Adaptive PID gain-scheduling for 3D smart quadrotor flight controller*

[1]Wassim Arfa, [2]Chiraz Ben Jabeur, [3]Yassine Faleh, [3]Hassene Seddik

[1]Electrical Engineering of Carthage National School of Engineers, Tunis, Tunisia; [2]Electrical Engineering of Higher Institute of Computer Science, Ariana, Tunsisia ; [2]Electrical Engineering of National High School of Engineers of Tunis, Tunis, Tunisia
Email: wassimarfa3@gmail.com

ABSTRACT

In this paper, we present a performance evaluation of PID controller gains for angle control of drones. The primary objective is to optimize the PID gains to enhance the performance of the drone. The proposed solution is named Adaptive PID flight controller for controlling the altitude dynamics of a UAV. This approach is based on three comparisons: Firstly, we compare the use of a single PID controller for all three angles. Secondly, we explore the option of using two PID controllers for the three angles, where the first controller is designed to control the Pitch and Roll angles, while the second controller is dedicated to the Yaw angle. Finally, we use three PID controllers for each angle (Pitch, Roll and Yaw).

Our ultimate aim is to identify the most effective PID controller configuration that optimizes drone angle control, leading to improved stability, responsiveness, and accuracy during flight.

Keywords : PID controller, drone, angle control, optimization, simulation, stability.

1 INTRODUCTION

Quadcopters have gained significant interest among researchers due to their ability to perform tasks that would otherwise be hazardous for humans. These unmanned aerial vehicles (UAVs) are equipped with four rotors arranged in a cross-like configuration, enabling them to hover, take off, land, and maneuver effectively. The versatility and maneuverability of quadcopters have led to their widespread use in various industries.

Quadcopters are no longer limited to military applications; they are now widely utilized in civilian, research, and commercial domains. Their versatility and maneuverability make them valuable tools in various industries [1]. Currently, quadcopter Unmanned Aerial Vehicles are used not only for military missions but also in civilian, research, and commercial domains [2].

It may include inspection of nuclear reactors, fire safety surveillance, inspection of power lines, law enforcement agencies, and also investigations in military and agricultural services.

Precising angle control is essential for the stable flight and accurate manoeuvring of drones in various applications. PID (Proportional-Integral-Derivative) controllers have been widely used for angle control due to their simplicity and effectiveness. However, the performance of PID controllers heavily relies on the proper selection and tuning of their gains.

Previous research by Ashfaq Ahmad Mianet (2008) proposed a nonlinear model and control algorithm for a 6-degree-of-freedom (DOF) quadcopter [3]. The model was derived using the Newton-Euler protocol, incorporating electric motors' aerodynamic coefficients and dynamics. However, this research faced challenges such as overshoot and time delays in reaching the prescribed value. Jun Li (2011) also presented a model to examine a quadcopter's dynamic response and PID control algorithm [3].

In our study, we propose a solution based on using three PID controllers, each one is dedicated to control a specific angle. We will compare this approach with two other methods. The first method involves using a single PID controller to control all three angles simultaneously. The second method utilizes two PID controllers, with one controller dedicated to controlling the Pitch and Roll angles, while the other controls the Yaw angle.

Through a comparative simulation study, we aim to evaluate and compare the performance of these different PID controller configurations in terms of stability, tracking accuracy, and disturbance rejection. By conducting simulations under noise-free conditions, we can isolate the effects of the PID controllers and determine the optimal PID gains for each angle.

Throughout our evaluation, we apply various comparative techniques such as Genetic Algorithms (GA) [4], the Crow Search Algorithm (CSA) [5], Particle Swarm Optimization (PSO) [6] and Ziegler-Nichols (ZN) to assess the performance of each control strategy. However, to ensure a rigorous analysis, the specific details of these comparative techniques will be elaborated upon in the paper.

The findings of this study will provide valuable insights into the selection and tuning of PID gains for optimal angle control in drone applications. This research contributes to enhancing the stability and manoeuvrability of drones by improving the control system's performance. Ultimately, it enables the utilization of drones in a wide range of applications that require precise angle control.

The rest of the paper is organized as follows:

Section 2 will focus on the mathematical model according to Newton-Euler.

Section 3, will delve into optimal PID gains identification, which plays a crucial role in controlling and stabilizing the drone.

Section 4, will focus on the simulation and optimization of the PID gains in the drone. Precise tuning of the PID controller parameters is crucial for achieving optimal performance in terms of stability, responsiveness, and control accuracy. The main conclusions and discussion are presented in Section 5.

2 Mathematical model according to Newton-Euler

This section provides a general overview of the quadcopter used in this paper, the mathematical model of the UAV and the control structure will then be presented.

The main configuration is described in ‘ **Error! Reference source not found.** ’ the motors are numbered clockwise, with motor 1 being the one at the front of the device relative to the reference frame F_b [7].

Motors 1 and 3 rotate clockwise, unlike motors 2 and 4 [8].

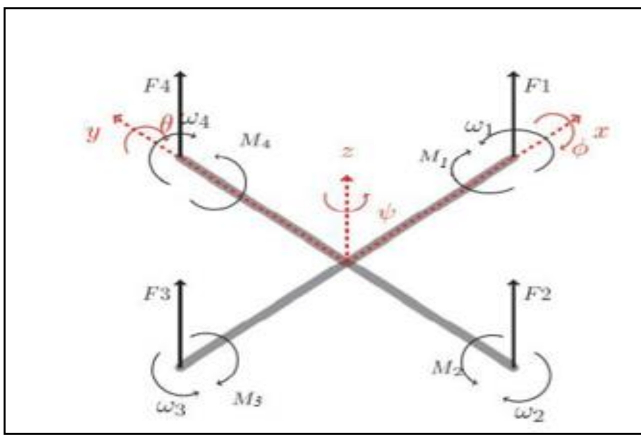

Fig. 1. Identification of the direction of rotation of the rotors [8]

Based on the equations of forces applied to the quadcopter and the moments acting on the quadrotor, the equations describing the complete model of the Quadrotor using the Newton-Euler formulation and the dynamic system model are as follows [9]:

$$\begin{cases} \dot{\zeta} = v \\ m\ddot{\zeta} = F_f + F_t + F_g \\ R = \dot{R}S(\Omega) \\ J\Omega = -\Omega \wedge J\dot{\Omega} + M_f - M_a - M_{gh} \end{cases} \quad (1)$$

With:

ζ : The vector representing the position of the quadrotor

m: The total mass of the quadrotor

Ω : The angular velocity expressed in the fixed reference frame

R: The rotation matrix

$\wedge$: The vector product

The gyromagnetic moment of the propellers is given by the following equation:

$$M_{gh} = \sum_1^4 \Omega \wedge J_r [0\ 0\ (-1)^{i+1} w_i]^T \quad (2)$$

With

J_r : denotes the inertia of the rotors and w_i is the angular velocity of the ith rotor.

