The Art of Navigation: Pure Pursuit Controller Strategies for Four-Wheeled Mobile Robots

Mohammed R. Hashim Al-Dahhan [*1], Mahmood Abdulrazzaq Alsaadi [2], Ruqayah R. Al-Dahhan [1], Salah A. Aliesawi [1]

[1]College of Computer Science and Information Technology; University of Anbar, Ramadi, Anbar, Iraq
[2]Computer Science Department, Al-Maarif University Collage, Ramadi, Anbar, Iraq

Emails: mohammed.rabeea@uoanbar.edu.iq ; alsaadi.m@uoa.edu.iq; ruqayah85@uoanbar.edu.iq; salah_eng1996@uoanbar.edu.iq
*Corresponding Author: mohammed.rabeea@uoanbar.edu.iq

Abstract

The Pure Pursuit Algorithm (PPA) is used in this paper to explain how a car with four wheels moves. The MATLAB environment has extensive simulation capabilities that can accurately represent complex robotic behaviors. It was these that were deployed for an extended analysis of the robot's operational dynamics. In the MATLAB/Simulink framework, waypoints obtained from different algorithms define robot trajectory. An odometer sensor helped to localize the robot thus giving accurate real-time information on its position. After critically evaluating several performance indices, it became clear just how well this control algorithm worked because it smoothly moved the robot from its initial state to its target with almost no oscillations at all. The findings of the simulation confirmed that if an appropriate lookahead distance is selected then the robot can effectively track waypoints and maintain optimal path along a trajectory up until reaching the target point at last

Keywords: quad-wheel mobile robot; Pure Pursuit Algorithm (PPA); lookahead distance

1. Introduction

An increasing attention over the last few years amongst researchers towards mobile robots that are wheeled. In general, there is a rapid scientific growth in the field of mobile robots ranging from legged to wheeled, underwater or aerial [1]. In addition, these mobile platforms are used in various industries such as medicine, customer care, agriculture, warehousing, and logistics industry for parcel delivery [2], [3], [4]. Mobile robots find use both outside industrial environments and within them. There are several advantages to using wheeled mobile robots for some applications like speed, precision, ease of performing repetitive and hard tasks among others. The two primary classes of the wheeled mobile robots are holonomic and non-holonomic types. The non-holonomic robots on wheels have only two degrees of freedom (2DOF). One of them allows linear motion along the x-axis while another enables rotation around the z-axis on a plane surface. On the other hand, holonomic wheeled robot has three degrees of freedom which makes it more flexible than its nonholonomic counterpart. In fact, Holonomic wheeled robots can only move linearly on x axis, linearly on y axis and rotate about z axis [5]. However, despite the advantages of Omni-wheels, there are a limitations like high cost, weight, reduced speed, and increased friction [6]. Therefore, in mobile robot applications non-holonomic wheels are preferred to holonomic wheels. Currently there is growing interest in differential drive wheeled mobile robots (DDWMRs) and their use across a wide range of application areas. DDWMRs have many benefits including flexible movement capabilities; simple but sturdy design; and low costs [7]. The configuration can be two wheels (2W) [8], three wheels (3W) [9], or 4-wheel [10] where two motorized wheels are present for each with additional passive castor wheels which allow free rotation around their own axis to provide stability and balance. Among these three-wheel DDWMRs, the most widely used

Doi: https://doi.org/10.54216/FPA.150207
Received: August 28, 2023 Revised: December 14, 2023 Accepted: March 24, 2024

