Research Article

IMPLEMENTATION AND CONTROL SYSTEM DEVELOPMENT OF A DIFFERENTIAL DRIVE WHEELED MOBILE ROBOT

Tri-Vien VU[1,*], *Thanh-Quang NGUYEN*[2], *Anh-Minh Tran DUC*[2], *Phuong-Uyen Le NGUYEN*[3], *Chan-Kiet TRINH*[3]

[1]Modeling Evolutionary Algorithms Simulation and Artificial Intelligence, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, 19 Nguyen Huu Tho Rd, Tan Phong, District 7, Ho Chi Minh City, Vietnam
[2]Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, 19 Nguyen Huu Tho Rd, Tan Phong, District 7, Ho Chi Minh City, Vietnam
[3]Undergraduate Student from Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, 19 Nguyen Huu Tho Rd, Tan Phong, District 7, Ho Chi Minh City, Vietnam

vutrivien@tdtu.edu.vn, nguyenthanhquang@tdtu.edu.vn, tranducanhminh@tdtu.edu.vn, 42101058@student.tdtu.edu.vn, nhoxkiet14@gmail.com

[*]Corresponding author: Tri-Vien Vu; vutrivien@tdtu.edu.vn

DOI: 10.15598/aeee.v23i3.241213

Article history: Received Dec 27, 2024; Revised Jan 23, 2025; Accepted Feb 06, 2025; First Published March, 2025.

Abstract. *This study presents a cost-effective differential wheeled mobile robot system designed for educational and research applications. The robot employs an ESP32 microcontroller for control and communication, integrating sensors such as ultrasonic modules, an IMU 9050, and a line-following module via the I2C protocol. PID controllers were designed and fine-tuned using MATLAB's System Identification Toolbox to address motor asymmetries, ensuring stable and synchronized performance. A custom-built user interface developed in C# enables real-time monitoring and control via Bluetooth, allowing users to configure modes and visualize robot trajectory. The experimental setup was intentionally simple and reproducible, enabling straightforward deployment for instructional purposes. The system was validated through tasks such as line-following, obstacle avoidance, and trajectory tracking, with the latter demonstrating superior accuracy. While limitations, such as sensor inaccuracies in reflective environments and computational constraints of the ESP32, were identified, proposed future enhancements include integrating advanced sensors and machine learning algorithms. The robot's modularity, affordability, and adaptability make it a versatile platform for introducing students to robotics and conducting foundational research in control systems.*

Keywords

Differential Drive Wheeled Robot, ESP32, Experimental, Control System, Trajectory Control.

1. Introduction

Robotics is a multidisciplinary field of research and application that focuses on the design, construction, operation, and practical use of robots. This field integrates diverse disciplines, including mechanics, electronics, information technology, artificial intelligence, and even biology, to develop robotic systems that can operate autonomously or semi-autonomously. Robotics plays a pivotal role in addressing complex challenges across various domains, including manufacturing, healthcare, agriculture, and space exploration. With advancements in machine learning and sensor technologies, robotics continues to evolve, enabling the creation of more intelligent, adaptive, and efficient systems. This convergence of disciplines not only enhances the functionality of robots but also opens new frontiers for innovation and problem-solving.

As a specialized branch of robotics, mobile robots are characterized by their flexible mobility and are extensively applied across a wide range of fields, including industrial manufacturing, transportation, healthcare, agriculture, public services, education, and research. These robots play a crucial role in improving work efficiency, minimizing risks to humans in hazardous environments, and enhancing the overall quality of life. Mobile robots come in various forms, such as humanoids, autonomous vehicles, drones, and animal-inspired designs, each tailored for specific applications. Among these, differential wheeled mobile robots are the most widely adopted due to their exceptional maneuverability, mechanical simplicity, and operational reliability, particularly in structured or simple environments requiring agility. These robots utilize two independently powered wheels for movement, enabling precise control of direction and speed. Additionally, they are typically equipped with one or more caster wheels to maintain balance and enhance load-carrying capacity, making them versatile and practical for diverse tasks.

Mobile robots remain a compelling topic of interest for scientists across multiple disciplines, including kinematics and motion dynamics, system modeling and identification, sensing and control, artificial intelligence and machine learning, and human-machine interaction. Comprehensive overviews of published scientific works are available in the literature, such as references [1] for robot control and localization, [2] for trajectory planning of differential wheeled mobile robots, and [3] for applications, data acquisition, and communication.

In education, mobile robots have become invaluable tools for learning and research, widely adopted in universities and research institutes worldwide. They play a crucial role in equipping students and researchers with practical knowledge and skills in robotics and artificial intelligence. Some popular mobile robot models include:

Sphero SPRK: A compact, spherical mobile robot designed for interactive educational activities that help students explore programming and scientific concepts [4].

TurtleBot: An open-source mobile robot built on the Robot Operating System (ROS). It is widely used for teaching and research in robot control and computer vision [5]. Multiple versions of TurtleBot cater to different needs and budgets. The latest, TurtleBot 4, is priced at $2,188.00 USD for the standard edition [6].

Pioneer P3-DX: A wheeled mobile robot commonly employed in research on control, localization, and applications in constrained environments [7]. Although no longer available on the market, it was studied at Can Tho University by the author Ma Truong Thanh and colleagues [8].

Pepper: A humanoid mobile robot developed by SoftBank Robotics, used for research in artificial intelligence and human-machine interaction, particularly in services and education [9].

KUKA youBot: A mobile robot equipped with sensors and omni-directional wheels. It supports teaching and research in control, programming, and human-machine interaction [10]. This robot is actively studied at the Vietnamese-German University [11].

Anki Cozmo: A small, endearing mobile robot designed to teach children programming through fun and interactive activities. It is currently available in Vietnam for 7.35 million VND [12].

NAO Robot: NAO is a humanoid mobile robot designed for voice recognition and human-machine interaction. It is extensively used in education and research on artificial intelligence and robotics. At Ton Duc Thang University, two NAO robots are currently utilized for research and teaching, managed by the Applied Information Technology Center. As of now, the price of a NAO Robot is $12,990 USD [13].

These robots demonstrate the breadth of applications and accessibility in mobile robotics, from educational tools to advanced research platforms, underscoring their importance in shaping the future of robotics and AI education. These products are characterized by their advanced functionality and high level of integration, making them highly effective for teaching and learning. However, their commercialized and pre-packaged nature limits users' ability to explore, modify, or customize the mechanical design and control algorithms. Additionally, their high costs pose significant challenges, particularly for students from developing countries such as Vietnam, where affordability remains a critical concern.

