Innovative Fuzzy PID Algorithm: Advancing Quadcopter Control with Adaptive Precision

Zhenghang Guo*

School of Science and Technology, North China University, Hebei Tangshan, China

*Corresponding Author Email: 1029739193@qq.com

Abstract. The integration of fuzzy logic techniques into PID controllers shows promising advantages in quadcopter control. By incorporating fuzzy logic, the PID controller achieves enhanced adaptability and robustness, resulting in more dynamic and resilient control performance. This is achieved through the automatic adjustment of PID parameters based on real-time system feedback, enabling a control strategy that closely resembles human reasoning. Overall, the combination of fuzzy logic and PID control offers significant potential for improving quadcopter control in various environmental conditions.

Keywords: PID controller, Fuzzy logic, Quadcopter control.

1. Introduction

In recent years, the utilization of quadcopters has significantly increased in various applications, ranging from aerial photography and surveillance to parcel delivery and search and rescue missions. Efficient control of quadcopters is essential to ensure stable flight and accurate maneuvering in dynamic environments. Traditional Proportional-Integral-Derivative (PID) controllers have been widely employed for quadcopter control due to their simplicity and effectiveness in tracking desired trajectories. However, PID controllers have limitations in adapting to uncertainties and variations in environmental conditions, leading to suboptimal performance in complex scenarios.

To address these challenges, researchers have turned to fuzzy logic techniques to enhance PID controllers' adaptability and robustness. Fuzzy logic offers a more intuitive approach to control by incorporating linguistic variables and IF-THEN rules, enabling controllers to mimic human-like reasoning. This integration allows for automatic adjustment of PID parameters based on real-time system feedback, improving control performance in varying conditions.

Previous studies have demonstrated the effectiveness of fuzzy PID controllers in various control applications, including robotic systems, industrial processes, and unmanned aerial vehicles. However, there remains a need for further exploration and optimization of fuzzy PID controllers, particularly in the context of quadcopter control, where precise trajectory tracking and stability are crucial.

In this paper, we propose a fuzzy PID controller tailored for quadcopter control, aiming to improve adaptability, robustness, and performance compared to traditional PID controllers. We integrate fuzzy logic techniques into the PID control framework to enable dynamic adjustment of control parameters based on environmental conditions and system feedback. Through simulation studies and experimental validation, we aim to demonstrate the effectiveness and potential superiority of the proposed fuzzy PID controller in quadcopter control applications.

2. PID controller

In cascade control systems, PID type controllers are frequently utilised to ensure accurate tracking and good performance [2]. The PID controller minimises this discrepancy by modifying the control variable in this process, which involves computing the difference between the setpoint and the intended setpoint [1][2]. Equation (1) defines the link between the PID controller output and the system error, whereas Equation (2) shows the discrepancy between the desired and actual outputs [19][21][22][25].

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \quad (1)$$

$$e(t) = \text{desired}\ (t) - \text{generated}\ (t) \quad (2)$$

Where are the integral gain, derivative gain, and proportional gain? While increasing can lower the steady-state error, too high of a value could cause unstable oscillations. While increasing can lessen overshoot and the settling time of the feedback system output signal, increasing can also remove steady-state error in feedback systems [3][4]. The PID controller's block diagram is displayed in Figure 1.

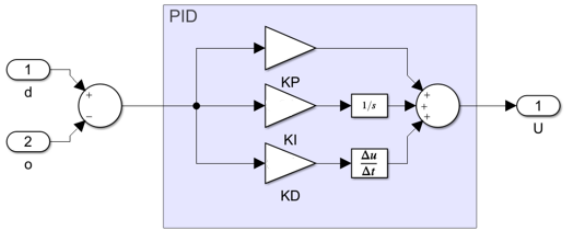


Fig.1. Simulink blocks of PID controller

3. PID adjustment method

PID controllers are unable to adapt and react to changes because they do not comprehend the process plant. As a result, for performance to be adequate, tuning must be done correctly. The Ziegler-Nichols approach, which is separated into step response and frequency response rules, is one of the most widely utilised techniques. In both rules, the proportional gain is increased to a point where the system oscillates while the integral and derivative gains are initially set to zero [5][6].

But this technique cannot be used to tune systems that do not exhibit prolonged oscillation. In this work, the Ziegler-Nichols method was not able to produce persistent oscillation, hence the developed model uses a manual tuning strategy to modify the PID parameters. The transient response of the system is impacted by increasing each PID gain; Table 1 summarises these effects. Table 2 displays the PID parameters that were determined using the specified rules.

The following guidelines apply when manually adjusting PID parameters [7][8]:

1. Make and equal zero.
2. Intensify until oscillations happen entirely or nearly entirely. Setting a long simulation time is advised if full oscillation cannot be attained.
3. Increase only a little bit to bring the oscillation time down to one cycle. It is possible to employ simulations that have shorter stop times.
4. Increase until there is just one oscillation around the setpoint and steady-state is almost at the setpoint.
5. Keep following these guidelines until the intended result is obtained.

Table 1. The effect of increasing the PID gains value

Parameters	Kp	Ki	Kd
Overshoot(%)	Increase	Increase	Decrease
Rise time(s)	Decrease	Decrease	Minor change
Settling time(s)	Minor change	Increase	Decrease
Steady-state error	Decrease	Eliminate	No effect(Theoretically)
Stability	Degrade	Degrade	Improved(Small Kd)

Table 2. The gains of PID parameters for PID controller

Variable	Kp	Ki	Kd
Position (x,y), Altitude (z)	0.50	0.10	1.00
Attitude (ϕ, θ, ψ)	1.00	0.01	0.10

4. Fuzzy Controller

Manually tuning methods are beneficial for PID controllers because they offer intuitive settings and minimal process knowledge requirements. Nevertheless, obtaining suitable gains through manual tuning can be time-consuming, and it's challenging to determine if the final settings are optimal. Additionally, there's no guarantee that the system will reach a stable solution, posing a risk to the entire factory. Due to PID controllers' limited understanding of process plants, they cannot automatically adjust when the system undergoes changes. Therefore, heuristic methods, particularly fuzzy logic techniques, are integrated into PID controllers to ensure desired outputs are achieved and parameters are automatically adjusted when system changes occur.

