



Article Type : Research Article

Received : July 30, 2025

Revised : November 09, 2025

Accepted : December 23, 2025

DOI : [10.17798/bitlisfen.1753970](https://doi.org/10.17798/bitlisfen.1753970)

Year : 2025

Volume : 14

Issue : 4

Pages : 2523-2551



TRAJECTORY TRACKING OF A MOBILE ROBOT WITH GWO-BASED TYPE-II FUZZY LOGIC CONTROLLER

Tayfun ABUT^{1,*} , **Enver SALKIM**² 

¹ *Muş Alparslan University, Mechanical Engineering Department, Muş, Türkiye*

² *Muş Alparslan University, Electronics and Automation Department, Muş, Türkiye*

** Corresponding Author: t.abut@alparslan.edu.tr*

ABSTRACT

This study addresses the trajectory tracking problem of wheeled mobile robots and proposes two different control strategies, which are comparatively evaluated through simulations. Initially, the mathematical model of the mobile robot is derived, followed by the design of two controllers: a conventional Proportional-Integral-Derivative (PID) controller and a Type-II Fuzzy Logic Controller (Type-II FLC) optimized utilizing the Grey Wolf Optimizer (GWO) algorithm. The suggested control methods are assessed under three different reference trajectory scenarios—circle, square, and star-shaped paths. Simulation results indicate that the PID controller exhibits significant deviations, particularly during sharp turns and sudden maneuvers in square and star trajectories, leading to increased tracking errors. In contrast, the GWO-based Type-II FLC demonstrates smoother and more stable maneuvers, resulting in lower tracking errors and higher trajectory-following accuracy. These findings suggest that the GWO-optimized Type-II FLC enables more reliable and precise navigation of mobile robots in dynamic environments contrasted to the classical PID controller. Furthermore, the Type-II FLC method achieved considerable percentage improvements in tracking performance for circular, square, and star trajectories when contrasted with the PID controller. Specifically, in a square orbit, the X and Y position errors have been improved by 98% and 97%, respectively, according to the Type II FLC, PID control method. Overall, the results highlight the effectiveness of the suggested control method in enhancing position control accuracy.

Keywords: Mobile Robot, Type II Fuzzy Logic Control, PID Controller, Grey Wolf Optimization Algorithm.

1 INTRODUCTION

Wheeled mobile robots (WMRs) have garnered increasing attention in recent years due to their increasing use across various fields such as industrial automation, military applications, agriculture, healthcare, and service industries. Particularly, wheeled mobile robots are widely preferred for both indoor and outdoor tasks owing to their structural simplicity, low energy consumption, and high maneuverability. One of the fundamental tasks of these robots is to follow a predefined reference trajectory with minimal error. However, uncertainties in the system model, environmental disturbances, and trajectory scenarios involving sharp turns can significantly challenge the effectiveness of conventional control strategies. In response to these challenges, a range of advanced control methods have been developed to increase the trajectory tracking performance of WMRs. Caracciolo et al. [1] suggested a trajectory tracking control (TTC) design for four-wheeled (4W) differentially driven mobile robots by developing a dynamic model that accounts for ground slippage effects. To mitigate excessive slippage, the control approach incorporates a nonlinear controller along with nonholonomic constraints. Additionally, uncertainties related to ground surface parameters were addressed using robust linear techniques. Simulation results demonstrated that the suggested control strategy achieved high performance in virtual terrain environments. Kozłowski and Pazderski suggested a control method by extending a kinematic control law to the dynamic and actuator levels utilizing Lyapunov-based analysis and the backstepping technique [2]. The approach was implemented in simulation, where various trajectory scenarios were tested. The results indicated that the controller's performance is influenced by the dimensionality of the state vector. In a related study, a FLC and a path-tracking control strategy were developed for a 4W, differentially steered mobile robot [3]. The suggested controller enables high-level control of the robot, allowing it to follow predefined waypoints along a path without the need for continuous trajectory computation. It dynamically adjusts the robot's velocity in response to environmental changes. Experiments conducted on the P3-AT mobile robot demonstrated that the suggested method is effective in respect of both accuracy and response time. Kumar and Sukavanam developed a trajectory tracking controller based on the backstepping technique, utilizing the kinematic model of a four-wheeled mobile robot [4]. Considering the system's nonholonomic constraints specifically, pure rolling and no lateral slipping the robot's model was transformed into a chained form through coordinate transformations. The control law was then created based on this structure. Simulation results demonstrated that the suggested approach enables the robot to asymptotically follow the desired trajectory, confirming the effectiveness of the control strategy.

Cariou et al. [5] developed a high-precision path tracking control system for four-wheel steering mobile robots (4WSMRs). The suggested approach compensates for slippage effects encountered on low-friction surfaces by addressing both lateral and angular deviations, achieving accurate path tracking with a lateral error of ± 10 cm and an angular error of ± 2 degrees. Furthermore, by integrating feedback and predictive algorithms, the system effectively mitigates steering delays and transient deviations.

Lee and Li developed an integrated control system for mobile robots equipped with independently steered and driven wheels [6]. Due to the individual actuation of each wheel by separate motors, the study suggested a trajectory tracking algorithm that combines nonlinear kinematic control with dynamic sliding mode control. The accuracy of the system was confirmed by simulation results, and the stability of the suggested control schemes was examined using Lyapunov-based techniques. A neural network-based adaptive tracking controller that can account for uncertainties such as wheel slippage and outside disturbances was suggested by Hoang and Kang [7]. The controller utilizes online weight adaptation laws to effectively mitigate the impact of these uncertainties. System stability was formally proven using Lyapunov-based analysis. Simulation results demonstrated that the suggested controller achieved high tracking accuracy and maintained robust performance in the presence of slippage and external forces. Liao et al. [8] suggested a two-level coordinated adaptive robust control architecture to address vibration issues arising from varying ground conditions in four-wheeled independently driven skid-steer mobile robots. At the high level, an adaptive robust control law was designed for trajectory tracking, while at the low level, a torque allocation algorithm was implemented to balance the motor torques. Additionally, a novel feedback mechanism was integrated into the system to mitigate wheel slippage. Simulation results illustrated that the suggested approach efficaciously reduced vibrations, enhanced robustness against terrain variations, and outperformed conventional methods in terms of control performance. Deremetz et al. [9] improved a control approach for 4WS mobile robots aimed at achieving precise path tracking in challenging terrain conditions. The study employed an extended kinematic model based on the bicycle model, combined with a backstepping control strategy, to ensure stable orientation and lateral position control of the robot. Two separate control laws were suggested based on symmetric steering of the front and rear wheels. Additionally, slip angles were estimated using an observer to compensate for lateral slippage. This method enhances maneuverability in confined environments and offers more robust path tracking compared to previous approaches.

Liao et al. [10] developed a model-based, coordinated, adaptive, and robust control system for four-wheeled independently driven mobile robots. Addressing the limitations of conventional kinematic control methods, the study introduced a comprehensive model that incorporates chassis kinematics, overall system dynamics, and individual wheel dynamics, with the goal of mitigating vibration issues and preventing over-actuation. The suggested three-layer control architecture generates motor torque commands for each wheel, with control implemented at each level by considering slip references. System stability and tracking performance are guaranteed by theoretical analysis, and actual findings show that the suggested strategy performs better than conventional techniques. Qiu et al. [11] suggested an extended Ackermann Steering Principle and a coordinated motion control strategy for four-wheel drive and front/rear-steering agricultural mobile platforms. Designed for agricultural tasks such as soil data collection and pesticide spraying, the platform utilizes a control approach aimed at minimizing linear velocity deviations during turning maneuvers. The suggested strategy includes a predictive model for estimating front and rear steering angles, determines the optimal location of the virtual turning center, and minimizes the angle differences between inner and outer wheels while computing linear velocities based on the Ackermann principle. Experimental results indicate that the suggested method significantly reduces wheel slippage during turns and lowers motor energy consumption. Zeidis and Zimmermann investigated the dynamics of 4W Mecanum mobile robots under nonholonomic constraints and demonstrated the suitability of using Chaplygin equations in modeling such systems [12]. The study showed that, particularly for certain specific motion types, solutions derived through the use of a pseudo-inverse matrix closely align with exact analytical solutions, confirming the validity of the suggested modeling approach. Abut and Huseyinoglu created a mathematical model for the WMR and utilized the Firefly Optimization Algorithm based on the Linear Quadratic Regulator control method for optimal trajectory tracking [13]. Positive results were obtained from the controller. Jiang et al. [14] suggested a robust trajectory tracking control method for 4W autonomous mobile robots, designed to withstand system disturbances and uncertainties. Within the framework of direct yaw moment control, a restructured robust super-twisting sliding mode controller was developed to suppress both matched and time-varying disturbances. This approach, based on a novel reaching law, enables the robot to maintain stable trajectory tracking within a defined region and ensures finite-time convergence. Lyapunov-based analysis was used to mathematically demonstrate the system's stability, and simulation results validated the efficacy of the recommended approach. Zhang et al. [15] developed a flexible and robust navigation system for four-wheel independently driven and steered mobile

robots equipped with multiple maneuvering modes. The system employs a low-complexity algorithm that utilizes hybrid maneuvering modes for global path planning, while local planning is used to ensure collision avoidance. For trajectory tracking, a fuzzy adaptive control approach supported by neural networks was suggested. The controller was designed using second-order programming and neural network optimization techniques. According to experimental data, the suggested system functioned more accurately and efficiently than traditional techniques. Raikwar et al. [16] developed a model-based navigation and control system for an orchard robot designed to operate in agricultural environments where GNSS signal interruptions are common. The system relies on data from non-GNSS sensors, specifically wheel and steering encoders, to guide the robot. The entire system was modeled using MATLAB/Simulink. Due to the robot's low-speed operation, dynamic effects were considered negligible, and the control architecture was divided into speed and steering command components for navigation. The suggested system was validated through both simulations and outdoor field tests. Reported results indicated a steering error of approximately 0.2–0.4 degrees during turns and around 0.05 degrees during straight-line motion, with model correlation values approaching 0.99. These findings demonstrate that the suggested method ensures stable navigation with minimal sensor dependency.

