



HAL
open science

Optimal model-free fuzzy logic control for autonomous unmanned aerial vehicle

Hossam Eddine Glida, Latifa Abdou, Abdelghani Chelihi, Chouki Sentouh,
Gabriele Perozzi

► **To cite this version:**

Hossam Eddine Glida, Latifa Abdou, Abdelghani Chelihi, Chouki Sentouh, Gabriele Perozzi. Optimal model-free fuzzy logic control for autonomous unmanned aerial vehicle. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 2021, 235 (6), 10.1177/09544100211025379 . hal-03424975

HAL Id: hal-03424975

<https://uphf.hal.science/hal-03424975v1>

Submitted on 24 Sep 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Optimal model-free fuzzy logic control for autonomous unmanned aerial vehicle

Hossam E Glida¹, Latifa Abdou², Abdelghani Chelhi^{2,3}, Chouki Sentouh⁴ and Gabriele Perozzi⁵

Abstract

This article deals with the issue of designing a flight tracking controller for an unmanned aerial vehicle type of quadrotor based on an optimal model-free fuzzy logic control approach. The main design objective is to perform an automatic flight trajectory tracking under multiple model uncertainties related to the knowledge of the nonlinear dynamics of the system. The optimal control is also addressed taking into consideration unknown external disturbances. To achieve this goal, we propose a new optimal model-free fuzzy logic-based decentralized control strategy where the influence of the interconnection term between the subsystems is minimized. A model-free controller is firstly designed to achieve the convergence of the tracking error. For this purpose, an adaptive estimator is proposed to ensure the approximation of the nonlinear dynamic functions of the quadrotor. The fuzzy logic compensator is then introduced to deal with the estimation error. Moreover, the optimization problem to select the optimal design parameters of the proposed controller is solved using the bat algorithm. Finally, a numerical validation based on the Parrot drone platform is conducted to demonstrate the effectiveness of the proposed control method with various flying scenarios.

Keywords

Unmanned aerial vehicle-drone, model-free control, optimization, bat algorithm

Introduction

Over the last decade, the unmanned aerial vehicle (UAV) type of quadrotor has motivated many research studies due to their several applications such as today's shooting techniques which require drones equipped with digital systems to make films that cannot be achieved by classical means. Another application where quadrotor UAVs are used is road monitoring in order to collect data about the traffic situation and road safety. The drone type of quadrotor is also used in the field of agriculture to control crop fields more effectively. Many other applications related to the UAV can be found in Refs. 1, 2. Regardless of whether the quadrotor UAV is in remotely controlled or fully autonomous, the aircraft presents high nonlinearities and strong coupling effect between their state variables. They are multi-input–multi-output, open-loop unstable, and underactuated systems. Moreover, in most cases, they are intended to operate in harsh environments which may possibly include physical constraints, measurement noise, and strong disturbance winds. For these reasons, the design of trajectory tracking control for the quadrotor UAV still remains challenging.

Recently, several linear and nonlinear control approaches were applied on the UAV type of quadrotor using

various control strategies. Although these approaches have given interesting results, they have certain limitations. The proportional–integral–derivative (PID) and linear quadratic (LQ) regulators are sensitive to parametric variations of the quadrotor system and the effect of the disturbance.^{3,4} The feedback linearization approach relies on full knowledge of the dynamics to guarantee robust tracking performances.⁵ The chattering problem well known in the sliding mode control (SMC) approach was resolved using the disturbance observer-based SMC in

¹ Department of Electrical Engineering, LMSE Laboratory, University of Biskra, Biskra, Algeria

² Department of Electrical Engineering, LI3CUB Laboratory, University of Biskra, Biskra, Algeria

³ Department of Electronics, Faculty of Technology, Constantine I University, Constantine, Algeria

⁴ Hauts-de-France Polytechnic University, CNRS, INSA Hauts-de-France, LAMIH, Valenciennes, France

⁵ Univ.Lille, INRIA, Centrale Lille, CNRS, UMR 9189-CRISTAL, F-59000 Lille, France

Corresponding author:

Hossam E Glida, Department of Electrical Engineering, LMSE Laboratory, University of Biskra, Biskra 07000, Algeria.

Email: hossam.gld@hotmail.com

Refs. 6, 7 and modified SMC in Refs. 8, 9. In the other hand, the backstepping controller (BC) has known successful wide uses in the literature for quadrotor control as in Ref. 10. Moreover, hybrid control strategies have also been reported by combining two or more methods for improving the control performances. For example, in Ref. 11, the BC approach is combined with terminal SMC to design robust adaptive tracking to control the quadrotor orientation and position. A hierarchical control scheme which combines model predictive control (MPC) with nonlinear H_∞ control for quadrotor stabilization was proposed in Ref. 12. Intelligent controllers such as fuzzy logic control (FLC),^{13–15} and neural network control^{16–18} are also investigated with the previous approaches to solve such control problems.

Since the quadrotor is characterized by its nonlinearities (i.e., external disturbance parameter variation which includes the aerodynamics force effect), the control approaches should achieve the tracking performance and guarantee the system stability. However, the control strategies applied to the drones have become increasingly more complicated, which need huge computing to be implemented in real time. Furthermore, most of the mentioned works are based on the required knowledge of the ideal model, while it exists an unmodeled dynamics in the system. Although the existing controllers might be sufficient to the UAV control and can track the desired trajectory, the drawback is the inevitable presence of parametric uncertainties and unknown external disturbances which could compromise the tracking performance. Hence, it is necessary to design the corresponding controller for the system considering the unknown nonlinearities. Approaches such as model-free control (MFC) seem promising ones.¹⁹ The main benefit of this control approach is that it does not require prior knowledge about the dynamic model which can be easily applied to control unknown/complex systems, such as quadrotor UAV. Up to now, many research results on MFC of the quadrotor UAV have been presented.^{20–27}

In Ref. 20, the proposed model-free control law is composed of two parts: the first part is a linear control term used to specify the dynamics of the closed-loop system, and the second part is a compensator of the effects of uncertainties and external disturbances which is synthesized using the nonlinear estimator. In Ref. 21, a MFC based on ultra-local model control was designed for a tracking control of an UAV. Authors in Ref. 22 proposed a model-free control applied on a quadrotor UAV based on adaptive proportional–derivative–sliding mode control. Model-free control based on the proportional–derivative (PD) controller was proposed for a quadrotor in Ref. 23. However, the control strategies which were developed in these works are based on the known ultra-local model that must be defined in a short operating time, but they do not define any specific criteria to select the designed parameters which leads to a great limitation of these approaches. Optimal model-free control based on the backstepping technique using an estimator approach to approximate the unknown dynamic model functions was investigated in our previous work,²⁴ where the controller's parameters are tuned using a new metaheuristic algorithm called “the cuckoo search

technique.” This approach is based on many derivatives about the variables which lead to a significant increase in the design complexity of such a controller.

In Ref. 25, an indirect adaptive model-free fuzzy fault-tolerant tracking control for the attitude quadrotor system with actuator and sensor failures is developed. In this work, 15 fuzzy systems are used in the control law with a complex updating algorithm to deal with the unknown dynamics and failures.

An adaptive fuzzy MFC approach is proposed in Ref. 26 for 3D trajectory tracking of the quadrotor. The authors propose a combination of classical PD controller with fuzzy system which is used to tune online the PD control gains and to deal with the uncertainties and external disturbances. However, the stability proof of the closed-loop control system is not guaranteed which reduces its reliability. In Ref. 27, a model-free and learning control algorithm using type-2 fuzzy neural networks (T2-FNNs) are designed to achieve a desirable translational and rotational control of the quadcopter for agricultural purposes. This study proposes a novel hybrid optimization algorithm combining particle swarm optimization theory and sliding mode control for the training of the T2-FNNs. The cited works mainly aim to enhance the UAV's control performance face challenges in incorporating and handling various uncertainties, disturbances, and failures. However, the proposed fuzzy systems in these works are based on complex learning methods, have a hefty structure, and cannot adapt to rapidly changing environments in more tricky flights.

Motivated by the requirement for the simple and efficient controller which has strong robustness to external disturbances and can be implemented easily in practical applications, we propose in this work an optimal model-free fuzzy logic controller (OMFFC) based on the bat algorithm applied to a quadrotor UAV. The aim of this work is to design a control law without any prior knowledge about the model system which allows to track the desired path for the quadrotor navigation in presence of external disturbances. The OMFFC design is performed in four steps. First, the model-free control approach is applied to reduce the controller's complexity, to take into consideration the required performance, and to reduce the high-order derivative output. Then, an adaptive estimator–reliant input–output data of the quadrotor system are used to approximate the unknown nonlinear dynamics and disturbances of the quadrotor UAV. After that, the fuzzy logic system of Refs. 28, 29 is introduced to compensate the estimation error and improve the tracking control properties which ensure robustness. Finally, to overcome the drawback of the design parameters selection via trial and error, we propose to incorporate the bat algorithm (BA) which has been widely used in several applications such as in Refs. 30, 31. Thus, this article investigates an optimal model-free controller design for a UAV-drone where the main contributions can be summarized as follows:

- A new robust model-free control strategy–based tracking control approach for an UAV type of quadrotor is proposed. For the controller design, a PID

controller and an adaptive estimator of unknown dynamics and external disturbance with the fuzzy logic control approach are combined to guarantee the robust tracking convergence of the closed-loop quadrotor system.

- The stability of the overall closed-loop system and the asymptotic convergence of the tracking errors are analyzed using the Lyapunov approach which guarantees the performance of the designed controller.
- Based on the bat metaheuristic method, an optimization algorithm is proposed to solve the optimization problem of selecting the design parameters for the proposed controller while minimizing an objective function to ensure the best performance of the quadrotor system.
- A decentralized structure of the designed control scheme that only requires local measurements is used to ensure that the control law is simpler and easier to implement in real time. The effectiveness of the proposed OMFFC approach is clearly proved with numerical validations based on the Parrot drone platform.

The paper is organized as follows: In the Quadrotor Model section, the mathematical model of the quadrotor UAV and the problem statement used for the simulation purposes are presented. The Flight Controller Design section gives a detailed design procedure of the optimal model-free fuzzy controller. The Optimal Model-Free Fuzzy Logic Controller section presents several numerical simulations performed using the Parrot drone platform proposed in the MATLAB/Simulink toolbox to demonstrate the efficiency of the proposed controller. Finally, the Numerical Simulation Results section provides some concluding remarks and future works.