J: The symmetric inertia matrix of dimension (3x3) is given by:

$$J=\begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix} \quad (3)$$

S (Ω) The antisymmetric matrix; For a velocity vector, $\Omega = [\Omega_1, \Omega_2, \Omega_3]$ it can be represented as:

$$S(\Omega) = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix} \quad (4)$$

Where The rotation speeds $\Omega_1, \Omega_2, \Omega_3$ in the fixed frame of reference, are expressed in terms of the rotation speeds $\dot{\phi}\ \dot{\theta}\ \dot{\psi}$ in the moving frame of reference

F_f The total force generated by the four rotors can be represented by the following equation:

$$F_f = R.[0\ 0\ \textstyle\sum_1^4 F_i]^T \quad (5)$$

With:

$$F_i = b.w_i^2 \quad (6)$$

With b: The coefficient of lift depends on the shape and number of blades and the air density.

i =1…4.

F_t Drag force along the axes $(\vec{x}, \vec{y}, \vec{z})$:

$$F_f = \begin{bmatrix} -K_{ftx} & 0 & 0 \\ 0 & -K_{fty} & 0 \\ 0 & 0 & -K_{ftz} \end{bmatrix} \quad (7)$$

$K_{ftx}, K_{fty}, K_{ftz}$ The coefficients of translational

F_g The force of gravity is given by:

$$F_g = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} \quad (8)$$

M_f The moment caused by the thrust and drag forces is:

$$M_f = \begin{bmatrix} l(F_4 - F_2) \\ l(F_3 - F_1) \\ ld(w_1^2 - w_2^2 + w_3^2 - w_4^2) \end{bmatrix} \quad (9)$$

With

l : the length of the arm between the rotor and the center of gravity of the quadcopter

M_a The moment resulting from aerodynamic friction, also known as aerodynamic drag:

$$M_a = \begin{bmatrix} K_{fax}\dot{\phi}^2 \\ K_{fay}\dot{\theta}^2 \\ K_{faz}\dot{\psi}^2 \end{bmatrix} \quad (10)$$

$K_{fax}, K_{fay}, K_{faz}$. The coefficients of aerodynamic friction

2.1 Equations of translational motion for drone control

After presenting the force equations in the previous sections, we can now proceed to the complete model of the Quad-rotor by utilizing Newton's second law for linear motion. The formula is as follows.

$$m\ddot{\zeta} = F_t + F_f + F_g \quad (11)$$

we replace each force with its corresponding formula, we obtain:

$$m\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} cos\phi cos\psi sin\theta + sin\phi sin\psi \\ cos\phi sin\psi sin\theta - sin\phi cos\psi \\ cos\phi cos\theta \end{bmatrix} \sum_1^4 F_i - \begin{bmatrix} K_{ftx}\dot{x} \\ K_{fty}\dot{y} \\ K_{ftz}\dot{z} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix}$$

We obtain the differential equations that define the translational

$$\begin{cases} \ddot{x} = \frac{1}{m}\,(\cos\phi\,\cos\psi\,\sin\theta + \sin\phi\sin\psi)(\sum_1^4 F_i) - \frac{K_{ftx}}{m}\dot{x} \\ \ddot{y} = \frac{1}{m}\,(\cos\phi\,\sin\psi\,\cos\theta + \sin\phi\cos\psi)(\sum_1^4 F_i) - \frac{K_{fty}}{m}\dot{y} \\ \ddot{z} = \frac{1}{m}\,(\cos\phi\,\cos\theta)(\sum_1^4 F_i) - \frac{K_{ftz}}{m}\dot{z} - g \end{cases} \quad (13)$$

2.2 Equations of rotational motion for drone control

Applying the same principle of Newton for the case of rotation, we find the following formula

$$J\dot{\Omega} = -\Omega \wedge J\,\Omega + M_f - M_a - M_{gh} \qquad (14)$$

When we replace each moment with its corresponding expression

$$\begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = -\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \wedge \left(\begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}\right) - \begin{bmatrix} J_r\bar{\Omega}_r\dot{\theta} \\ -J_r\bar{\Omega}_r\dot{\theta} \\ 0 \end{bmatrix} - \begin{bmatrix} K_{fax}\dot{\phi}^2 \\ K_{fay}\dot{\theta}^2 \\ K_{faz}\dot{\psi}^2 \end{bmatrix} + \begin{bmatrix} lb(w_4^2 - w_2^2) \\ lb(w_3^2 - w_1^2) \\ ld(w_1^2 - w_2^2 + w_3^2 - w_4^2) \end{bmatrix}$$

We then obtain the differential equations defining the rotational motion:

$$\begin{cases} I_x\ddot{\phi} = -\dot{\theta}\dot{\psi}\,(I_z - I_y) - J_r\bar{\Omega}_r\dot{\theta} - K_{fax}\dot{\phi}^2 + lb(w_4^2 - w_2^2) \\ I_y\ddot{\theta} = \dot{\phi}\dot{\psi}\,(I_z - I_x) - J_r\bar{\Omega}_r\dot{\theta} - K_{fax}\dot{\theta}^2 + lb(w_3^2 - w_1^2) \\ I_z\ddot{\psi} = \dot{\phi}\dot{\theta}\,(I_y - I_x) - K_{faz}\dot{\psi}^2 + ld(w_1^2 - w_2^2 + w_3^2 - w_4^2) \end{cases} \quad (16)$$

