International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Journal Européen des Systèmes Automatisés
Vol. 59, No. 2, February, 2026, pp. 477-486
Journal homepage: http://iieta.org/journals/jesa

Proposed Fuzzy Logic Control System to Track the Trajectory of the Robotic System

Ibrahim A. Hassan[1], Issa A. Abed[2*], Walid A. Al-Hussaibi[2], Kadhim Sabah Rahimah[3], Hassanin F. Abdul-Hassan[2]

[1] Ministry of Education, General Directorate of Basra Education, Basra 61001, Iraq
[2] Basra Engineering Technical College, Southern Technical University, Basra 61001, Iraq
[3] Ministry of Oil-South Refineries Company, Basra 61001, Iraq

Corresponding Author Email: issaahmedabed@stu.edu.iq

https://doi.org/10.18280/jesa.590216

Received: 1 November 2025
Revised: 20 January 2026
Accepted: 31 January 2026
Available online: 28 February 2026

Keywords:
mobile robot, trajectory tracking, fuzzy logic controller, membership functions

ABSTRACT

This paper describes the design and modeling of a fuzzy logic controller (FLC) for trajectory tracking in a differential-drive robot. The robot is kinematically characterized with a control goal of maintaining a preset reference path while minimizing tracking error. The suggested controller accepts distance and orientation errors as inputs and generates linear and angular velocity commands. For this purpose, the membership functions (MFs) and rule base are manually designed using heuristic knowledge. MATLAB/Simulink is utilized for performing simulation tests on two different MFs using the step path as a reference trajectory. Triangle and Gaussian membership functions are used to compare the performance of FLC. Simulation findings demonstrate that the Triangle MF performs far better than the Gaussian MF approach. For instance, the findings showed that Triangle MF is more effective and can minimize mean position error by 78.8387%, rising time by 8.3333%, peak time by 8.00%, overshoot by 50.6758%, and steady-state error by 100.00%. Besides, the outcomes show that the FLC allows the robot to follow the course smoothly while maintaining adequate tracking performance and error convergence. The implemented system is resistant to initial condition fluctuations, providing a simple and understandable alternative to traditional controllers.

1. INTRODUCTION

Robotics is an essential field of study that combines knowledge from various disciplines, including mechanics, electronics, and computer engineering, that acts a mobile robot to function autonomously in a given environment [1]. Mobile robots are robotic systems that can move around a specified location and execute a variety of activities independently. The robots can be designed with wheels or legs [2]. For many years, researchers have been interested in the concept of mobile robots. The reason behind this is that people had many exciting ideas about what intelligent agents could achieve, and computers on wheels with basic sensors had just the start of these ideas. People researched to find features, spot trends and regularities, gain knowledge from past experiences, locate themselves, create maps of their environment, find their way to a destination, and complete particular tasks [3]. Recently, many real-world applications depend on autonomous mobile robots. Numerous industries, such as the military, technology, agriculture, transportation, automotive, healthcare, science, and hazardous settings, use robots [4]. Due to their smaller size and comparatively lower investment costs compared to flying and humanoid robots, mobile robots are widely used. In addition, humans can understand the movement patterns of mobile robots. This facilitates the development of mobile robots with wheels for a variety of applications [5].

The controller's performance is critical to a mobile robot's operation; therefore, it is a vital component in the design and development process. The primary challenge with a mobile robot is its stability and tracking control. The robot navigates the intended path using geometric parameters and trajectory tracking control. Since the mobile robot must travel to a specific destination within a certain timeframe, tracking control is crucial for motion control [6-8]. Trajectory tracking is a critical component for mobile robots. The fundamental purpose of mobile robots is to navigate a predetermined route and decide where to travel. This data comes from a leader robot or a reference path. Trajectory tracking is the process of determining the robot's speed and steering parameters for every moment and ensuring it follows a specific trajectory. A trajectory is a set of points that represent the locational coordinates of a given path [8]. In recent years, various mobile robot controllers have been developed. Among them, the PID controller is still one of the most popular due to its ease of use and efficacy in delivering both stabilization and basic tracking control. However, PID controllers frequently fall short of accuracy when working with complex, nonlinear, or dynamic situations, limiting their performance in advanced robotic applications [6]. PID controllers, for instance, may have difficulty handling system nonlinearities or staying stable during large shocks. A key challenge for PID controllers is their capacity to swiftly minimize the disparity between the setpoint and the process outcome [9].

The PID controller is commonly utilised for trajectory tracking due to its uncomplicated nature and ability to provide consistent performance over time. However, when used for mobile robot applications, there are significant limitations in how it manages to accommodate non-linear dynamics and time-varying uncertainty. One of these limitations stems from the fact that PID controllers utilize fixed gain values; therefore, they cannot adjust to dynamic disturbances sufficiently, which results in reduced tracking accuracy. In order to address these limitations, this research suggests that an FLC is a viable alternative because of its ability to provide an adaptive inference mechanism that is better suited for controlling complex non-linear robotic motion.

An FLC is introduced to solve this problem. Prof. Lofti Zadeh developed fuzzy logic in 1965, which uses fuzzy sets and membership functions to make decisions under uncertainty. Fuzzification, rule base, inference, and defuzzification make up a fuzzy logic controller (FLC). It can outperform conventional PID controllers in applications like DC motor speed control because it solves nonlinearities and uncertainties well while providing robust control [9]. The principle of fuzzy logic is comparable to human perception and thinking. FLC is a range-to-point or range-to-range control, as opposed to the point-to-point control of the classical control method. Fuzzifications of inputs and outputs employing the corresponding membership functions yield the fuzzy controller's output. With respect to its value, a crisp input will be utilized by the various members of the related membership functions. According to this perspective, an FLC's memberships, which can be thought of as a range of inputs, determine its output. The FLC is presented as a solution to this issue [10-12]. The linguistically based structure of FLC and its strong performance for nonlinear systems have made it a prominent technique that has gained more attention in recent decades. However, FLC, including features such as membership functions and language control rules and restrictions, must be adjusted for a particular system. The tuning procedure becomes increasingly challenging and time-consuming as the number of system inputs and outputs increases, which is a significant disadvantage of FLC [13].

Although there are industrial trajectory tracking applications on the PID controller, this research will try to focus on the fundamental behavior of the fuzzy membership function. There are controlled environments used to evaluate the sensitivity of the Gaussian membership function versus the triangular membership function by using the same rule base. This will allow determining the best fuzzy configuration of the Best Fuzzy Configuration of the Robot Platform being studied.

In this work, an FLC is used to control trajectory tracking for two-wheel mobile robots using two different membership functions. FLC's performance is compared using triangle and Gaussian MFs. The controller's overshoot, rise time, settling time, and peak time will be compared. The work includes software (controller design in MATLAB Simulink) for a kinematic model for two-wheel mobile robots.