Quadrotor model

Mathematical modeling

The UAV used in this study is a quadrotor drone as depicted in Figure 1, which is an underactuated system with six outputs. Three position outputs (x , y , z) and three orientation outputs (ϕ , θ , ψ), controlled by the total forces and torques (u_z , u_ϕ , u_θ , u_ψ), obtained by varying the rotor speeds. The quadrotor motion is described in 3D space according to two coordinate frames which are the body-frame $B_{(x_b, y_b, z_b)}$ and Earth-frame $E_{(x_e, y_e, z_e)}$, and the relationship between these two frames is given by the transformation matrix as follows

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

where $c^* \equiv \cos^*$ and $s^* \equiv \sin^*$. The quadrotor model is derived via Euler–Newton formalism using force/moment dynamics and kinematics via the following equation

$$\begin{aligned} m\dot{\Sigma} &= -mg\varpi_z + u_z R\varpi_z \\ I\dot{\Theta} &= -\Theta \times I\Theta - F_g + \tau \end{aligned} \quad (2)$$

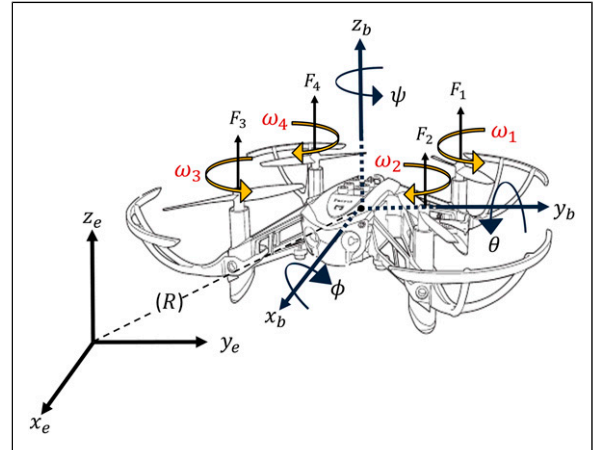


Figure 1. Quadrotor configuration.

where $\Sigma = [\dot{x}, \dot{y}, \dot{z}]^T$ is the translation velocity vector, m denotes the total mass of the quadrotor, g is the gravitational constant and $\varpi_z = [0, 0, 1]^T$ is the unit vector expressed in the frame, $I = \text{diag}[I_x, I_y, I_z]$ is the symmetric positive definite inertia matrix, $\Theta = [\phi, \theta, \psi]^T$ represents the quadrotor angular vector, F_g is the gyroscopic effect due to rigid body rotation, and $\tau = [u_\phi, u_\theta, u_\psi]^T$ is the torque inputs vector. More detailed description of the quadrotor dynamics is described thoroughly in literature.^{16,24} The adopted mathematical model considering the relation between $B_{(x_b, y_b, z_b)}$ and $E_{(x_e, y_e, z_e)}$ can be described as follows

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} ((c\psi s\theta c\phi + s\psi s\phi)u_z + d_x)/m \\ ((c\psi s\theta s\phi - c\psi c\phi)u_z + d_y)/m \\ ((c\phi c\theta)u_z + d_z)/m - g \\ ((I_y - I_z)\dot{\theta}\dot{\psi} - (J_r\Omega_r)\dot{\theta} + lu_\phi + d_\phi)/I_x \\ ((I_z - I_x)\dot{\phi}\dot{\psi} + (J_r\Omega_r)\dot{\phi} + lu_\theta + d_\theta)/I_y \\ ((I_x - I_y)\dot{\phi}\dot{\theta} + u_\psi + d_\psi)/I_z \end{bmatrix} \quad (3)$$

where l is the distance between the center of the quadrotor and the propeller, I_x , I_y , and I_z represent the inertia constants of the quadrotor, J_r is the rotor's moment, d_x , d_y , d_z , d_ϕ , d_θ , and d_ψ are the unknown disturbances which are added to the quadrotor model to represent the external forces effects like aerodynamics friction or wind, and Ω_r is the propeller angular rate computed as

$$\Omega_r = \omega_1 - \omega_2 + \omega_3 - \omega_4 \quad (4)$$

where ω_i for $i = 1, 2, 3, 4$ are the rotation speeds of the motors. The total force and torque values u_z , u_ϕ , u_θ , and u_ψ can be calculated through the rotor speeds as

$$\begin{bmatrix} u_z \\ u_\phi \\ u_\theta \\ u_\psi \end{bmatrix} = \begin{bmatrix} kl & kl & kl & kl \\ kl & 0 & -kl & 0 \\ 0 & kl & 0 & -kl \\ k\zeta & -k\zeta & k\zeta & -k\zeta \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \quad (5)$$

where k is the drag coefficients and ζ is the rotor's reaction torque constant.

Remark 1. The quadrotor model in equation (3) is obtained after some simplifications of its physical conception, so that its body is assumed to be rigid and symmetric and the rotors and propellers have the same model parameters. Furthermore, the Euler angles (roll ϕ , pitch θ , and yaw ψ) are bounded as $|\phi| < \pi/2$, $|\theta| < \pi/2$, and $|\psi| < \pi$.

Problem statement

From the mathematical model (3), we can observe that the quadrotor has six outputs $y_i \in \{x, y, z, \phi, \theta, \psi\}$ controlled by four inputs $u_i \in \{u_z, u_\phi, u_\theta, u_\psi\}$ to track the reference path $y_{id} \in \{x_d, y_d, z_d, \phi_d, \theta_d, \psi_d\}$, which leads to conclude that the quadrotor is an underactuated system. The altitude and the attitude of the quadrotor fixed by z, ϕ, θ , and ψ are controlled directly by u_z, u_ϕ, u_θ , and u_ψ , respectively. However, its Cartesian position located by x and y is conducted indirectly by two virtual control inputs noted u_x and u_y , respectively, and their expressions are obtained from equation (3) according to the following form¹¹

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = \begin{bmatrix} (c\psi s\theta c\phi + s\psi s\phi)u_z \\ (c\psi s\theta s\phi - c\psi c\phi)u_z \end{bmatrix} \quad (6)$$

The values of u_x and u_y are computed through the proposed controller that allows to generate the desired roll angle ϕ_d and the desired pitch angle θ_d by setting equation (6) as

$$\begin{bmatrix} \phi_d \\ \theta_d \end{bmatrix} = \begin{bmatrix} \arcsin(u_x s\psi - u_y c\psi) \\ \arcsin\left(\frac{u_x c\psi + u_y s\psi}{c\theta_d}\right) \end{bmatrix} \quad (7)$$

From the previous development, the quadrotor model can be seen as a complex nonlinear multi-input–multi-output system composed of six interconnected subsystems when each subsystem describes only one dynamic output. Then, to get a more comprehensible quadrotor model, the dynamic equations of motion will be rewritten in state-space form. First, let the global state variable $\mathcal{X} = [x, \dot{x}, y, \dot{y}, z, \dot{z}, \phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}]^T$, where $x_i = [x_{i1}, x_{i2}]^T$ is the local state vector of each subsystem with $y_i = x_{i1}$ and x_{i2} , its derivative, that is, $x_{i2} = \dot{x}_{i1}$, and $u_i \in \{u_x, u_y, u_z, u_\phi, u_\theta, u_\psi\}$ are the control inputs. The state equations of each subsystem are then given by

$$\begin{cases} \dot{x}_{i1} = x_{i2} \\ \dot{x}_{i2} = f_i(\mathcal{X}) + g_i(\mathcal{X})u_i + d_i(t) \\ y_i = x_{i1}, i \in \{x, y, z, \phi, \theta, \psi\} \end{cases} \quad (8)$$

where $f_i(\mathcal{X})$ and $g_i(\mathcal{X})$ are smooth functions with respect to \mathcal{X} which presents the nonlinear dynamic and control functions of the i th subsystem, respectively. With

$$\begin{aligned} f_x(\mathcal{X}) &= 0, & f_y(\mathcal{X}) &= 0, & f_z(\mathcal{X}) &= -g \\ f_\phi(\mathcal{X}) &= (I_y - I_z)/I_x x_{\theta 2} x_{\psi 2} - J_r \Omega_r / I_x x_{\theta 2} \\ f_\theta(\mathcal{X}) &= (I_z - I_x)/I_y x_{\phi 2} x_{\psi 2} + J_r \Omega_r / I_y x_{\phi 2} \\ f_\psi(\mathcal{X}) &= (I_x - I_y)/I_z x_{\theta 2} x_{\phi 2} \\ g_x(\mathcal{X}) &= 1/m, & g_y(\mathcal{X}) &= 1/m, & g_z(\mathcal{X}) &= cx_{\phi 1} cx_{\theta 1} / m \\ g_\phi(\mathcal{X}) &= l/I_x, & g_\theta(\mathcal{X}) &= l/I_y, & g_\psi(\mathcal{X}) &= 1/I_z \end{aligned} \quad (9)$$

The term $d_i(t)$ denotes the disturbances that are added to the quadrotor model considering the effects of parameters uncertainties, unmodeled dynamics, and wind effects. From equation (8), it is easy to check that the nonlinear input function $g_i(\mathcal{X})$ for each subsystem is assumed to be positive, that is, $g_i(\mathcal{X}) > 0, \forall t \geq 0$. Thus, the controllability of each subsystem as well as of the quadrotor is ensured on the all operating space.

The control problem considered in this article is the asymptotic tracking of a reference trajectory y_{id} for the quadrotor outputs y_i . The reference paths and their derivatives are assumed to be smooth, continuous, and bounded. In the control literature of nonlinear systems,³² this assumption is frequently adopted to resolve the trajectory tracking problem, especially the control algorithms incorporate successive differentiation of the reference signal, and it is an essential condition to ensure the boundedness of all closed-loop system signals. In our case, this condition is fulfilled by an introduction of a filter differentiator in the control scheme to deal with the computational explosion problem.

For the controller design, the state-space equations of the quadrotor will be rewritten so that the nonlinear functions and the disturbances of each subsystem are lumped in one nonlinear function. For that, we have multiplied the above dynamic model (8) by $g_i(\mathcal{X})^{-1}$ which yields

$$0 = -g_i(\mathcal{X})^{-1} \dot{x}_{i2} + g_i(\mathcal{X})^{-1} (f_i(\mathcal{X}) + d_i(t)) + u_i \quad (10)$$

By adding \dot{x}_{i2} to both sides of equation (10), we can write the model of each subsystem as follows

$$\begin{cases} \dot{x}_{i1} = x_{i2} \\ \dot{x}_{i2} = \Psi_i(\mathcal{X}, t) + u_i \end{cases} \quad (11)$$

where $\Psi_i(\mathcal{X}, t)$ denote the total nonlinear dynamic functions with disturbances and their expressions are given by

$$\Psi_i(\mathcal{X}, t) = (1 - g_i(\mathcal{X})^{-1}) \dot{x}_{i2} + g_i(\mathcal{X})^{-1} (f_i(\mathcal{X}) + d_i(t)) \quad (12)$$

The aim of the next section is to design a model-free control without the requirement of a prior knowledge of the total nonlinear function (12). The proposed controller is able to achieve a null steady-state tracking error, to handle the unknown disturbances, and to perform a faster convergence rate to determine its parameters through a metaheuristic algorithm.

Flight controller design

To avoid the complexity related to obtaining an accurate nonlinear model of the quadrotor UAV system, a model-free fuzzy control (MFFC) is proposed to deal with the trajectory tracking problem. For the quadrotor UAV-drone dynamics formulated in equation (8), the control design process is carried out in three phases. First, a PID controller is used to impose a closed-loop dynamics combined with an adaptive estimator used to approximate the unknown nonlinear dynamics and disturbances of the quadrotor UAV. Then, to compensate the estimation error, a fuzzy logic system is introduced. Finally, the designed control law is applied to the drone system through two cascade control loops, the first loop is related to the position control, while the second loop is devoted to the attitude stabilization. The block diagram of the overall control system is shown in Figure 2.

