

Analysis and Tuning of PID and Fuzzy Logic Controllers for a DC Motor-Driven Quadcopter Drone Altitude Control System

Adel Rawea* , Mohammed Almodhwah

Department of Mechatronics Engineering, Faculty of Engineering and Technology, Ar-Rasheed Smart University, Sana'a, Yemen.

*Corresponding Author:

E-mail: adel.rawea@ar-rasheed.edu.ye

Received: 18 May 2026

Revised: 14 June 2026

Accepted: 23 June 2026

تحليل وضبط متحكمات PID & FLC لنظام التحكم بارتفاع طائرة رباعية المرواح المدفوعة بمحركات التيار المستمر

عادل راعو* , محمد المصواحي

قسم هندسة الميكاترونكس، كلية الهندسة والتكنولوجيا، جامعة الرشيد الذكية، صنعاء، اليمن

*المؤلف المراسل:

الإيميل: adel.rawea@ar-rasheed.edu.ye

تاريخ التقديم: 18 مايو 2026

تاريخ إرسال التعديلات: 14 يونيو 2026

تاريخ القبول: 23 يونيو 2026

Abstract

Quadcopter drones require precise and robust control systems to maintain stable flight under varying operating conditions. This article presented a modeling, design, tuning, and comparative evaluation of proportional-integrator-derivative (PID) and fuzzy logic controllers (FLC) for a DC-powered quadcopter altitude control system. Unlike simplified models that only consider altitude, the proposed approach integrated the entire control chain, including DC motor dynamics, propeller thrust generation, and the drone's vertical movement. The mathematical model of the motor was derived from electrical and mechanical principles, while the nonlinear thrust generation process was linearized around the hovering point. The PID controller was tuned for fast response, low over-oscillation, and zero steady-state error. To enhance robustness against nonlinear effects and distortions, a Mamdani-type fuzzy logic controller with two inputs and seven membership functions per variable was developed. MATLAB simulations were performed to compare transient response, frequency response, tracking accuracy, and interference rejection performance. The results found that the PID controller provided faster response (0.96 sec versus 1.53 sec for FLC), but exhibited higher overshoot (23.35% versus 12.81% for FLC). Both controllers achieved near-zero steady-state error (PID: 1.00001 rad/s, FLC: 0.9842 rad/s), while the fuzzy logic controller provided superior robustness and more stable operation against external interferences.

Keywords: Quadcopter drone; PID controller; FLC controller; altitude control; DC motor modeling; MATLAB/simulink.

الملخص

تتطلب طائرات الدرون رباعية المرواح أنظمة تحكم دقيقة، ومتينة للحفاظ على أنظمة طيران مستقرة في ظل ظروف تشغيل متغيرة. تقدم هذه الدراسة نهجاً لنمذجة وتصميم المتحكم التقليدي التناسبي التكاملي التفاضلي (PID)، والمتحكم المنطقي الضبابي (FLC) لنظام التحكم وضبطها، كما تقدم تقبلاً مقارناً لتأثيرهما على ارتفاع طائرة الدرون رباعية المرواح التي تعمل بنظام التيار المستمر. وعلى عكس النماذج المبسطة التي تأخذ الارتفاع فقط بعين الاعتبار، يقدم هذا النهج المقترح سلسلة تحكم كاملة، بما في ذلك ديناميكيات محرك التيار المستمر، وتوليد قوة دفع المروحة، والحركة الرأسية لطائرة الدرون. تم اشتقاق النموذج الرياضي للمحرك الكهربائي من المبادئ الكهربائية والميكانيكية، بينما يتم تحويل عملية توليد قوة الدفع غير الخطية إلى الشكل الخطي حول نقطة التحليق (Hovering Point). بعد ذلك تم ضبط بارامترات المتحكم PID لتحقيق استجابة سريعة، وتذبذب زائد منخفض، وخطأ صفري في الحالة المستقرة. ولتعزيز متانة نظام التحكم ضد التأثيرات والتشوهات غير الخطية، تم تطوير متحكم منطقي ضبابي يعتمد خوارزمية مامداني (Mamdani) بمدخلين وسبع دوال عضوية لكل متغير. أُجريت المحاكاة باستخدام برنامج MATLAB لمقارنة الاستجابة العابرة، ودقة التتبع، وأداء رفض التداخلات الخارجية. وتوضح النتائج أن المتحكم التقليدي PID يوفر استجابة أسرع (0.96 ثانية مقابل 1.53 ثانية لمتحكم المنطق الضبابي FLC)، لكنها تظهر تجاوزاً أكبر (23.35٪ مقابل 12.81٪ لمتحكم المنطق الضبابي FLC). وقد حققت كلتا وحدتي التحكم خطأً شبه معدوم في الحالة المستقرة (PID: 1.00001 راديان/ثانية، FLC: 0.9842 راديان/ثانية)، في حين توفر وحدة تحكم المنطق الضبابي متانة فائقة وتشغيلاً أكثر استقراراً في مواجهة التداخلات الخارجية.