Most commercial and research mobile robots utilize electric motors for actuation. Wheeled mobile robots (WMRs) commonly employ permanent magnet direct current (PMDC) motors [2], brushless direct current (BLDC) motors [14, 15], stepper motors [16], or direct current servo motors [17]. Among these, PMDC motors are particularly popular due to their ease of mathematical modeling and straightforward control mechanisms.

Modern commercial PID controllers frequently incorporate auto-tuning features [18], streamlining their application in practical systems. In addition, a wide range of online and offline PID tuning algorithms have been developed, leveraging techniques such as genetic algorithms [19, 20, 21, 22], particle swarm optimization [23, 24], and soft computing methods [25, 26, 27, 28]. Despite these advancements, most research focuses on PMDC motors as standalone systems, where assumptions about load torque being constant, undergoing step changes, or being subjected to minimal noise may not accurately reflect real-world conditions.

This paper presents the design and implementation of a differential wheeled mobile robot. The robot is equipped with two motor-driven wheels on each side and a caster wheel at the front for balance. A variety of sensors, including motor speed sensors, distance sensors, and optical sensors, are integrated into the system. An ESP32 microcontroller serves as the control unit, offering robust communication and data transmission capabilities via Bluetooth and Wi-Fi. The robot includes on-board control buttons and an LCD screen to display critical information. It is designed for wireless programming and can perform tasks such as following colored lines, avoiding obstacles, and tracking target objects.

To enhance user interaction, a monitoring and control interface has been developed. This interface allows users to configure operational parameters, choose operating modes, collect status data, and visualize real-time information such as wheel speeds and the robot's trajectory through intuitive graphical charts.

Compared to existing solutions, the proposed system offers a cost-effective and scalable approach to mobile robot control, making it particularly suitable for educational and research applications in resource-constrained settings. Unlike commercial robots, which are often pre-packaged and expensive, our system allows for customization and modification of both the mechanical design and control algorithms. Key unique features include the integration of a cost-effective ESP32 microcontroller, a comprehensive suite of sensors, and a custom-built user interface for real-time monitoring and control.

The remainder of this paper is structured as follows: Section 2 details the system description and modeling of the proposed robot. Section 3 addresses motion planning and trajectory generation. Section 4 covers system identification, control system design, and experimental results. Finally, Section 5 summarizes the primary contributions and outlines future research directions.

2. System Description and Modelling

2.1. System Description

The overall block diagram of the proposed Differential Drive Wheeled Mobile Robot (DDWMR) is illustrated in Fig. 1. The robot operates using two permanent-magnet direct-current (PMDC) motors, each integrated with a gearhead and encoders for precise speed and position feedback. These motors are powered through a dual H-bridge L298N motor driver, ensuring bidirectional control and efficient power delivery.

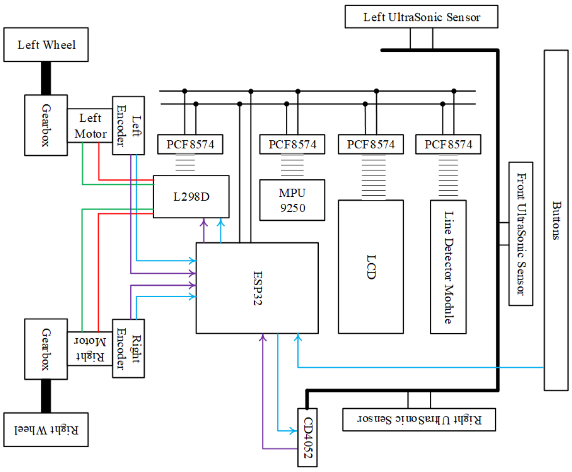


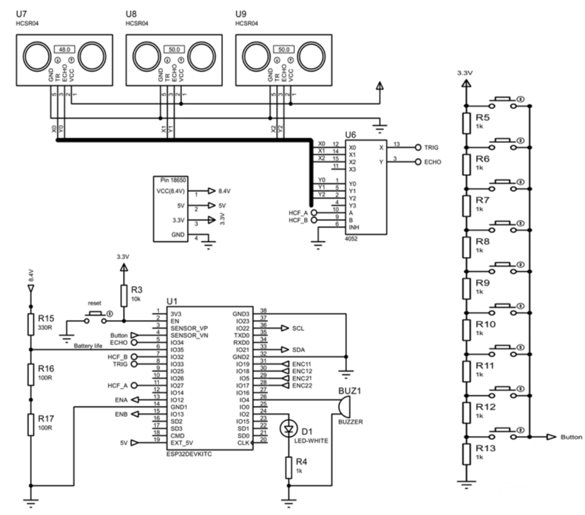
Fig. 1: System Block Diagram.

For environmental interaction, the robot is equipped with three ultrasonic sensors, strategically placed to detect obstacles in the front, left, and right directions. A line detector module is mounted underneath the robot's base, enabling it to perform line-following tasks with accuracy. Additionally, nine push buttons are installed on the front panel, allowing users to configure operational parameters and select specific operating modes with ease.

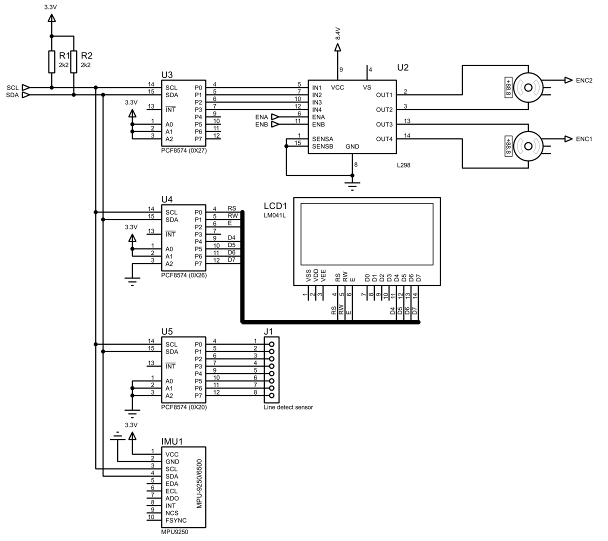
The ESP32 DEVKIT board serves as the central microcontroller, providing robust processing power and wireless communication capabilities. To enhance the robot's navigation and stability, a 9-DOF gyro-accelerometer-magnetometer module (MPU9050) is incorporated. This sensor module enables precise measurement of orientation, acceleration, and magnetic field, improving the robot's overall navigation and control performance.

Due to the limited number of general-purpose input/output (GPIO) pins on the microcontroller, the three HC-SR04 ultrasonic sensors are interfaced with the ESP32 microcontroller via a CMOS analog switch (CD4052B). This configuration optimizes GPIO usage by allowing the microcontroller to manage multiple sensors efficiently.