The paper is organized as follows: part II covers related work on trajectory tracking using FLC. Section III describes the mathematical model for a kinematic two-wheel mobile robot. Section IV includes FLC-based trajectory tracking control. Section V displays the simulation results for FLC. Lastly, Section VI provides the conclusion and future work.

2. RELATED WORKS

Trajectory tracking has been the subject of numerous papers. One essential task for mobile robots that has recently garnered significant attention from publications is trajectory tracking. The development of numerous intelligent and conventional control systems has made controlling mobile robots an essential field of engineering research. Trajectory tracking control aims to minimize tracking errors between the reference trajectory and the mobile robot's actual path by establishing control rules for its velocity. Research on the FLC has attracted a lot of attention for many years. This technique is a well-known method for managing mobile robot navigation.

For obstacle avoidance in mobile robots operating in uncertain environments, particularly in warehouse settings, fuzzy logic navigation techniques have been employed in this study [14]. In this study [15], the development and deployment of a fuzzy logic-based path tracking controller for facilitating mobile robot navigation in indoor spaces are described. The suggested controller includes two steps. The first stage's controller, known as the "Orientation Error Controller," positions the robot perpendicular to the planned path, while the second stage's controller, known as the "Distance Error Controller," minimizes the robot's distance from the intended path. To maintain the desired trajectory, the two controls are actuated successively. The robot and its controller are tested indoors, with a variety of intended paths as input.

Fuzzy logic has been employed in designing and building a trajectory tracking controller that enables a mobile robot to navigate within designated regions. For navigation and obstacle avoidance, a single fuzzy controller has been used. The mobile robot features two optical encoders for accurate position and speed measurement, a DC motor, and nine infrared range (IR) sensors to assess obstacle distances [1]. In this study [16], an FLC for trajectory tracking has been proposed. The FLC allows for creating a dynamic model of the system by handling uncertainties in system dynamics. The control method relies on a feed-forward velocity profile and a corrective velocity signal produced by the FLC in response to position errors.

Due to the difficulty of building a precise mathematical model for the mobile robot employed in this study [17], the "Robotino® from Festo," fuzzy logic was chosen as the method of choice for designing a controller that would ensure safe navigation for the Robotino®. The FLC, which draws inspiration from human experience in these types of applications, requires information on the characteristics and behavior of the Robotino to construct its rule base. An effective controller can be created by simply programming these rules. The Sugeno algorithm is used, and the outcomes of the experiments verify that it works. One FLC with 27 fuzzy rules is utilized for obstacle avoidance, while another FLC with 153 fuzzy rules is used to handle Robotino®'s course tracking. The tool used to implement the two suggested fuzzy controllers is MATLAB.

This study [18] has presented a Fuzzy Logic control based on Z-numbers for tracking the trajectory of mobile robots with differential wheels. This method's primary innovation is its capacity to encode reliability and constraints within multi-input, multi-output rules, in such a way the Input domain Focuses solely on the distance and orientation gap measurements that are currently available. Using the graded mean integration approach and interpolative reasoning, the consequent universe is identified. Since encoding error gradients is complicated, the approach not only makes the procedure simpler but also offers a versatile technique to create control rules that can deal with noisy actuator inputs and missing observations.

In an effort to discover which approach makes navigation easier and allows the robot to overcome real-world unknowns and track an optimal course in a very short amount of time, fuzzy logic (FL) and interval type-2 FL (IT-2FL) controllers have been deployed to a mobile robot in this study [19].

The robot in question is a mobile, non-holonomic unicycle robot that is evolving in two distinct contexts and is represented by a kinematic model. One action module has been introduced to move the robot from a beginning to a goal location in the first environment, which is barrier-free. The same problem followed in a more realistic setting, with items impeding the robot's progress. In such a case, an additional controller known as obstacle avoidance is required. This system enables the robot to navigate independently to a predefined target while avoiding obstacles. The resilience of the defined controllers' structures is verified using MATLAB simulations.

Several trajectory tracking control strategies for two-wheel mobile robots have been examined in this study [20]. To track time-varying desired trajectories, a discrete control strategy is first developed. The effectiveness of this technique is then compared to that of fuzzy and discrete PID controllers. A constructed discrete-time linear model for the dynamics of a two-wheel mobile robot has been used to solve a Riccati problem to design the discrete controller. To examine the efficacy of the suggested controllers, several numerical simulations are presented.

In this study [21], an FLC has been implemented for mobile robot navigation in an unknown, dynamically cluttered environment is presented. an FLC has been employed due to it can conclude even in the face of uncertainty. To get to the desired position in a variety of circumstances, it enhances in the creation of rules and the decision-making procedures. The output of the FLCs is the acceleration of the corresponding wheels, while the inputs are the robot's sensor readings and the intended path of motion. As a result, the mobile robot gets to the desired location while avoiding impediments.

FLC and PID have been utilized for formation control and trajectory tracking in this study [22], which examines the coordinated movement of mobile robots grouped in a triangle. Under optimal circumstances, this study shows that both controllers can maintain formation and navigate effectively.

In the aforementioned studies, different strategies have been proposed and examined. However, most of these studies utilize large-scale fuzzy rule bases and limit their testing to only one type of membership function. In this work, the proposed controller is designed using a highly compact 3×3 rule base and is uniquely verified using two Various categories of membership functions (triangular and Gaussian) to evaluate their impact on tracking performance. This distinguishes our approach by achieving high efficiency with structural simplicity compared to previous studies.

3. MODEL OF MOBILE ROBOT

Two separate actuators control the two typical, parallel wheels on either side of the robot. It is also assumed that all wheels are perpendicular to the ground and that there is only

4. CONTROL BASED ON FUZZY LOGIC CONTROLLER

rolling and no slippage of the wheels on the ground [7]. This paper examines a kinematic mobile robot that is nonholonomic and has two wheels.

Kinematic modeling concentrates on the geometric relationships that control the system and examines the mathematics of movement without taking into account outside influences [12]. The differential drive mobile robot's location along the Global Cartesian system of {O, X, Y} [6]. Under these constraints, the differential drive robot's motion equations in the world frame are as follows [1, 7, 8 ,12].

The motion of a two-wheel mobile robot has been represented by Eq. (1).

$$\left.\begin{aligned} V_r &= r\,\omega_r \\ V_l &= r\,\omega_l \end{aligned}\right\} \tag{1}$$

In this case, ω_r represents the right driving wheel's angular velocity (rads^{-1}), ω_l represents the left driving wheel's angular velocity (rads^{-1}). V_r : the right wheel linear velocity (ms^{-1}), V_l: the left wheel linear velocity(ms^{-1}). r: right and left wheel radius (m).