Fuzzy logic presents a potent approach that operates without the need for exact mathematical models [10]. Drawing from human insights into the system during both control design and decision-making phases, fuzzy control relies on fuzzy set theory, linguistic variables, and fuzzy inference [11][12]. Fuzzy control systems typically encompass fuzzification, a fuzzy rule base, fuzzy inference, and defuzzification. Figure 2 provides a visual representation of the typical components of a fuzzy inference system [13].

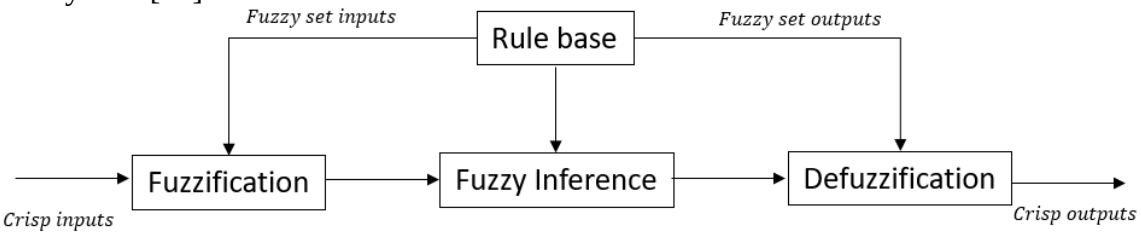


Fig. 2. Fuzzy inference system

For controlling the position, altitude, and attitude of quadcopters, the inputs consist of errors, representing the deviation between the desired trajectory and the actual trajectory, as well as the rate of change of errors. As depicted in Figure 2, these inputs, presented as crisp values, are inputted into the fuzzy inference system and converted into linguistic variables through the process of fuzzification. During fuzzification, predefined knowledge is utilized to establish the ranges of input variables, facilitating the mapping of these variables to discrete intervals.

Fuzzy logic controllers necessitate a rule base and fuzzy set outputs to adjust PID parameters. In this study, the chosen rule base type is the Mamdani-type, which constitutes a multi-input multi-output (MIMO) system, accepting 2 inputs and generating 3 outputs: K_{pf}, K_{if}, and K_{df}. For position and altitude control, the fuzzy set inputs are designated as $e(t) \in [-10,10]$ and $de(t) \in [-15,15]$, whereas for attitude control, they are denoted as $e(t) \in [-0.5,0.5]$ and $de(t) \in [-50,50]$. These regions are illustrated in Figure 3 (position and altitude) and Figure 4 (attitude).

