Adaptive Fuzzy model-free control for 3D trajectory tracking of Quadrotor

Zakaria Chekakta* and Mokhtar Zerikat

Department of Electrical Engineering, Ecole Nationale Polytechnique d'Oran Maurice Audin, LAAS Laboratory, Oran, Algeria.
E-mail: zakaria.chekakta@gmail.com, zerikat@yahoo.fr

Yasser Bouzid

Ecole militaire polytechnique, CSCS laboratory, Algiers, Algeria.
E-mail: yasseremp@gmail.com

Anis Koubaa

Prince Sultan University, Robotics and Internet of Things Research Lab, Saudi Arabia.
E-mail: anis.koubaa@gmail.com
*Corresponding author

Abstract:
This paper presents a novel adaptive control strategy with rejection ability for Unmanned Aerial Vehicles (UAVs), namely Fuzzy Model-Free Control (FMFC). It is based on the Model-Free Control (MFC) concept, where the control parameters are tuned on-line using Fuzzy Logic. The controller assumes an ultra-local model that can compensate unknown/unmodeled dynamics, uncertainties and external disturbances, ensuring a good robustness level. Moreover, the fuzzy logic system is used to tune on-line the Proportional-Derivative terms due to its heuristic aspect. These compensation and adaptation mechanisms allow ensuring good compromise robustness-performance even in the presence of disturbances. Several experiments, using RotorS Gazebo Micro Aerial Vehicle (MAV) Simulator, are provided to demonstrate the effectiveness of the proposed controller compared with other techniques. The Fuzzy model free controller shows superior performance without the time-consuming and tedious tuning task.

Keywords: Micro Aerial Vehicle (MAV); Model free control; Fuzzy logic; Adaptive control; Robust control.

Biographical notes:
Zakaria Chekakta is a PhD Student at National School Polytechnic of Oran (ENPO) in Algeria, He received his B.S in Electrical Engineering and M.S degree in Electronic and Embedded system in 2014, and 2016 respectively from ENPO. He is currently conducting researches on advanced optimal control of quadrotor including reinforcement learning control using artificial intelligent algorithms.
Mokhtar Zerikat received his B.S degree in electrical engineering and his M.S. and Ph.D degrees in electronics from the University of Sciences and Technology of Oran, Algeria, in 1982, 1992, and 2002, respectively. He is currently a professor in the Department of Electrical Engineering at Ecole Nationale Polytechnique d'Oran, Algeria. His current research includes embedded electronic systems, modeling and adaptive control, image processing and intelligent applications. He is involved in industry projects while engaged in teaching, research and consulting. He is the chairman of the IEEE in Algeria state. And a member of the industrial COST Education Committee.
Yasser Bouzid received the Master of Science (MSc) from the University of Paris XI, France (in 2015). He received Ph.D. degree in Automatic from Paris Saclay University in 2018. He is currently an assistant professor at the Ecole Militaire Polytechnique, Bordj El-Bahri, Algeria and a member in the CSCS Lab. His research interests include nonlinear control, fault tolerant control, observation and estimation, path planning, heuristic algorithms, unmanned aerial vehicles, multi-agent systems, soft robotics.
Anis Koubaa is currently a Professor of computer science and the Leader of the Robotics and Internet of Things Research Laboratory, Prince Sultan University. He is also a Senior Researcher with CISTER and ISEP-IPP, Porto, Portugal, and a Research and Development Consultant with Gaitech Robotics, China. His current research deals with the integration of robots and drones into the Internet of Things (IoT) and clouds. He is also a Senior Fellow of the Higher Education Academy (HEA), U.K. He has been the Chair of the ACM Chapter in Saudi Arabia, since 2014.

1 Introduction

Nowadays, Unmanned Aerial Vehicles (UAVs) play a crucial role in many services such as aerial surveillance (1), tracking (2), Internet-of-Drones (3), search and rescue operations (4), remote sensing (5), delivery, etc. Such an interest has attracted academic and industry researchers to solve problems related to their large scale deployment. Currently, a variety of UAVs are either deployed or under development. Nevertheless, quadrotors are the most popular configurations of multi-rotors UAVs. This is likely due to some of their characteristics such as the mechanical structure simplicity, vertical take-off and landing (VTOL) capability and high maneuverability.

In the literature of quadrotor control, a variety of control techniques have been developed and successfully implemented for attitude stabilization problem, and trajectory tracking as well. First studies were primarily focused on linear control (6; 7). However, they were limited to non-agile motions due to the fact that they were based on the linearized dynamics model. Besides linear control techniques, many researchers have proposed several non-linear methods to deal with the non-linear dynamics of the quadrotor. (8; 9; 10). Several approaches have also developed and presented comprehensively in (11).

The non-linear approaches are interesting. However, they suffer from the dependence on the dynamic model of the system. One of the main issues is the lack of precise information about the non-linear high coupling terms, especially under the effect of disturbances. To solve this problem, considerable attention has been recently paid to the development of an improved technique to deal with these unknown/unmodeled dynamics and disturbances. Among these interesting approaches, the Model Free Control (MFC) proposed by Fliess (12), is noteworthy. It is called Intelligent PID (iPID). It is important to note that the terminology of model free control is not new in the literature, while the term Intelligent PID has also been used previously, but with different significations.

1.1 Paper Contributions

Even though the efficiency of model free control has been improved in recent years, most of the improvements have been achieved due to different combinations between model free control and non-linear auxiliary controllers. However, one of the major challenging in the control system development is the robustness against disturbances and uncertainties. Consequently, a new model free controller with Fuzzy Logic self-tuning PD feedback (FiPD is referred to Fuzzy Logic Model Free Control) is presented. The PD gains are refined during the control process to provide good performance. The control action can be taken based on the estimation of the ultra-local model to compensate all the unknown/unmodeled dynamics, uncertainties and disturbances. Moreover, a deep study of robustness level compared to Fuzzy Logic PID and classic model free control is highlighted even when disturbances occur.

In addition, the control problem was divided into two levels supported by the dynamic structure of the multi-rotor system MAV. The low-level (inner loop) contains the attitude controller and computes the angular speed of each rotor. The high-level controller (outer loop) deals with the position controller and generates the desired roll and pitch angles. The proposed high-level controller strategy is applied to control the position of the quadrotor, and to ensure the tracking of a desired trajectory. Specific attention is given to the accuracy of tracking and energy consumption based on several performance metrics, such as the integral Squared Error (ISE) and the Integral Squared Control Input (ISCI).

Moreover, to evaluate the proposed strategy, the framework RotorS Gazebo MAV Simulator (13) was used, which allows to use the same controller algorithm including their parameters in the simulation as well as in real application.

1.2 Outlines

The remaining of the paper is organized as follows: Section 2 provides a brief background on the model free control approach and its related work. The dynamic model of quadrotor is described in Section 3. Section 4 presents the proposed control strategy. Experimental validation, using realistic simulator under various scenarios to support the main goal of this paper, is presented in Section 5. Concluding remarks are given in Section 6.

2 Brief background and related work

Initially, the basic of model free control was introduced by M. Fliess and C. Join (14; 15). The fundamental idea of this technique is to use an observation mechanism to estimate online the ultra-local model, which represents the dynamic model of the system in addition to the unknown/unmodeled dynamics, uncertainties and external disturbances. It is important to note that this estimation is valid for a short time interval.

Usually, model free control also uses one of the linear/non-linear control technique as an auxiliary input. In most cases, the linear controller could be used due to its simplicity and efficiency. Otherwise, the non-linear controller must be used to deal with the non-linear terms of the system and achieve certain control performance.

The major benefit of this control approach is that it does not require a vast knowledge about the system dynamics whose parameters are difficult to be identified relatively accurately using system identification. Thus, it can be easily applied to control unknown/complex systems.

Several studies have been carried out, including the work of A. Chemori (16), who designed a control structure using model free to handle the under-actuation in stable limit cycle generation for the Inertia Wheel Inverted Pendulum. Experiments in real-time shows the effectiveness of the proposed control strategy and its ability for external disturbances rejection. The model free controller also applied by H. Abouaïssa and S. Chouraqui (17) to a six degree of freedom PUMA-560 arm manipulator to track a desired trajectory. Simulation results illustrated that the model free controller is able to provide outstanding tracking performance robustly. Maroua Haddar et al. (18) proposed an Intelligent PD controller based on robust model free control strategy for attenuating vibrations caused by road disturbance inputs in vehicle suspension system.

