Improvement of PID Algorithm by Fuzzy Control: Comparative And Analysis

Yinghan Leng

School of Advanced Technology, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China

Yinghan.leng22@student.xjtlu.edu.cn

Abstract. This study addresses the challenge of enhancing conventional PID control algorithms to cope wit h the complexities of modern industrial systems, characterized by significant interference and errors. Despite its long history and widespread use, PID control has limitations in such contexts. The primary goal is to investigate how integrating fuzzy control techniques can improve the performance of traditional PID algorithms. Fuzzy control is seen as a more advanced strategy capable of handling complex systems effectively. The study conducts a thorough literature review to understand previous research in fuzzy control. Simulation tools like Falstad and MATLAB are then utilized to compare PID and fuzzy control algorithms in controlling DC motors, with mathematical models and simulation circuits for validation. Simulation outcomes demonstrate that integrating fuzzy control with PID algorithms leads to faster convergence towards target values and improved stability, particularly in complex systems like DC motors.While traditional PID controllers are widely used for their simplicity, this study suggests that incorporating fuzzy control techniques can enhance their performance in managing higher-level system control tasks. Moreover, it hints at the potential of integrating AI technologies, such as deep learning algorithms, to further refine traditional control methodologies, advancing industrial automation efficiency and adaptability.

Keywords: Fuzzy Control, PID Control, Falstad, MATLAB

1 Introduction

PID control theory has a very long history in the field of automatic control. James Watt invented the first steam engine and governor in 1769, which is generally accepted as the first negative feedback device. In 1868 Maxwell developed a mathematical model for a steam engine governor. It was not until 1911 that Elmer Sperry developed the first real PID controller for the US Navy [1]. The basic flow chart of PID control is as shown in the Fig. 1. The PID controller will perform three different processings on the error, including proportion (P), integral (I), and differential (D). The proportional term will proportionally amplify the output error

Y. Yue (ed.), *Proceedings of the 2024 International Conference on Mechanics, Electronics Engineering and Automation (ICMEEA 2024)*, Advances in Engineering Research 240,
https://doi.org/10.2991/978-94-6463-518-8_51

and provide feedback. The integral term accumulates errors to achieve rapid correction in the case of sustained large errors. The differential term can monitor the rate of change of the error and perform timely calibration when an abnormality occurs.

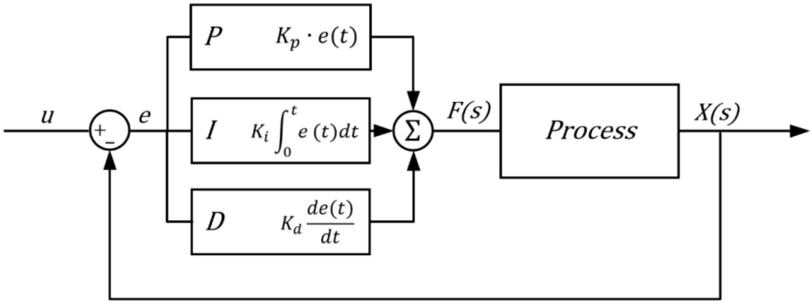

Fig. 1. PID control flow block diagram [1]

To this day, PID controllers are still used in most control fields due to their simple structure, easy implementation and maintenance, and low difficulty in getting started without losing efficiency [2]. Including but not limited to motor control, motion control, electricity, hydraulics, manufacturing and other industries. And the application of PID control can also be widely seen in the current emerging autonomous vehicles, drones, AI robots, etc.

Fuzzy control is a mathematical model of language analysis specifically designed for systems or processes that are more complex and susceptible to interference from more factors and cause larger errors. It was first proposed by Zadeh L.A in 1965. Fuzzy control is an advanced control strategy based on fuzzy mathematics, using language rule representation methods and advanced computer technology, and making decisions based on fuzzy reasoning. Compared with traditional PID control, it can better deal with systems that are more complex or susceptible to more interference.