الكلمات المفتاحية: الطائرات رباعية المرواح، المتحكم التقليدي PID، المتحكم الضبابي FLC، التحكم في الارتفاع، نمذجة محرك التيار المستمر.

This is an open access article distributed under the terms of the [Creative Commons Attribution 4.0 International License \(CC BY 4.0\)](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ISSN: 3135-6842 | E-ISSN: 3135-6850 / © 2026 The Authors. Published by Ar-Rasheed Smart University, Yemen.

Journal website: <https://journals.ar-rasheed.edu.ye/RUJBAS/en>

Ar-Rasheed University Journal of Basic and Applied Sciences 2026, 1(1), 49-60

<https://doi.org/10.67151/rujbas.v1i1.40>

1. Introduction

Drones play a vital role in numerous fields, including monitoring, surveillance, environmental tracking, agricultural applications, and autonomous transportation, etc. [1]. Despite the variety of available drones, quadcopters are the most widespread owing to their many advantages, such as the ability to take off and land vertically, simple design, and high flight and maneuvering performance [2].

All these features and performance characteristics depend largely on the selection of appropriate control systems. Operating in highly dynamic environments, and being subject to disturbances caused by wind, load variations, and modeling uncertainties, such systems require robust control strategies to achieve stable flight.

Traditional proportional-integrator-derivative (PID) controllers are the most commonly used control methods in industrial applications for their simplicity and ease of implementation [3]. However, their performance may decrease when nonlinear effects and uncertainties become significant [4]. To improve robustness and adaptability, intelligent control techniques such as fuzzy logic controllers have been proposed.

This research article presents a comparison between the performance of PID controller and fuzzy logic controllers (FLC), which are the most common controllers in various control systems. It utilizes the features of each type to improve the performance of a quadcopter's altitude control system, taking into account the actual dynamics of DC motors, to generate thrust for each propeller. The objectives of this study are as follows [5]:

1. To develop a complete mathematical model of a DC motor-propeller-drone system.
2. To design a PID controller that meets the specified performance requirements.
3. To develop a FLC based on system expert knowledge.

4. To compare the two controllers through simulation.
5. To evaluate their effectiveness for drone altitude control.

2. Literature Review

2.1. PID-Controlled Drones

Several studies have demonstrated the effectiveness of PID controllers for stabilizing and controlling the altitude of quadcopters. Bouhabza and Kara [6] developed a control system based on a PID controller for quadcopter attitude and altitude, which provided satisfactory performance under regular operating conditions. Their work confirmed stable control of roll, pitch, yaw, and altitude, while maintaining a relatively simple controller structure.

Similarly, Castillo et al. [7] used PID controllers for self-directed drone navigation and reported reliable altitude tracking in hover and at low speeds.

Despite its widespread adoption, PID control has some limitations. Controller performance is highly dependent on the tuning of proportional, integral, and derivative constants. Inappropriate tuning may cause excessive overshoot, slow response, or even instability [8, 9]. Moreover, the nonlinear aerodynamic characteristics of quadcopters, combined with variations in environmental conditions and load, can significantly affect controller performance [10, 11]. In practice, a set of PID gains that perform well under specific operating conditions, may not offer the same level of performance under others. Hence, achieving optimal and consistent control remains a challenge, especially in dynamic and uncertain environments.

2.2. FLC-Based Drone Control Systems

To overcome the limitations of PID controllers, researchers have increasingly turned to intelligent control techniques, among which fuzzy logic controller has appeared as a

promising type. Unlike traditional controllers that rely heavily on mathematical models, fuzzy logic controllers rely on linguistic rules and expertise. This characteristic allows them to more effectively manage nonlinearities and uncertainties without requiring a precise representation of the system's dynamics.

One of the initial motivations for applying FLC to drone systems was the observation that experienced human operators are often able to stabilize and maneuver drones using qualitative reasoning rather than precise mathematical calculations. FLC aims to replicate this reasoning process by translating expertise into a set of IF-THEN control rules [12].

Several studies have demonstrated the effectiveness of FLC in drone applications. Coza and Macnab [13] proposed a FLC for quadcopter attitude stabilization, and found increased robustness compared to PID controllers. Their results showed smoother responses, and reduced oscillations during aggressive flight maneuvers. Similarly, Amrouche et al. [14] used a Mamdani-type FLC for altitude control and observed improved disturbance rejection performance when the drone was subjected to wind disturbances.

An important advantage of FLC lies in its ability to represent system behavior, using linguistic variables such as Negative Big (NB), Little Negative (LS), Zero (ZE), Little Positive, and Big Positive. By using these linguistic descriptions rather than exact numerical models, FLC can adjust more naturally to nonlinear operating conditions and parameter variations [12].