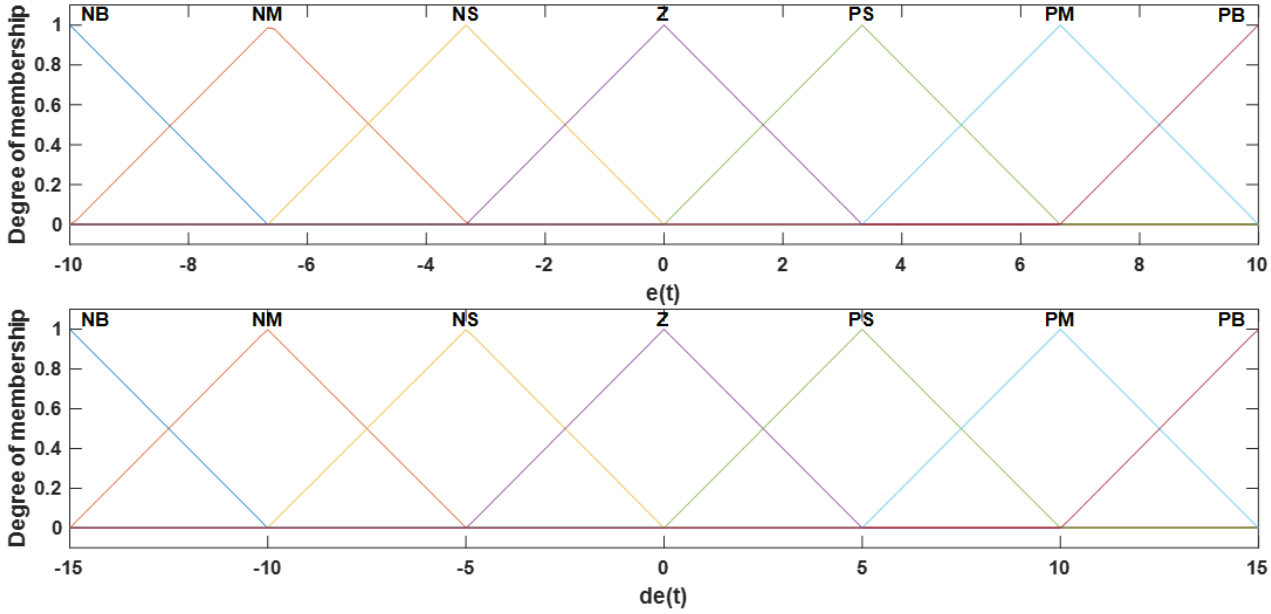


Fig.3. The inputs membership functions of position and altitude

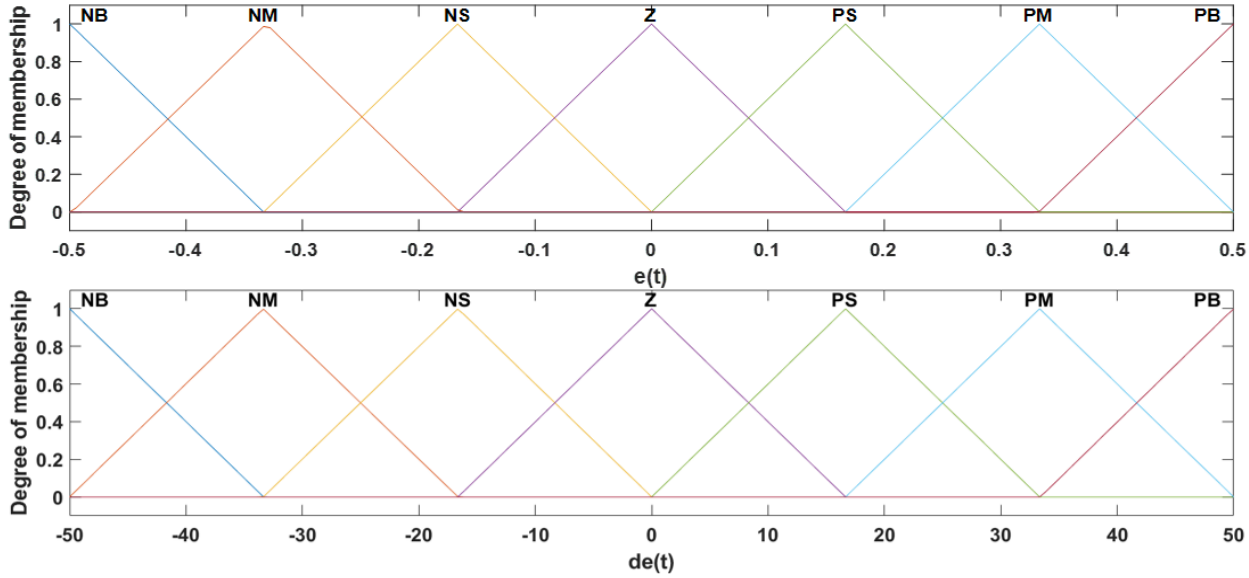


Fig.4. The imputs membership functions of attitude

Concurrently, the fuzzy set outputs are determined according to $K_p \in [K_{pmin}, K_{pmax}] K_i \in [K_{imin}, K_{imax}]$ and $K_d \in [K_{dmin}, K_{dmax}]$. For position and altitude, the fuzzy set outputs are specified as $K_p \in [0.5,5.0], K_i \in [0.1,1.0]$ and $K_d \in [1.0,10]$, as depicted in Figure 5. In contrast, for attitude, they are represented as $K_p \in [1.0,10], K_i \in [0.01,0.10]$ and $K_d \in [0.1,1.0]$, as demonstrated in Figure 6. These fuzzy sets are calibrated within the interval [0, 1], with triangular membership functions defining the degree of membership.