For the quadrotor system, the model free control is implemented. It is combined with LQR feedback controller and tested on a real Qball-X4 quadrotor by Younes et. al (19). Flight control evaluation in real-world results confirm the significance of employing the model free control. Same authors proposed a combination between the non-linear integral backstepping controller and model free controller (20). This combination shows superior performance with respect to other control techniques in terms of trackability of the desired path, stability, and robustness with the presence of the actuator fault. Wang et al. (21) developed a model free based terminal sliding-mode control (MFTSMC) to eliminate the tracking error in the finite time. Simulation results demonstrate that MFTSMC has an advantage over the PID, backstepping, sliding-mode, and iPD control in term of the rise time, steady-state error, robustness, and disturbance rejection. Bouzid et al (22) derived an improved energy based control via an online disturbance compensation based on the model free control approach. The efficiency of this approach is demonstrated in various scenarios. High-level of robustness is ensured with respect to parameters uncertainties or external disturbances. Z. Li et al (23) developed a model free control using an adaptive proportional derivative sliding mode control in the presence of external disturbances. Cascaded model free control was tested by Bekcheva et al (24) on a realistic quadrotor model in the presence of unknown time-varying disturbance for the tracking of an aggressive trajectory.

3 Dynamic Model of Quadrotor

This section presents a dynamic mode based on Wang's model (25) that describes the quadrotor using Newton-Euler. However, some assumptions were made:
Assumption 1: The framework of the vehicle and the propellers are rigid and quite symmetrical.
Assumption 2: The origin of the Body-fixed frame coincides with the CoG of the vehicle.
Assumption 3: The dynamics of the rotors are very fast and then will be ignored.

Let $I = \{x_i, y_i, z_i\}$ denotes the inertial frame and $B = \{x_b, y_b, z_b\}$ denotes the Body-fixed frame (see Figure 1).

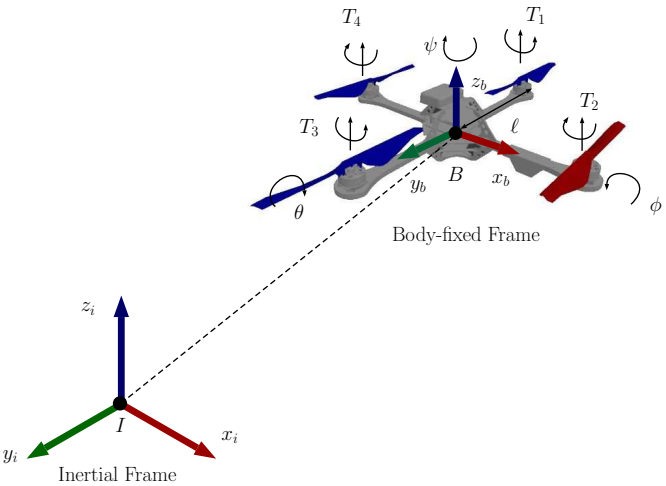


Figure 1 Forces and moments acting on quadrotor with attached coordinate frames

The orientation from the Body-fixed frame to the Inertial frame can be described using the rotational matrix $\mathfrak{R}(1)$. The order of rotation used in this work is yaw (ψ) followed by pitch (θ) followed by roll (ϕ) around the z, y and x axes respectively.

$$\mathfrak{R} = \begin{bmatrix} c_\psi c_\theta & s_\phi s_\theta - c_\phi s_\psi & s_\phi s_\psi + c_\phi c_\psi s_\theta \\ c_\theta s_\psi & c_\phi c_\psi + s_\phi s_\psi s_\theta & c_\phi s_\psi s_\theta - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\phi c_\theta \end{bmatrix} \tag{1}$$

where: $c_{(.)} = \cos(.)$ and $s_{(.)} = \sin(.)$

Based on the previous assumptions, and using the Newton-Euler laws that relate the motion of the quadrotor with the sum of external forces and moments acting on the vehicle in the Body-fixed frame, the following Equation is obtained:

$$\begin{bmatrix} mJ & O_{3\times3} \\ O_{3\times3} & J \end{bmatrix} \begin{bmatrix} \dot{V}_b \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} \omega_b \times (mV_b) \\ \omega_b \times (J\omega_b) \end{bmatrix} = \begin{bmatrix} F_b \\ M_b \end{bmatrix} \tag{2}$$

where:

- m is the mass of the quadrotor;
- $O_{3\times3}$ is a 3 by 3 dimensional zero matrix and $\times$ denotes the cross product;
- $J \in \mathbb{R}^{3\times3}$ is the inertia matrix about the center of gravity, by Assumption 1, the inertia matrix turn into $J = dig(J_x, J_y, J_z)$;
- $V_b = [u, v, w] \in \mathbb{R}^{3\times1}$ is the linear velocity of the quadrotor;
- $\omega_b = [p, q, r] \in \mathbb{R}^{3\times1}$ is the angular velocity in the Body-fixed frame;

- $F_b = [F_x, F_y, F_z]^T \in \mathbb{R}^{3\times 1}$ and $M_b = [M_x, M_y, M_z]^T \in \mathbb{R}^{3\times 1}$ are respectively the external forces and moments acting on the quadrotor in the Body-fixed frame.

Since the quadrotor model contains rotational and translational motions, using the relationship between the linear accelerations and forces within the Body-fixed frame, the translational motion can be computed by the following Equation:

$$\begin{bmatrix} \ddot{x_b} \\ \ddot{y_b} \\ \ddot{z_b} \end{bmatrix} = \begin{bmatrix} r\dot{x_b} - q\dot{z_b} \\ p\dot{z_b} - r\dot{x_b} \\ q\dot{x_b} - p\dot{y_b} \end{bmatrix} + \frac{1}{m} F_b \tag{3}$$

Further, the Equation describes the total external forces as follows:

$$F_b = \begin{bmatrix} F_x^b \\ F_y^b \\ F_z^b \end{bmatrix} = \begin{bmatrix} -\sum_{i=1}^{4} H_{ix} \\ -\sum_{i=1}^{4} H_{iy} \\ \sum_{i=1}^{4} T_i \end{bmatrix} + \mathfrak{R}^T \begin{bmatrix} 0 \\ 0 \\ -G \end{bmatrix} \tag{4}$$

where:

- $G = -mg$ is the gravity along z_i axis in the Inertial frame and g is the gravity acceleration. Here, the force in the Inertial frame can be translated into the Body-fixed frame using the rotation matrix $\mathfrak{R}^{-1} = \mathfrak{R}^T$;
- H_{ix} the hub force of the i^{th} rotor along x_b axis;
- H_{iy} the hub force of the i^{th} rotor along y_b axis;
- T_i the thrust force of the i^{th} rotor along z_b axis.

Then, the rotational motion id derived using the relation between the moments and the angular acceleration in the Body-fixed frame according to Equation (5):

$$\begin{bmatrix} J_x\dot{p} \\ J_y\dot{q} \\ J_z\dot{r} \end{bmatrix} = \begin{bmatrix} (J_y - J_z)qr \\ (J_z - J_x)pr \\ (J_x - J_y)pq \end{bmatrix} + M_b \tag{5}$$

Otherwise, the total moments $M_b = [M_x \ M_y \ M_z]^T$ can be written as follows:

$$M_b = \begin{bmatrix} \mathcal{B}_x + \mathcal{M}_\phi + \mathcal{G}_x \\ \mathcal{B}_y + \mathcal{M}_\theta + \mathcal{G}_y \\ \mathcal{Q} \end{bmatrix} \tag{6}$$

where:

- $\mathcal{M}_\phi$ and $\mathcal{M}_\theta$ are the rolling and pitching moments around x and y axis respectively; they can be described by:

$$\mathcal{M}_\phi = l(T_2 - T_4) \qquad \mathcal{M}_\theta = l(-T_1 + T_3)$$

where ℓ is the arm length.