Szeremeta and Szuster aimed to improve a neural control algorithm for a four-wheeled Mecanum mobile robot [17]. Simulations carried out in MATLAB/Simulink, together with experiments on the real platform, showed that the neural controller achieved higher motion accuracy than a proportional–derivative controller. Designed within a Lyapunov stability framework, the algorithm maintained robust performance despite model perturbations and uncertainties. Overall, the neural control scheme improved both the motion quality and the tracking accuracy of the robot. Qu et al. [18] conducted control parameter optimization to enhance the motion mode switching performance of a 4WSMR used in agricultural applications. The effects of key factors such as stepper motor speed and locking voltage were experimentally evaluated, and optimal parameter combinations were identified through statistical analysis. Validation tests performed on a prototype robot demonstrated that the motion mode transitions occurred smoothly and stably under the optimized settings. Bae and Lee developed a compact, research-oriented robotic platform along with an adaptive manual steering control algorithm for four-wheel steering mobile robots [19]. The system enhances maneuverability in confined spaces by enabling steering capabilities on both the front and rear wheels, providing intuitive control during manual operation. The adaptive algorithm primarily

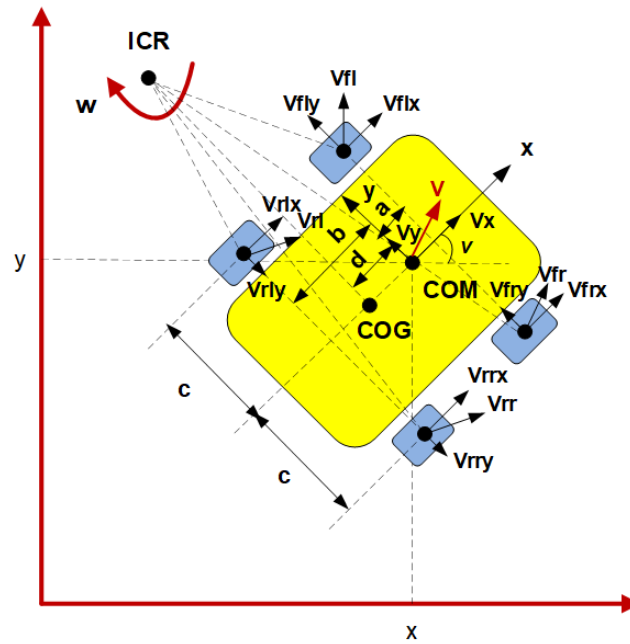
employs front-wheel steering under normal conditions, while activating rear-wheel steering inputs as the turning demand increases to reduce the turning radius. Experimental results demonstrated the system's high steering accuracy and validated the effectiveness of the suggested control algorithm. Zhang et al. [20] developed a trajectory tracking control method for four-wheeled skid-steered mobile robots subject to disturbances. The suggested approach employs a variable-gain disturbance observer to estimate unknown uncertainties and external disturbances. These estimates are integrated into a nonsingular terminal sliding mode controller that adapts to time-varying slippery surface conditions, enabling active disturbance rejection and guaranteeing the reduction of tracking errors within a finite time. Additionally, a virtual driving force allocation scheme is implemented to appropriately distribute motor torques. The stability and performance of the control strategy were validated through both theoretical analyses and experimental results. Ryu et al. [21] presented a study proposing seven performance criteria to evaluate the mobility capabilities of four-wheeled mobile robots. These criteria address issues such as oscillation, tipping, and slipping, while aiming to assess the robots' stability and energy efficiency. The study included simulations in a 3D environment, and factor effects were analyzed using Taguchi orthogonal arrays. The results highlighted the influence of design parameters particularly width, height, and length on overall performance. The authors also suggested that the suggested criteria could be applied for future evaluations of different robotic mechanisms.

Zhang et al. [22] developed an adaptive control strategy aimed at mitigating slip effects for four-wheeled climbing manipulation robots operating on inclined and vertical surfaces. The approach utilized integral terminal sliding mode control based on kinematic and dynamic modeling, with a focus on enhancing tracking accuracy during nonlinear trajectory following in three-dimensional environments. The stability of the suggested algorithm was examined utilizing Lyapunov theory, while simulation and experimental results demonstrated superior performance in both tracking accuracy and robustness contrasted to conventional PID and adaptive kinematic controllers. Chen developed a nonlinear adaptive fuzzy hybrid sliding mode control strategy for trajectory tracking of 4W Mecanum autonomous mobile robots [23]. This approach directly addresses inherent system nonlinearities, parameter uncertainties, torque saturation, and external disturbances. The suggested controller integrates sliding mode control, fuzzy logic, and adaptive learning components. Experimental research and simulation findings under 60% model uncertainty showed that the suggested approach outperforms traditional sliding mode control in terms of tracking accuracy, control signal smoothness, and energy

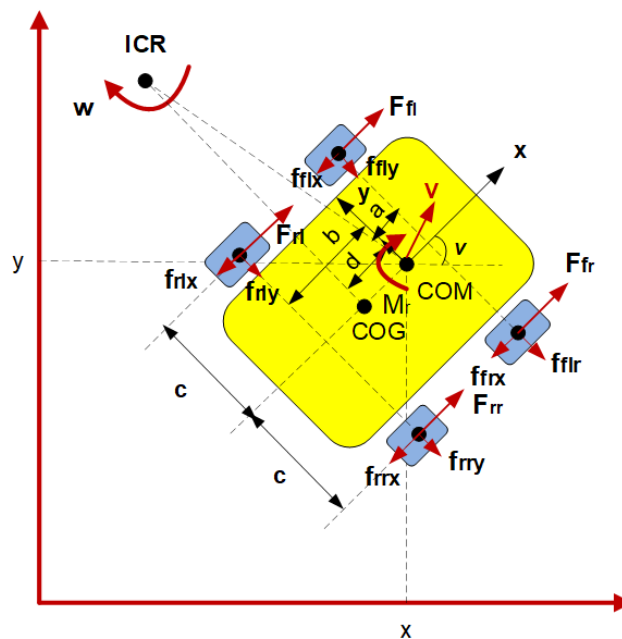
economy. Cipriano et al. [24] addressed the singularity problem in trajectory tracking for steerable WMRs. They suggested a real-time nonlinear model predictive control approach to manage the robot's motion. The suggested controller calculates the robot's movements while avoiding singularities and considering wheel speed constraints. To prevent singularities during motion start and stop phases, the system is supported by a finite state machine and dynamic feedback linearization techniques. In summary, control studies of mobile robots conducted in both simulation and experimental environments have been increasing in recent years [25-29]. In this study, a Type II FLC optimized by GWO is suggested and compared with a conventional PID controller. The GWO algorithm plays a significant role in reducing both positional and orientation errors owing to its efficacy in solving global optimization problems, while the Type II FLC demonstrates enhanced capability to make reliable decisions under system uncertainties. The application of this combined approach to different reference trajectories defined by geometric shapes such as circle, square, and star provides an ideal scenario for evaluating and comparing controller performance. In conclusion, this study aims to demonstrate the superiority of the GWO-based Type II FLC over the conventional PID method in enabling mobile robots to track dynamic and complex reference trajectories with greater accuracy and reliability. This article is arranged into five sections. The introduction defines the trajectory tracking problem, presents a literature review, and outlines the study's objectives. The second section details the kinematic model of the mobile robot and the control strategies employed. The third section explains the PID and GWO-based Type-II FLC controllers used in the study. In the fourth section, simulation results are presented, followed by a comparative analysis. The final section provides a general evaluation and discusses potential directions for future research.

2 MODELING OF THE MOBILE ROBOT

The mobile robots are examined in the current research based on various wheel types and designs. Two separate wheels on the right and left sides make up the robot used in this investigation. Usually, the angular velocities of the DC motors powering these wheels are controlled to move the system. Mobile robots can turn by varying the angular velocities of the two drive wheels or change orientation by traveling along a curved track. To implement high-performance controllers in robot control, modeling the robot is essential. In a mobile robot, the kinematic control approach can be applied provided that appropriate non-slip conditions are selected. The kinematics of a four-wheeled mobile robot are indicated in Figure 1.



