Trajectory tracking control based on genetic algorithm and proportional integral derivative controller for two-wheel mobile robot

Vo Thanh Ha[1], Than Thi Thuong[2], Le Ngoc Truc[3]
[1]Department of Cybernetics, Faculty of Electrical and Electronic Engineering, University of Transport and Communications, Hanoi, Vietnam
[2]Department of Control and Automation, Faculty of Electrical-Automation, University of Economics-Technology for Industries, Hanoi, Vietnam
[3]Department of Electromechanics-Refrigeration and Air-Conditioning Technology, Faculty of Automobile Engineering, Hung Yen University of Technology and Education, Hung Yen, Vietnam

Article Info

Article history:

Received Sep 7, 2023
Revised Jan 10, 2024
Accepted Feb 24, 2024

Keywords:

Genetic algorithm
Genetic algorithm proportional integral derivative
Mobile robot
Proportional integral derivative
Two-wheel mobile robot

ABSTRACT

This paper uses the genetic algorithm (GA) to optimize the proportional integral derivative (PID) controller parameters to present the motion control design for a two-wheeled mobile robot autonomous system. The GA algorithm determines a collision-free travel curve for a robot with a tangential velocity restriction constraint. A trajectory-tracking controller based on the PID control structure is developed to monitor the calculated route curves for the mobile robot. Simulation results show the effectiveness of the GA-PID controller compared to the PID controller. The GA-PID controller demonstrates improved performance in trajectory tracking and collision avoidance, making it suitable for controlling the motion of two-wheeled mobile robots. The GA's optimization process allows for better tuning of the PID controller parameters, resulting in more efficient and accurate robot motion control. The results suggest that the proposed GA-PID controller is a promising approach for enhancing mobile robots' autonomous navigation capabilities.

Corresponding Author:

Vo Thanh Ha
Department of Cybernetics, Faculty of Electrical and Electronic Engineering
University of Transport and Communications
Room 505, Building A6, No.3 Cau Giay Street, Lang Thuong Ward, Dong Da District, Hanoi, Vietnam
Email: vothanhha.ktd@utc.edu.vn

1. INTRODUCTION

Nowadays, Robots play a vital role in Industry 4.0 and smart factories. Autonomous robots have improved automation and addressed production continuity issues [1]–[3]. Self-propelled robots operate in open environments, necessitating path planning and trajectory monitoring. This challenge has sparked interest among many scientists and manufacturing companies [4], [5]. The trajectory and navigation setup for self-propelled robots involves using linear, intelligent, and hybrid control methods. A common approach includes employing two control loop structures, such as an external kinematic loop using the Lyapunov function to synthesize the position and speed controller [6]. Furthermore, various dynamic loop control techniques like proportional integral derivative (PID) control [7], slip control [8]–[10], and backstepping command [11] have been applied. However, these control strategies effectively handle self-propelled robots in ideal working conditions without external disturbances. Some studies [12]–[15] have introduced uncertainty parameters into the dynamic equations of autonomous robots, leading to the utilization of hybrid adaptive control methods that merge adaptive control with neural network approximation of unknown components [16]. Another research direction combines adaptive management with fuzzy logic control [17]. These hybrid control approaches demonstrate strong performance, compensating for model deviations and system input noise. Despite the ongoing research and development of advanced controllers, traditional PID controllers remain a favored choice for controlling orbital self-propelled robots due to their effectiveness in ensuring stability and traction, albeit with slightly lower accuracy [18]–[20]. An additional factor impacting the precision of traction control in autonomous robots is the robot's parameters, such as weight, cargo mass, wheel movement, and environmental friction. These variables can cause the PID controller to struggle to adjust or maintain control along the robot's path. To enhance the PID controller's effectiveness in this scenario, this study suggests utilizing a genetic algorithm (GA) to optimize three PID controller parameters [21]–[23]. A GA is a metaheuristic inspired by natural selection and falls under the evolutionary algorithms category. Using biologically inspired operators like mutation, crossover, and choice, GA are commonly u0sed to find high-quality optimization and search problem solutions [24]–[26]. The accuracy of the control solution is verified through simulation in matrix laboratory MATLAB/Simulink software, comparing the GA-PID controller with the PID controller. The simulation results indicate that the GA-PID controller offers improved control accuracy and convergence speed.

The article is divided into five main sections. The first part introduces the research aim of autonomous robot navigation. Section 2 then outlines the dynamic and control models of the self-propelled robot, providing the foundation for the control design in section 3. Section 3 uses an algorithm to enhance the PID controller solution. The optimal parameter genetics of the PID controller are discussed. Section 4 compares the correctness and feasibility of the GA-PID controller with the PID controller using MATLAB simulation software. The concluding section will present the findings, assessment, and future research directions for autonomous robot motion control.