- $\mathcal{Q}$ is the yawing moment;
- $\mathcal{B}_x$ and $\mathcal{B}_y$ are the blad flapping moments of i^{th} rotor along x and y axis respectively;
- $\mathcal{G}_x$ The gyroscopic moment along x axis defined as:

$$\mathcal{G}_x = (-1)^{i+1} \sum_{i=1}^{4} J_r q \Omega_i$$

where, J_r is the inertia of a rotor;

- $\mathcal{G}_y$ is the gyroscopic moment along y axis and is obtained as:

$$\mathcal{G}_y = -(-1)^{i+1} \sum_{i=1}^{4} J_r p \Omega_i$$

On the other hand, the equation that represents the relation between the angular velocities $\omega_b = [p, q, r]^T$ of the quadrotor in the Body-fixed frame and the angular velocities $\dot{\eta} = [\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$ in the Inertia frame is written as follows:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s_\phi \tan\theta & c_\phi \tan\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi \sec\theta & c_\phi \sec\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{7}$$

Near to the equilibrium state, a small angles assumption is made where $\cos\phi \approx 1$, $\cos\theta \approx 1$ and $\sin\phi \approx \sin\theta \approx 0$. Therefore according to (7) the following is obtained:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} - s_\theta\dot{\psi} \\ c_\phi\dot{\theta} + s_\phi c_\theta\dot{\psi} \\ -s_\phi\dot{\theta} + c_\phi c_\theta\dot{\psi} \end{bmatrix} \approx \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{8}$$

$\mathcal{X} = [x, y, z]^T$ denotes the position of the quadrotor in the Inertial frame. The following presents the complete mathematical model of the quadrotor(9)(10):

It is commonly assumed that the thrust and drag forces are proportional to the square of the rotor's speeds:

$$T_i = C_f \Omega_i^2 \qquad \mathcal{Q}_i = C_m \Omega_i^2$$

where $C_f|_m$ is aerodynamic coefficient.

$$\begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{z}\end{bmatrix} = \Re\left(\begin{bmatrix}\dot{\psi}\dot{x}_b - \dot{\theta}\dot{z}_b\\ \dot{\phi}\dot{z}_b - \dot{\psi}\dot{x}_b\\ \dot{\theta}\dot{x}_b - \dot{\phi}\dot{y}_b\end{bmatrix} + \frac{1}{m}\begin{bmatrix}-\sum_{i=1}^{4} H_{ix}\\ -\sum_{i=1}^{4} H_{iy}\\ \sum_{i=1}^{4} f_i\end{bmatrix}\right) + \begin{bmatrix}0\\0\\-g\end{bmatrix} \tag{9}$$

$$\begin{bmatrix}J_x\ddot{\phi}\\ J_y\ddot{\theta}\\ J_z\ddot{\psi}\end{bmatrix} = \begin{bmatrix}(J_y - J_z)\dot{\theta}\dot{\psi}\\ (J_z - J_x)\dot{\phi}\dot{\psi}\\ (J_x - J_y)\dot{\phi}\dot{\theta}\end{bmatrix} + \begin{bmatrix}(-1)^{i+1}\sum_{i=1}^{4} k_{\mathcal{B}}\rho A(\Omega_i R_{rad})^2 R_{rad} + \mathcal{M}_\phi + (-1)^{i+1}\sum_{i=1}^{4} J_r q\Omega_i\\ (-1)^{i+1}\sum_{i=1}^{4} k_{\mathcal{B}}\rho A(\Omega_i R_{rad})^2 R_{rad} + \mathcal{M}_\theta - (-1)^{i+1}\sum_{i=1}^{4} J_r p\Omega_i\\ (-1)^{i+1}\sum_{i=1}^{4} k_{\mathcal{Q}}\rho A(\Omega_i R_{rad})^2 R_{rad}\end{bmatrix} \tag{10}$$

Hereinafter, the input control vector is defined as follows:

$$\begin{cases} u_1 = \Gamma = C_f \sum_{i=1}^{i=4} \Omega_i^2 \\ u_2 = \mathcal{M}_\phi = C_f \ell \sum_{i=1}^{i=4} \Omega_i^2 \\ u_3 = \mathcal{M}_\theta = C_f \ell \sum_{i=1}^{i=4} \Omega_i^2 \\ u_4 = \mathcal{Q} = C_m \sum_{i=1}^{i=4} \Omega_i^2 \end{cases} \tag{11}$$

It is well-known that the control-oriented model neglects some forces and moments such as hub forces, blade-flapping moments and the gyroscopic moments. Under this approximation and the above simplifications, the simplified model can be written as follows:

$$\begin{cases} \ddot{x} = \frac{u_1}{m}(c_\phi s_\theta c_\psi + s_\phi s_\psi) + \rho_x \\ \ddot{y} = \frac{u_1}{m}(c_\phi s_\theta s_\psi - s_\phi c_\psi) + \rho_d \\ \ddot{z} = \frac{u_1}{m}(c_\phi c_\theta) - g + \rho_d \\ \ddot{\phi} = \frac{J_y - J_z}{J_x}\dot{\theta}\dot{\psi} + \frac{u_2}{J_x} + \rho_\phi \\ \ddot{\theta} = \frac{J_z - J_x}{J_y}\dot{\phi}\dot{\psi} + \frac{u_3}{J_y} + \rho_\theta \\ \ddot{\psi} = \frac{J_x - J_y}{J_z}\dot{\phi}\dot{\theta} + \frac{u_4}{J_z} + \rho_\psi \end{cases} \tag{12}$$

where $\mathcal{D} = [\rho_x, \rho_y, \rho_z, \rho_\phi, \rho_\theta, \rho_\psi]^T \in \mathbb{R}^{6\times 1}$ is a bounded vector holds unknown noise and disturbances.

Contemporary research on UAVs tend to focus on the control system rather than the system modeling and identification. Further, the enhancement of the control performance depends on the accurate knowledge of the system parameters. In fact, precise estimation of the non-linear high coupling system parameters such as quadrotor is often not possible.

4 Control strategy

Although non-linear approaches exhibit superior performance, they require a wide knowledge about the system parameters and dynamics. It is difficult to obtain a precise model for quadrotor, due to the influence of high coupling terms, uncertainties and unknown external disturbances. Recently, an interesting approach was proposed to solve this problem (12). It is called Model Free Control (MFC) or intelligent PID (iPID). The efficiency of this approach has been improved by different alternatives of non-linear controllers.

Therefore, it is possible to improve the effectiveness of the model free control approach by introducing self-tuning and adaptation mechanism to the controller gains via Fuzzy Logic. As a result, the control performance for 3D trajectory tracking can be improved as well as the robustness level against different disturbances.

4.1 Model Free Control

Generally, a non-linear n^{th} order single input single output (SISO) system can be expressed by the following form:

$$x^{(n)} = g(x, \dot{x}, \ldots, x^{(n)}) + au \tag{13}$$

where, $g(\cdot)$ is the model of the system, $u \in \mathbb{R}$ is the input, and $a \in \mathbb{R}$ is unkonwn input gain.

The basic idea of model free control is to locally approximate the system presented in Equation (13) by using a simple model $F \in \mathbb{R}$:

$$x^{(n)} = F + \lambda u \tag{14}$$

where:

$F \in \mathbb{R}$ is the ultra-local model;

$\lambda \in \mathbb{R}$ is non-physical parameter.

It should be noted that the ultra-local model F represents the dynamic model of the system in addition to the unknown/unmodeled dynamics, uncertainties and

disturbances at time t, and it can be estimated by the following equation:

$$\widehat{F}(t) = \widehat{x}(t)^{(n)} - \lambda u(t-\varepsilon) \tag{15}$$

where:

- $x(t)$ the last output;
- $x^{(n)}$ is the derivative of order $n \geq 1$ of x. The value of n is determined by the control designer. Several precedent cases show that $n \in \mathbb{N}$ should constantly be chosen minimum at possible (12);
- $\lambda \in \mathbb{R}$ is a non-physical constant to estimate the unknown input gain a. It is chosen by the designer to attain certain performance;
- $u(t-\varepsilon)$ is the latest applied control input;
- ε is the time-delay and the smaller ε is, the best estimation of the ultra-local model $\widehat{F}(t)$

The estimation of the ultra-local model $\widehat{F}$ will be updated on-line at each iteration. Usually, model free control uses the classical Proportional Derivative (PD) controller as auxiliary input due to its simplicity. It has the following form:

$$u_c(t) = k_p e(t) + k_d \dot{e}(t) \tag{16}$$

where, $e(t) = x(t) - x_d(t)$ denotes the tracking error between the reference trajectory $x_d(t)$ and the output $x(t)$. k_p and k_d are the proportional and derivative gains respectively.

