An Adaptive Fuzzy Dynamic Surface Control Tracking Algorithm for Mecanum Wheeled Mobile Robot

Dong Nguyen Minh [1], Hiep Do Quang [1], Nam Dao Phuong [2], Tien Ngo Manh [3,*], and Duy Nam Bui [4]

[1] Faculty of Electrical, University of Economics-Technology for Industries, Hanoi, Vietnam;
Email: nmdong@uneti.edu.vn (D.N.M.), dqhiep@uneti.edu.vn (H.D.Q.)
[2] School of Electrical Engineering, Hanoi University of Science and University, Hanoi, Vietnam;
Email: nam.daophuong@hust.edu.vn (N.D.P.)
[3] Institute of Physics, Vietnam Academy of Science and Technology, Hanoi, Vietnam
[4] Faculty of Electrical, Hanoi National University, Hanoi, Vietnam; Email: duynam.robotics@gmail.com (D.N.B.)
*Correspondence: nmtien@iop.vast.vn (T.N.M.)

***Abstract*—This paper presents an Adaptive Fuzzy Dynamic Surface Control (AFDSC) algorithm for trajectory tracking of a four-Mecanum Wheeled Mobile Robot (MWMR). Firstly, the kinematics and dynamics equations for the robot mecanum can be constructed based on the Lagrange equation. Next, the proposed algorithm combines the advantages of dynamic surface control and fuzzy logic to enhance the tracking performance and robustness of the robot system. By incorporating a fuzzy logic system, the control algorithm is able to adaptively adjust the controller parameters based on the changing dynamics and uncertainties of the MWMR. The dynamic surface control technique effectively mitigates the "explosion of complexity" problem associated with traditional backstepping control methods, allowing for a simpler and more computationally efficient control architecture. The designed control algorithm is implemented and verified on a MWMR platform, and its performance is compared against other existing Dynamic Sliding Surface Control (DSC) methods. The simulation results demonstrated that the proposed adaptive fuzzy DSC algorithm achieves superior trajectory tracking accuracy, and improved convergence properties. The experimental results on Gazebo simulator further proved the feasibility of the proposed algorithm.**

***Keywords*—mecanum robot, dynamic surface control, fuzzy logic**

I. Introduction

Mecanum Wheeled Mobile Robots (MWMRs) have gained significant attention in recent years due to their unique omnidirectional capabilities, making them well-suited for various applications in fields such as warehouse automation, surveillance, and mobile manipulation [1–3]. Accurate trajectory tracking is a crucial requirement for MWMRs to perform tasks effectively in dynamic and uncertain environments [4–6]. Therefore, the development of advanced control algorithms that can ensure precise tracking performance has become a topic of extensive research.

Numerous control techniques have been proposed for trajectory tracking of MWMRs. Proportional-Integral-Derivative (PID) controllers are commonly used due to their simplicity and effectiveness [6–8]. For example, Thai *et al.* [7] designed a PID controller with time-varying parameters for trajectory tracking of a mecanum-wheeled robot with a minor error. However, PID controllers may struggle to handle nonlinear dynamics, leading to suboptimal tracking performance in complex environments.

To address the limitations of PID controllers, researchers have explored advanced control strategies [4, 5, 9–12]. Alakshendra, *et al.* [9] presented an Adaptive Robust Second-order Sliding Mode Control (ARSSMC) on a mobile robot with four mecanum wheels to deal with the tracking problem. In [10], an adaptive sliding mode controller is proposed for the omnidirectional mobile robot with four Mecanum wheels to track the trajectory with uncertainties. Sun *et al.* [11] proposed a novel method for the path-following problem of an MWOMR named robust Nonsingular Terminal Sliding Mode (NTSM) control. Yadav *et al.* [12], authors presented a trajectory-tracking four Mecanum wheeled wheelchair control approach based on a sliding mode control technique to control the trajectory of a three-degree-of-freedom MWC subject to a lot of noise and uncertainty. However, sliding mode control may introduce chattering phenomena, impacting the smoothness of the control inputs.

Fuzzy logic control has also been applied to MWMRs [13–16]. Cao *et al.* [13] combined the advantages of PID and fuzzy control, and proposed a method named fuzzy adaptive PID control method to enhance the accuracy and stability of the MWMR. Zou *et al.* [14], a Fuzzy-PID controller and its implementation on a four Mecanum-wheeled robot are presented. The Fuzzy logic is used as tuner of the PID controller coefficients to improve the

Manuscript received June 27, 2023; revised July 25, 2023; accepted August 18, 2023.

doi: 10.18178/ijmerr.12.6.354-361

performance. Elrehem *et al.* [15] presented an approach for the three-wheeled mobile robot mecanum in obstacle avoidance using fuzzy logic controller to achieve all desired motion directions and improve the performance of the mobile robot. The fuzzy logic controller demonstrated good robustness, but it lacked adaptive capabilities to adjust its parameters based on changing operating conditions.