The following is the nonholonomic constraint equation for the robot:

$$\dot{y}\cos(\theta) - \dot{x}\sin(\theta) = 0 \tag{2}$$

The following is the definition of the robot's dynamic function:

$$\left.\begin{aligned} \dot{x} &= V\cos(\theta) \\ \dot{y} &= V\sin(\theta) \\ \dot{\theta} &= \omega \end{aligned}\right\} \tag{3}$$

V represents the linear velocity (ms^{-1}), ω: the angular velocity (rads^{-1}). The previously mentioned formulas can be expressed in matrix form, and Eq. (4) is the modified kinematic model for the differential drive mobile robot that is utilized in its design:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix} \tag{4}$$

The model above can be enhanced by converting these velocity components into rotational velocities (ω_r, ω_l), as follows:

$$\begin{bmatrix} V \\ \omega \end{bmatrix} = \begin{bmatrix} r/2 & r/2 \\ r/L & -r/L \end{bmatrix} \begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix} \tag{5}$$

L represents the distance between right and left wheels (m).

The total of the two robot speeds, V_r and V_l, is its velocity. Furthermore, the following is the interdependence between the robot's angular speed, V_r and V_l, and the distance between the two wheels:

$$\left.\begin{aligned} V &= \frac{1}{2}(V_r + V_l) \\ \dot{\theta} &= \frac{1}{L}(V_r - V_l) \end{aligned}\right\} \tag{6}$$

A description of the trajectory tracking problem is given below: Assumed to be true are the positions of the robot ($p_r = [x_r \quad y_r \quad \theta_r]^T$) as well as the target ($p_t = [x_t \quad y_t \quad \theta_t]^T$). The distance between the robot's present location and its target position can be expressed utilizing the following equation [1, 12]:

$$d = \sqrt{\left(X_{target} - X_{robot}\right)^2 + (Y_{target} - Y_{robot})^2} \tag{7}$$

The trajectory's desired angle, θ, is calculated as follows:

$$\theta = \tan^{-1}\frac{Y_{target} - Y_{robot}}{X_{target} - X_{robot}} \tag{8}$$

The present location of the robot, or its previous location in the x-y plane, is represented by X_{robot} and Y_{robot}, whereas X_{target} and Y_{target} are the desired target or the new position.

In this research, FLC has been used for trajectory tracking for a two-wheel mobile robot. Fuzzification, the fuzzy inference engine (decision logic), and the defuzzification phases are the three primary parts of the FLC. Figure 1 displays the FLC block diagram. Fuzzification, the first block in the figure, looks up each element of input data in one or more membership functions to convert it to degrees of membership.

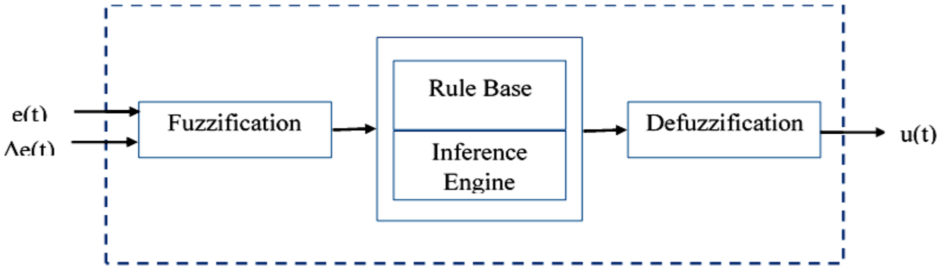


Figure 1. The fuzzy logic controller (FLC)'s structure

It is known as defuzzification. Defuzzification transforms the resultant fuzzy set into a clear control signal [13]. By using fuzzy implication and the rules of inference in fuzzy logic, the rule base and inference base may infer fuzzy control actions and simulate human decision-making based on fuzzy notions. Control performance is significantly impacted by the fuzzy control rules and fuzzy sets' membership functions.

The third operation, the Eqs. (9) and (10) represent the distance and angle errors, respectively between the robot and reference target [1].

$$e_d = \sqrt{\left(\hat{X}_x\right)^2 + \left(\hat{X}_y\right)^2} \tag{9}$$

$$e_\theta = (\theta_r - \theta_f) = \hat{X}_\theta \tag{10}$$

In the $X - Y$ plane as the robot moves. On the $X - Y$ plane, the robot's coordinates are $R_f(x_f, y_f)$ and the reference trajectory is expressed as $R_r(x_r, y_r)$. The discrepancy between the follower robot and the desired path is represented by the tracking error vector, which is represented by $\hat{X}$ [8].
where $\hat{X}$ represents in Eq. (11):

$$\hat{X} = \begin{bmatrix} x_r - x_f \\ y_r - y_f \\ \theta_r - \theta_f \end{bmatrix} = \begin{bmatrix} \hat{X}_x \\ \hat{X}_y \\ \hat{X}_\theta \end{bmatrix} \tag{11}$$

5. RESULTS AND DISCUSSION

Using the FLC controller, the autonomous wheeled mobile robot's trajectory is tracked. The general block structure of a mobile robot with the block diagram of the suggested FLC for the trajectory-tracking control system is shown in Figure 2. The structure of each FLC contains two inputs: the error e(t) and the change of error Δe(t), and one output.

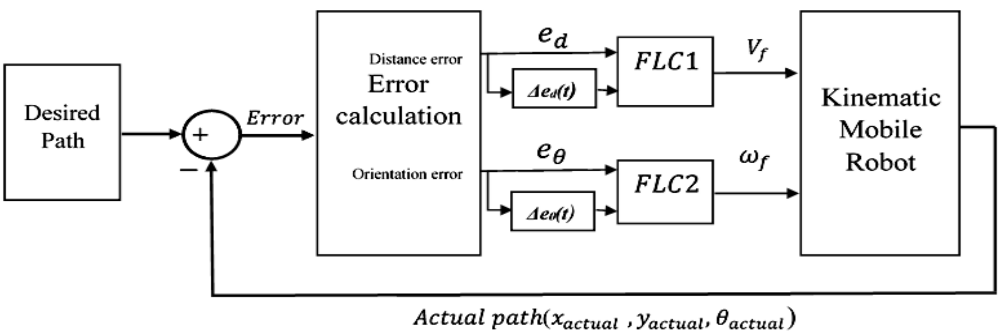


Figure 2. The general block structure of a mobile robot

In this study, two FLCs are used to eliminate orientation and distance tracking errors. The controller has to reduce both errors to zero, ensuring that the robot maintains the intended trajectory. As a result, two FLC are designed to reach and track the desired path. The first fuzzy controller, known as the Distance Error Controller, reduces the robot's perpendicular distance from the target path so that it can approach and follow it. Its output is velocity. The second controller, known as the Orientation Error Controller, aligns the robot with the target path by using the variation between the goal angle and the present robot angle as input. Its output is angular velocity.

The computer specifications used for the simulation experiment had been as follows: the processor consisted of an Intel(R) Core(TM) i7-1165G7U CPU running at 2.80 GHz, and the operating system was Windows 10, 16-bit. To minimize the tracking error between the reference and real-time trajectories, manual experiments have been conducted to address this problem using the proposed FLC controllers.