However, FLC presents several design challenges. The selection of membership functions, scaling factors, and rule bases often relies on engineering experience and iterative tuning. As the number of controller inputs and outputs increases, the rule base can grow rapidly, leading to increased computational

complexity. Furthermore, because the performance of FLC controllers is highly dependent on the quality of their rule set, developing effective FLC controllers may require considerable effort and specialized expertise.

In general, previous studies indicate that FLC offer greater robustness and smoother response than conventional PID controllers, particularly in nonlinear and uncertain systems.

2.3. Research Gaps and Motivation

Studies demonstrate that PID controllers remain the dominant solution for drone applications, given their simplicity and low computational requirements [15]. On the other hand, fuzzy logic controllers offer enhanced robustness and better disturbance rejection capabilities in nonlinear environments [16]. While advanced intelligent controllers, such as FLC and reinforcement learning, have shown promising results, their implementation complexities often limit their adoption in low-cost drone platforms [17, 18].

Therefore, a comprehensive comparison between conventional PID control and FLC is essential, especially for quadcopter drone altitude control systems that incorporate realistic motor dynamics, thrust generation mechanisms, and vertical motion dynamics. While most recent studies focus on either simplified altitude models or attitude control systems, research addressing the entire motor-propeller-altitude control chain is relatively scarce.

This study, therefore, develops a complete altitude model for a quadcopter powered by a DC motor and performs a detailed comparative evaluation of PID and FLC under the same operating conditions. The aim is to identify the advantages and limitations of each type in terms of transient response, steady-state accuracy, robustness, and disturbance rejection performance.

3. System Modeling

3.1. DC Motor Dynamics

A quadcopter's propulsion system uses electric motors to convert electrical energy into mechanical rotational motion, which produces propeller thrust. Therefore, an accurate mathematical model of the DC motor is crucial for the design and evaluation of control strategies. In this study, motor dynamics are represented by interconnected electrical and mechanical subsystems [19].

1) Electrical Subsystem

Kirchhoff's voltage law describes the electrical behavior of armature circuit. The voltage applied to the armature is distributed between the armature resistance and inductance and the back electromotive force (EMF) generated by the motor's rotation. The equation governing the operation of the motor is expressed as follows [20]:

$$V(t) = L \frac{d}{dt} i(t) + R \cdot i(t) + K_b \cdot \omega(t) \quad (1)$$

where:

$V(t)$ is the armature voltage (V),

$i(t)$ is the armature current (A),

R is the armature resistance (Ω),

L is the armature inductance (H),

K_b is the back electromotive constant (V·s/rad),

$\omega(t)$ is the angular velocity (rad/s).

The first term on the right side represents the voltage drop across the armature inductance, the second term represents the resistive voltage drop, and the third term represents the back electromotive force induced by the motor rotation.

Applying the Laplace transform to equation (1) with zero initial conditions, we obtain:

$$V(s) = (Ls + R)I(s) + K_b\Omega(s) \quad (2)$$

where $I(s)$ and $\Omega(s)$ represent the Laplace transforms of the armature current and angular velocity, respectively.

2) Mechanical Subsystem

The mechanical dynamics of the motor are governed by Newton's second law for rotating systems. The generated electromagnetic torque accelerates the rotor by compensating for friction losses. The equation governing this event is [20]:

$$J \left(\frac{d}{dt} \omega(t) \right) + B \cdot \omega(t) = K_t \cdot i(t) \quad (3)$$

where:

J is the moment of inertia of the rotor ($\text{kg}\cdot\text{m}^2$),

B is the viscous friction coefficient ($\text{N}\cdot\text{m}\cdot\text{s}/\text{rad}$),

K_t is the torque constant of the motor ($\text{N}\cdot\text{m}/\text{A}$).

The first term represents the inertia torque required for the acceleration of the rotor, while the second term models mechanical friction. The right side corresponds to the electromagnetic torque produced by the motor. The Laplace transform of (3) gives:

$$(J \cdot s + B) \cdot \Omega(s) = K_t \cdot I(s) \quad (4)$$

Solving this equation for the armature current, we get:

$$I(s) = \frac{(J \cdot s + B) \cdot \Omega(s)}{K_t} \quad (5)$$

3) Motor Transfer Function

By substituting (5) into (2), we eliminate the armature current and obtain:

$$V(s) = \left[\frac{(L \cdot s + R) \cdot (J \cdot s + B)}{K_t} + K_b \right] \cdot \Omega(s) \quad (6)$$

Multiplying both sides by K_t , we obtain:

$$K_t \cdot V(s) = [(L \cdot s + R) \cdot (J \cdot s + B) + K_t K_b] \cdot \Omega(s) \quad (7)$$

Therefore, the transfer function relating the angular velocity of the motor to the applied voltage is:

$$G_{motor}(s) = \frac{\Omega(s)}{V(s)} = \frac{K_t}{(L \cdot s + R)(J \cdot s + B) + (K_t K_b)} \quad (8)$$

Equation (8) represents a second-order electromechanical system that describes the

electrical and mechanical dynamics of the motor.

