DOI: https://doi.org/10.62985/j.huit_ojs.vol25.no6.362

# A CONTROL OF MOBILE ROBOT FOLLOWING PREDEFINED PATH USING FUZZY-PID ALGORITHM

**Van Quoc Nguyen, Khanh Hoa Hoang Nguyen, Thien M. Tran***

*Department of Mechatronics, Faculty of Mechanical Engineering,*
*Ho Chi Minh City University of Technology and Engineering (HCM-UTE),*
*Ho Chi Minh City, Vietnam*

*Email: *thientm@hcmute.edu.vn*

Received: 22 December 2024; Revised: 12 January 2025; Accepted: 17 January 2025

## ABSTRACT

In this paper, the aim of this article proposes the combination of PID control and Fuzzy Logic law to control the autonomous mobile robot (AMR) to improve the work efficiency of the robot's movement while still following the path, based on the tracking error. To do this task, the kinematic equation of the AMR is obtained via using mathematical analysis. However, in reality, controlling AMR moves along the desired trajectory often encounters several formulas of uncertainties that affect the performance of the AMR's work efficiency. Hence, the combination of PID control and Fuzzy Logic law is investigated. Thanks to Fuzzy Logic law, the control gains of PID can be updated following the tracking error, which highly improves the performance of AMR. Finally, the comparison of the proposed controller (Fuzzy-PID) and PID control is carried out to demonstrate feasibility and effectiveness.

*Keywords:* Fuzzy-PID control, fuzzy logic, PID control, AMR, mobile robot.

## 1. INTRODUCTION

In the context of the 4.0 revolution, applying digital technology platforms to daily life is an inevitable trend, improving productivity, competitive advantage, optimizing corporate governance, and so forth. The most prominent point of this revolution is that robots replace humans in a number of occupations: Automation Guided Vehicles (AGVs) transport goods in factories, forklifts (used to remove goods in warehouses), Robots serve restaurants, and so on. The tourism service industry is no exception, many hotels around the world have begun to develop plans to bring robots into service - bringing many new and exciting experiences.

A mobile robot is a self-operating machine designed to move and interact with its environment, representing a vital research area within robotics and information engineering [1]. Unlike modern industrial robots, which are stationary and typically equipped with mechanical arms for operation, mobile robots can navigate freely without being restricted to a fixed location [2]. Autonomous Mobile Robots (AMRs), in particular, are capable of traversing unstructured environments without requiring physical or electro-mechanical guidance systems. Alternatively, they can follow predetermined paths in semi-controlled spaces [3, 4]. These capabilities make them highly versatile in both industrial and healthcare settings. Due to their compact design and mobility, mobile robots have gained widespread adoption. In hospitals, AMRs have been utilized for years in developed countries to transport materials or assist patients [5]. In industrial environments, warehouses employ AMRs to streamline operations, moving items from storage areas to order fulfillment zones efficiently [6]. In contrast, traditional industrial robots are typically stationary, featuring multi-jointed arms and end-effectors that are attached to fixed bases. Consequently, mobile robots have become a significant focus in robotics research, with many universities establishing dedicated labs to explore their potential and variations [7].The essential components of a mobile robot include a controller, sensors, actuators, and a power system. The controller often takes the form of a microprocessor, embedded microcontroller, or a personal computer (PC). Sensors, tailored to the robot's specific needs, enable various functionalities such as dead reckoning, proximity detection, collision avoidance, triangulation ranging, and precise positioning, depending on the application [8]. Fuzzy logic offers a versatile approach to addressing uncertainty and partial truths. Unlike traditional binary logic, which confines truth values to either 0 or 1, fuzzy logic operates on a continuum, allowing truth values to take any real number between 0 and 1 [9]. This approach was formalized in 1965 by mathematician Lotfi Zadeh through his proposal of fuzzy set theory [10]. However, its foundations can be traced back to the early 20th century, with contributions from Łukasiewicz and Tarski in infinite-valued logic [11]. At its core, fuzzy logic mirrors how humans make decisions based on incomplete or imprecise information. It uses fuzzy sets as mathematical tools to model vagueness and ambiguity effectively [12]. These models excel in interpreting and processing data that lacks clarity or certainty, making them indispensable in fields ranging from control systems to artificial intelligence [13]. Over time, fuzzy logic has become an integral part of numerous applications, thanks to its ability to manage complexity and provide actionable insights across various domains [14, 15].

Based on the mentioned analysis, the problem of controlling the AMR to track along a trajectory, the robot's velocity will highly affect whether it follows a straight path and a curve. To track the path smoothly as a curve, AMR's response has to be low and constant response entire the moving process, however, this leads to inefficiency at work. To handle this problem, the aim of the article proposes the combination of PID control and Fuzzy Logic law to control the AMR to improve the work efficiency of the robot's moving still following the path. To do this task, the control gains of PID are done by Fuzzy Logic law with input as AMR's tracking error. Thanks to Fuzzy Logic law, the control gains of PID can be updated following the tracking error. Finally, the comparison of the proposed controller (Fuzzy-PID) and PID control is carried out to demonstrate feasibility and effectiveness.