With:

$$\bar{\Omega}_r = w_1 - w_2 + w_3 - w_4 \qquad (17)$$

As a result, the complete dynamic model governing the quadrotor is given by the following system of equations:

$$\begin{cases} \ddot{\phi} = -\dot{\theta}\dot{\psi}\frac{I_z - I_y}{I_x} - \frac{J_r}{I_x}\bar{\Omega}_r\dot{\theta} - \frac{K_{fax}}{I_x}\dot{\phi}^2 + \frac{lb}{I_x}(w_4^2 - w_2^2) \\ \ddot{\theta} = \dot{\phi}\dot{\psi}\frac{I_z - I_x}{I_y} - \frac{J_r}{I_y}\bar{\Omega}_r\dot{\theta} - \frac{K_{fay}}{I_y}\dot{\theta}^2 + \frac{lb}{I_y}(w_3^2 - w_1^2) \\ \ddot{\psi} = \dot{\phi}\dot{\theta}\frac{I_y - I_x}{I_z} - \frac{K_{faz}}{I_z}\dot{\psi}^2 + \frac{ld}{I_z}(w_1^2 - w_2^2 + w_3^2 - w_4^2) \\ \ddot{x} = \frac{1}{m}\,(\cos\phi\,\cos\psi\,\sin\theta + \sin\phi\sin\psi)(\sum_1^4 F_i) - \frac{K_{ftx}}{m}\dot{x} \\ \ddot{y} = \frac{1}{m}\,(\cos\phi\,\sin\psi\,\cos\theta + \sin\phi\cos\psi)(\sum_1^4 F_i) - \frac{K_{fty}}{m}\dot{y} \\ \ddot{z} = \frac{1}{m}\,(\cos\phi\,\cos\theta)(\sum_1^4 F_i) - \frac{K_{ftz}}{m}\dot{z} - g \end{cases} \quad (18)$$

2.3 Modelling the Propeller-Thrust Relationship in Drone Control

We can calculate the motor speeds from the forces and moments applied to the Quad-rotor. This relationship is crucial for controller implementation. Therefore, we can rewrite the equations in matrix form as follows:

$$\begin{bmatrix} F \\ M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -bl & 0 & bl \\ -bl & 0 & bl & 0 \\ d & -d & d & -d \end{bmatrix}\begin{bmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{bmatrix} \qquad (19)$$

With d is the coefficient of drag depends on the design of the propeller.

By inverting the matrix, we obtain the relationship between the motor speeds:

$$\begin{bmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{4b} & 0 & \frac{1}{2bl} & -\frac{1}{4b} \\ \frac{1}{4b} & -\frac{1}{2bl} & 0 & \frac{1}{4b} \\ \frac{1}{4b} & 0 & -\frac{1}{2bl} & -\frac{1}{4b} \\ \frac{1}{4b} & \frac{1}{2bl} & 0 & \frac{1}{4b} \end{bmatrix}\begin{bmatrix} F \\ M_x \\ M_y \\ M_z \end{bmatrix} \qquad (20)$$

3 Optimal PID gains identification

This section presents the computation of gains for the PID controller for the UAV. Initially, the optimization of a single PID controller is discussed, followed by the optimization of the two PID controllers, and finally, the proposed solution of three PID controllers is presented. The proposed algorithm is described at the end of this section.

Existing controllers are typically designed to optimize trajectory tracking performance while maintaining stability. However, maximizing the performance can result in higher energy consumption, which reduces battery lifespan. The hidden cost of high controller performance needs to be considered in the development of future controllers. Factors such as vehicle safety, reliability in minimizing position/velocity tracking errors and maintenance costs should also be considered.

The objective of this work goes beyond the conventional application of PID control to Quad-rotor dynamics. The aim is to explore approaches that can further minimize control error, directly influencing the reduction of battery energy consumption and increasing the drone's autonomy. The following section will present different strategies to minimize trajectory error.

The proposed method involves manually refining the PID gains for each angle of the drone through an iterative process. We initially adjust the gain values by a factor of 1 and then We refine them by a factor of (0.1), based on the analysis of the drone's performance.

The process starts by setting initial values for the PID gains of each angle. Flight tests are then conducted using these initial gains, and performance data are collected to evaluate the drone's stability, accuracy, and responsiveness to altitude commands.

By analyzing the results, we manually adjust the PID gain values, either increasing or decreasing them based on the observed performance. We iterate this process, initially refining the gains by a factor of 1, and if further improvements are needed, we perform additional refinement using a factor of (0.1).

This iterative and manual refinement process allows us to optimize the control performance of the drone's altitude dynamics by precisely adjusting the gains for each angle. The goal is to achieve enhanced stability, improved accuracy, and quicker response to altitude commands.

The developed method of manually refining the PID gains for each angle of the drone involves an iterative process. By adjusting the gains step by step, we enhance the drone's stability and accuracy, enabling it to respond more effectively to variations in altitude commands.

In this section, we discuss the PID (Proportional-Integral-Derivative) controller, focusing on simplified models. The main objective is to design an adaptive PID controller for the flight of a Quad-rotor drone. The controller utilizes a control input, denoted as u, to regulate the position and angle of the drone concerning a reference input [10].

The PID control law consists of three basic feedback control actions: proportional, integral, and derivative. The related gains are denoted as K_p , K_p and K_d. The mathematical representation of the PID controller is as follows:

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{d}{dt} e(t) \qquad (21)$$

with:

K_p The proportional gain, K_i The integral gain and K_d The derivative gain

can be formulated as a function of the error:

$$e(t) = s_p - p_v(t) \qquad (22)$$

Where:

s_p is the setpoint or desired position and $p_v(t)$ is the process variable at the instantaneous moment according to s_p

A high-quality controller should be capable of establishing a desired position in which the yaw, pitch, and roll angles remain constant and stable [11].