2. KINEMATICS AND DYNAMICS FOR DIFFERENTIAL MOBILE ROBOTS

In research, a mobile robot with two-wheel rolls without slip and no side slip, the passive wheel has a negligible impact on the dynamics. Attach a fixed frame of reference in the plane of the moving medium $[O, x, y, z]$ as depicted in Figure 1 where: A is the midpoint of two-wheels, C is the coordinates of the center of gravity of the robot, a is the distance between the coordinates of the center of gravity to the wheel axle, Ra is the radius of the active wheel, 2L is the distance between the two wheels. wheel, m is the mass of the robot, m_ω is the mass of the wheel and engine, mc is the mass of chassis, I moment of inertia, v, ω is velocity and angular velocity, q is funding the robot's set direction, $\dot{\varphi}_r, \dot{\varphi}_l$ are angular velocity, θ is the orientation angle.

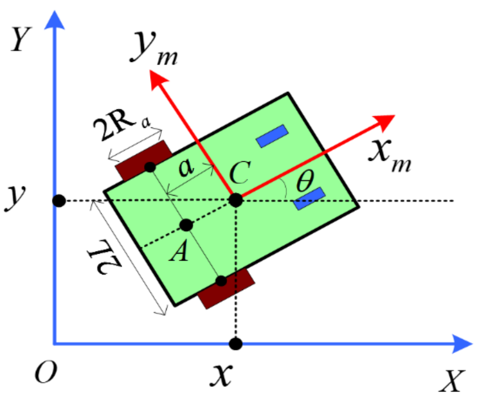

Figure 1. Kinetic relationship of differential mobile robot

2.1. Forward kinematics model

The forward kinematics of the differential autonomous robot is written by (1), to determine the robot's position and orientation based on the motion of its wheels. These equations allow us to calculate the robot's new position and orientation as it moves. By updating these values based on the wheel velocities, we can track the robot's trajectory accurately. These equations allow us to calculate the robot's new position and orientation as it moves. By updating these values based on the wheel velocities, we can track the robot's trajectory accurately.

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} R_a \frac{\dot{\varphi}_R + \dot{\varphi}_L}{2} cos(\theta) \\ R_a \frac{\dot{\varphi}_R + \dot{\varphi}_L}{2} sin(\theta) \\ \frac{R_a}{2L}(\dot{\varphi}_R - \dot{\varphi}_L) \end{bmatrix} \Leftrightarrow \dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} cos\,\theta & -a\sin\theta \\ sin\,\theta & a\cos\theta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (1)$$

In inverse kinematics, the constraint equation in the form of a matrix as (2):

$$A(q)\dot{q} = \begin{bmatrix} -sin(\theta) & cos(\theta) & 0 & 0 & 0 \\ -cos(\theta) & -sin(\theta) & -a & R_a & 0 \\ -cos(\theta) & -sin(\theta) & a & 0 & R_a \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\varphi}_R \\ \dot{\varphi}_L \end{bmatrix} \Rightarrow \dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\varphi}_R \\ \dot{\varphi}_L \end{bmatrix} \quad (1)$$

2.2. Dynamics model

The kinetic energy of the robot is 3 components (kinetic energy of the vehicleK_c, kinetic energy of the right band$K_{\omega R}$, and kinetic energy of the left wheel $K_{\omega L}$). The forces along the axes are as Figure 2. The forces acting along the axes, as shown in Figure 2, play a crucial role in determining the motion and energy distribution within the system. By considering these components and forces, we can analyze the robot's dynamics and understand how energy is transferred and utilized during its operation.

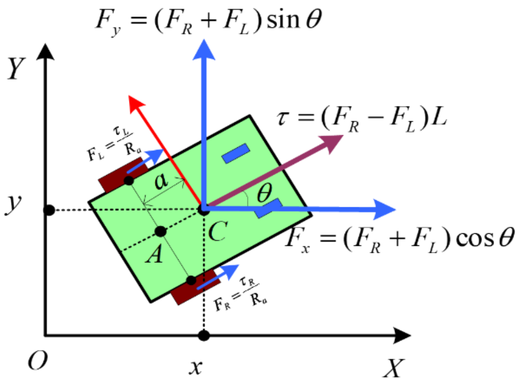

Figure 2. Force diagram acting on the robot's directions

The dynamic model can also be written in the form (3):

$$\begin{bmatrix} F_x \\ F_y \\ F_\theta \end{bmatrix} = \begin{bmatrix} \frac{cos\,\theta}{R_a}(\tau_R + \tau_L) \\ \frac{sin\,\theta}{R_a}(\tau_R + \tau_L) \\ \frac{L}{R_a}(\tau_R - \tau_L) \end{bmatrix} = \frac{1}{R_a} \begin{bmatrix} cos\,\theta & cos\,\theta \\ sin\,\theta & sin\,\theta \\ L & -L \end{bmatrix} \begin{bmatrix} \tau_R \\ \tau_L \end{bmatrix} \quad (2)$$