## 2. MATHEMATICAL MODEL

The kinematic and dynamic model for the robot is built based on the model with Mecanum wheels arranged at an angle of 45 degrees relative to the dynamic coordinates, seen in Figure 1. In the frame of reference $\partial_R(t)$ by mobile robots, the velocity of the wheels are noted as follows [16].

$$\begin{bmatrix} V_1(t) \\ V_2(t) \\ V_3(t) \\ V_4(t) \end{bmatrix} = \begin{bmatrix} r\omega_1(t) \\ r\omega_2(t) \\ r\omega_3(t) \\ r\omega_4(t) \end{bmatrix} \tag{1}$$

$$V_1(t) = V_{Gx}(t) - \frac{L}{2}(t) + \left( V_{Gy}(t) - \frac{L}{2}\Omega(t) \right) \frac{1}{tg\gamma} \tag{2}$$

where *r* is defined as the radius of the Mecanum wheel. With $\gamma$ is the middle angle $V_1(t)$ and $V_{L1}(t)$, the velocity of wheel 1 obtained as follows [17]. By using the similar principle of the wheel 1, the velocity of all wheels is denoted as follows

$$\begin{aligned} V_1(t) &= V_{Gx}(t) - \frac{L}{2}(t) + \left( V_{Gy}(t) - \frac{d}{2}\Omega(t) \right) \frac{1}{tg\gamma} \\ V_2(t) &= V_{Gx}(t) + \frac{L}{2}(t) - \left( V_{Gy}(t) - \frac{d}{2}\Omega(t) \right) \frac{1}{tg\gamma} \\ V_3(t) &= V_{Gx}(t) + \frac{L}{2}(t) + \left( V_{Gy}(t) - \frac{d}{2}\Omega(t) \right) \frac{1}{tg\gamma} \\ V_4(t) &= V_{Gx}(t) - \frac{L}{2}(t) - \left( V_{Gy}(t) - \frac{d}{2}\Omega(t) \right) \frac{1}{tg\gamma} \end{aligned} \tag{3}$$

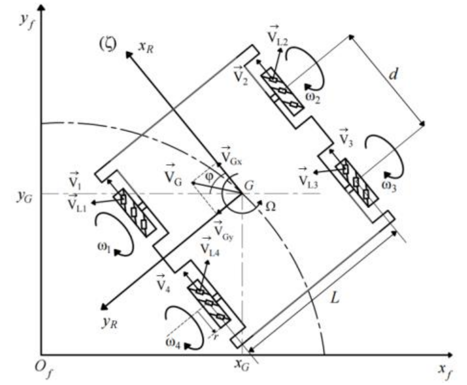

*Figure 1.* Mecanum wheel-type omnidirectional AGV kinematics

If $\gamma = 45^0$ Equation (3) can be rewritten as follows

$$\begin{aligned} V_1(t) &= V_{Gx}(t) + V_{Gy}(t) - \left(\frac{L}{2} + \frac{d}{2}\right)\Omega(t); \\ V_2(t) &= V_{Gx}(t) - V_{Gy}(t) + \left(\frac{L}{2} + \frac{d}{2}\right)\Omega(t); \\ V_3(t) &= V_{Gx}(t) + V_{Gy}(t) + \left(\frac{L}{2} + \frac{d}{2}\right)\Omega(t); \\ V_4(t) &= V_{Gx}(t) - V_{Gy}(t) - \left(\frac{L}{2} + \frac{d}{2}\right)\Omega(t). \end{aligned} \tag{4}$$

Substituting Equation (4) into Equation (1), we can obtain as follows

$$\begin{bmatrix} \omega_1(t) \\ \omega_2(t) \\ \omega_3(t) \\ \omega_4(t) \end{bmatrix} = \begin{bmatrix} \frac{1}{r} & \frac{1}{r} & \frac{-1}{2r}(L+d) \\ \frac{1}{r} & \frac{-1}{r} & \frac{1}{2r}(L+d) \\ \frac{1}{r} & \frac{1}{r} & \frac{1}{2r}(L+d) \\ \frac{1}{r} & \frac{-1}{r} & \frac{-1}{2r}(L+d) \end{bmatrix} \begin{bmatrix} V_{Gx}(t) \\ V_{Gy}(t) \\ \Omega(t) \end{bmatrix} \tag{5}$$

The position and direction of the Mecanum wheel omnidirectional Mobile Robot is obtained as follows

$$\begin{aligned} x_f(t) &= \frac{r}{4}\int_0^t \left(\omega_1(t) + \omega_2(t) + \omega_3(t) + \omega_4(t)\right)\cos\gamma(t)\,dt \\ &\quad + \frac{r}{4}\int_0^t \left(-\omega_1(t) + \omega_2(t) - \omega_3(t) + \omega_4(t)\right)\sin\gamma(t)\,dt \\ y_f(t) &= \frac{r}{4}\int_0^t \left(\omega_1(t) + \omega_2(t) + \omega_3(t) + \omega_4(t)\right)\cos\gamma(t)\,dt \\ &\quad + \frac{r}{4}\int_0^t \left(\omega_1(t) - \omega_2(t) + \omega_3(t) - \omega_4(t)\right)\sin\gamma(t)\,dt \\ \gamma(t) &= \frac{r}{2(L+d)}\int_0^t \left(-\omega_1(t) + \omega_2(t) - \omega_3(t) + \omega_4(t)\right)dt \end{aligned} \tag{6}$$

Substituting Equation (6) into Equation (5), we can obtain as follows

$$\begin{aligned}
\omega_1(t) &= \frac{1}{r}(\cos\gamma - \sin\gamma)x_f(t) + (\cos\gamma + \sin\gamma)y_f(t) - \frac{L+d}{2}\gamma(t) \\
\omega_2(t) &= \frac{1}{r}(\cos\gamma + \sin\gamma)x_f(t) - (\cos\gamma - \sin\gamma)y_f(t) + \frac{L+d}{2}\gamma(t) \\
\omega_3(t) &= \frac{1}{r}(\cos\gamma - \sin\gamma)x_f(t) + (\cos\gamma + \sin\gamma)y_f(t) + \frac{L+d}{2}\gamma(t) \\
\omega_4(t) &= \frac{1}{r}(\cos\gamma + \sin\gamma)x_f(t) - (\cos\gamma - \sin\gamma)y_f(t) - \frac{L+d}{2}\gamma(t)
\end{aligned} \tag{7}$$

The dimensions of the robot, measuring length x width (e.g., 40cm x 30cm), were considered during the development of the kinematic and dynamic models, ensuring accurate representation of its real-world behavior.