1. This can be achieved by using the Pythagorean theorem and applying the following assumptions and cancellations:
2. The Quad-rotor is considered a rigid body with constant mass and symmetric structure;
3. The inertia matrix (I) of the vehicle is very small and negligible;
4. The center of gravity and the center of mass coincide;

The thrust is proportional to the square of the propeller's velocity. Based on the above assumption and considering the drone as a point mass, whose rotation angles can be identified using the desired position. The desired angles (Roll, Pitch and Yaw) can be extracted from the following expressions:

$$\begin{cases} \phi_d = tan^{-1}(\frac{Z_d}{Y_d}) \\ \theta_d = sin^{-1}\left(\frac{X_d}{\sqrt{Z_d^2+X_d^2}}\right) \\ \psi_d = cos^{-1}(\frac{Y_d}{\sqrt{X_d^2+Y_d^2+Z_d^2}}) \end{cases} \qquad (23)$$

From the geometric modelling of our model, the inputs required to control the spatial localization ($\vec{X}$, $\vec{Y}$, $\vec{Z}$) are given in the following expressions [12]:

$$\begin{cases} u_x = K_p(X_d - X) + K_i \int_0^t (X_d - X) + K_d \frac{d(X_d - X)}{dt} \\ u_y = K_p(Y_d - Y) + K_i \int_0^t (Y_d - Y) + K_d \frac{d(Y_d - Y)}{dt} \\ u_z = K_p(Z_d - Z) + K_i \int_0^t (Z_d - Z) + K_d \frac{d(Z_d - Z)}{dt} \end{cases} \qquad (24)$$

K_p, $K_i$et K_d are the gains of the PID controller for each coordinate position. The orientation angles are controlled as described in the following equations:

$$\begin{cases} u_\phi = K_{pa}(\phi_d - \phi) + K_{ia} \int_0^t (\phi_d - \phi) + K_{da} \frac{d(\phi_d - \phi)}{dt} \\ u_\theta = K_{pa}(\theta_d - \theta) + K_{ia} \int_0^t (\theta_d - \theta) + K_{da} \frac{d(\theta_d - \theta)}{dt} \\ u_\psi = K_{pa}(\psi_d - \psi) + K_{ia} \int_0^t (\psi_d - \psi) + K_{da} \frac{d(\psi_d - \psi)}{dt} \end{cases} \qquad (25)$$

K_{pa}, K_{ia}et K_{da} are the parameters of the PID controller for controlling the roll, pitch, and yaw angles [13].

3.1 A single PID controller optimization

'**Error! Reference source not found.**' shows the PID Controller block receives a setpoint value corresponding to the desired goal for the drone, such as altitude, velocity, or position. It compares this setpoint to a current measurement (error) of the drone and generates an output command based on the PID's proportional, integral, and derivative terms.

The output of the PID Controller is then sent to the Drone Controller, which interprets the command to adjust various control variables of the drone, such as motor speed, tilt angles.

Finally, the output of the Drone Controller is transmitted to the drone sensors which make the necessary adjustments to achieve the defined goal.

The PID controller uses the proportional term to directly respond to the current error, the integral term to accumulate past errors and correct persistent errors, and the derivative term to anticipate future errors and react accordingly. This allows for a gradual adjustment of the output command to reach and maintain the desired setpoint [14].

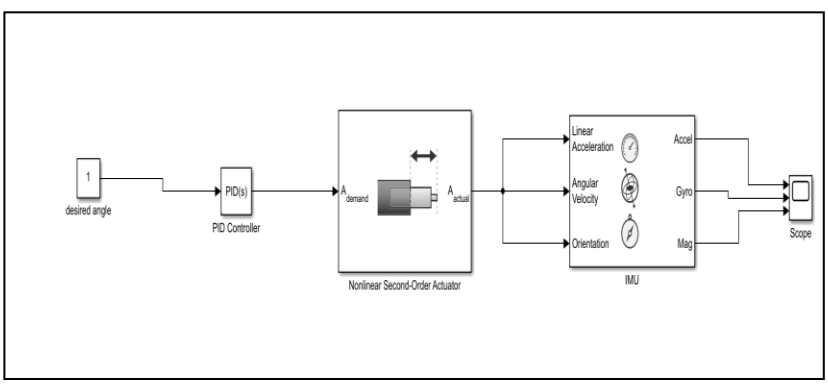

Fig. 2. Control system for a single PID

To simplify the control system and reduce complexity, a single PID controller can be utilized to regulate all three angles of the drone. This integrated approach enables coordinated drone orientation control along multiple axes.

The PID controller for the three angles combines the errors between the desired angles and the current angles of the drone.

By using a single PID controller for the three angles, the control system achieves synchronized control of the drone's orientation. This coordinated control enables smooth transitions between different manoeuvres and enhances overall stability during flight. Additionally, it simplifies the tuning process, as a single set of gains can be adjusted to optimize the performance of all three angles simultaneously.

It is important to emphasize that selecting appropriate PID gains for the single controller is crucial in achieving the desired control performance. The gains should be carefully tuned to strike a balance between stability and responsiveness. This ensures accurate tracking of the desired angles while minimizing oscillations or overshoots in the control system's response.

3.2 A two PID controllers' architecture

As we can observe in '**Error! Reference source not found.**' the control system for the drone consists of two PID blocks: PID 1 and PID 2, responsible for controlling different aspects of the drone's movements.

PID 1 is specifically designed to control the roll and pitch of the drone. It receives a setpoint value that represents the desired angle for roll and pitch. The setpoint is compared to the current measurement of the drone's roll and pitch angles (error). Based on this error, PID1 generates an output command that is then sent to the drone controller. The drone controller interprets this command and adjusts the appropriate motor to modify the roll and pitch angles of the drone.

PID 2, on the other hand, focuses on controlling the yaw or heading of the drone. Similar to PID 1, PID 2 receives a setpoint value that represents the desired angle. It compares it to the current measurement of the drone's yaw angle (error) and then generates an output command. This command is then sent to the drone controller, which adjusts the relevant motors to enable rotation around the vertical axis, thereby controlling the drone's yaw.

By working in tandem, these two PID controllers regulate the drone's movements in all three axes (roll, pitch, and yaw). Each PID controller utilizes proportional, integral, and derivative control to adjust the output command and maintain the drone in the desired position.

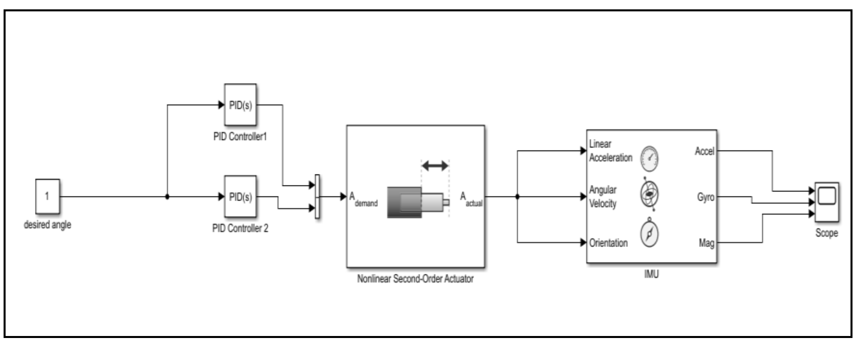

Fig. 3. Control system with two PID controllers

By using these PID controllers, the drone can precisely control its theta and phi couple angles, ensuring stable flight and accurate maneuverability. The separate PID controller for psi allows independent control of the drone's yaw motion, enabling efficient changes in direction and orientation.