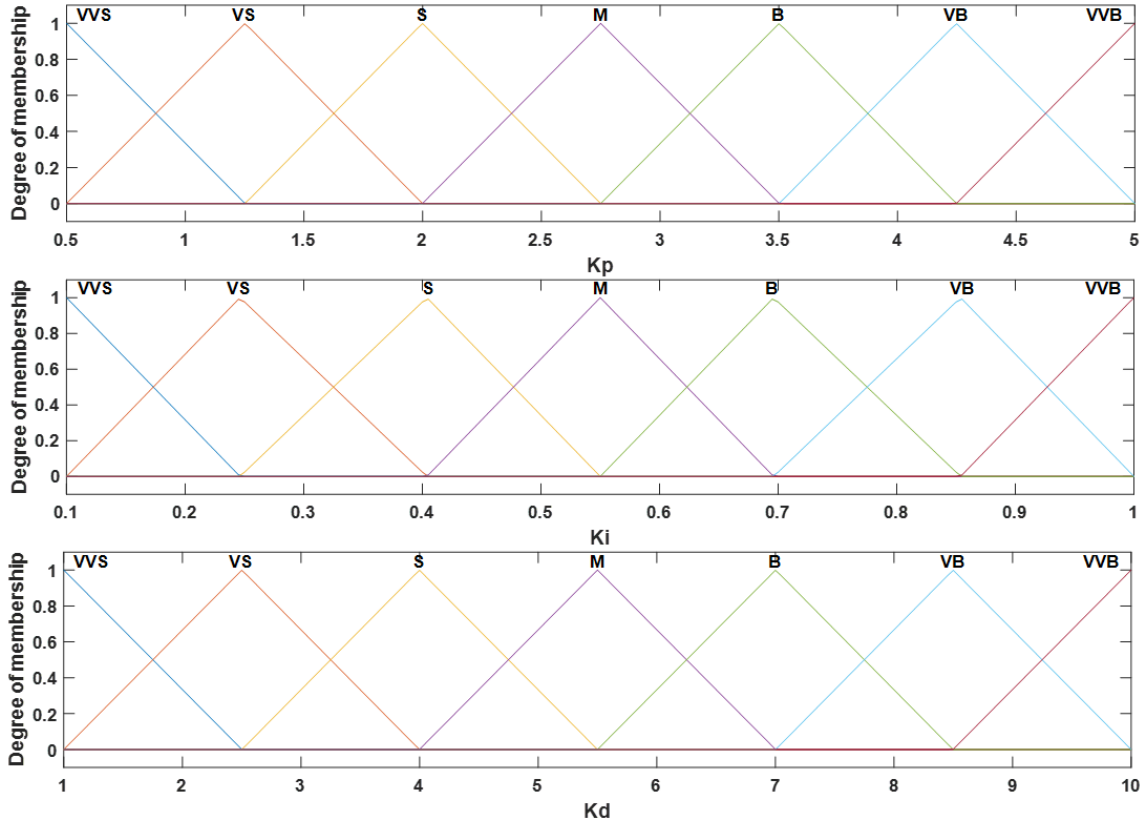


Fig.5. The output membership functions of position and altitude

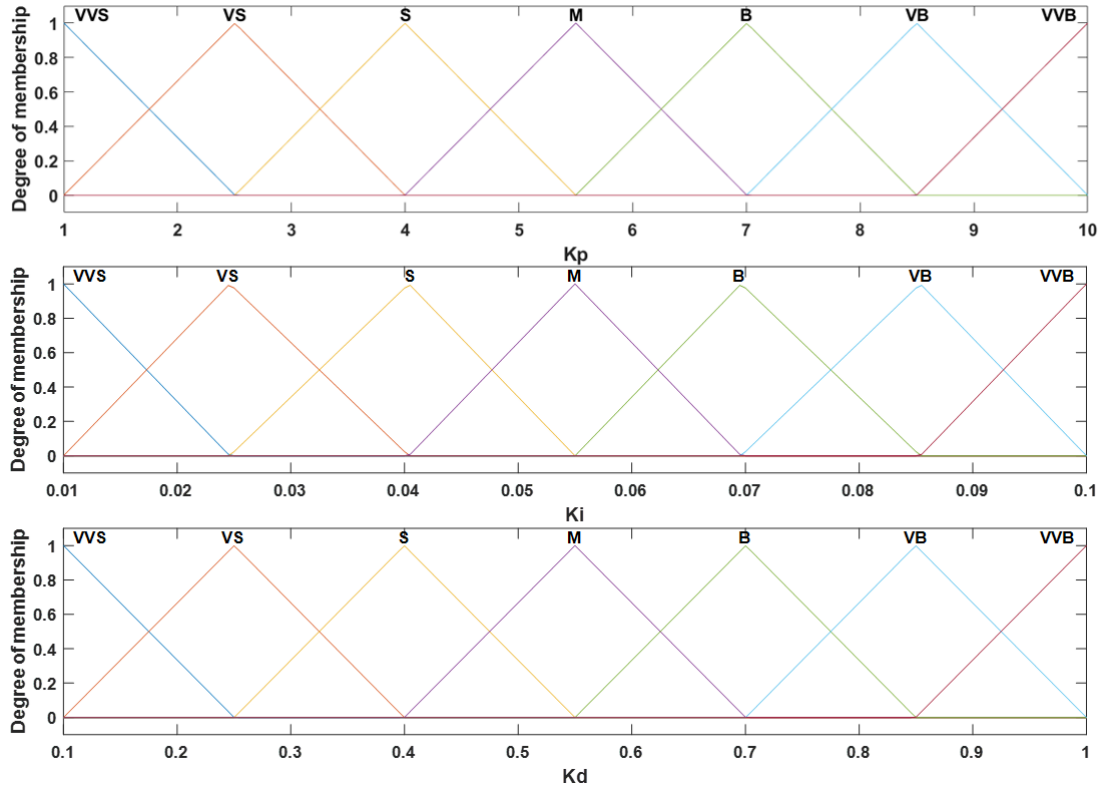


Fig.6. The output membership functions of attitude

The Mamdani-type fuzzy controller employs IF-THEN rules to construct linguistic representations of fuzzy logic, which are known as membership functions. Utilizing seven linguistic variables, 49 membership function rules are generated for each output. The membership function rules for the quadcopter are outlined in Table 3 and Table 4 [14].

The final step of the fuzzy controller involves defuzzification, where the outputs obtained from fuzzy inference are presented in linguistic form. To be inputted into the PID controller, these outputs need to be converted into precise numbers. For Mamdani-type fuzzy controllers, the centroid method is the most commonly used defuzzification technique, with the specific calculation defined in Equation (3).

$$\text{Output} = \frac{\sum_{i=1}^{q} \mu_j(w_i) \cdot w_i}{\sum_{i=1}^{q} \mu_j(w_i)} \tag{3}$$

Table 3. Fuzzy rules for Kp and Ki

Error changes,	Error, e						
&e	NB	NM	NS	Z	PS	PM	PB
NB	M	S	VS	VVS	VS	S	M
NM	B	M	S	VS	S	M	B
NS	VB	B	M	S	M	B	VB
Z	VVB	VB	B	M	B	VB	VVB
PS	VB	B	M	S	M	B	VB
PM	B	M	S	VS	S	M	B
PB	M	S	VS	VVS	VS	S	M

Table 4. Fuzzy rules for Kd

Error changes,	Error, e						
&e	NB	NM	NS	Z	PS	PM	PB
NB	M	B	VB	VVB	VB	B	M
NM	S	M	B	VB	B	M	S
NS	VS	S	M	B	M	S	VS
Z	VVS	VS	S	M	S	VS	VVS
PS	VS	S	M	B	M	B	VS
PM	S	M	B	VB	B	M	S
PB	M	B	VB	VVB	VB	B	M

5. Fuzzy PID Controller

The outputs produced by the fuzzy logic controller, K_{pf}, K_{if} and K_{df}, are integrated into the PID controller to constitute a fuzzy PID controller. The general block diagram of the fuzzy PID controller is depicted in Figure 7, while Equation (4) provides the definition of the fuzzy PID controller [9][14][15][16].

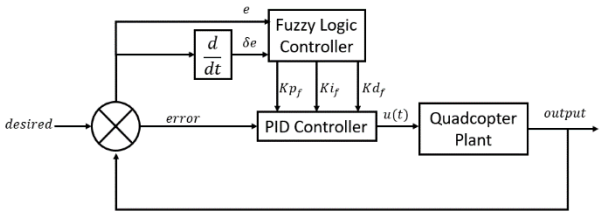


Fig.7. The block diagram of Fuzzy PID controller

$$u(t)_{FPID} = \left((K_p + K_{pf}) e(t) + (K_i + K_{if}) \int_0^t e(t)dt + (K_d + K_{df}) \frac{de(t)}{dt} \right) \tag{4}$$

The minimum and maximum ranges of PID controller gains are determined via trial-and-error simulations employing the PID controller. These values are subsequently applied in Equations (5), (6), and (7) to simulate the fuzzy PID controller, as illustrated in Figure 8 [9][15][17].

$$K_{p_f} = \frac{K_p - K_{p_{\min}}}{K_{p_{\max}} - K_{p_{\min}}} \quad (5)$$

$$K_{i_f} = \frac{K_i - K_{i_{\min}}}{K_{i_{\max}} - K_{i_{\min}}} \quad (6)$$

$$K_{d_f} = \frac{K_d - K_{d_{\min}}}{K_{d_{\max}} - K_{d_{\min}}} \quad (7)$$

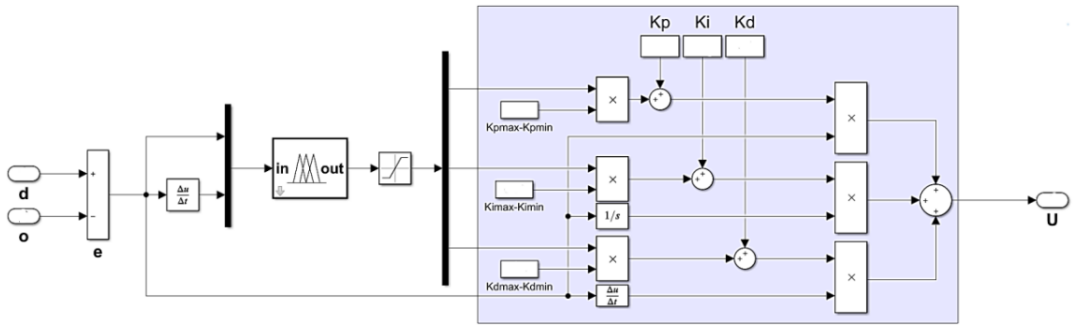


Fig.8. The Simulink blocks of self-tuning Fuzzy PID controller

1. Results and Discussion

Simulations employing the suggested controllers were performed on various trajectories within MATLAB Simulink, with a simulation time step of 0.01 seconds. Each trajectory underwent simulation utilizing both PID controllers and the proposed fuzzy PID controllers. Results were acquired through comparative analysis utilizing the Integral of Absolute Error (IAE) equation (8) and the Integral of Squared Error (ISE) equation (9) [10][18]. Additionally, the Root Mean Square Error (RMSE) of position was computed using equation (10) [40][41].