For other peripherals, communication with the microcontroller is established through the I2C protocol. The MPU9250 sensor is directly connected to the I2C bus for real-time orientation and motion data. However, the LCD LM041L, the dual H-bridge motor driver L298N, and the line detector module require additional I2C expansion. These devices are interfaced with the microcontroller using a PCF8574 remote 8-bit I/O expander, enabling seamless integration and reducing GPIO pin



(a) Part I



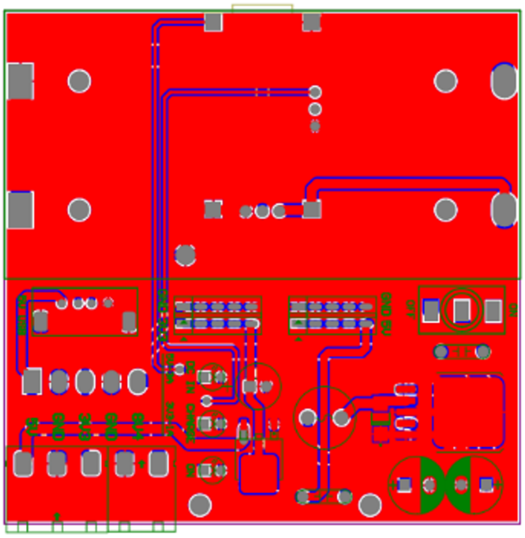
(b) Part II

Fig. 2: Detail Schematic Diagram.

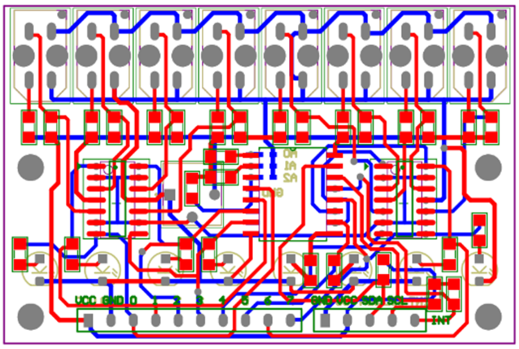
constraints. This setup ensures efficient communication and control of the robot's various components.

$$VA_i = 3.3\frac{i}{9}\,[V] \quad for \quad i = 1...9 \tag{1}$$

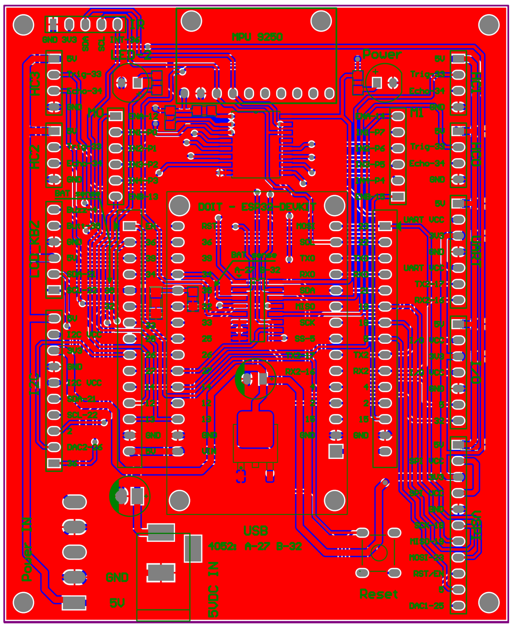
By monitoring the voltage at the Out pin (or its corresponding ADC value), the system can reliably detect which push button (PB) has been pressed. However, a limitation of this approach is that simultaneous pressing of two or more PBs cannot be detected accurately due to the overlapping signal values. Despite this constraint, the method remains effective for single-button operations. The detailed schematic of the proposed mobile robot system is presented in Fig. 2, illustrating the connections and configurations of all components. The printed-circuit-board of the robot is divided into three



(a) The Power Circuit



(b) The Sensor Circuit



(c) The Control Circuit

Fig. 3: The Printed-Circuit-Boards.

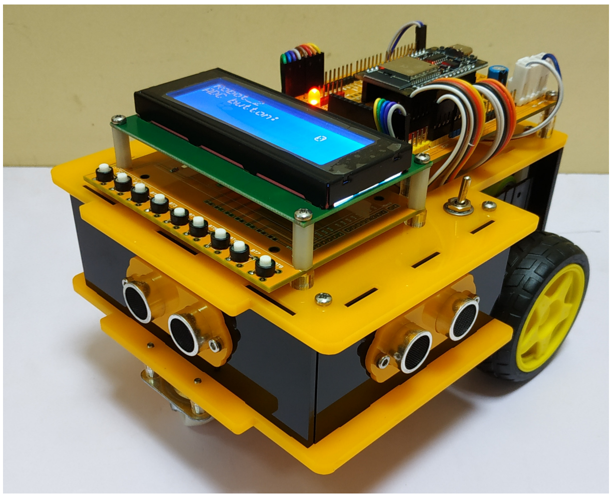


Fig. 4: Implemented Differential Drive Wheel Mobile Robot.

sub-modules: the power circuit, the control circuit, and the sensor circuit boards as shown in Fig. 3 and the implemented model is shown in Fig. 4.

In addition to the hardware components, the proposed system includes a custom-built user interface (UI) developed in C#. This UI allows users to configure operating modes, monitor robot performance, and retrieve real-time data through Bluetooth communication. The design ensures flexibility and adaptability for both educational and research purposes.

2.2. System Modelling

In our previous work, we a symmetrically half-weight model of the DDWMR has been proposed. The robot model is presented as a standard 2$^{\text{nd}}$ order system as

$$G_p = \frac{\Omega_m}{V_a} = \frac{K_{dc}\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{2}$$

where Ω_m (s) is the Laplace transform of the angular speed, V_a is the armature voltage voltage of the motors; ω_n is the natural frequency; ζ is the damping ratio, and K_{dc} is the dc gain that can be determined from the robot parameters as:

$$\begin{aligned} \omega_n &= \sqrt{\frac{B_m R_a + K_T^2}{J_T L_a}} \\ \zeta &= \frac{J_T R_a + B_m L_a}{2\sqrt{J_T L_a\left(B_m R_a + K_T^2\right)}} \\ K_{dc} &= \frac{K_T}{B_m R_a + K_T^2} \end{aligned} \tag{3}$$

Detail about the models and parameters can be found in [29].

3. System Identification, Control System Design, and Experimental Results

3.1. System Calibration and Experimental Setup

The ultrasonic sensors were calibrated by measuring their response to known distances in a controlled environment. Temperature compensation was applied to account for variations in environmental conditions. The IMU (MPU9050) was calibrated using factory-defined offsets, followed by real-world tests to correct for drift and noise. The line-following module was calibrated using a high-contrast black-and-white surface, with thresholds adjusted dynamically during operation to handle variations in lighting conditions.