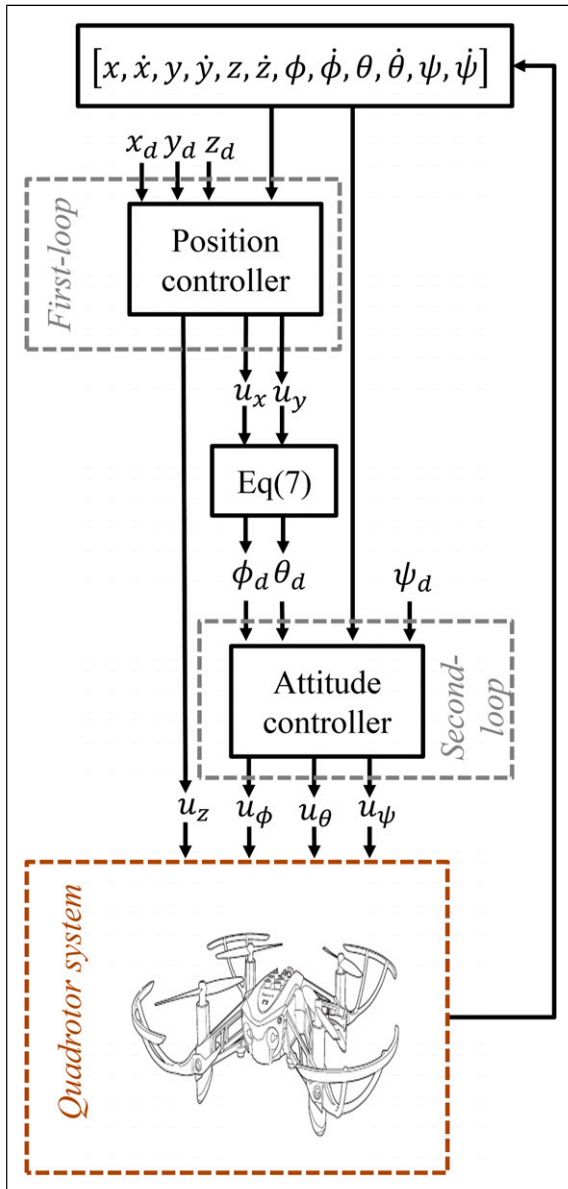


Figure 2. Quadrotor control structure via two cascaded loops.

Adaptive estimator-based model-free controller

The MFC approach has many advantages which makes it approved in many real applications like UAVs control.^{21–23} The main advantage of this control strategy is that the control law depends only on the real-time measurement data of the controlled plant, without needing an accurate dynamic model. In this study, the proposed MFC control law is designed for quadrotor subsystems considering that the total dynamic functions with external disturbances $\Psi_i(\mathcal{X}, t)$ are unknown. First, the mathematical model of the quadrotor drone formulated in equation (11) is replaced by the following simple representation

$$\ddot{y}_i = \Psi_i(\mathcal{X}, t) + u_i \tag{13}$$

In which the unknown nonlinear functions $\Psi_i(\mathcal{X}, t)$ should be estimated. Then, let us define $e_i(t)$ and $\dot{e}_i(t)$, the tracking errors and their derivatives, respectively, as follows

$$\begin{aligned} e_i(t) &= x_{i1}^d - x_i \\ \dot{e}_i(t) &= \dot{x}_{i2}^d - \dot{x}_i \end{aligned} \tag{14}$$

where $x_{i1}^d = y_{id}$, x_{i2}^d and \dot{x}_{i2}^d are the first and second derivative of y_{id} , respectively. Their values are estimated by the homogeneous finite time differentiator, as in Ref. 33, which is defined by

$$\begin{aligned} \dot{\gamma}_1(t) &= -a_1[\gamma_1(t) - y_{id}]^{\beta_1} + \gamma_2(t) \\ \dot{\gamma}_2(t) &= -a_2[\gamma_1(t) - y_{id}]^{\beta_2} + \gamma_3(t) \\ \dot{\gamma}_3(t) &= -a_3[\gamma_1(t) - y_{id}]^{\beta_3} \end{aligned} \tag{15}$$

where $\gamma \in \mathbb{R}^3$ is the differentiator state with $\beta_i = 1 - i\tau$ for $i = 1, 2, 3$ and any $\tau \in (-(1/3), 0)$, while the parameters a_1, a_2, a_3 are selected to ensure the Hurwitz property of the matrix

$$A = \begin{bmatrix} -a_1 & 1 & 0 \\ -a_2 & 0 & 1 \\ -a_3 & 0 & 0 \end{bmatrix}$$

Then the system (15) performs an estimation of the first and second derivative of y_{id} in a finite time: $x_{i2}^d = \gamma_2(t), \dot{x}_{i2}^d = \gamma_3(t)$. Increasing the smallest eigenvalue of A improves the rate of convergence. In our work, $\tau < 0$ is sufficiently big, and the eigenvalues of A are chosen by trial and error accordingly to the desired dynamics and in such a way to avoid undesired response delay. It should be mentioned that the used differentiator requires only the output measurement and its convergence is independent of the task of the closed-loop controller subsequently developed.

Taking the quadrotor control in formula (13), the model-free control law¹⁹ is designed as follows

$$\begin{aligned} u_i(t) &= u_{ip}(t) - \hat{\Psi}_i(\mathcal{X}, t) \\ u_{ip}(t) &= K_{iP}e_i(t) + K_{iD}\dot{e}_i(t) + K_{iI} \int e_i(t) + \dot{x}_{i2}^d \end{aligned} \tag{16}$$

where K_{iP}, K_{iI} , and K_{iD} for $i \in \{x, y, z, \phi, \theta, \psi\}$ are the classical proportional–integral–derivative gains and

$\hat{\Psi}_i(\mathcal{X}, t)$ are the estimated values of $\Psi_i(\mathcal{X}, t)$. Substituting (16) into (13) yields

$$\ddot{e}_i(t) + K_{iP}e_i(t) + K_{iD}\dot{e}_i(t) + K_{iI} \int e_i(t) = -\tilde{\Psi}_i(\mathcal{X}, t) \quad (17)$$

with $\tilde{\Psi}_i(\mathcal{X}, t) = \Psi_i(\mathcal{X}, t) - \hat{\Psi}_i(\mathcal{X}, t)$ are the estimation errors. From equation (17), it is clear that the steady error dynamics of the closed-loop system is related to the design parameters K_{iP} , K_{iI} , and K_{iD} . Their values are generally chosen by the user in the way that the closed-loop stability is guaranteed and the tracking errors can converge to zero. Hence, when $\tilde{\Psi}_i(\mathcal{X}, t) \rightarrow 0$ for $t \rightarrow \infty$, the errors dynamics of quadrotor is related to the following equation

$$\ddot{e}_i(t) + K_{iP}e_i(t) + K_{iD}\dot{e}_i(t) + K_{iI} \int e_i(t) = 0 \quad (18)$$

where the stability analysis of the control system can be absolutely proved similarly as in Refs. 19, 34.

As an estimation law for $\hat{\Psi}_i(t)$, we introduce the following first-order differential equation²⁰

$$\dot{\hat{\Psi}}_i(t) = -\frac{1}{\alpha_i}(\hat{\Psi}_i(t) - \Psi_i(t)) = \frac{1}{\alpha_i}\tilde{\Psi}_i(t) \quad (19)$$

where $\alpha_i > 0$ are designed parameters to be chosen. We can define equation (19) in the Laplace form as

$$\hat{\Psi}_i(s) = \frac{1}{1 + \alpha_i s} \Psi_i(s) \quad (20)$$

where s represents the Laplace variable. It can be noticed from equation (20) that $\hat{\Psi}_i$ is obtained from filtering the unknown term Ψ_i via the filter $1/(1 + \alpha_i s)$ and we also have $\hat{\Psi}_i(t)$ that converges toward $\Psi_i(t)$ when the parameter α_i tends toward 0.

Let us use equation (13) to rewrite equation (20), we can get

$$\hat{\Psi}_i(s) = \frac{1}{1 + \alpha_i s} (s^2 \mathbf{y}_i(s) - \mathbf{u}_i(s)) \quad (21)$$

where $\mathbf{y}_i(s)$ be the Laplace transform of the variable state $\mathbf{y}_i(t)$. Substituting equation (21) into (16), we obtain

$$\mathbf{u}_i(s) = \mathbf{u}_{ip}(s) - \frac{1}{1 + \alpha_i s} (s^2 \mathbf{y}_i(s) - \mathbf{u}_i(s)) \quad (22)$$

then

$$\mathbf{u}_i(s) = \left(1 + \frac{1}{\alpha_i s}\right) \mathbf{u}_{ip}(s) - \frac{s}{\alpha_i} \mathbf{y}_i(s) \quad (23)$$

With some mathematical manipulations, the control law can be further rewritten as follows

$$u_i(t) = u_{ip}(t) + \frac{1}{\alpha_i} \int u_{ip}(t) - \frac{1}{\alpha_i} \dot{y}_i(t) \quad (24)$$

With an adequate choice of K_{iP} , K_{iI} , K_{iD} , and α_i for $i \in \{x, y, z, \phi, \theta, \psi\}$, the tracking of the desired trajectories for the nonlinear system (11) will be achieved.

Model-free fuzzy control

In the previous subsection, the proposed MFC law based on the linear PID controller is obtained with unrealistic assumption that the quadrotor dynamic and disturbances $\Psi_i(x, t)$ are ideally approximated by adaptive estimators and that the existing estimation errors in equation (17) are neglected. In addition, the PID controller can improve alone the precision in trajectory tracking. To overcome these problems, a robust fuzzy logic compensator is added as an extra input to the MFC law (24) in which the corresponding model-free fuzzy control (MFFC) law is finally expressed as

$$u_i(t) = u_{ip}(t) + \frac{1}{\alpha_i} \int u_{ip}(t) - \frac{1}{\alpha_i} \dot{y}_i(t) + u_{if}(t) \quad (25)$$

where u_{if} are the fuzzy logic control terms.