(a)



b)

Figure 1. Schematic representation a) kinematics and b) dynamics of the mobile robot

Pursuant, the velocity references applied to the actuators on the wheels enable the robot to move with the corresponding linear and angular velocities. In the global coordinate system, the configuration vector of the vehicle is described as follows:

$$q = [X \ Y \ \theta]^T \in R^3 \quad (1)$$

where X , Y and θ stand for the vehicle's position and orientation, respectively. The transformation between the local velocities, defined in the local coordinate frames attached to the vehicle's center of mass (COM), and the generalized velocities are given as follows:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ w \end{bmatrix} \quad (2)$$

The center of mass (COM) is located at a distance $d \in (0, b)$ from the Center of Geometry (COG). As illustrated in Figure 1, the Instantaneous Center of Rotation (ICR) lies on the axis intersecting the COG. Under these conditions, the nonholonomic operational constraint that limits lateral slippage can be described as follows [1].

$$v_y - dw = 0 \quad (3)$$

$$[-\sin\theta \ \cos\theta \ -d] = [\dot{X} \ \dot{Y} \ \dot{\theta}]^T = A(q)\dot{q} = 0 \quad (4)$$

Equation (2) can then be reformulated as follows; where $S(q) \in R^{3 \times 2}$ is a matrix, and $\eta \in R^2$ is defined as the kinematic-level control input vector, expressed as:

$$\dot{q} = S(q)\eta \quad (5)$$

$$S(q) = \begin{bmatrix} \cos\theta & -d\sin\theta \\ \sin\theta & d\cos\theta \\ 0 & 1 \end{bmatrix}, \eta = [v_x \ w]^T \quad (6)$$

Since the columns of $S(q)$ always lie in the null space of $A(q)$, the following condition is satisfied:

$$S^T A^T = 0 \quad (7)$$

The dynamics of the mobile robot platform are depicted in Figure 1. Initially, the forces acting on the left and right sides are introduced, followed by a description of the wheel dynamics.

$$\frac{F_L}{2} = F_{fl} = F_{rl} \quad (8)$$

$$\frac{F_R}{2} = F_{fr} = F_{rr} \quad (9)$$

$$I_w \dot{w} = \tau - DF \quad (10)$$

where, I_w denotes the wheel inertia, $w_w = [w_{fl} \ w_{rl} \ w_{fr} \ w_{rr}]^T \in R^4$ represents the vector of wheel angular velocities, and $\tau = [\tau_{fl} \ \tau_{rl} \ \tau_{fr} \ \tau_{rr}]^T \in R^4$ is the vector of wheel torques. The force vector is given by $F = [F_L \ F_R]^T \in R^2$, and $D \in R^{4 \times 2}$ denotes the force-to-torque transformation matrix, defined as follows:

$$D = \frac{r}{2} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}^T \quad (11)$$

where r denotes the wheel radius. The equations governing the robot's motion in the global coordinate frame are then expressed as follows:

$$(F_L + F_R)\cos\theta - (f_x\cos\theta - f_y\sin\theta) = m\ddot{X} \quad (12)$$

$$(F_L + F_R)\sin\theta - (f_x\sin\theta + f_y\cos\theta) = m\ddot{Y} \quad (13)$$

$$(F_R - F_L)c - M_r = I_w\ddot{\theta} \quad (14)$$

Here, m represents the mass of the vehicle, I denotes the moment of inertia about the z -axis, f_x and f_y correspond to the rolling and lateral (slip) friction forces, respectively, and M_r is the resistive torque around the z -axis. Under ideal conditions, rolling friction forces are significantly smaller than slip friction forces. A more realistic model for rolling friction in the presence of slip can be found, for instance, in [30–31]. For the front-left wheel only, the friction forces are defined as follows:

$$f_{flx} = \mu_x N_{fl} \text{sign}(v_{flx}) \quad (15)$$

$$f_{fly} = \mu_y N_{fl} \text{sign}(v_{fly}) \quad (16)$$

The following formula is used to determine the normal forces that gravity exerts on the wheel-ground contact points:

$$N_{fl} = N_{fr} = \frac{b}{a+b} * \frac{mg}{2} \quad (17)$$

$$N_{rl} = N_{rr} = \frac{a}{a+b} * \frac{mg}{2} \quad (18)$$

finally, the resistance moment is computed as follows.

$$a[f_{fly} + f_{fry}] - b[f_{rly} + f_{rry}] + c[f_{frx} + f_{rrx} - f_{flx} - f_{rlx}] = M_r \quad (19)$$

The general form of the vehicle dynamics incorporating nonholonomic constraints utilizing the Euler-Lagrange principle and introducing an additional vector to represent disturbances is expressed as follows:

$$M(q)\ddot{q} + R(\dot{q}) + F_d = B(q)F + A^T\lambda \quad (20)$$

$M \in R^{3 \times 3}$ denotes the mass and inertia matrix, $R \in R^3$ represents the vector of resistance forces and torques, $F_d \in R^3$ is the disturbance vector, $B \in R^{3 \times 2}$ is the input matrix, F is the control input at the dynamic level as previously described, A is the constraint vector as given in Equation (4), and λ is the vector of Lagrange multipliers.

$$M(q) = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & I \end{bmatrix}, B(q) = \begin{bmatrix} \cos\theta & \cos\theta \\ \sin\theta & \sin\theta \\ -c & c \end{bmatrix}, R(\dot{q}) = \begin{bmatrix} f_x \cos\theta - f_y \sin\theta \\ f_x \sin\theta + f_y \cos\theta \\ M_r \end{bmatrix} \quad (21)$$

When the time derivative of Equality (5) is taken

$$\ddot{q} = S(q)\dot{\eta} + \dot{S}(q)\eta \quad (22)$$

and Equality (5), Equality (7), and Equality (22) relationships are used to derive the dynamic system in Equality (20).

$$\overline{M}\dot{\eta} + \overline{C}\eta + \overline{R} + \overline{F}_d = \overline{B}F \quad (23)$$

$$\overline{M} = S^T M S, \overline{C} = S^T M \dot{S}, \overline{R} = S^T R, \quad \overline{F}_d = S^T F_d, \overline{B} = S^T B \quad (24)$$

3 CONTROLLER DESIGNS OF THE SYSTEM

Reducing the robot's trajectory tracking error is the primary goal of mobile robot control. In this context, PID and Type II Fuzzy Logic Control (Type II FLC) methods have been suggested and applied for mobile robot control. The membership function (MF) boundary values of the Type II FLC method were calculated utilizing the Grey Wolf Optimization (GWO) algorithm. The first method suggested for the mobile robot's trajectory tracking control is the Proportional-Integral-Derivative (PID) control method. Despite being a classical method, the PID control method still demonstrates good performance compared to many other methods. Its mathematical equation is illustrated in Equation 25.

$$u(t) = K_P e(t) + K_I \int_0^t e(t) dt + K_D \frac{d}{dt} e(t) \quad (25)$$

The Ziegler-Nichols method was employed to determine the control coefficients, and a closed-loop control type was utilized within this method [32]. The terms given in the equation are referred to as the control signal $u(t)$, proportional gain K_P , integral gain K_I , derivative gain K_D , and error signal $e(t)$, respectively. The fuzzy logic algorithm is a method suggested by Zadeh and composed of five stages [33-34]. In the first stage, the input variables are transformed into a fuzzy set through the process of fuzzification. In the second stage, rule tables, MFs, and a rule base are formed. This rule base comprises a group of IF-THEN rules obtained from the verbal expressions of experts knowledgeable about the system. The third stage, the inference mechanism executes the inference process on the system using these rules and produces fuzzy outputs. The fourth stage defines the MFs and definition sets of fuzzy sets in the database. In the final stage, defuzzification, a fuzzy set is converted into a clear value for the output. Type II Fuzzy Logic is an approach that has emerged to minimize the effects of uncertainty in Type I Fuzzy Logic Control systems [35-38]. With Type II Fuzzy Logic, uncertainties can be modeled because their MFs are fuzzy. The MFs created in this method are three-dimensional. The uncertainty of the words obtained in fuzzy rules, information obtained from experts with differing opinions, noisy measurements that operate the system, and noisy data that adjust parameters, etc., are all represented in the third dimension of Type II Fuzzy Logic sets [39]. The structure of the Type II Fuzzy Logic system is illustrated in Figure 2.

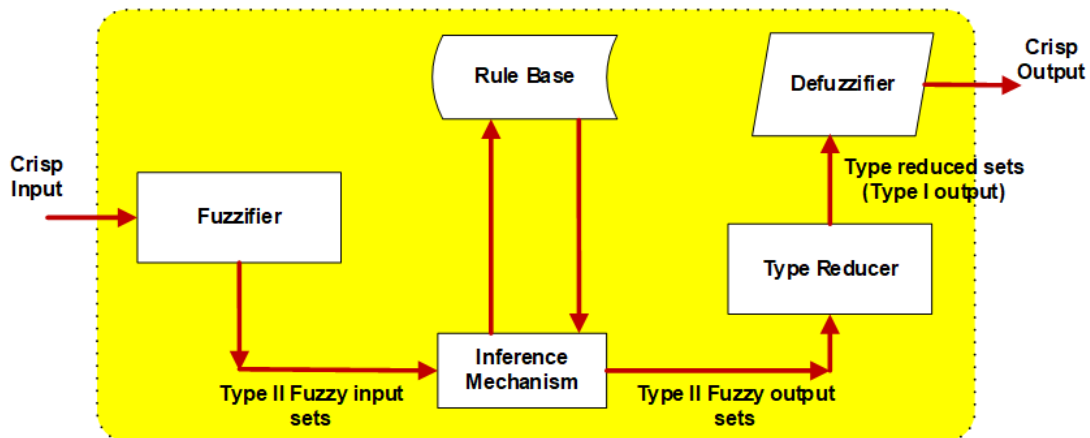


Figure 2. Structure of Type-II Fuzzy Logic System

The fuzzyfier transforms the sharp input into a fuzzy set, as illustrated in Figure 2. It resembles the rule base and inference mechanism structure. Another difference from Type I is the presence of a type reducer. Therefore, the structure of the rules is used in the same way as Type I, the only difference being that the MFs in Type II are Type II intervals. The Type II Fuzzy Logic Control (Type II FLC) algorithm was used in this study. The controller utilizes the error

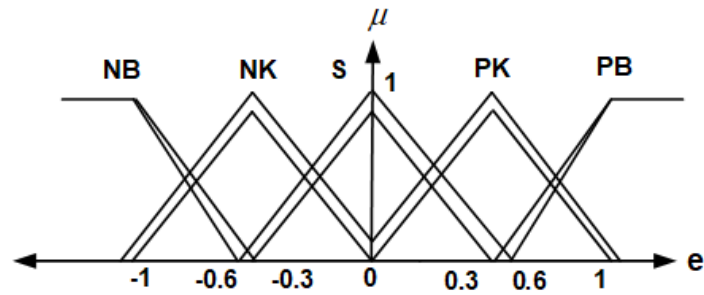
(e) and the rate of change of the error ($\dot{e}$) as input values. Triangular MFs were used in the Type II FLC method. The boundary values of the MFs were computed using the GWO algorithm, one of the swarm-based optimization algorithms. The GWO algorithm is preferred because it provides a global optimum solution compared to deterministic methods, thanks to its speed, simplicity, and swarm-based meta-heuristic optimization methods. The GWO algorithm was first suggested by Mirjalili and colleagues [40]. Due to its simple computation, the GWO algorithm is fast and provides a global optimal solution, demonstrating superior performance compared to other optimization techniques in solving many problems found in the literature [41-44]. The fundamental philosophy of this algorithm is based on mimicking the social behavior and hunting strategies of grey wolves living in nature. To minimize error values Mean Squared Error (MSE) was utilized as the objective function. The Mamdani method and triangular MFs were utilized. The steps of the GWO algorithm implemented to the Type II FLC control method are provided below.