Regarding fuzzy control, many people have done relevant research. The earlier research can be traced back to the fuzzy control of automatic train stop made by Hirokazu IHARA and others [3]. And the fuzzy control of factory kiln systems developed by Stephen E. Sheridan's team [4]. The more recent related research includes the fuzzy position servo system for stepper motors published by Shi Jingzhuo et al. [5], the fuzzy PID control for kinematic control of pneumatic systems proposed by Parnichkun M. [6], and Bingwei Qu The fuzzy PID based on genetic algorithm is proposed to optimize the speed regulation of stepper motor [7]. These studies on fuzzy control algorithms all express the same point of view, that is, fuzzy control does have better control performance. Taking the kinematic control of pneumatic systems studied by Parnichkun M as an example, the author conducted a series of experiments to compare the performance of different control algorithms. The experimental results showed that the PID controller mixed with fuzzy control performed best and could not only reach the piston faster. The target position has higher accuracy and can be recovered in time after being disturbed by external interference.

Regarding the traditional PID control algorithm and the more advanced fuzzy algorithm, this article will use tools such as Falstad and MATLAB to study and compare different control algorithms in controlling DC motors through simulation, and make corresponding evaluations.

2 Methods and Materials

2.1 Specific Methods and Establishment of Mathematical Models

In this study, simulation is mainly used for theoretical verification. The tools used are mainly Falstad and MATLAB for simulation. In Falstad, an operational amplifier circuit is used to build a PID control circuit, and in MATLAB, its fuzzy function is used for fuzzy control. Model construction and simulation.

First, we need to build a mathematical model of the DC motor. Referring to the equivalent circuit model proposed by A. Altay and A. B. Yildiz for analyzing DC motors [8], we can mathematically model the DC motor and convert it into a transfer function in the following form:

The left side of the formula is the ratio of the rotation angle and the input voltage, the right side Ra and La are the resistance and inductance in the armature circuit, s represents the complex variable, J is the torque, and Kb and Km are constants.

$$\frac{\theta(s)}{V_a(S)} = \frac{K_m}{s[(R_a+L_a s)(Js+b)+K_b K_m]} \tag{1}$$

Next, parameter selection was carried out. After many tests, a parameter that was more suitable for this study was selected, and finally the following transfer function was obtained, where S is still a complex variable:

$$\frac{\theta(s)}{V_a(S)} = \frac{3}{0.1s^2+s} \tag{2}$$

2.2 Falstad Modeling and Simulation

From this transfer function, you can use an operational amplifier in Falstad to build a circuit as shown in the Fig. 2 below. The left half is the main body of the motor, but since the output is negative, an additional inverting voltage follower is added to convert the output value. Convert to a positive number.

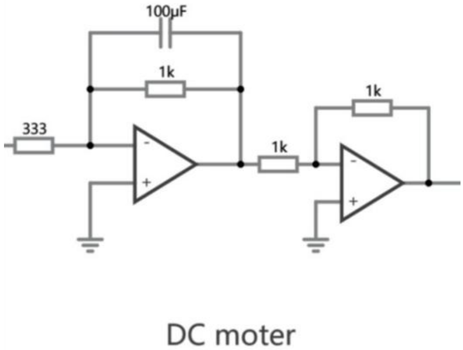

Fig. 2. DC motor equivalent circuit model (Photo credited: Original)

Next, the entire simulation circuit is built. As shown in the Fig. 3, an integrator is added to the right side of the DC motor model to convert the motor speed output into an angle output. The three branches on the left are the corresponding proportional, integral, and differential control branches. And there is an adder on the far left to obtain the error value for negative feedback. The PID controller built with operational amplifiers can calculate the values of KP, KI, and KD through the circuit and formula shown in the Fig. 4 below [9].

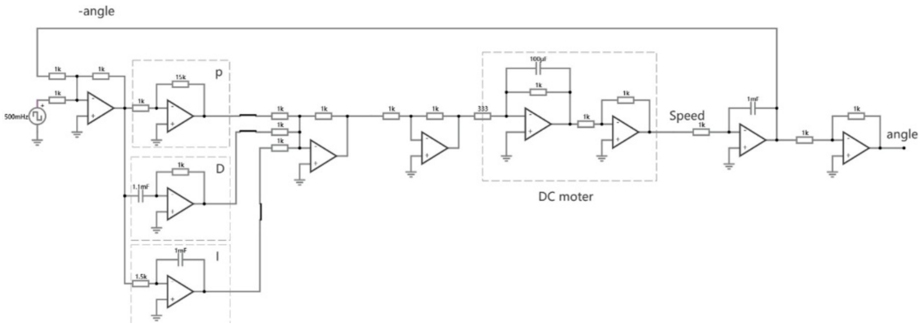

Fig. 3. DC motor simulation circuit with PID controller (Photo credited: Original)

$$u_0 =-\left(\frac{R_f}{R_1}+\frac{C_1}{C_f}\right)u_1-\frac{1}{R_1C_f}\int u_1dt-R_fC_1\frac{du_1}{dt} \quad (3)$$