$$\mathrm{IAE}_{\mathrm{position}} = \int_0^t \left(|e_x| + |e_y| + |e_z|\right)dt \quad (8)$$

$$\mathrm{ISE}_{\mathrm{position}} = \int_0^t \left(e_x^2 + e_y^2 + e_z^2\right)dt \quad (9)$$

$$\mathrm{RMSE}_{\mathrm{position}} = \sqrt{\frac{1}{n_i}\sum_{i=1}^{n_i} \left((e_x^i)^2 + \left(e_y^i\right)^2 + (e_z^i)^2\right)} \quad (10)$$

In trajectory tracking scenarios, the anticipated roll and pitch are no longer specified since they are outcomes of the position control. Consequently, the focus shifts to defining the expected position, namely x and y. The definition for the expected altitude z remains consistent with altitude and attitude tracking, where the initial value is 0 and the expected value is 1. The expected yaw ψ is adjusted from 0.5 to 0. These predefined inputs serve as input values for circular, square, lemniscate, and zigzag trajectories. The simulation stop time for each trajectory is standardized at 180 seconds.

5.1. Circular Trajectory

The predefined desired positions for the circular trajectory are $x_d = 10\sin\left(\frac{\pi}{10}t\right)$ and $y_d = 10\cos\left(\frac{\pi}{10}t\right)$. These predetermined desired outputs are employed for both PID and fuzzy PID controllers. Figure 17 demonstrates the circular trajectory tracking outputs of the quadcopter utilizing PID controllers and fuzzy PID controllers.

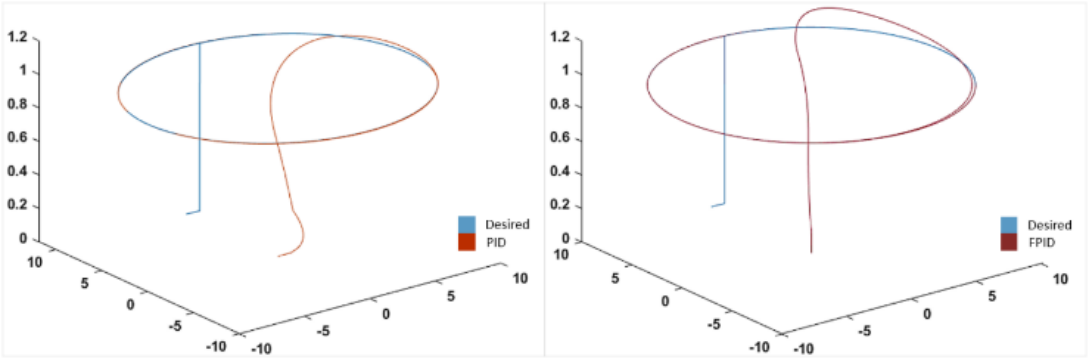


Fig.9. The outputs of circular tracking

Based on Figure 9, both controllers effectively tracked the desired trajectory, demonstrating notable enhancements in altitude tracking. Tracking information for both controllers is provided in Table 5. From the data presented, it is evident that the performance of both controllers is largely comparable. Nevertheless, the Fuzzy-PID controller exhibits improvements in altitude tracking, with a reduction in rise time by 31.18%. Significant enhancements are also observed in the y-position, with a decrease in rise time by 21.10% compared to the PID controller. Moreover, the overshoot percentage for the y-position decreased by 34.32%. Conversely, there is a marginal improvement in the rise time for the x-position, with a reduction of 0.21%.

Evaluation of both controllers' performance is conducted through tracking error analysis. The tracking error values for circular tracking are presented in Table 6. While improvements are primarily observed in the y-position and altitude, the employment of the Fuzzy PID controller leads to a significant decrease in IAE, ISE, and RMSE values.

Table 5. The step info of circular tracking

Controller	Variable	Rise time	Setting Time	Overshoot
PID	x	6.6517	179.5460	33.0210
	y	3.4664	179.5649	78.3456
	z	6.1436	22.2837	3.0739
FPID	x	6.6676	179.4681	33.1420
	y	2.8216	179.5860	51.4574
	z	4.2693	18.2178	3.8334

Table 6. The tracking errors of circular trajectory

Error Calculation	PID	FPID	Reduction(%)
LAE	119.3	72.27	39.62
ISE	584.2	319.6	45.33
RMSE	1.801	1.395	22.44

5.2. Square Trajectory

The position inputs necessary for tracking the square trajectory are defined as $x_d = [10,10,-10,-10,10]$ and $y_d = [10,-10,-10,10,10,$ where $x_d = [10,10,-10,-10,10]$ represents the x-position and $y_d = [10,-10,-10,10,10]$ represents the y-position. Figure 18 depicts the outputs of both the PID controller and the fuzzy PID controller.