The adjustment of PID gains is crucial to optimize the performance of each controller, ensuring a balance between stability, responsiveness, and tracking accuracy.

The selection and fine-tuning of appropriate gains are influenced by factors such as the drone's dynamics, desired control performance, and external conditions such as wind disturbances.

3.3 A three PID controllers' architecture

In '**Error! Reference source not found.**' the first PID controller is dedicated to controlling the roll angle. It compares the desired roll angle (setpoint) value with the current measurement of the drone's roll angle, calculates the error and generates an output command. This command is then transmitted to the drone controller, which adjusts the appropriate motors to achieve the desired roll angle.

Similarly, the second PID controller, PID 2, is responsible for controlling the pitch angle. It compares the desired pitch angle (setpoint) with the current measurement of the drone's pitch angle, calculates the error, and generates an output command. This output command is sent to the drone controller, which adjusts the motors to perform pitch movements and achieve the desired pitch angle.

The third PID controller, PID 3, focuses on controlling the yaw angle (heading) of the drone. It operates similarly to PID 1 and PID 2 but is dedicated to the yaw axis. PID 3 compares the desired yaw angle (setpoint) with the current measurement of the drone's yaw angle, calculates the error, and generates an output command. This command is then transmitted to the drone controller, which adjusts the motors to perform rotations around the vertical axis, achieving the desired yaw angle.

These three PID controllers work together to regulate the drone's movements in the three axes (roll, pitch, and yaw) based on the defined setpoints. Each PID utilizes the principles of proportional, integral, and derivative control to adjust the output command and maintain the drone in the desired position.

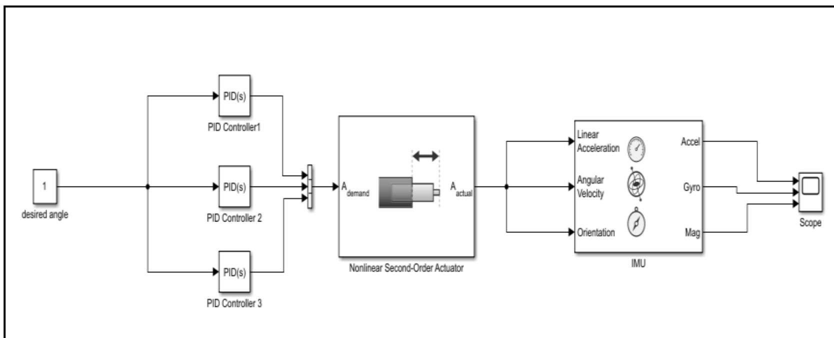

Fig. 4. Control system with three PID controllers

4 SIMULATION MODEL

In this section, simulation and experimental results are provided to evaluate the performance of the proposed PID controller design procedure under several gains obtained. In all tests, the quadrotor parameters.

We present the simulation of our control approach for the dynamic model of a quadcopter.

We have chosen a model that allows stabilizing the drone by bringing it to an equilibrium state characterized by constant or zero translation coordinates and orientation angles.

The control technique adopted to achieve this goal is PID control, which involves determining control parameters for each coordinate.

To account for the digital control of the quadcopter, we decided to design a discrete controller using MATLAB, based on the nonlinearity of the system. The discretization of the system was performed using the PID controller design method based on simplified assumptions presented in the previous section and the following parts of this section.

4.1 Model presentation

'**Error! Reference source not found.**' illustrates the complete architecture of the quadcopter simulation model in MATLAB. The model considers the quadcopter as a rigid body with a constant mass and symmetric geometry aligned with the principal axis of inertia, in a plus (+) configuration. The motors are depicted in two different colors to indicate the required synchronization to ensure the stability of the drone concerning the yaw axis.

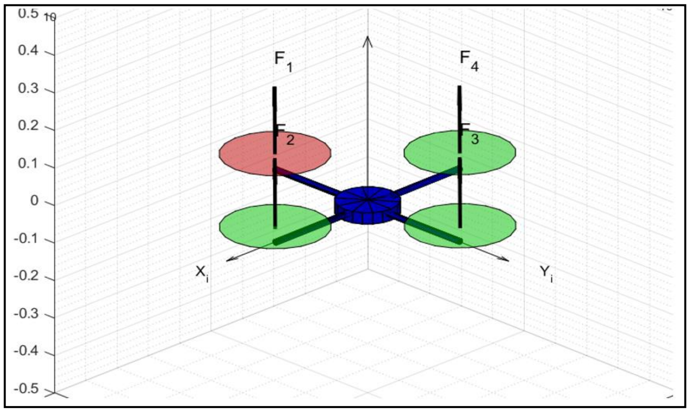

Fig. 5. Quadcopter model

4.2 Simulation model parameters

This paragraph lists the different parameters used in the simulations. We used the physical parameters of the quadcopter for the simulation tests. These parameters were used as initial conditions in the quadcopter's dynamic model. We also developed an adaptive control to regulate the quadcopter's rotational dynamics.

Error! Reference source not found. presents the different parameters used in the quadcopter's dynamic model

TABLE 1

PARAMETERS USED IN THE QUADCOPTER'S DYNAMIC MODEL

Symbol	Description	Value	Unit
G	Acceleration due to gravity	9.81	$m.s^{-2}$
mt	Weight of the motor and propellers	0.084	kg
mq	Mass of the Quadrotor	0.742	kg
At	The thickness of the arms	0.014	m
rp	The radius of the propeller	0.127	m
Lq	Length of the quadcopter arms	0.295	m
Jx= Jy	Moment of inertia around the x and y axis	0.0163	$kg.m^2$
Jz	Moment of inertia around the z-axis	0.0326	$kg.m^2$
F1,F2,F3,F4	Motors of Quadcopter	-	-

In this section, we conducted several flight simulations to test the performance of our control approach.