As shown in Figure 2 the two inputs to FLC1 are e_d and Δe_d(t), i.e., 'Distance Error Controller', are described by three fuzzy linguistic values: Negative, Zero, Positive for each input. The output, i.e., for FLC1, is velocity described by three fuzzy sets: Low, Medium, and High. Normalization is applied to the fuzzy controller's inputs in the interval [-5, 5] for position error and the interval [-2, 2] for Δ(position error), while the interval for velocity outputs is [0, 1].

The two inputs to FLC2 are e_θ and Δe_θ(t), i.e., 'Orientation Error Controller', are described by three fuzzy linguistic values: Negative, Zero, Positive for each input. The output, i.e., for FLC2, is angular velocity described by three fuzzy sets: Low, Medium, and High. Normalization is applied to the fuzzy controller's inputs in the interval [-3.14, 3.14] for theta error and the interval [-3.14, 3.14] for Δ(theta error), while the interval for angular velocity outputs is [-180, 180].

A two-dimensional linear rule base and an inference engine of the Mamdani type are used in Table 1 to demonstrate the relationship between input and output.

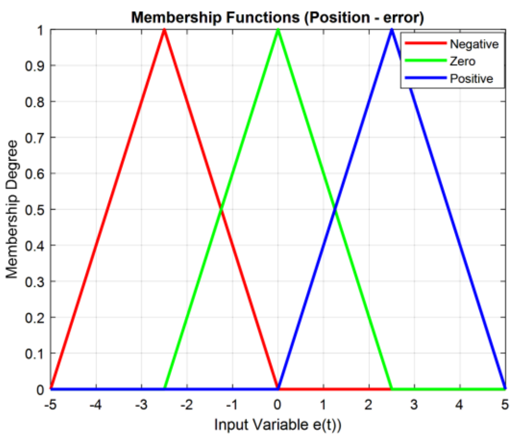
Table 1. The rule base for error

e(t) / Δe(t)	Negative	Zero	Positive
Negative	Low	Low	Medium
Zero	Low	Medium	High
Positive	Medium	High	High

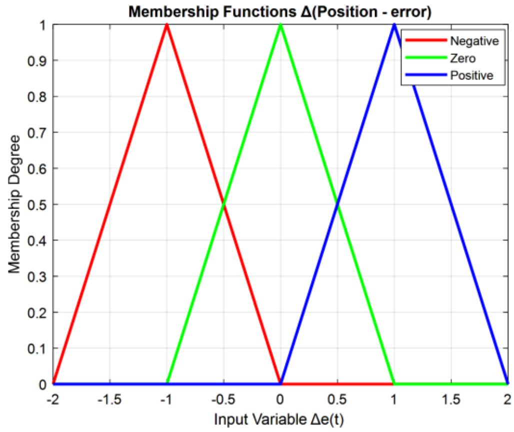
The simulation results tested with the Step trajectory shape by using two different membership functions are as follows.

5.1 First scenario: Triangle membership function

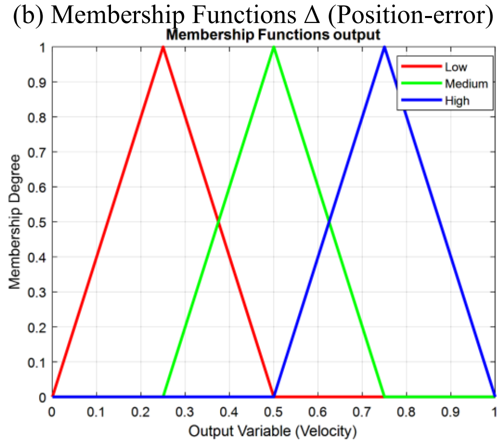
Figures 3(a)-(c) illustrate the triangle MFs for the proposed FLC1, and Figure 4 depicts the control surface.



(a) Membership functions (position-error)



(b) Membership Functions Δ (Position-error)



(c) Membership Functions Output

Figure 3. Triangle membership functions (MFs) for the distance error controller

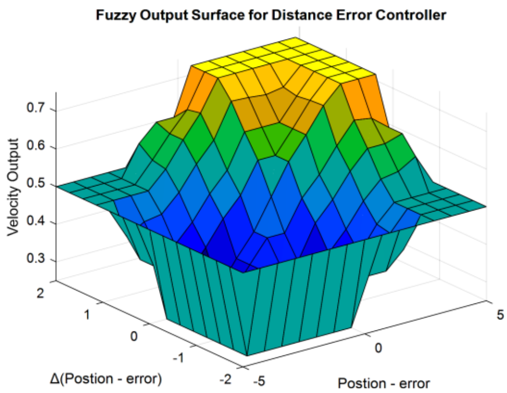
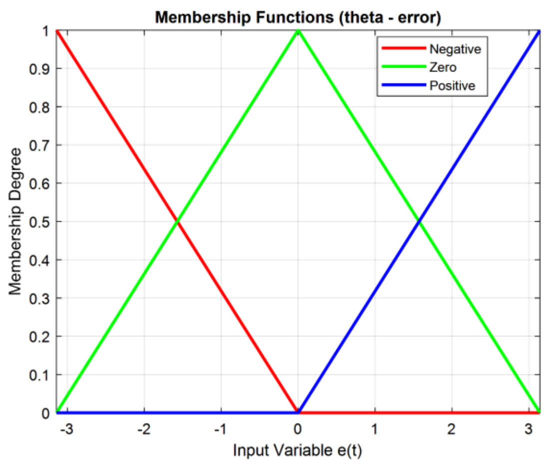
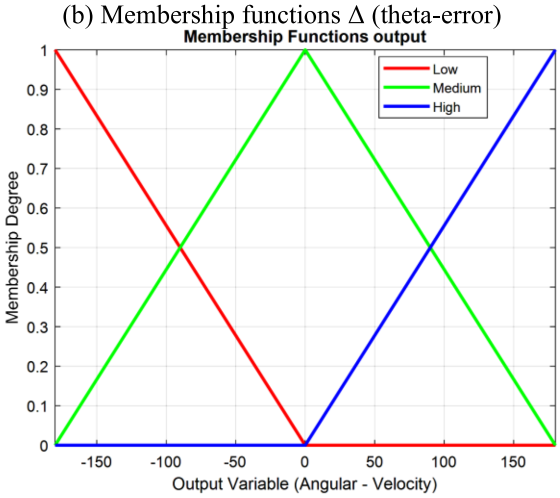


Figure 4. Output surface for distance error controller



(a) Membership functions (theta-error)

(b) Membership functions Δ (theta-error)



(c) Membership functions output

Figure 5. Triangle membership functions (MFs) for the orientation error controller

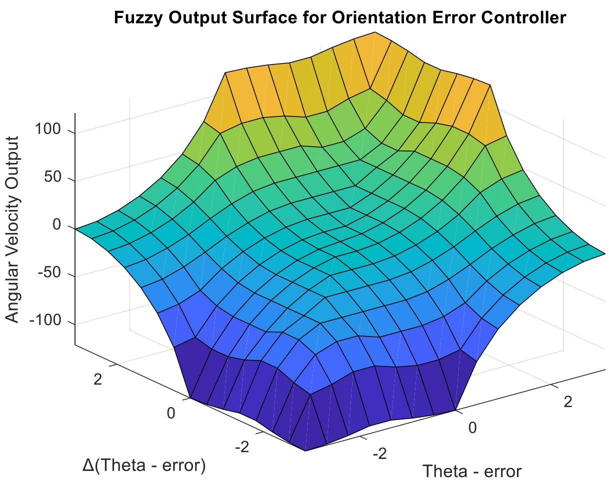


Figure 6. Output surface for orientation error controller

Figures 5(a)-(c) illustrate the triangle MFs for the proposed FLC2, and Figure 6 depicts the control surface.