Calibration challenges included environmental noise and reflections affecting ultrasonic sensor accuracy, temperature drift in the IMU, and variations in lighting conditions impacting line-following performance. These challenges were mitigated through filtering algorithms, adaptive threshold, and real-time adjustments. However, extreme environmental variations may still affect performance, and future work will focus on developing adaptive calibration techniques to further enhance robustness.

The experimental setup for the proposed system was intentionally designed to be simple, cost-effective, and reproducible, ensuring its accessibility for educational purposes. By using a flat, smooth surface and minimal environmental variables, the setup enables straightforward replication by students and educators while maintaining flexibility for further experimentation with line-following, obstacle avoidance, and trajectory tracking tasks. The testing area measured approximately 2 meters by 2 meters and was illuminated under standard indoor lighting conditions, avoiding potential interference from extreme light or reflective surfaces. The simplicity of the setup allows students and educators to easily replicate and adapt the experiments for instructional purposes.

To ensure consistent performance during testing, the system incorporates dynamic pulse-width modulation (PWM) adjustments to compensate for the gradual decline in battery voltage over time. As the battery discharges, the PWM duty cycle is automatically increased to maintain a stable armature voltage supply to the motors. This mechanism prevents degradation in motor speed and trajectory tracking accuracy, ensuring reliable operation throughout the experimental tasks. This approach, while simple, is sufficient for educational applications and provides a platform for students to explore more advanced control and power management techniques in future iterations of the system.

3.2. System Identification

To design a control system for the experimental setup, it is essential to determine the system's parameters accurately. While some parameters can be obtained directly from manufacturers' datasheets, this is not always feasible, especially for non-standard components. Furthermore, the integration of individual components can introduce slight variations in the overall system parameters that need to be accounted for during the identification process.

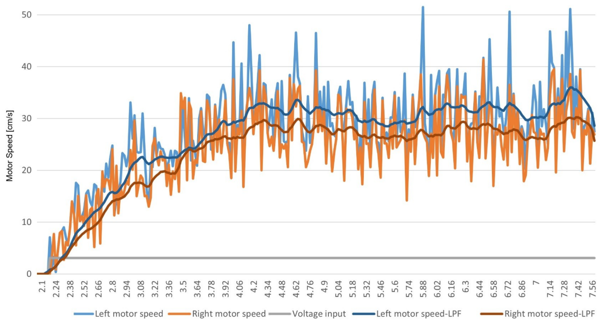
Initially, the motor speeds were controlled by adjusting the pulse-width modulation (PWM) signals generated by the ESP32 microcontroller's GPIO pins, which were then fed into the dual H-bridge L298N driver. Since the battery voltage decreases gradually during operation, the PWM duty cycle must be adjusted to maintain a consistent armature voltage supply to both motors. The motors were configured to receive the same armature voltage, ensuring uniform power distribution. The robot was tested by running freely on a flat floor while the motor speeds were monitored using encoders. Low-pass filters were applied to the encoder data to minimize noise and improve measurement accuracy. For an armature voltage of 3.1V, the experimental results are illustrated in Fig. 5.

Using the data collected, the transfer functions for the left motor-wheel and the right motor-wheel were identified using MATLAB's System Identification Toolbox. The resulting transfer functions are presented as follows:

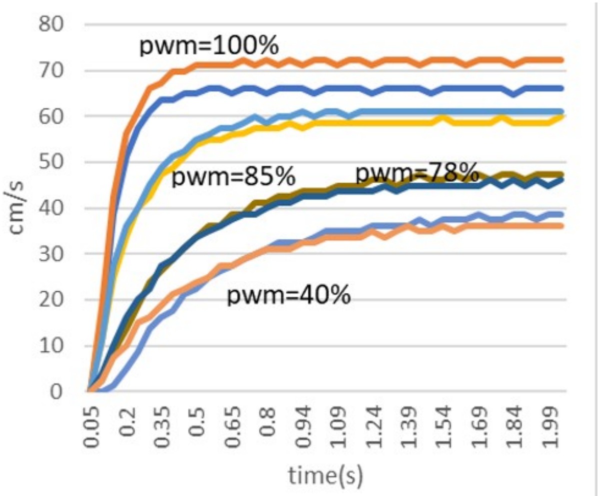
$$\begin{aligned} G_R &= \frac{V_R}{V_{aR}} = \frac{66.42}{s^2+6.008s+7.411} \\ G_L &= \frac{V_L}{V_{aL}} = \frac{175.1}{s^2+14.3s+17.12} \end{aligned} \tag{4}$$

in which V_R and V_L denote the linear speeds, while V_{aR} and V_{aL} represent the armature voltages of the right and left motors, respectively. The analysis reveals that the transfer functions of the left and right motor-wheel systems are not identical. This asymmetry arises despite the motors and wheels being selected with identical parameters from the same manufacturer. The disparity is primarily attributed to the non-uniform allocation of other mechanical components in the system.

The characteristics of the Differential Drive Wheeled Mobile Robot (DDWMR) can be expressed in terms of the motor-wheel systems, as detailed in Table 1. From this table, we observed significant asymmetry between the left and right motors, with the left motor exhibiting a natural frequency of 4.14 rad/s and a damping ratio of 1.73, while the right motor had a natural frequency of 2.72 rad/s and a damping ratio of 1.10. This asymmetry was attributed to mechanical imbalances in the robot's structure. To compensate, we fine-tuned the PID parameters for each motor individually, ensuring synchronized performance despite the differences in dynamics. Experimental validation confirmed that this approach effectively balanced the motor responses and improved the robot's overall stability and control.



(a) Measured Data without and with Filter



(b) Motors Speed with different PWM Values

Fig. 5: Open-loop Tests.

Tab. 1: Motor-Wheel Systems Characteristics

No	Symbol	Params	Left	Right
1	ω_n [rad/s]	Natural Frequency	4.14	2.72
2	ζ [-]	Damping Ratio	1.73	1.10
3	K_{dc} [-]	DC Gain	10.23	8.96

Furthermore, the open-loop characteristics of the motor-wheel systems were analyzed using MATLAB's "Linear System Analysis" toolbox. The results, presented in Figure 6, confirm that both motor-wheel systems exhibit stable dynamics in their open-loop configurations. This stability forms a solid foundation for the design and implementation of robust closed-loop control systems for the robot.