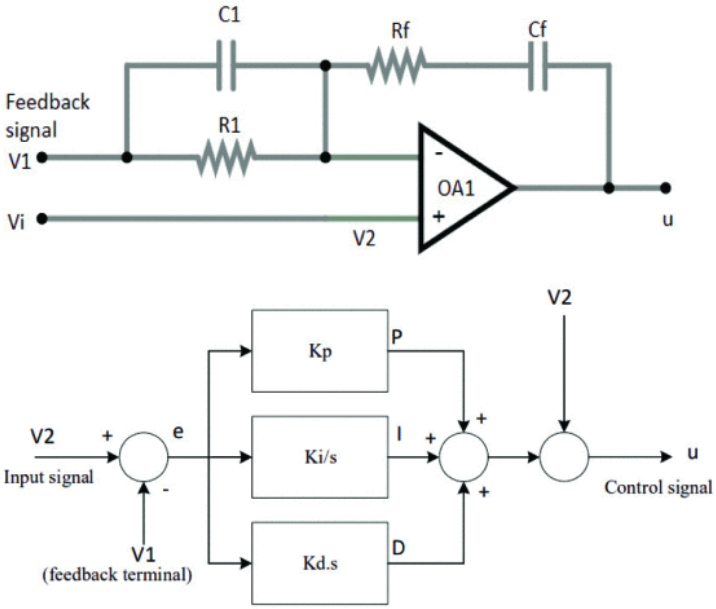

Fig. 4. Example circuit of PID control built by operational amplifier (Photo credited: Original)

2.3 Simulink modeling and simulation

For the fuzzy control part, this article studied the articles of James, Dave and L.A. Zadeh [10-12], and finally chose to use simulink for model establishment and simulation verification. As shown in Fig. 5, the fuzzy module that comes with MATLAB is used to establish a two-input and three-output fuzzy control, which performs proportional, integral and differential output respectively.

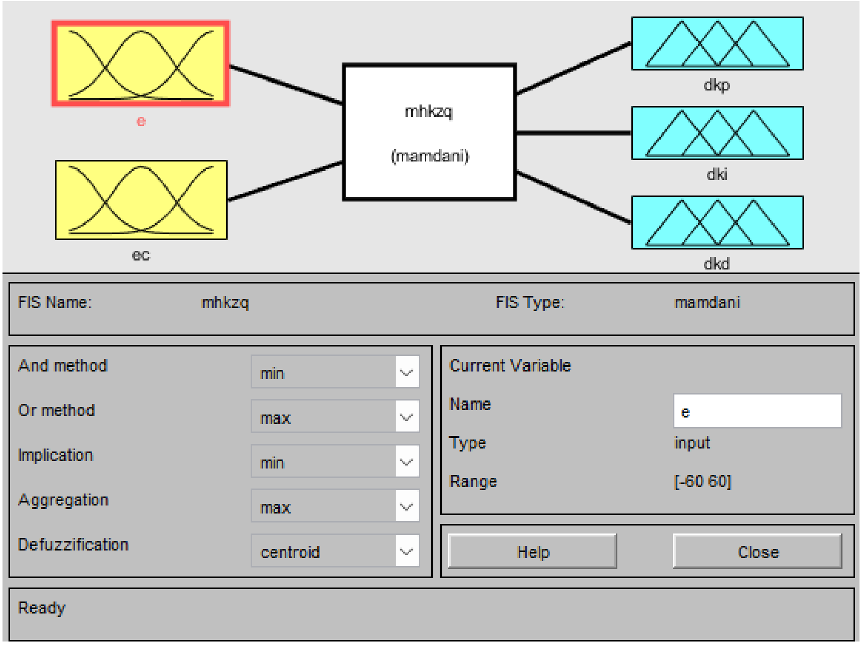

Fig. 5. Fuzzy controller settings (Photo credited: Original)

As shown in Fig. 6, this fuzzy controller uses a total of seven fuzzy subsets. The most extreme situations on both sides of the input part are set to z-type distribution, the middle part is triangular distribution, and the domain of e is [-100 100], and the domain of ec is is [-60 60]. The output part all asks for triangular distribution, and the domain value range of the three outputs is known to be the transfer function of the system that needs to be controlled, so it can be solved through the Routh criterion, which will not be described too much here.

