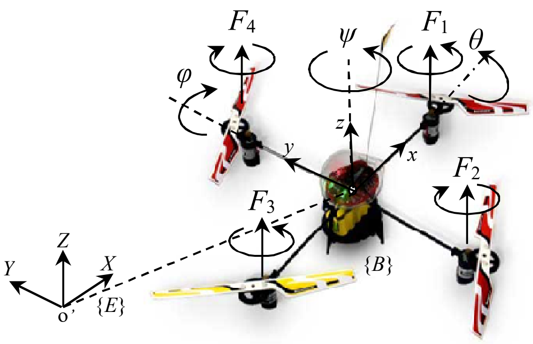


Fig.1. A quadrotor configuration [4], [5]

The rotational transformation matrix *R* between the earth frame and the body frame (see Fig. 1) is given by the resultant transformation matrix of *x, y* and *z* axes [14].

$$R = \begin{pmatrix} C\psi\, C\theta & C\psi\, S\theta\, S\varphi - S\psi\, C\varphi & C\psi\, S\theta\, C\varphi - S\psi\, S\varphi \\ S\psi\, C\theta & S\psi\, S\theta\, S\varphi + C\psi\, C\varphi & S\psi\, S\theta\, C\varphi - S\varphi\, C\psi \\ -S\theta & C\theta\, S\varphi & C\theta\, C\varphi \end{pmatrix} \quad (1)$$

Where *S* and *C* represent the Sinus and Co-sinus functions, respectively.

Using the Newton-Euler mechanics laws, the quadrotor motion equations can be written as follows:

$$\begin{cases} \dot{\xi} = \upsilon \\ m\,\dot{\upsilon} = F_f + F_t + F_g \\ \dot{R}_t = R_t\,\Gamma(\Omega) \\ J\dot{\Omega} = -\Omega \wedge J\Omega + \tau_f - \tau_a - \tau_g \end{cases} \quad (2)$$

Where J = diag(I_x,I_y,I_z) $\in \mathbf{R}^{3\times3}$. ($\wedge$ denote the vector cross-product).

$\Gamma(\Omega)$ is a skew-symmetric matrix, which is given as

$$\Gamma(\Omega) = \begin{pmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{pmatrix} \quad (3)$$

The quadrotor dynamic is given by the following equations:

$$\begin{cases} \ddot{\varphi} = \frac{1}{I_x}\left\{ \dot{\theta}\dot{\psi}\left(I_y - I_z\right) - K_{fax}\dot{\varphi}^2 - J_r\overline{\Omega}\dot{\theta} + lU_2 \right\} \\ \ddot{\theta} = \frac{1}{I_y}\left\{ \dot{\varphi}\dot{\psi}\left(I_z - I_x\right) - K_{fay}\dot{\theta}^2 + J_r\overline{\Omega}\dot{\varphi} + lU_3 \right\} \\ \ddot{\psi} = \frac{1}{I_z}\left\{ \dot{\theta}\dot{\varphi}\left(I_x - I_y\right) - K_{faz}\dot{\psi}^2 + U_4 \right\} \\ \ddot{x} = \frac{1}{m}\left\{ (\cos\varphi.\sin\theta.\cos\psi + \sin\varphi.\sin\psi)U_1 - K_{fdx}\dot{x} \right\} \\ \ddot{y} = \frac{1}{m}\left\{ (\cos\varphi.\sin\theta.\sin\psi - \sin\varphi.\cos\psi)U_1 - K_{fdy}\dot{y} \right\} \\ \ddot{z} = \frac{1}{m}\left\{ \cos\varphi\cos\theta.U_1 - K_{fdz}\dot{z} \right\} - g \end{cases} \quad (4)$$

Where K_{fa} = diag(K_{fax},K_{fay},K_{faz}), K_{fd} = diag(K_{fdx},K_{fdy},K_{fdz}), u_x and u_y are two virtual control inputs

$$\begin{cases} U_x = \cos\varphi.\sin\theta.\cos\psi + \sin\varphi.\sin\psi \\ U_y = \cos\varphi.\sin\theta.\sin\psi - \sin\varphi.\cos\psi \end{cases} \quad (5)$$

The system control inputs U_1, U_2, U_3 and U_4 are represents altitude, pitch, roll and yaw controls, respectively. Are written according to the angular velocities of the four rotors as follows:

$$\begin{aligned} U_1 &= C_p\,(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ U_2 &= C_p\,(-\omega_1^2 + \omega_3^2) \\ U_3 &= C_p\,(-\omega_2^2 + \omega_4^2) \\ U_4 &= C_d\,(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{aligned} \quad (6)$$

Where $\overline{\Omega} = (\omega_1 - \omega_2 + \omega_3 - \omega_4)$ is the total gyroscopic torques affected the quadrotor.