This approach is easy to implement and has the ability of generalization according to the theory of the universal approximation.³⁵ For designing the add-term u_{if} , let e_i and \dot{e}_i be the linguistic inputs variable of the fuzzy control block of each subsystem. To simplify the used fuzzy system structure, the universe of the fuzzy variables discourse is divided into three fuzzy sets given by N , Z , and P (i.e., the labels N , P , and Z denote ‘‘Negative,’’ ‘‘Positive,’’ and ‘‘Zero,’’ respectively), and the fuzzy rules are given as

$$R_k : \text{IF } e_i \text{ is } A_n \text{ and } \dot{e}_i \text{ is } B_m \text{ THEN } u_{if} \text{ is } C_{n+m} \quad (26)$$

where $-1 \leq n, m \leq 1$, and $k = 1, \dots, 9$ is the rules number. We suggest to use triangular membership functions to represent the linguistic variables as shown in Figure 3. The defuzzification process is based on the center of gravity method³⁶ leading to express the output by the following relationship

$$u_{if}(t) = \frac{\sum_{m=-1}^{1,1} \mu_n(e_i) \mu_m(\dot{e}_i) C_{n+m}}{\sum_{m=-1}^{1,1} \mu_n(e_i) \mu_m(\dot{e}_i)} = \zeta_i^T(e_i) r_i \quad (27)$$

where $0 < \mu_{n,m} < 1$ is the firing strengths of rules R_k for $k = 1, \dots, 9$ which satisfies the condition $\sum_{m=-1}^{1,1} \mu_n(e_i) \mu_m(\dot{e}_i) = 1$; $c_1 = -c_3$ and $c_2 = 0$ are the center of the membership functions of the output u_{if} ; $r_i = [r_i^1, \dots, r_i^{n,m}]^T$ are the vectors of the conclusion values in the fuzzy rules;

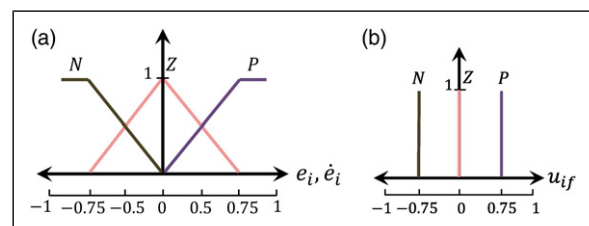


Figure 3. (a) Membership functions of input variables e_i and \dot{e}_i ; (b) membership functions of output u_{if} .

and $\zeta_i = [\zeta_i^1, \dots, \zeta_i^{n,m}]^T$ are the vectors of the height of the membership functions of u_{if} with $\zeta_i^{n,m}(e_i) = \mu_{n,m} / \sum_{m=-1}^{1,1} \mu_n(e_i) \mu_m(\dot{e}_i)$.³⁷

Substituting the new control law (25), the closed-loop dynamics (17) becomes

$$\ddot{e}_i(t) + K_{ip}e_i(t) + K_{id}\dot{e}_i(t) + K_{il} \int e_i(t) = -\tilde{\Psi}_i(t) + u_{if} \quad (28)$$

By selecting $E_i = [e_i, \dot{e}_i]^T$, we can write (28) in state equation as follows

$$\dot{E}_i(t) = \Lambda_i E_i + \Gamma_i (u_{if} - \tilde{\Psi}_{im}) \quad (29)$$

with

$$\Lambda_i = \begin{bmatrix} 0 & 1 \\ -k_{ip} & -k_{id} \end{bmatrix},$$

$$\Gamma_i = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \text{and} \quad \tilde{\Psi}_{im} = k_{il} \int e_i + \tilde{\Psi}_i$$

The objective is to suppress the effect of $\tilde{\Psi}_{im}$ by applying fuzzy control terms u_{if} to improve the accuracy and robustness of the quadrotor control system. Using equation (27), ideal approximations of the error functions $\tilde{\Psi}_{im}$ are introduced over a compact set $U_{E_i} = \text{argmin} \{ \sup_{E_i \in U_{E_i}} |\tilde{\Psi}_{im}(t) - \tilde{\Psi}_{im}(e_i, r_i^*)| \}$ as

$$\tilde{\Psi}_{im}(t) = \tilde{\Psi}_{im}(e_i, r_i^*) = \zeta_i^T(e_i) r_i^* + \Delta_i(t) \quad (30)$$

where r_i^* are unknown optimal parameter vectors minimizing the approximation errors $\Delta_i(t)$. According to the universal approximation property, $\Delta_i(t)$ are bounded for all $E_i \in U_{E_i}$ as

$$|\Delta_i(t)| < \bar{\Delta}_i \quad (31)$$

where $\bar{\Delta}_i$ are unknown positive constants. To compensate the effect of estimation errors, that is, to let $\lim_{t \rightarrow \infty} e_i(t) = 0$, the following equation should be satisfied

$$u_{if} = \tilde{\Psi}_{im}(e_i, r_i) = \zeta_i^T(e_i) r_i \quad (32)$$

where r_i are the estimates of r_i^* . By substituting (30) and (32) in (29), we can write

$$\dot{E}_i(t) = \Lambda_i E_i - \Gamma_i (\zeta_i^T(e_i) (r_i^* - r_i) + \Delta_i) \quad (33)$$

Based on the above analysis, the following theorem is given to explain the MFFC performance of the overall closed-loop quadrotor system.

Theorem. Considering the control problem of the quadrotor system (13), the proposed model-free fuzzy controller (MFFC) (25) with the estimators (19) and the fuzzy control terms (32) ensures the stability of the overall closed-loop system and the asymptotic convergence of the tracking errors toward zero, that is, $\lim_{t \rightarrow \infty} e_i(t) = 0$ if the following adaptation law holds

$$r_i = -\lambda_i \zeta_i(e_i) E_i^T P_i \Gamma_i \quad (34)$$

where λ_i are positive constants and P_i are positive definite matrices which satisfy

$$\Lambda_i^T P_i + P_i \Lambda_i = -Q_i \quad (35)$$

with Q_i are diagonal positive definite matrices.

Proof. Consider the following Lyapunov function for all quadrotor systems³⁸

$$V = \sum_{i \in \{x, y, z, \phi, \theta, \psi\}} V_i = \sum_{i \in \{x, y, z, \phi, \theta, \psi\}} \frac{1}{2} E_i^T P_i E_i + \frac{1}{2\lambda_i} \tilde{r}_i^T \tilde{r}_i \quad (36)$$

where $\tilde{r}_i = r_i^* - r_i$. Taking the derivative of V , it then comes to

$$\dot{V} = \sum_{i \in \{x, y, z, \phi, \theta, \psi\}} \dot{V}_i$$

$$= \frac{1}{2} \dot{E}_i^T P_i E_i + \frac{1}{2} E_i^T P_i \dot{E}_i - \frac{1}{\lambda_i} \tilde{r}_i^T \dot{r}_i \quad (37)$$

Using equation (33), one has

$$\dot{V}_i = \frac{1}{2} \dot{E}_i^T P_i E_i + \frac{1}{2} E_i^T P_i \dot{E}_i - \frac{1}{\lambda_i} \tilde{r}_i^T \dot{r}_i$$

$$= \frac{1}{2} (\Lambda_i E_i - \Gamma_i (\zeta_i^T(e_i) \tilde{r}_i + \Delta_i))^T P_i E_i$$

$$+ \frac{1}{2} E_i^T P_i (\Lambda_i E_i - \Gamma_i (\zeta_i^T(e_i) \tilde{r}_i + \Delta_i)) - \frac{1}{\lambda_i} \tilde{r}_i^T \dot{r}_i \quad (38)$$

Considering (35), equation (38) becomes

$$= \frac{1}{2} E_i^T (\Lambda_i^T P_i + P_i \Lambda_i) E_i - E_i^T P_i \Gamma_i \Delta_i$$

$$- \tilde{r}_i^T \zeta_i(e_i) E_i^T P_i \Gamma_i - \frac{1}{\lambda_i} \tilde{r}_i^T \dot{r}_i$$

$$= -\frac{1}{2} E_i^T Q_i E_i - E_i^T P_i \Gamma_i \Delta_i - \tilde{r}_i^T [\dot{r}_i + \lambda_i \zeta_i(e_i) E_i^T P_i \Gamma_i] \quad (39)$$

Combining the update law equation (34) into (39)

$$\dot{V} = \sum_{i \in \{x, y, z, \phi, \theta, \psi\}} -\frac{1}{2} E_i^T Q_i E_i - E_i^T P_i \Gamma_i \Delta_i$$

$$\leq -\frac{1}{2} E_i^T Q_i E_i + |E_i^T P_i \Gamma_i \Delta_i| \leq -\frac{1}{2} E_i^T Q_i E_i + |E_i^T| P_i \bar{\Delta}_i \quad (40)$$

In the equation, $-(1/2) E_i^T Q_i E_i < 0$, therefore, the approximation errors $\bar{\Delta}_i$ can be sufficiently small by fuzzy systems which make $\dot{V}_i < 0$ and $V \in L_\infty$. This implies that the stability of the MFFC closed-loop system of the quadrotor is guaranteed and the signals $e_i, \dot{e}_i, \tilde{\Psi}_i$, and \tilde{r}_i are bounded. By using Barbalat's lemma,³² the errors e_i and \dot{e}_i converge asymptotically to zero. Thus, the theorem is proved.

Remark 2. The quadrotor system formulated in equation (13) is stabilized by using the MFCC law designed in equation (25) which is only linked to the tracking errors e_i and \dot{e}_i where no prior knowledge of the dynamic model is needed. The convergence of the outputs y_i , position, and attitude of the quadrotor to the reference trajectories y_{id} is achieved by using the Lyapunov method without any restriction on system dynamics and controller parameters. Only, PID gains K_{iP} , K_{iD} , K_{iI} , and α_i must be positive and competently adjusted. In our control strategy, these parameters are computed automatically via a metaheuristic optimization algorithm to have better control performance for the quadrotor system.

Model-free fuzzy control for position and attitude tracking

The MFCC control laws applied to the quadrotor are formulated according to equation (25) for position and attitude subsystems as it will be detailed in the following subsections.

Position control design. In order to track the desired path in 3D motion, a model-free fuzzy control for position trajectory tracking control is implemented. The dynamic model of the altitude z subsystem according to (11) is defined

$$\begin{cases} \dot{x}_{z1} = x_{z2} \\ \dot{x}_{z2} = \Psi_z(\mathcal{X}, t) + u_z \end{cases} \quad (41)$$

where u_z is the control input signal of the altitude subsystem defined as

$$u_z(t) = u_{zp}(t) + \frac{1}{\alpha_z} \int u_{zp}(t) - \frac{1}{\alpha_z} \dot{y}_z(t) + u_{zf}(t) \quad (42)$$

In order that the quadrotor tracks its desired path in x and y planes, virtual control inputs u_x and u_y of Cartesian position subsystems are computed as

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = \begin{bmatrix} u_{xp}(t) + \frac{1}{\alpha_x} \int u_{xp}(t) - \frac{1}{\alpha_x} \dot{y}_x(t) + u_{xf}(t) \\ u_{yp}(t) + \frac{1}{\alpha_y} \int u_{yp}(t) - \frac{1}{\alpha_y} \dot{y}_y(t) + u_{yf}(t) \end{bmatrix} \quad (43)$$

Attitude control design. The attitude quadrotor is composed of three rotational motions that allow to be transmitted from the normal mode to the inverted mode or vice versa according to the roll, pitch, and yaw subsystems. The roll subsystem is

$$\begin{cases} \dot{x}_{\phi 1} = x_{\phi 2} \\ \dot{x}_{\phi 2} = \Psi_\phi(\mathcal{X}, t) + u_\phi \end{cases} \quad (44)$$

where u_ϕ is the control input signal of the roll subsystem defined as

$$u_\phi(t) = u_{\phi p}(t) + \frac{1}{\alpha_\phi} \int u_{\phi p}(t) - \frac{1}{\alpha_\phi} \dot{y}_\phi(t) + u_{\phi f}(t) \quad (45)$$

The pitch subsystem is given by

$$\begin{cases} \dot{x}_{\theta 1} = x_{\theta 2} \\ \dot{x}_{\theta 2} = \Psi_\theta(\mathcal{X}, t) + u_\theta \end{cases} \quad (46)$$

where u_θ is the control input signal of the roll subsystem defined as

$$u_\theta(t) = u_{\theta p}(t) + \frac{1}{\alpha_\theta} \int u_{\theta p}(t) - \frac{1}{\alpha_\theta} \dot{y}_\theta(t) + u_{\theta f}(t) \quad (47)$$

The yaw subsystem is given by

$$\begin{cases} \dot{x}_{\psi 1} = x_{\psi 2} \\ \dot{x}_{\psi 2} = \Psi_\psi(\mathcal{X}, t) + u_\psi \end{cases} \quad (48)$$

where u_ψ is the control input signal of the roll subsystem defined as

$$u_\psi(t) = u_{\psi p}(t) + \frac{1}{\alpha_\psi} \int u_{\psi p}(t) - \frac{1}{\alpha_\psi} \dot{y}_\psi(t) + u_{\psi f}(t) \quad (49)$$

Remark 3. From equations (42), (43), (45), (47), and (49), it is obvious that the quadrotor inputs are totally independent since the control values of each subsystem are only calculated through its local measurements, that is, u_i for $i \in \{x, y, z, \phi, \theta, \psi\}$ are related only to the local tracking errors e_i and \dot{e}_i . Thus, no interaction exists between the control of position and attitude subsystems which lead in effect to have a decentralized (decoupled) control structure. Besides, the proposed control scheme is simple and model-free which makes it easy to be implemented in real time. The incorporation of adaptive estimators (20) enables to have a robust control law against unknown dynamics and external disturbances, and the fuzzy logic control components are introduced just to deal with estimation errors. This proposed strategy can guarantee an optimal control behavior against minimizing an objective function which will be the subject of the next section.