In the first step, the Type-II FLC model is defined. In this stage, the input and output variables of the system to be controlled are determined, and the general structure of the FLC is established. In Type-II systems, it should be noted that, unlike classical (Type-I) systems, MFs include an uncertainty range (Footprint of Uncertainty - FOU) to model uncertainties. In the second step, the Type-II MFs to be used in the system are defined. These functions represent linguistic terms like “low,” “medium,” and “high,” and both upper and lower MFs are specified for each MF. This increases the system's sensitivity to uncertainty. The parameters of these MFs (e.g., center, width, FOU limits) will be optimized later. In the third step, the GWO algorithm is initiated. Each wolf in the GWO algorithm is a combination of parameters from the MF. Initially, the wolf pack is created with randomly selected solution candidates. Each wolf represents a parameter set of a Type-II fuzzy system. In the fourth step, the fitness function that the algorithm will use is defined. This function measures the performance of the fuzzy controller. For example, the sum of the squares of the errors of the controlled system, the MSE over time, or another performance metric can be selected as the fitness function. In the fifth step, this fitness function is calculated for each wolf. The parameters represented by each wolf are implemented to the Type-II FLC, and the system is run. The performance of each solution is measured based on the simulation or real system output, and a fitness value is assigned. In the sixth step, the wolves are ranked according to their fitness values, and the top three wolves are identified. The best wolf is called Alpha, the second best is called Beta, and the third best is called Delta. These three wolves are in a leadership position, determining the direction of the

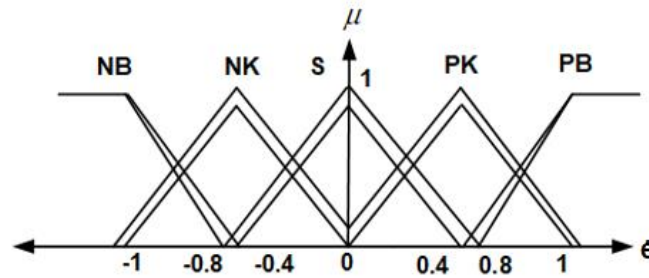
other wolves. In the seventh step, the positions of the wolves are updated according to the swarm behavior model of the GWO algorithm. The wolves change their parameters according to the positions of the Alpha, Beta, and Delta wolves and move toward better points in the solution space. This ensures the optimization of the system. In the eighth step, it is checked whether the specified maximum number of iterations has been reached. If the iteration is not complete, the algorithm returns to the fifth step and evaluates the new positions of the wolves. The quality of the solution improves with each iteration. In the ninth step, the best solution obtained at the end of the iterations is determined. The parameters represented by the Alpha wolf form the optimized Type-II FLC MFs, and these parameters are selected for use in the system. In the tenth step, the final Type-II FLC controller model is created using the best MF parameters obtained. This model is integrated into the control system and applied in real time or in a simulation environment. The control system's performance is assessed in the last phase. The wolf population for the GWO algorithm was set at 40, and the maximum number of iterations was set at 100. Optimal values were observed to be obtained. The population size and maximum iteration count were determined by reviewing similar optimization-based control studies in the literature and through preliminary tests (trial and error). These values provide an appropriate balance between solution quality and computation time. The controller utilizes the error (e) and the rate of change of the error ($\dot{e}$) as input values. The MFs and rule table created for the suggested method are given below. Figure 4(a), Figure 3(b), and Figure 3(c) show the MFs and values described for the system's input values e , $\dot{e}$, and τ output values. The rule base composed for the control of the system is given in Table 1. The rule base composed for the control of this system consists of 25 rules, as shown in Table 1 below.

Table 1. The rule base created for Type II FLC

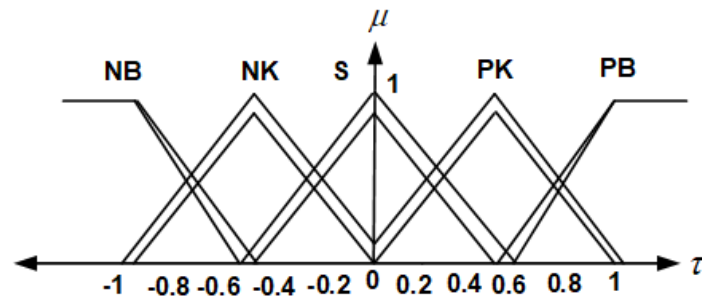
		$\dot{e}$					
		τ	NB	NK	S	PK	PB
e	NB	NB	NB	NK	NK	S	
	NK	NB	NK	NK	S	PK	
	S	NB	NK	S	PK	PB	
	PK	NK	S	PK	PB	PB	
	PB	S	PK	PB	PB	PB	



(a)



(b)



(c)

Figure 3. Membership functions of the Type II FLC method defined by input and output values a) e b) $\dot{e}$ and c) τ

In control algorithms, the system inputs are the error e and the change in values of the error $\dot{e}$. Here, the error is the difference between the system outputs and the desired values. The system outputs are the appropriate torque values implemented to the motors that drive the system in such a way as to minimize the error and the instantaneous change values of the error over time. The rule table composed for the Type II FLC method, given in Table 1, is provided. In the table, the fuzzy control variables e , $\dot{e}$, and τ represent the error, error change, and torque expression, respectively. The terms NB, NK, S, PK, and PB in the table represent Negative Large, Negative Small, Zero, Positive Small, and Positive Large, respectively.

4 SIMULATION RESULTS AND DISCUSSION

The numerical and graphical results of applying the planned control methods for the trajectory control of the mobile robot are shown in this section. The GWO-based Type II FLC and PID control approaches were suggested to achieve trajectory tracking of the mobile robot. The primary objective of the trajectory control is to complete the desired path with minimal tracking error. The model parameters of the mobile robot are $m=1.2 \text{ kg}$; $J=0,5 \text{ kg-m}^2$; $a=0,2 \text{ m}$; $L=0,25 \text{ m}$; $r=0,05 \text{ m}$. In the simulation studies of the mobile robot, the operation time was set to 60 seconds. The tracking performance was examined for three different trajectory scenarios. The initial system input parameters were set as $x = 0 \text{ m}$ and $y = 0 \text{ m}$. $K_P = 30$, $K_I = 10$, and $K_D = 4$ was taken. In tests conducted in a simulation environment, the average optimization time for the GWO algorithm was measured to be approximately 30 seconds per trial. All work was performed on a computer equipped with an Intel i7 processor and 16 GB of RAM. Figure 4 presents the position trajectories obtained using the PID and GWO-based Type II FLC methods in the simulation environment. Figure 5(a), 5(b), and 5(c) illustrate the time-dependent changes of the X and Y positions, as well as the orientation angle, obtained with the suggested control methods.

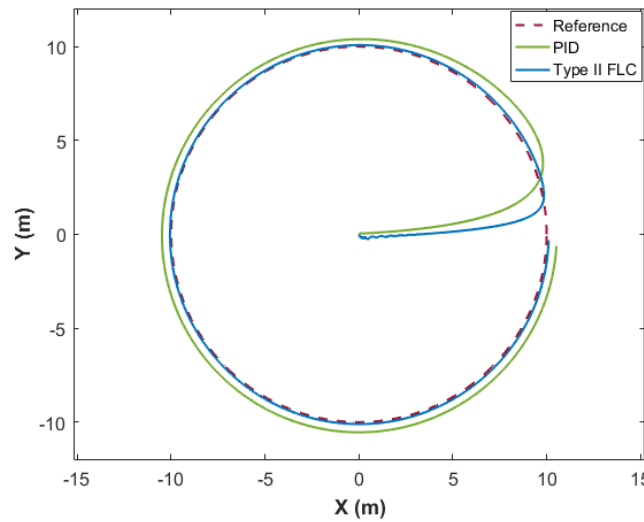
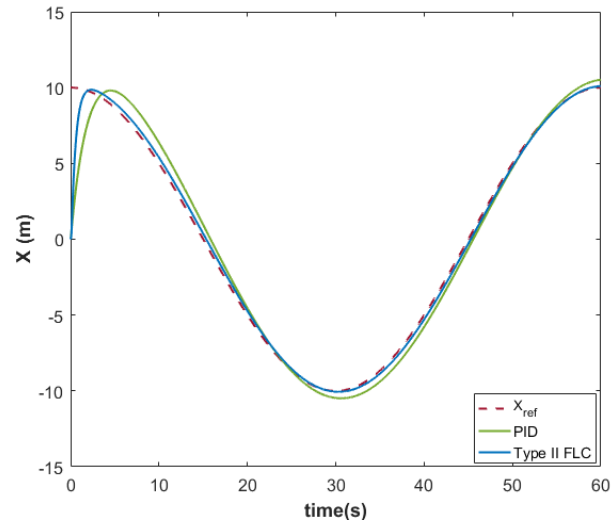
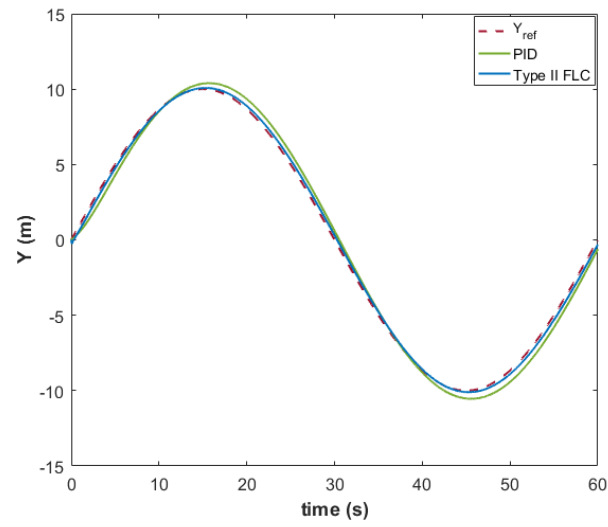