The global control input u is given by:

$$u(t) = -\frac{\widehat{F}(t) - x_d^{(n)}(t) + u_c(t)}{\lambda} \tag{17}$$

The value of n has been chosen according to the system stability and the type of the feedback controller used in the system (12) . For this work, a second order system was used with $n = 2$. Furthermore, the time-delay ϵ was neglected:

$$\ddot{x} = F + \lambda u \tag{18}$$

Therefor, the ultra-local model can be estimated as follows:

$$\widehat{F} = \widehat{\ddot{x}} + \lambda u \tag{19}$$

Finally, the control input of model free controller shown in Figure 2 has the following form:

$$u = -\frac{\widehat{F} - \ddot{x}_d + k_p e + k_d \dot{e}}{\lambda} \tag{20}$$

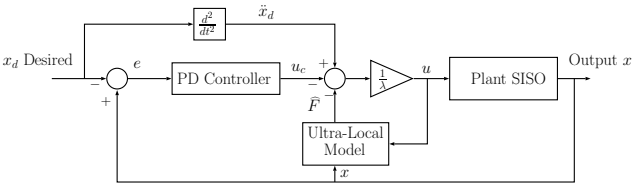


Figure 2 Structure of model free control

Stability of model free control By substituting equation (20) in (18), we get the following relation:

$$\ddot{x} = \ddot{x}_d + F - \widehat{F} - k_p e - k_d \dot{e} \tag{21}$$

If the estimation error of F as e_{est} is defined as, it is possible to obtain:

$$\ddot{e} + k_p e + k_d \dot{e} - e_{est} = 0 \tag{22}$$

where $e_{est} \in \mathbb{R}$ is time-varying.

Otherwise, each small time interval h is written as follows:

$$e_{est} = |\ddot{x} - \widehat{\ddot{x}} - \lambda(u(t) - u(t-\epsilon))| \leqslant |\frac{d\widehat{\dot{x}}}{dt}h| + |\frac{d\widehat{u}}{dt}h| \tag{23}$$

Usually, in physical system $\frac{d\widehat{\dot{x}}}{dt}$ and $\frac{d\widehat{u}}{dt}$ are bounded and not equal to infinity, hence it gives the following:

$$\frac{d\widehat{\dot{x}}}{dt} \leqslant \epsilon_{\dot{x}} \qquad\qquad \frac{d\widehat{u}}{dt} \leqslant \epsilon_u$$

where, the estimation $\widehat{\dot{x}}$ is supposed to be filtered, if a time interval h is rather small with respect to system dynamic selected. e_{est} will be bounded in small limit $|e_{est}| \leq \epsilon$, $\epsilon \geq 0$. Thus, by choosing appropriate gain k_p and k_d, the error in equation (22) will converge to this small limit ϵ. Consequently, the system is practically stable (25).

4.2 Fuzzy logic PD controller

The best performance of PD controller depends on the optimal selection of the controller gains. Most likely, tuning these gains is a tedious task. Herein, the fuzzy logic system was chosen to tune these gains due to its heuristic characteristic and effectiveness. This approach turns the PD controller into an adaptive controller.

Figure 3 shows the fuzzy logic system used in this work, which has two inputs (the tracking position error e and the tracking velocity error $\dot{e}$), and two outputs (k_p, k_d). Furthermore, the fuzzy system uses the Mamdani method for the basic rules.

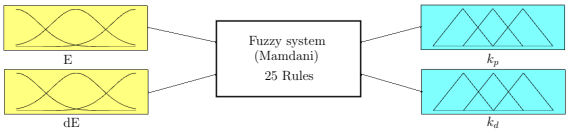


Figure 3 Fuzzy logic system for PD controller

Details of membership functions are represented in Figure 4. For each input variable e and $\dot{e}$, the fuzzy logic system is designed to have five triangular membership functions: Negative Large (NL), Negative Small (NS), Zero (ZE), Positive Small (PS) and Positive Large (PL). The input membership functions are contained in the universal sets. The universal sets of all inputs ranging have the same range [-1 1].

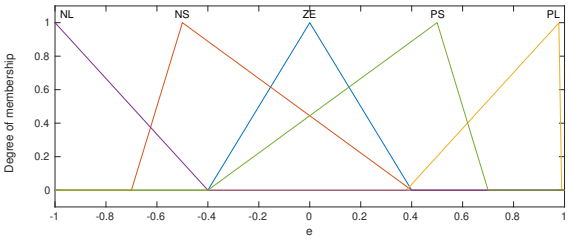


Figure 4 Membership functions for the all inputs

For the output variables, the fuzzy logic system is designed to have seven triangular membership functions that are considered as positive very small (PVS), positive small (PS), positive medium small (PMS), positive medium (PM), positive medium large (PML), positive large (PL) and positive very large (PVL).

The membership functions for the output variables k_p and k_d are shown in Figure 5. The universal sets ranging can be chosen by hit and trial. For this work they have been chosen between [0.1 5].

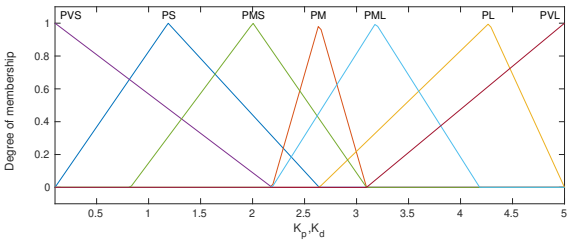


Figure 5 Membership functions for outputs k_p and k_d

The set linguistic rules is the major part of the fuzzy logic system. In many cases, it is easy to translate an expert experience into these rules and any number of such rules can be created to define the actions of the system outputs. In some other cases these rules can be obtained from some trial and error approaches Tables 1-2 show the 25 rules used in this work for all possible combinations of the inputs for tuning PD gains.

Table 1 Rules for k_p

$\dot{e}/e$	NL	NS	ZE	PS	PL
NL	PVL	PVL	PVL	PVL	PVL
NS	PML	PML	PML	PL	PVL
ZE	PVS	PVS	PS	PMS	PMS
PS	PML	PML	PML	PL	PVL
PL	PVL	PVL	PVL	PVL	PVL

Table 2 Rules for k_d

$\dot{e}/e$	NL	NS	ZE	PS	PL
NL	PVS	PML	PM	PL	PVL
NS	PMS	PML	PL	PVL	PVL
ZE	PM	PL	PL	PVL	PVL
PS	PML	PVL	PVL	PVL	PVL
PL	PVL	PVL	PVL	PVL	PVL

All rules shown in Table 1 and Table 2 are connected with the AND operator. For example:
IF e is NS **AND** $\dot{e}$ is ZE, **THEN** k_p is PVS **AND** k_d is PL.

The output is fuzzy information. Since the output for PD controller requires a precise real number, it will be necessary to convert the membership function obtained to a precise value. This operation called defuzzification.

4.3 Cascaded control approach for Multirotor system

Figure 6 illustrates the control architecture used for this work. It is named as a hierarchical or cascaded control structure. Moreover, this control scheme divides the control problem into two loops. The first one is the outer loop, which provides the desired roll and pitch angles in addition to the thrust. The last one is the inner loop, which is used to follow these angles to achieve the desired position.

However, the main objective of this work is to increase the trajectory tracking performance. Therefore, the proposed strategy was applied in the outer loop for the position controller. Whereas the controller used in the inner loop is a simple PID controller, as it has proved to be effective in attitude control.