From Eq.5, it is easy to show that:

$$\begin{cases} \varphi_d = \arcsin\left(U_x \sin(\psi_d) - U_y \cos(\psi_d)\right) \\ \theta_d = \arcsin\left(\left(U_x \cos(\psi_d) + U_y \sin(\psi_d)\right)\big/\cos(\varphi_d)\right) \end{cases} \quad (7)$$

3.1 Rotor dynamics

A standard DC motor is usually a 2nd order system. It is possible to model the dynamics of a DC motor system as a first order system [15]. In this paper, the transfer function Eq.8 of the dynamics of a DC motor system is used:

$$G(s) = \frac{k}{\tau s + 1} \quad (8)$$

where *k* and *τ* are the *gain* and the *time constant* of the *motor*, respectively.

3.2 State space representation

The model Eq.4 can be written in the state-space as:

$$\dot{X} = f(X) + g(X)U + \delta \quad (9)$$

where *δ* represent the perturbation and *X* the state vector.

The quadrotor robot is six degrees of freedom system defined with twelve states. The following state and control vectors are adopted:

$$X = [\varphi, \dot{\varphi}, \theta, \dot{\theta}, \psi, \dot{\psi}, x, \dot{x}, y, \dot{y}, z, \dot{z}]^T \quad (10)$$

U is the input vector such as:

$$U = [U_1\ U_2\ U_3\ U_4]^T \quad (11)$$

Where U_i is the motor control input, *i* =1, 2, 3, 4: motor number.

$$f(X)=\begin{pmatrix} x_2 \\ a_1x_4x_6+a_2x_2^2+a_3\,\overline{\Omega}\,x_4 \\ x_4 \\ a_4x_2x_6+a_5x_4^2+a_6\overline{\Omega}\,x_2 \\ x_6 \\ a_7x_2x_4+a_8x_6^2 \\ x_8 \\ a_9x_8 \\ x_{10} \\ a_{10}x_{10} \\ x_{12} \\ a_{11}x_{12}-g \end{pmatrix},$$

$$g(X)=\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & b_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & b_2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b_3 \\ 0 & 0 & 0 & 0 \\ \dfrac{U_x}{m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \dfrac{U_y}{m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \dfrac{\cos x_1.\cos x_3}{m} & 0 & 0 & 0 \end{pmatrix} \qquad (12)$$

From Eqs. (9)-(12) the following state representation is obtained:

$$\begin{cases} \dot{x}_1=x_2 \\ \dot{x}_2=a_1x_4x_6+a_2x_2^2+a_3\,\overline{\Omega}\,x_4+b_1U_2+\delta \\ \dot{x}_3=x_4 \\ \dot{x}_4=a_4x_2x_6+a_5x_4^2+a_6\overline{\Omega}\,x_2+b_2U_3+\delta \\ \dot{x}_5=x_6 \\ \dot{x}_6=a_7x_2x_4+a_8x_6^2+b_3U_4+\delta \\ \dot{x}_7=x_8 \\ \dot{x}_8=a_9x_8+U_xU_1/m+\delta \\ \dot{x}_9=x_{10} \\ \dot{x}_{10}=a_{10}x_{10}+U_yU_1/m+\delta \\ \dot{x}_{11}=x_{12} \\ \dot{x}_{12}=a_{11}x_{12}+\cos(x_1)\cos(x_3)U_1/m-g+\delta \end{cases} \qquad (13)$$

where:

$a_1=(I_y-I_z)/I_x$ $a_2=-K_{fax}/I_x$ $a_3=-J_r/I_x$ $a_4=(I_z-I_x)/I_y$
$a_5=-K_{fay}/I_y$ $a_6=J_r/I_y$ $a_7=(I_x-I_y)/I_z$ $a_8=-K_{faz}/I_z$
$a_9=-K_{ftx}/m$ $a_{10}=-K_{fty}/m$ $a_{11}=-K_{ftz}/m$ $b_1=d/I_x$
$b_2=d/I_y$ $b_3=1/I_z$

4. Design of the fuzzy controller

The main objective is to design control laws in order to stabilize the position, altitude and attitude of the quadrotor and tracking the desired trajectory by using a FLC [16],[17].

From nonholonomic constraints developed in Eq.7, the roll (ϕ) and pitch (θ) angles depend not only on the yaw angle (ψ) but also on the movements along (*x*, *y*) axes and their dynamics. However the adopted control strategy is summarized in the control of two subsystems: the first relates to the orientation (attitude) control, taking into account the position control along (*x, y*) axes while the second is that of the altitude control along *z* axis as shown in the synoptic scheme below (see Fig. 2).

In Fig. 2 the controller that will be implemented consists of six FLCs, three attitude controls (FLC$_{\text{roll}}$, FLC$_{\text{pitch}}$, FLC$_{\text{yaw}}$) and three position controllers (FLC$_x$, FLC$_y$, FLC$_z$) who are in the form of zero order Sugeno fuzzy inference system Eq.14 [18].

$$U=FLC(e,\dot{e})=\frac{\sum_{k=1}^{n}\mu_{k,1}(e).\mu_{k,2}(\dot{e}).C_k}{\sum_{k=1}^{n}\mu_{k,1}(e).\mu_{k,2}(\dot{e})} \qquad (14)$$

The controller inputs are the state variables errors. The error is calculated as the difference between the desired state and the current state (X_d-X).