2.3. Kinetic error model

Determine the deviation of the kinematic model between the real position (R coordinate frame), and the set position (C coordinate frame) as shown in Figure 3. The kinetic error function is written by (4). The goal of optimization is to minimize this error function by adjusting the parameters of the model.

$$\dot{e}_q = \begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_\theta \end{bmatrix} = \begin{bmatrix} \cos(e_\theta) & 0 \\ \sin(e_\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_r \\ \omega_r \end{bmatrix} + \begin{bmatrix} -1 & e_y \\ 0 & -e_x \\ 0 & -1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (3)$$

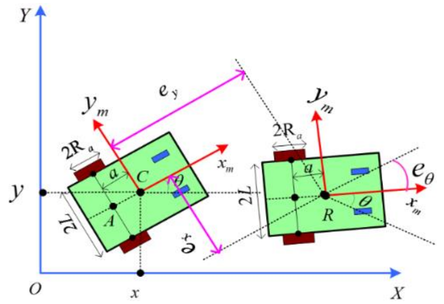

Figure 3. The error between real trajectory (R coordinate system) and set trajectory (C coordinate system)

3. DESIGN A GA–PID CONTROLLER

The parameter optimization of the PI controller using the GA algorithm is performed as Figure 4. The proposed GA algorithm in building a GA-PID controller to track the trajectory of an object is a self-propelled robot with a kinematic model as suggested in part 2 above. The proposed GA-PID controller to suppress the velocity error $e(t)$ is determined by (5):

$$e(t) = \upsilon - \upsilon_a = \begin{bmatrix} \upsilon \\ \omega \end{bmatrix} - \begin{bmatrix} \upsilon_a \\ \omega_a \end{bmatrix} \quad (4)$$

where $\upsilon = \begin{bmatrix} \upsilon \\ \omega \end{bmatrix}$ is the velocity according to the robot's set trajectory and $\upsilon_a = \begin{bmatrix} \upsilon_a \\ \omega_a \end{bmatrix}$ is the real speed of the robot.

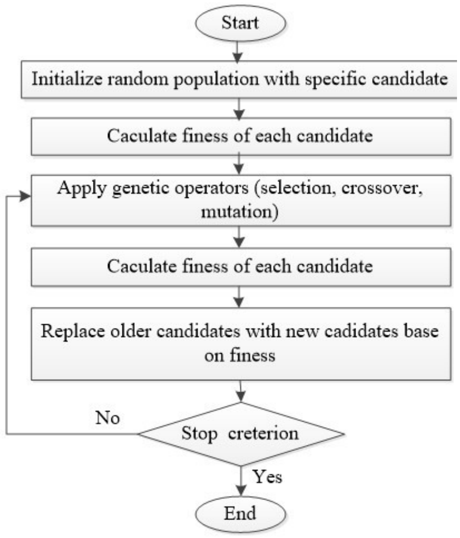

Figure 4. The flowchart of GA algorithm

The proposed controller controls the speed of the self-propelled robot with the input of the orbital deviation e(t) and the speed set as (4). The speed deviation gradually reaches zero by the PID controller according to (6):

$$\tau_{dk} = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt} \quad (5)$$

where K_p, K_i, K_d are the positive coefficients of determination matrices. Tests with GA algorithm and error or trial and error on the values of K_P, K_I, and K_D were performed to determine the comparison of the obtained PID parameters. The objective function of the controller tuning process, in this problem, is defined as integral of time and absolute error (IATE) (7):

$$J_2 = \int_0^{\infty} t|e(t)|dt \quad (6)$$

4. SIMULATION RESULTS

The block diagram to simulate the motion control system for robots on MATLAB/SIMULINK as Figure 5. The simulation the robot to move with 2 types of trajectories: Case 1 is a circular orbit and case 2 is a square orbit to evaluate the effectiveness of the proposed controller. Robot model parameters is $m = 12$ kg; $J = 0.5$ kgm^2; $a = 0.2$ m; L=0.25 m; r=0.05 m

The GA algorithm is supported by MATLAB software and is detailed in. In this context, the GA tool in MATLAB is introduced. It only solves the optimization problem to obtain the values {Kp_opt, Kd_opt, Ki_opt} with the search space limited by (8):

$$\begin{cases} \alpha K_p \leq K_{p\_opt} \leq \beta K_p \\ \alpha K_d \leq K_{d\_opt} \leq \beta K_d \\ \alpha K_i \leq K_{i\_opt} \leq \beta K_i \end{cases} \tag{7}$$