Optimal model-free fuzzy logic controller

This section presents an optimal model-free fuzzy control (OMFFC) design for the quadrotor system. In order to deal with the control parameter selection issue, the bat algorithm was included to find the optimal control parameters of the quadrotor controller. This proposed strategy can guarantee optimal control behavior against minimizing an objective function. The block diagram of the overall control system is shown in Figure 4.

Bat algorithm description

The bat algorithm (BA) is a metaheuristic algorithm that was proposed by Yang,³⁹ which is inspired from the behavior of bats while searching food. Its main concept is formulated in the three following rules:

- Each bat (solution) uses its echolocation property to estimate distance to a prey and can figure out the difference between prey and surroundings;
- Bats fly randomly with velocity v at position p and a fixed range of frequency $Q \in [Q_{\min}, Q_{\max}]$ with continuously different loudness ϑ and pulse emission rate $r \in [0, 1]$ for looking of prey;
- The loudness assumed to be varying from big positive value to a small predefined value (i.e., $\vartheta \in [\vartheta_{\min}, \vartheta_0]$).

Based on these approximations, the main rules of the BA can be summarized in pseudo-code shown in Algorithm 1. The positions p_j and velocities v_i of the virtual bats have to be defined in a d-dimensional search space that is updated for each generation j . The new solutions x_{kj} and velocities v_{kj} at time step k are generated via a frequency Q_j as

$$Q_j = Q_{\min} + (Q_{\max} - Q_{\min}) \times rand(0, 1) \quad (50)$$

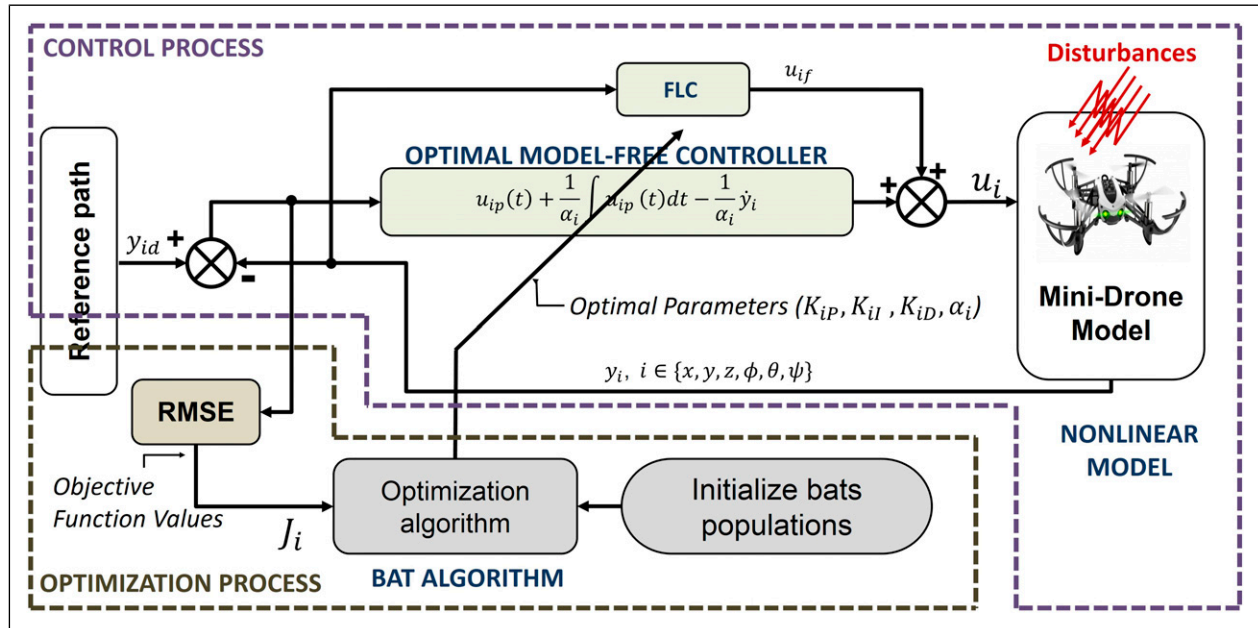


Figure 4. Optimal model-free fuzzy control structure.

Algorithm 1. Pseudo-code of bat algorithm (BA)

Cost function Min J
Initialize bat population: locations p_k and velocities v_k with random solution
Initialize frequencies Q_k , pulse rates r_k , and the loudness ϑ_k
Find the best solution p^{best} in the initial population
while $J >$ desired tolerance **do**
 Generate new solutions by adjusting frequency $\rightarrow Q_j$ equation (50), and update velocities $\rightarrow v_j^{new}$ equation (51), and position/solution p_j^{new} equation (52)
 if $rand > r_k$ **then**
 Generate a local solution around the selected best solution
 Select a solution p_j^{new} among the best solutions
 end if
 if $(J(p_j^{new}) < J(p_{kj}))$ **then**
 Accept the new solutions
 Increase r_k and reduce ϑ_k
 end if
 Rank the bats and find the current best p^{bes}
end while
Output the best solution found p^{best}

$$v_j^{new} = v_{kj} + (p_{kj} - p_j^{best}) \times Q_k \quad (51)$$

$$p_j^{new} = p_{kj} + v_j^{new} \quad (52)$$

Then the position p_j^{new} is replaced by the solution generated through random walk as⁴⁰

$$p_j^{new} = p_j^{best} + rand(-1, 1) \times \widehat{\vartheta}_k \quad (53)$$

while $\widehat{\vartheta}_k$ is the average loudness of bats, where the new bat solution p_j^{new} is generated as

$$p_j^{new} = \begin{cases} p_j^{new} & \text{if } rand(0, 1) > \widehat{r}_k \\ p_{kj} + v_j^{new} & \text{otherwise} \end{cases} \quad (54)$$

The evolution rules for loudness and pulse rate are as follows

$$\widehat{r}_k = r_k^0 (1 - \exp(-\gamma t)) \quad (55)$$

$$\vartheta_k = \sigma \vartheta_k \quad (56)$$

where γ and σ are constants. For any $0 < \gamma < 1$ and $\sigma > 0$, the rate of pulse emission decreases and loudness as $\vartheta_k(t) \rightarrow 0, r_k(t) \rightarrow r_k^0$, as $t \rightarrow \infty$.

The best solution p_j^{best} is selected based on the objective function values J

$$p_j^{best} = \begin{cases} p_j^{new} & \text{if } J(p_j^{new}) < J(p_{kj}) \\ p_{kj} & \text{otherwise} \end{cases} \quad (57)$$

Remark 4. The BA has been included to MFFC to achieve a better performance for the quadrotor against minimizing the RMSE objective function.

BA-based model-free fuzzy controller

In this part, we propose the BA to tune the parameters of the designed controller (25) against minimizing an

objective function J_i for $i \in \{x, y, z, \phi, \theta, \psi\}$ in order to converge the error dynamics (28) of the closed loop to zero. The step by step of the BA process applied on the quadrotor control is given below.

Step 1. Initialize the maximum number of iterations max_j , the frequencies Q_k , pulse rates r_k , and the loudness ϑ_k .

Step 2. Initialize randomly the bat population vector as many N as follows

$$p_{kj} = LB_j + (UB_j - LB_j) \times rand(0, 1) \quad (58)$$

where LB_j and UB_j are upper and lower bounds of the bat search space, respectively, and then select the best solution p^{best} according to the objective function J_i .

Remark 5. In the literature of performance evaluation, various metrics can be found like the root mean square error (RMSE), the mean absolute error (meanAE), the maximum absolute error (maxAE), and integral square error (ISE).^{24,27,31} Both the RMSE and the meanAE are regularly employed in model evaluation studies and they are often suggested as a good indicator of performance.

In this study, we propose to use the RMSE metric, where the error is computed between the actual and the desired path for N sampling time size as

$$J_i = \sqrt{\frac{\sum_{l=1}^N (x_{il}^d - x_{il})^2}{N}} \quad (59)$$

It can be seen from equation (3) that all the position subsystems and the rotation subsystems have the same mathematical structure. So, we propose to consider two global objective functions during the optimization process as

$$J_p = \frac{\sum_{i \in \{x, y, z\}} J_i}{3} \quad \text{and} \quad J_R = \frac{\sum_{i \in \{\phi, \theta, \psi\}} J_i}{3}$$

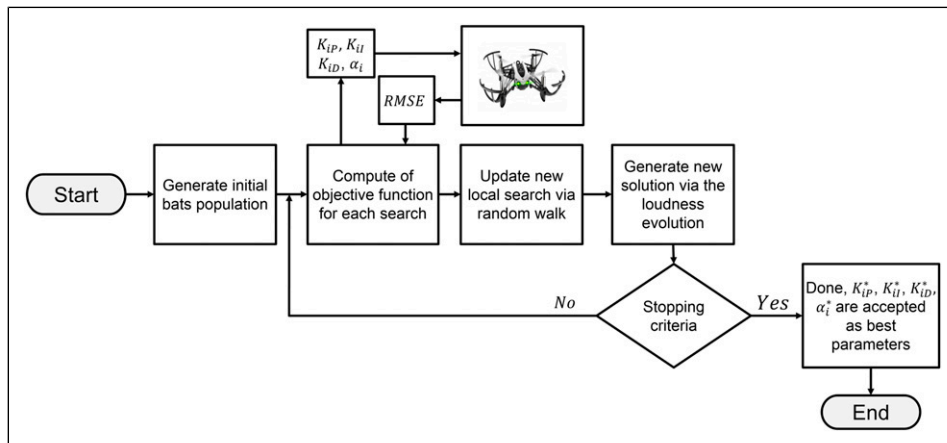


Figure 5. Flowchart of bat algorithm.

Step 3. Use a local search (53) to define a new solution of the controller parameters; else, shuffle the solutions actual via equation (54).

