Intelligent Control for Mobile Robots Based on Fuzzy Logic Controller

Than Thi Thuong[1], Vo Thanh Ha[2](✉), and Le Ngoc Truc[3]

[1] Faculty of Electrical Engineering, University of Economics - Technology for Industries, Hai Bà Trưng, Vietnam
ttthuog.dien@uneti.edu.vn

[2] Faculty of Electrical and Electronic Engineering, University of Transport and Communications, Hanoi, Vietnam
vothanhha.ktd@utc.edu.vn

[3] Hung Yen University of Technology and Education, Hải Dương, Vietnam

Abstract. This paper recommends intelligent control for mobile robots based on fuzzy logic controllers (FLC). This controller is designed with only two input state variables, such as position error, position deviation derivative of the robot, and one output variable, velocity. The robot is moved according to the trajectories set by fuzzy selection rules with an 9×9 matrix. The proposed FLC controller is compared with classical PID controller. The robot with the FLC controller moves to follow the trajectory with lower error and faster setup time than the PID controller. The efficiency of this controller is demonstrated by MATLAB/Simulink.

Keywords: Mobile Robot · PID · Fuzzy Logic Control · FLC

1 Introduction

The mobile robot is an innovative solution for the future of digitization and industry 4.0. The self-propelled robot ensures the certainty and flexibility of the product. At the same time, it makes it easier to move goods inside factories and warehouses. Besides, robots also improve automation and solve production continuity problems [1, 2]. In the world, in recent decades, autonomous robot control has received extensive research and development attention, and many methods, from classical control to modern management, have been proposed to apply to self-propelled robots. Previously, most publications used the structure of two control loops as the outer kinematic loop uses the Lyapunov function to synthesize the position-tracking controller, and the dynamic inner circle controls the speed tracking. Many active loop control methods have been proposed, such as slip control [3–6] and backstepping control [7–9]. When the dynamic equation has uncertain parameters, adaptive management is included in the design [10–13]. The adaptive control combines with neurons to approximate the result. Unpredictable parts [14–16] and adaptive control combined with fuzzy logic [17–20] gave reasonable control quality, compensating for model error and system input noise.

T. D. L. Nguyen et al. (Eds.): ICISN 2023, LNNS 752, pp. 566–573, 2023.
https://doi.org/10.1007/978-981-99-4725-6_67

Although many advanced controllers have been researched and developed, traditional PID controllers are still chosen to be used in the problem of controlling orbital self-propelled robots because of their effectiveness of the controller. This ensures stability and traction. However, the accuracy achieved is not high. In the process of developing control techniques, Intelligent control, fuzzy logic is applied to work in many fields with the role of an observer. The fuzzy inference mechanism is considered a Simple and effective method for fine tuning classic controllers (Leonid Reznik, 1997; Jan Jantzen, 1998). Therefore, the sustainable controller FLC, when used to control the self-propelled robot, although it can.

This paper is organized into five main parts. Part 1 and part 2 present the introduction to the target study and kinematics and dynamics model. The fuzzy logic controller is designed in Sect. 3. Part 4 shows the simulation and simulation results. The last section is the conclusion.

2 Kinematic and Dynamic Model

2.1 Kinematic Model

The equation describing the kinematics of the mobile robot is expressed in Eq. (1) [1].

$$\dot{q} = \begin{bmatrix} \frac{r}{2}\cos(\theta) & \frac{r}{2}dcos(\theta) \\ \frac{r}{2}\sin(\theta) & \frac{r}{2}d\sin(\theta) \\ \frac{r}{2a} & \frac{r}{2a} \end{bmatrix} \begin{bmatrix} \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \tag{1}$$

where: r is right and left wheel radius; $2a$ is distance between the actuated wheels and the symmetry axis; $\dot{\varphi}_r$, $\dot{\varphi}_l$ are angular velocity of the right and left wheels; v, ω are Angular velocity of the right and left wheels; q is linear and angular velocities of robot; θ is orientation angle; $\dot{q}$ is robot speed.