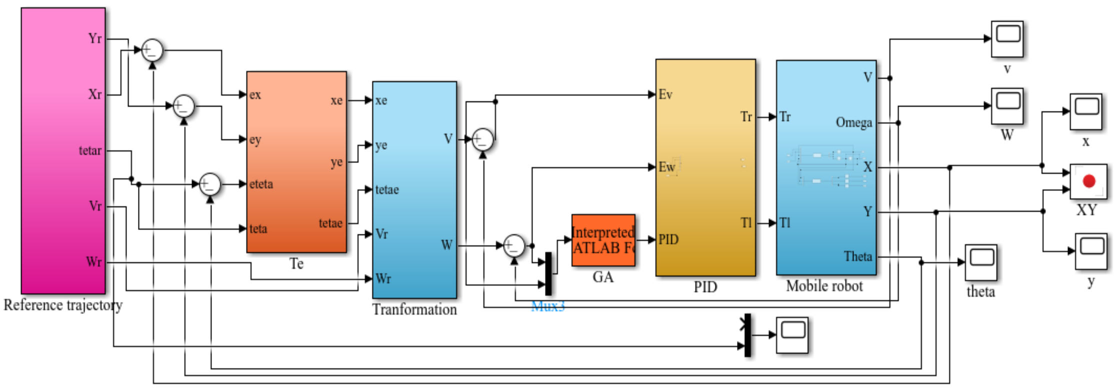

Figure 5. Structure diagram to simulate the motion control system for mobile robot

The parameters of the GA algorithm in this study are selected with evolution over is 100 generations, population size is 100, crossover frequency is 0,8, and mutation probability is adjusted in the range from 0.001 to 0.01, upper bound i_b=[0 0 0], lower bound u_b=[50 50 50]. The optimal time of three PID parameters with chromosome max is 100.

Case 1: the orbit is set as a circle with the equation $x = 5 * cos(\frac{\pi}{15})$; $y = 5 * sin(\frac{\pi}{15})$. The result of the GA of PID_ITAE to optimize the three parameters of PID is shown in Figure 6. The robot's trajectory moves in a circle, trajectory and trajectory deviation in x, y for robot is shown Figures 7 to 9.

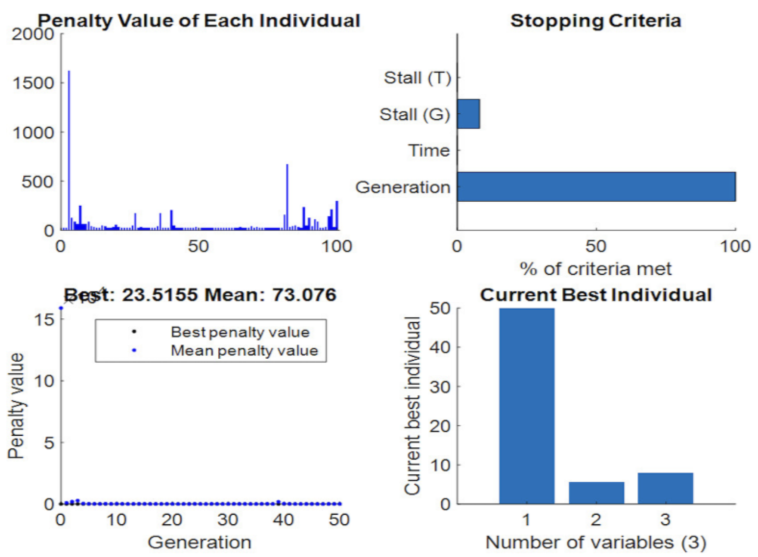

Figure 6. The result of the GA of PID_ITAE to optimize the three parameters of PID

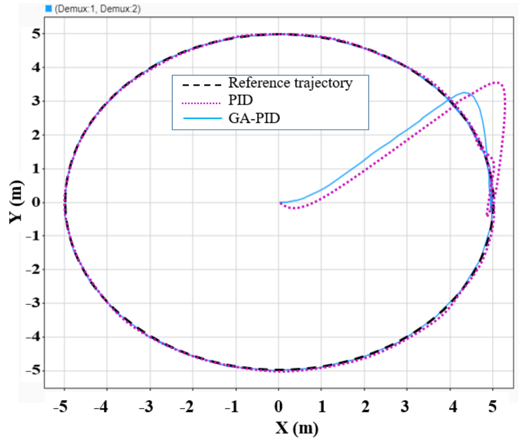

Figure 7. The robot's trajectory moves in a circle by the PID and GA-PID controllers