in many industrial processes is due to its fine control and great maneuverability [11], [12]. They can work autonomously or follow point-to-point trajectories manually by introducing suitable control mechanisms into mobile robots [13]. Path tracking is done using Pure Pursuit Algorithm (PPA) [14]which is simpler to implement with better tracking results than other algorithms on this matter like [15], [16]. PPA implementation could take different forms such as proportional-integral-derivative (PID) control law whereby PID feedback controller is used as the main actuator in closed loop systems involving the dynamic model of robotic which has two driving wheels rotating about a fixed point on the ground [17]. There are several feedback control methods that can be applied in the case of manipulators such as linear quadratic regulator controller (LQR) [18] or model predictive controller (MPC) [19]. In direct feedback control, the wheel is mainly rotated based on system feedback while it does not consider the parameters of robot, models, or other available facts. For instance, in this category are PID control [20] and Fuzzy control [21]. This was further proposed to improve sensitivity of tracking by using genetic algorithm (GA), integrated with PID in [22], which eventually controlled disturbances from outside. Similarly, in reference [23], FLC was employed alongside predictive control (PDC) to enhance the resilience and resistance of the controller to time delays induced by sensors. The combination of FLC and Adaptive Neuro-Fuzzy in a hybrid algorithm provides insights with value among others in [24], [25]. The robot dynamics model is an established one that examines primarily how forces acting on the robot relate to dynamic parameters like velocity described by dynamics model-based-control. Methods including MPC control [26] and LQR control [27] belong to such control. Lately, in [28] it was done a model predictive controller-based interval path tracking control method for higher speed and better stability of AGV motion inside a working area beyond the yaw angle limit, while still not violating any given constraint. Moreover, [29] introduced an accelerated rolling optimization technique that built upon LQR controller for the mobile robot tracker. Both optimization of weight coefficient Q and R can enhance tracking the precision and effectiveness of the controller. Geometric tracking control (GTR) is responsible for calculating wheel rotation angles through measuring the geometric located between the reference path and the robot. For example, there are well-known PP control method [30], Stanley’s control method. The paper [31] described an improved PP control which adjusted the distance of pre-sight dynamically by selecting temporary pre-sight point to result in more accurate robot tracking.

The primary contribution of this study is how can we follow the generated waypoints by 4WD mobile robot with small oscillation and chose the accurate speed and lookahead angle for the controller in each position of movement. we next discuss the mobile robot system desertion, pure pursuit controller and results.

2. Mobile Robot System Description

Because of constraints related to available model parameters and computational for real-time applications, a simplified 3-degree-of-freedom (3DOF) vehicle body model is employed for the state estimation of the 4-wheel drive (4WD) vehicle. The vehicle body model depicted in Figure 1 encompasses three degrees of freedom: (longitudinal, lateral, yaw angle) displacement, and features four wheels.

Where:

x: The vehicle's lateral position within a global coordinate system, expressed in meters.
y: The vehicle's lateral position within a global coordinate system, expressed in meters.
θ: The orientation of the vehicle with respect to the global reference frame, expressed in radians.
ψ : The angle of steering for the vehicle, measured in radians.
L: The line between the front and rear axles of the vehicle, known as the wheelbase, measured in meters.
v: The velocity or speed of the vehicle, measured in meters per second.
The kinematic equation for the above model is:

Doi: https://doi.org/10.54216/FPA.150207
Received: August 28, 2023 Revised: December 14, 2023 Accepted: March 24, 2024

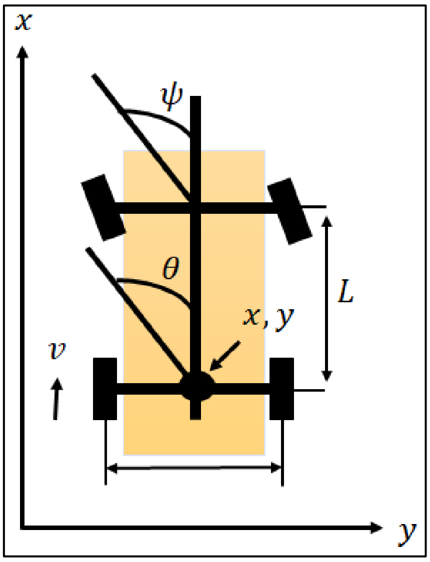

Figure 1: The 3DoF vehicle body model with four wheels.

$$\begin{bmatrix} x^2 \\ y' \\ \theta' \\ \psi' \end{bmatrix} = \begin{bmatrix} cos(\theta) & 0 \\ sin(\theta) & 0 \\ tan(\psi')/L & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \psi' \end{bmatrix} \quad (1)$$

Prior to establishing the estimation system, certain assumptions and simplifications are made in MATLAB to facilitate the process.