4.3.1 Position control (Outer-loop):

The altitude control input of the quadrotor is the resulting thrust force of the four rotors, which is based on the following equation:

$$\sum_{1}^{4} T_i = u_1 \tag{24}$$

where, the derived control law is written as follows:

$$u_1 = -\frac{1}{\lambda_z}(\widehat{F}_z - \ddot{z}_d + k_{pz}e_z + k_{dz}\dot{e}_z) \tag{25}$$

Model free control for positions x and y is given by the virtual control input u_x and u_y respectively. The corresponding control laws can be written as:

$$\begin{aligned} u_x &= -\tfrac{1}{\lambda_x}(\widehat{F}_x - \ddot{x}_d + k_{px}e_x + k_{dx}\dot{e}_x) \\ u_y &= -\tfrac{1}{\lambda_y}(\widehat{F}_y - \ddot{y}_d + k_{py}e_y + k_{dy}\dot{e}_y) \end{aligned} \tag{26}$$

where:

$$\begin{cases} \widehat{F}_x = \widehat{\ddot{x}} - \lambda_x u_x \\ \widehat{F}_y = \widehat{\ddot{y}} - \lambda_y u_y \\ \widehat{F}_z = \widehat{\ddot{z}} - \lambda_z u_1 \end{cases} \tag{27}$$

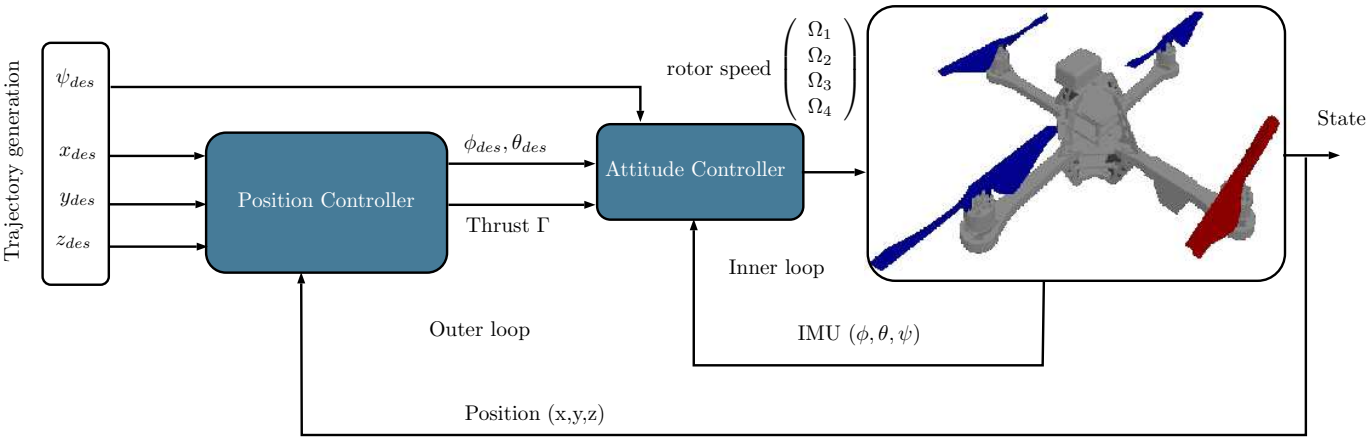


Figure 6 Control structure of quadrotor

4.3.2 Attitude control (Inner-loop):

The roll control input u_2 is the difference in thrust between rotors 1 and 3. The pitch control input u_3 is the difference in thrust between rotors 2 and 4. The yaw control input u_4 is the difference in torque between the two clockwise turning rotors 2 and 4 and the two counter-clockwise 1 and 3. The outputs of PID controller are given by:

$$\begin{cases} u_2 = k_{p\phi} e_\phi + k_{i\phi} \int e_\phi + k_{d\phi} \dot{e}_\phi \\ u_3 = k_{p\theta} e_\theta + k_{i\theta} \int e_\theta + k_{d\theta} \dot{e}_\theta \\ u_4 = k_{p\psi} e_\psi + k_{i\psi} \int e_\psi + k_{d\psi} \dot{e}_\psi \end{cases} \tag{28}$$

where:

$$\begin{cases} e_\phi = \phi_{des} - \phi \\ e_\theta = \theta_{des} - \theta \\ e_\psi = \psi_{des} - \psi \end{cases} \tag{29}$$

The desired yaw angle ψ_{des} is given by the operator or the trajectory generator, while the desired roll ϕ_{des} and pitch θ_{des} angles are generated from Equation (30):

$$\begin{cases} \phi_{des} = \frac{1}{g} \times (u_x \cos(\psi_{des}) - u_y \sin(\psi_{des})) \\ \theta_{des} = \frac{1}{g} \times (u_x \sin(\psi_{des}) + u_y \cos(\psi_{des})) \end{cases} \tag{30}$$

5 Results and Discussion

In this work, the AscTech hummingbird quadrotor was chosen for the experiments. Table 3 indicates the quadrotor parameters. In addition, the yaw angle is assumed to be zero due to the high coupling in position and attitude dynamics. The proposed control strategy was compared with two other control techniques, Fuzzy Logic PID control and model free control iPD, to illustrate the effectiveness of the proposed controller.

5.1 Experimental setup

All the experiments were conducted on a laptop with Intel® Core™ i5-2410M CPU @ 2.30GHz×4, 8 GB RAM, and 600 GB disk drive, using RotorS Gazebo MAV Simulator. This framework permits the employment of the same designed controllers, including their parameters, in the simulation as well as on real application. Furthermore, the strategy followed for ROS (Robot Operating System) integration of the Fuzzy model free controller is to create a ROS node to interface the controller to ROS, while model free control was implemented using C++ (26). The fuzzy logic node was implemented within Matlab/Simulink block. Auto-code generation was used to extract the node package. The low-level attitude PID controller node, used in this work, was previously implemented by Kamal et al. (27) for Model predictive control. The initial position of the quadrotor is $x = 0(m), y = 0(m), z = 0.06(m)$.

Table 3 AscTech hummingbird quadrotor parameters

Parameter	Symbol	Value	Unit
Inertia on x-axis	J_x	0.007	kg m^2
Inertia on y-axis	J_y	0.007	kg m^2
Inertia on z-axis	J_z	0.012	kg m^2
Quadrotor's Mass	m	0.716	kg
Arm length	ℓ	0.17	m
Thrust coefficient	C_f	$8.54858e^{-6}$	N kg s^2
Drag coefficient	C_m	0.016	Nm kg s^2

Table 4 shows the control gains of model free controller.

Table 4 Model free controller gains (P: parameter,V:value)

P	V	P	V	P	V
k_{px}	0.65	k_{dx}	1.3	λ_x	1
k_{py}	0.65	k_{dy}	1.3	λ_y	1
k_{pz}	8.3	k_{dz}	4.6	λ_z	5

5.2 In-depth trajectory tracking analysis

In this experiment, a square trajectory ($2m \times 2m$) was used as a reference trajectory (Figure 7), where the mathematical expression can be found in the Appendix .

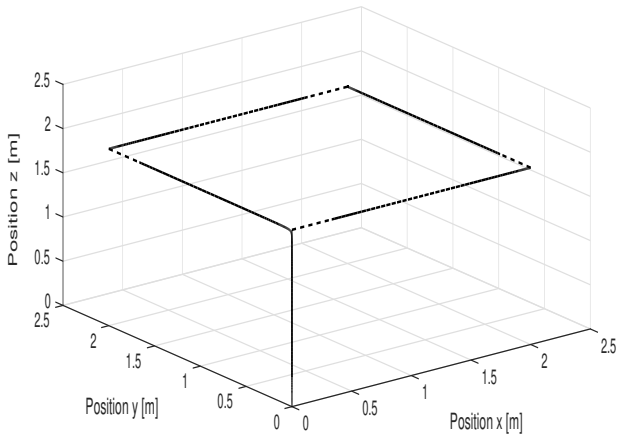


Figure 7 Desired Path

Several realistic scenarios were performed to show the effectiveness of the proposed control strategy. These scenarios are as follows:

(1) Basic scenario (nominal case): For this scenario, the quadrotor should follow a desired trajectory under ideal conditions without any external disturbances.