Step 4. Select the best controller parameters $p^{best} \in \{K_{iP}^*, K_{iI}^*, K_{iD}^*, \alpha_i^*\}$ via equation (57). Repeat all the steps until the stopping criterion is satisfied.

Based on these approximation steps, the BA process applied for the quadrotor control can be summarized in Figure 5.

Numerical simulation results

The effectiveness and the performances of the proposed OMFFC were tested using a full template of Parrot Drone support from MATLAB.⁴¹ This template considers all modules of the real Mambo Parrot mini-drone including trajectory generation, sensor fusion, and a nonlinear model of the Parrot drone. The physical parameters of this type of quadrotor as given by the manufacturer are listed in Table 1. In order to highlight the efficiency of the proposed flight controller, different cases were tested for the quadrotor control. In the first case, there was a comparison study between the optimal model-free fuzzy controller (OMFFC), the model-free fuzzy controller (MFFC), and PID controller without disturbances. In the second case, the same controllers were tested for another reference trajectory in the presence of step external disturbances. In the third and final case, the simulation test is made for aggressive trajectory with disturbances varying in frequency and amplitude. The objective is to show the robustness of the proposed controller against the nonlinearities of system and the external disturbances effect. According to the control algorithm, initially, the parameters K_{iP} , K_{iD} , K_{iI} , and α_i have been optimized under the RMSE function J_i for $i \in \{x, y, z, \phi, \theta, \psi\}$ in bounded search space $p_i^{best} \in [p_i^{min}, p_i^{max}] = [0.001, 5]$. Figure 6 shows the evaluation of the best objective function values on the iterations for the position and rotation subsystems, showing that the convergence of the best objective function to constant values is always achieved. The parameters of the proposed controllers are given in Table 2.

Table 1. Parameters of the quadrotor.

Parameters	Value
I_x	$0.068 \times 10^{-3} \text{ kgm}^2$
I_y	$0.92 \times 10^{-3} \text{ kgm}^2$
I_z	$0.1366 \times 10^{-2} \text{ kgm}^2$
J_r	$6 \times 10^{-5} \text{ kgm}^2$
g	9.81 m/s^2
l	0.065 m
m	0.068 kg

Case 1. Comparison study

In this case, the OMFFC is compared with MFFC and PID controllers. The parameters of the latter two controllers were selected by trial and error without any optimization tool. The desired trajectory of the quadrotor is $y_{vd} = 0^\circ$ and $y_{zd} = 1.5 \text{ m}$

$$y_{xd} = \begin{cases} 0 \text{ m}, & t < 10 \text{ s} \\ 2 \text{ m}, & 10 < t < 20 \text{ s} \\ 1 \text{ m}, & 20 < t < 25 \text{ s} \\ 0.5 \text{ m}, & 25 < t < 35 \text{ s} \\ 0 \text{ m}, & 35 \text{ s} < t \end{cases}, \quad y_{yd} = \begin{cases} 0 \text{ m}, & t < 15 \text{ s} \\ 2 \text{ m}, & 15 < t < 25 \text{ s} \\ 1 \text{ m}, & 25 \text{ s} < t \end{cases}$$

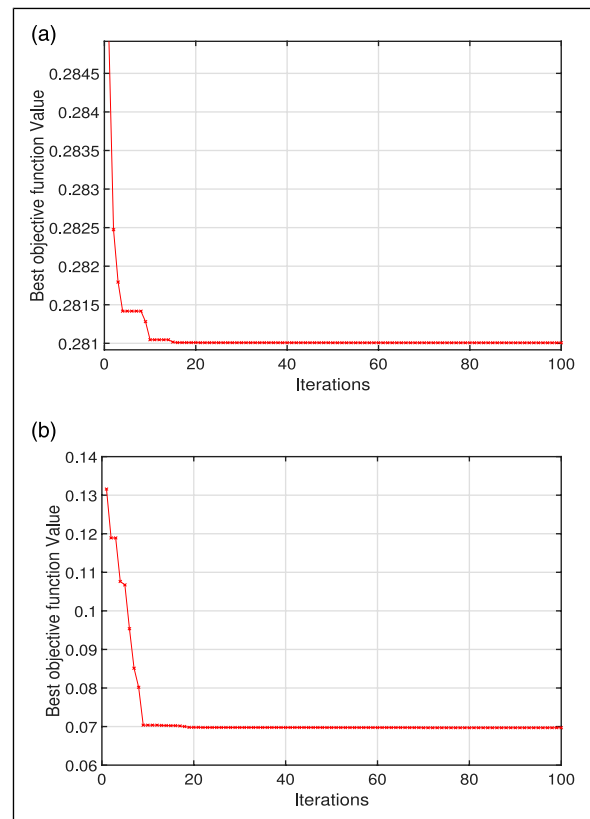


Figure 6. Best objective function versus iterations: (a) rotation system; (b) position system.

Table 2. Gains for the tested controllers.

Controllers	x, y	z	ϕ, θ	ψ
OMFFC	$\alpha = 0.254$	$\alpha = 0.185$	$\alpha = 0.094$	$\alpha = 0.085$
	$k_p = 0.312$	$k_p = 2.16$	$k_p = 0.152$	$k_p = 0.247$
	$k_i = 0.115$	$k_i = 1.013$	$k_i = 0.025$	$k_i = 0.035$
	$k_D = 0.097$	$k_D = 1.958$	$k_D = 0.13$	$k_D = 0.012$
MFFC	$\alpha = 0.3$	$\alpha = 0.1$	$\alpha = 0.1$	$\alpha = 1$
	$k_p = 1.5$	$k_p = 1.5$	$k_p = 0.1$	$k_p = 0.2$
	$k_i = 0.2$	$k_i = 1.02$	$k_i = 0.03$	$k_i = 0.02$
	$k_D = 1$	$k_D = 1$	$k_D = 0.1$	$k_D = 0.02$
PID	$k_p = 0.24$	$k_p = 1$	$k_p = 0.02$	$k_p = 0.04$
	$k_i = 0.1$	$k_i = 0.2$	$k_i = 0.01$	$k_i = 0.01$
	$k_D = 0.1$	$k_D = 1$	$k_D = 0.03$	$k_D = 0.02$

OMFFC: optimal model-free fuzzy control; MFFC: model-free fuzzy control; PID: proportional–integral–derivative.

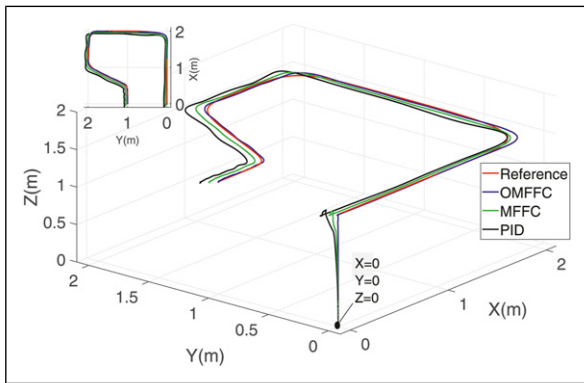


Figure 7. Quadrotor position response in 3D space (case 1).

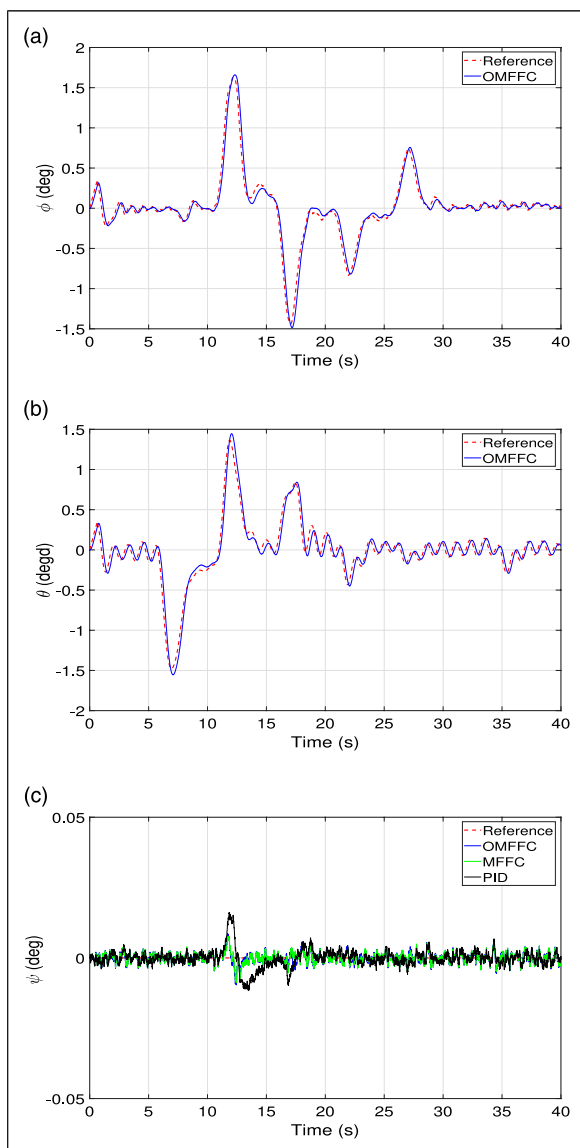


Figure 8. Quadrotor attitude tracking response (case 1). (a) Roll (ϕ) subsystem response; (b) Pitch (θ) subsystem response; (c) Yaw (ψ) subsystem response.

The initial position and rotation values of the quadrotor during this simulation case are chosen as $[\phi_0, \theta_0, \psi_0] = [0, 0, 0]^\circ$ and $[x_0, y_0, z_0] = [0, 0, 0]$ m.

The trajectory tracking performance in 3D space is shown in Figure 7, where the position of the quadrotor response is depicted. Figure 8 presents the attitude evolution of the quadrotor during the simulation. The transients of the control inputs are shown in Figure 9 which illustrates the smooth control signals without saturation. The proposed controller provides always better trajectory tracking compared with the others. The desired yaw trajectory is 0 deg as plotted in Figure 8(c). The changes of reference signals of the Cartesian positions x_d and y_d between 10 and 15s cause changes in orientation in the roll and pitch angles. This variation provides an effect on the yaw angle dynamics as we can see in Figure 8(c). This is due to the interconnection effects of the yaw dynamics (ψ) with the other UAV flight dynamics, especially with the orientation subsystems roll (ϕ) and pitch (θ).