The linguistic levels of the inputs signal error (*e*) are assigned as : (N) negative, (Z) zero, and (P) positive, where the range of the error (*e*) input is from -100 to +100 and the linguistic levels of the derivative error (*de*) input signal are the same as the error (*e*) inputs but the range value is from -100 to +100, and they are presented in Fig. 4. The output of the FLC is the control action *U*. The output membership functions are fuzzy singletons CN = −1, CZ = 0 and CP = 1.

$\mu_N(e) = Trapezoidal\{4^x(-25, -25, -3, 0)\}$
$\mu_Z(e) = Triangular\{4^x(-3, 0, 3)\}$
$\mu_P(e) = Trapezoidal\{4^x(0, 3, 25, 25)\}$
$\mu_N(de) = Trapezoidal\{4^x(-25, -25, -3, 0)\}$
$\mu_Z(de) = Triangular\{4^x(-3, 0, 3)\}$
$\mu_P(de) = Trapezoidal\{4^x(0, 3, 25, 25)\}$
$\mu_N(U) = -1$ $\mu_Z(U) = 0$ $\mu_P(U) = +1$

The rule set of FLC containing nine rules which governing the input-output relationship of the FLC and this adopts the Sugeno-style inference engine [18]. The rules set whose forms are established as follows (see Table 1).

Rule *i*: If *e is* $\mu_{k,1}$ and *de is* $\mu_{k,2}$ Then *u is* C_i

where $\mu_{k,1}$ and $\mu_{k,2}$ are the fuzzy set associated with each input variable and C_i is a constant associated with output variable with *i=1,2,3*, when the output of each rule is constant.

Table 1. Fuzzy rule set

		e		
		Z	N	P
	Z	CN	CN	CZ
de	N	CN	CZ	CP

P $\overline{CZ}$ CP CP

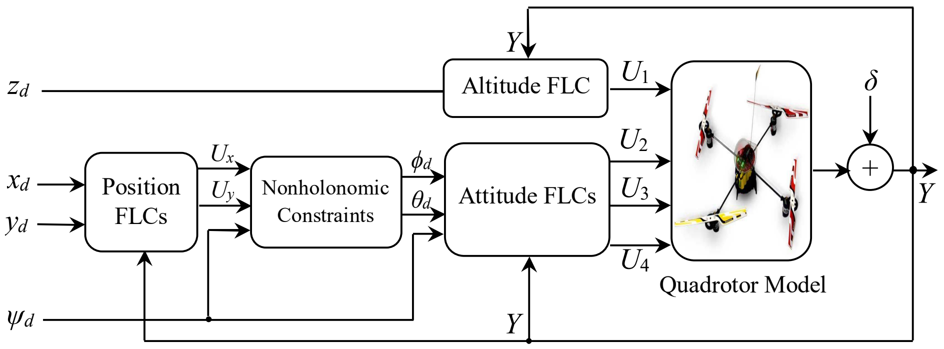


Fig.2. Synoptic scheme of the proposed control strategy

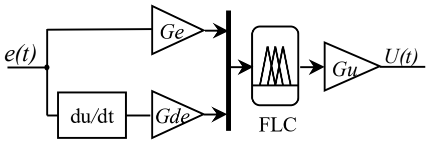


Fig.3. Structure of individual fuzzy controller

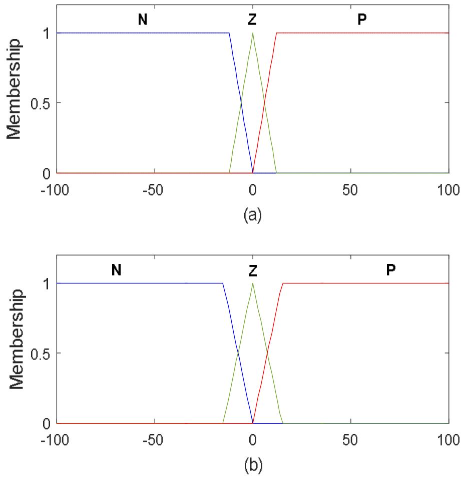


Fig.4. Membership functions for inputs variable (a) *e*, (b) *de*

These input variables are multiplied by the gains G_e and G_{de}, respectively. The output is the control action U and the corresponding gain is G_u. The tuning of these gains have been done by trial and error way. The implication and aggregation methods are fixed and the weighted average method to realize the defuzzification procedure is used.

5. Simulation Results

In this section, we will test the effectiveness of the proposed control framework. Proposed control applied to the above Quadrotor is simulated on a PC using MatLab/simulink environment (version 8.6.0.267246).