## 3. CONTROL SYSTEM DESIGN

The trajectory controller gets all data of the desired trajectory and calculates all desired values, the linear velocity $V_d(t)$ and the angular velocity $\omega_d(t)$. These values are sent to the kinematic block and this one has a duty to use the values of $V_d(t)$ and $\omega_d(t)$ to generate motor control commands rotated with the actual speed (*v*) and the angular velocity (*w*). The proposed controller (Fuzzy-PID) is a kind of nonlinear control that has the advantages of both Fuzzy and PID. In this article, control gains of PID are calculated by Fuzzy Logic law. Essentially, the rule of fuzzy plays an important role in calculating the coefficients as $K_p$, $K_i$ and $K_d$. The membership function (Mamdani) conducts directly adjusting the control parameters of the main control PID. The input of the Mamdani system undergoes fuzzification by mapping real input values (e and de) into fuzzy sets using membership functions such as triangular or trapezoidal functions. This process converts information from numerical values into fuzzy language, making it suitable for fuzzy inference. The output of the Mamdani system consists of fuzzy sets that are aggregated (combined) through fuzzy rules using the max method (taking the highest value from the rules). The results are then de-fuzzifier to convert fuzzy sets back into crisp (numerical) values using the Centroid method, ensuring smooth and precise output values. The Fuzzy Logic law with inputs is the error of tracking position and rotation angle, and time derivative of tracking error, seen in Figure 2.

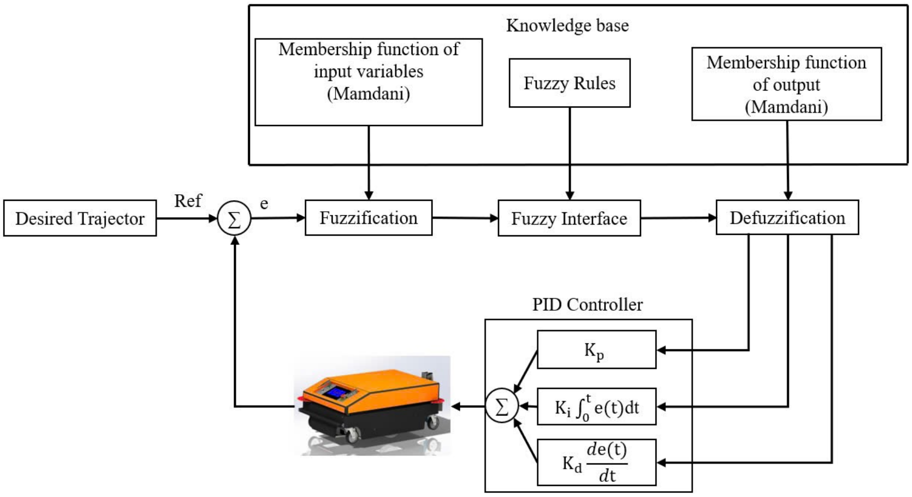

*Figure 2*. The structure of the controller for a mobile robot with four Mecanum wheels to follow the trajectory.

The advantage of the algorithm can update the control gains of PID based on the error of AMR instead of fixing them.

$$\begin{aligned} K_p &= \left(K_{p\max} - K_{p\min}\right)K_p^* + K_{p\min} \\ K_i &= \left(K_{i\max} - K_{i\min}\right)K_i^* + K_{i\min} \\ K_d &= \left(K_{d\max} - K_{d\min}\right)K_d^* + K_{d\min} \end{aligned} \tag{8}$$

where $\left[K_{p\min}, K_{p\max}\right]$, $\left[K_{i\min}, K_{i\max}\right]$, and $\left[K_{d\min}, K_{d\max}\right]$ are inferred by experiences and trial methods. The proposed controller Fuzzy-PID with inputs is tracking error $e(t)$ and time derivative of error with 5 values, listed in Table 1 as Negative Big(NB), Negative Small(NS), Zero(Z), Positive Small(PS), and Positive Big(PB). Based on the rule in Table 1, the Fuzzy sets corresponding to input and output are revealed in Figure 3.

*Table 1.* The rules of the controller Fuzzy-PID

| **e/ce** | **NB** | **NS** | **Z** | **PS** | **PB** |
|---|---|---|---|---|---|
| **NB** | VB/S/S | VB/M/S | S/M/VB | S/M/B | M/S/S |
| **NS** | NS/B/S | B/B/S | S/B/B | S/B/B | B/B/S |
| **Z** | B/B/S | M/B/M | S/VB/B | S/VB/M | VB/VB/S |
| **PS** | M/B/M | S/B/B | S/B/B | M/B/S | VB/B/S |
| **PB** | S/S/M | S/S/VB | S/M/VB | B/M/S | VB/S/S |