2.2 Dynamic Model

The kinetic energy of the self-propelled robot is calculated by:

$$T_c = \frac{1}{2}m_c\vartheta_c^2 + \frac{1}{2}I_c\dot{\theta}^2 \tag{2}$$

$$T_{\omega R} = \frac{1}{2}m_\omega\vartheta_\omega^2 + \frac{1}{2}I_m\dot{\theta}^2 + \frac{1}{2}I_\omega\dot{\varphi}_r^2 \tag{3}$$

$$T_{\omega L} = \frac{1}{2}m_\omega\vartheta_\omega^2 + \frac{1}{2}I_m\dot{\theta}^2 + \frac{1}{2}I_\omega\dot{\varphi}_l^2 \tag{4}$$

where: where Tc is the kinetic energy of the DWMR without the wheels, $T_{\omega R}$ is the kinetic energy of the actuated wheels in the plane and $T_{\omega L}$ is the kinetic energy of all the wheels considering the orthogonal plane; m_c is mass of the robot without wheels and motors; m_ω is mass of each wheel and motor assembly; m_t is total mass of the DWMR; m_t is moment of inertia of the DWMR without wheels and motors about the vertical axis through P; I_c is moment of inertia of the DWMR without wheels and motors about the vertical axis through P; I_{w} is Moment of inertia of each wheel and motor about the wheel axis; I is Total inertia moment of the robot; $\dot{\varphi}_r$, $\dot{\varphi}_l$ are angular velocity of the right and left wheels; v, ω are Angular velocity of the right and left wheels; θ is orientation angle.

Mobile robot speed is calculated by:

$$\vartheta_i^2 = \dot{x}_i^2 + \dot{y}_i^2 \tag{5}$$

The coordinates of the wheels are therefore determined as follows:

$$\begin{cases} x_{\omega \mathrm{r}} = x + a \sin\theta \\ y_{\omega \mathrm{r}} = y + a \cos\theta \end{cases} \tag{6}$$

$$\begin{cases} x_{\omega l} = x - a \sin\theta \\ y_{\omega l} = y + a \cos\theta \end{cases} \tag{7}$$

From Eq. (2) to Eq. (7), the total kinetic energy:

$$T = \frac{1}{2} m_t \Big(\left(\dot{x}^2 + \dot{y}^2 \right) - \dot{y} d \dot{\theta} \cos(\theta) + m_t \dot{x} d \dot{\theta} \sin(\theta) \Big) + \frac{1}{2} I \dot{\theta}^2 + \frac{1}{2} I_w \left(\dot{\varphi}_r^2 + \dot{\varphi}_I^2 \right) \tag{8}$$

where: $m_t = m_c + 2m_\omega$; $I = m_c d^2 + I_c + 2m_\omega \left(d^2 + a^2\right) + 2I_m$ and $\dot{\theta} = \omega$

The robot's equation of motion is described by the system of equations:

$$\begin{cases} m\ddot{x} - m_c d \ddot{\theta} \sin\theta - m d \dot{\theta}^2 \cos\theta = F_1 - C_1 \\ m\ddot{y} - m_c d \ddot{\theta} \cos\theta - m_c d \dot{\theta}^2 \sin\theta = F_2 - C_2 \\ -m_c d \sin\theta \ddot{x} + m_c d \cos\theta \ddot{y} + I\ddot{\theta} = F_3 - C_3 \\ I_\omega \ddot{\varphi}_r = \tau_r - C_4 \\ I_\omega \ddot{\varphi}_l = \tau_l - C_5 \end{cases} \tag{9}$$

The matrix linking the kinematic constraints:

$$\Lambda^T(q) = \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \end{bmatrix} \tag{10}$$

From Eqs. (9)- (10) The motion of the robot can be represented by the equation:

$$M(q)\ddot{q} + V(q, \dot{q}) + F(\dot{q}) + G(q) + \tau_d = B(q)\tau - \Lambda^T(q)^\lambda \tag{11}$$

where: $M(q)$ is positive inertia matrix; $V(q, \dot{q})$ is centripetal Matrix; $F(\dot{q})$ is surface friction; $G(q)$ is gravity acceleration matrix; τ_d is noise component; $B(q)$ is input matrix; $\Lambda^T(q)$ is binding matrix; λ is Lagrange multiplier vector.