In Figures 7 and 8, the step and theta responses for the trajectory show the result of testing the mobile robot's ability to follow the step path using triangle membership functions.

The linear and angular velocities have been examined using the two FLC controllers, as seen in Figure 9. Figure 10 shows the wheel velocities on the right and left. Theta and position errors are displayed in Figure 11.

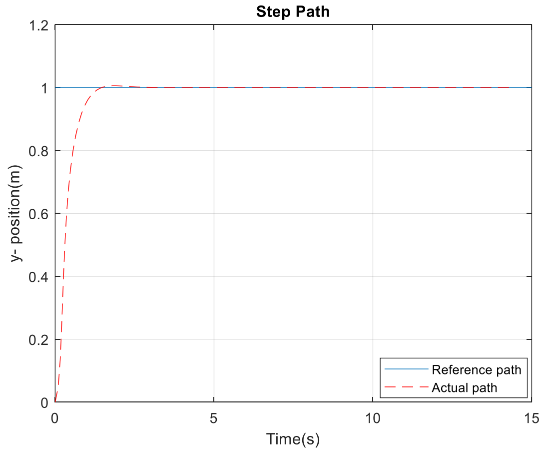


Figure 7. The step trajectory response

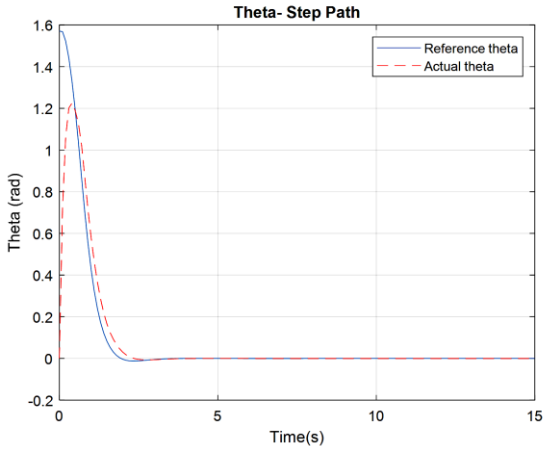
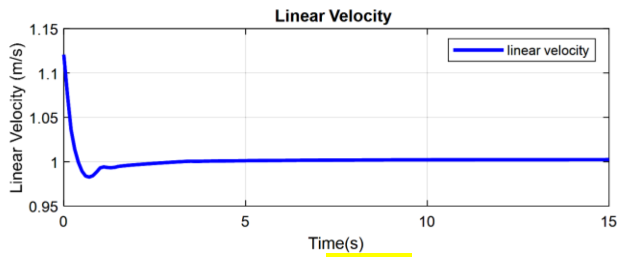
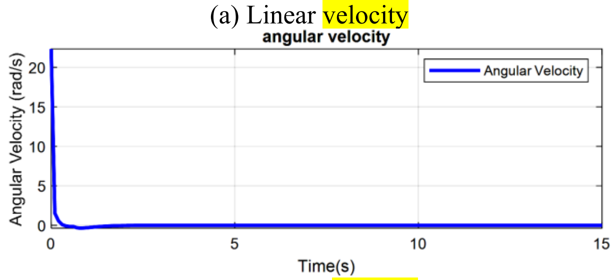