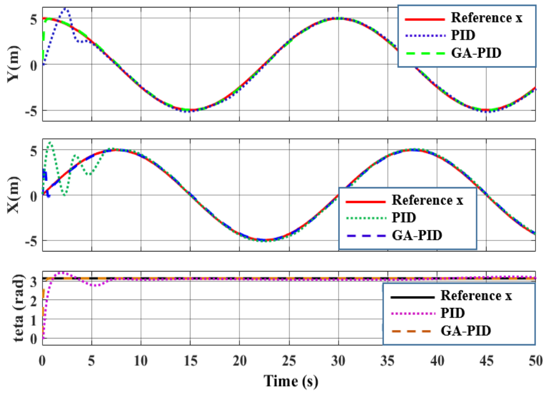

Figure 8. Trajectory in x, y, and θ for PID and GA-PID controllers

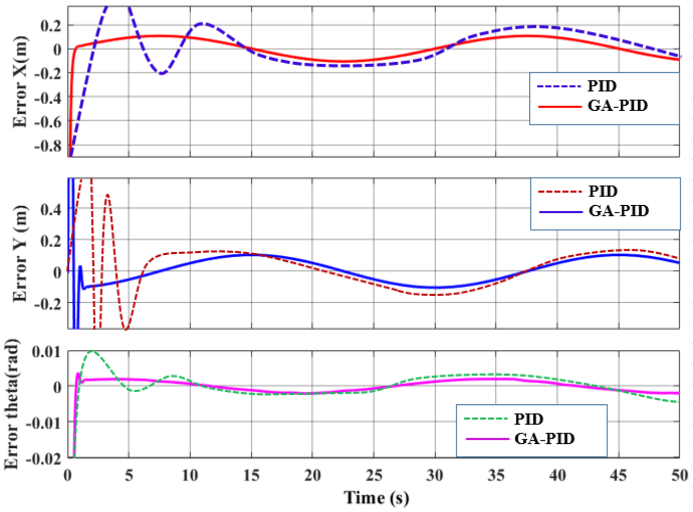

Figure 9. Trajectory deviation in x, y, and θ for PID and GA-PID controllers

Based on the results of Figures 8 and 9, it is evident that both the PID and GA-PID controller enable the robot to track the reference trajectory at 5 s. However, the PID controller initially exhibits a 10% overshoot and a slow setup time of 5 s. In contrast, the GA-PID controller shows minimal adjustments and a rapid setup time of 0.1 s. Integrating GA optimization notably enhances the PID controller's performance.

The GA-PID controller swiftly adapts and stabilizes the robot's trajectory, effectively reducing overshoot and setup time. These findings underscore the efficacy of GA-based optimization techniques in improving autonomous system control, especially in scenarios requiring precision, speed, and stability.

Case 2: simulation for a mobile robot moving along a square reference trajectory from the starting point with coordinates (10,0,0). The result of the GA of PID_ITAE to optimize the three parameters of PID is shown in Figure 10. The robot's trajectory moves in a circle, trajectory and trajectory deviation in x, y for robot is shown Figures 11 to 13.

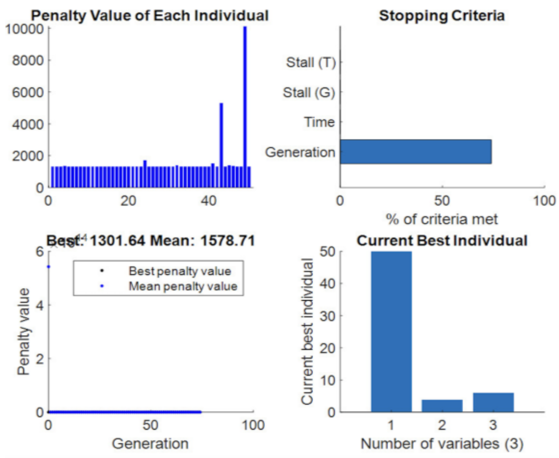

Figure 10. The result of the GA of PID_ITAE to optimize the three parameters of PID