- Vehicle track: Dual.
- Ale forces: External longitudinal forces.
- In terms of longitudinal dynamics, the vehicle has two wheels on its front axle and two wheels on its rear axle. Its mass is about 2000 kg, and the longitudinal distance between the center-point of the mass and the front axle is (1.4m), while the distance between the center-point of the mass and the rear axle is (1.6m). Besides, the vertical distance between the center of mass and the axle plane stands at (0.35m).
- Moving to lateral dynamics, we find that the corner stiffness for front tires equals 12e3, while for rear ones, it amounts to 11e3; the relaxation length for a front tire measures 0.1 m, whereas the relaxation length for a rear tire is again 0.1 m.
- In terms of the aerodynamics of the vehicle, we have values such as Longitudinal drag area 2 m^2, relative wind angle vector [0:0.01:0.3] rad, and Side force vector [0:0.01:0.3].
- For the environmental factors, we consider absolute pressure at 101325 Pa, air temperature at 273 K, and the gravitational acceleration is set at ($9.8m/s^2$), accompanied by a nominal friction scaling factor of 1.

The above explanation illustrates the detailed indicators utilized in the MATLAB program for the purpose of designing a mobile robot, aiming to apply control over it in a more realistic manner.

3. PURE PURSUIT Controller

The Pure Pursuit Algorithm (PPA) stands out as a widely used path tracking algorithm for different types of mobile robots [32]. This algorithm computes both the linear velocity on x-axis (v) and angular velocity (ω) of the mobile robot based on its actual position and a set of waypoints. Illustrated in Figure 2 below.

Doi: https://doi.org/10.54216/FPA.150207
Received: August 28, 2023 Revised: December 14, 2023 Accepted: March 24, 2024

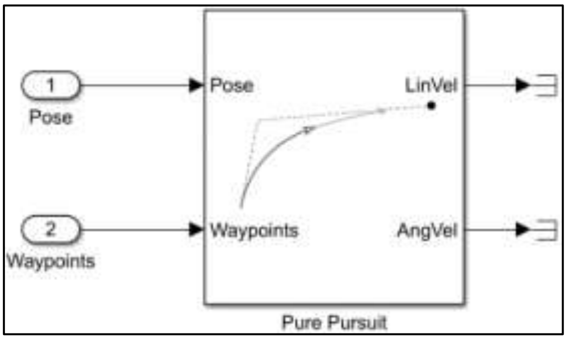

Figure 2: Pure pursuit block in MATLAB Simulink

The geometric interpretation to explain the PPA algorithms was inserted in figure 3 below:

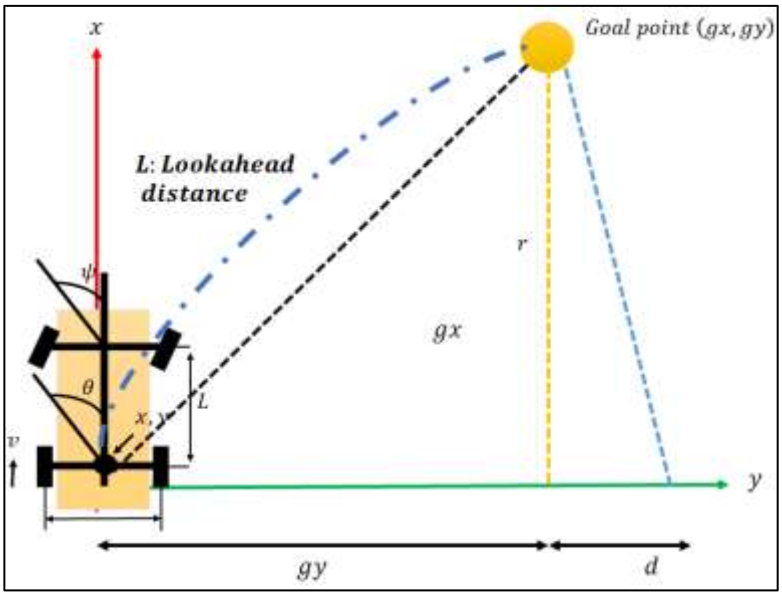

Figure 3: The geometric interpretation.