4) Numerical Model of the Motor

The motor parameters used in this study are listed in Table 1.

Substituting these values into (8), we obtain:

$$G_{motor}(s) = \frac{0.01}{[(10^{-4}s + 0.1)(10^{-5}s + 10^{-4}) + 0.0001]} \quad (9)$$

By expanding the denominator, we obtain:

$$G_{motor}(s) = \frac{0.01}{10^{-9}s^2 + 1.01 \times 10^{-6}s + 1.1 \times 10^{-4}} \quad (10)$$

Equation (10) represents the final transfer function of the DC motor, which will be used in the controller design and simulations presented in this paper. This transfer function forms the first block of the quadcopter's complete altitude control model, where the motor speed is then converted into propeller thrust and vertical motion.

Table 1. DC motor parameters

No.	Parameter	Description	Value
1	J	Rotor inertia	10^{-5} kg.m^2
2	B	Viscous friction coefficient	$10^{-4} \text{ N.m.s/rad}$
3	K_t	Torque constant	0.01 N.m/A
4	K_b	Back emf constant	0.01 V.s/rad
5	R	Armature resistance	0.1Ω
6	L	Armature inductance	10^{-4} H

3.2. Propeller Thrust Model

The thrust force produced by a propeller is proportional to the square of its angular velocity [21]:

$$T = K_f \omega^2 \quad (11)$$

where:

T is the thrust force,

K_f is the thrust coefficient.

This model is not linear. For controller design, a linearization is performed around hover speed point $\omega = \omega_0 + \Delta\omega$, and using Taylor expansion, $T \approx T_0 + 2K_f \omega_0 \Delta\omega$; where $T_0 = K_f \omega_0^2$.

By neglecting the constant hovering component, we obtain: $\Delta T = K_T \Delta\omega$, where $K_T = 2K_f \omega_0$ is the equivalent linearized thrust gain.

3.3. Hover Condition

The mass of a four-propeller drone is assumed to equal [22], $m = 4 \text{ kg}$, then the total hover thrust is given by $T_{hover} = m \cdot g$. By substituting $g = 9.81 \text{ m/s}^2$, the total hover thrust is evaluated as $T_{hover} = 39.24 \text{ N}$

Then the drone force by four motors will equal $T_{motor} = \frac{39.24}{4} = 9.81 \text{ N}$.

Assuming the hovering speed $\omega_0 = 500 \text{ rad/s}$, the thrust coefficient becomes $K_f = \frac{9.81}{500^2} = 3.924 \times 10^{-5}$ and the linearized thrust gain is defined as follows [9].

$$K_T = 2K_f \omega_0 = 0.03924$$

3.4. Altitude Dynamics

The vertical axis of the drone's motion follows Newton's second law $m\ddot{z} = T - m \cdot g$. By linearizing around the altitude [23], $m\ddot{z} = \Delta T$, and applying the Laplace transform, the transfer function is as follows:

$$G_{altitude}(s) = \frac{Z(s)}{\Delta T(s)} = \frac{1}{m s^2}, \text{ for } m = 4 \text{ kg, then } G_{altitude}(s) = \frac{1}{4 s^2}$$

3.4. Complete Drone Model

By combining all blocks, the total drone model becomes $G_{Drone}(s) = G_{motor}(s) \times K_T \times G_{altitude}(s)$ where: $G_{motor}(s) = \frac{\Omega(s)}{V(s)}$, $K_T = \frac{\Delta T(s)}{\Delta \Omega(s)}$, and $G_{altitude}(s) = \frac{Z(s)}{\Delta T(s)}$. Therefore, the complete transfer function will be:

$$\begin{aligned} G_{Drone}(s) &= \frac{Z(s)}{V(s)} \\ &= \frac{0.01}{10^{-9}s^2 + 1.01 \times 10^{-6}s + 1.1 \times 10^{-4}} \\ &\quad \times 0.03924 \times \frac{1}{4 s^2} \\ &= \frac{9.81 \times 10^{-5}}{s^2(10^{-9}s^2 + 1.01 \times 10^{-6}s + 1.1 \times 10^{-4})} \end{aligned}$$

4. Designing of PID Controller

PID controllers are one of the most widely used feedback control strategies in numerous engineering applications and systems. They are used in a wide range of applications, from simple control systems to very complex ones, in automobiles and industrial process automation.

The PID controller continuously estimates an error according to the equation [24]:

$$e(t) = r(t) - y(t)$$

Where:

$r(t)$ = setpoint value (reference input).

$y(t)$ = actual output.

$e(t)$ = control error

4.1. PID Controller Structure

The PID controller then gives the control signal $u(t)$, which is represented by the equation:

$$u(t) = K_p \cdot e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t)$$

$$U(s) = K_p \cdot E(s) + K_i \cdot \frac{E(s)}{s} + K_d \cdot s \cdot E(s)$$

Where:

K_p = proportional gain.