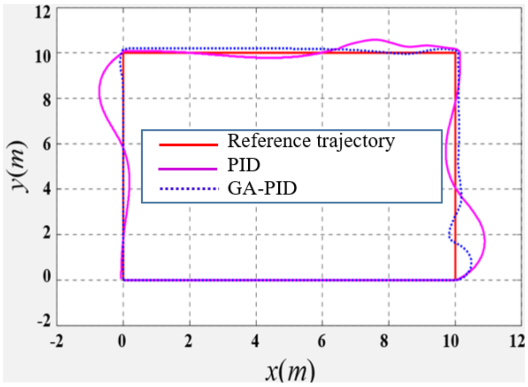

Figure 11. The robot's trajectory moves for PID and GA-PID controllers

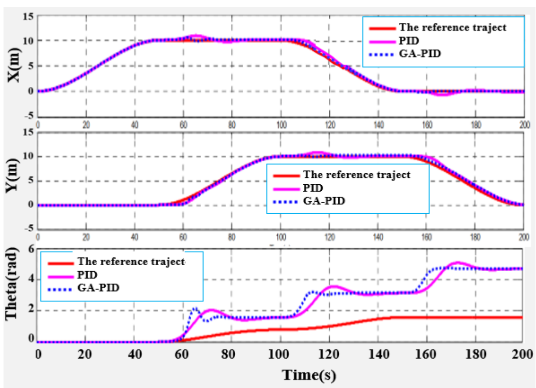

Figure 12. Trajectory in x, y, and θ for PID and GA-PID controllers

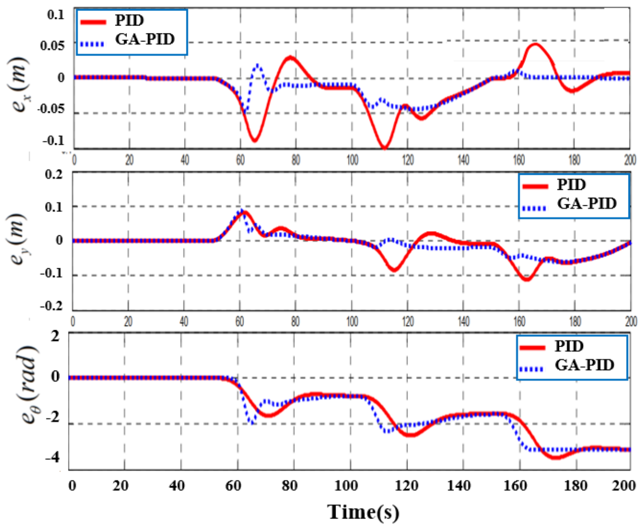

Figure 13. Trajectory deviation in x, y, and θ for PID and GA-PID controllers

Based on the results of figure, in Figures 12 and 13, it is seen that the robot motion trajectory of the GA-PID controller follows the reference trajectory better than the PID controller. Meanwhile, PID has the robot's motion trajectory with correction at transient times. The tracking performance of the robot with two PID and GA-PID controllers are shown in the Figure 14.

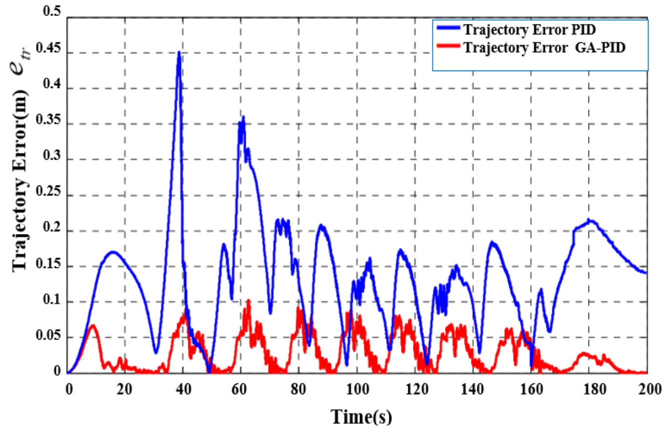

Figure 14. Comparison of orbital deviation between PID and GA-PID controllers

A better comparison between the tracking performance of the controller can be obtained by calculating the trajectory error based on (9):

$$e_{tr} = \sqrt{{e_x}^2 + {e_y}^2} = \sqrt{(x - x_r)^2 + (y - y_r)^2} \tag{8}$$

From (9) and simulation response as above, the error table of monitoring of two PID and GA-PID controllers for each trajectory as Table 1.

Table 1. Orbital tracking performance

Cases	Motion trajectory type	Trajectory error PID controller (m)	Trajectory error GA-PID controller (m)	Percentage of error improvement between 2 controllers (%)
1	Circle line	0.0976	0.0468	52.04
2	Square line	0.1243	0.0521	58.08

The tracking performance of the GA-PID controller compared to the PID controller is clearly improved (52% for circular orbit and 58% for square orbit), which means false orbital deviation will also decrease by this percentage results. The GA-PID controller achieves better tracking accuracy by optimizing the PID parameters through GAs, resulting in smoother and more precise control of the orbital path. This improvement in tracking performance directly translates to a reduction in false orbital deviation, leading to more accurate and reliable orbit control for various mission requirements.