Figure 8. Theta response

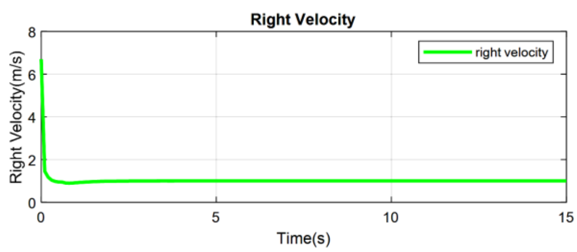


(a) Linear velocity

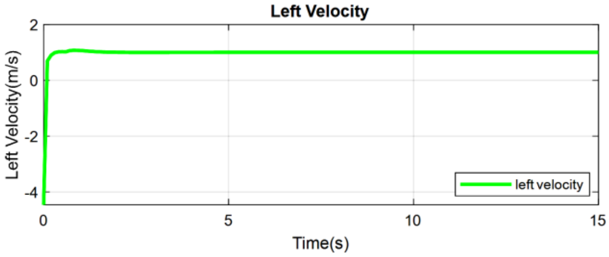


(b) Angular velocity

Figure 9. Linear and angular velocities

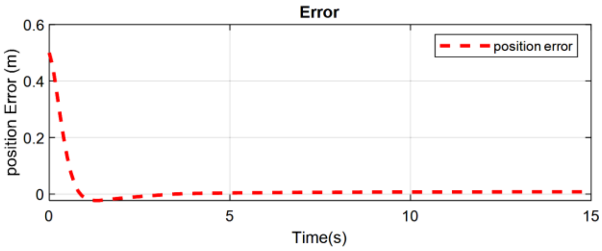


(a) Right velocity

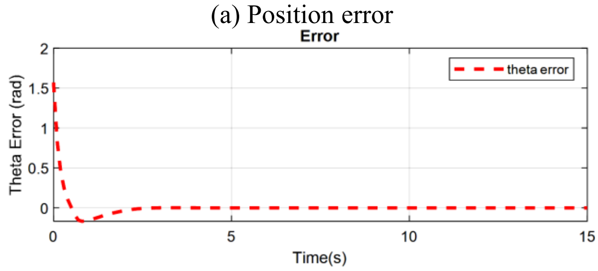


(b) Left velocity

Figure 10. Right and left velocities

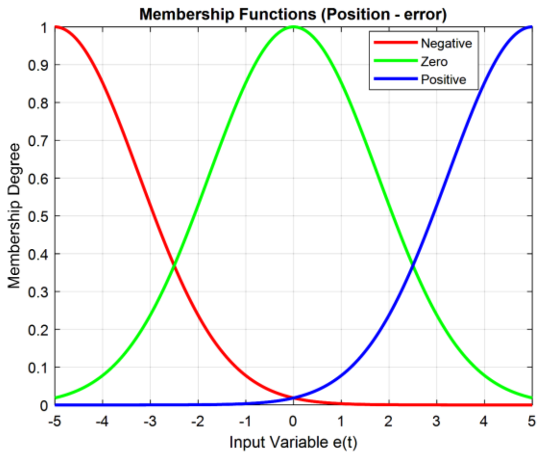


(a) Position error

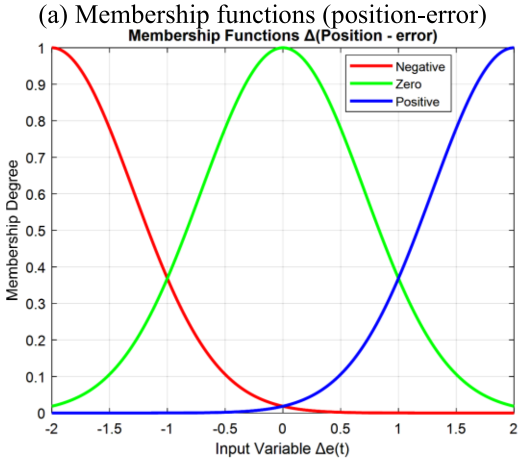


(b) Theta error

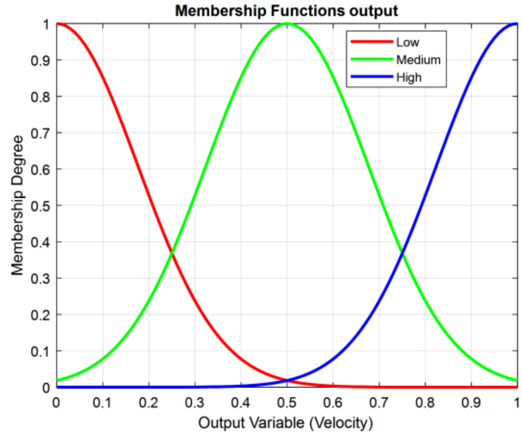
Figure 11. Position and theta error



(a) Membership functions (position-error)



(b) Membership functions Δ (position-error)



(c) Membership functions output

Figure 12. Gaussian membership functions (MFs) for distance error controller

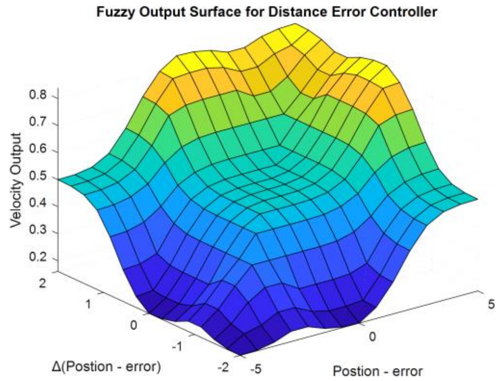


Figure 13. Output surface using Gaussian membership functions (MFs) for the distance error controller

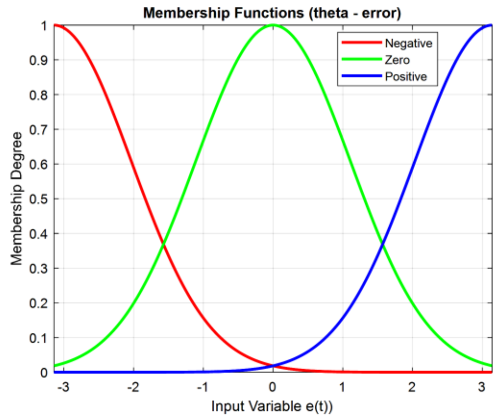
5.2 Second scenario: Gaussian membership function

Figures 12(a)-(c) illustrate the Gaussian MFs for the proposed FLC1, and Figure 13 depicts the control surface.

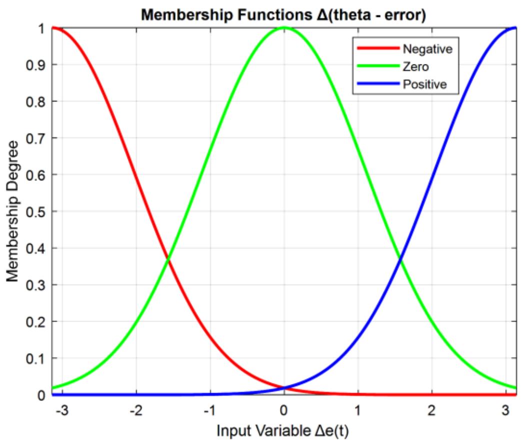
Also, Figures 14(a)-(c), respectively, show the Gaussian membership function for the proposed FLC2, and Figure 15 depicts the control surface.

In Figures 16 and 17, the step and theta responses for the trajectory show the result of testing the mobile robot's ability to follow the step path using Gaussian membership functions.

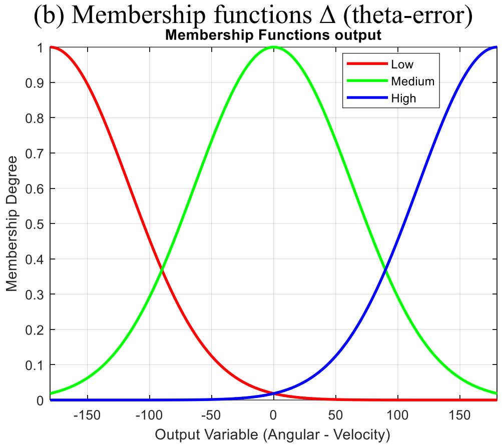
The linear and angular velocities have been examined using the two FLC controllers, as seen in Figure 18. Figure 19 shows the wheel velocities on the right and left. Theta and position errors are displayed in Figure 20.



(a) Membership functions (theta-error)



(b) Membership functions Δ (theta-error)



(c) Membership functions (position-error)

Figure 14. Gaussian membership functions (MFs) for the distance error controller

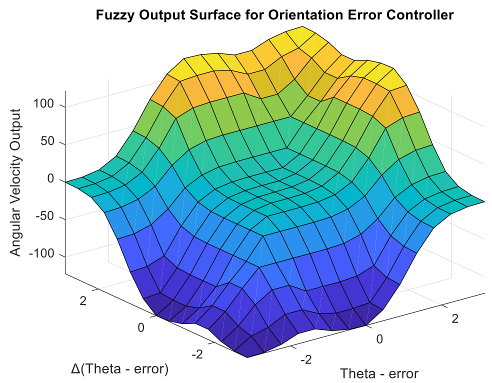


Figure 15. Output Surface using Gaussian membership functions (MFs) for distance error controller