K_i = integral gain.

K_d = derivative gain.

4.2. Design Objectives

The altitude control loop is designed to achieve:

- Rise time ≤ 0.3 s;
- Overshoot $\leq 13\%$;
- Steady-state error ≈ 0 .

These system specifications are realistic for 4 kg drone altitude controller.

4.3. PID Controller Tuning

MATLAB PID Tuner is employed to regulate and select optimal PID gains. The final tuned gains values are [12, 13]:

$$K_p = 161.4 \text{ [V/(rad/s)]};$$

$$K_i = 50 \text{ [V. s/rad]};$$

$$K_d = 21.52 \text{ [V. s/rad]}.$$

The PID controller transfer function is:

$$\begin{aligned} C_{PID} &= \frac{(K_d \cdot s^2 + K_p \cdot s + K_i)}{s} \\ &= \frac{(21.52s^2 + 161.4s + 50)}{s} \end{aligned}$$

The open loop PID transfer function is:

$$\begin{aligned} OL_{PID} &= C_{PID} * G_{Drone} \\ &= \frac{(21.52s^2 + 161.4s + 50)}{s} \\ &\quad \times \frac{9.81 \times 10^{-5}}{s^2(10^{-9}s^2 + 1.01 \times 10^{-6}s + 1.1 \times 10^{-4})} \end{aligned}$$

The closed loop PID transfer function is:

$$CL_{PID} = \frac{OL_{PID}}{1 + OL_{PID}} = \frac{0.00211 s^2 + 0.01583s + 0.004905}{10^{-9}s^5 + 1.01 \times 10^{-6}s^4 + 1.1 \times 10^{-4}s^3 + 0.00211 s^2 + 0.01583s + 0.004905}$$

5. Design of FLC Controller

5.1. FLC Controller Structure

The fuzzy logic controller contains [25, 26]:

- Fuzzification
- Rule Base
- Inference Engine
- Defuzzification

5.2. FLC Inputs and Output

- Inputs:
 - $e(t) = r(t) - y(t)$
 - $\Delta e(t) = \frac{d}{dt}e(t)$
- Output is $u(t)$.

5.3. Membership Functions

In this paper, seven linguistic variables are employed:

- NB (Negative Big)
- NM (Negative Medium)
- NS (Negative Small)
- ZE (Zero)
- PS (Positive Small)
- PM (Positive Medium)
- PB (Positive Big)

Input and output variables are linked to a standard discourse domain, ranging within the intervals $[-1, 1]$, using the following measurement factors, as shown in Table 2.

Table 2. Normalization parameters for FLC variables (error, delta error, and output)

Variable	Symbol	Physics range	Scaling factor	Normalized range
Error	e	$[-10, +10]$ m	$G_e = 0.1$	$[-1, +1]$
Delta error	Δe	$[-50, +50]$ m/s	$G_{de} = 0.02$	$[-1, +1]$
Output	U	$[-5, +5]$	$G_u = 5.0$	$[-1, +1]$

Triangular membership functions are selected for all inputs and output, as shown in Figure 1 from the MATLAB simulation:

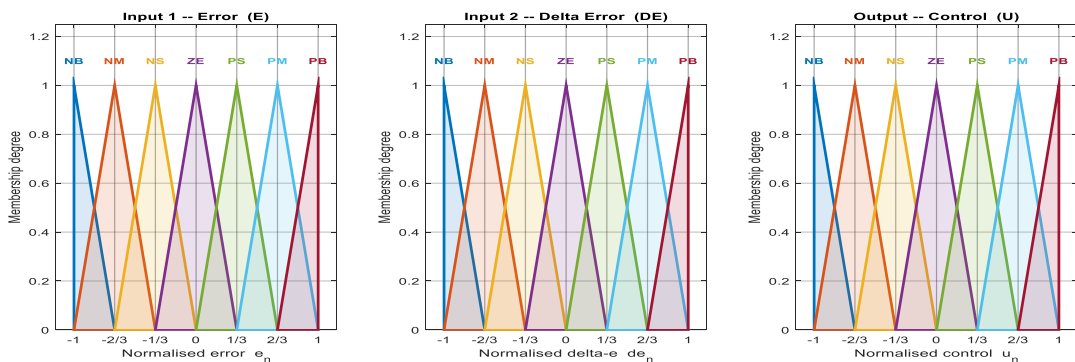


Figure 1. Fuzzy logic controller membership functions (inputs/output).

5.4. Base Rules

A 49-rule (7×7) matrix is developed, as shown in Table 3, where:

- Row = E index (1=NB ... 7=PB);
- Col = DE index (1=NB ... 7=PB);
- Cell = Output MF index (1=NB ... 7=PB).

The Mamdani inference mechanism and centroid defuzzification method are applied.