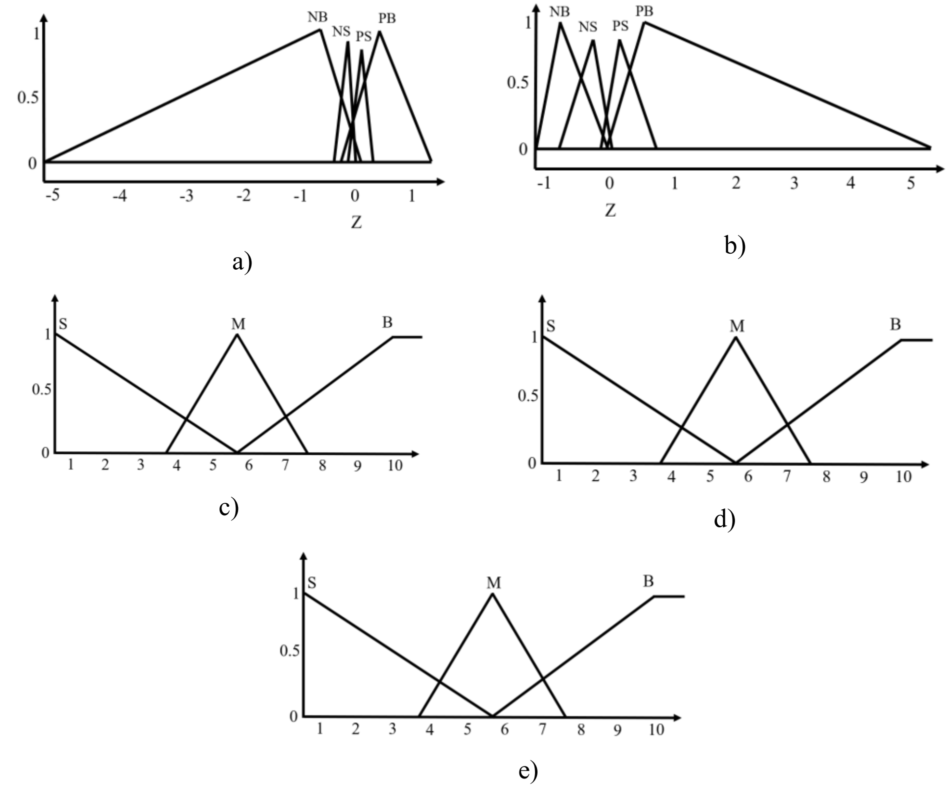

*Figure 4.* Law of Fuzzy (Mamdani): a) Fuzzy sets corresponding to input variables; b) Fuzzy sets corresponding to input variables; c) Fuzzy sets corresponding to output variables; d) Fuzzy sets corresponding to output variables; e) Fuzzy sets corresponding to output variables

## 4. SIMULATION RESULTS

To demonstrate the feasibility and effectiveness of the proposed controller Fuzzy-PID, the comparison of the proposed controller and classical PID control is carried out. The simulation is conducted in two scenarios. All initial parameters of both scenarios are defined as zero. The simulation is implemented using MATLAB R2020a. The sampling time for the simulation is set to $L = 0.01(s)$ to ensure accurate and responsive control behavior in both scenarios.

### 4.1. Numerical results in scenario 1

The control parameters for the Fuzzy-PID in this scenario are initialized with the following ranges: $K_p = [1,10]$, $K_i = [1,10]$, $K_d = [1,10]$. For the classical PID controller, the initial parameters are set as: $K_p = 9.77$, $K_i = 19.22$, $K_d = -0.36$. In scenario 1, the circle trajectory is conducted to show the performance of the proposed controller Fuzzy-PID at a curve. Figure 5 shows the Fuzzy-PID controller appears to track the desired path more accurately than the PID controller, which deviates significantly at certain points. Setting time of PID and Fuzzy-PID is 4.8 seconds and 2.5 seconds respectively. In addition, the overshoot of PID and Fuzzy-PID is 25% and 8% respectively, steady-state error of PID and Fuzzy-PID is 0.02 meters and 0.15 meters correspondingly. These results demonstrate that the Fuzzy-PID controller exhibits better performance in terms of faster settling time, lower overshoot, and reduced steady-state error compared to the classical PID controller. Figure 6 shows a sinusoidal pattern with a range of approximately ±6 meters. It achieves a maximum value of 4.5 meters and a minimum value of -5.5 meters, reflecting smooth and consistent motion along the X-axis over time. Figure 7 also follows a sinusoidal pattern within the range of approximately ±6 meters. It shows a maximum value of 4.5 meters and a minimum value of -5.5 meters, indicating well-coordinated movement along the Y-axis, synchronized with the X-axis trajectory.

In Figure 8, the traditional PID controller shows a significant initial overshoot in the x-axis, indicating overcompensation and a struggle to balance response speed with stability. After the initial spike, the PID error oscillates before eventually stabilizing. In contrast, the Fuzzy-PID controller has a smoother response, quickly reducing the x-axis error with minimal overshoot and settling to near-zero error. This demonstrates the Fuzzy-PID controller's superior adaptability, maintaining stability and accuracy without excessive corrections. Figure 9 highlights the tracking error performance of the Fuzzy-PID and classical PID controllers in the y-axis. The PID controller exhibits a significant initial overshoot of approximately 1.8 m, followed by oscillations that stabilize after around 15 seconds, demonstrating slow damping and prolonged tracking error. In contrast, the Fuzzy-PID controller achieves minimal overshoot of only 0.5 m and stabilizes within 5 seconds, showcasing superior damping and faster error minimization. These results confirm the Fuzzy-PID's effectiveness in reducing tracking error and ensuring smoother control. Figure 10 presents the traditional PID controller exhibiting a large initial spike in rotational angle error, indicating an excessive, delayed adjustment in orientation due to difficulty with dynamic changes. While this error gradually decreases, it converges slowly. Figure 11 shows a smooth response with a maximum overshoot of approximately 80 m/s for the PID controller and 60 m/s for the Fuzzy-PID. The system achieves steady-state conditions within 10 seconds, with negligible steady-state error for both controllers. Figure 12 has a peak overshoot of approximately 50 m/s for the PID controller and 40 m/s for the Fuzzy-PID. The settling time is around 10 seconds, and the system exhibits minimal steady-state error. Figure 13 shows a maximum overshoot of approximately 30 m/s for the PID controller and 20 m/s for the Fuzzy-PID. The system stabilizes within 10 seconds, with steady-state conditions reached effectively and no significant steady-state error. Figure 14 demonstrates the highest overshoot, reaching 100 m/s for the PID controller and 80 m/s for the Fuzzy-PID. Steady-state is achieved within 10 seconds, with both controllers having negligible steady-state error.