The obtained controller gains are:

G_{e1}=30 G_{de1}=13 G_{U1}=15 ; G_{e2}=1 G_{de2}=1 G_{U2}=2 ; G_{e3}=1 G_{de3}=1 G_{U3}=2 ; G_{e4}=1 G_{de4}=1 G_{U4}=3 ; G_{ex}=1 G_{dex}=1 G_{Ux}=4 ; G_{ey}=1 G_{dey}=1 G_{Uy}=4

In a first simulation, a state initial condition of the system is assumed as: $x=y=z$ = 0 and $\phi = \vartheta = \psi$ = 0.

The desired end state is represented by x_d = 0.5m, y_d = 0.5m z_d = 3m and ψ = 160º. In this test the system was not subject to any disturbance. The following results are obtained, with the term IC representing the initial conditions:

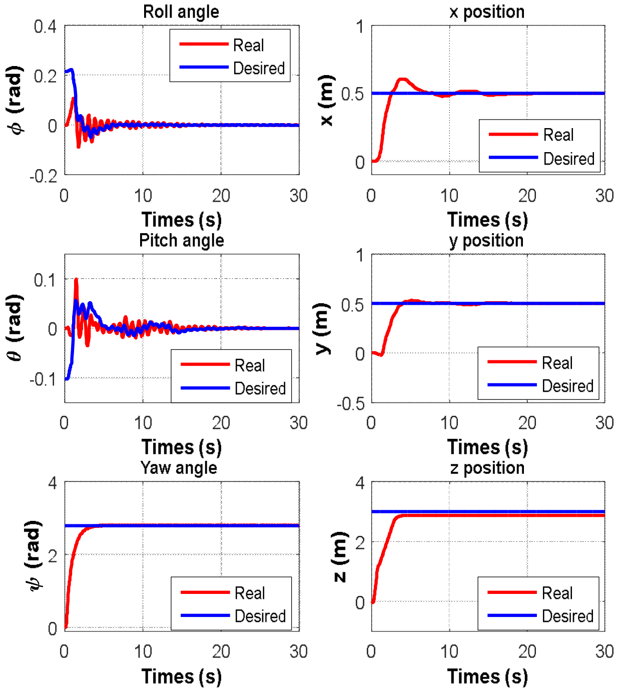


Fig.5. Trajectory tracking of Position and Euler angles with IC=0

Another desired trajectory is taken now with: x_d= 0.2 sin ((t+π)/2), y_d= 0.2 sin ((t+π)/2+1), z_d= t , ψ = 60°. In this test the system is not subject to any disturbance and the following results are obtained:

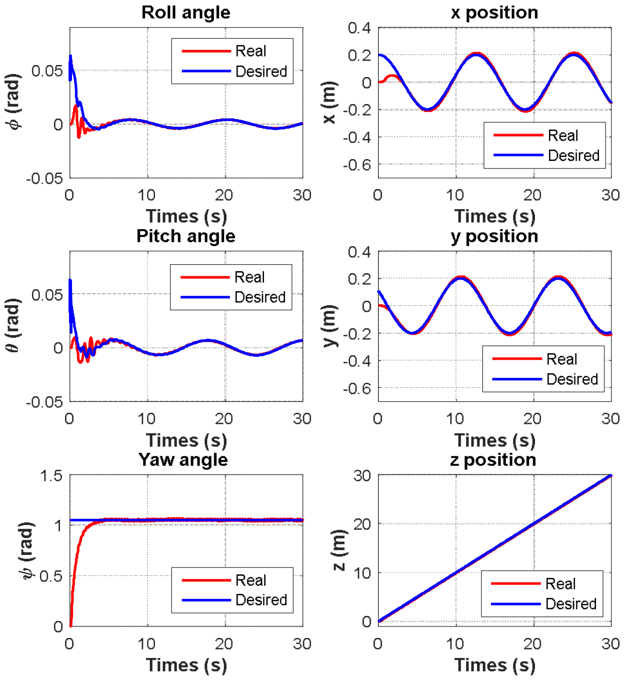


Fig. 6. Position and Euler angles tracking with IC=0

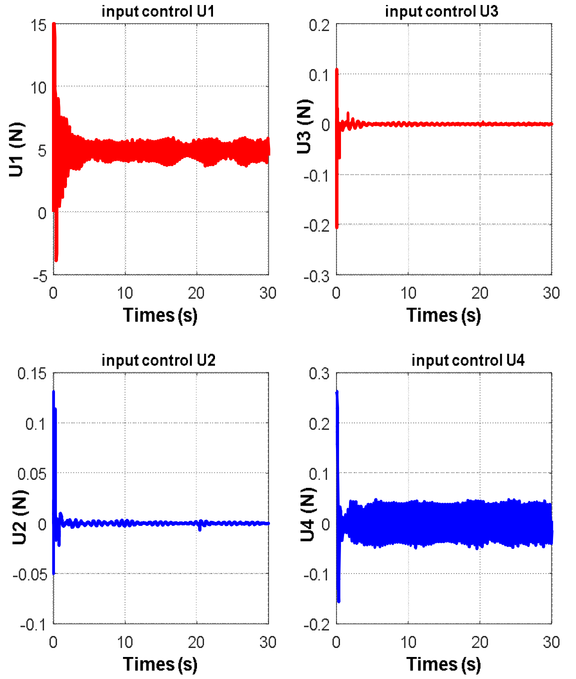


Fig. 8. Control inputs tracking with IC=0.