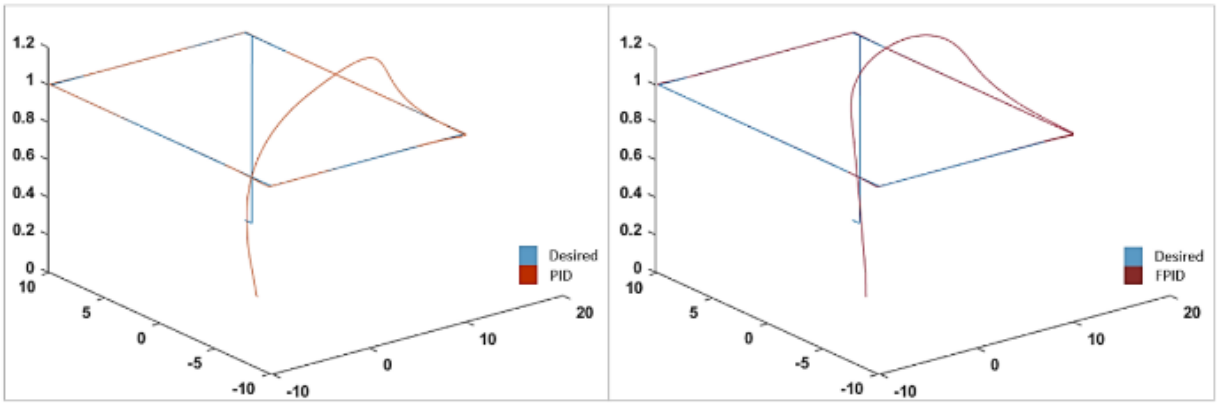


Fig.10. The outputs for square tracking

Both controllers effectively followed the desired trajectory. As depicted in Figure 8, apart from the elevated overshoot produced by the Fuzzy PID controller, no notable distinctions were apparent. A comparison between the outcomes from Figure 12 and the data provided in Table 7 confirms that the Fuzzy PID controller yields a higher overshoot percentage, showing an increase of 29.17% compared to the PID controller.

Table 7. The step info of circular tracking

Controller	Variable	Rise time	Setting Time	Overshoot
PID	x	12.0035	179.1964	1.7157
	y	10.8445	149.3956	1.9265
	z	6.15463	21.9731	2.9796
FPID	x	12.0173	179.2754	1.8702
	y	10.8972	149.3915	1.2713
	z	4.2658	18.2293	3.8494

The tracking error values for square trajectory tracking outlined in Table 8 suggest that incorporating Fuzzy techniques into the PID controller can enhance the system's performance, as all errors derived from the Fuzzy PID are inferior to those of the PID controller.

Table 8. The tracking errors of circular trajectory

Error Calculation	PID	FPID	Reduction(%)
LAE	189.3	141	25.22
ISE	1074	675.2	36.93
RMSE	2.445	1.934	20.54

5.3. Lemniscate Trajectory

The position inputs required for lemniscate trajectory tracking are specified as $x_d = 10\sin\left(\frac{\pi}{10}t\right)$ and $y_d = 10\cos\left(\frac{\pi}{20}t\right)$. The computational outcomes of both the PID controller and the fuzzy PID controller are illustrated in Figure 11.

Figure 11 displays nearly identical tracking of position x and y by both controllers, while discrepancies are evident in altitude tracking outputs. In comparison to the PID controller, the fuzzy PID controller showcases a larger altitude overshoot trajectory. This observation aligns with the data presented in Table 9, where the overshoot value produced by the fuzzy PID controller is 26.05% higher than that of the PID controller. Nonetheless, the rise time and settling time for altitude are reduced by 30% and 17.62%, respectively. Furthermore, there are slight enhancements in position x and y, with the rise time for position x decreased by 0.22% and for position y by 0.91%. The tracking output of the fuzzy PID outperforms that of the PID controller, a conclusion supported by the data in Table 10, where errors are diminished when employing the fuzzy PID controller in the simulation.

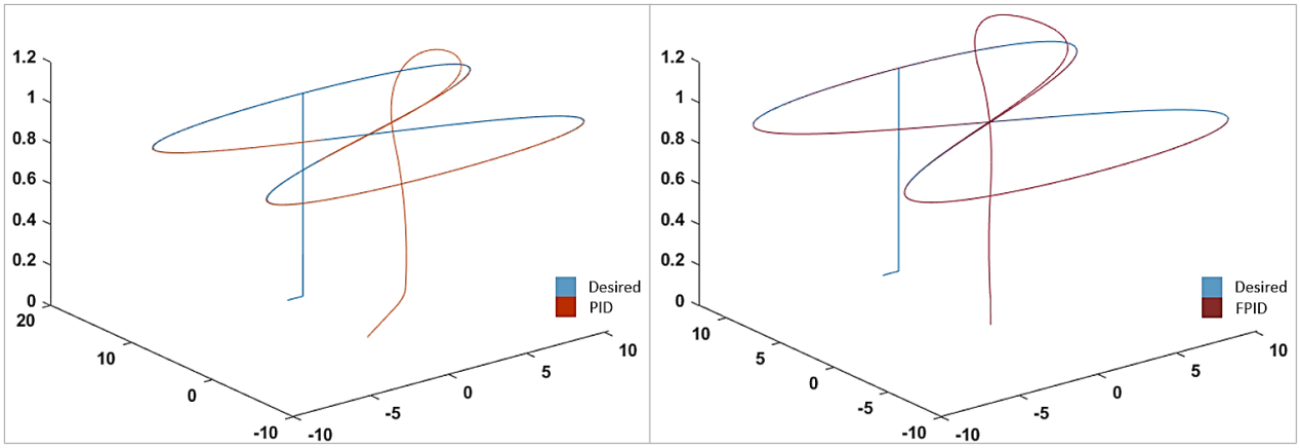


Fig.11. The outputs of lemniscate tracking

Table 9. The step info of circular tracking

Controller	Variable	Rise time	Setting Time	Overshoot
PID	x	6.68192	179.4444	33.0231
	y	17.4841	177.8955	9.76118
	z	6.16093	22.14436	3.06172
FPID	x	6.66753	179.4476	33.1402
	y	17.3254	178.0984	9.75358
	z	4.26895	18.24343	3.85924

Table 10. The tracking errors of circular trajectory

Error Calculation	PID	FPID	Reduction(%)
LAE	123.3	80.67	34.95
ISE	642.5	378.3	41.13
RMSE	1.891	1.452	23.22

5.4. Zigzag Trajectory

The desired position input for the zigzag trajectory is specified as $x_d = [4,4,10,10,4,4,-2,-2,-8,-8,-2,-2,4]$ and $y_d = [4,12,8,0,4,-4,0,-8,-4,4,0,8,4]$, where $x_d = [4,4,10,10,4,4,-2,-2,-8,-8,-2,-2,4]$ is a function of time. Figure 12 illustrates the outputs of both controllers.