For a fuzzy controller containing seven fuzzy subsets, a 7*7 fuzzy control table can be established. For fuzzy control PID, the proportional, integral, and differential fuzzy control tables are different. You need to refer to the following Table 1, Table 2 and Table 3 for fuzzy control rule settings.

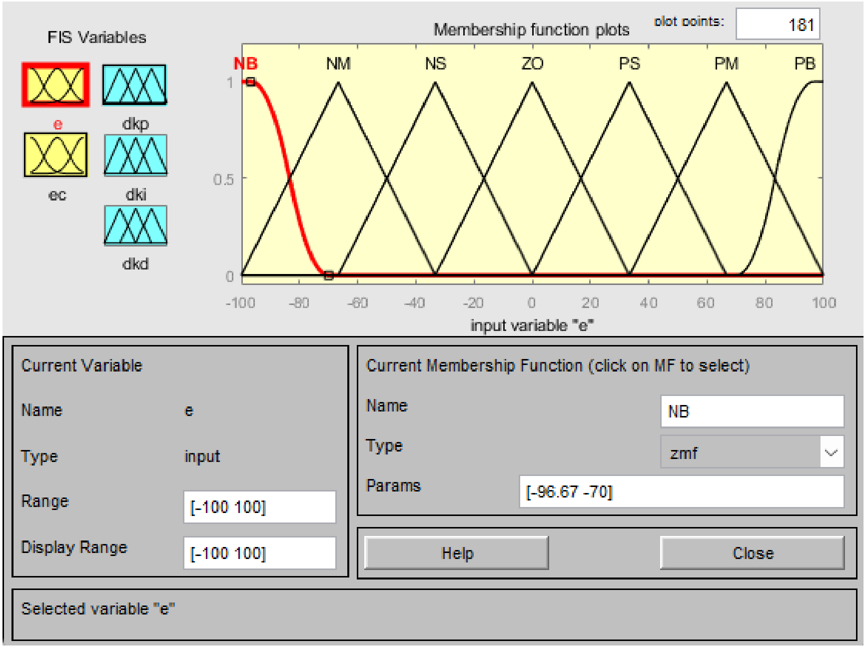

Fig. 6. Fuzzy controller settings(1) (Photo credited: Original)

Table 1. KP fuzzy control table

e	ec						
	NB	NM	NS	ZO	PS	PM	PB
NB	PB	PB	PM	PM	PS	ZO	ZO
NM	PB	PB	PM	PS	PS	ZO	NS
NS	PM	PM	PM	PS	ZO	NS	NS
ZO	PM	PM	PS	ZO	NS	NM	NM
PS	PS	PS	ZO	NS	NS	NM	NM
PM	PS	ZO	NS	NM	NM	NM	NB
PB	PB	ZO	NM	NM	NM	NB	NB

Table 2. Ki fuzzy control table

e	ec						
	NB	NM	NS	ZO	PS	PM	PB
NB	NB	NB	NM	NM	NS	ZO	ZO
NM	NB	NB	NM	NS	NS	ZO	ZO
NS	NB	NM	NS	NS	ZO	PS	PS
ZO	NM	NM	NS	ZO	PS	PM	PM
PS	NM	NS	ZO	PS	PS	PM	PB
PM	ZO	ZO	PS	PS	PM	PB	PB
PB	ZO	ZO	PS	PM	PM	PB	PB

Table 3. Kd fuzzy control table

e	ec						
	NB	NM	NS	ZO	PS	PM	PB
NB	PS	NS	NB	NB	NB	NM	PS
NM	PS	NS	NB	NM	NM	NS	ZO
NS	ZO	NS	NM	NM	NS	NS	ZO
ZO	ZO	NS	NS	NS	NS	NS	ZO
PS	ZO	ZO	ZO	ZO	ZO	ZO	ZO
PM	PB	NS	PS	PS	PS	PS	PB
PB	PB	PM	PM	PM	PS	PS	PB

The Simulink model construction begins with setting the system's target value at 1. The error value undergoes fuzzy processing before being fed into the fuzzy controller. The fuzzy controller then generates the ultimate decision value based on its internal rules. Subsequently, this decision value is transmitted to the PID controller for further action. This sequential process enables the system to adjust its operations based on the input error, effectively integrating fuzzy logic with PID control for enhanced performance in controlling dynamic systems. The PID controller can be specifically expressed in the following form:

$$u(k) = K_p e(k) + K_I \sum_{i=0}^{k} e(i) + K_D e(k) \quad (4)$$

where kp, ki and kd are proportional, integral and differential coefficients respectively.