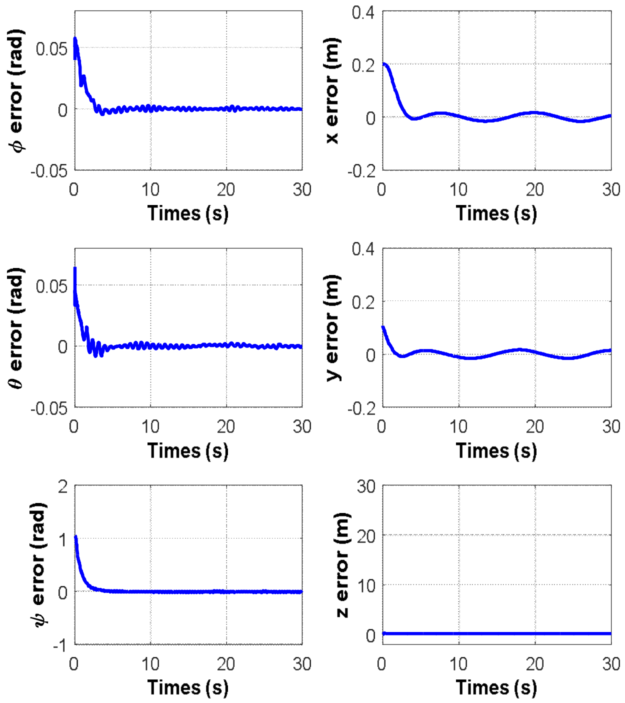


Fig.7. Positions and Orientations error tracking with IC=0

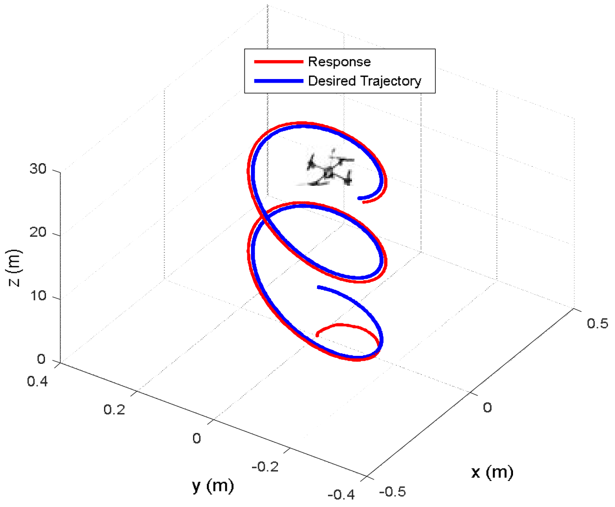


Fig. 9. Quadrotor displacement in 3D with IC=0

The considered initial conditions are *X*(*0*) = IC = [5°, 0, 7°, 0, 0, 0, 0.5, 0.2, 1, 0, 2, 0].

The desired trajectory is : x_d = -sin (π.t/30 + π/2) ; y_d = -3 sin (π.t/30 +π) ; z_d = -3 sin (π.t/30 +π/2) +5 ; ψ = 90°.

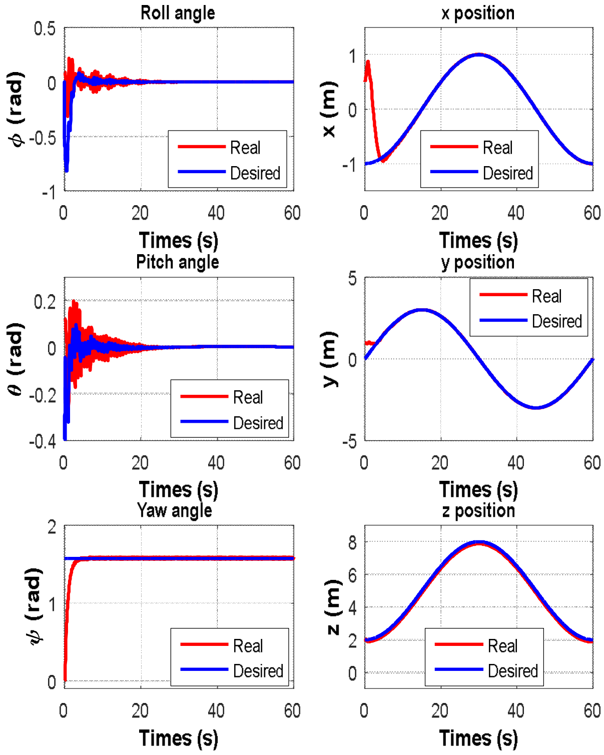


Fig.10. Position and Euler angles tracking with IC = [5°,0,7°,0,0,0,0.5,0.2,1,0,2,0].