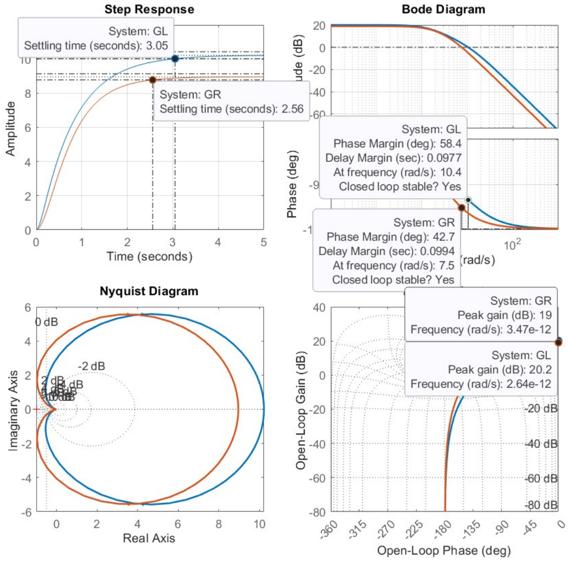


Fig. 6: Open-loop characteristics.

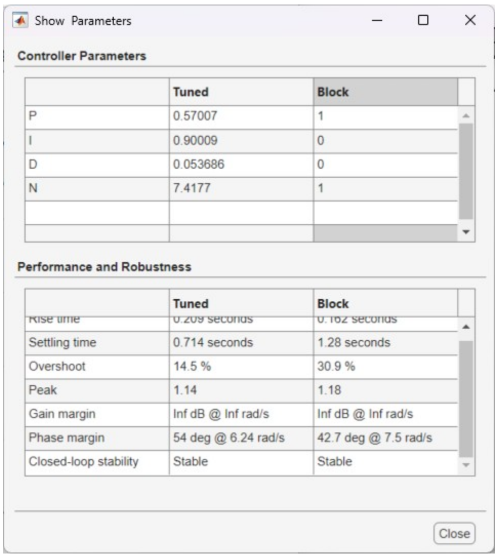
3.3. PID Controllers Design

The PID controllers for the left and right motors were tuned using MATLAB's 'PID Tuner App,' which provided an intuitive interface for optimizing the proportional, integral, and derivative gains. The tuning process involved iteratively adjusting the parameters to achieve desired dynamic responses, such as minimizing overshoot, ensuring stability, and optimizing settling time. The initial tuning was based on the identified transfer functions for each motor as shown in Fig. 7, and the parameters were further fine-tuned experimentally to account for real-world conditions.

3.4. Experimental Results

To assess the performance of the proposed system, various case studies were implemented. In the first scenario, the robot was tasked with moving forward 60 cm from its initial position and then rotating 180 degrees about its wheelbase-center. Three distinct control strategies were applied: motor speed references, robot speed references, and trajectory tracking.

Battery voltage variation was managed by dynamically adjusting the PWM duty cycle to maintain a consistent armature voltage supply to the motors. This approach ensured stable motor performance despite gradual decreases in battery voltage. Additionally, real-time voltage monitoring was implemented to provide feedback and allow for corrective actions if necessary. Future work could explore the use of voltage regulators or adaptive control algorithms to further optimize power management.


Show Parameters

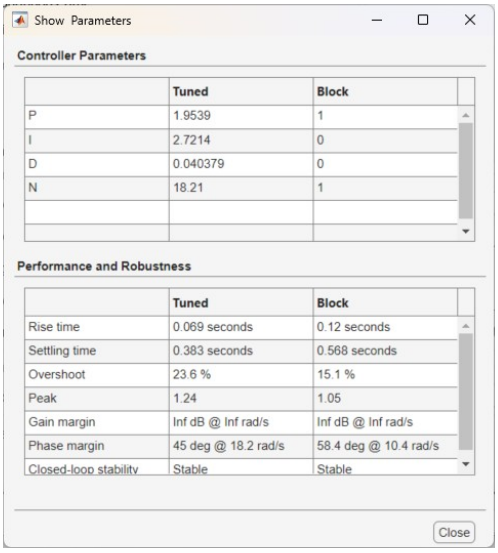
Controller Parameters

	Tuned	Block
P	0.57007	1
I	0.90009	0
D	0.053686	0
N	7.4177	1

Performance and Robustness

	Tuned	Block
Rise time	0.209 seconds	0.162 seconds
Settling time	0.714 seconds	1.28 seconds
Overshoot	14.5 %	30.9 %
Peak	1.14	1.18
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	54 deg @ 6.24 rad/s	42.7 deg @ 7.5 rad/s
Closed-loop stability	Stable	Stable

(a) Right Motor


Show Parameters

Controller Parameters

	Tuned	Block
P	1.9539	1
I	2.7214	0
D	0.040379	0
N	18.21	1

Performance and Robustness

	Tuned	Block
Rise time	0.069 seconds	0.12 seconds
Settling time	0.383 seconds	0.568 seconds
Overshoot	23.6 %	15.1 %
Peak	1.24	1.05
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	45 deg @ 18.2 rad/s	58.4 deg @ 10.4 rad/s
Closed-loop stability	Stable	Stable

(b) Left - Motor

Fig. 7: Tuned Parameters of the PID Controllers.

1) Motor Speed Reference (MSR)

Using the S-curve trajectory generation methodology, reference trajectories for specific points were generated. Based on the backward kinematic model of a Differential Drive Wheeled Mobile Robot (DDWMR), the reference angular speeds for the left and right motors were derived. PID controllers were employed to ensure the motors' speeds followed the reference values.

Performance: As shown in Fig. 8, delays in PID controller performance led to a mismatch in motor speed synchronization. Although the motors reached their reference speeds after a short delay, this resulted in significant errors in the robot's position and orientation.

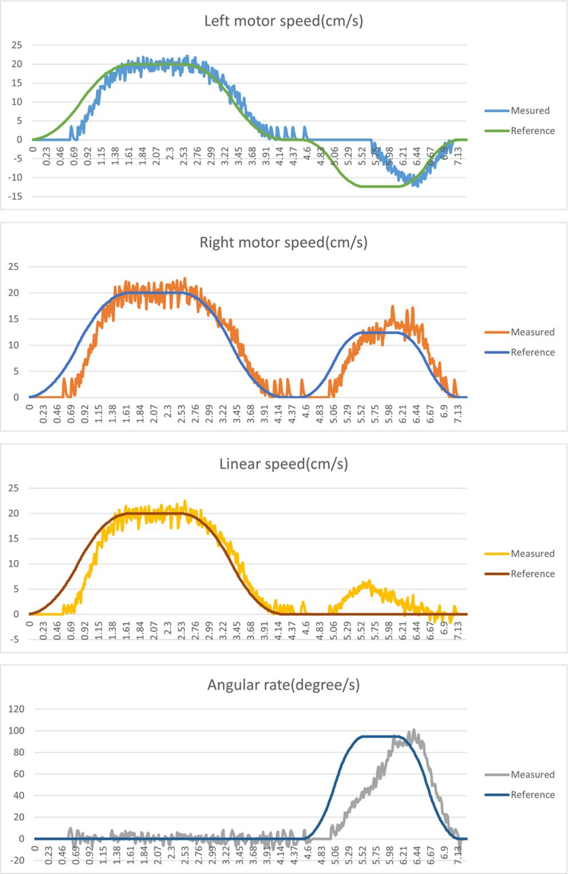


Fig. 8: Motor Speed References.