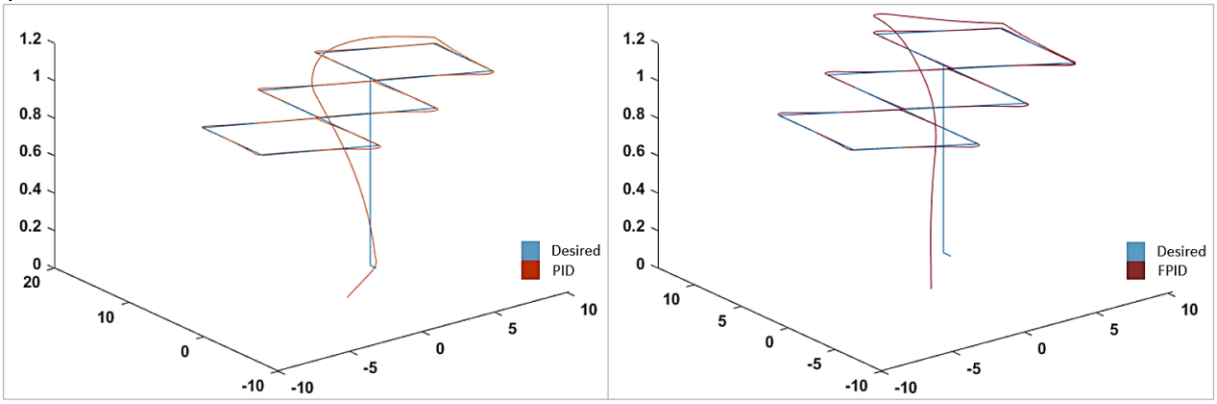


Fig.12. The outpurs of zigzag tracking

Based on Figure 12, both controllers effectively tracked the desired trajectory. There were no substantial disparities observed in the tracking of x and y positions. However, notable differences were evident in altitude tracking between the controllers, with the Fuzzy PID controller exhibiting a larger overshoot compared to the PID controller. This observation is corroborated by the data in Table

11, where employing the Fuzzy PID controller led to a 26% increase in the overshoot percentage for altitude. Nevertheless, the rise time and settling time increased by 31.24% and 15.36%, respectively.

Table 11 indicates variations in the rise time and overshoot percentage for x and y positions. The rise time for x position decreased by 0.51%, while the overshoot for y position decreased by 4.71%. A significant discrepancy in the rise time for x position between the controllers was noted, with a decrease of 88.54% recorded when using the Fuzzy PID controller. The tracking error values obtained from the Fuzzy PID controller were lower than those from the PID controller, as outlined in Table 12.

Table 11. The step info of circular tracking

Controller	Variable	Rise time	Setting Time	Overshoot
PID	x	2.6803	179.5939	304.4755
	y	0.1167	179.1856	1.8462e+04
	z	6.2077	21.5778	3.0283
FPID	x	2.6672	179.5886	307.6504
	y	0.0135	179.3122	1.7592e+04
	z	4.2686	18.2627	3.8186

Table 12. The tracking errors of circular trajectory

Error Calculation	PID	FPID	Reduction(%)
LAE	127.6	91.23	28.55
ISE	290.4	138.7	52.23
RMSE	1.271	0.878	30.77

5.5. Spiral Trajectory

For spiral trajectory tracking, the desired positions are defined as $x_d = 10\sin\left(\frac{\pi}{10}t\right)$ and $y_d = 10\cos\left(\frac{\pi}{10}t\right)$, akin to circular trajectory tracking, where z requires an increasing value. Thus, a ramp with a slope of 0.5 is designated as the desired value for z. The spiral tracking output is depicted in Figure 13.

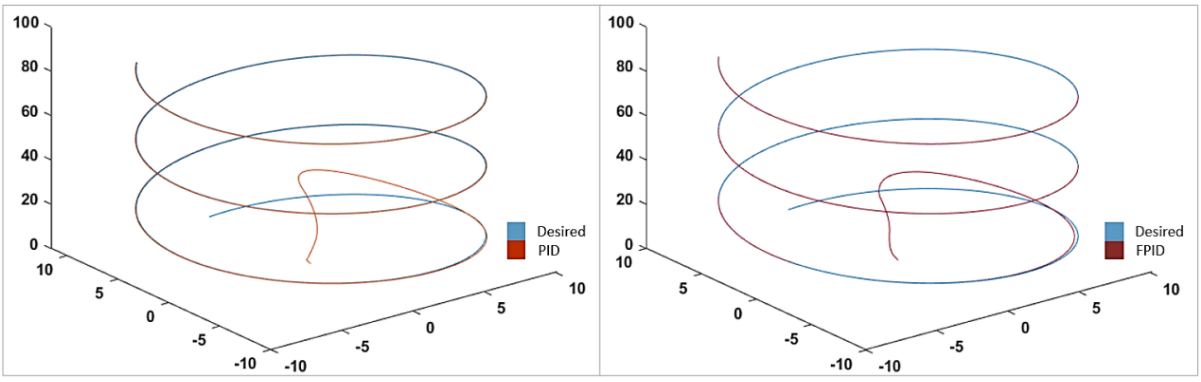


Fig. 13. The outputs of spiral trajectory tracking

With a reduced overshoot. The numerical data of overshoot percentages documented in Table 13 reveal that the Fuzzy PID controller exhibits an overshoot 0.35% higher than that of the PID controller in position and 30.54% lower in velocity. There are no notable distinctions between the controllers in other metrics. Nevertheless, the tracking error values outlined in Table 14 indicate that the Fuzzy PID controller yields superior tracking outcomes, evidenced by smaller error values.