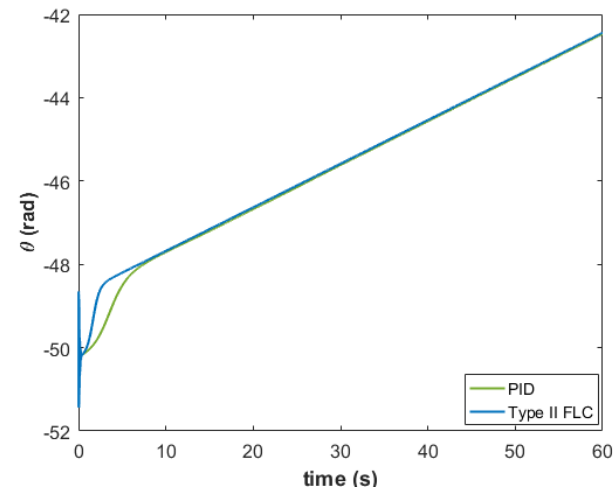
Figure 4. Position trajectory obtained using the suggested control method



(a)



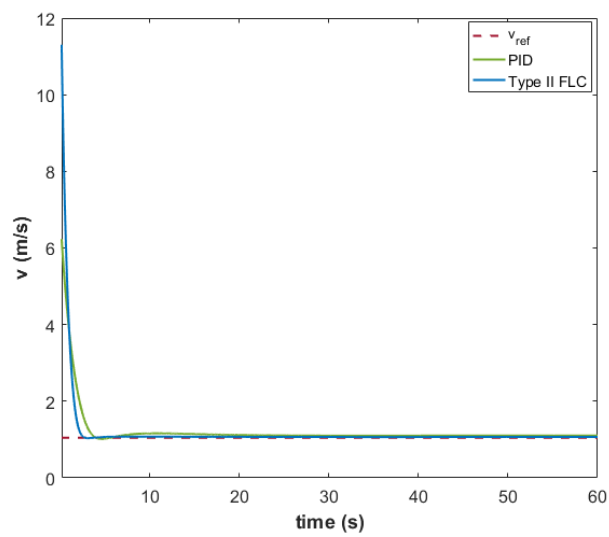
(b)



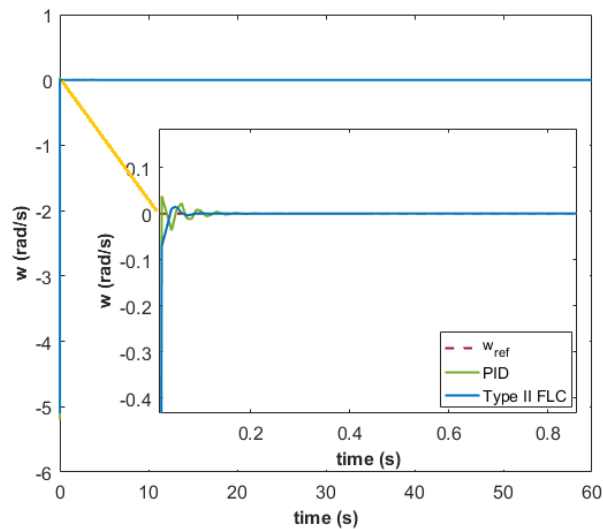
(c)

Figure 5. Graphs of the time-dependent variations of a) X position, b) Y position, and c) theta angle obtained using the suggested control methods

In the simulation, the reference trajectory was initially set as a circle (Figure 4). It can be observed from this figure that the mobile robot was able to complete the trajectory tracking with high motion accuracy using the Type II FLC method. In contrast, the PID control method struggled with trajectory tracking at the beginning and subsequently followed the trajectory with a certain error margin (Figure 5). Due to the optimization component, the Type II FLC method demonstrated less deviation and tracked the reference trajectory more accurately compared to the PID method. In terms of orientation angle control, both controllers give similar results. Figure 6(a) and 6(b) present the linear and angular velocity graphs obtained as a result of applying the suggested control methods.



(a)



(b)

Figure 6. Graphs of a) linear velocity and b) angular velocity obtained using the suggested control methods

Examining Figure 6(a) and 6(b), which present the linear and angular velocities, it is observed that the robot's velocity signals exhibit smooth variations necessary for executing maneuvers. During the initial seconds of the simulation, higher errors and more fluctuating velocities were noted. These values are attributed to the controller's initialization phase, as errors tend to rise at the start of the simulation. As seen in the graphs, once the robot passes the startup phase, the control approach continues to follow the trajectory with consistent signals. Figure 7 shows the position graph obtained by applying the PID and GWO-based Type II FLC methods in the simulation environment.

Furthermore, time-dependent changes in the X and Y positions as well as the orientation angle acquired using the suggested approaches are shown in Figure 8(a), 8(b), and 8(c).

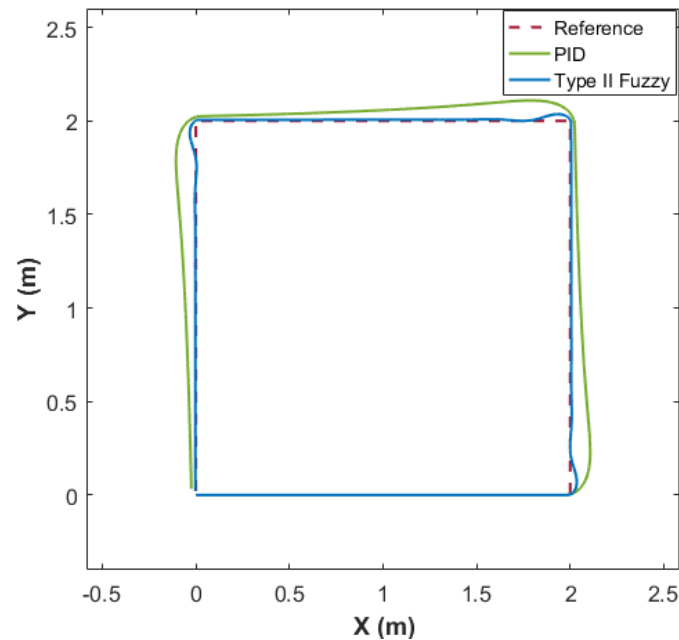


Figure 7. Position graph obtained using the suggested control methods

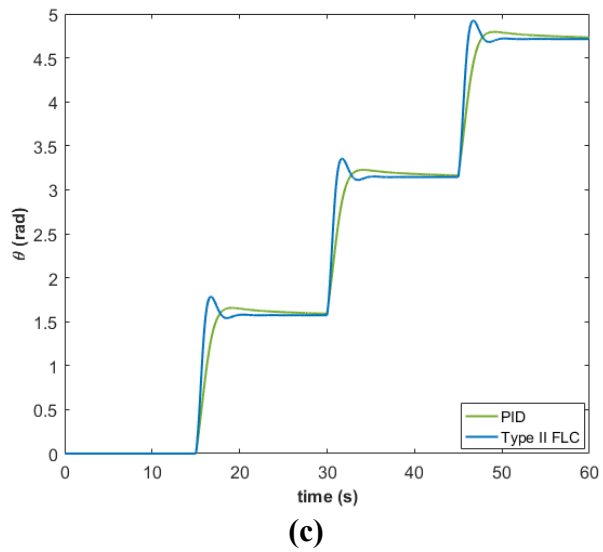
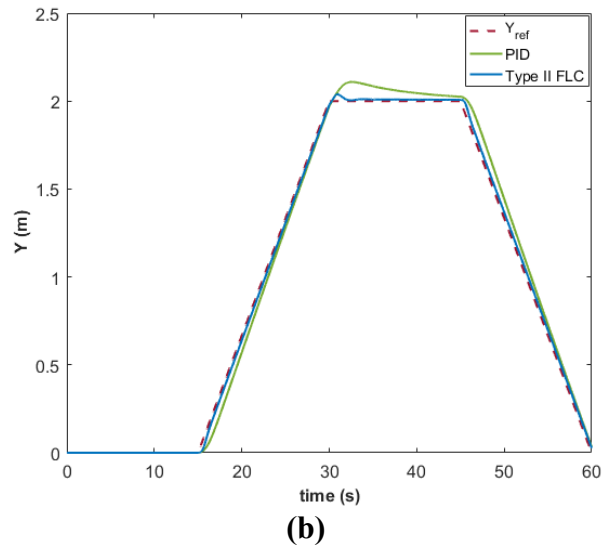
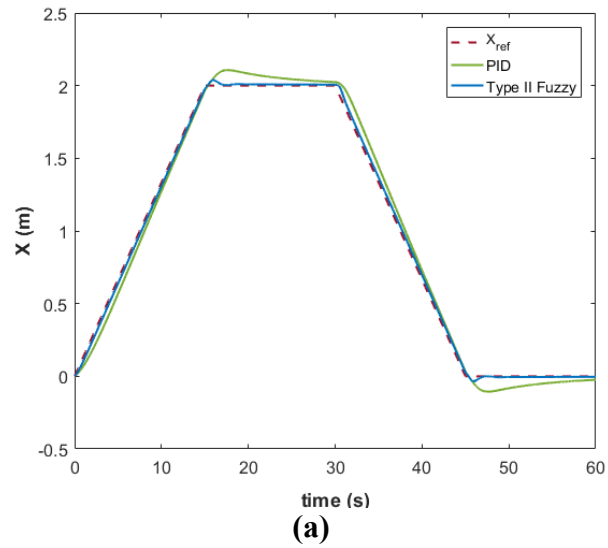