2) Robot Speed References (RSR)

In this strategy, the reference trajectory was converted into linear and angular velocity references for the robot. This control approach also employed PID controllers to regulate the robot's motion. Similar to the motor speed control strategy, the robot experienced position and orientation errors due to delays in velocity tracking, as illustrated in Fig. 9.

3) Trajectory Tracking (TT)

In the proposed control strategy, a TT system as shown in Fig. 10a was implemented to directly regulate the robot's motion along the desired path. This approach ensured tighter adherence to the reference trajectory.

As depicted in Fig. 10b, the proposed control system allowed the robot to follow the reference trajectory with minimal error, demonstrating significant improvement over the previous strategies.

The Root-Mean-Square Error (RMSE) for the different control strategies is presented in Table 2. The data highlights that the trajectory tracking method outperformed both motor speed and robot speed reference strategies, delivering superior accuracy and control.

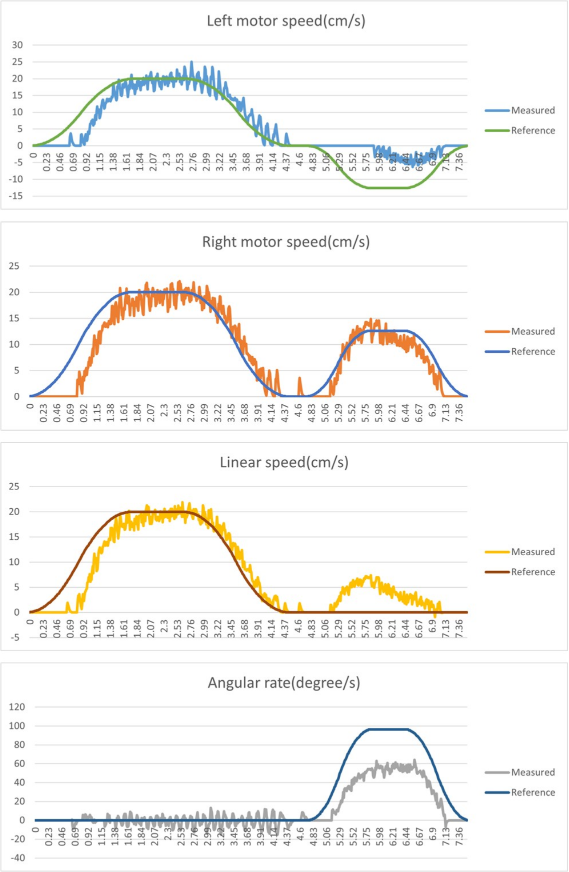


Fig. 9: Robot Speed References.

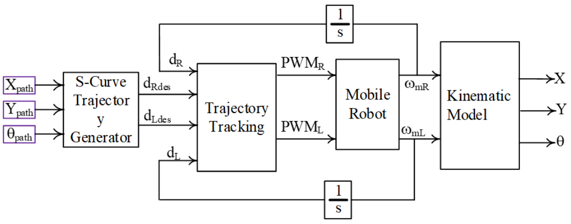
Tab. 2: Evaluation of Trajectory Tracking Strategies

No.	Quan.	Unit	MSR	RSR	TT
1	Left motor	cm/s	4.35	5.33	1.95
2	Right motor	cm/s	2.66	2.88	1.61
3	Angular speed	cm/s	18.81	20.03	8.85
4	Linear speed	deg/s	2.61	3.3	1.35

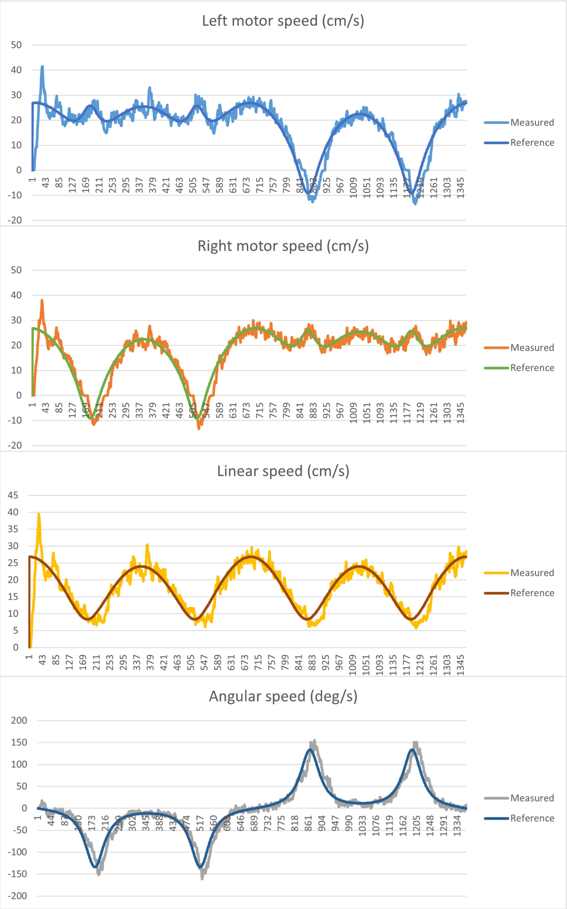
This analysis confirms that the proposed trajectory tracking approach effectively minimizes positional and orientation errors, providing a robust and precise solution for controlling the DDWMR.

In addition, the effectiveness of the PID tuning process was validated through trajectory tracking tests, as shown in Fig. 10. The results demonstrated that the tuned PID controllers enabled the robot to follow predefined paths with minimal error, confirming the robustness of the control system.

Additionally, in this work, a User Interface (UI) also designed and programmed. The UI allows for the evaluation of the effectiveness of different controller parameters across various tasks. Different scenarios can be tested, including Point-to-Point, 8-Shape, and Pure-Pursuit modes, as illustrated in Fig. 11. Each of these modes serves a specific purpose, providing a comprehensive evaluation of the system's performance under

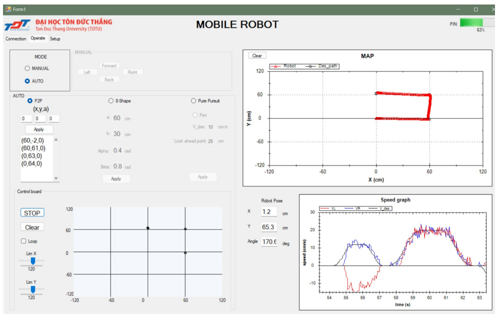


(a) Schematic



(b) Results

Fig. 10: Trajectory Tracking.



