Experimental Study on Fuzzy PD Control for Logistics Transportation Mobile Robot

Quoc-Khai Tran(✉), Chan Thanh Nguyen Huu, and Cong-Thanh Pham

Faculty of Electrical-Electronic Engineering, VAA University,
Ho Chi Minh City 700000, Vietnam
{khaitq,thanhnhc,thanhpc}@vaa.edu.vn

Abstract. This paper presents the method of designing a line following Fuzzy-PD controller for a differential-driven logistics transport mobile robot (LTMR). The line following controller is in charge of assisting the robot in moving along a predetermined path that has been installed or painted on the ground. The experimental results in this study demonstrate that the Fuzzy-PD controller can adjust the control parameters to better adapt to a rapid change in direction of the reference line when compared to using the PD controller. Furthermore, this paper provides an innovative approach to designing the control algorithm for differential-driven LTMR to enable a simple transition from manual to line following mode or vice versa.

Keywords: PD controller · Fuzzy-PD controller · Differential-driven logistics transportation mobile robots · manual mode · line following mode

1 Introduction

In general, regardless of sensor type, the line following control method uses position sensor information installed on the robot body to drive the robot such that the sensor's midpoint is as close to the reference line as possible [1–10]. The research results in these studies [7–10] have shown the effectiveness and ease of implementation of conventional Proportional-Integral-Derivative (PID) control technique for line following robot. Balaji et al. [7] also presented the roles of proportional, integral and derivative terms of PID controller in line following control. For example, with a large value of proportional gain - K_p, the robot is able to response quickly when the reference line changes its direction but oscillates strongly in the straight lines. With a small value of K_p, in contrast, the robot runs very stably on the straight lines but its ability to change the direction is much worse. Next, the derivative term of PID controller is known for its overshoot reduction function and transient response improvement related to the sudden change in direction of the path. However, given the large value of the derivative gain - K_d, the system can become quite sensitive to disturbing signals related to the quality of the sensor or of the reference line. The system also becomes unstable if the K_d is too large.

N.-S. Vo and H.-A. Tran (Eds.): INISCOM 2023, LNICST 531, pp. 225–235, 2023.
https://doi.org/10.1007/978-3-031-47359-3_17

In basic PID-controlled systems the integral term is used to eliminate steady error provided that the reference value is fixed or little changed. However, in line following control system, the characteristic of the predefined line often changes, thus the reference value for the line following controller is not fixed. With the presence of the integral term in PID controller the robot's performance can become unstable because of the instability of reference value for whole system. So that, to increase the stability of the control system, the integral term should be removed. The PD controller should be implemented with the K_p that can be adjusted, while the K_d should be selected appropriately and unchanged.

As an intelligent controller, fuzzy logic controller (FLC) is known with its ability to supervise the nonlinear systems and to be used to adjust the PID parameters to help the system achieve optimal state [11–13] . For the above reasons, this paper proposes a fuzzy PD controller with the ability to change K_p parameter by itself for line following robots to accommodate the variation of characteristics of the entire system.

In this study, the line following control mode can conventionally be called automatic control mode (ACM). Although the robot is built for the ACM, it may not always work in this manner. When the sensor system or magnetic line is damaged, or the robot has to operate in locations where there is no install line, the operator must control the robot directly. Manual control mode (MCM) refers to the control mode in which the robot is directly instructed by the operator. As a result, for research, an easy technique to transition from MCM to ACM or vice versa is required. There are commonly two techniques in the MCM for controlling the robot moving forward or backward, turning left or right using reference linear and angular velocity values. The first technique converts its reference values into two distinct signals, each of which is utilized to individually regulate the velocity of each wheel using its own PID controller with proper settings [14]. In the second technique, the robot velocities are directly controlled by two PID controllers, linear and angular PID controllers, without any reference value conversion [15]. Because the first technique makes the transition between MCM and ACM more difficult. Therefore, the second technique is used in this study to make programming and control easier. The second technique enables the system to go from MCM to ACM by simply replacing the angular velocity PID controller with the following PID controller, resulting in a control structure that is not significantly different from the original.

The rest of this paper is organized as follows. The MCM is built in Sect. 2. The method of designing the ACM is explained in Sect. 3. The experimental results and discussion are given in Sect. 4. Finally, the conclusion is presented in Sect. 5.