Figure 8. Graphs of the time-dependent variations of a) X position, b) Y position, and c) theta angle obtained using the suggested control methods

The second reference trajectory was set as a square (Figure 7). An examination of Figure 8 reveals that corner transitions in the square reference pose challenges for the PID control method, with abrupt changes in direction leading to increased tracking errors. In contrast, the Type II FLC method provides smoother transitions during sudden changes, resulting in lower tracking errors. In terms of orientation angle control, both controllers give similar results. Figure 9(a) and 9(b) present the linear and angular velocity graphs obtained from the application of the suggested methods.

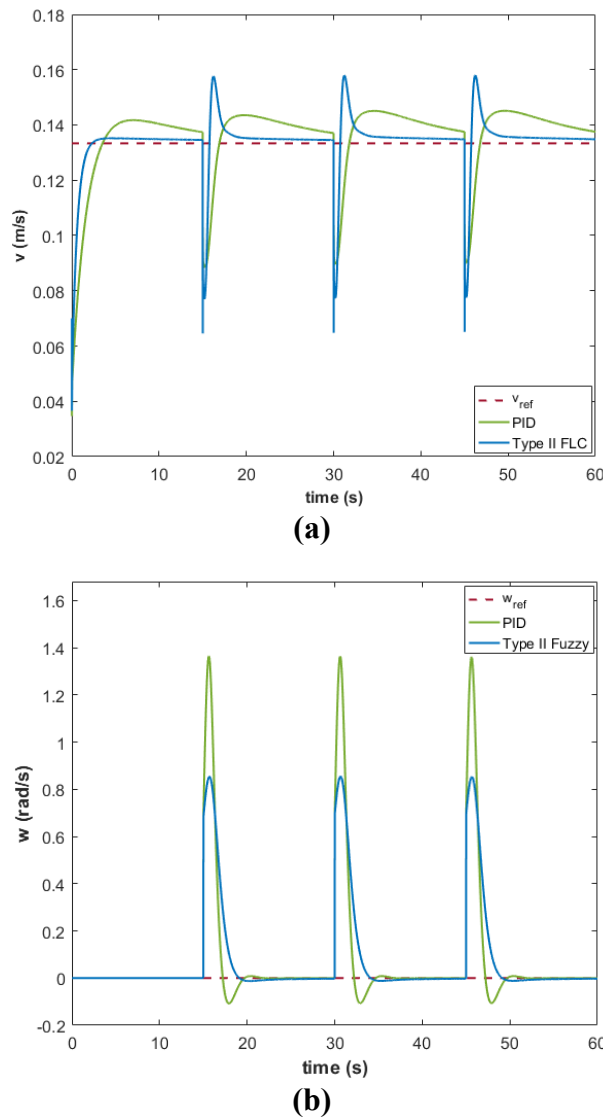


Figure 9. Linear and angular velocity graphs obtained using the suggested control methods: a) linear velocity, and b) angular velocity

Analysis of Figure 9(a) and 9(b) reveals that the robot's velocity signals exhibit abrupt changes to perform maneuvers, with the PID control method notably struggling in these instances. This behavior is inherent to the control approach, as the fixed control gains in PID

make it prone to higher error rates during sudden directional changes. As illustrated in the graphs, the Type II FLC method follows the path with lower velocity errors compared to the PID controller. Figure 10 presents the position graph obtained from simulations using both the PID and the GWO-based Type II FLC methods. Additionally, Figure 11(a), 11(b), and 11(c) show the time-dependent changes in the X and Y positions and the orientation angle, respectively, derived from the application of the suggested control methods.

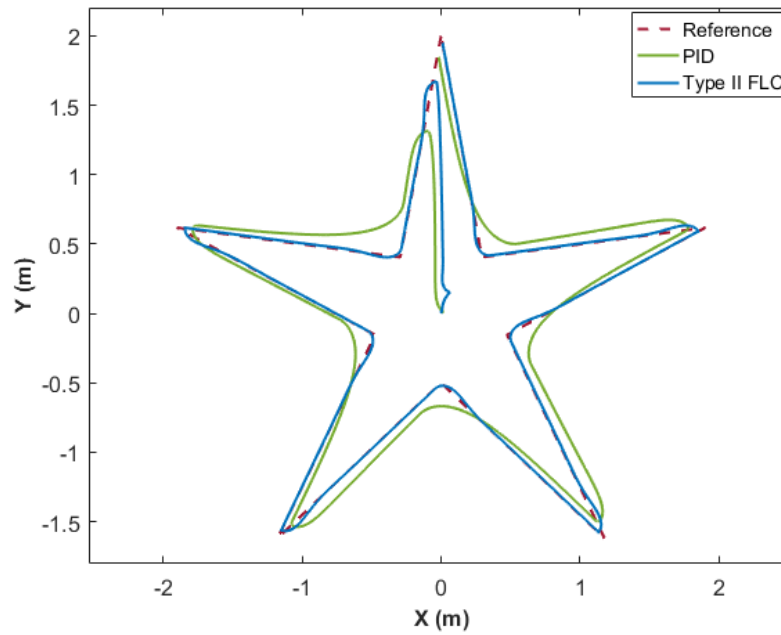
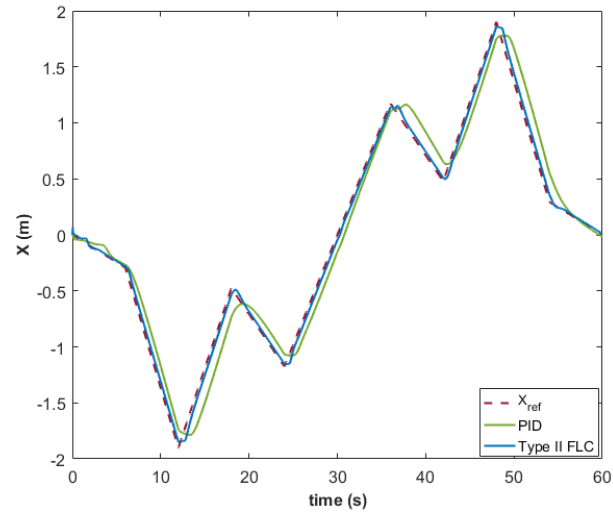
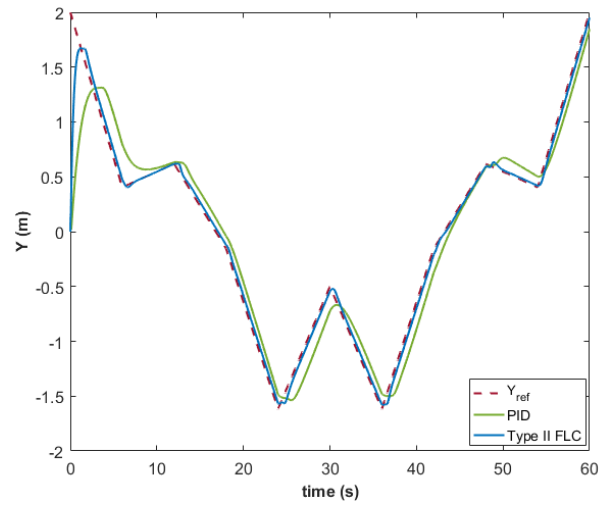


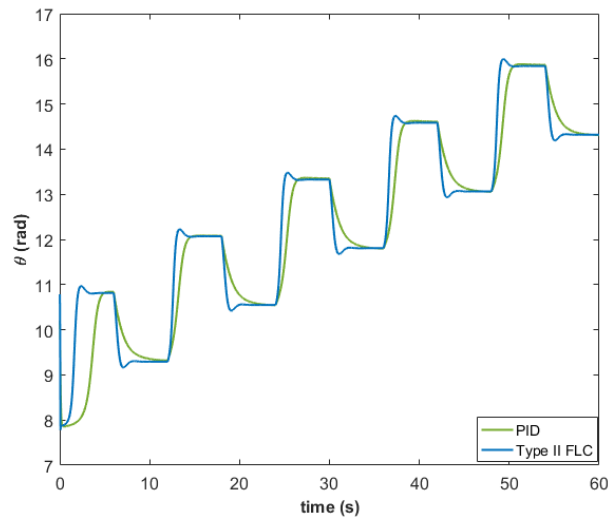
Figure 10. Position graph obtained using the suggested control methods



(a)



(b)



(c)

Figure 11. Graphs of the time-dependent variations of a) X position, b) Y position, and c) theta angle obtained using the suggested control methods

The third reference trajectory used is a star shape (Figure 10). As observed in Figure 11, due to the sharp turns and directional changes in the star reference input, the capabilities of the controllers are more challenged. The PID control method proves to be demanding, with sudden directional changes increasing the tracking error. Due to the sharp turns at the star reference point, the Type II FLC provided better results in terms of orientation angle control, as it placed greater demands on the controllers' capabilities. In contrast, the Type II FLC method provides smoother transitions during abrupt changes, resulting in lower tracking errors. This is attributed to the Type II FLC controller's superior ability to manage uncertainties. Figure 12(a) and 12(b) present the linear and angular velocity graphs obtained as a result of applying the suggested methods.