In order to control the controlled target, a traditional PID controller is added to the upper part for comparison, and a step function (pulse output) is added to the output part to simulate external interference to observe and compare the capabilities of the traditional PID controller and fuzzy PID control. The processor handles external interference factors. The complete circuit diagram is shown in Fig. 7 below. It can be seen that the target value set by the system is 1. The error value is fuzzy and then input into the fuzzy controller. After making a decision, the result is input into the pid controller and output after corresponding processing.

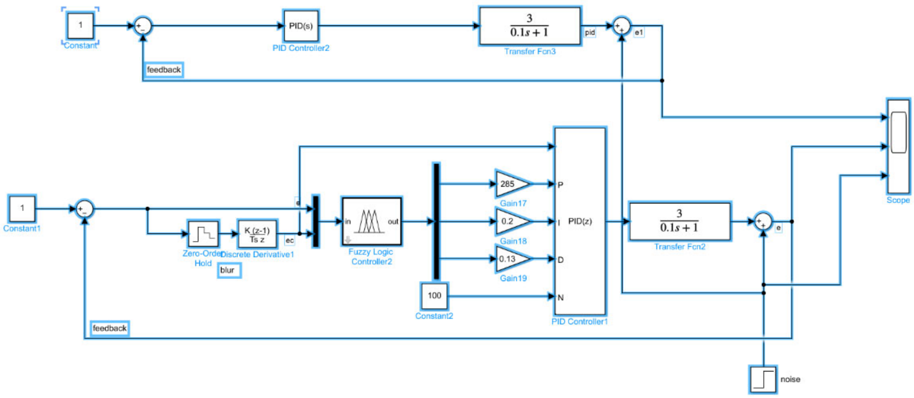

Fig. 7. Simulink modeling (Photo credited: Original)

Furthermore, to enhance the assessment of fuzzy control's efficacy in handling intricate systems, this research incorporated an additional third-order transfer function. This extension enabled simulation of more complex scenarios, offering deeper insights into the capabilities and adaptability of fuzzy control algorithms. The circuit diagram after modifying the transfer function is shown in Fig. 8 below.

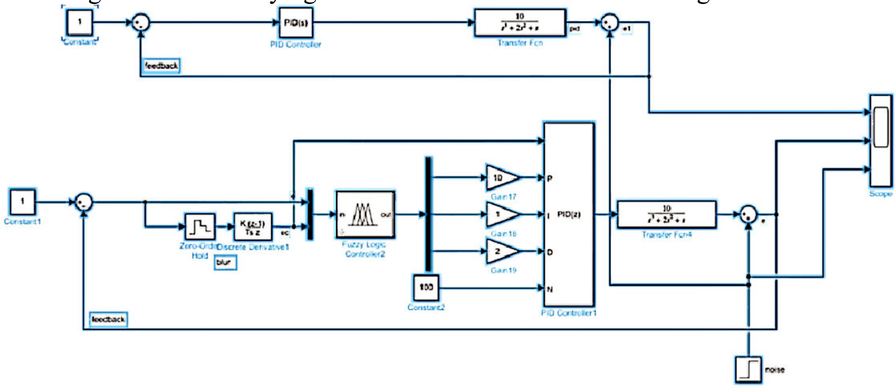

Fig. 8. Simulink modeling (1) (Photo credited: Original)

3 Results and Discussion

Figures 9-11 shows the simulation results of separate proportional control (P), proportional + differential control (PD) and proportional + integral + differential control (PID) in Falstad. As can be seen from the image, in only In proportional control, an excessively large KP value will lead to excessive overshoot and even uncontrollable oscillation in the end. And too small a KP value will cause the convergence to the target value to be too slow, so simple proportional control cannot achieve a more ideal control result. After adding differential control, the existence of KD can effectively suppress the oscillation caused by continuous overshoot, but it will also slow down the speed of approaching the target value. Therefore, it is necessary to increase KP at this time to quickly approach the target value and maintain stability. the goal of. After adding differential control, the possible instability of the system has been effectively suppressed, but at this time a new problem arises. There will be a small but long-term gap between the actual output value of the PD controller and the target value. At this time, the cumulative effect of integral control is needed to make up for the last error. At the same time, integral control can also improve the speed of the system approaching the target value and its stability.