In contrast, the Fuzzy-PID controller quickly reduces rotational angle error with minimal overshoot and oscillation, showing better adaptability and stability. Fuzzy logic enables real-time adjustments, minimizing corrective actions and enhancing control efficiency.

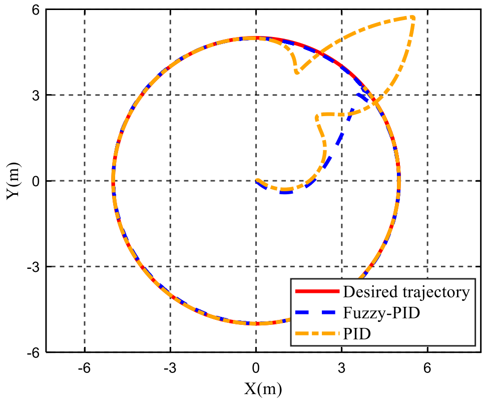

*Figure 5.* The performance of proposed Fuzzy-PID controller and the classical PID.

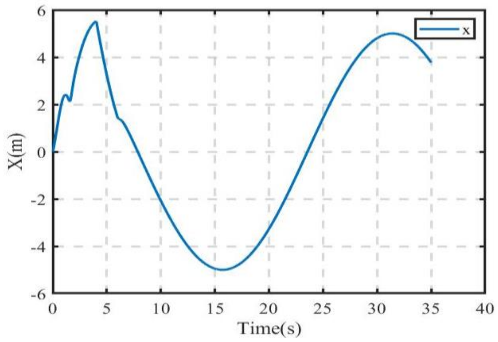

*Figure 6.* The trajectory of X-axis.

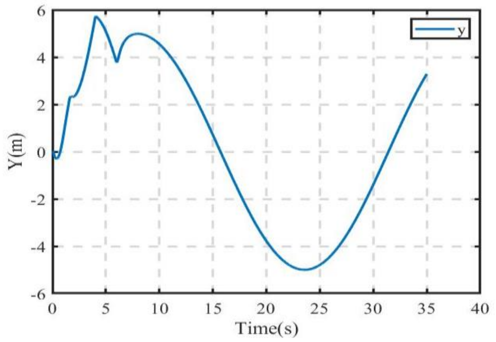

*Figure 7.* The trajectory of Y-axis.

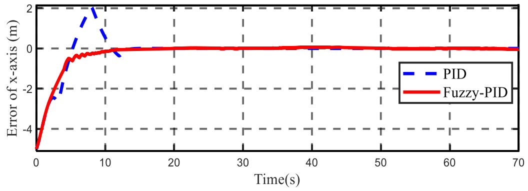

*Figure 8.* The tracking error of proposed Fuzzy-PID controller and the classical PID in X-axis.

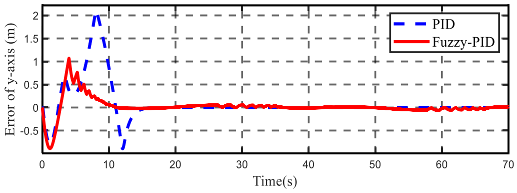

*Figure 9.* The tracking error of proposed Fuzzy-PID controller and the classical PID in Y-axis.

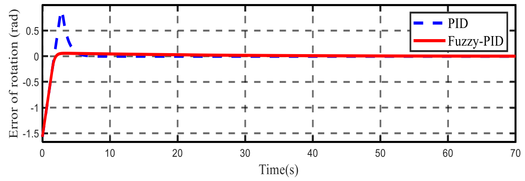

*Figure 10.* The tracking error of proposed Fuzzy-PID controller and the classical PID in rotation.

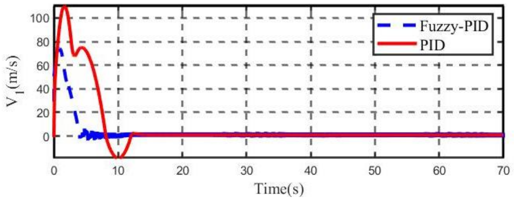

*Figure 11.* Velocity of wheel 1.

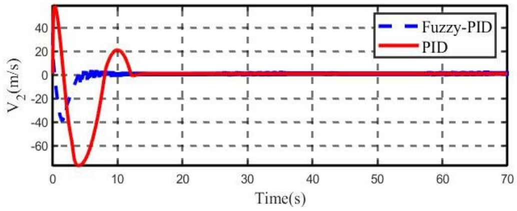

*Figure 12.* Velocity of wheel 2.

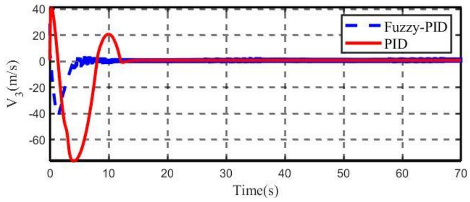

*Figure 13.* Velocity of wheel 3.