MODE: ○ MANUAL ● AUTO

AUTO: ● P2P ○ 8-Shape ○ Pure Pursuit

(x,y,a): 0 0 0

(60,-2,0)
(60,61,0)
(0,63,0)
(0,64,0)

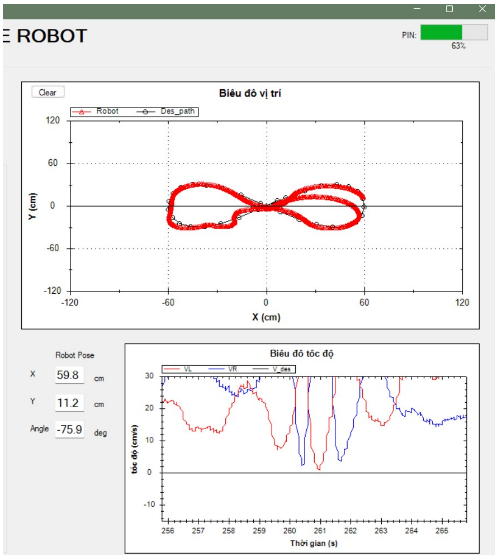
a: 60 cm
b: 30 cm
Alpha: 0.4 rad
Beta: 0.8 rad

○ Pen
V_des: 10 cm/s
Look ahead point: 25 cm

☐ Loop

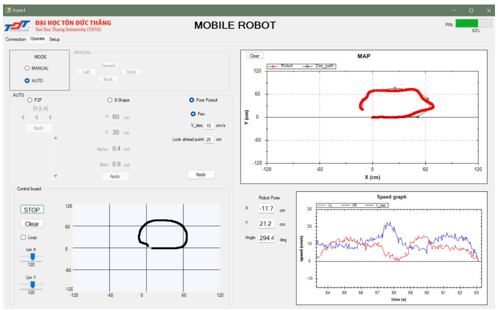
Robot Pose
X 1.2 cm
Y 65.3 cm
Angle 170.6 deg

(a) Point-to-Point



Robot Pose
X 59.8 cm
Y 11.2 cm
Angle -75.9 deg

(b) 8-Shape



MODE: ○ MANUAL ● AUTO

AUTO: ○ P2P ○ 8-Shape ● Pure Pursuit

(x,y,a): 0 0 0

a: 60 cm
b: 30 cm
Alpha: 0.4 rad
Beta: 0.8 rad

● Pen
V_des: 10 cm/s
Look ahead point: 25 cm

☐ Loop

Robot Pose
X -11.7 cm
Y 21.2 cm
Angle 294.4 deg

(c) Pure Pursuit

Fig. 11: Different Operating Mode with UI.

different conditions. By adjusting controller parameters for each mode, the system's response can be fine-tuned, helping to optimize its performance and assess its capability in real-world applications. These scenarios offer valuable insights into how well the robot can adapt to different movement patterns, enhancing its overall functionality.

4. Conclusion

In this study, we developed a differential drive wheeled mobile robot system that integrates precise control, accurate system modelling, and an intuitive user interface. The PID-based control system ensures stable and synchronized motor performance, with parameters meticulously tuned using MATLAB's PID Tuner App to address asymmetries in motor dynamics. A comprehensive system identification process, conducted using MATLAB's System Identification Toolbox, enabled the derivation of accurate transfer functions for both motors, ensuring optimal controller design tailored to the robot's characteristics. Additionally, a custom-built user interface developed in C# provides real-time monitoring and control through Bluetooth communication, allowing users to configure operating modes, adjust parameters, and visualize real-time data such as wheel speed and trajectory adherence. This integration of robust control mechanisms, precise modelling, and interactive monitoring creates a scalable and adaptable platform ideal for both educational and research applications in robotics.

The robot's versatility makes it suitable for various educational applications, ranging from introductory robotics courses to advanced experiments in trajectory tracking and adaptive control. This flexibility addresses the needs of diverse users, from students to researchers. The modular nature of the system ensures scalability for more advanced research applications.

The current implementation exhibits several notable limitations that warrant consideration. The sensor suite's accuracy is constrained by environmental factors, with ultrasonic sensors being susceptible to acoustic noise and surface reflectivity variations, while the line-following sensors demonstrate sensitivity to ambient lighting conditions. The ESP32 microcontroller's computational capacity presents inherent restrictions on implementing more sophisticated control algorithms, particularly those requiring intensive real-time calculations. Furthermore, the Bluetooth communication protocol introduces bandwidth limitations that may impact data transmission capabilities in applications requiring high-frequency sensor updates or complex control commands. Additionally, the system's current optimization for controlled environments suggests potential performance degradation in unstructured or dynamic settings, where environmental variables may exceed the designed operational.

Future enhancements to the system could focus on advancing its autonomous capabilities and environmental adaptability through several key developments. The integration of machine learning algorithms, particularly reinforcement learning for dynamic obstacle avoidance and deep learning for advanced perception tasks, could significantly improve the robot's decision-making capabilities. Enhanced sensor integration, such as LiDAR or stereo cameras, would enable more precise environmental mapping and obstacle detection. The implementation of model predictive control (MPC) could provide more robust handling of varying loads and surface conditions, while incorporating SLAM (Simultaneous Localization and Mapping) would facilitate more effective navigation in unstructured environments. Additionally, the development of adaptive control algorithms could optimize power consumption and extend operational time, further improving the system's overall efficiency and utility in real-world applications.

Author Contributions

T.-V. V. and A.-M. D. T. developed the theoretical framework, performed analytical calculations, and conducted numerical simulations. T.-Q. N. and C.-K. T. designed and implemented the hardware model, while C.-K. T. and P.- U. L. N. carried out the experiments and data collection. P.-U. L. N. also designed and developed the graphical user interface (GUI). All authors contributed to the discussion, interpretation of results, and revision of the manuscript. T.-V. V. supervised the project and provided overall guidance throughout the study.

References

[1] TZAFESTAS, S. G. Mobile Robot Control and Navigation: A Global Overview. *Journal of Intelligent & Robotic Systems.* 2018, vol. 91, pp. 35–58. DOI: 10.1007/s10846-018-0805-9.

[2] GARCÍA-SÁNCHEZ, J. R., et. al. Tracking Control for Mobile Robots Considering the Dynamics of All Their Subsystems: Experimental Implementation. *Complexity.* 2017. DOI: 10.1155/2017/5318504.

[3] DING, N., P. CHUNRONG, L. MIN, W. CELIMUGE. A Comprehensive Review on Automatic Mobile Robots: Applications, Perception, Communication and Control. *TechRxiv.* 2017. DOI: 10.36227/techrxiv.19619880.v1.