Case 2. Trajectory tracking performances under constant external disturbances

In this case, the OMFFC is also compared with MFFC and PID controller in presence of step external disturbances defined as

$$\begin{cases} d_x(t) = 0.2N, t > 20s \\ d_y(t) = 0.2N, t > 25s \\ d_z(t) = 0.5N, t > 5s \end{cases}$$

The desired trajectory of the quadrotor is made as $y_{yd} = 0^\circ, y_{zd} = 1.5^\circ\text{m}, y_{xd} = 2(1 - \cos(0.2t))$, and $y_{yd} = \sin(0.2t)$. The initial position and rotation values of the quadrotor during this simulation case are the same of case 1. The simulation results for this case are shown through

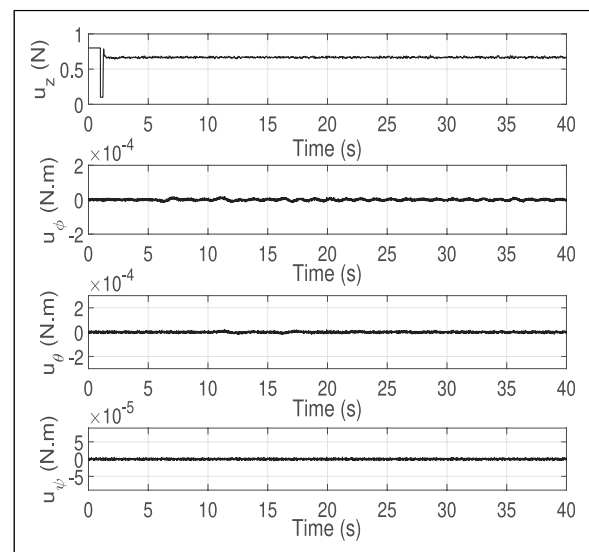


Figure 9. The control inputs of optimal model-free fuzzy control (case 1).

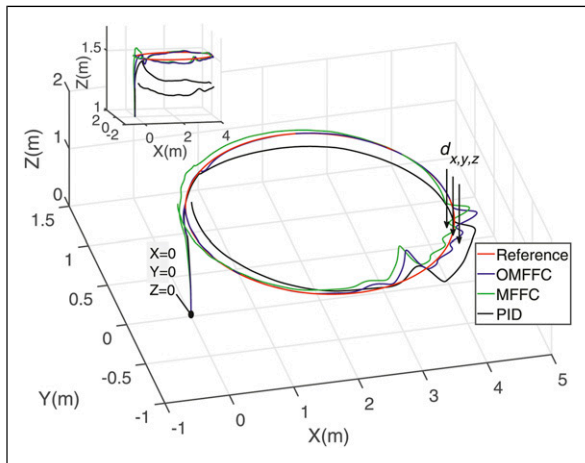


Figure 10. Quadrotor position response in 3D space (case 2).

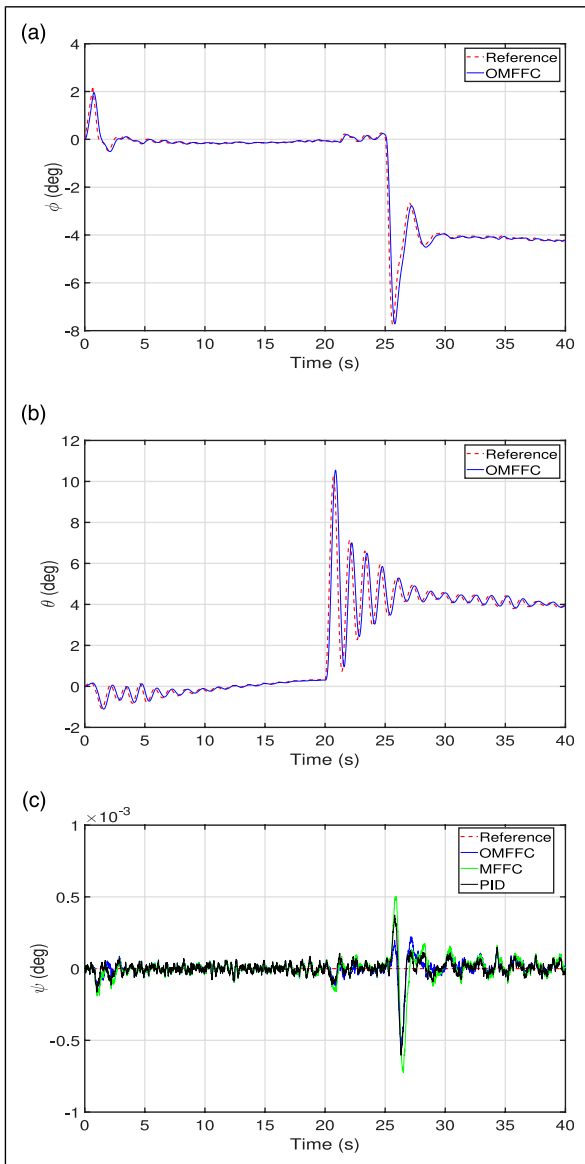


Figure 11. Quadrotor attitude tracking response (case 2). (a) Roll (ϕ) subsystem response; (b) Pitch (θ) subsystem response; (c) Yaw (ψ) subsystem response.

Figures 10–12. Figure 10 gives a comparison of the proposed controllers for the trajectory tracking in a 3D space under external disturbances. The transients of the attitude are shown in Figure 11. The pitch and roll angles present high pitches and some minor oscillations caused by the wind force effect. The stability is well preserved even in presence of external disturbance. The OMFFC can achieve better position trajectory tracking than the other controllers. Figure 12 illustrates the control signals for the OMFFC where we can notice that it follows the output evolution.

Case 3. Trajectory tracking performances under varying external disturbances

In this scenario, the vehicle is assumed to fly under varying external disturbances (as shown in Figure 13) simulating a real scenario where the wind speed is always unpredictable. The disturbance’s shape was selected to cover the most nonlinearities that can affect the quadrotor dynamics. The desired trajectory is defined as

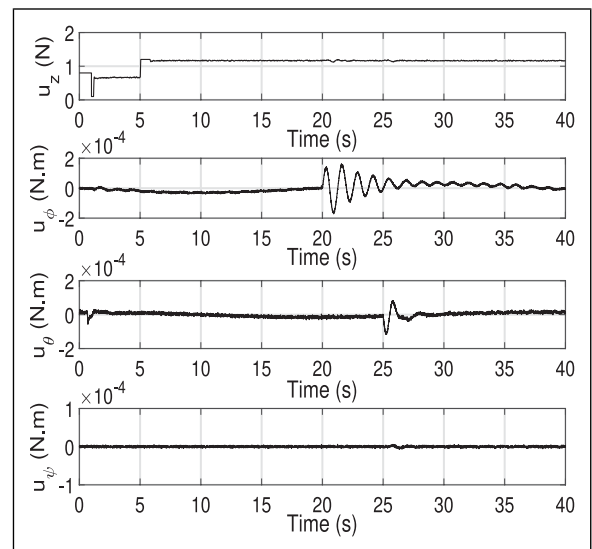


Figure 12. The control inputs of optimal model-free fuzzy control (case 2).

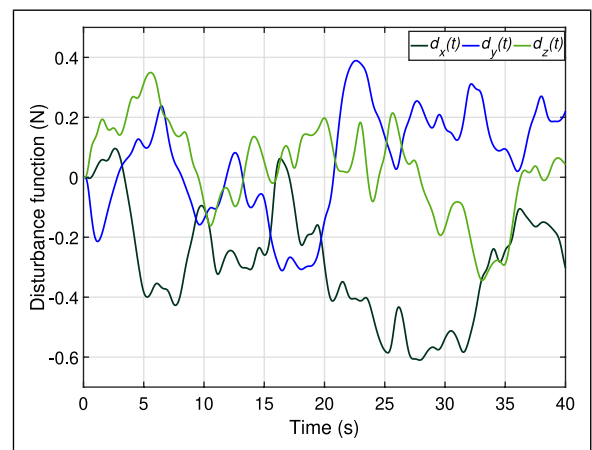


Figure 13. External disturbances function.

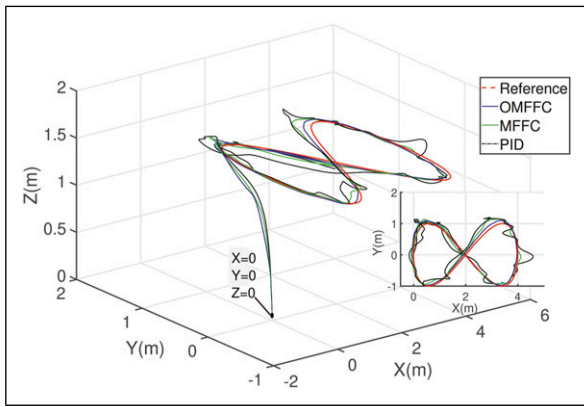


Figure 14. Quadrotor position response in 3D space (case 3).

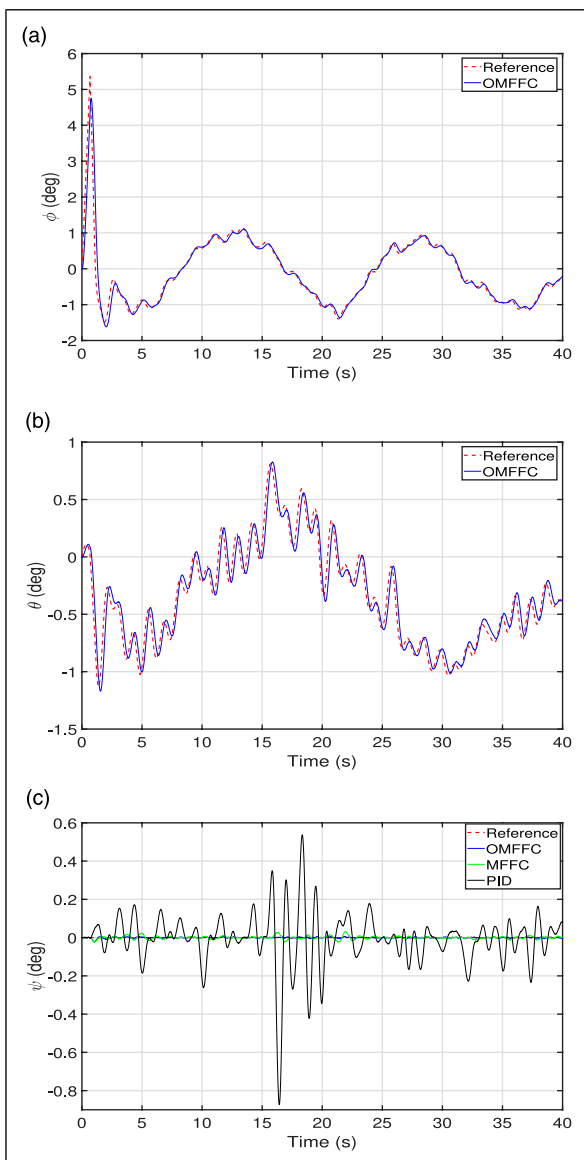


Figure 15. Quadrotor attitude tracking response (case 3). (a) Roll (ϕ) subsystem response; (b) Pitch (θ) subsystem response; (c) Yaw (ψ) subsystem response

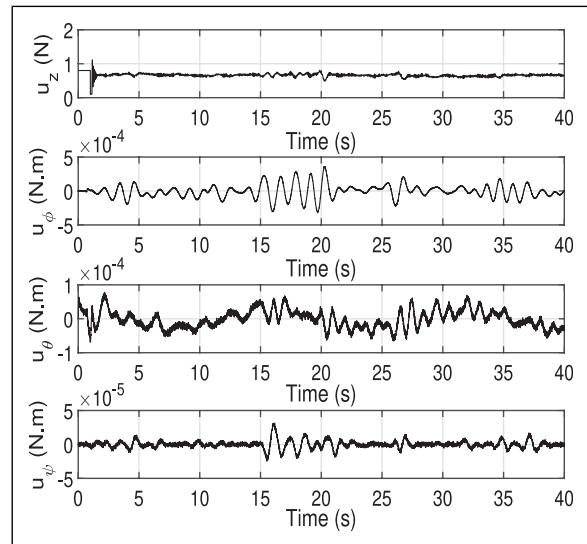


Figure 16. The control inputs of optimal model-free fuzzy control (case 3).