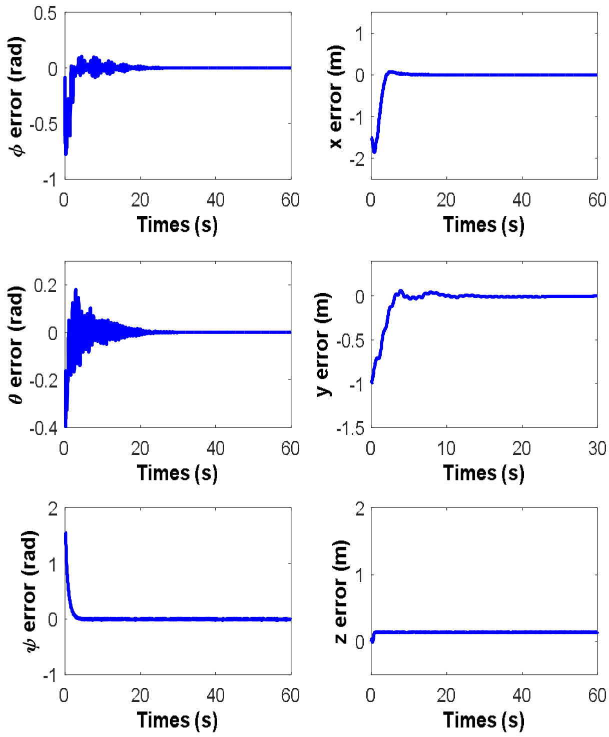


Fig.11. Positions and orientations error tracking with IC = [5°,0,7°,0,0,0,0.5,0.2,1,0,2,0].

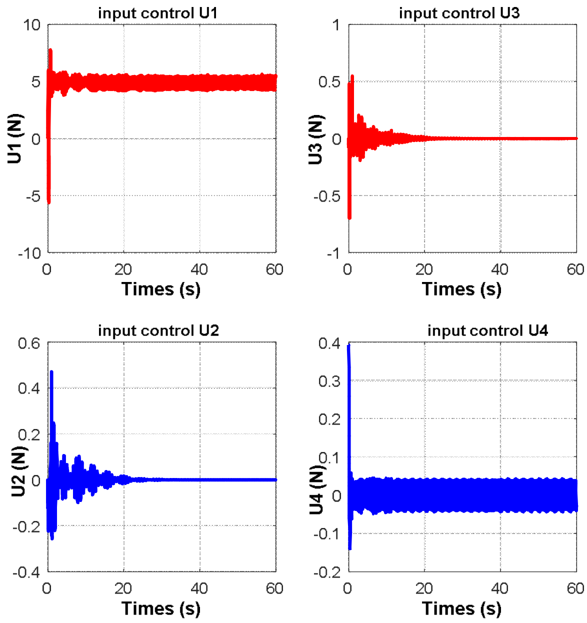


Fig.12. Control inputs with IC = [5°,0,7°,0,0,0,0.5,0.2,1,0,2,0].

From Figs. 5, 6 and 10 we can see that *x, y, z* and *ψ* (yaw Angle) achieve the desired value in a relatively short period of time, *ϕ* (roll angle) and *ϑ* (pitch angle) are more stable (equal 0) with the short period of time. Errors and control inputs are depicted in Figs. 7, 11 and Figs. 8 and 12, respectively for different initial conditions.

In a second simulation the system Eq.9 is subject to disturbance. To illustrate the disturbance rejection performance, the external disturbance is assumed to be $\delta = \exp\left(-(t-c_i)^2 / (2.b_i^2)\right)$ with $b_i = 1.0$ and $c_i = 30$ for $t = 60$s.

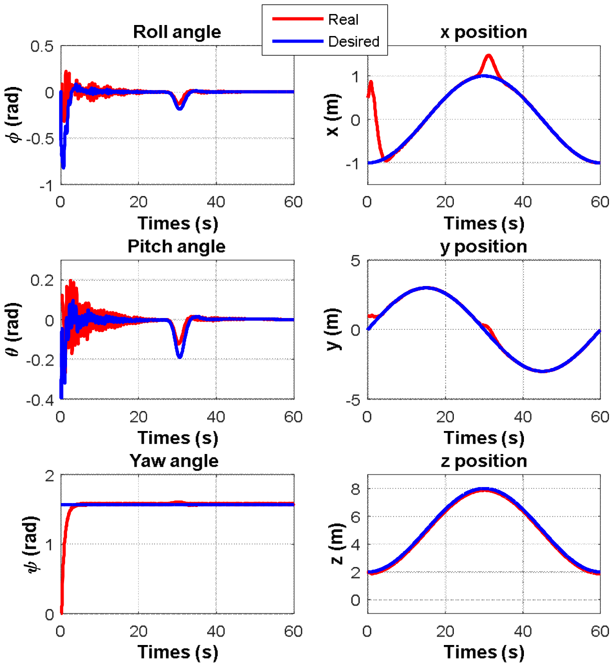


Fig.13. Position and Euler angles tracking with IC = [5°,0,7°,0,0,0,0.5,0.2,1,0,2,0] and disturbance.