Table 13. The step info of circular tracking

Controller	Variable	Rise time	Setting Time	Overshoot
PID	x	6.6794	179.4487	33.0255
	y	3.4213	179.5656	74.0704
	z	144.0482	176.4007	0
FPID	x	6.6667	179.4476	33.1442
	y	3.0753	179.5655	51.4517
	z	143.9653	176.4003	0

Table 14. The tracking errors of circular trajectory

Error Calculation	PID	FPID	Difference(%)
LAE	98.76	72.25	26.65
ISE	439.2	319.3	27.33
RMSE	1.563	1.334	14.67

6. Conclusion

Our creation of the fuzzy PID controller has the potential for improved effectiveness compared to traditional PID controllers. By integrating fuzzy logic techniques, our controller can adapt more dynamically to changes in the environment and system dynamics. The linguistic variables and IF-THEN rules in the fuzzy logic system allow for a more intuitive and human-like control strategy, enabling smoother and more precise control actions. Additionally, the automatic adjustment of PID parameters based on fuzzy logic reasoning enhances the controller's robustness and adaptability, leading to better performance, especially in complex and uncertain systems like quadcopter control. Overall, the fuzzy PID controller has the potential to outperform traditional PID controllers by offering enhanced adaptability, robustness, and performance.

References

[1] A. Iyer and H. O. Bansal, "Modelling, Simulation, and Implementation of PID Controller on Quadrotors," 2021 International Conference on Computer Communication and Informatics (ICCCI), pp. 1-7, 2021, https://doi.org/10.1109/iccci50826.2021.9402301.

[2] A. Noordin, M. A. Mohd Basri, Z. Mohamed, and I. Mat Lazim, "Adaptive PID Controller Using Sliding Mode Control Approaches for Quadrotor UAV Attitude and Position Stabilization," Arabian Journal for Science and Engineering, vol. 46, no. 2, pp. 963-981, 2020, https://doi.org/10.1007/s13369-020-04742w.

[3] A. T. Bayisa and G. Li-Hui, "Controlling quadcopter altitude using PID-control system," International Journal of Engineering Research & Technology (IJERT), vol. 8, pp. 195-199, 2019, https://doi.org/10.17577/ijertv8is120118.

[4] M. Karahan and C. Kasnakoglu, "Modeling and Simulation of Quadrotor UAV Using PID Controller," 2019 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), pp. 1-4, 2019, https://doi.org/10.1109/ECAI46879.2019.9042043.

[5] V. V. Patel, “Ziegler-Nichols Tuning Method,” Resonance, vol. 25, no. 10, pp. 1385-1397, 2020, https://doi.org/10.1007/s12045-020-1058-z.

[6] S. M. Othman et al., "Position Tracking Performance with Fine Tune Ziegler-Nichols PID Controller for Electro-Hydraulic Actuator in Aerospace Vehicle Model," Journal of physics, vol. 2107, no. 1, pp. 012064-012064, 2021, https://doi.org/10.1088/1742-6596/2107/1/012064.

[7] N. H. Sahrir and M. A. Mohd Basri, "Modelling and Manual Tuning PID Control of Quadcopter," in Control, instrumentation and mechatronics: Theory and practice, pp. 346-357, 2022, https://doi.org/10.1007/978-981-19-3923-5_30.

[8] M. Nassim and A. Abdelkader, "Speed Control of DC Motor Using Fuzzy PID Controller," arXiv preprint arXiv:2108.05450, 2021, https://doi.org/10.48550/arXiv.2108.05450.

[9] H. H. Manap, A. K. Abdul Wahab, and F. Mohamed Zuki, "Control for Carbon Dioxide Exchange Process in a Membrane Oxygenator Using Online Self-Tuning Fuzzy-PID Controller," Biomedical Signal Processing and Control, vol. 64, p. 102300, 2021, https://doi.org/10.1016/j.bspc.2020.102300.

[10] L. Farah, A. Haddouche, and A. Haddouche, "Comparison between proposed fuzzy logic and ANFIS for MPPT control for photovoltaic system," International Journal of Power Electronics and Drive Systems (IJPEDS), vol. 11, no. 2, p. 1065, 2020, https://doi.org/10.11591/ijpeds.v11.i2.pp1065-1073.

[11] Y. Di and Z. Chen, "Discussion on tuning method of quantization factor of fuzzy PID controller," Journal of physics, vol. 2232, no. 1, pp. 012007-012007, 2022, https://doi.org/10.1088/17426596/2232/1/012007.

[12] S. S. Kumar and G. Anitha, "A Novel Self-Tuning Fuzzy Logic-Based PID Controllers for Two-Axis Gimbal Stabilization in a Missile Seeker," International Journal of Aerospace Engineering, vol. 2021, pp. 1-12, 2021, https://doi.org/10.1155/2021/8897556.

[13] R. Baz, K. E. Majdoub, F. Giri, and A. Taouni, "Self-tuning fuzzy PID speed controller for quarter electric vehicle driven by In-wheel BLDC motor and Pacejka's tire model," IFAC-PapersOnLine, vol. 55, no. 12, pp. 598-603, 2022, https://doi.org/10.1016/j.ifacol.2022.07.377.

[14] E. Güçlü, İ. Aydın, and E. Akın, "Fuzzy PID Based Autonomous UAV Design for Railway Tracking," 2021 International Conference on Information Technology (ICIT), pp. 456-461, 2021, https://doi.org/10.1109/icit52682.2021.9491687.

[15] D. Nimawat and N. Nagar, "Self-Tuning Fuzzy Pid Controllers Application on Industrial Hydraulic Actuator Using Unique System Identification Approach," Journal of Communications Technology, Electronics and Computer Science, vol. 22, pp. 1-6, 2019, https://doi.org/10.22385/jctecs.v22i0.283.

[16] M. Rabah, A. Rohan, Y.-J. Han, and S.-H. Kim, "Design of Fuzzy-PID Controller for Quadcopter Trajectory-Tracking," International Journal of Fuzzy Logic and Intelligent Systems, vol. 18, no. 3, pp. 204-213, 2018, https://doi.org/10.5391/ijfis.2018.18.3.204.

[17] R. Kristiyono and W. Wiyono, "Autotuning Fuzzy PID Controller for Speed Control of BLDC Motor," Journal of Robotics and Control (JRC), vol. 2, no. 5, 2021, https://doi.org/10.18196/jrc.25114.

[18] M. Rabah, A. Rohan, Y.-J. Han, and S.-H. Kim, "Design of Fuzzy-PID Controller for Quadcopter Trajectory-Tracking," International Journal of Fuzzy Logic and Intelligent Systems, vol. 18, no. 3, pp. 204-213, 2018, https://doi.org/10.5391/ijfis.2018.18.3.204.