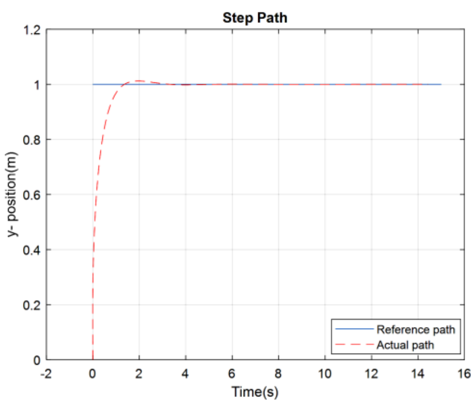


Figure 16. The step trajectory response

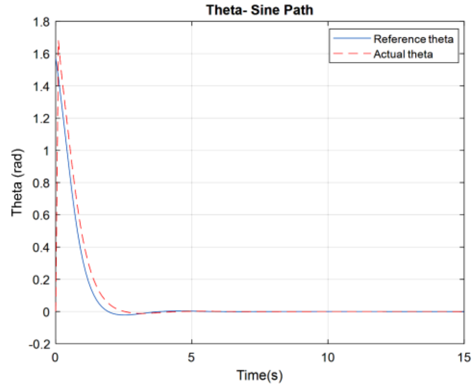
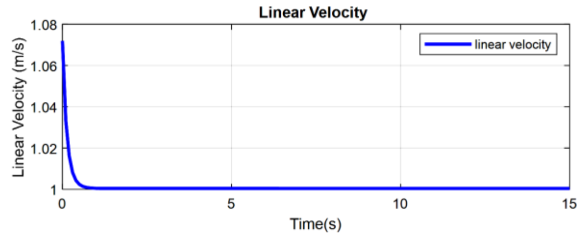
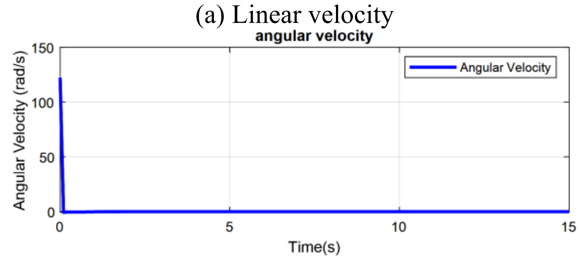


Figure 17. Theta response

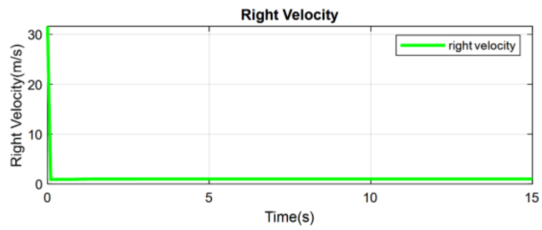


(a) Linear velocity

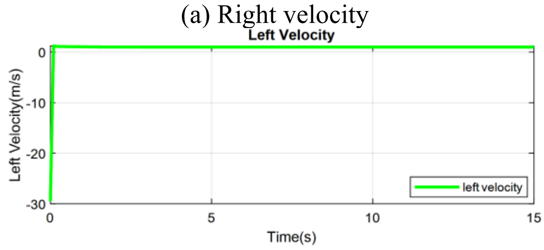


(b) Angular velocity

Figure 18. Linear and angular velocities by using Gaussian membership functions (MFs)

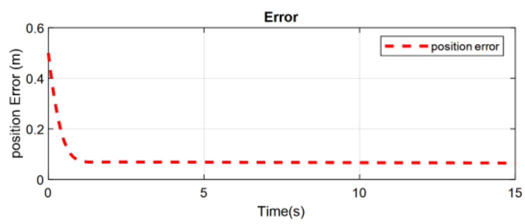


(a) Right velocity

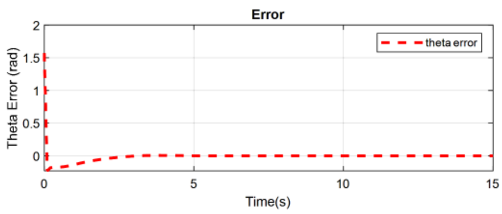


(b) Left velocity

Figure19. Right and left velocities by using gaussian membership functions (MFs)



(a) Position error



(b) Theta error

Figure 20. Position and theta error using Gaussian membership functions (MFs)

According to the results obtained in this section, the system response using two types of membership functions (triangular and Gaussian) is compared in Tables 2 and 3.

The test results revealed the following significant findings: The simulation demonstrates that Triangle MF outperforms Gaussian MF. The results showed that Triangle MF is more successful than Gaussian MF, reducing steady-state error by 100.00%, overshoot by 50.6758%, rising time by 8.3333%, peak time by 8.0000%, and mean position error by 78.8387%.

The percentage (P%) improvement in performance for the Triangle MF compared to the Gaussian MF was calculated using the following equation:

$$P\% = \frac{|value\,1\,(Triangle) - value\,2(Gaussian)|}{sum\ of\ two\ values} * 100\% \qquad (12)$$

Table 2. The system response for the step path

Membership Function	Steady State Error (m)	Overshoot %	Peak Time (s)	Rise Time (s)	Settling Time (s)
Triangle	2.2204e-14	0.5948	2.300	1.100	1.600
Gaussian	7.8554e-05	1.2059	2.500	1.200	1.600

Table 3. The angular and linear velocity specifications for the two membership functions' response

Membership Function	Mean Linear Velocity (m/s)	Mean Angular Velocity (rad/sec)	Mean Position Error (m)	Mean Absolute Theta Error (rad)
Triangle	1.0021	0.1435	0.0164	0.0106
Gaussian	1.0013	0.7809	0.0775	0.0073

6. CONCLUSIONS AND FUTURE WORKS

This study provides a clear demonstration of how various membership function configurations influence the trajectory tracking performance of differential drive mobile robots. The various kinds of MFs have been described. This work looked at the two most common kinds, the Gaussian MF and the triangular MF. These two MFs have been used in the implementation of the FLC, with the same MF applied to the input and output variables. Analysis and comparison of the response revealed that the triangular MF outperformed the others with respect to peak time, rising time, overshoot, and steady-state behavior. Additionally, it decreased the average position inaccuracy. In contrast, the Gaussian MF typically performed worse in the majority of the outcomes. Based on the results from the simulations, Triangle MF significantly outperforms the Gaussian MF method in terms of performance. As a result, Triangle MF has been demonstrated in the simulation to be more efficient, reducing mean position errors by 78.8387%, rise times by 8.3333%, peak times by 8.00%, overshoot by 50.6758%, and steady state errors by 100.00%. The findings also demonstrate that the FLC provides sufficient tracking performance and error convergence while enabling the robot to follow a smoother path.