[4] Spero SPRK+: Robot Toy. https://sphero.com/collections/all/family_sprk.

[5] TurtleBot. https://www.turtlebot.com/

[6] Chironix, TurtleBot4, https://www.chironix.com/base-platforms/turtlebot-4.

[7] ProE, Robot Turtle 3 Burger. https://www.proe.vn/robot-turtlebot-3-burger.

[8] MA, T.-T., D. THANH-NGHI, P. NGUYEN-KHANG, C. NGAN-KHANH. Điều khiển robot pioneer P3-DX bằng tiếng nói với đặc trưng MFCC và giải thuật naïve bayes nearest neighbors. *Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin FAIR 2015, ĐHCN Hà Nội.* 2015. https://sj.ctu.edu.vn/ql/docgia/tacgia-42773/baibao-39008.html.

[9] SoftBank Robotics, For better business just add Pepper. https://us.softbankrobotics.com/pepper.

[10] You-bot store, KUKA YouBot Kinematics, Dynamics And 3D Model. http://www.youbot-store.com/developers/kuka-youbot-kinematics-dynamics-and-3d-model.

[11] KHANG H. V. N., T. B. MINH, V. C. LE, P. X. DO. Robot gesture control using online feedback data with multi-tracking capture system. *AETA 2022—Recent Advances in Electrical Engineering and Related Sciences: Theory and Application.* 2017. DOI: 10.1007/978-981-99-8703-0.

[12] Jola.vn, Robot Anki Cozmo - Trí tuệ nhân tạo – USA, https://jola.vn/robot-anki-cozmo-tri-tue-nhan-tao-usa.

[13] Robotlab, NAO robot power V6 standard edition for research, https://www.robotlab.com/higher-ed-robots/store/nao-power-v6-standard-edition.

[14] BEN SAAD, K., A. BARKAOUI, B. FRANCOIS. Sliding mode control of a four brushless motor robot with independent wheels. *2016 International Conference on Electrical Sciences and Technologies in Maghreb (CISTEM), Marrakech & Bengrir, Morocco*. 2016, pp. 1-5. DOI: 10.1109/CISTEM.2016.8066805.

[15] LUAN, Y., et al. Design of Motion Control System for Omnidirectional Four-Drive Mobile Robot. *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China*. 2019, pp. 1409-1413. DOI: 10.1109/ITAIC.2019.8785450.

[16] VACHHANI, L., A. D. MAHINDRAKAR, K. SRIDHARAN. Mobile Robot Navigation Through a Hardware-Efficient Implementation for Control-Law-Based Construction of Generalized Voronoi Diagram. *IEEE/ASME Transactions on Mechatronics*. 2011, vol. 16, no. 6, pp. 1083-1095. DOI: 10.1109/TMECH.2010.2076825.

[17] ZHANG, J., N. WANG, S. WANG. A developed method of tuning PID controllers with fuzzy rules for integrating processes. *Proceedings of the 2004 American Control Conference, Boston, MA, USA*. 2004, vol.2, pp. 1109-1114. DOI: 10.23919/ACC.2004.1386720.

[18] PID Control with PID_compact for SIMATIC S7-1200/S7-1500. Retrieved from https://support.industry.siemens.com/cs/document/100746401/pid-control-with-pid_compact-for-simatic-s7-1200-s7-1500?dti=0&lc=en-WW.

[19] BINDU, R., M. K. NAMBOOTHIRIPAD. Tuning of PID Controller for DC Servo Motor using Genetic Algorithm. *International Journal of Emerging Technology and Advanced Engineering*. 2012, vol. 2, iss. 3, pp. 310-314. https://www.ijetae.com/files/Volume2Issue3/IJETAE_0312_50.pdf.

[20] THOMAS, N., P. POONGODI. Position Control of DC Motor Using Genetic Algorithm Based PID Controller. *Proceedings of the World Congress on Engineering*. 2009, vol. II. https://masters.donntu.ru/2017/etf/buzulov/library/article8.pdf.

[21] EL-DEEN, A. T., A. A. H. MAHMOUD, A. R. EL-SAWI. Optimal PID tuning for DC motor speed controller based on genetic algorithm. *International Review of Automatic Control (IREACO)*. 2015, vol. 8, no. 1, pp. 80-85. DOI: 10.15866/ireaco.v8i1.4839.

[22] JAISWAL, M., M. PHADNIS. Speed Control of DC Motor Using Genetic Algorithm Based PID Controller. *International Journal of Advanced Research in Computer Science and Software Engineering*. 2013, vol. 3, no. 7, pp. 247-253. https://www.scirp.org/reference/referencespapers?referenceid=1602724.

[23] SOLIHIN, M. I., L. F. TACK, M. L. KEAN. Tuning of PID Controller Using Particle Swarm Optimization (PSO). *International Journal on Advanced Science, Engineering and Information Technology*. 2011.

[24] PAYAKKAWAN, P., K. KLOMKARN, P. SOORAKSA. Dual-line PID controller based on PSO for speed control of DC motors. *9th International Symposium on Communications and Information Technology, Icheon, Korea (South)*. 2009, pp. 134-139. DOI: 10.1109/ISCIT.2009.5341272.

[25] BANSAL, U. K., R. NARVEY. Speed Control of DC Motor Using Fuzzy PID Controller. *Advance in Electronic and Electric Engineering*. 2013, vol. 3, no. 9, pp. 1209-1220.

[26] TIR, Z., E. OUED, E. OUED. Implementation of a fuzzy logic speed controller for a permanent magnet dc motor using a low-cost Arduino platform. *5th International Conference on Electrical Engineering - Boumerdes (ICEE-B), Boumerdes, Algeria*. 2017, pp. 1-4. DOI: 10.1109/ICEE-B.2017.8192218.

[27] KUSHWAH, M., P. A. PATRA. Tuning PID Controller for Speed Control of DC Motor Using Soft Computing Techniques-A Review. *Advance in Electronic and Electric Engineering*. 2014, vol. 4, no. 2, pp. 141-148.

[28] VELAGIC, J., A. GALIJASEVIC. Design of fuzzy logic control of permanent magnet DC motor under real constraints and disturbances. *IEEE Control Applications, (CCA) & Intelligent Control, (ISIC), St. Petersburg, Russia*. 2009, pp. 461-466. DOI: 10.1109/CCA.2009.5281099.

[29] VU, T. V., A. M. TRAN, B. H. NGUYEN, H. V. TRAN. Development of Decentralized Speed Controllers for a Differential Drive Wheel Mobile Robot. *Journal of Advanced Engineering and Computation*. 2023, vol. 7. no. 2. DOI: 10.55579/jaec.202372.399.