2 Manual Control Mode

2.1 Model of Logistics Transportation Mobile Robot

To prevent the wheels from rolling onto the magnetic line during the line following test as well as the actual run in the case when the magnetic line is not

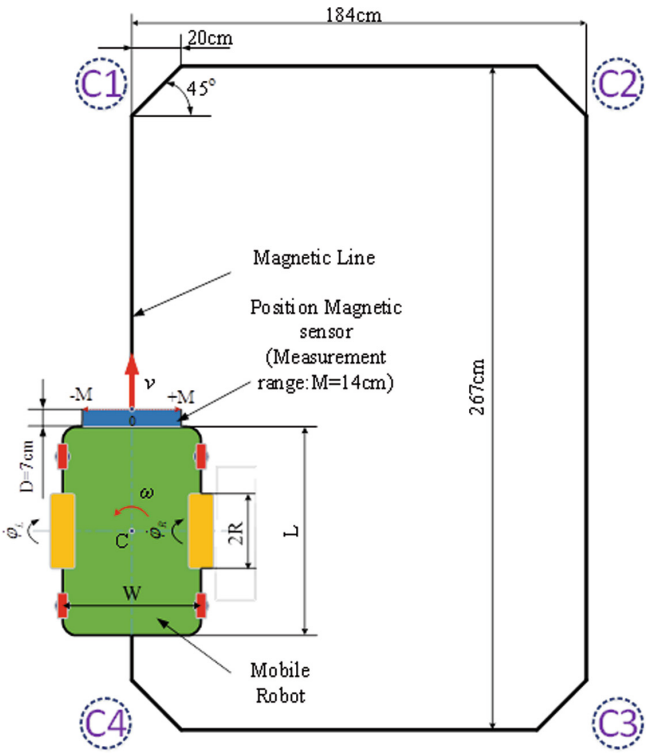

Fig. 1. Model of LTMR and magnetic line.

buried deep in the ground, the robot model with differential drive, as shown in Fig. 1, is used.

Two separate coaxial driving wheels with a 2R diameter are used to move the W×L-sized robot. Each of these wheels will be propelled by a BLDC motor that is managed by a motor drive. The velocities at which the left and right wheels rotate are $\dot{\varphi}_R$ and $\dot{\varphi}_L$, respectively. The placement of four castor wheels close to the rear of the robot body aids in improved robot balance and prevents the robot from rolling on the magnetic line and destroying it since the two driving wheels are positioned directly on the robot body's horizontal axis of symmetry. A magnetic sensor is attached at the head of the robot body for use while carrying out the line following function.

2.2 Velocity Control

Typically, the rotating velocities $\dot{\varphi}_R$ and $\dot{\varphi}_L$ are calculated based on signals returned from the encoder mounted directly on the motor shaft. The robot's

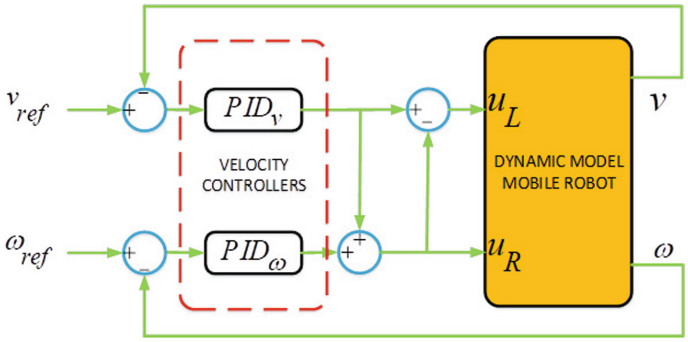

Fig. 2. Velocity control of LTMR.

angular and linear velocities may be calculated as

$$\begin{cases} v = \frac{v_R + v_L}{2} = \frac{R}{2}(\dot{\varphi}_R + \dot{\varphi}_L) \\ \omega = \frac{v_R - v_L}{2L} = \frac{R}{2L}(\dot{\varphi}_R - \dot{\varphi}_L) \end{cases}. \tag{1}$$