To calculate the required look ahead distance, some calculation should be made as below.

$$r = |gy| + d \quad (2)$$

$$d^2 + gx^2 = r^2 \quad (3)$$

$$(r - |gy|)^2 + gx^2 = r^2 \quad (4)$$

$$r^2 + |gy|^2 - 2r|gy| + gx^2 = r \quad (5)$$

$$r^2 + L^2 - 2r|9y| = r^2 \quad (6)$$

$$r = \frac{L^2}{2*|gy|} \quad (7)$$

$$\alpha = \frac{h}{r} = \frac{2|gy|}{L^2} \quad (8)$$

As it's clear from the calculation above steering angle should be proportional to the curvature of are. The PPA involves two inputs and three outputs. The robot's pose comprises its position on the x and y plane, representing the robot x and y coordinates, denoting its position and orientation comparative to the coordinate frame. This includes its distance from the origin along the x axis and angle with esteem to the x axis. The L.D is a critical factor in the algorithm's functioning. The look-ahead distance plays a significant role in influencing the algorithm's performance. Opting for a smaller look-ahead distance can enhance the robot's path tracking capabilities. However, inadequate tuning of this parameter may lead to system instability, necessitating careful adjustment. Hence, it's crucial to determine this value appropriately. Figure 4 illustrates the analysis of the look-ahead effect for various values.

Doi: https://doi.org/10.54216/FPA.150207
Received: August 28, 2023 Revised: December 14, 2023 Accepted: March 24, 2024

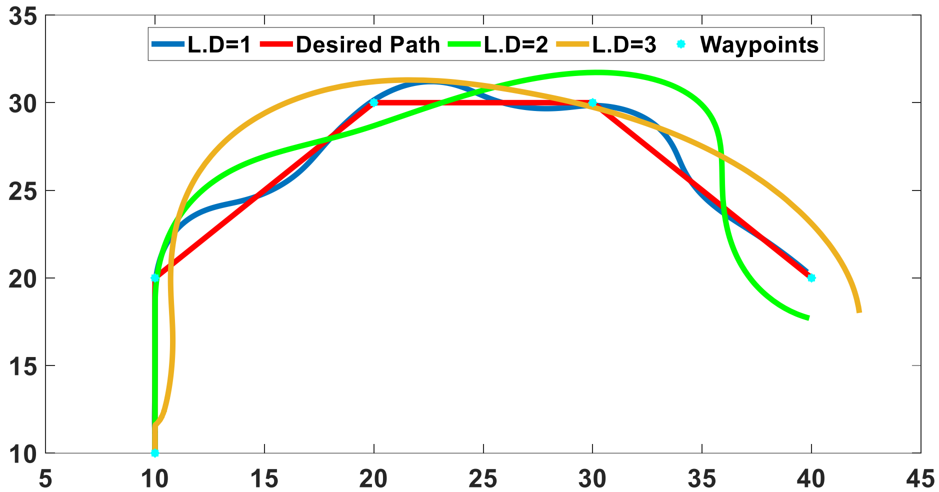

Figure 4: Different value for look ahead (L.D).

Increasing the look-ahead (L.D) distance typically results in smoother path tracking. However, it may lead to a deviation from the intended course before reaching the next waypoints. Conversely, reducing the L.D distance causes the robot to alter its course only after reaching the waypoints. This condition leads to undesired oscillations in the robot's trajectory, as depicted in Figure 4 above. The study exclusively relied on kinematic equations, neglecting dynamic conditions. Consequently, when the lookahead value is set to (L. D=1m), the robot's path exhibits increased oscillations and covers a longer distance, as highlighted by the blue line. Thus, it becomes imperative to determine the most suitable value for the Pure Pursuit Algorithm (PPA). However, when the lookahead is large (L. D=3m) as shown in Figure 4 above, the path generated by moving the robot tends to be somewhat further from the actual path, yet without oscillations. The controller's scenario can be condensed into the following points:

1. Identify the nearest waypoint to the vehicle.
2. Ascend toward that waypoint while computing the steering angle.
3. Utilize localization to determine the updated position.

Figure 5 illustrates the system's block diagram, where waypoints have been pre-selected on the map. The robot's actual position is determined by its odometer. Both the waypoints and the actual position feed into the pure pursuit controller to guide the robot towards its goal.