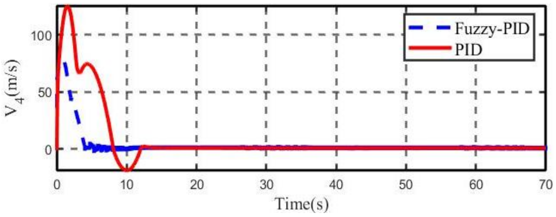

*Figure 14.* Velocity of wheel 4.

### 4.2. Numerical results in scenario 2

The control parameters for the Fuzzy-PID in this scenario are initialized with the following ranges: $K_p = [1,10]$, $K_i = [1,10]$, $K_d = [1,10]$. For the classical PID controller, the initial parameters are set as: $K_p = 9.77$, $K_i = 19.22$, $K_d = -0.36$. The complex trajectory is conducted and shown in Figure 15. The Fuzzy-PID controller demonstrates superior tracking accuracy, with minimal deviation from the desired path, while the PID controller shows larger deviations, particularly at curve intersections. Key metrics indicate that the Fuzzy-PID achieves a lower steady-state error (0.01 m) and faster settling time (3.0 seconds) compared to the PID (0.05 m and 4.2 seconds, respectively). Overall, the Fuzzy-PID provides smoother and more consistent control, making it better suited for complex trajectories. Figure 16 shows a smooth sinusoidal motion within the range of approximately ±6 meters. The robot exhibits consistent oscillations with peaks at around 4.5 meters and troughs near -5.5 meters, indicating accurate path tracking along the X-axis. Figure 17 also demonstrates sinusoidal behavior, with a range of approximately ±6 meters. The motion along the Y-axis is synchronized with the X-axis, maintaining consistent oscillation patterns and effective tracking with minimal deviation.

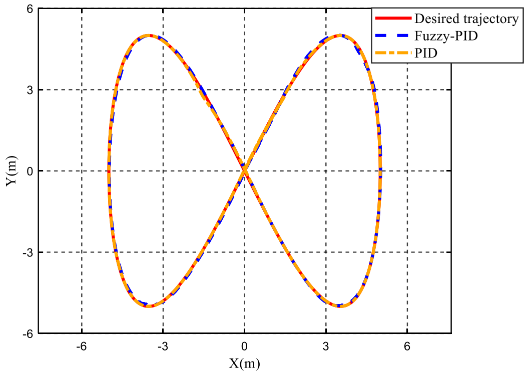

*Figure 15.* The complicated performance of proposed Fuzzy-PID controller and the classical PID.

Figure 18 describes an initial error of PID controller spiking due to overcorrection and taking longer to stabilize, with early oscillations indicating difficulty balancing speed and stability. In contrast, the Fuzzy-PID controller quickly reduces the error with a smooth response, stabilizing around zero without oscillations. The Fuzzy Logic component dynamically can update control parameters, maintaining minimal error despite changing conditions. Figure 19 illustrates that the PID controller maintains a stable but slightly offset y-axis error, indicating it is consistently off-target due to its fixed parameters, which limit quick adjustments. In contrast, the Fuzzy-PID controller actively adapts to minimize error, resulting in temporary fluctuations but consistently bringing the error back to near zero. This dynamic adaptability allows the Fuzzy-PID controller to correct deviations more effectively, reducing prolonged tracking errors. Figure 20 presents that the PID controller has larger, sustained oscillations in rotational error early on, indicating difficulty in handling rotational dynamics due to its fixed-gain parameters. Figure 21 shows an overshoot of approximately 80 m/s for PID and 60 m/s for Fuzzy-PID. The system settles within 10 seconds, with Fuzzy-PID providing a smoother response and reduced steady-state error compared to PID. Figure 22 exhibits an overshoot of around 40 m/s for PID and 30 m/s for Fuzzy-PID. The settling time is approximately 8–10 seconds, with Fuzzy-PID achieving a more stable steady-state. Figure 23 demonstrates an overshoot of 20 m/s for PID and 15 m/s for Fuzzy-PID. The settling time is within 8 seconds, and Fuzzy-PID provides less oscillation and no steady-state error. Figure 24 shows a significant overshoot of 100 m/s for PID and 80 m/s for Fuzzy-PID. The system stabilizes within 10 seconds, with Fuzzy-PID effectively minimizing oscillations and steady-state error. This leads to initial overreactions and delayed damping. In contrast, the Fuzzy-PID controller provides a stable response, quickly minimizing rotational error with minimal overshoot. Its adaptive nature allows for smoother corrections and faster stabilization, making it well-suited for managing unpredictable rotational changes in dynamic systems.

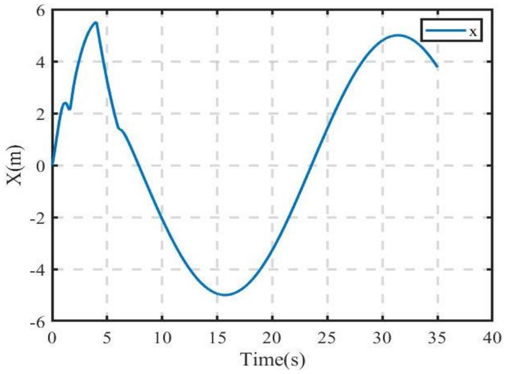

*Figure 16.* The trajectory of X-axis.