The current research has confirmed the effectiveness of human-configured fuzzy logic controllers (FLCs) as a means of tracking the path of two-wheeled mobile robots. Future research will expand on this by studying how to use these FLCs to track different geometric paths, such as circles and sine waves. In addition to this work, future research will also explore the robustness of the FLC design by performing a detailed analysis of the performance of the controller to find areas where improvements may be made. Although manual tuning provides a simple way to develop an FLC, it can be very time-consuming to get the results one desires, and there is no guarantee that the result will be an optimal one. The author will use optimization techniques to tune the membership functions and rule bases for the FLC in this research, and these include particle swarm optimization (PSO) and other meta-heuristic optimization techniques. Ultimately, this tuning will result in enhanced dynamic performance of the system as evidenced by reduced values of critical performance metrics such as overshoot, settling time, rise time, and steady state error.

REFERENCES

[1] Omrane, H., Masmoudi, M.S., Masmoudi, M. (2016). Fuzzy logic based control for autonomous mobile robot navigation. Computational Intelligence and Neuroscience, 2016(1): 9548482. https://doi.org/10.1155/2016/9548482

[2] Bingol, M.C. (2025). A safe navigation algorithm for differential-drive mobile robots by using fuzzy logic reward function-based deep reinforcement learning. Electronics, 14(8): 1593. https://doi.org/10.3390/electronics14081593

[3] Niloy, M.A., Shama, A., Chakrabortty, R.K., Ryan, M.J., et al. (2021). Critical design and control issues of indoor autonomous mobile robots: A review. IEEE Access, 9: 35338-35370. https://doi.org/10.1109/ACCESS.2021.3062557

[4] Hassan, I.A., Abed, I.A., Al-Hussaibi, W.A. (2023). Navigation and path planning of mobile robots. Journal Européen des Systèmes Automatisés, 56(5): 751-756. https://doi.org/10.18280/jesa.560505

[5] Puriyanto, R.D., Mustofa, A.K. (2024). Design and

implementation of fuzzy logic for obstacle avoidance in differential drive mobile robot. Journal of Robotics and Control, 5(1): 132-141. https://doi.org/10.18196/jrc.v5i1.20524

[6] Lee, K., Im, D.Y., Kwak, B., Ryoo, Y.J. (2018). Design of fuzzy-PID controller for path tracking of mobile robot with differential drive. International Journal of Fuzzy Logic and Intelligent Systems, 18(3): 220-228. https://doi.org/10.5391/IJFIS.2018.18.3.220

[7] Nguyen, A.T., Vu, C.T. (2022). Mobile robot motion control using a combination of fuzzy logic method and kinematic model. In Intelligent Systems and Networks: Selected Articles from ICISN 2022, Vietnam, pp. 495-503. https://doi.org/10.1007/978-981-19-3394-3_56

[8] Hassan, I.A., Abed, I.A., Al-Hussaibi, W.A. (2024). Path planning and trajectory tracking control for two-wheel mobile robot. Journal of Robotics and Control, 5(1): 1-15. https://doi.org/10.18196/jrc.v5i1.20489

[9] Ridzuan, A., Rahman, H.A. (2024). Real time comparison between PID and fuzzy logic controller for DC motor speed control. Evolution in Electrical and Electronic Engineering, 5(1): 512-520. https://penerbit.uthm.edu.my/periodicals/index.php/eeee/article/view/15396.

[10] Ali, A.M., Y Ali, A., Salem Sumait, B. (2015). Comparison between the effects of different types of membership functions on fuzzy logic controller performance. International Journal of Emerging Engineering Research and Technology, 3: 76-83. https://www.researchgate.net/publication/282506091.

[11] Gaurav, A.K., Kaur, A. (2012). Comparison between conventional PID and fuzzy logic controller for liquid flow control: Performance evaluation of fuzzy logic and PID controller by using MATLAB/Simulink. International Journal of Innovative Technology and Exploring Engineering, 1(1): 84-88.

[12] Štefek, A., Pham, V.T., Krivanek, V., Pham, K.L. (2021). Optimization of fuzzy logic controller used for a differential drive wheeled mobile robot. Applied Sciences, 11(13): 6023. https://doi.org/10.3390/app11136023

[13] Bingül, Z., Karahan, O. (2011). A Fuzzy Logic Controller tuned with PSO for 2 DOF robot trajectory control. Expert Systems with Applications, 38(1): 1017-1031. https://doi.org/10.1016/j.eswa.2010.07.131

[14] Faisal, M., Hedjar, R., Al Sulaiman, M., Al-Mutib, K. (2013). Fuzzy logic navigation and obstacle avoidance by a mobile robot in an unknown dynamic environment. International Journal of Advanced Robotic Systems, 10(1): 37. https://doi.org/10.5772/54427

[15] Farooq, U., Hasan, K.M., Hanif, A., Amar, M., Asad, M.U. (2014). Fuzzy logic based path tracking controller for wheeled mobile robots. International Journal of Computer and Electrical Engineering, 6(2): 145-150. https://doi.org/10.7763/IJCEE.2014.V6.811

[16] Faress, K.N., El Hagry, M.T., El Kosy, A. (2005). Trajectory tracking control for a wheeled mobile robot using fuzzy logic controller. WSEAS Transactions on Systems, 4(7): 1017-1021.

[17] Al-Dahhan, M.R.H., Ali, M.M. (2016). Path tracking control of a mobile robot using fuzzy logic. In 2016 13th International Multi-Conference on Systems, Signals & Devices (SSD), Leipzig, Germany, pp. 82-88. https://doi.org/10.1109/SSD.2016.7473656

[18] Abdelwahab, M., Parque, V., Elbab, A.M.F., Abouelsoud, A.A., Sugano, S. (2020). Trajectory tracking of wheeled mobile robots using z-number based fuzzy logic. IEEE Access, 8: 18426-18441. https://doi.org/10.1109/ACCESS.2020.2968421

[19] Kasmi, B., Hassam, A. (2021). Comparative study between fuzzy logic and interval type-2 fuzzy logic controllers for the trajectory planning of a mobile robot. Engineering, Technology & Applied Science Research, 11(2): 7011-7017. https://doi.org/10.48084/etasr.4031

[20] Al-Momani, F., Al-Tamimi, A., Al-Jarrah, A., Al-Jarrah, A., Salah, M. (2022). Tracking control for a two-wheel differentially driven nonholonomic mobile robot: Performance comparison. International Journal of Mechanical Engineering and Robotics Research, 11(10): 724-736. https://doi.org/10.18178/ijmerr.11.10.724-736

[21] Kumar, A., Sahasrabudhe, A., Nirgude, S. (2024). Fuzzy logic control for indoor navigation of mobile robots. arXiv preprint arXiv:2409.02437. https://doi.org/10.48550/arXiv.2409.02437

[22] Zakipour Bahambari, S.M., Khankalantary, S. (2025). Trajectory tracking and formation control of mobile robots using fuzzy logic controller with obstacle avoidance. AUT Journal of Electrical Engineering, 57(1): 55-70. https://doi.org/10.22060/eej.2024.23254.5596