In the first test, we generated a single PID controller for the three angles Pitch, Roll and Yaw. We carefully selected appropriate gains after multiple attempts. We also examined the evolution of ($\overline{X}$, $\overline{Y}$, $\overline{Z}$) concerning the desired trajectories. This simulation allowed us to assess the performance of our control approach.

The second test involved simulating our drone with two PID controllers. The first controller was responsible for controlling the Pitch and Roll angles, while the second controller controlled the Yaw angle. We validated the performance of our control approach in tracking more complex trajectories, including those with non-zero derivatives.

Finally, we added a third PID controller for the theta angle to evaluate the robustness of our control approach.

Throughout these simulations, we analyzed various performance metrics, such as tracking errors, settling time, and stability, to assess the effectiveness and reliability of our control approach.

Error! Reference source not found. lists the initial conditions used for the three simulations, which include the initial linear and angular positions.

TABLE 2

THE INITIAL CONDITIONS FOR SIMULATIONS

Angles	Scenario 1	Scenario 2	Scenario 3
X	0	$\frac{\pi}{2}$	0
Y	0	0	0
Z	0	0	$\frac{\pi}{2}$
Pitch	0	0	0
Roll	0	0	0
Yaw	0	0	0

Error! Reference source not found. shows the lists of parameters used in the simulation tests:

TABLE 3

THE PARAMETERS USED IN THE SIMULATION TESTS

Parameter	Value
Simulation time	10 seconds
Trajectory type	Third-order polynomial
Initial conditions	Positions, velocities, accelerations
Added noise	None

4.3 Scenario 1 analysis

In this flight simulation, we will conduct tests to demonstrate the theoretical performance of our control approach.

In the first simulation test, we selected the PID controller parameters and performed the simulation to observe the behaviour of our control approach. The objective of this initial test is to determine the role of the control function in maintaining a bounded total thrust force.

The optimal PID controller parameters gains were chosen after multiple tests in **Error! Reference source not found.**.

TABLE 4

OPTIMAL PARAMETERS FOR SCENARIO 1

Ang	1 PID controller	2 PID controllers		3 PID controllers		
	All 3 angles	*Pitch and Roll*	*Yaw*	*Pitch*	*Roll*	*Yaw*
K_p	10.5	111.5	9	10	10	14
K_d	6	19.1	6	1	14.5	8
K_i	0.4	0.2	0.1	0.1	0.1	0.3

Error! Reference source not found.' illustrates the tracking of the desired trajectory by the Quadrotor during the flight simulation. It can be observed that the Quad-rotor accurately follows the trajectory, despite a slight initial error on all three axes.

For the Pitch angle, the perturbation is negligible when using a single controller, while it becomes significant when using two controllers. However, with three controllers, the perturbation is nearly eliminated. This suggests that using multiple PID controllers improves stability and reduces the impact of disturbances on the pitch angle.

Similarly, for the Roll angle, the same trend is observed. The perturbation is minimal with three controllers, whereas it increases when using two controllers and becomes significant when using only one controller. Again, this highlights the effectiveness of using multiple PID controllers for achieving better stability and minimizing angle errors in the roll direction.

Regarding the Yaw angle, the stabilization time differs among the controller configurations. With three PID controllers, the stabilization occurs after 2 seconds, while with two controllers, it takes 8 seconds. However, when using a single controller, the error in the angle is significant, indicating poorer performance in terms of yaw stabilization.

These observations emphasize the importance of using multiple PID controllers, as it helps improve stability, reduce perturbations, and minimize angle errors in different directions.

It showcases the advantage of employing a distributed control system that can effectively handle the control of multiple angles and enhance the overall performance of the drone.

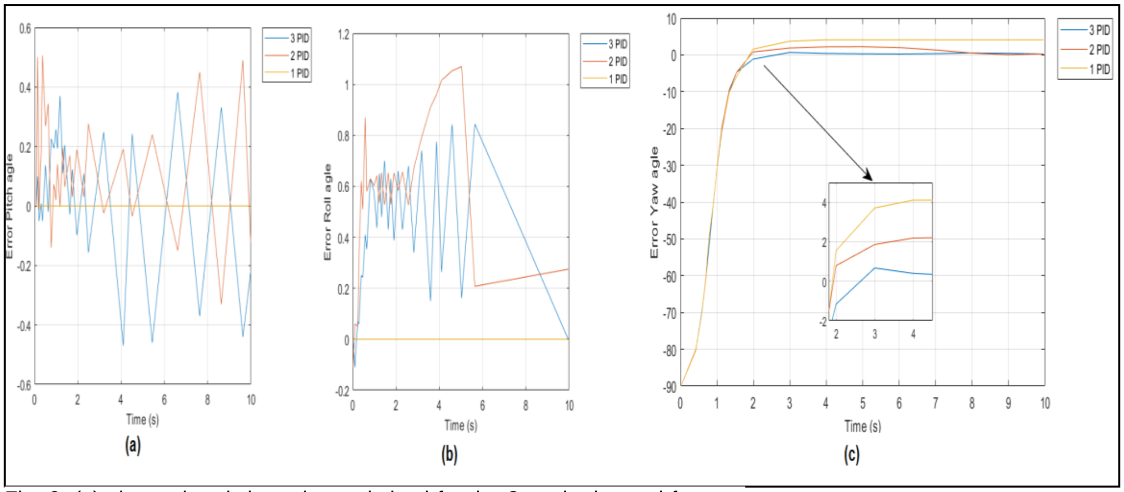

Fig. 6. (a) shows the pitch angles optimized for the 3 methods used for scenario 1 ; (b) shows the roll angles optimized for the 3 methods used for scenario 1; (c) shows the yaw angles optimized for the 3 methods used for scenario 1

4.4 Scenario 2 analysis

The optimal PID controller parameters gains were chosen after multiple tests in **Error! Reference source not found.**

TABLE 5

OPTIMAL PARAMETERS FOR SCENARIO 2

Ang	1 PID controller	2 PID controllers		3 PID controllers		
	All 3 angles	*Pitch and Roll*	*Yaw*	*Pitch*	*Roll*	*Yaw*
K_p	9	89	9	10	110	14
K_d	8	14.5	6	1	15	7.1
K_i	0.4	0.2	0.1	0.1	0.1	0.3

In the second scenario, as we can observe in '**Error! Reference source not found.**' when we used an initial condition on the $(\vec{X}, \vec{Y}, \vec{Z})$ axes $(\frac{\pi}{2}, 0, 0)$ we observed similar results to those mentioned earlier.