In the field of cargo transportation, it is not necessary to control a mobile robot at the highest speed but the robot needs to run at an appropriate and constant velocity even when the load of the freight is changed. The PID controller of linear velocity, denoted as PID_v, and the PID controller of angular velocity, denoted as PID_ω, can both regulate the robot's speeds. It is possible to acquire the output control signals u_v of PID_v and u_ω of PID_ω by

$$\begin{bmatrix} u_v \\ u_\omega \end{bmatrix} = \begin{bmatrix} K_{pv}e_v + K_{iv}\int_0^t e_v d\tau + K_{dv}\frac{d}{dt}e_v \\ K_{p\omega}e_\omega + K_{i\omega}\int_0^t e_\omega d\tau + K_{d\omega}\frac{d}{dt}e_\omega \end{bmatrix}, \tag{2}$$

Here, $[K_{pv}, K_{iv}, K_{dv}]$ and $[K_{p\omega}, K_{i\omega}, K_{d\omega}]$ are, respectively, parameters of PID_v and PID_ω.

The left and right motor drivers receive control voltage signals directly, which can be computed as

$$\begin{bmatrix} u_L \\ u_R \end{bmatrix} = \begin{bmatrix} u_v - u_\omega \\ u_v + u_\omega \end{bmatrix}. \tag{3}$$

Based on Eqs. (1) and (2), the velocity control scheme illustrated in Fig. 2.

3 Automatic Control Mode

In the ACM, the line following PD controller replaces the PID controller of angular velocity. Thus, Eq. (2) can be written as

$$\begin{bmatrix} u_v \\ u_\varepsilon \end{bmatrix} = \begin{bmatrix} K_{pv}e_v + K_{iv}\int_0^t e_v d\tau + K_{dv}\frac{d}{dt}e_v \\ K_{p\varepsilon}e_\varepsilon + K_{d\varepsilon}\frac{d}{dt}e_\varepsilon \end{bmatrix}, \tag{4}$$

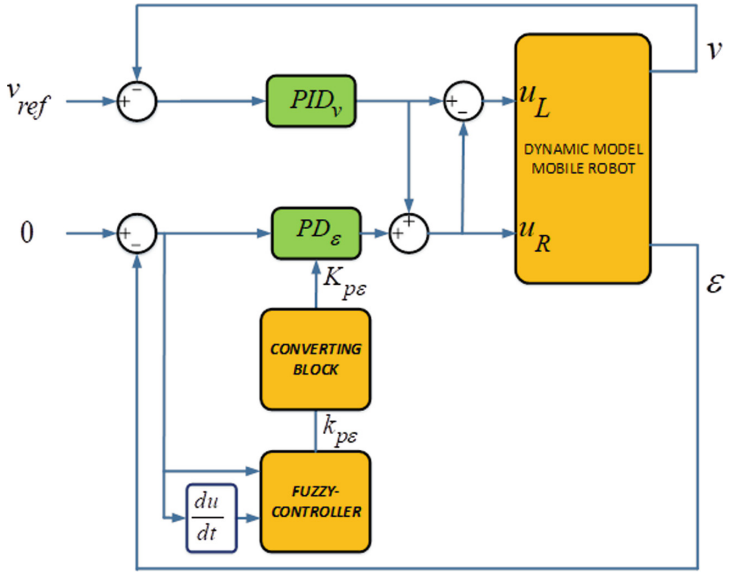

Fig. 3. Automatic control mode.

Here, $K_{p\varepsilon}$ and $K_{d\varepsilon}$ are parameters of the line following PD controllers. The line position and position error are represented by ε and $e_\varepsilon = -\varepsilon$, respectively.

Therefore, Eq. (3) can be written as

$$\begin{bmatrix} u_L \\ u_R \end{bmatrix} = \begin{bmatrix} u_v - u_\varepsilon \\ u_v + u_\varepsilon \end{bmatrix}, \tag{5}$$

As mentioned above, to adapt to the changes of the system, the Sugeno fuzzy controller known with its efficiency in computational processing [16] is used for adjusting the $K_{p\varepsilon}$ parameter, as shown in Fig. (3).

The fuzzy controller has two inputs: position error ε and position error change de_ε. The fuzzy controller's output signal $k_{p\varepsilon} \in [0, 1]$ is used to calculate $K_{p\varepsilon} \in [K_{p\varepsilon Min}, K_{p\varepsilon Max}]$ according to the following conversion formula.

$$K_{p\varepsilon} = (K_{p\varepsilon Max} - K_{p\varepsilon Min})k_{p\varepsilon} + K_{p\varepsilon Min}, \tag{6}$$

Here, $K_{p\varepsilon Min}$ and $K_{p\varepsilon Max}$ can be found by experimental testing explained in Sect. 4.