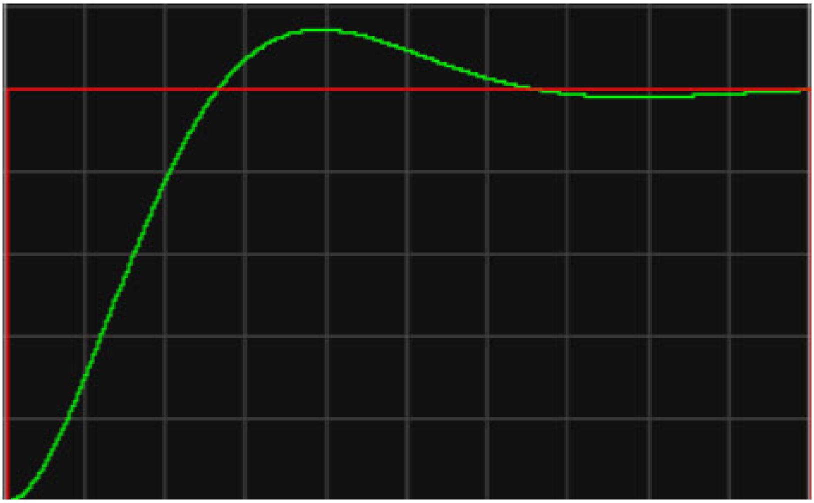

Fig. 9. Simulation result diagram under P control (Photo credited: Original)

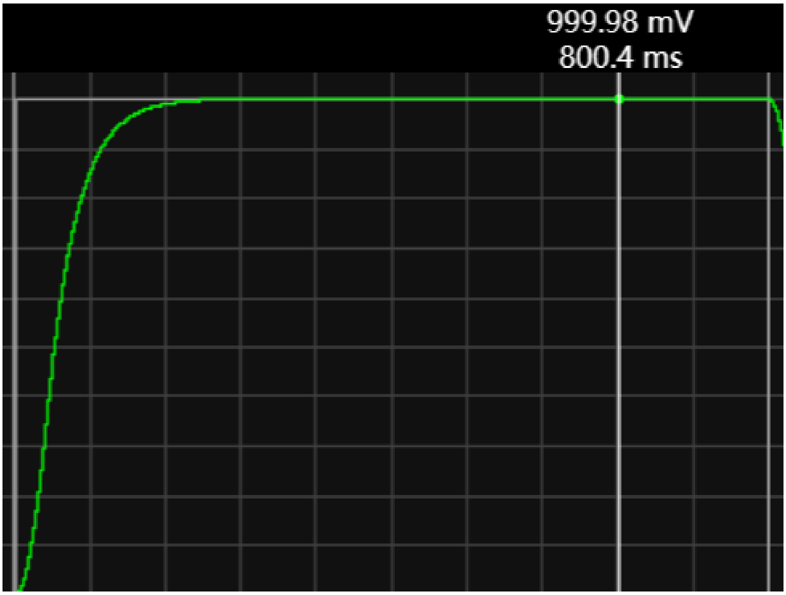

Fig. 10. Simulation result diagram under PD control(Photo credited: Original)

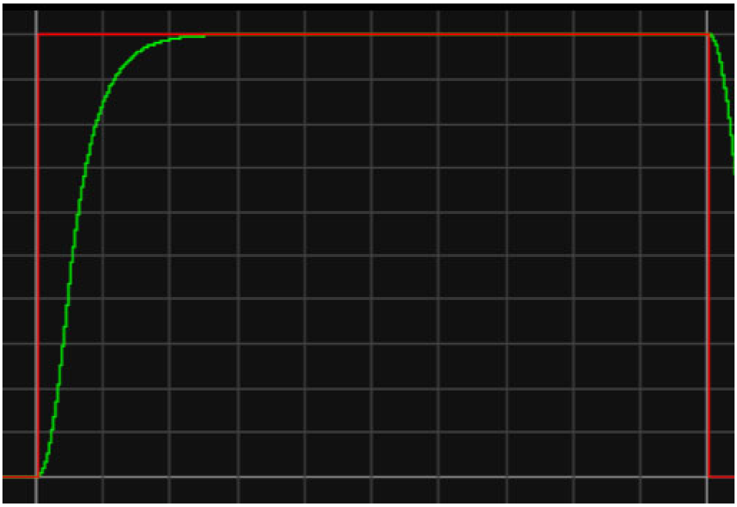

Fig. 11. Simulation result diagram under PID control (Photo credited: Original)