For the pitch angle, the drone stabilizes faster when using three PID controllers, while it takes more time with two controllers, and even longer with just one controller. This confirms the advantage of using several PID controllers to achieve faster stabilization and better pitch angle performance.

Similarly, for the roll angle, we observed similar results. The drone stabilizes faster with three PID controllers, takes more time with two controllers, and is even slower with just one controller. This indicates the importance of using multiple PID controllers to achieve better roll angle stabilization.

As previously stated, the stabilization time for the yaw angle varies depending on the controller configuration. Stabilization takes 2 seconds with three PID controllers and 3 seconds with two controllers. Using only one controller causes a significant angle error, demonstrating less than adequate yaw angle stabilization performance.

These consistent observations reinforce the idea that using multiple PID controllers improves stabilization and reduces angle errors in different directions. It is clear that combining multiple PID controllers provides better results in terms of shorter stabilization times and improved stabilization performance for pitch, roll, and yaw angles.

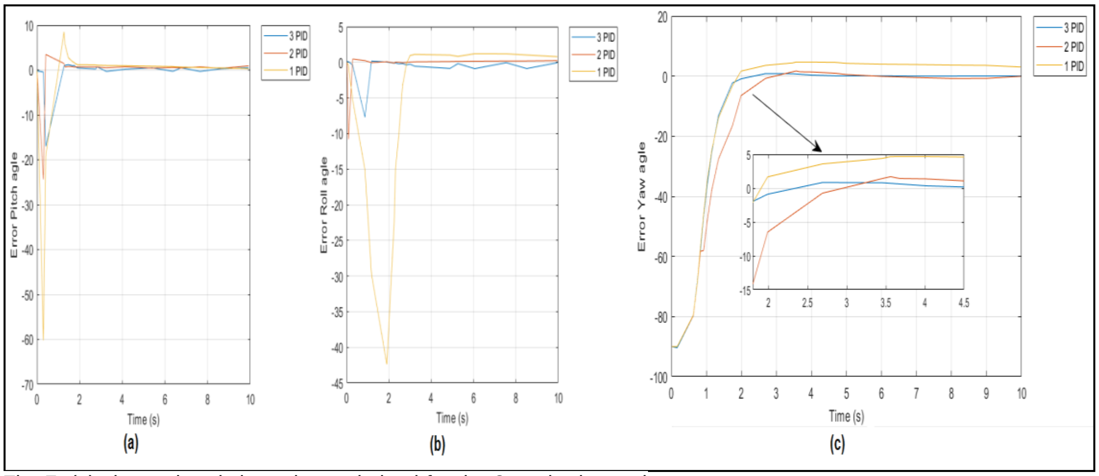

Fig. 7. (a) shows the pitch angles optimized for the 3 methods used for scenario 2 ; (b) shows the roll angles optimized for the 3 methods used for scenario 2; (c) shows the yaw angles optimized for the 3 methods used for scenario 2

4.5 Scenario 3 analysis

The optimal PID controller parameters gains were chosen after multiple tests in **Error! Reference source not found.**

TABLE 6

OPTIMAL PARAMETERS FOR SCENARIO 3

Ang	1 PID controller	2 PID controllers		3 PID controllers		
	All 3 angles	*Pitch and Roll*	*Yaw*	*Pitch*	*Roll*	*Yaw*
K_p	13	89	10	10	110	14
K_d	10	14.5	7	1	15	9.1
K_i	0.4	0.2	0.1	0.1	0.1	0.1

'**Error! Reference source not found.**' illustrates the flying robot's tracking of the desired trajectory in three-dimensional space during the flight.

The observations mentioned highlighting the differentiated performances of PID controllers for the pitch, roll, and yaw angles of the drone.

Regarding the pitch angle, when a single controller is used, the disturbance is negligible, but it becomes significant with two controllers. However, with three controllers, the disturbance is practically eliminated. These results indicate that using multiple PID controllers improves stability and reduces the impact of disturbances on the pitch angle.

Similarly, we observe a similar pattern with roll angle. When three controllers are utilized, the disturbance is minimal, increases with two controllers, and becomes severe when only one controller is used. These findings highlight the utility of employing several PID controllers to improve stability and reduce angle errors in the roll direction.

As for the yaw angle, the stabilization time differs depending on the controller configurations. With three PID controllers, stabilization occurs after 4 seconds, while with two controllers, it takes longer. However, using a single controller result in a significant angle error, indicating poorer performance in terms of yaw stabilization.

These observations highlight the crucial importance of using multiple PID controllers, as they improve stability, reduce disturbances, and minimize angle errors in different directions. They also underscore the advantage of adopting a distributed control system capable of efficiently managing the control of multiple angles and enhancing the overall performance of the drone.

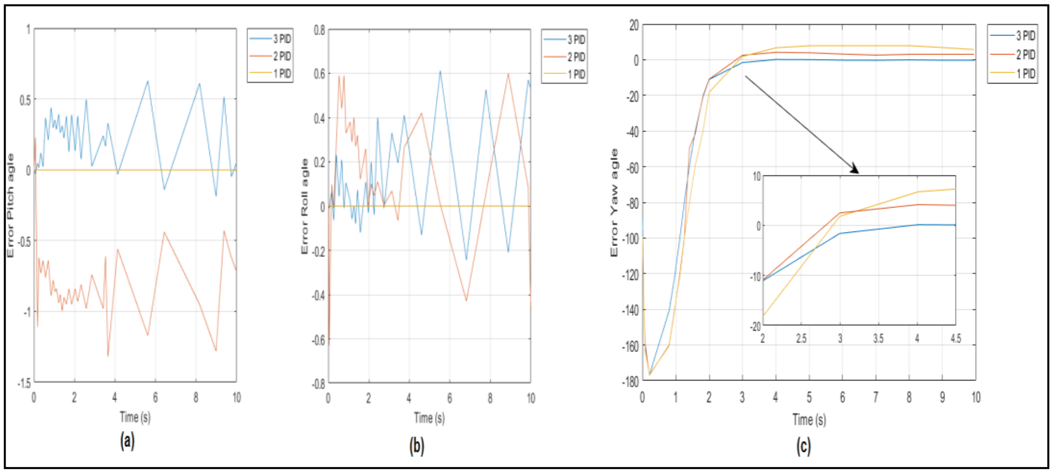

Fig. 8. (a) shows the pitch angles optimized for the 3 methods used for scenario 3 ; (b) shows the roll angles optimized for the 3 methods used for scenario 3; (c) shows the yaw angles optimized for the 3 methods used for scenario 3

5 Results and discussion

The objective of this comparative study aimed at optimizing PID gains is to compare the effectiveness of our approach, which involves using an improved PID controller with the Kalman filter to minimize disturbances, to other methods such as Genetic Algorithms (GA) [4], the Crow Search Algorithm (CSA) [5],Particle Swarm Optimization (PSO) [6] and Ziegler-Nichols (ZN) tuning method. The evaluation criteria we have selected are the stabilization time and the stability of the optimal PID gains.