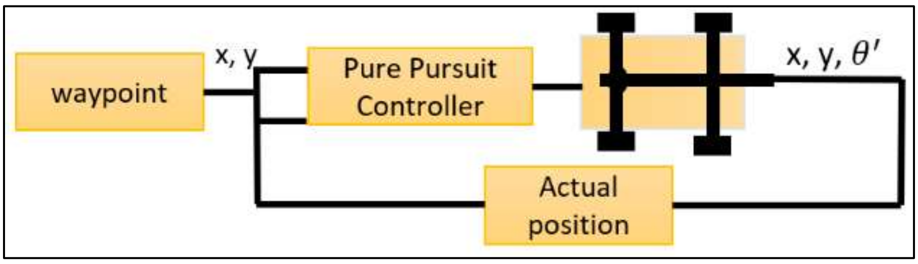

Figure 5: Overall system block diagram.

All robot movements adhere to its specifications, with linear velocity capped at 0.15 m/s and angular velocity within ±3 rad/s.

4. Results

After configuring the control settings and the mobile robot's settings, the control was tested by tracking some points on the map. Random points were selected on a specific map, and these points were then inputted into the control system. Afterwards, the control system began calculating the current location of the robot and the desired location, then started moving towards the goal. Each time, the control system considered the nearest point as the next goal. Upon reaching it, it compared the current location obtained from the odometry with the intended point. Once reached, it continued moving towards the next point, and so on until reaching the destination. Figure 6 shows an image of a random map, but specific maps for buildings can also be obtained via the camera. Afterwards, the image is processed and analyzed in MATLAB. The circular points in blue color represent the randomly selected waypoints on the map, which can be generated using any algorithm to find the path between two points. The yellow

Doi: https://doi.org/10.54216/FPA.150207
Received: August 28, 2023 Revised: December 14, 2023 Accepted: March 24, 2024

color indicates the shortest line between two points, representing the ideal path for the robot to follow. Finally, the red color illustrates the robot's movement from start to finish.
It is evident that at the beginning of the motion, the robot rotates to achieve the required angle to reach the second waypoint after rotation. Then, it starts moving according to the input points until reaching the goal. It is observed that the path does not contain many oscillations except at consecutive points with sharp angles. Overall, the control seems to have successfully guided the robot from start to finish with high accuracy and minimal oscillations, particularly between waypoints with sharp angles.

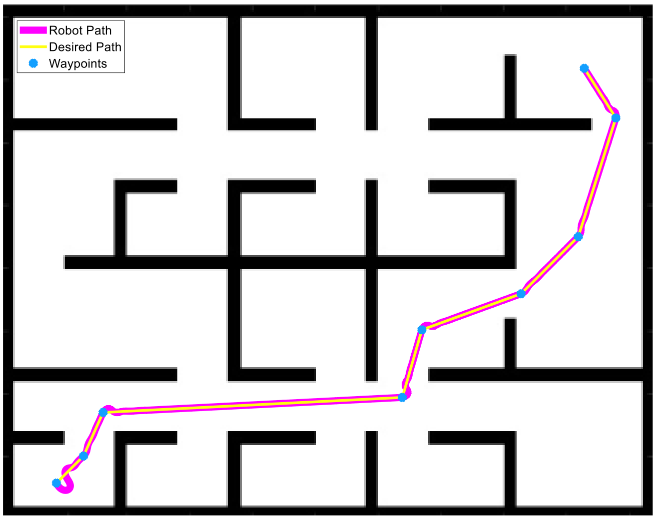

Figure 6: Maze map.

Figure 7 illustrates the data read from the odometry sensor, showing the actual movement of the robot from start to the end. The blue line represents the robot's movement on the y-axis, while the red line indicates the robot's movement on the x-axis. Therefore, according to the diagram, the starting point would be (36, 319), and the ending point would be (387, 43).