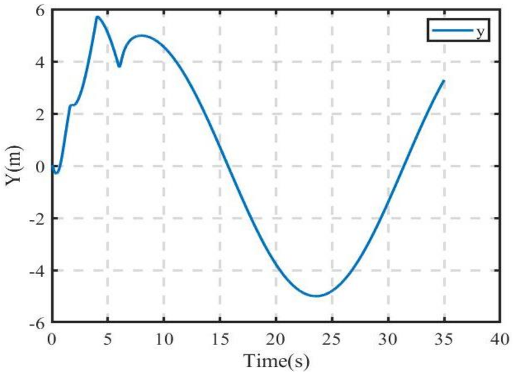

*Figure 17.* The trajectory of Y-axis.

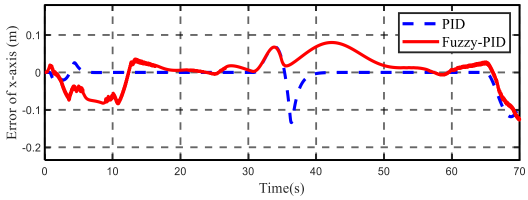

*Figure 18.* The complicated tracking error of proposed Fuzzy-PID controller and the classical PID in X-axis.

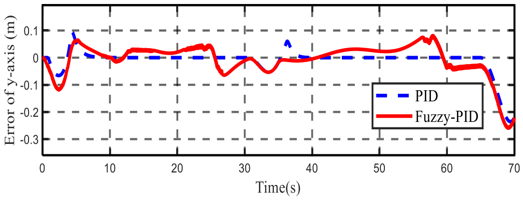

*Figure 19.* The complicated tracking error of proposed Fuzzy-PID controller and the classical PID in Y-axis.

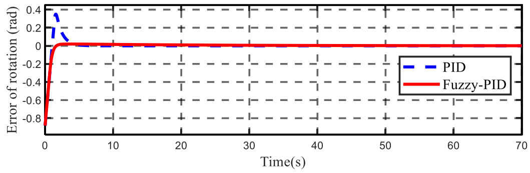

*Figure 20.* The complicated tracking error of proposed Fuzzy-PID controller and the classical PID in rotation.

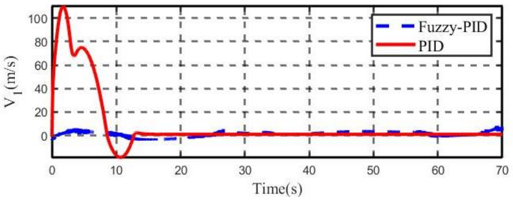

*Figure 21.* Velocity of wheel 1.

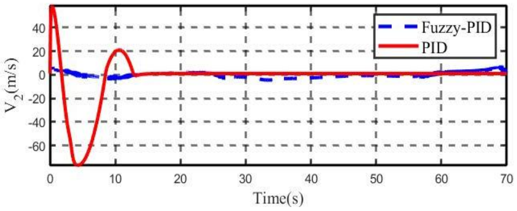

*Figure 22.* Velocity of wheel 2.

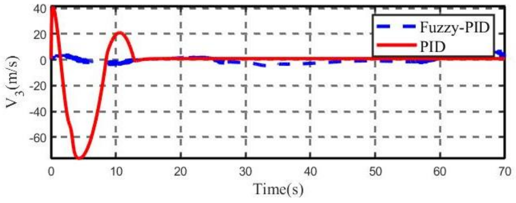

*Figure 23.* Velocity of wheel 3.

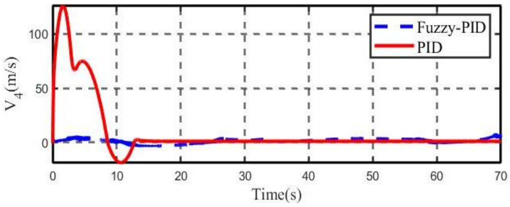

*Figure 24.* Velocity of wheel 4.

Overall, the Fuzzy-PID controller outperforms the traditional PID controller by providing better accuracy, stability, and adaptability. It ensures smoother responses, quickly stabilizes with minimal error, and maintains closer alignment with the desired paths, making it the preferred choice for controlling both circular and complex trajectories effectively.

## 5. CONCLUSION

The article presents the dynamic modeling of a mobile robot equipped with four mecanum wheels and introduces a Fuzzy-PID controller, designed and implemented to enhance the control capabilities of the robot. Through comparative analysis and various experimental trials, the results clearly indicate that the proposed Fuzzy-PID controller exhibits significantly superior performance compared to the traditional PID controller. This is particularly evident in the robot's trajectory and error handling across X, Y, and angular orientations.

The Fuzzy-PID controller demonstrates a remarkable ability to maintain the desired trajectory with minimized error in both linear and rotational movements. In contrast to the standard PID controller, which shows noticeable deviations in rapid directional changes and is slower in settling to steady states, the Fuzzy-PID controller adapts more effectively to dynamic adjustments, offering faster response times and better stability. This adaptive quality allows the Fuzzy-PID to precisely control the robot's movements even under varied command inputs, achieving high accuracy and efficiency.

**Acknowledgment:**

The authors are indebted to editor, anonymous reviewers, and HCM-UTE for encouraging comments, suggestions, and conditions which have enhanced the quality of the research paper.