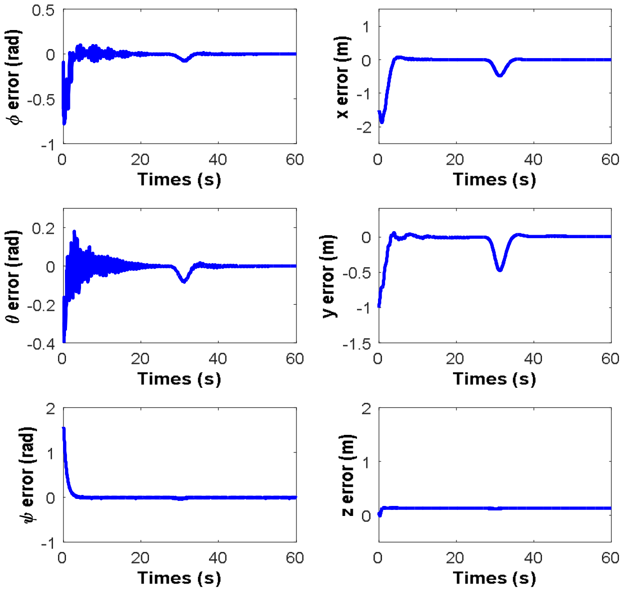


Fig. 14. Positions and orientations error tracking with IC = [5°,0,7°,0,0,0,0.5,0.2,1,0,2,0] and disturbance.

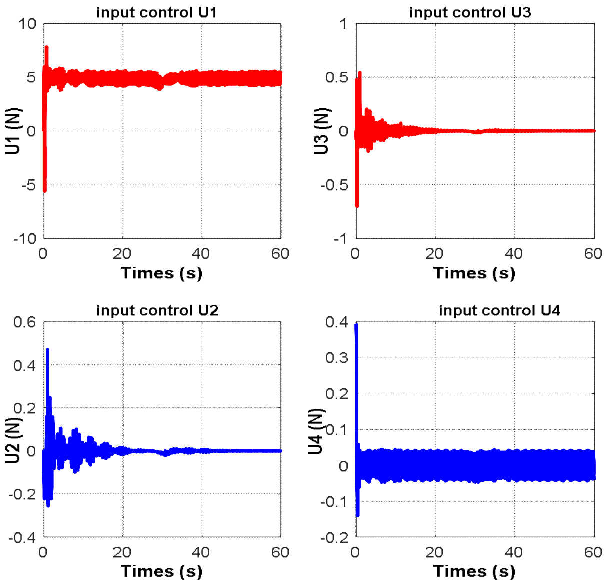
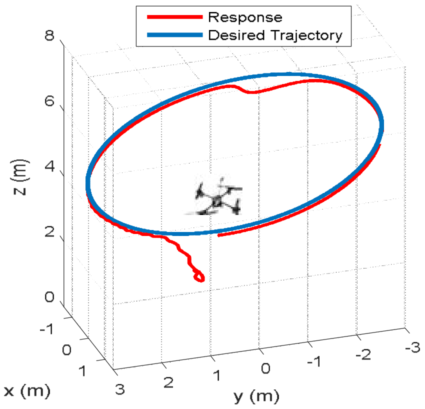
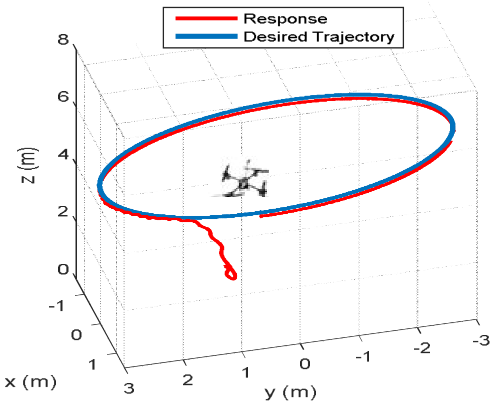


Fig.15. Control inputs with IC = [5°,0,7°,0,0,0,0.5,0.2,1,0,2,0] and disturbance.



(a)



(b)

Fig.16. Quadrotor displacement in 3D with IC = [5°,0,7°,0,0,0,0.5,0.2,1,0,2,0] (a) with disturbance (b) without disturbance.

From Fig. 14 we can see *x, y, z* and *ψ* achieve the desired value in a relatively short period of time despite the presence of disturbances. *ϕ* and *ϑ* would be more stable (equal to zero) in a short period of time despite the presence of disturbances. Errors and control inputs are depicted in Figs. 15 and 16, respectively. As represented in Fig. 17 the performances without and under the occurrence external disturbance are satisfactory. So the obtained results of simulation works show indeed that the proposed controller is really robust.

The values of the parameters used in simulation can be found in [4, 8] and can be seen in Table 2.