5. CONCLUSION

The article presents the successful design of a GA-PID controller for trajectory tracking in a two-wheel mobile robot. This controller offers advantages such as zero-to-zero tracking error, quick achievement, and low bias. The GA-PID generates a smooth motion trajectory suitable for practical applications without requiring exact system dynamics model parameters. Its adaptive nature allows it to handle changes in robot kinematics, dynamics, reference trajectories, and external disturbances. Testing on a real self-propelled robot confirmed the controller's effectiveness in improving motion control for autonomous robots. Experimental results revealed superior trajectory tracking accuracy and robustness compared to traditional PID controllers, enabling precise tracking of complex trajectories with minimal error. The study highlights the potential of the GA-PID controller to enhance autonomous mobile robot performance in real-world scenarios, with ongoing research planned to explore its capabilities further in various environments and situations

REFERENCES

[1] A. De Luca, G. Oriolo, and M. Vendittelli, "Control of Wheeled Mobile Robots: An Experimental Overview," pp. 181–226, 2001, doi: 10.1007/3-540-45000-9_8.
[2] R. Hercik, R. Byrtus, R. Jaros, and J. Koziorek, "Implementation of Autonomous Mobile Robot in SmartFactory," *Applied Sciences*, vol. 12, no. 17, Sep. 2022, doi: 10.3390/app12178912.
[3] Z. Jian, Y. Bo, and P. Xiaofei, "An Optimal Controller for Trajectory Tracking of Automated Guided Vehicle," Feb. 2020, doi: 10.4271/2020-01-5024.
[4] Z. Fu, X. Feng, X. Duan, and Z. Fu, "An improved integrated navigation method based on RINS, GNSS and kinematics for port heavy-duty AGV," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 234, no. 8, pp. 2135–2153, Jul. 2020, doi: 10.1177/0954407019900031.
[5] D. Chen *et al.*, "Trajectory tracking control method and experiment of AGV," in *2016 IEEE 14th International Workshop on Advanced Motion Control (AMC)*, pp. 24–29, Apr. 2016, doi: 10.1109/AMC.2016.7496323.
[6] Q. Xu, C. Ai, D. Geng, G. Ren, and Z. Wang, "Research on Truck AGV Control System," in *2020 IEEE 8th International Conference on Computer Science and Network Technology (ICCSNT)*, pp. 176–180, Nov. 2020, doi: 10.1109/ICCSNT50940.2020.9305011.
[7] T. D. Tran, T. T. Nguyen, V. T. Duong, H. H. Nguyen, and T. T. Nguyen, "Parameter-Adaptive Event-Triggered Sliding Mode Control for a Mobile Robot," *Robotics*, vol. 11, no. 4, Aug. 2022, doi: 10.3390/robotics11040078.
[8] L. Yang and S. Pan, "A Sliding mode control method for trajectory tracking control of wheeled mobile robot," *Journal of Physics: Conference Series*, vol. 1074, no. 1, 2018, doi: 10.1088/1742-6596/1074/1/012059.
[9] B. Jiang, J. Li, and S. Yang, "An improved sliding mode approach for trajectory following control of nonholonomic mobile AGV," *Scientific Reports*, vol. 12, no. 1, Oct. 2022, doi: 10.1038/s41598-022-22697-w.
[10] L. Lin, P. Wu, B. He, Y. Chen, J. Zheng, and X. Peng, "The sliding mode control approach design for nonholonomic mobile robots based on non-negative piecewise predefined-time control law," *IET Control Theory & Applications*, vol. 15, no. 9, pp. 1286–1296, Jun. 2021, doi: 10.1049/cth2.12122.
[11] M. J. Rabbani and A. Y. Memon, "Trajectory Tracking and Stabilization of Nonholonomic Wheeled Mobile Robot Using Recursive Integral Backstepping Control," *Electronics*, vol. 10, no. 16, Aug. 2021, doi: 10.3390/electronics10161992.
[12] S. T. Dang, X. M. Dinh, T. D. Kim, H. L. Xuan, and M.-H. Ha, "Adaptive Backstepping Hierarchical Sliding Mode Control for 3-Wheeled Mobile Robots Based on RBF Neural Networks," *Electronics*, vol. 12, no. 11, May 2023, doi: 10.3390/electronics12112345.
[13] B. Ibari, "Backstepping controller with force estimator applied for mobile robot," *PRZEGLĄD ELEKTROTECHNICZNY*, vol. 1, no. 10, pp. 20–23, Oct. 2019, doi: 10.15199/48.2019.10.03.
[14] S. M. Ahmadi, M. B. Taghadosi, and A. H. M., "A state augmented adaptive backstepping control of wheeled mobile robots," *Transactions of the Institute of Measurement and Control*, vol. 43, no. 2, 2021.
[15] S. Shit, D. K. Das, and D. N. Ray, "Design and Development of a Terrain Adaptive Mobile Robot," in *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1–6, Jul. 2021, doi: 10.1109/ICCCNT51525.2021.9580075.
[16] P. Petrov, V. Georgieva, I. Kralov, and S. Nikolov, "An Adaptive Mobile Robot Control for Autonomous Following in Front of a Person," in *2021 12th National Conference with International Participation (ELECTRONICA)*, pp. 1–4, May 2021, doi: 10.1109/ELECTRONICA52725.2021.9513709.
[17] K. E. Dagher, R. A. Hameed, I. A. Ibrahim, and M. Razak, "An adaptive neural control methodology design for dynamics mobile robot," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 20, no. 2, Apr. 2022, doi: 10.12928/telkomnika.v20i2.20923.
[18] S. Harun and M. F. Ibrahim, "A genetic algorithm based task scheduling system for logistics service robots," *Bulletin of Electrical Engineering and Informatics*, vol. 8, no. 1, pp. 206–213, Mar. 2019, doi: 10.11591/eei.v8i1.1437.
[19] A. J. Moshayedi, A. Abbasi, L. Liao, and S. Li, "Path planning and trajectroy tracking of a mobile robot using bio-inspired optimization algorithms and PID control," in *2019 IEEE International Conference on Computational Intelligence and Virtual*