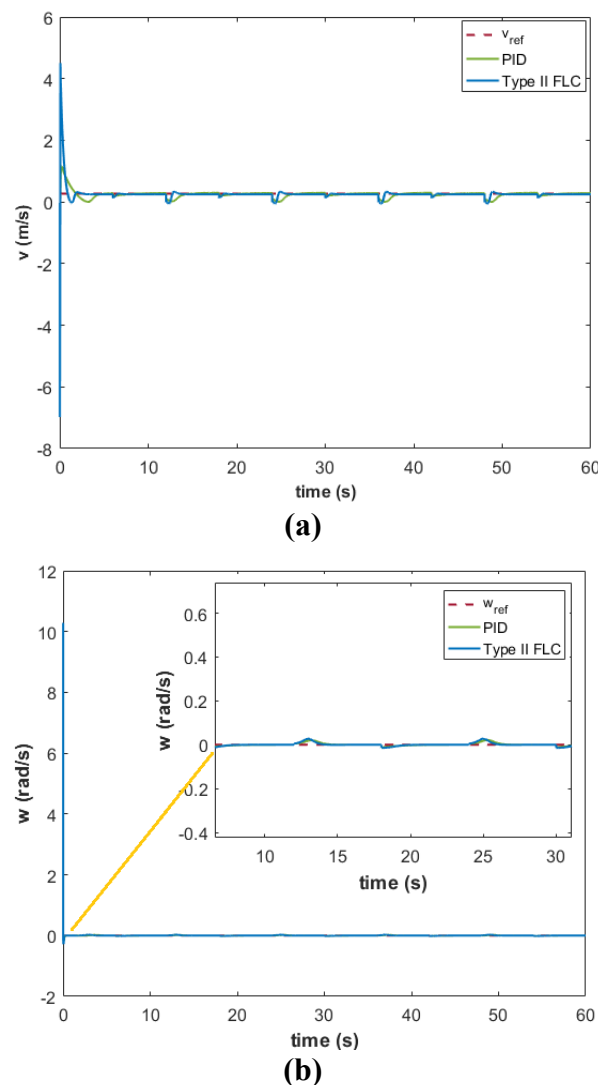


Figure 12. Linear and angular velocity graphs obtained using the suggested control methods: a) linear velocity, and b) angular velocity

Examining Figure 12, which presents the linear and angular velocities, it is observed that the robot's velocity signals exhibit variations during maneuvers similar to previous scenarios, with the PID control method showing particular difficulties. As indicated by the graphs, the Type II FLC method follows the path with lower error values compared to the PID controller. The simulation results demonstrate that, especially in complex star-shaped trajectories, the Type II FLC controller produces lower tracking errors and smoother control signals than the PID controller. In square-shaped reference paths, noticeable overshoots were observed in the PID controller around corner regions, whereas the Type II FLC controller executed these transitions more smoothly and stably. For circular trajectories, both controllers performed similarly; however, the Type II FLC controller achieved a lower Root Mean Square Error (RMSE). These findings suggest that Type II FLC is particularly effective in tasks involving sharp and uncertain transitions. In this study, the goal is for both control methods to complete the suggested trajectories with minimal error across all three different reference scenarios. To evaluate the performance of the control methods, a comparison based on the RMSE performance index (Equation 26) is presented in Table 2.

$$RMSE = \int_0^t tedt \quad (26)$$

Table 2. The position comparisons of controllers

Performance Criteria	Circle		Square		Star	
RMSE	X (m)	Y (m)	X (m)	Y (m)	X (m)	Y (m)
PID	1.285	1.584	1.174	1.164	1.169	1.27
Type II FLC	0.087	0.099	0.024	0.034	0.051	0.058
Improved (%)	93	94	98	97	96	95

Upon examining Table 2, which presents the comparison of RMSE performance criteria, the lowest error values are highlighted in bold. In Table 2, the X and Y position error results obtained using the Type II FLC method for the Circle trajectory are 0.087 m and 0.099 m, respectively, whereas the PID control method yielded position errors of 1.285 m and 1.584 m. For the Square trajectory, the X and Y position errors using the Type II FLC method are 0.024 m and 0.034 m, respectively, compared to the PID control errors of 1.174 m and 1.164 m. Finally, for the Star trajectory, the Type II FLC method achieved X and Y position errors of 0.051 m and 0.058 m, respectively, while the PID control method resulted in errors of 1.169 m and 1.27 m. The percentage improvement values presented in Table 2 indicate that the greatest relative enhancement of the Type II FLC method over the PID controller occurred in both X

and Y positions for the Square trajectory. The smallest improvements in X and Y positions were observed in the Circle trajectory. Overall, the RMSE error metrics and percentage improvements demonstrate that the suggested Type II FLC method achieves significantly lower errors, enabling more effective trajectory tracking and outperforming the PID control approach. The controllers developed in our study (particularly FLC and PID) are primarily designed based on kinematic error signals (position and orientation difference). However, these control signals are then effectively applied through the robot's dynamic model. In particular, dynamic effects such as friction, inertia, and mass were modeled in the simulation environment, and the controller was tested to ensure it operates stably and successfully under such effects. The dynamic modeling process was carried out to ensure a realistic simulation of the system.

5 CONCLUSIONS

This study addresses the trajectory tracking problem of wheeled mobile robots by proposing and comparatively evaluating two different control strategies through simulations. Initially, the mathematical model of the mobile robot was developed, followed by the design of a conventional PID controller and a GWO-based Type-II FLC. The suggested control methods were tested on three distinct reference trajectories defined as circular, square, and star-shaped paths. Simulation results, analyzed through numerical and graphical data, indicated that the PID controller caused deviations, especially during sharp turns and maneuvers in the square and star trajectories—resulting in increased tracking errors. In contrast, the GWO-based Type-II FLC demonstrated smoother and more stable maneuvers, achieving lower tracking errors and higher trajectory accuracy. Specifically, in a square orbit, the X and Y position errors have been improved by 98% and 97%, respectively, according to the Type II FLC, PID control method. These findings suggest that the GWO-based FLC provides more reliable and precise control for mobile robots operating in dynamic environments, in contrast to the traditional PID controller. It has been concluded that the inability of the PID controller to provide the desired performance in complex systems and its failure to respond to sudden changes in the robot's direction stem from the non-adaptive nature of the method, due to the limited ability of the PID controller's inherent structure to model uncertainties and parameter changes in the system. Superior results were obtained due to the global optimization capability of the GWO algorithm and the fuzzy logic system's ability to handle uncertainties. Although Type-II FLC offers the advantage of better modeling uncertainties, it has a higher computational cost compared to Type-I. However, the structure used in the study can operate with low latency thanks to an optimized type-

reduction algorithm (e.g., Karnik-Mendel) and provides cycle times at the millisecond level on modern processors. Accordingly, the controller has been assessed as being real-time feasible. Future studies aim to test these methods on a physical prototype that will be developed for implementation in an experimental environment. Comparative analyses with methods such as SMC, LQR, MPC, and adaptive control are also planned.

Acknowledgements

This work was supported by Muş Alparslan University Scientific Research Coordination Unit Project Number, as a project numbered BAP-24-MMF-4901- 03.

Conflict of Interest Statement

There is no conflict of interest between the authors.

Statement of Research and Publication Ethics

The study is complied with research and publication ethics.

Artificial Intelligence (AI) Contribution Statement

This manuscript was entirely written, edited, analyzed, and prepared without the assistance of any artificial intelligence (AI) tools. All content, including text, data analysis, and figures, was solely generated by the authors

Contributions of the Authors

Writing original draft, T.Abut.; Writing review & editing, T. Abut., E. Salkim.; Visualization, T. Abut. And E. Salkim. All authors have read and agreed to the published version of the manuscript.

REFERENCES

- [1] L. Caracciolo, A. De Luca, and S. Iannitti, "Trajectory tracking control of a four-wheel differentially driven mobile robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 4, Detroit, MI, USA, May 1999, pp. 2632–2638, doi: 10.1109/robot.1999.773994.
- [2] K. Kozłowski and D. Pazderski, "Modeling and control of a 4-wheel skid-steering mobile robot," *Int. J. Appl. Math. Comput. Sci.*, vol. 14, no. 4, pp. 477–496, 2004.
- [3] E. Maalouf, M. Saad, and H. Saliah, "A higher level path tracking controller for a four-wheel differentially steered mobile robot," *Robot. Auton. Syst.*, vol. 54, no. 1, pp. 23–33, 2006, doi: 10.1016/j.robot.2005.10.001.