Table 2. Parameters used in simulation

m = 400 [g]
l = 20.5 [cm]
g = 9.81 [m·s^{-2}]
a_1 = -1
a_2 = -0.1454
a_3 = -0.0074
a_4 = 1
a_5 = -0.1454
a_6 = 0.0074
a_7 = -1.3061×10^{-4}
a_8 = -0.0830
a_9 = -0.0011
a_{10} = -0.001
a_{11} = -0.0013
b_1= 65.3117
b_2= 65.2946
b_3= 130.6063
C_p= 2.9842×10^{-5} [N/rad/s]
C_d= 3.2320×10^{-7} [N.m/rad/s]
J_r = 2.8385×10^{-5} [N.m/rad/s^2]
K_{fa}= diag(5.5670; 5.5670; 6.3540)×10^{-4} [N/rad/s]
K_{fd}= diag(0.032; 0.032; 0.048) [N/m/s]
J= diag(3.8278; 3.8278; 7.1345)×10^{-3} [N.m/rad/s^2]

6. Conclusion

An intelligent approach using a fuzzy logic controller based on Takagi-Sugeno property to solve the stability and trajectory tracking problems of a nonlinear model is proposed. The simulation results verify that the dynamics model of the quadrotor is correct and the control system is feasible and valid. Moreover, the designed fuzzy logic control system is capable to reach the desired hovering point quickly. Also, simulation results confirmed the ability of the designed control scheme to ensure stability and tracking trajectory of the quadrotor model, in the presence of both nonholonomic constraints and external disturbances.

References

[1] _Min BC et al. Development of a Micro Quad-Rotor UAV for Monitoring an Indoor Environment, In: Proc Springer Advances in Robotics, FIRA RoboWorld Congr, Korea, p. 262-271, 2009. Doi: 10.1007/978-3-642-03983-6_30.

[2] Tayebi A, Gilvray SMc. Attitude stabilization of a VTOL Quadrotor Aircraft. IEEE Trans Control Syst Technol 2006;14:562–571.

[3] Bouadi H et al. Modeling and Stabilizing Control Laws Design Based on Backstepping for an UAV Type-Quadrotor, In: Proc Elsevier, 6th IFAC Symp on Intelligent Autonomous Vehicles, 2007. Doi:10.3182/20070903-3-FR-2921.00043.

[4] Derafa L et al. Dynamic modelling and experimental identification of four rotor helicopter parameters, IEEE-ICIT, Mumbai, India, p. 1834–1839, 2006.

[5] Castillo P et al. Real-Time Stabilization and Tracking of a Four-Rotor Mini Rotorcraft. IEEE Trans Control Syst Technol 2004;12:510–516.

[6] Hoffmann GMet al. Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment, AIAA Guidance, Navigation and Control Conf and Exhibit, South Carolina, August,2007. Doi: 10.2514/6.2007-6461

[7] Bouabdallah S, Siegwart R. Backstepping and Sliding-mode Techniques Applied to an Indoor Micro Quadrotor, In: Proc IEEE Int Conf Robot Autom, Barcelona, Spain, p. 2247–2252, 2005.

[8] Bouadi H et al. Sliding Mode Control Based on Backstepping Approach for an UAV Type-Quadrotor. Int J Applied Mathematics and Comput Sciences 2007;4:12–17.

[9] Voos H. Nonlinear Control of a Quadrotor Micro-UAV using Feedback-Linearization, In: Proc of the IEEE Int Conf on Mechatronics, Malaga, Spain, 2009.

[10] Saif A et al. Modified backstepping control of Quadrotor, In: Proc of the 9th Int Multi-Conf on Syst, Signals and Devices, Germany, p. 1–6, 2012.

[11] Carrillo LRG et al. Hovering Quad-Rotor Control: A Comparison of Nonlinear Controllers Using Visual Feedback. IEEE Trans Aerosp Electron Syst 2012,48:3159–3170.

[12] Rosales C et al. Trajectory Tracking of a Mini Four-Rotor Helicopter in Dynamic Environments - a Linear Algebra Approach, Robotica 2015;33:1628–1652.

[13] Freddi A et al. Nonlinear Decentralized Model Predictive Control for Unmanned Vehicles Moving in Formation. Inf Technol and Control 2015;44:89–97.

[14] Castillo P et al. The Quad-rotor Rotorcraft. In: Modelling and Control of Mini-flying Machines. Springer-Verlag, London; 2005, p. 39–59.

[15] Franklin GF et al. Feedback Control of Dynamic Systems. 4th ed. Prentice Hall; 2002.

[16] Passino KM, Yurkovich S. Fuzzy Control. Addison Wesley Longman, USA; 1998.

[17] El-Shimy ME, Zaid SA. Fuzzy PID Controller for Fast Direct Torque Control of Induction Motor Drives. J Elect Syst 2016;12:687–700.

[18] Sugeno M. Industrial Applications of Fuzzy Control. 1st ed. Elsevier Science Inc. New York: NY, USA; 1985.