Environments for Measurement Systems and Applications (CIVEMSA), pp. 1–6, Jun. 2019, doi: 10.1109/CIVEMSA45640.2019.9071596.
[20] T. T. Thuong and V. T. Ha, "Adaptive Control for Mobile Robots Based on Inteligent Controller," *Journal of Applied Science and Engineering (Taiwan)*, vol. 27, no. 5, pp. 2481–2487, 2023, doi: 10.6180/jase.202405_27(05).0012.
[21] T. T. Thuong, V. T. Ha, and L. N. Truc, "Intelligent Control for Mobile Robots Based on Fuzzy Logic Controller," *Lecture Notes in Networks and Systems*, vol. 752 LNNS, pp. 566–573, 2023, doi: 10.1007/978-981-99-4725-6_67.
[22] V. T. Ha, T. T. Thuong, and V. T. Ha, "Experiment study of an automatic guided vehicle robot," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, vol. 14, no. 2, pp. 1300–1308, Jun. 2023, doi: 10.11591/ijpeds.v14.i2.pp1300-1308.
[23] H. W. Kim and S. Jung, "Control of a two-wheel robotic vehicle for personal transportation," *Robotica*, vol. 34, no. 5, pp. 1186–1208, May 2016, doi: 10.1017/S0263574714002173.
[24] Q. Li, A. C. Adriaansen, J. T. Udding, and A. Y. Pogromsky, "Design and Control of Automated Guided Vehicle Systems: A Case Study," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 13852–13857, Jan. 2011, doi: 10.3182/20110828-6-IT-1002.01232.
[25] H. Ren and C. Zhou, "Control System of Two-Wheel Self-Balancing Vehicle," *Journal of Shanghai Jiaotong University (Science)*, vol. 26, no. 5, pp. 713–721, Oct. 2021, doi: 10.1007/s12204-021-2361-x.
[26] A. G. Kallapur, I. G. Vladimirov, and I. R. Petersen, "Robust filtering for uncertain nonlinear systems satisfying a sum quadratic constraint," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 44, no. 1 PART 1, pp. 1–7, 2011, doi: 10.3182/20110828-6-IT-1002.01767.

BIOGRAPHIES OF AUTHORS

Vo Thanh Ha received her Ph.D. in Control and Automation Engineering from Hanoi University of Science and Technology, Vietnam, in 2020. She has been a lecturer at the University of Transport and Communications since 2005. Her research interests include electric drives, power electronics, control applications for traction drives, robots, autonomous vehicles, driving assistance systems for electric cars, and renewable energy. She can be contacted at email: vothanhha.ktd@utc.edu.vn.

Than Thi Thuong received a Control and Automation Engineering degree from the University of Economics-Technology for Industries, Vietnam in 2011, the Master's degree from University of Transport and Communications, Vietnam in 2016. She has worked in Faculty of Electrical Engineering, University of Economics-Technology for Industries since 2018, Vietnam. Than Thi Thuong′s research are robot control, process control, modeling and simulation. She can be contacted at email: ttthuong.dien@uneti.edu.vn.

Le Ngoc Truc majored in Control Engineering and Automation, and he received his B.Sc., M.Sc., and Ph.D. degrees from Hanoi University of Science and Technology (HUST), Vietnam, in 2006, 2010, and 2021, respectively. He has been working as a lecturer in Hung Yen University of Technology and Education since 2007. His researches are in fields of control engineering, robotics, system dynamics, and modeling. He can be contacted at email: lengoc.truc@utehy.edu.vn.