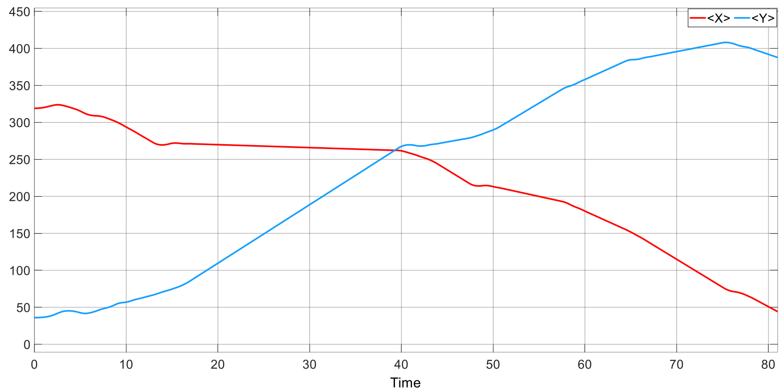

Figure 7: Odometry data on x and y axis.

To verify the settings and capabilities of the controller for moving the robot and placing it on the optimal path, the controller was tested again using a new map and random points on the map as well. Once again, the robot starts by rotating at the starting point, then begins moving towards the goal, with the controller almost aligning the robot on the ideal path along the route. Once again, the results confirm the controller's ability to achieve rotation angles and reach the goal with high efficiency. The second map shown clearly in figure 8 below.

Doi: https://doi.org/10.54216/FPA.150207
Received: August 28, 2023 Revised: December 14, 2023 Accepted: March 24, 2024

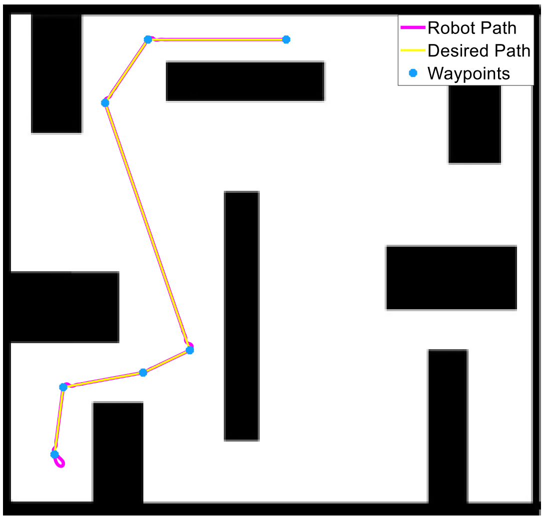

Figure 8: Map with different shape of obstacles.

To verify the settings and capabilities of the controller for moving the robot and placing it on the optimal path, the controller was tested again using a new map and random points on the map as well. Once again, the robot starts by rotating at the starting point, then begins moving towards the goal, with the controller almost aligning the robot on the ideal path along the route. Once again, the results confirm the controller's ability to achieve rotation angles and reach the goal with high efficiency.

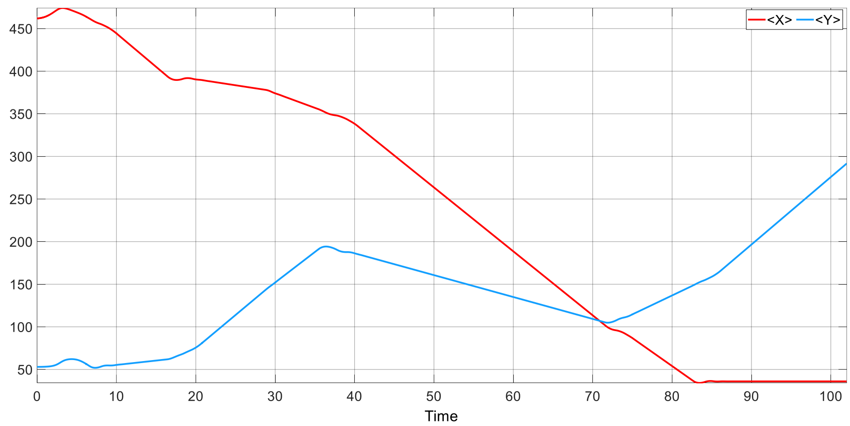

Figure 9: Odometry data on x and y axis.

6. Conclusion

This research demonstrated the implementation of PPA (Pure Pursuit Algorithm) control for a Differential Drive Wheeled Mobile Robot (DDWMR). Initially, the robot model was utilized to illustrate the impact of varying lookahead L.D distances on PPA performance. The findings revealed that small look-ahead values led to undesirable oscillations in the robot's trajectory. Subsequently, a comprehensive robot model was employed using MATLAB Simulink within the simulation environment. The robot's control was executed through PPA developed in the MATLAB environment. The outcomes from the comprehensive study indicated that adjusting the values of look-ahead affected the robot's approach to reference points, with no observed oscillations in motion. It can be inferred that higher L.D distances certain influenced the robot's travel duration, enabling the mobile robot to alter its trajectory without directly approaching the goal.