Table 3. Rule table (rows=Error MF, cols=Delta Error MF, cell=Output MF)

DE E	NB	NM	NS	ZE	PS	PM	PB
NB	1	1	1	2	3	4	4
NM	1	1	2	3	4	5	5
NS	1	2	3	4	5	6	6
ZE	2	3	4	4	4	5	6
PS	3	4	5	5	6	7	7
PM	4	5	5	6	7	7	7
PB	4	5	6	6	7	7	7

6. Simulation for Results and Discussion

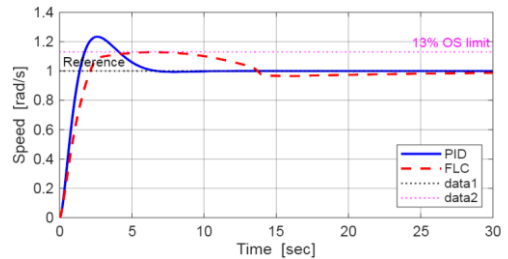
6.1. Time Response

The step response is obtained for both controllers (PID & FLC), and the performance metrics as shown in Table 4 include Rise Time, Overshoot and Steady-State Error [27, 28]:

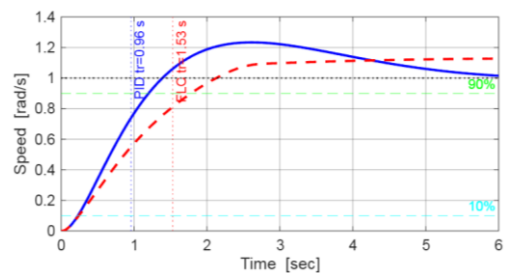
Table 4. Performance comparison table: PID vs FLC

No.	Metric	PID controller	FLC Controller
1	Rise Time [sec]	0.96	1.53
2	Overshoot [%]	23.348	12.812
3	Steady-State [rad/s]	1.00001	0.9842
4	ISE [$\times 10^{-6}$]	595.561	754.397
5	ITAE [$\times 10^{-6}$]	2.410	13.625

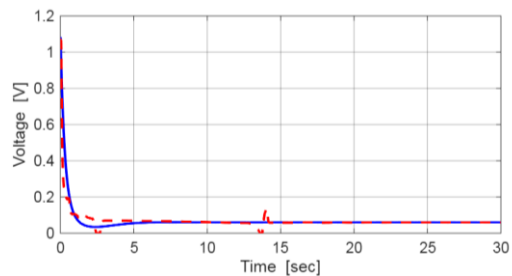
The step speed response of both controllers, as well as rise transient, control effort and disturbance rejection from MATLAB simulation are shown in Figure 2 (a–d):



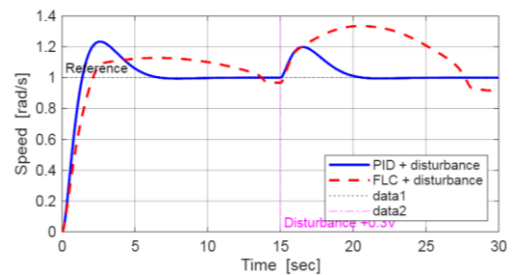
(a) Step speed response -- Full view



(b) Zoom: rise transient (0 - 6 sec)



(c) Control effort $u(t)$



(d) Disturbance rejection (d injected at plant input)

Figure 2. Time response: PID vs. FLC.

6.2. Error Analysis

The tracking error is computed as $e(t) = r(t) - y(t)$, and the convergence

characteristics of both controllers (PID & FLC) are analyzed. Figure 3 shows the results [29-31]:

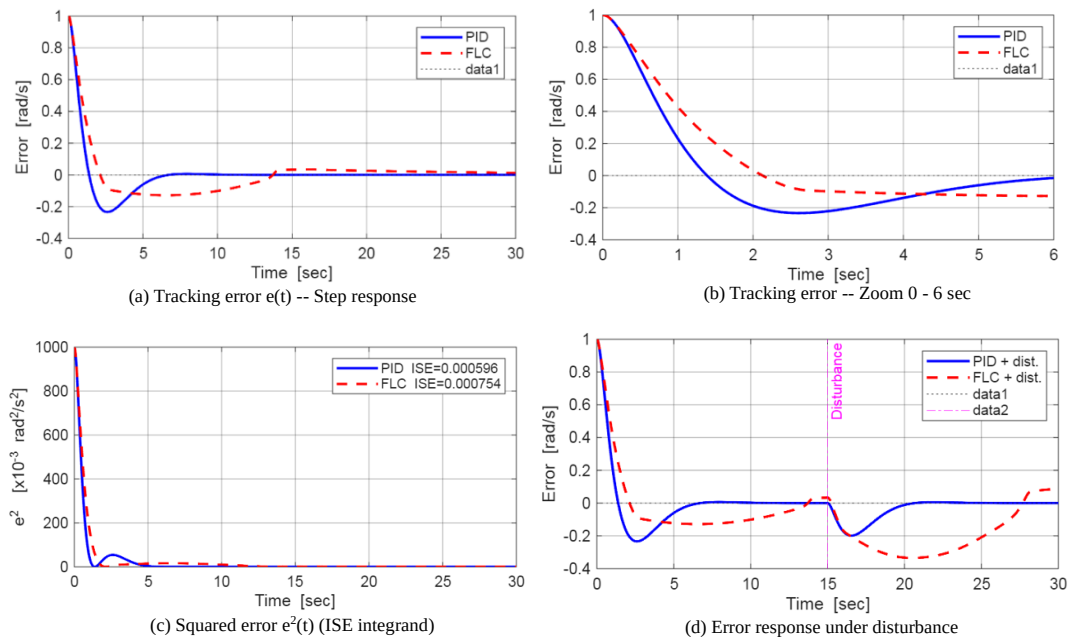


Figure 3. Error analysis: PID vs. FLC.

7. Conclusion

The paper presented a complete integration of motor dynamics and height control through a comprehensive comparative analysis under identical conditions. It demonstrated that, while a PID controller provides fast response and simple implementation, its performance might degrade under significant parameter changes and nonlinear operating conditions. In contrast, a FLC controller exhibits smoother responses and better robustness against distortions and model uncertainties. However, these results are limited because the study is based on simulations, as well as assumptions regarding the linearity of the drive model and idealized motor parameters. Therefore, recommendations for future work could include a hybrid architecture of (PID) and (FLC) controllers, optimization of FLC

parameters using a genetic algorithm, and advanced perturbation modeling.

Data Availability

The data are available from the corresponding author upon request.

Funding

This research received no external funding.

References

- [1] Mohsan SAH, Othman NQH, Li Y, Alsharif MH, Khan MA. Unmanned aerial vehicles (UAVs): practical aspects, applications, open challenges, security issues, and future trends. *Intelligent Service Robotics*. 2023;16(1):109-137. <https://doi.org/10.1007/s11370-022-00452-4>

- [2] Zulu A, John S. A review of control algorithms for autonomous quadrotors. *Open Journal of Applied Sciences*. 2014;4(14):547-556.
<http://dx.doi.org/10.4236/ojapps.2014.414053>
- [3] Ang KH, Chong G, Li Y. PID control system analysis, design, and technology. *IEEE Transactions on Control Systems Technology*. 2005;13(4):559-576.
<https://doi.org/10.1109/TCST.2005.847331>
- [4] Hu N. The limitations of traditional PID controllers and modern optimization methods. *Proceedings of the 3rd International Conference on Mechatronics and Smart Systems*. 2025:238-244.
<https://doi.org/10.54254/2755-2721/2025.22912>
- [5] Isdaryani F, Shidiq MN, Wicaksono Y. Flight controller design for altitude control of a quadcopter using PID and fuzzy methods. *Jurnal Teknologi Rekayasa*. 2022;7(2):251-258.
<https://doi.org/10.31544/jtera.v7.i2.2022.251-258>
- [6] Bouhabza N, Kara K. Optimized sliding mode based PID controller for a quadrotor system. 2022 2nd International Conference on Advanced Electrical Engineering (ICAEE); 2022 Oct 29-31. Constantine, Algeria: IEEE; 2022. p. 1-6.
<https://doi.org/10.1109/ICAEE53772.2022.9962024>
- [7] Castillo P, Lozano R, Dzul AE. *Modelling and control of mini-flying machines*. London: Springer-Verlag; 2015. <https://doi.org/10.1007/1-84628-179-2>
- [8] Åström KJ, Hägglund T. *Advanced PID Control*. Research Triangle Park, NC: ISA – Instrumentation, Systems, and Automation Society; 2006.
- [9] O'Dwyer A. *Handbook of PI and PID controller tuning rules*. 3rd ed. London: Imperial College Press; 2009.
- [10] Hoffmann GM, Huang H, Waslander SL, Tomlin CJ. Quadrotor helicopter flight dynamics and control: theory and experiment. In: *AIAA Guidance, Navigation and Control Conference and Exhibit*; 2007 Aug 20-23. Reston, VA: American Institute of Aeronautics and Astronautics; 2007.
<https://doi.org/10.2514/6.2007-6461>
- [11] Waslander SL, Wang C. Wind disturbance estimation and rejection for quadrotor position control. In: *AIAA Infotech@Aerospace Conference*; 2009 Apr 6-9. Reston, VA: American Institute of Aeronautics and Astronautics; 2009. <https://doi.org/10.2514/6.2009-1983>
- [12] Zadeh LA. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man, and Cybernetics*. 1973;SMC-3(1):28-44.
<https://doi.org/10.1109/TSMC.1973.5408575>
- [13] Coza C, Macnab CJB. A new robust adaptive-fuzzy control method applied to quadrotor helicopter stabilization. In: *NAFIPS 2006 – 2006 Annual Meeting of the North American Fuzzy Information Processing Society*; 2006 Jun 3-6. Montreal, QC, Canada: IEEE; 2006. p. 454-8.
<https://doi.org/10.1109/NAFIPS.2006.365452>
- [14] Amrouche R, Fas ML, Benrabah M, Ghoul A. Adaptive TLBO-based fuzzy logic controller for quadrotor trajectory tracking. *IEEE Access*. 2026;14:66711-66727.
<https://doi.org/10.1109/ACCESS.2026.3688552>