The fuzzy table rule is composed of nine rules, as stated in Table 1, in which $\mathbf{E} = \{\text{NE, ZE, PE}\}$, $\mathbf{DE} = \{\text{NDE, ZDE, PDE}\}$, and $\mathbf{KP} = \{\text{Z, M, B}\}$ represent the sets of linguistic variables of fuzzy inputs and output. Each rule is built as following:

Rule_n : if e_ε is $e_{\varepsilon i}$ and de_ε is $de_{\varepsilon j}$ then $k_{p\varepsilon}$ is $C_{i,j}$,

Where, $n = 9$ denotes the number of fuzzy control rules, and $C_{i,j}$ denotes the value of the cell corresponding to i^{th} row and j^{th} column in Table 1.

The membership functions of fuzzy inputs can be seen as in Figs. 4 and 5.

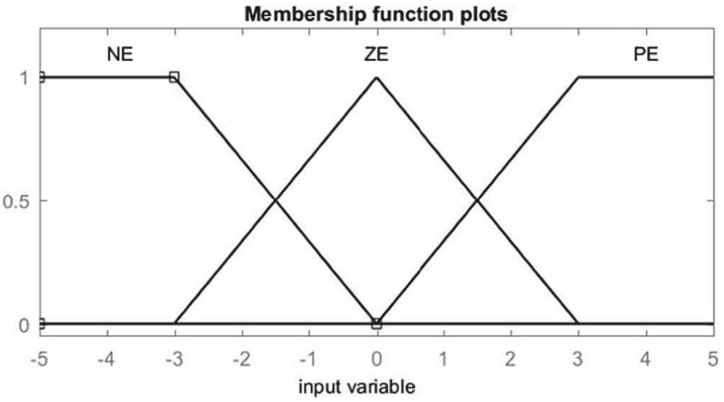

Fig. 4. Membership of position error.

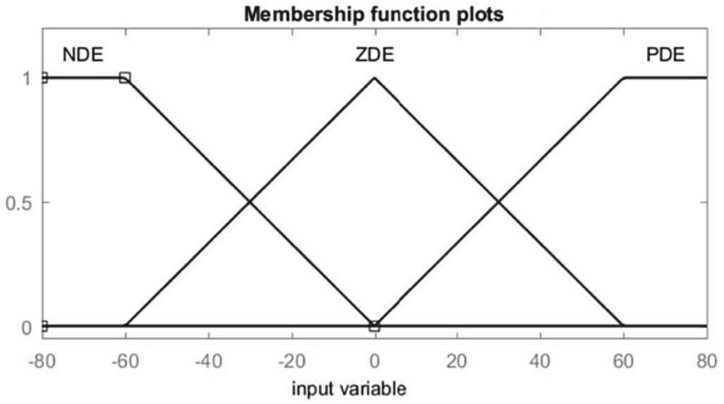

Fig. 5. Membership of change in error.

4 Experimental Results and Discussion

The LTMR created in our laboratory is shown in Fig. 6. The robot body's W and length L are 0.56 m and 0.84 m, respectively. Each wheel has a 2R diameter of 0.3 m. While not carrying things, the robot weighs 125 kg. For the robot's ACM, a magnetic position sensor with a measuring range M of 14 cm and a measurement resolution of 0.01 cm is utilized.

The velocities of the left and right wheel are measured based on digital signals generated by Hall effect sensors mounted inside two brushless DC motors (BLDC motor), and satisfy the constraint max $(v_L, v_R) \leq 0.94$ m (the maximum corresponding speed of the wheels is 60 rpm).

The robot's velocities are controlled by two PID controllers, which was programmed directly on the main control board of the robot. In our experimental

Table 1. Fuzzy rule table.