(2) Parameters uncertainties: Precisely parameter estimation and well-knowledge about the dynamics of the quadrotor are often challenging to obtain. Hence, it is assumed that the aerodynamic coefficients and the inertia elements are underestimated by 50% from their real values. The control technique should be able to tolerate the uncertainties in the system parameters.

(3) Payload: In this scenario, we assume the quadrotor is carrying an extra weight of $0.3kg$. The controller should be able to generate the appropriate control effort to follow the desired trajectory even in presence of mass variation.

(4) Wind disturbance: The wind disturbance is a big challenge for the quadrotor control problem due to the underactuated nature in the translational motion. To further understand the situation, if a continuous and constant wind disturbance is appeared along the $x-$axis, the quadrotor must maintain a small pitch angle to keep the quadrotor in the desired x position. In this case, to test wind robustness, an unknown and time- varying external wind disturbance is applied in the x-direction, and constant wind in the y-direction around the linear velocity indicated in Figure 8.

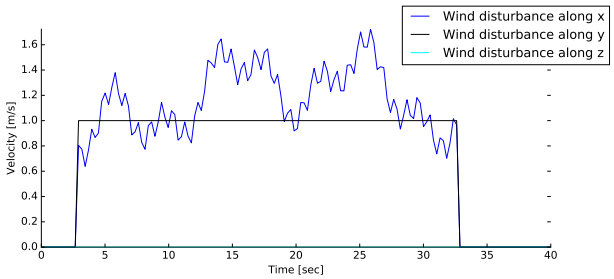


Figure 8 Wind disturbance

For the basic scenario, the tracking errors along x, y and z-axes, the desired roll ϕ_{des} and pitch θ_{des} angles, generated by the outer-loop, (high-level controller) are plotted using virtual control input u_x, u_y, and the last plot is the control effort (thrust Γ).

Figure 9 presents the results obtained from the basic scenario. Based on curves in Figure 9(a-c), it can be observed that the proposed controller FiPD ensures the tracking of the desired trajectory with small errors, and it is roughly the same compared to model free control (iPD). However, the Fuzzy Logic PID is the least accurate. Otherwise, the thrust curves (control effort) in Figure 9(f) show the energy consumption, we can see that the FiPD and iPD consume less energy compared to Fuzzy Logic PID, the reason is the initial measured altitude position $z = 0.06m$. Herein, the initial error for the position z, computed by the controller is 0.06. In Figure 9(c), the proposed controller and model free control deal with this error and no control effort would be generated until the controller receives the desired trajectory and take-off, while the Fuzzy logic PID is generating control effort signal even when there is no desired altitude given as an input. Also, after quadrotor landing, the proposed controller and iPD would generate a null control effort signal, while Fuzzy Logic PID keep generating control effort. Curves in Figure 9(d-e) show the desired pitch and roll angles given to the low-level attitude controller to achieve the motion along x and y-axes respectively.

Overall, in term of trajectory tracking accuracy as a performance criteria, the proposed FiPD controller shows superior performance with respect to other techniques. This is due to the compensation mechanism provided by the ultra-local model, and the adaptation technique obtained by the fuzzy logic system. In addition, it is important to note that the proposed FiPD is robust and able to reject the disturbances from the environment .

To obtain a deeper analysis of robustness, two metrics were introduced, Integral Square Error (ISE) quantifying the trajectory tracking error and Integral Squared Control Input (ISCI) quantifying the energy consumption. The metrics are defined as follows:

- ISE= $\int_{t_0}^{t_f} \left(e_x^2(t) + e_y^2(t) + e_z^2(t)\right) dt$

- ISCI= $\int_{t_0}^{t_f} \left(u_1^2(t)\right) dt$

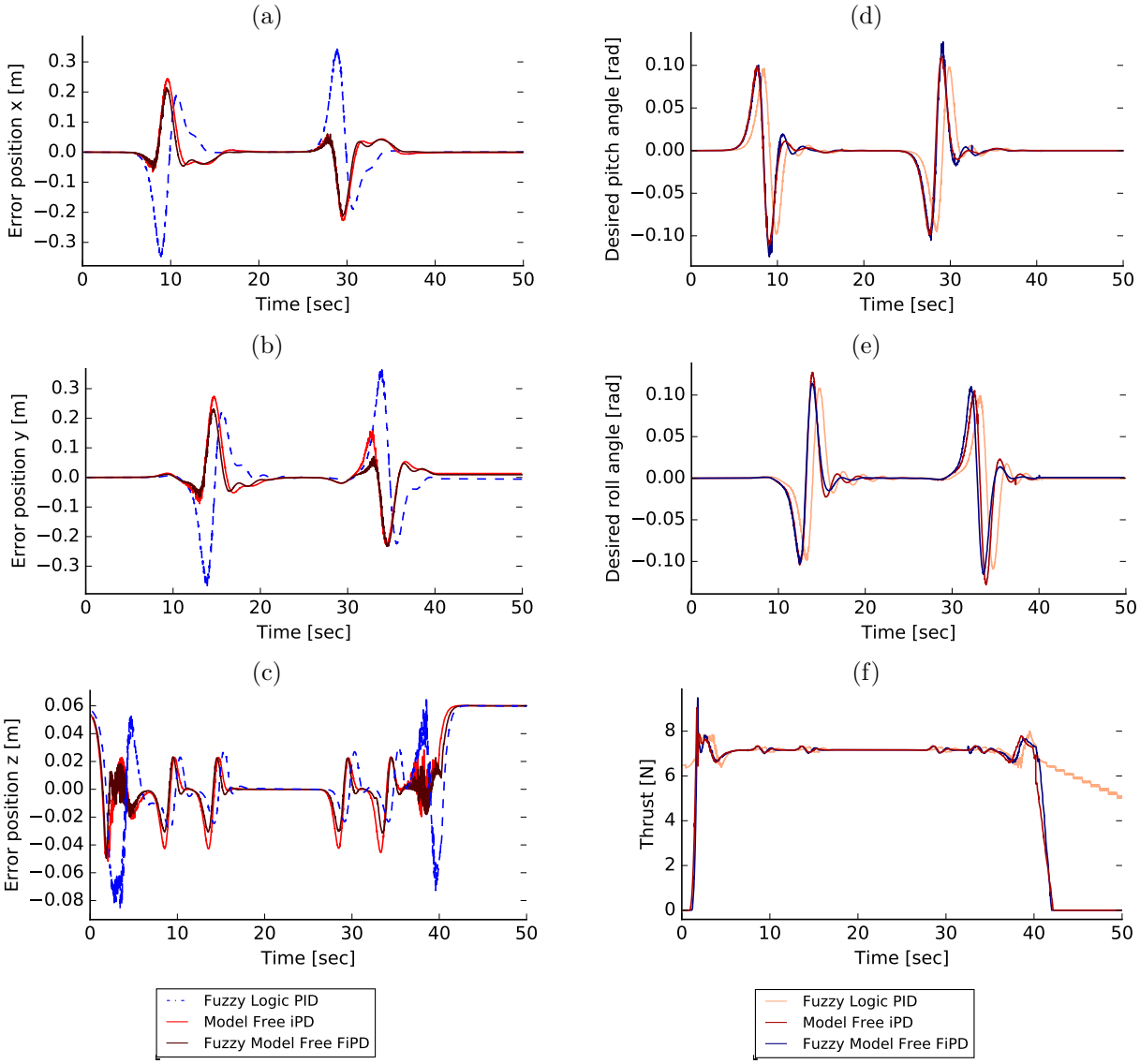


Figure 9 Basic scenario results

Table 5 In-depth comparison

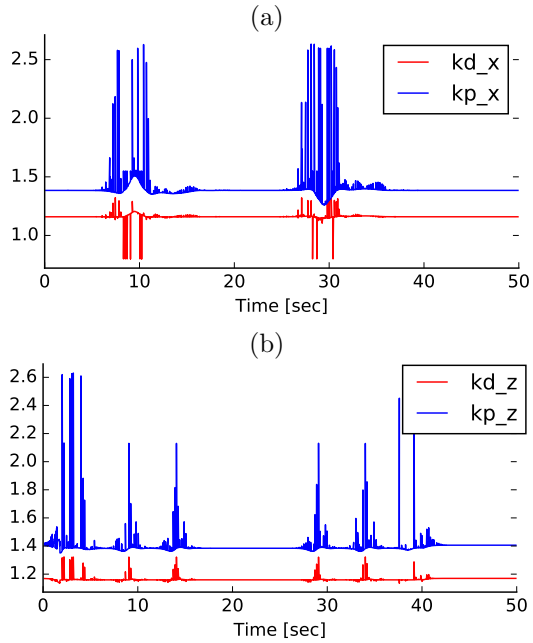
Type		Fuzzy PID	iPD	FiPD
Basic scenario	ISE	0.7969	0.3616	0.2653
	ISCI	2383	2031	2025
Parameters uncertainties	ISE	1.0318	0.3321	0.2475
	ISCI	4853	4210	4140
Payload	ISE	1.0066	0.3955	0.2660
	ISCI	4727	4118	4040
Wind disturbance	ISE	1.8811	0.5410	0.3517
	ISCI	2375	2039	2024

Table 5 summarizes the overall quantified metrics.