- [15] Abbas NH, Sami AR. Tuning of PID controllers for quadcopter system using hybrid memory based gravitational search algorithm - particle swarm optimization. *International Journal of Computer Applications*. 2017;172(4):9-18.
- [16] Ojo KE, Adegbola OA, Aborisade DO. Comparative of Ziegler Nichols, fuzzy logic and extremum seeking based proportional integral derivative controller for quadcopter unmanned aerial vehicle stability control. *Journal of Electrical, Control and Technological Research*. 2021;3:1-10.
- [17] Koch W, Mancuso R, West R, Bestavros A. Reinforcement learning for UAV attitude control. *Transactions on Cyber-Physical Systems*. 2019;3(2):22. <https://doi.org/10.1145/3301273>
- [18] Bou-Ammar H, Voos H, Ertel W. Controller design for quadrotor UAVs using reinforcement learning. In: 2010 IEEE International Conference on Control Applications; 2010 Sep 8-10. Yokohama, Japan: IEEE; 2010. p. 2130-5. <https://doi.org/10.1109/CCA.2010.5611206>
- [19] Xia CL. Permanent magnet brushless DC motor drives and controls. John Wiley & Sons Singapore Pte. Ltd.; 2012.
- [20] Kose O, Oktay T. Dynamic modeling and simulation of quadrotor for different flight conditions. *European Journal of Science and Technology*. 2019;15:132-42. <https://doi.org/10.31590/ejosat.507222>
- [21] Mustapa MZ. Altitude controller design for quadcopter UAV. *Jurnal Teknologi (Science & engineering)*. 2015;74(1):181-8.
- [22] Ahmad F, Kumar P, Bhandari A, Patil PP. Simulation of the quadcopter dynamics with LQR based control. *Materials Today Proceedings*. 2020; 25(part 2): 326-32. <https://doi.org/10.1016/j.matpr.2020.04.282>
- [23] Priyambodo TK, Dhewa OA, Susanto T. Model of linear quadratic regulator (LQR) control system in waypoint flight mission of flying wing UAV. *Journal of Telecommunication, Electronic and Computer Engineering*. 2020;12(4):43-9.
- [24] Abbas NH, Sami AR. Tuning of PID controllers for quadcopter system using cultural exchange imperialist competitive algorithm. *Journal of Engineering*. 2018;24(2):80-99. <https://doi.org/10.31026/j.eng.2018.02.06>
- [25] Indrawati V, Prayitno A, Kusuma TA. Waypoint navigation of AR.Drone quadrotor using fuzzy logic controller. *Telkomnika*. 2015;13(3):381–91.
- [26] Fnu VS, Cohen K. Intelligent fuzzy flight control of an autonomous quadrotor UAV. In: 52nd Aerospace Sciences Meeting; 2014 Jan 13-17. National Harbor, Maryland: AIAA; 2014. <https://doi.org/10.2514/6.2014-0992>
- [27] Rodríguez-Abreo O, Rodríguez-Reséndiz J, García-Cerezo A, García-Martínez JR. Fuzzy logic controller for UAV with gains optimized via genetic algorithm. *Heliyon*. 2024;10(4):e26363. <https://doi.org/10.1016/j.heliyon.2024.e26363>
- [28] Beloglazov DA, Kobersi IS, Kosenko EY, Solovyev VV, Shadrina VV. Analysis of application aspects of regulators in quadrotor automatic control. *Engineering Journal of Don*. 2015;3:3078.
- [29] Indrawati V, Prayitno A, Utomo G. Comparison of two fuzzy logic controller schemes for position control of AR.Drone. In: 2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE); 2015 Oct 29-30. Chiang Mai, Thailand:

- IEEE; 2015. p. 360-3.
<https://doi.org/10.1109/ICITEED.2015.7408972>
- [30] Polat O, Sezgin A. Position control of a quadcopter with PID and fuzzy-PID controller. *Journal of Engineering Sciences and Design*. 2024;12(1):34-48.
<https://doi.org/10.21923/jesd.1223998>
- [31] Vadakkepat P, Chong TC, Arokiasami WA, Weinan X. Fuzzy logic controllers for navigation and control of AR.Drone using Microsoft Kinect. In: 2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE); 2016 Jul 24-29. Vancouver, BC, Canada: IEEE; 2016. p. 856-63. <https://doi.org/10.1109/FUZZ-IEEE.2016.7737778>