## REFERENCES

1. Junyan H., Parijat B., and Alexander L. - Group coordinated control of networked mobile robots with applications to object transportation, IEEE Transactions on Control Systems Technology **30** (2022) 1071-1080. https://doi.org/10.1109/TVT.2021.3093157
2. Bozinovski S. - Signal processing robotics using signals generated by a human head: From pioneering works to EEG-based emulation of digital circuits, in Proceedings of International Conference on Robotics in Alpe-Adria-Danube Region (2016). https://doi.org/10.1007/978-3-319-49058-8_49
3. Hu J., Bhowmick P., Jang I., Arvin F., and Lanzon A. - A decentralized cluster formation containment framework for multirobot systems, IEEE Transactions on Robotics **37** (6) (2021) 1936-1955. https://doi.org/10.1109/TRO.2021.3071615
4. Hu J., Turgut A., Lennox B., and Arvin F. - Robust formation coordination of robot swarms with nonlinear dynamics and unknown disturbances: design and experiments, IEEE Transactions on Circuits and Systems II: Express Briefs **69** (1) (2021) 114-118. https://doi.org/10.1109/TCSII.2021.3074705
5. Moubarak P. and Ben-Tzvi P. - Adaptive manipulation of a hybrid mechanism mobile robot, in Proceedings of IEEE International Symposium on Robotic and Sensors Environments (ROSE), Canada (2011). https://doi.org/10.1109/ROSE.2011.6058520
6. Gopalakrishnan B., Tirunellayi S. and Todkar R. - Design and development of an autonomous mobile smart vehicle: a mechatronics application, Mechatronics **14** (5) (2004) 491-514. https://doi.org/10.1016/j.mechatronics.2003.10.003
7. Novák V., Perfilieva I. and Močkoř J. - Mathematical principles of fuzzy logic, Springer (1999). https://doi.org/10.1007/978-1-4615-5217-8
8. Cintula P., Christian G. F. and Carles N. - Fuzzy logic, Committee on Publication Ethics (2011).
9. Zadeh L. A. - Fuzzy sets, Information & Control **8** (3) (1965) 338-353. https://doi.org/10.1016/S0019-9958(65)90241-X

10. Saatchi R. - Fuzzy logic concepts, developments and implementation, Information **15** (10) (2024) 1-24. https://doi.org/10.3390/info15100656

11. James K.P. - Introduction to fuzzy logic, Wiley (2021). https://doi.org/10.1002/9781119772644

12. Tolossa T. D., Gunasekaran M., Halder K., Verma H. K., Parswal S. S., Jorwal N., and Joseph F. O. M. - Trajectory tracking control of a mobile robot using fuzzy logic controller with optimal parameters, Robotica **42** (8) (2024) 2801-2824. https://doi.org/10.1017/S0263574724001140

13. Do K. T. - A design of a fuzzy PID controller for path tracking of a mobile robot with differential drive, Journal of Science And Technology **2021** (7) 127-132.

14. Xu Q., Kan J., Chen S., and Yan S. - Fuzzy PID based trajectory tracking control of mobile robot and its simulation in Simulink, International Journal of Control and Automation **7** (8) (2014) 233-244. https://doi.org/10.14257/ijca.2014.7.8.20

15. Mohamed M. J. and Abbas M. Y. - Design a fuzzy PID controller for trajectory tracking of mobile robot, Engineering and Technology Journal **36** (1) (2018), 100-110. https://dx.doi.org/10.30684/etj.36.1A.15

16. Dang X. K., Nguyen N. T. T., and Tran T. D. - Fuzzy logic controllers to control the robotino robot model moving in a given fixed path. Journal of Transportation Science and Technology **34** (11) (2019) 45-50.

17. Mac T. T., Vu V. T. and Tran D. T. - Path following of mobile robot using fuzzy controller, Vietnam Journal of Sciences and Technology **53** (2) (2015) 244-253. https://doi.org/10.15625/0866-708X/53/2/4149

## TÓM TẮT

## ĐIỀU KHIỂN MOBILE ROBOT THEO ĐƯỜNG DẪN ĐƯỢC XÁC ĐỊNH TRƯỚC SỬ DỤNG THUẬT TOÁN FUZZY-PID

Nguyễn Văn Quốc, Hoàng Nguyễn Khánh Hòa, Trần Minh Thiên*

*Bộ môn Cơ điện tử, Khoa Cơ khí Chế tạo máy,*

*Trường Đại học Công nghệ Kỹ Thuật Thành phố Hồ Chí Minh,*

*TP. Hồ Chí Minh, Việt Nam*

*Email: *thientm@hcmute.edu.vn*

Trong bài báo này, mục tiêu của bài báo đề xuất kết hợp điều khiển PID và luật Fuzzy Logic để điều khiển robot di động tự hành (AMR) nhằm nâng cao hiệu quả công việc của chuyển động của robot trong khi vẫn đi theo đường dẫn đã được quy định, dựa trên sai lệch vị trí. Để thực hiện nhiệm vụ này, phương trình động học của AMR được tính toán dựa trên việc sử dụng phân tích toán học. Tuy nhiên, trong thực tế, việc điều khiển AMR di chuyển theo quỹ đạo mong muốn thường gặp phải một số trường hợp không xác định trước ảnh hưởng đến hiệu suất công việc của AMR. Do đó, sự kết hợp của điều khiển PID và luật Fuzzy Logic được nghiên cứu trong bài báo này. Nhờ luật Fuzzy Logic, các hệ số điều khiển của PID có thể được cập nhật theo sai lệch thực tế, giúp cải thiện đáng kể hiệu suất của AMR. Cuối cùng, việc so sánh bộ điều khiển được đề xuất (Fuzzy-PID) và điều khiển PID cổ điển được thực hiện để chứng minh tính khả thi và hiệu quả.

*Từ khóa:* Bộ điều khiển Fuzzy-PID, logic mờ, bộ điều khiển PID, AMR, mobile robot.