According to the results from the basic scenario, the proposed strategy has a small tracking error (ISE 0.2653), and low energy consumption (ISCI 2025), followed by model free control, which has an acceptable tracking error (ISE 0.3616), and a moderate energy consumption (ISCI 2031). The second baseline, Fuzzy Logic PID, was the least accurate technique (ISE 0.7969), and it has the highest energy consumption (ISCI 2383). Furthermore, the results under parameters uncertainties scenario show that the tracking error is almost the same as the basic scenario in the proposed technique as well as in model free control. The fuzzy logic PID, however, delivers a poor tracking accuracy. These results provide additional support for the capability of tolerating parameters uncertainties in the proposed control technique. In addition, in the presence of extra payload, the ISE in Fuzzy model free control is close to

the ISE value of the basic scenario. Whereas, a lack of accuracy is observed (higher ISE) in model free control. In contrast, there is a major increment of tracking error in Fuzzy Logic PID. As a result, these outcomes of Fuzzy model free control highlight the robustness level to mass variation. Lastly, the results under wind disturbance, show a small increment in the tracking error in Fuzzy model free control, and a considerable increase in the error for model free control. However, a loss of accuracy in Fuzzy Logic PID is observed. These results provide compelling evidence for the disturbance rejection ability offered by the proposed strategy. Overall, the accuracy of the proposed technique FiPD is roughly the same for all scenarios. On the other hand, it is worthy mentioning that the Fuzzy model free control technique is not sensitive to the disturbance nature. As a conclusion, Table 5 indicates the outstanding performance and high robustness level for the proposed strategy against different kinds of disturbances .

Figure 10 illustrates the variation of the controller gains over time. Specifically, Figure 10(a), represents the gains adaptation for both controllers in the translation along x and y motion. The other Figure 10(b) shows the altitude controller gains. The adaptive behavior during the control process resulting better performance compared to static gains as observed.



b

Figure 10 Controller gains

5.3 Agressive trajectory

The last experiment was conducted using an aggressive trajectory to test the effectiveness of the proposed strategy in case of an aggressive motion. The expression of the aggressive trajectory can be found in the Appendix. Figure 11(a) shows the 3D tracking with the 2D Figure 11(b) in the $x-y$ plane. Furthermore, in the first case, the environment was assumed ideal (nominal case), no external disturbances occurred. The results depicted in 11(basic scenario)(a-b) indicate that the FiPD controller has the best performance compared to iPD and Fuzzy Logic PID controller. In the second case, the previous wind disturbances were introduced into the environment. Figure 11(wind disturbance)(a) shows the result with the 2D Figure 11(wind disturbance)(b) in the $x-y$ plane. It can be observed that the FiPD technique shows satisfactory results with respect to other techniques.

6 Conclusion

This paper presented a new Fuzzy model free control approach (FMFC) for Unmanned Aerial Vehicles (UAVs). The purpose of the controller design is to ensure the tracking of a predefined trajectory, without well-knowledge about the dynamics model and physical system parameters.

In addition, a comprehensive assessment of control performance was conducted in the context of various realistic scenarios to test and demonstrate the tracking capability of the proposed strategy with respect to conventional model free control iPD and Fuzzy Logic PID. The proposed approach FiPD controller shows superior performance among other techniques.

The performance exhibited by the proposed control approach based on model free control depends on the good estimation of the ultra-local model. Furthermore, tuning the gains parameters during the control process increases the robustness level. However, it can affect the estimation of the ultra-local model due to the time-delay required to compute the appropriate gains. It should be noted that, the complexity of the algorithms that compute on-line the controller gains is significant to avoid the time-delay and maintain the good estimation.

As future work, and given the lower complexity and better performance of the proposed strategy, the experiments would be conducted on a real quadrotor. In addition, we aim to develop other control methods such as non-linear model free control. By comparing them, we will carefully select the best one based on several performance-based aspects such as disturbance sensitivity, robustness, power consumption, and so on.

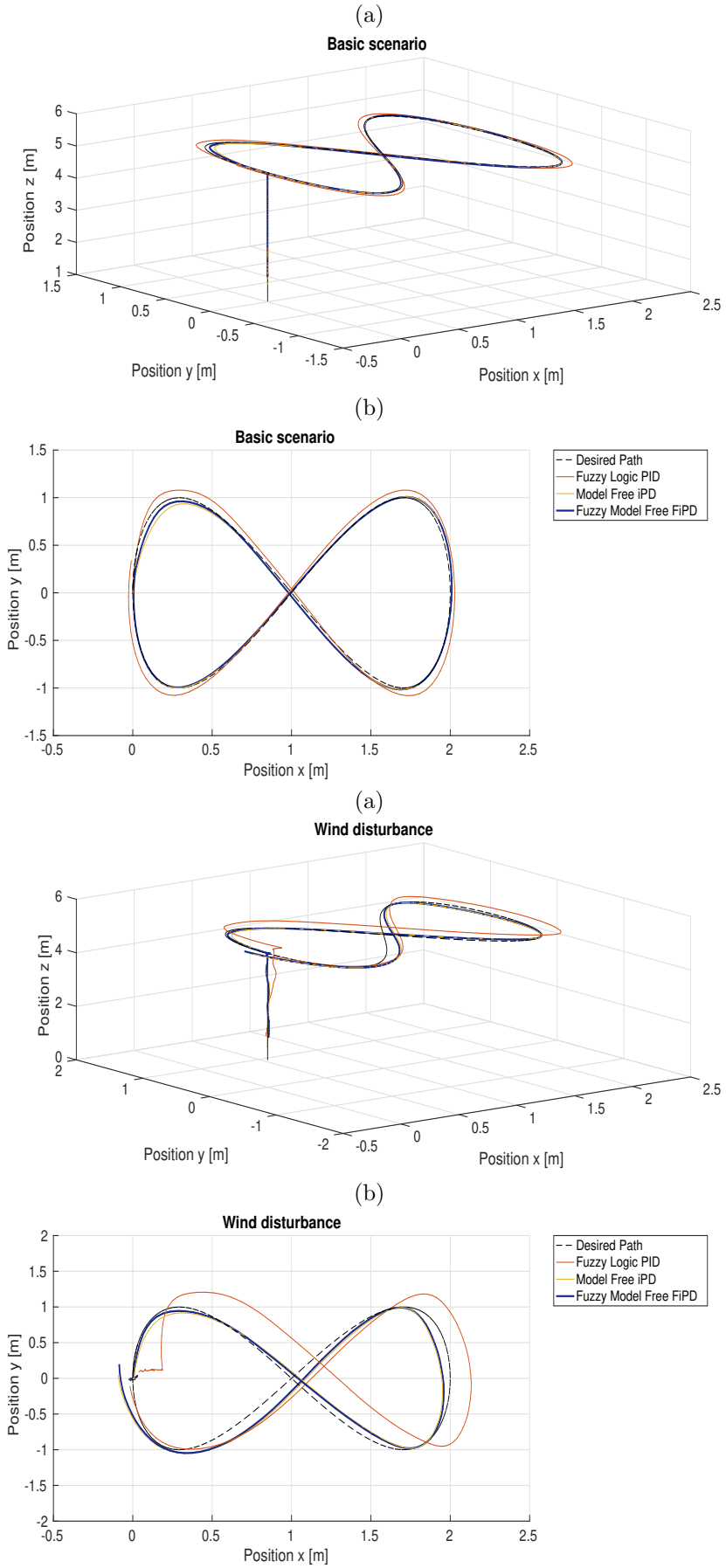


Figure 11 Agressive trajectory response

Appendix

The square trajectory can be expressed as follows :