Regarding the MATLAB part, after debugging the parameters of the corresponding PID controller, we can get the results as shown in the Fig. 12 below. The blue lines in the figure are the results of traditional PID control, and the red lines are the results of PID control under fuzzy control. The yellow line is the external interference value. It can be clearly seen from the figure that under the system represented by the second-order transfer function, the PID control integrated with fuzzy control can approach the target value faster than the traditional PID, and the overshoot is smaller. At the same time, It can also respond faster to continuous external interference. However, the difference between the two is not very obvious, and it may be difficult to see a significant difference in performance in reality. So perhaps in relatively simple motor control requirements, the fuzzy control algorithm will only consume greater development costs, but in the end, a system state that is almost the same as the traditional PID will be obtained.

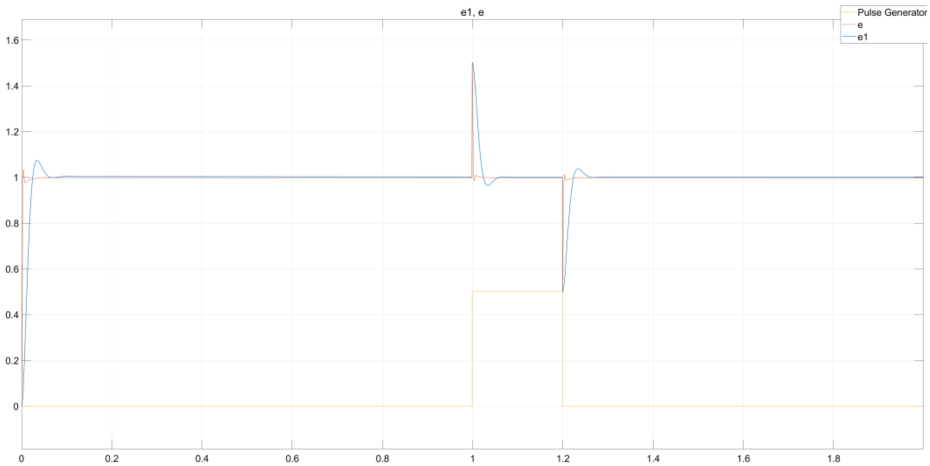

Fig. 12. Control of second-order transfer function by traditional PID algorithm and fuzzy control algorithm(Photo credited: Original)

When the controlled object becomes a more complex third-order transfer function, a significant gap appears between the two control methods. As shown in the Fig. 13 below, the traditional PID control algorithm is difficult to approach the target value and reach a steady state, and small changes in PID parameters will have a greater impact on the system. If traditional PID continues to be used to control the controlled object , the parameters of the controller will become extremely difficult to adjust, which will consume a lot of development costs. Moreover, its stability is poor. In actual conditions, there is a high probability that it will become unbalanced due to interference from external factors, which is something people do not want to see. But fortunately, the PID controller integrated with fuzzy control has good performance. It can still reach the target value quickly and maintain its stability just like the second-order transfer function expression system, and can quickly return to the original state after being disturbed by the outside world, all thanks to fuzzification processing. It is equivalent to ignoring a certain amount of error, so that complex systems will not be unable to remain stable due to some small disturbances.

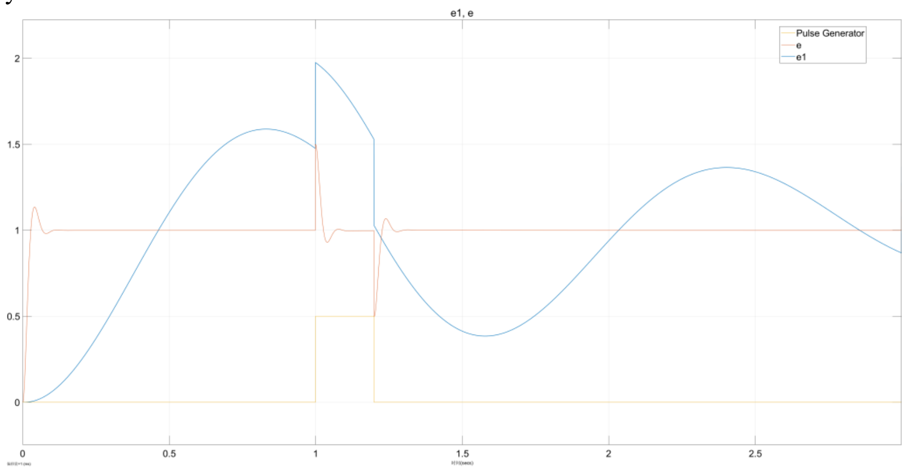

Fig. 13. Control of third-order transfer function by traditional PID algorithm and fuzzy control algorithm (Photo credited: Original)