$y_{xd} = 2(1 - \cos(0.2t))m$, $y_{yd} = \sin(0.4t)m$, $y_{zd} = 1.5$ m, and $\psi_d = 0^\circ$. Under the same initial conditions of cases 1 and 2, the simulation results are shown in Figures 14–16. It can be seen from Figure 14 that the best trajectory tracking is achieved again by the OMFFC compared to the MFFC and PID controllers. We can note that the MFFC algorithm provides a better trajectory tracking than the PID controller. The transient of the attitude dynamics is shown in Figure 15 where we can observe some oscillations due to the wind force effect. The PID controller shows a bad tracking performance compared to the other controllers.

Summarizing the three cases, the control signals are smooth and remain in the eligible region enough to preserve the rotors in the Parrot drone. The proposed OMFFC ensures the robustness against external aggressive and unpredictable wind force conditions, and the stability is preserved under complex maneuvers.

Conclusion

In this article, we propose a new robust optimal model-free fuzzy control approach to investigate the tracking control problem of a quadrotor under external disturbances. This control approach is based on an adaptive estimator to approximate the lumped unknown dynamics and external disturbances of the quadrotor system. In order to deal with the estimation error, a fuzzy logic compensator was introduced. The optimal behavior of the designed control scheme is achieved by tuning automatically its parameters with a metaheuristic bat algorithm. The decentralized control allowed to manage independently the position and orientation dynamic subsystems which makes the control architecture more simple and easy to implement in real time. The new adaptive model-free fuzzy control structure

has been implemented in the Parrot Drone support from MATLAB/Simulink. Extensive simulation results obtained have strongly demonstrated the effectiveness of the proposed control approach. For future works, the proposed approach will be validated experimentally in real time using our “Parrot Mambo mini-drone” platform which is being instrumented with the various sensors.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was sponsored by the Hauts-de-France Polytechnic University, Hauts-de-France Region, the Ministry of Higher Education and Research, and the French National Center for Scientific Research. The authors acknowledge the support of the University of Biskra and the Algerian Ministry of Higher Education and Scientific Research.

ORCID iD

Hossam E Glida  <https://orcid.org/0000-0003-2973-2742>

References

- Bhardwaj A, Sam L, Akanksha FJ, et al. Uavs as remote sensing platform in glaciology: Present applications and future prospects. *Remote sensing Environ* 2016; 175: 196–204.
- Alzahrani B, Oubbati OS, Barnawi A, et al. Uav assistance paradigm: State-of-the-art in applications and challenges. *J Netw Computer Appl* 2020; 166: 102706.
- Hasseni S, Abdou L and Glida HE. Parameters tuning of a quadrotor pid controllers by using nature-inspired algorithms. *Evol Intelligence* 2021; 14: 61–73.
- Martins L, Cardeira C and Oliveira P. Linear quadratic regulator for trajectory tracking of a quadrotor. *IFAC-PapersOnLine* 2019; 52(12): 176–181.
- Mahmood A and Kim Y. Decentralized formation flight control of quadcopters using robust feedback linearization. *J Franklin Inst* 2017; 354(2): 852–871.
- Perozzi G, Efimov D, Biannic J-M, et al. Trajectory tracking for a quadrotor under wind perturbations: sliding mode control with state-dependent gains. *J Franklin Inst* 2018; 355(12): 4809–4838.
- Lin X, Yu Y and Sun C-y. A decoupling control for quadrotor uav using dynamic surface control and sliding mode disturbance observer. *Nonlinear Dyn* 2019; 97(1): 781–795.
- Vahdanipour M and Khodabandeh M. Adaptive fractional order sliding mode control for a quadrotor with a varying load. *Aerospace Sci Technol* 2019; 86: 737–747.
- Ghadiri H, Emami M and Khodadadi H. Adaptive super-twisting non-singular terminal sliding mode control for tracking of quadrotor with bounded disturbances. *Aerospace Sci Technol* 2021; 112: 106616. doi:10.1016/j.ast.2021.106616
- Choi I-H and Bang H-C. Adaptive command filtered backstepping tracking controller design for quadrotor unmanned aerial vehicle. *Proc Inst Mech Eng G: J Aerospace Eng* 2012; 226(5): 483–497.
- Labbadi M and Cherkaoui M. Robust adaptive backstepping fast terminal sliding mode controller for uncertain quadrotor uav. *Aerospace Sci Technol* 2019; 93: 105306.
- Raffo GV, Ortega MG and Rubio FR. An integral predictive/nonlinear hinf control structure for a quadrotor helicopter. *Automatica* 2010; 46(1): 29–39. doi:10.1016/j.automatica.2009.10.018
- Glida H-E, Abdou L and Chelili A. Optimal fuzzy adaptive backstepping controller for attitude control of a quadrotor helicopter. In: 2019 International conference on control, automation and diagnosis (ICCAD), 2 July 2019. pp. 1–6. doi:10.1109/ICCAD46983.2019.9037915
- Mohd Basri MA, Husain AR and Danapalasingam KA. A hybrid optimal backstepping and adaptive fuzzy control for autonomous quadrotor helicopter with time-varying disturbance. *Proc Inst Mech Eng Part G: J Aerospace Eng* 2015; 229(12): 2178–2195. doi:10.1177/0954410015569583
- Ferdaus MM, Anavatti SG, Pratama M, et al. Towards the use of fuzzy logic systems in rotary wing unmanned aerial vehicle: a review. *Artif Intell Rev* 2020; 53(1): 257–290. doi:10.1007/s10462-018-9653-z
- Razmi H and Afshinfar S. Neural network-based adaptive sliding mode control design for position and attitude control of a quadrotor uav. *Aerospace Sci Technol* 2019; 91: 12–27.
- Xu Q, Wang Z and Zhen Z. Adaptive neural network finite time control for quadrotor uav with unknown input saturation. *Nonlinear Dyn* 2019; 98(3): 1973–1998.
- Ullah S, Khan Q, Mehmood A, et al. Neuro-adaptive fast integral terminal sliding mode control design with variable gain robust exact differentiator for under-actuated quadrotor UAV. *ISA Trans* 2021; S0019-0578: 00129. doi:10.1016/j.isatra.2021.02.045
- Fliess M and Join C. Model-free control. *Int J Control* 2013; 86(12): 2228–2252.
- Boubakir A, Labiod S, Boudjema F, et al. Model-free controller with an observer applied in real-time to a 3-dof helicopter. *Turk J Elec Eng Comp Sci* 2014; 22(6): 1564–1581.
- Al Younes Y, Drak A, Noura H, et al. Robust model-free control applied to a quadrotor uav. *J Intell Robot Syst* 2016; 84(1-4): 37–52.
- Li Z, Ma X and Li Y. Model-free control of a quadrotor using adaptive proportional derivative-sliding mode control and robust integral of the signum of the error. *Int J Adv Robotic Syst* 2018; 15(5): 1729881418800885.
- Barth JM, Condomines JP, Moschetta JM, et al. Full model-free control architecture for hybrid UAVs. In: 2019 American control conference (ACC), 10 July 2019. pp. 71–78. IEEE.
- Glida HE, Abdou L, Chelili A, et al. Optimal model-free backstepping control for a quadrotor helicopter. *Nonlinear Dyn* 2020; 100: 3449–3468.
- Bounemour A, Chemachema M and Essounbouli N. Indirect adaptive fuzzy fault-tolerant tracking control for MIMO nonlinear systems with actuator and sensor failures. *ISA Trans* 2018; 79: 45–61. doi:10.1016/j.isatra.2018.04.014
- Chekakta Z, Zerikat M, Bouzid Y, et al. Adaptive fuzzy model-free control for 3D trajectory tracking of quadrotor. *Ijma* 2020; 7(3): 134–146. doi:10.1504/IJMA.2020.109058
- Camci E, Kripalani DR, Ma L, et al. An aerial robot for rice farm quality inspection with type-2 fuzzy neural networks tuned by particle swarm optimization-sliding mode control hybrid algorithm. *Swarm Evol Comput* 2018; 41: 1–8. doi:10.1016/j.swevo.2017.10.003

28. Mohd Basri MA, Husain AR and Danapalasingam KA. Intelligent adaptive backstepping control for mimo uncertain non-linear quadrotor helicopter systems. *Trans Inst Meas Control* 2015; 37(3): 345–361.
29. Zeghlache S, Mekki H, Bouguerra A, et al. Actuator fault tolerant control using adaptive rbfn fuzzy sliding mode controller for coaxial octorotor uav. *ISA Trans* 2018; 80: 267–278.
30. Hasançebi O and Carbas S. Bat inspired algorithm for discrete size optimization of steel frames. *Adv Eng Softw* 2014; 67: 173–185.
31. Bento PMR, Pombo JAN, Calado MRA, et al. Optimization of neural network with wavelet transform and improved data selection using bat algorithm for short-term load forecasting. *Neurocomputing* 2019; 358: 53–71.
32. Slotine JJE and Li W. *Applied nonlinear control*, Vol. 199. Englewood Cliffs, NJ: Prentice-Hall, 1991.
33. Perruquetti W, Floquet T and Moulay E. Finite-time observers: Application to secure communication. *IEEE Trans Automat Contr* 2008; 53(1): 356–360. doi:[10.1109/TAC.2007.914264](https://doi.org/10.1109/TAC.2007.914264)
34. Join C, Robert G and Fliess M. Model-free based water level control for hydroelectric power plants. *IFAC Proc Volumes* 2010; 43(1): 134–139. doi:[10.3182/20100329-3-PT-3006.00026](https://doi.org/10.3182/20100329-3-PT-3006.00026)
35. Mustafa GIY, Wang HP and Tian Y. Vibration control of an active vehicle suspension systems using optimized model-free fuzzy logic controller based on time delay estimation. *Adv Eng Softw* 2019; 127: 141–149.
36. Boubertakh H, Tadjine M, Glorennec P-Y, et al. Tuning fuzzy pd and pi controllers using reinforcement learning. *ISA Trans* 2010; 49(4): 543–551.
37. Glida HE, Abdou L, Chelihi, et al. Optimal direct adaptive fuzzy controller based on bat algorithm for UAV quadrotor. In: 2019 8th International conference on systems and control (ICSC), 23 October 2019. pp. 52–57). IEEE. doi:[10.1109/ICSC47195.2019.8950585](https://doi.org/10.1109/ICSC47195.2019.8950585)
38. Zhang X, Wang H, Tian Y, et al. Model-free based neural network control with time-delay estimation for lower extremity exoskeleton. *Neurocomputing* 2018; 272: 178–188.
39. Yang X-S. A new metaheuristic bat-inspired algorithm. *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*. Berlin, Heidelberg: Springer, 2010, pp. 65–74.
40. Yang X. *Nature-inspired Optimization Algorithms*. Amsterdam, the Netherlands: Elsevier, 2014.
41. Bello Guisado Á. *Diseño de controladores de vuelo para un dron modelo PARROT Mambo Minidrone*. PhD Thesis, 2019.