2.3 Kinematic Model

The kinematic error model q_e of a self-propelled robot is a mathematical equation describing the deviation of the robot's position and posture, when the motion-controlled robot follows a desired trajectory ξd. The system of error function equations as follows:

$$\dot{q}_e = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} \cos(\theta_e) & 0 \\ \sin(\theta_e) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \vartheta_r \\ \omega_r \end{bmatrix} + \begin{bmatrix} -1 & y_e \\ 0 & -x_e \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \vartheta \\ \omega \end{bmatrix} \tag{12}$$

3 Fuzzy Logic Controller Design

Use the inputs as bias and the difference derivative to configure the fuzzy logic controller, such as in Fig. 1 and Fig. 2. The transformed fuzzy block matches the data with the conditions of the given fuzzy rule. The output of the fuzzy set is converted to the clarity values through the centroid defuzzification method and converted into a control signal, as in Fig. 3. The FLC controller is controlled by rule table 1.

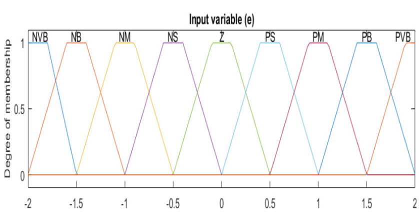

Fig. 1. Input of bias variable *e*

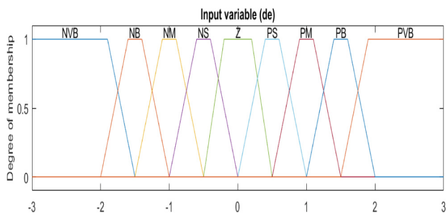

Fig. 2. Deviated variable derivative input *de*

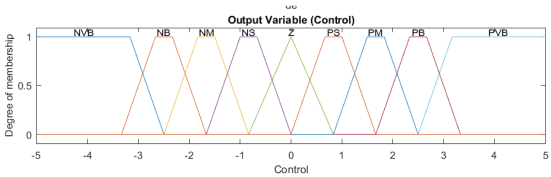

Fig. 3. Output variable

Table 1. The rule control for FLC controller

De/e	NVB	NB	NM	NS	Z	PS	PM	PB	PVB
NVB	PVB	PVB	PVB	PB	PM	PM	PS	Z	Z
NB	PVB	PVB	PB	PM	PS	PS	PS	Z	Z
NM	PVB	PB	PM	PS	PS	Z	Z	Z	NS
NS	PB	PM	PM	PS	PS	Z	Z	NS	NS
Z	PM	PM	PS	Z	Z	Z	NS	NS	NM
PS	PM	PS	PS	Z	NS	NS	NM	NM	NB
PM	PS	PS	Z	NS	NS	NM	NB	NB	NB
PB	PS	Z	Z	NS	NM	NM	NB	NVB	NVB
PVB	Z	Z	NS	NM	NM	NB	NB	NVB	NVB

4 Simulation Results on MATLAB/Simulink

The FLC controller is compared with PID controller. The parameters of the PID set are determined through the tuning simulation method on MATLAB/Simulink as $K_p = 0.7$; $K_I = 0.6$; $K_D = 0.01$.

Case 1: the trajectory is a circular orbit with radius 1, center is origin.

The results of the two controllers when the robot follows the same circular trajectory and the simulated response is shown in Figs. 4 and 5.