$k_{p\varepsilon}$	de_ε		
e_ε	NDE	ZDE	PDE
NE	*B*	*B*	*M*
ZE	*M*	*Z*	*M*
PE	*M*	*B*	*B*

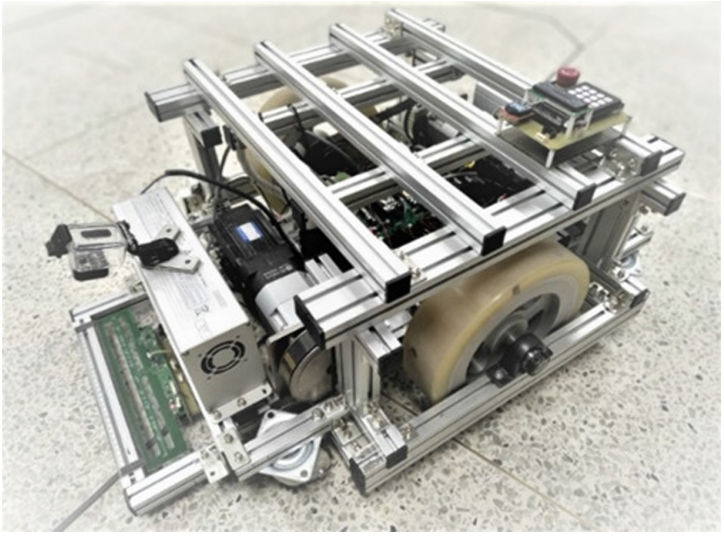

Fig. 6. Logistics transportation mobile robot.

research, the parameters of PID_v and PID_ω were set with [0.2, 0.4, 0.15] and [1.0, 0.4, 0.15], respectively.

The PID controller for angular velocity was replaced in the ACM by the line following Fuzzy-PD controller. The $K_{p\varepsilon Min}$, $K_{p\varepsilon Max}$ and $K_{d\varepsilon}$ can be determined based on the test method when the robot ran on closed magnetic line as shown in Fig. 1 as follows.

1. Increase $K_{p\varepsilon}$ gradually until a value considered as $K_{p\varepsilon Min}$ so that the robot can flow the straight line. However, the robot is still unable to follow the line when the direction of the line changes at an angle of 45° from the original direction.
2. Increase $K_{d\varepsilon}$ until the robot can follow the closed line entirely. $K_{d\varepsilon}$ can be chosen larger but just enough so the robot can change direction faster.

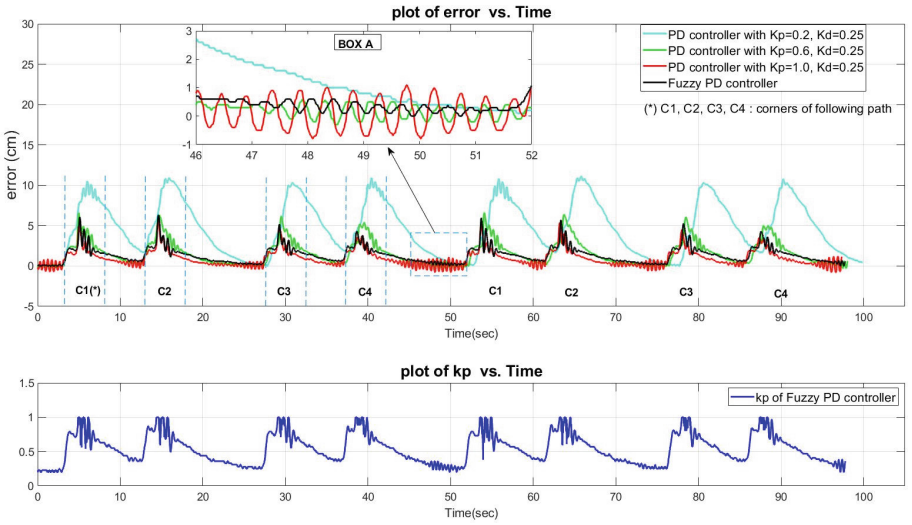

Fig. 7. Position errors of different PD controllers at 0.157 m/s.

3. Keep $K_{d\varepsilon}$ value unchanged, increase $K_{p\varepsilon}$ until the value considered as $K_{p\varepsilon Max}$ so that the robot oscillates quite strongly on the straight lines but still ensure the robot can follow the closed line.

With the values of $K_{p\varepsilon}$ and $K_{d\varepsilon}$ found, we can create a table with three following PD controllers as shown in Table 2. In fuzzy controller, the values of variables of fuzzy inputs and output were chosen as $\mathbf{E} = \{\text{NDE, ZDE, PDE}\} = \{-3, 0, 3\}$, $\mathbf{DE} = \{\text{NE, ZE, PE}\} = \{-60, 0, 60\}$ and $\mathbf{KP} = \{\text{Z, M, B}\} = \{0, 0.5, 1\}$.