- [4] U. Kumar and N. Sukavanam, "Backstepping based trajectory tracking control of a four wheeled mobile robot," *Int. J. Adv. Robot. Syst.*, vol. 5, no. 4, p. 38, 2008, doi: <https://doi.org/10.5772/6224>.
- [5] C. Cariou, R. Lenain, B. Thuilot, and M. Berducat, "Automatic guidance of a four-wheel-steering mobile robot for accurate field operations," *J. Field Robot.*, vol. 26, no. 6–7, pp. 504–518, 2009, doi: 10.1002/rob.20282.
- [6] M. H. Lee and T. H. S. Li, "Kinematics, dynamics and control design of 4WIS4WID mobile robots," *J. Eng.*, vol. 2015, no. 1, pp. 6–16, 2015, doi: 10.1049/joe.2014.0241.
- [7] N. B. Hoang and H. J. Kang, "Neural network-based adaptive tracking control of mobile robots in the presence of wheel slip and external disturbance force," *Neurocomputing*, vol. 188, pp. 12–22, 2016, doi: 10.1016/j.neucom.2015.02.101.
- [8] J. Liao, Z. Chen, and B. Yao, "Performance-oriented coordinated adaptive robust control for four-wheel independently driven skid steer mobile robot," *IEEE Access*, vol. 5, pp. 19048–19057, 2017, doi:10.1109/ACCESS.2017.2754647.
- [9] M. Deremetz et al., "Path tracking of a four-wheel steering mobile robot: A robust off-road parallel steering strategy," in *Proc. Eur. Conf. Mobile Robots (ECMR)*, Paris, France, Sep. 2017, pp. 1–7, doi: [10.1109/ECMR.2017.8098670](https://doi.org/10.1109/ECMR.2017.8098670).
- [10] J. Liao, Z. Chen, and B. Yao, "Model-based coordinated control of four-wheel independently driven skid steer mobile robot with wheel-ground interaction and wheel dynamics," *IEEE Trans. Ind. Inf.*, vol. 15, no. 3, pp. 1742–1752, 2018, doi: 10.1109/TII.2018.2869573.
- [11] Q. Qiu et al., "Extended Ackerman steering principle for the coordinated movement control of a four-wheel drive agricultural mobile robot," *Comput. Electron. Agric.*, vol. 152, pp. 40–50, 2018, doi: 10.1016/j.compag.2018.06.036.
- [12] I. Zeidis and K. Zimmermann, "Dynamics of a four-wheeled mobile robot with Mecanum wheels," *ZAMM J. Appl. Math. Mech.*, vol. 99, no. 12, pp. e201900173, 2019, doi: 10.1002/zamm.201900173.
- [13] T. Abut and M. Huseyinoğlu, "Modeling and optimal trajectory tracking control of wheeled a mobile robot," *Caucasian J. Sci.*, vol. 6, no. 2, pp. 137–146, 2019.
- [14] L. Jiang et al., "Anti-disturbance direct yaw moment control of a four-wheeled autonomous mobile robot," *IEEE Access*, vol. 8, pp. 174654–174666, 2020, doi: 10.1109/ACCESS.2020.3025575.
- [15] X. Zhang et al., "Motion planning and tracking control of a four-wheel independently driven steered mobile robot with multiple maneuvering modes," *Front. Mech. Eng.*, vol. 16, no. 3, pp. 504–527, 2021, doi: 10.1007/s11465-020-0626-y.
- [16] S. Raikwar, J. Fehrmann, and T. Herlitzius, "Navigation and control development for a four-wheel-steered mobile orchard robot using model-based design," *Comput. Electron. Agric.*, vol. 202, p. 107410, 2022, doi: 10.1016/j.compag.2022.107410.
- [17] M. Szeremeta and M. Szuster, "Neural tracking control of a four-wheeled mobile robot with mecanum wheels," *Appl. Sci.*, vol. 12, no. 11, p. 5322, 2022, doi: 10.3390/app12115322.
- [18] J. Qu et al., "Performance analysis and optimization for steering motion mode switching of an agricultural four-wheel-steering mobile robot," *Agronomy*, vol. 12, no. 11, p. 2655, 2022, doi: 10.3390/agronomy12112655.
- [19] B. Bae and D. H. Lee, "Design of a four-wheel steering mobile robot platform and adaptive steering control for manual operation," *Electronics*, vol. 12, no. 16, p. 3511, 2023, doi: 10.3390/electronics12163511.
- [20] J. Zhang et al., "Variable gain based composite trajectory tracking control for 4-wheel skid-steering mobile robots with unknown disturbances," *Control Eng. Pract.*, vol. 132, p. 105428, 2023, doi: 10.1016/j.conengprac.2022.105428.
- [21] S. Ryu, J. Won, and T. Seo, "Simulation study on four-wheeled mobile robot mechanisms using various performance criteria," *Robot. Auton. Syst.*, vol. 179, p. 104749, 2024, doi: 10.1016/j.robot.2024.104749.
- [22] L. Zhang et al., "An integral terminal sliding mode-based adaptive control approach for traversing unknown inclined surfaces," *Robot. Auton. Syst.*, vol. 186, p. 104928, 2025, doi: 10.1016/j.robot.2025.104928.
- [23] Y. H. Chen, "Nonlinear adaptive fuzzy hybrid sliding mode control design for trajectory tracking of autonomous mobile robots," *Mathematics*, vol. 13, no. 8, p. 1329, 2025, doi: 10.3390/math13081329.

- [24] M. Cipriano, G. Oriolo, and A. Cherubini, "Singularity-free trajectory tracking for steerable wheeled mobile robots," *IEEE Robot. Autom. Lett.*, 2025, doi: 10.1109/LRA.2025.3564209.
- [25] G. Shadrin et al., "Application of compensation algorithms to control the speed and course of a four-wheeled mobile robot," *Sensors*, vol. 24, no. 22, p. 7233, 2024, doi: 10.3390/s24227233.
- [26] L. Bruzzone, S. E. Nodehi, and P. Fanghella, "WheTLHLoc 4W: Small-scale inspection ground mobile robot with two tracks, two rotating legs, and four wheels," *J. Field Robot.*, vol. 41, no. 4, pp. 1146–1166, 2024, doi: 10.1002/rob.22314.
- [27] S. Amerttet, G. Gebresenbet, and H. M. Alwan, "Optimizing the performance of a wheeled mobile robots for use in agriculture using a linear-quadratic regulator," *Robot. Auton. Syst.*, vol. 174, p. 104642, 2024, doi: 10.1016/j.robot.2024.104642.
- [28] T. Lan and G. Yang, "Four-wheeled mobile robot with flexible posture control," *J. Field Robot.*, vol. 42, no. 4, pp. 1287–1297, 2025, doi: 10.1002/rob.22450.
- [29] D. Seo and J. Kang, "Controller design for active four-wheel steering and four-wheel independent drive-based mobile robot: Enhancing cornering performance in negative phase," *Int. J. Control Autom. Syst.*, vol. 23, no. 1, pp. 235–248, 2025, doi: 10.1007/s12555-024-0631-8.
- [30] J. Yi, D. Song, J. Zhang, and Z. Goodwin, "Adaptive trajectory tracking control of skid-steered mobile robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, Rome, Italy, Apr. 2007, pp. 2605–2610, doi: 10.1109/ROBOT.2007.363858.
- [31] S. Arslan and H. Temeltaş, "Robust motion control of a four wheel drive skid-steered mobile robot," in *Proc. Int. Conf. Electr. Electron. Eng. (ELECO)*, Bursa, Turkey, Dec. 2011, pp. II-415.
- [32] C. C. Hang, K. J. Åström, and W. K. Ho, "Refinements of the Ziegler–Nichols tuning formula," *IEE Proc. D, Control Theory Appl.*, vol. 138, no. 2, pp. 111–118, 1991, doi: 10.1049/ip-d.1991.0015.
- [33] L. A. Zadeh, "Fuzzy logic," *Computer*, vol. 21, no. 4, pp. 83–93, 1988, doi: 10.1109/2.53.
- [34] L. A. Zadeh, G. J. Klir, and B. Yuan, *Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems*. Singapore: World Scientific, 1996.
- [35] L. A. Zadeh, "The role of fuzzy logic in the management of uncertainty in expert systems," *Fuzzy Sets Syst.*, vol. 11, no. 1–3, pp. 199–227, 1983, doi: 10.1016/S0165-0114(83)80081-5.
- [36] N. N. Karnik, J. M. Mendel, and Q. Liang, "Type-2 fuzzy logic systems," *IEEE Trans. Fuzzy Syst.*, vol. 7, no. 6, pp. 643–658, 1999, doi: 10.1109/91.811231.
- [37] J. M. Mendel, R. I. John, and F. Liu, "Interval type-2 fuzzy logic systems made simple," *IEEE Trans. Fuzzy Syst.*, vol. 14, no. 6, pp. 808–821, 2006, doi: 10.1109/TFUZZ.2006.879986.
- [38] T. Abut, E. Salkim, and H. Tugal, "Active suspension control for improved ride comfort and vehicle performance using HHO-based type-I and type-II fuzzy logic," *Biomimetics*, vol. 10, no. 10, p. 673, 2025, doi: 10.3390/biomimetics10100673.
- [39] J. R. Castro, O. Castillo, and P. Melin, "An interval type-2 fuzzy logic toolbox for control applications," in *Proc. IEEE Int. Fuzzy Syst. Conf.*, London, U.K., Jul. 2007, pp. 1–6, doi: 10.1109/FUZZY.2007.4295341.
- [40] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.
- [41] T. Abut, E. Salkim, and A. Demosthenous, "Performance improvement in a vehicle suspension system with FLQG and LQG control methods," *Actuators*, vol. 14, no. 3, 2025, doi: 10.3390/act14030137.
- [42] H. Faris, I. Aljarah, M. A. Al-Betar, and S. Mirjalili, "Grey wolf optimizer: A review of recent variants and applications," *Neural Comput. Appl.*, vol. 30, pp. 413–435, 2018, doi: 10.1007/s00521-017-3272-5.
- [43] T. Abut, "Optimal LQR controller methods for double inverted pendulum system on a cart," *Dicle Univ. J. Eng.*, vol. 14, no. 2, pp. 247–255, 2023, doi: 10.24012/dumf.1253331.
- [44] T. Abut, "An inverted pendulum system control with fuzzy linear quadratic regulator method: Experimental validation," *Comput., Mater. Continua*, vol. 85, no. 2, pp. 4023–4042, 2025, doi: 10.32604/cmc.2025.066920.