References

[1] F. V. and C. L.-A. F. Rubio, 'A review of mobile robots: Concepts, methods, theoretical framework, and applications', *Int. J. Adv. Robot. Syst.*, vol. 16, no. 2, 2019.

[2] T. and Y. Y. Namba, 'Risks of deep reinforcement learning applied to fall prevention assist by autonomous mobile robots in the hospital', *Big Data and Cognitive Computing*, vol. 2, no. 2, p. 13, 2018.

[3] G. and D. K. R. and S. F. and S. J. O. Fragapane, 'Planning and control of autonomous mobile robots for intralogistics: Literature review and research agenda', *Eur J Oper Res*, vol. 294, pp. 405–426, 2021.

Doi: https://doi.org/10.54216/FPA.150207
Received: August 28, 2023 Revised: December 14, 2023 Accepted: March 24, 2024

[4] X. and L. J. and F. L. and Z. Q. and Y. K. and W. J. and S. C. and H. L. and W. Z. Gao, 'Review of wheeled mobile robots' navigation problems and application prospects in agriculture', *IEEE access*, vol. 6, pp. 49248–49268, 2018.

[5] S. and G. G. and O. M. and P. A. and others Mellah, 'Actuator Health State Monitoring amp; Degradation Impact Study on a 4-Mecanum Wheeled Mobile Robot Behaviour', in *2021 29th Mediterranean Conference on Control and Automation (MED)*, 2021, pp. 1076–1081.

[6] R. R. and A.-D. M. R. H. and J. M. H. Al-Dahhan, 'Target seeking and obstacle avoidance of omni robot in an unknown environment', in *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, 2020, pp. 1–7.

[7] A. and V. P. T. and K. V. and P. K. L. Stefek, 'Energy comparison of controllers used for a differential drive wheeled mobile robot', *IEEE Access*, vol. 8, pp. 170915–170927, 2020.

[8] R. P. M. and S. K. A. and H. C. R. Chan, 'Review of modelling and control of two-wheeled robots', *Annu Rev Control*, vol. 37, no. 1, pp. 89–103, 2013.

[9] M. R. H. , A.-D. R. R. , & R. A. T. Al-Dahhan, 'Multi-layer perceptron neural network mobile robot navigator in unknown environment', *Indonesian Journal of Electrical Engineering and Computer Science* , vol. 31, pp. 725–733, 2023.

[10] M. and B. D. W. and M. N. A. Begnini, 'A robust adaptive fuzzy variable structure tracking control for the wheeled mobile robot: Simulation and experimental results', *Control Eng Pract*, vol. 64, pp. 27–43, 2017.

[11] P. and G. V. Petrov, 'Adaptive Velocity Control for a Differential Drive Mobile Robot', in *2018 20th International Symposium on Electrical Apparatus and Technologies (SIELA)*, 2018, pp. 1–4.

[12] M. R. H. and A. M. M. Al-Dahhan, 'Path tracking control of a mobile robot using fuzzy logic', in *2016 13th International Multi-Conference on Systems, Signals \& Devices (SSD)*, 2016, pp. 82–88.

[13] R. and L. Y. and F. J. and W. T. and C. X. Wang, 'A Novel Pure Pursuit Algorithm for Autonomous Vehicles Based on Salp Swarm Algorithm and Velocity Controller', *IEEE Access*, vol. 8, pp. 166525–166540, 2020.

[14] J. and M. J. Morales, 'Pure-Pursuit Reactive Path Tracking for Nonholonomic Mobile Robots with a 2D Laser Scanner', *EURASIP J Adv Signal Process*, pp. 1–10, 2009.

[15] Z. and B. Y. and W. J. and W. X. Wang, 'Vehicle path-tracking linear-time-varying model predictive control controller parameter selection considering central process unit computational load', *J Dyn Syst Meas Control*, vol. 141, no. 5, 2019.