Table 2. Following PD controllers.

PD controller	$K_{p\varepsilon}$	$K_{d\varepsilon}$
PD1 ($K_{p\varepsilon Min}$, $K_{d\varepsilon}$)	0.2	0.25
PD2 (($K_{p\varepsilon Min} + K_{p\varepsilon \mathrm{Max}}$)/2, $K_{d\varepsilon}$)	0.6	0.25
PD3 ($K_{p\varepsilon Max}$, $K_{d\varepsilon}$)	1.0	0.25

The robots with different PD controllers were tested by running two rounds (8 corners) at a speed of 0.157 m/s (corresponding to wheel speed of 10 rpm). Figure 7 shows the position errors ε of each PD controller and the change in $K_{p\varepsilon}$ when using fuzzy PD. Figure 8 shows the stability of linear velocity and the change over time of angular velocity when the robot changes its direction.

From the results presented in Fig. 7, we can draw conclusions as shown in Table 3. Based on the results of the maximum position errors and root-mean-square errors (RMSEs) in Table 3, it can be clearly seen that the larger $K_{p\varepsilon}$, the

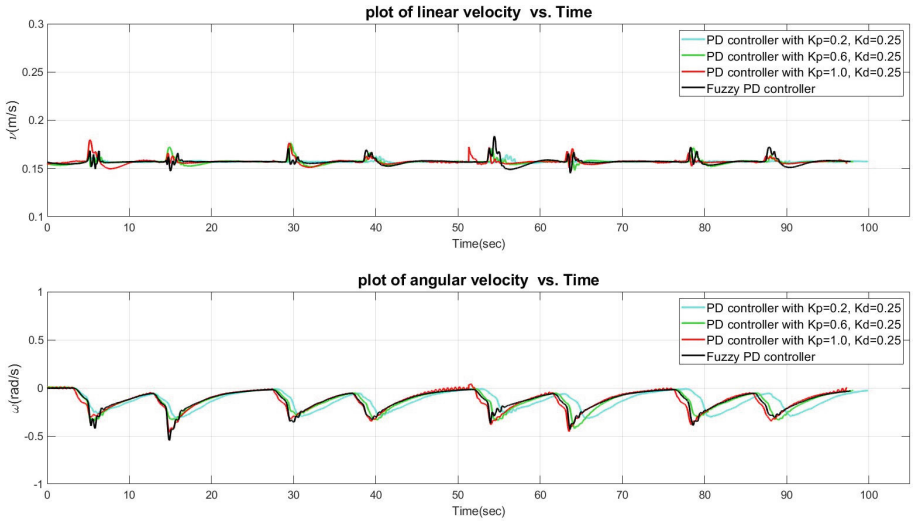

Fig. 8. Linear and angular velocities of the LTMR at 0.157 m/s.

better the robot follows the line. However, it has a disadvantage of making the system oscillate strongly on straight lines (maximal peak-to-peak error ripple is greatest when $K_{p\varepsilon} = K_{p\varepsilon Max}$). While with the smallest value of $K_{p\varepsilon}$, the system will become most stable, but the ability to follow the line becomes worse at the corners.

By comparing with the three PD controllers given above, the Fuzzy-PD controller has advantages such as not only in reducing ripple error but also in helping the robot to change direction quickly. With the low RMSE, comparing the Fuzzy-PD controller with the PD1 and PD2 controllers, it also shows that the use of Fuzzy-PD controller is a great good solution for line following.

Table 3. Experimental results at 0.157 m/s.

PD controller	MPPER in box A	MPE	AMCPE	RMSE	CT
PD1	0.1 cm	11.1 cm at 65.76 s	10.7 cm	5.70 cm	99.85 s
PD2	0.8 cm	6.6 cm at 54.08 s	6.08 cm	2.18 cm	98.07 s
PD3	1.8 cm	5.6 cm at 14.62 s	4.72 cm	1.39 cm	97.33 s
Fuzzy-PD	0.5 cm	6.3 cm at 14.66 s	5.39 cm	1.72 cm	97.77 s

MPPER: Maximal peak-to-peak error ripple
MPE: Maximal position error
AMCPE: Average of maximal corner position error
CT: Completion time

5 Conclusion

In this paper, a velocity and line following controllers were successfully built for the line following mobile robot. Switching from manual control mode to following control mode (automatic control mode) is also easy but still helps the robot keep the appropriate linear velocity even when the load changes. The use of PD controllers with fixed parameters shows that the following system cannot fully meet the operating requirements when environmental conditions change. The Fuzzy-PD following controller has the ability to change its own parameters, making the robot stable when running on straight lines but still ensuring good line following when there is a sudden change in the trajectory of predefined reference line.