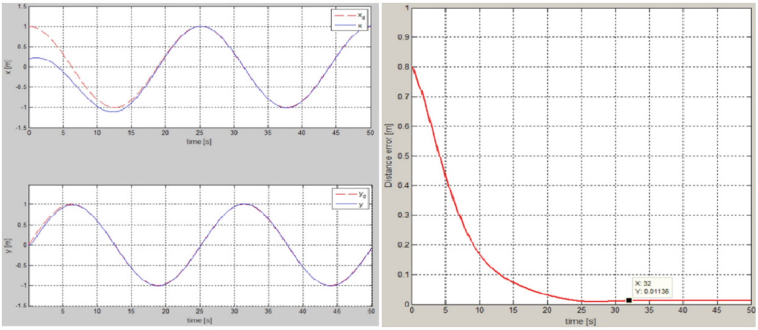

Fig. 4. x, y position and system error using PID controller

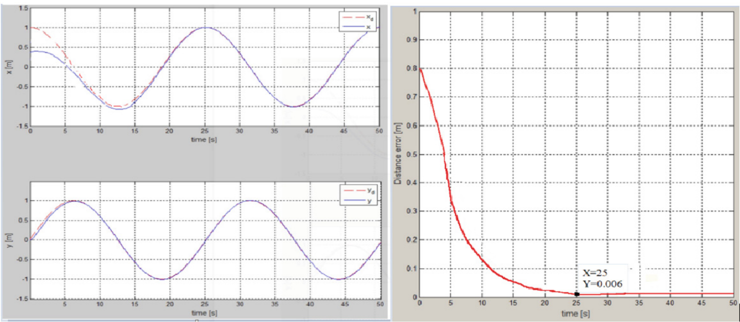

Fig. 5. X, Y position and system error using FLC controller

Based on the figure results in Figs. 4, 5, it is shown that both controllers respond to the stability of the moving robot following a preset trajectory. However, the robot position error is small for the PID controller (0.01), with a longer response time (32s). On the other hand, while the FLC controller system error is only 0.006, the response time is faster than the PID controller with 25s.

Case 2: The trajectory is simulated, which is the crackling trajectory.

The position response of the two controls is expressed in Figs. 6, 7. It is evident from the graph findings in Figs. 6, 7, that both controls react to the stability of a moving robot following a predetermined course. The PID controller, however, has a minor robot position error (0.01) and a slower reaction time (3s). However, even though the FLC controller's system error is just 0.006, it responds more quickly than the PID controller, which takes 2s.

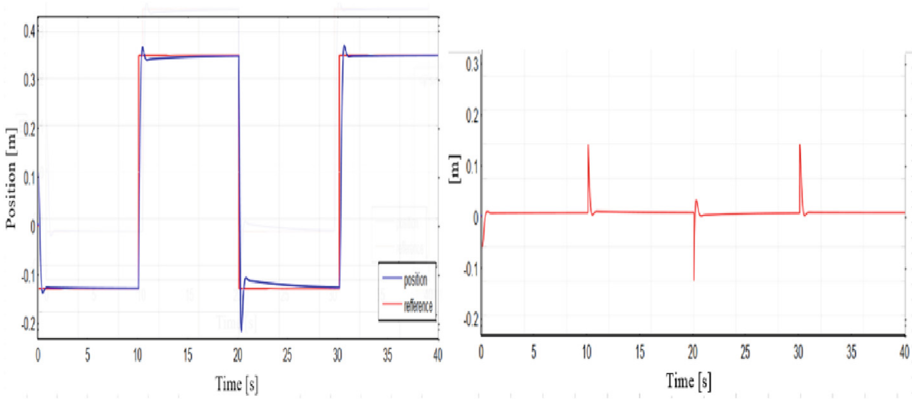

Fig. 6. Position and system error using traditional PID controller

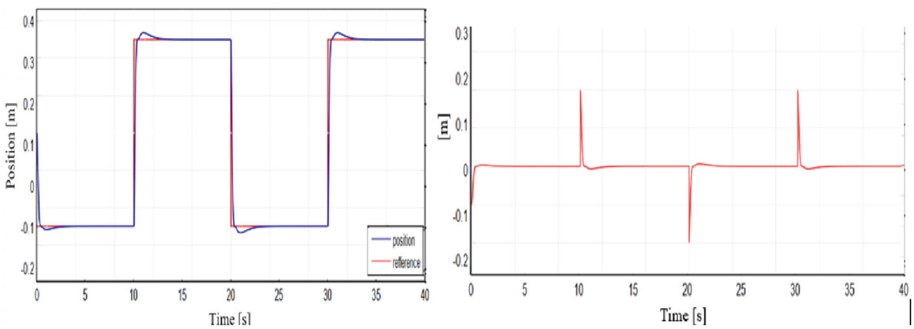

Fig. 7. Position and system error using FLC controller

5 Conclusion

The paper has proposed a kinematic and dynamic model for a mobile robot with a differential actuator based on the Lagrange dynamic approach. Mobile Robot is moved according to the trajectory set by PID and FCL controller. The FLC controller has the advantages of a simple design and better performance than the PID controller, with an orbital error of 0.006 and a setting time of 25s. However, to improve the moving quality of the mobile robot more accurately and faster, it is necessary to use intelligent control methods such as neural network, sliding mode controller with chattering controller, or hybrid controllers such as FLC controller combined with PID controller or sliding mode control connected with FLC controller.