[16] Q. and T. Y. Yao, 'A Model Predictive Controller with Longitudinal Speed Compensation for Autonomous Vehicle Path Tracking', *Applied sciences*, vol. 9, no. 22, p. 4739, 2019.

[17] R. and S. S. and N. M. Marino, 'Nested PID steering control for lane keeping in autonomous vehicles', *Control Eng Pract*, vol. 19, no. 12, pp. 1459–1467, 2011.

[18] A. and M. M. and D. V. and P. E. N. Bemporad, 'The explicit linear quadratic regulator for constrained systems', *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.

[19] G. V. and G. G. K. and N.-R. J. E. and K. C. R. and B. L. B. Raffo, 'A Predictive Controller for Autonomous Vehicle Path Tracking', *IEEE transactions on intelligent transportation systems*, vol. 10, no. 1, pp. 92–102, 2009.

[20] S. and Y. X. and L. P. and Z. M. and W. X. Wang, 'Trajectory tracking control for mobile robots using reinforcement learning and PID', *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 44, pp. 1059–1068, 2020.

[21] A.-T. and T. T. and E. L. and C. V. and P. R. and S. M. Nguyen, 'Fuzzy control systems: Past, present and future', *IEEE Comput Intell Mag*, vol. 14, no. 1, pp. 56–68, 2019.

[22] A. and W. Q. Alouache, 'Genetic algorithms for trajectory tracking of mobile robot based on PID controller', in *IEEE 14th International Conference on Intelligent Computer Communication and Processing (ICCP)*, 2018, pp. 237–241.

[23] J. and Y. M. and X. Z. Xianhua, 'Predictive fuzzy logic controller for trajectory tracking of a mobile robot', *IEEE Mid-Summer Wor shop on Soft Computing in Industrial Applications*, 2005.

[24] T. V. and H. N.-T. and V. N.-C. and K. V. N. and H. S.-C. Nguyen, 'Optimizing compliant gripper mechanism design by employing an effective bi-algorithm: Fuzzy logic and ANFIS', *Microsystem Technologies*, vol. 27, no. 9, pp. 3389–3412, 2021.

[25] N.-T. and N. T. V. and T. N. T. and N. Q.-M. Huynh, 'Optimizing magnification ratio for the flexible hinge displacement amplifier mechanism design', in *Proceedings of the 2nd Annual International Conference on Material, Machines and Methods for Sustainable Development*, 2021, pp. 769–778.

[26] S. and L. L. and G. H.-Q. and C. Z.-G. and S. P. Cheng, 'collision avoidance and lateral stability adaptive control system based on MPC of autonomous vehicles', *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2376–2385, 2019.

[27] C. and L. M. and P. F. Xu, 'The system design and LQR control of a two-wheels self-balancing mobile robot', *International Conference on Electrical and Control Engineering*, pp. 2786–2789, 2011.

Doi: https://doi.org/10.54216/FPA.150207
Received: August 28, 2023 Revised: December 14, 2023 Accepted: March 24, 2024

[28] H. and Z. J. Xu, 'Interval trajectory tracking for AGV based on MPC', in *Chinese Control Conference*, 2019, pp. 2835–2839.

[29] J. and W. Y. and L. H. and D. H. Ni, 'Path tracking motion control method of tracked robot based on improved LQR control', in *41st Chinese Control Conference (CCC)*, 2022, pp. 2888–2893.

[30] W.-J. and H. T.-M. and W. T.-S. Wang, 'The improved pure pursuit algorithm for autonomous driving advanced system', in *IEEE 10th international workshop on computational intelligence and applications (IWCIA)*, 2017, pp. 33–38.

[31] S. and Z. W. and M. L. and B. L. and J. C. and J. T. Qinpeng, 'Path tracking control of wheeled mobile robot based on improved pure pursuit algorithm', in *Chinese Automation Congress (CAC)*, 2019, pp. 4239–4244.

[32] R. C. and others Coulter, 'Implementation of the pure pursuit path tracking algorithm', *Carnegie Mellon University, The Robotics Institute*, 1992.

Doi: https://doi.org/10.54216/FPA.150207
Received: August 28, 2023 Revised: December 14, 2023 Accepted: March 24, 2024