We compared our simulations to the results proposed by Sheta, Alaa et al [15] for each approach. The simulations allowed us to measure and compare the obtained stabilization times.

In the developed approach, we achieved a stabilization time between 2 and 3 seconds, while the other approaches resulted in longer times.

The results are presented in **Error! Reference source not found.**, providing a clear visual representation of the performance of the approaches.

Table 7

The performance of the approaches in seconds

Angles	1 Contr	2 Contr	3 Contr	ZN	PSO	CSA	GA
Roll	3	0.5	1.5	50	20	20	20
Pitch	2	1.5	1.5	40	25	40	-
Yaw	10	3	2	80	40	80	17

We have thoroughly analyzed the obtained results. We examined the specific techniques used in each approach, highlighting the differences that could explain the variations in stabilization time.

The developed approach involved a manual tuning method for the PID gains, while the other approach utilized automated gain tuning methods. We emphasized the advantages of our approach in terms of shorter stabilization time, indicating better responsiveness and greater agility of the drone.

Regarding the stability of the optimal PID gains, we evaluated the ability of the approaches to maintain drone stability under various flight conditions and different initial conditions, as well as in the presence of disturbances. We found that our approach successfully maintained adequate stability of the optimal PID gains, whereas the other approaches exhibited some sensitivity to disturbances, resulting in longer stabilization times.

In conclusion, the developed comparative study demonstrated that our PID gain optimization approach led to a 3-second shorter stabilization time and improved stability of the optimal PID gains compared to the compared approach. These findings suggest that our manual PID gain tuning approach offers advantages in terms of drone responsiveness and stability.

6 CONCLUSION

In this paper, we divided the PID controller into three sub-controllers, each responsible for controlling a specific angle (roll, pitch, and yaw) of the quadrotor. This division into sub-controllers allowed us to improve the overall performance of the control model.

By assigning a sub-controller to each Euler angle, we were able to further fine-tune the control and adjust the parameters more specifically for each angle. This allowed for better control adaption to the specific characteristics of each angle and contributed to improved trajectory tracking accuracy.

The division into sub-controllers also helped reduce interactions between the different control components. By isolating the angles and applying specific control to each sub-controller, we were able to minimize the undesired coupling effects between the angles, resulting in better stability and overall improvement in the performance of the control model.

References

[1] R. S. e. a. SYAM, «Simulation and Experimental Works of Quadcopter Model for Simple Maneuver,» International Journal on Smart Material and Mechatronics, vol. 2, n° %11, p. 5, 2015.

[2] A. a. M. N. M. Safaei, «Lyapunov-based nonlinear controller for quadrotor position and attitude tracking with GA optimization,» chez EEE Industrial Electronics and Applications Conference, (IEACon), IEEE, pp. 342-347., 2016.

[3] K. K. Khuwaja, N.-u.-Z. Lighari, I. C. Tarca et R. CatalinTarca, «PID Controller Tuning Optimization with Genetic Algorithms for a Quadcopte,» Recent Innovations in Mechatronics (RIiM), vol. 5, n° %11, p. 7, 2018.

[4] V. G. N. R. J. e. a. GOMEZ, «Pareto Optimal PID Tuning for Px4-Based Unmanned,» Aerospace, vol. 7, n° %16, p. 20, 2020.

[5] A. B. M. S. &. A. S. Sheta, «Genetic algorithms: a tool for image segmentation,» chez international conference on multimedia computing and systems IEEE, Tangiers, Morocco, 2012.

[6] A. Askarzadeh, «A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm,» Computers & structures, vol. 169, pp. 1-12, 2016.

[7] S. Bouabdallah, A. Noth et R. Siegwart, «PID vs LQ control techniques applied to an indoor micro quadrotor,» chez 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), Sendai, Japan, 2004.

[8] U. e. E. A. C. TILKI, «HIGHER ORDER SLIDING MODE CONTROL OF FOUR ROTOR UNMANNED AERIAL VEHICLE,» chez International Science Mathematics and Engineering Sciences Congress, 2019.

[9] P. Jaiswal, «Demystifying Drone Dynamics!,» towardsdatascience, 30 October 2018. [En ligne]. Available: https://towardsdatascience.com/demystifying-drone-dynamics-ee98b1ba882f. [Accès le 01 June 2023].

[10] A. Sarhan et S. Qin, «Adaptive PID Control of UAV Altitude Dynamics Based on Parameter Optimization with Fuzzy Inference,» International Journal of Modeling and Optimization, vol. 6, n° %14, p. 6, 2016.

[11] S. Zouaoui, E. Mohamed et B. Kouider, «Easy Tracking of UAV Using PIDController,» Periodica PolytechnicaTransportation Engineering, vol. 1, n° %11, p. 7, 2018.

[12] S. Khatoon, M. Shahid, Ibraheem et H. Chaudhary, «Dynamic modeling and stabilization of quadrotor using PID controller,» chez IEEE, Delhi, India, 2014.

[13] K. J. Åström, «PIDControl,» Control System Design, 2002.

[14] Y. Chan, M. Moallem et W. Wang, «Efficient implementation of PID control algorithm using FPGA technology,» chez IEEE, Nassau, Bahamas, 2004.

[15] A. B. M. M. D. R. M. A. A. S. &. T. H. Sheta, «Optimization of pid controller to stabilize quadcopter movements using meta-heuristic search algorithm,» Applied Sciences MPDI, vol. 11, n° %114, p. 31, 2021.