4 Conclusion

This study endeavors to investigate the potential enhancement brought about by the fuzzy control algorithm to the performance of the conventional PID algorithm. Through an extensive literature review, coupled with mathematical modeling and circuit simulation, several insights have been gleaned. The ubiquitous use of the traditional PID controller across various domains stems from its simplicity and ease of implementation. However, it grapples with multifaceted complexities, particularly in managing systems with frequent and substantial external interferences, rendering it inadequate for higher-level system control. Fuzzy control algorithms offer a viable solution to address these challenges. Despite inherent limitations, such as insufficient system accuracy, fuzzy control demonstrates promising capabilities in augmenting traditional PID control, facilitating its application in higher-order systems. Moreover, fuzzy control proves simpler and more accessible, contributing to its wider adoption. Additionally, the integration of artificial intelligence technologies, such as deep learning algorithms, holds promise for further enhancing PID control performance. As science and technology continue to advance, leveraging such innovations can lead to significant improvements in traditional control methods, paving the way for more efficient and robust system control in various applications.

References

1. Bennett S. The past of PID controllers. Annual Reviews in Control, 25, 43–53, (2001).
2. Åström K. J. and Hägglund T. The future of PID control. Control Engineering Practice, 9, (11), 1163–1175, (2001).
3. Yasunobu S. MiyamotoI S. and Ihara H. A Fuzzy Control for Train Automatic Stop Control," Transactions of the Society of Instrument and Control Engineers, 19(11), 873–880, 1983.
4. S. E. Sheridan and P. Skjoth, "Automatic Kiln Control at Oregon Portland Cement Company's Durkee Plant Utilizing Fuzzy Logic," in IEEE Transactions on Industry ） Applications, IA-20(3), 562-568, 1984.
5. Shi Jingzhuo, Xu Dianguo and Wang Zongpei, "Hybrid stepping motor fuzzy position servo system based on DSP," 2000 26th Annual Conference of the IEEE Industrial Electronics Society. IECON 2000. 2000 IEEE International Conference on Industrial Electronics, Control and Instrumentation. 21st Century Technologies, Nagoya, Japan, 2, 1475-1479 (2000).
6. M. Parnichkun and C. Ngaecharoenkul, "Hybrid of fuzzy and PID in kinematics control of a pneumatic system," 2000 26th Annual Conference of the IEEE Industrial Electronics Society. IECON 2000. 2000 IEEE International Conference on Industrial Electronics, Control and Instrumentation. 21st Century Technologies, Nagoya, Japan, 2, 1485-1490, (2000).
7. B. Qu and J. Qi, "Research on speed control algorithm of stepper motor with fuzzy PID optimization based on genetic algorithm," 2023 IEEE 11th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 682-686, (2023).
8. A. Altay and A. B. Yildiz, "Exact Electrical Equivalent Circuit Model for Analysis of Permanent Magnet Direct Current (PMDC) Motor," 2023 IEEE 2nd Industrial Electronics Society Annual On-Line Conference (ONCON), SC, USA, 1-6, (2023).
9. B. E. Varghese and A. Swarup, "Novel Op-Amp based Hardware Design of Analog PID Circuit to Control the Altitude of Quadcopter," 2022 2nd Asian Conference on Innovation in Technology (ASIANCON), Ravet, India, 1-8, (2022).
10. J. Carvajal, "Fuzzy PID controller: Design, performance evaluation, and stability analysis," Information Sciences, 123(3–4), 249–270, (2000).
11. D. Misir, H. A. Malki, and Guanrong Chen, "Design and analysis of a fuzzy proportional-integral-derivative controller," Fuzzy Sets and Systems, 79(3), 297–314, (1996).
12. L. A. Zadeh, "Fuzzy sets," Information and Control, 8(3), 338–353, (1965).