Acknowledgement. The authors would like to thank the Vietnam Aviation Academy for Science and Technology Development for the support in 2022/2023.

References

1. Oltean, S.-E.: Mobile robot platform with arduino uno and raspberry PI for autonomous navigation. Proc. Manufact. **32**, 572–577 (2019)
2. Serrano Pèrez, E., Juàrez Lòpez, F., F.: An ultra-low cost line follower robot as educational tool for teaching programming and circuit's foundations. Comput. Appl. Eng. Educ. **27**(2), 288–302 (2019)
3. Pisarov, J.: Experience with mBot-wheeled mobile robot. Proc. XXXV. Jubileumi Kandó Konferencia 2019 (JKK2019), 47–51 (2019)
4. Saadatmand, S., Azizi, S., Kavousi, M., Wunsch, D.: Autonomous control of a line follower robot using a q-learning controller. In: 10th Annual Computing and Communication Workshop and Conference (CCWC), pp. 0556–0561, IEEE (2020)
5. Eleftheriou, G., Doitsidis, L., Zinonos, Z., Chatzichristofis, S.A.: A fuzzy rule-based control system for fast line-following robots. In: 2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS), pp. 388–395, IEEE (2020)
6. Oswal, S., Saravanakumar, D.: Line following robots on factory floors: significance and simulation study using coppeliasim. In: IOP Conference Series: Materials Science and Engineering, vol. 1012, no. 1. IOP Publishing, 2021, p. 012008 (2021)
7. Balaji, V., Balaji, M., Chandrasekaran, M., Elamvazuthi, I., et al.: Optimization of PID control for high speed line tracking robots. Proc. Comput. Sci. **76**, 147–154 (2015)
8. Kader, M.A., Islam, M.Z., Al Rafi, J., Islam, M.R., Hossain, F.S.: Line following autonomous office assistant robot with PID algorithm. In: 2018 International Conference on Innovations in Science, Engineering and Technology (ICISET), pp. 109–114, IEEE (2018)
9. Maàrif, A., Nuryono, A.A.: Vision-based line following robot in webots. In: 2020 FORTEI-International Conference on Electrical Engineering (FORTEI-ICEE), pp. 24–28, IEEE (2020)
10. Farkh, R., Aljaloud, K.: Vision navigation based PID control for line tracking robot. Intell. Autom. Soft Comput. **35**(1), 901–911 (2023)
11. Eltag, K., Aslamx, M.S., Ullah, R.: Dynamic stability enhancement using fuzzy PID control technology for power system. Int. J. Control Autom. Syst. **17**, 234–242 (2019)

12. Somwanshi, D., Bundele, M., Kumar, G., Parashar, G.: Comparison of fuzzy-PID and PID controller for speed control of dc motor using labview. Proc. Comput. Sci. **152**, 252–260 (2019)
13. Chao, C.-T., Sutarna, N., Chiou, J.-S., Wang, C.-J.: An optimal fuzzy PID controller design based on conventional PID control and nonlinear factors. Appl. Sci. **9**(6), 1224 (2019)
14. Saleh, A.L., Mohammed, M.J., Kadhim, A.S., Raadthy, H.M., Mohammed, H.J.: Design fuzzy neural petri net controller for trajectory tracking control of mobile robot. Int. J. Eng. Technol. **7**(4), 2256–2262 (2018)
15. Khai, T.Q., Ryoo, Y.-J., Gill, W.-R., Im, D.-Y.: Design of kinematic controller based on parameter tuning by fuzzy inference system for trajectory tracking of differential-drive mobile robot. Int. J. Fuzzy Syst. **22**, 1972–1978 (2020)
16. Topaloğlu, F., Pehlıvan, H.: Comparison of mamdani type and sugeno type fuzzy inference systems in wind power plant installations. In: 6th International Symposium on Digital Forensic and Security (ISDFS), pp. 1–4, IEEE (2018)