References

1. Alexander, J.C., Maddocks, J.H.: On the kinematics of wheeled mobile robots. Int. J. Robot. Res. **8**(5), 15–27 (1989)
2. Barraquand, J., Latombe, J.: Nonholonomic multibody mobile robots: controllability and motion planning in the presence of obstacles. Algorithmica **10**(2), 121 (1993)
3. Campion, G., Bastin, G., d'Aandrea Novel, B.: Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. IEEE Trans. Robot. Autom. **12**(1), 47–62 (1996)
4. Maaref, H., Barret, C.: Sensor-based navigation of a mobile robot in an indoor environment. Robot. Auton. Syst. **38**, 1–18 (2002)
5. Thongchai, S., Suksakulchai, S., Wilkes, D.M., Sarkar. N.: Sonar behavior -based fuzzy control for a mobile robot"**.** In: Proceedings of the 2000 IEEE International Conference on Systems, Man and Cybernetics, vol. 5, pp. 3532–3537 (2000)
6. Kolmanovsky, I., Harris McClamroch, N.: Developments in nonholonomic control problems. IEEE Control Syst. **15**(6), 20–36 (1995)
7. Lewis, F.L., Dawson, D.M., Abdallah, C.T.: Robot Manipulator Control: Theory and Practice,2nd edn. Marcel Dekker, Inc. (2003)
8. Li, Y.D., Zhu, L., Sun, M.: Adaptive neural-network control of mobile robot formations including actuator dynamics. In: Sensors, Measurement and Intelligent Materials, volume 303 ofApplied Mechanics and Materials, pp. 1768–1773. Trans Tech Publications (2013)
9. Li, Y.D., Zhu, L., Sun, M.: Adaptive RBFNN formation control of multi-mobile robots with actuator dynamics. Indo. J. Electr. Eng. **11**(4), 1797–1806 (2013)
10. DeCarlo, R.A., Zak, S.H., Drakunov, S.V.: Variable structure, sliding mode controller design. Control Handb. **57**, 941–951 (1996)
11. Derks, E.P.P.A., Pastor, M.S.S., Buydens, L.M.C.: Robustness analysis of radial base function and multilayered feedforward neural network models. Chemometr. Intell. Lab. Syst. **28**(1), 49–60 (1995)
12. Freire, F., Martins, N., Splendor, F.: A simple optimization method for tuning the gains of PID controllers for the autopilot of Cessna 182 aircraft using model-in-the-loop platform. J. Control Autom. Electr. Syst. **29**, 441–450 (2018)
13. Gao, W., Hung, J.C.: Variable structure control of nonlinear systems: a new approach. IEEE Trans. Ind. Electron. **40**(1), 45–55 (1993)
14. Lewis, F.L., Jagannathan, S., Yesildirek, A.: Neural Network Control of Robot Manipulatorsand Nonlinear Systems. Taylor & Francis, Ltd., 1 Gunpowder Square, London, EC4A 3DE (1999)

15. Lewis, F.L., Dawson, D.M., Abdallah. C.T., Robot Manipulator Control: Theory and Practice,2 edn. Marcel Dekker, Inc. (2003)
16. Li, Y., Qiang, S., Zhuang, X., Kaynak, O.: Robust and adaptive backstepping control for nonlinear systems using RBF neural networks. IEEE Trans. Ne ral Netw. **15**(3), 693–701 (2004)
17. Keighobadi, J., Mohamadi, Y.: Fuzzy sliding mode control of nonholonomic wheeled mobilerobot. In: Proceedings of the 9th IEEE International Symposium on Applied Machine Intelligence and Informatics—SAMI'2011, pp. 273–278. IEEE (2011)
18. Begnini, M., Bertol, D., Martins, N.: A robust adaptive fuzzy variable structure tracking control for the wheeled mobile robot: simulation and experimental results. Control Eng. Pract. **64**, 27–43 (2017)
19. Begnini, M., Bertol, D., Martins, N.: Design of an adaptive fuzzy variable structure compensatorfor the nonholonomic mobile robot in trajectory tracking task. Control Cybern. **47**, 239–275 (2018)
20. Begnini, M., Bertol, D., Martins, N.: Practical implementation of an effective robust adaptivefuzzy variable structure tracking control for a wheeled mobile robot. J. Intell. Fuzzy Syst. **35**, 1087–1101 (2018)