$$x_d(t) = \begin{cases} 2\frac{(t-10)^5}{(t-10)^5+(5-t+10)} & when\ 5s \leq t \leq 20s \\ 2 - 2\frac{(t-40)^5}{(t-40)^5+(5-t+40)} & when\ 20s \leq t \leq 35s \end{cases}$$

$$\mathrm{y}_d(t) = \begin{cases} 2\frac{(t-15)^5}{(t-15)^5+(5-t+15)} & when\ 5s \leq t \leq 25s \\ 2 - 2\frac{(t-40)^5}{(t-40)^5+(5-t+40)} & when\ 25s \leq t \leq 40s \end{cases}$$

$$\mathrm{z}_d(t) = \begin{cases} 2\frac{(t-35)^5}{(t-35)^5+(5-t+35)} & when\ 0s \leq t \leq 10s \\ 2 & when\ 10s \leq t \leq 35s \\ 2 - 2\frac{(t-40)^5}{(t-40)^5+(5-t+40)} & when\ 35s \leq t \leq 50s \end{cases}$$

The aggressive trajectory is given as follows:

$$\begin{cases} x_d = 0 \\ y_d = 0 \\ z_d = \sin(2\pi/60)t \end{cases} \quad (t < 15s)$$

$$\begin{cases} x_d = 0 \\ y_d = 0 \\ z_d = 5 \end{cases} \quad (15s \leq t < 20s)$$

$$\begin{cases} x_d = 1 - \cos(0.25(t-20)) \\ y_d = \sin(0.5(t-20)) \\ z_d = 5 \end{cases} \quad (t \geq s)$$

Acknowledgment

The authors would like to thank their colleagues Issam Chekakta, Nabil Rezoug and Mustapha Bouhali for their useful comments, discussions and valuable insights.

References

[1] B. Benjdira, T. Khursheed, A. Koubaa, A. Ammar, and K. Ouni, "Car detection using unmanned aerial vehicles: Comparison between faster r-cnn and yolov3," in *2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS)*, pp. 1–6, Feb 2019.

[2] A. Koubaa and B. Qureshi, "Dronetrack: Cloud-based real-time object tracking using unmanned aerial vehicles over the internet," *IEEE Access*, vol. 6, pp. 13810–13824, 2018.

[3] A. Koubaa, B. Qureshi, M.-F. Sriti, A. Allouch, Y. Javed, M. Alajlan, O. Cheikhrouhou, M. Khalgui, and E. Tovar, "Dronemap planner: A service-oriented cloud-based management system for the internet-of-drones," *Ad Hoc Networks*, vol. 86, pp. 46 – 62, 2019.

[4] E. T. Alotaibi, S. S. Alqefari, and A. Koubaa, "Lsar: Multi-uav collaboration for search and rescue missions," *IEEE Access*, vol. 7, pp. 55817–55832, 2019.

[5] B. Benjdira, Y. Bazi, A. Koubaa, and K. Ouni, "Unsupervised domain adaptation using generative adversarial networks for semantic segmentation of aerial images," *Remote Sensing*, vol. 11, no. 11, 2019.

[6] S. Bouabdallah, A. Noth, and R. Siegwart, "PID vs LQ control techniques applied to an indoor micro quadrotor," *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, pp. 2451–2456, 2004.

[7] A. Mokhtaril, A. Benallegue, and B. Daachi, "Robust feedback linearization and GH controller for a quadrotor unmanned aerial vehicle," *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, vol. 57, no. 1, pp. 1009–1014, 2005.

[8] a. Benallegue, A. Mokhtari, and L. Fridman, "Feedback linearization and high order sliding mode observer for a quadrotor UAV," *International Workshop on Variable Structure Systems, (VSS'06)*, pp. 365–372, 2006.

[9] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro Quadrotor," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2005, no. April, pp. 2247–2252, 2005.

[10] T. Lee and M. Leok, "Geometric Tracking Control of a Quadrotor UAV on SE (3) for Extreme Maneuverability," *Proc. IFAC World Congress*, no. 3, 2010.

[11] H. Lee and H. J. Kim, "Trajectory tracking control of multirotors from modelling to experiments: A survey," *International Journal of Control, Automation and Systems*, vol. 15, no. 1, pp. 281–292, 2017.

[12] M. Fliess and C. Join, "Model-free control," *International Journal of Control*, vol. 86, no. 12, pp. 2228–2252, 2013.

[13] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, *Robot Operating System (ROS): The Complete Reference (Volume 1)*, ch. RotorS—A Modular

Gazebo MAV Simulator Framework, pp. 595–625. Cham: Springer International Publishing, 2016.

[14] M. Fliess, C. Join, M. Mboup, and H. Sira-Ramirez, "Vers une commande multivariable sans mod\ele," *arXiv preprint math/0603155*, 2006.

[15] R. Bourdais, M. Fliess, and W. PERRUQUETTI, "Towards a model-free output tracking of switched nonlinear systems," *IFAC Proceedings Volumes*, vol. 40, no. 12, pp. 504–509, 2007.

[16] A. Chemori, "Model-free control of the inertia wheel inverted pendulum with real-time experiments," 2017.

[17] H. Abouaïssa and S. Chouraqui, "On the control of robot manipulator: A model-free approach," *Journal of Computational Science*, vol. 31, no. 5, pp. 6–16, 2019.

[18] M. Haddar, R. Chaari, S. C. Baslamisli, F. Chaari, and M. Haddar, "Intelligent PD controller design for active suspension system based on robust model-free control strategy," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 0, no. 0, p. 095440621983644, 2019.

[19] Y. Al Younes, A. Drak, H. Noura, A. Rabhi, and A. El Hajjaji, "Model-free control of a quadrotor vehicle," *2014 International Conference on Unmanned Aircraft Systems, ICUAS 2014 - Conference Proceedings*, no. May, pp. 1126–1131, 2014.

[20] Y. A. Younes, A. Drak, H. Noura, A. Rabhi, and A. E. Hajjaji, "Robust Model-Free Control Applied to a Quadrotor UAV," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 84, no. 1-4, pp. 37–52, 2016.

[21] H. Wang, X. Ye, Y. Tian, G. Zheng, and N. Christov, "Model-free-based terminal SMC of quadrotor attitude and position," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 5, pp. 2519–2528, 2016.

[22] Y. BOUZID, H. SIGUERDIDJANE, and Y. BESTAOUI, "Energy based 3D trajectory tracking control of quadrotors with model-free based on-line disturbance compensation," *Chinese Journal of Aeronautics*, vol. 31, no. 7, pp. 1568–1578, 2018.

[23] Z. Li, X. Ma, and Y. Li, "Model-free control of a quadrotor using adaptive proportional derivative-sliding mode control and robust integral of the signum of the error," *International Journal of Advanced Robotic Systems*, vol. 15, no. 5, pp. 1–15, 2018.

[24] M. Bekcheva, C. Join, and H. Mounier, "Cascaded Model-Free Control for trajectory tracking of quadrotors," *2018 International Conference on Unmanned Aircraft Systems, ICUAS 2018*, pp. 1359–1368, 2018.

[25] J. Wang, *Quadrotor analysis and model free control with comparisons*. PhD thesis, Université Paris Sud-Paris XI, 2013.

[26] Z. Chekakta, M. Zerikat, Y. Bouzid, and M. Abderrahim, "Model-free control applied for position control of quadrotor using ros," in *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*, pp. 1260–1265, IEEE, 2019.

[27] M. Kamel, M. Burri, and R. Siegwart, "Linear vs Nonlinear MPC for Trajectory Tracking Applied to Rotary Wing Micro Aerial Vehicles," *ArXiv e-prints*, Nov. 2016.