Dynamic surface control [17–21] has been widely utilized in robotic control systems to mitigate the computational complexity associated with backstepping control methods. In [20], an adaptive stability controller based on dynamic surface technique is developed for uncertain nonlinear systems with external disturbances. With the proposed approach, the problem of "dimensional explosion" is avoided in the controller design. Wang *et al.* [21] deals with the trajectory tracking of a four-wheeled Mecanum omnidirectional mobile robot with full state constraints and input saturation based on a neural network based adaptive surface control. However, the application of dynamic surface control specifically to MWMRs for trajectory tracking remains relatively unexplored.

To the best of our knowledge, there is limited research on combining dynamic surface control and fuzzy logic for trajectory tracking of MWMRs. Our proposed algorithm aims to address this research by providing an adaptive fuzzy AFSC algorithm that offers improved tracking accuracy, robustness, and computational efficiency.

In this paper, we propose an Adaptive Fuzzy Dynamic Surface Control (AFDSC) algorithm for trajectory tracking of MWMRs. The algorithm leverages the advantages of dynamic surface control and fuzzy logic to improve the tracking accuracy and robustness of the robot system. By incorporating fuzzy logic, the algorithm can adaptively adjust controller parameters based on changing dynamics and uncertainties, enhancing the overall control performance. Additionally, the dynamic surface control technique mitigates the computational complexity associated with traditional backstepping control methods, offering a simpler and more efficient control architecture. Our contributions therefore are threefold: (i) we mathematical analysis both kinematic and dynamic models of the MWMR based on the Lagrange equations; (ii) an adaptive fuzzy dynamic surface control algorithm which is combined by advantages of dynamic surface control and fuzzy logic to apply to the MWMR for dealing with the trajectory tracking problem; (iii) a number of simulation and comparison are conducted to confirm the performance of the proposed control algorithm compared with the convenient dynamic surface control method.

The rest of this study is structured as follows. Section II present both kinematic and dynamic model of the MWMR. Section III describes the design of the proposed AFDSC algorithm. The simulation and comparison confirm the effectiveness of the proposed controller is detailly presented in Section IV. Finally, the conclusion draw the end of this paper is presented in Section V.

II. The Four Mecanum Wheel Mobile Robot

A. Mecanum Wheels

The Mecanum wheel is first proposed by the Mecamum AB Company of Switzerland, applied in four Omni-directional mobile robots, as depicted in Fig. 1. There are many small rollers distributed on the hub of each Mecanum wheels [22].

Figure 1. General view of the mecanum structure.

When the motor drives the wheel to rotate, the wheel runs conventionally along a direction perpendicular to the drive shaft, and at the same time, the rollers along the periphery of the wheel run freely about the respective ax. The rollers of the wheel are cylindrical forms, transforming a part of one wheel's steering force into its sliding force and then makes a synthesis of the respective speed and direction of each wheel, the resultant force ensures the platform's required freedom of movement in the any direction. We consider the wheel structure as shown in Fig. 2.

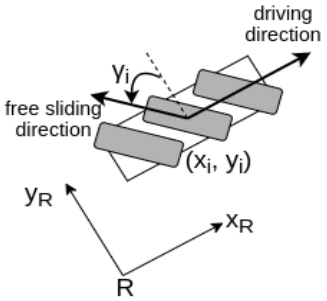


Figure 2. The mecanum wheel frame.

The wheel driving speed is obtained by the robot velocity follow the equation in below:

$$\omega_i = \frac{1}{r}\begin{bmatrix}1 & \tan\gamma\end{bmatrix}\begin{bmatrix}1 & 0 & -y_i \\ 0 & 1 & x_i\end{bmatrix}\boldsymbol{v} = \boldsymbol{h}_i\boldsymbol{v} \quad (1)$$

where:

r_i – wheel radius

v – robot velocity $(v_x, v_y, \omega)^T$; $h_i \in \mathbb{R}^{1\times 3}$

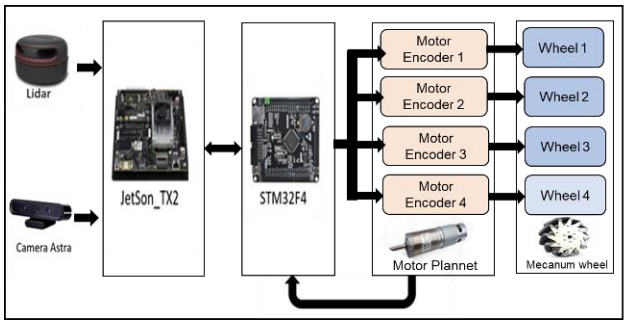
Assuming the robot has n mecanum wheels, the inverse kinematic matrix is shown as follow:

$$H' = \begin{bmatrix}h_i \\ \vdots \\ h_n\end{bmatrix} \in \mathbb{R}^{n\times 3} \quad (2)$$

B. Build Experimental Robot Model

According to Fig. 3, robot model is built with 3 parts: Central processing unit, sensor and actuator control. Regarding the structure of the moving part, there are 4 mecanum wheels that make moving more flexible and easier. Jetson TX2 embedded computer acts as a central processing unit with ROS operating system installed along with SLAM algorithm calculation nodes. After the calculation is complete, the Jetson TX2 sends a control command to the actuator controller, which is the STM32 circuit. RPLidar scanning 360° degree builds maps and detects obstacles. 3D Camera uses Deep Camera to help identify mid-range and low-range obstacles in front of the Robot.

(a)



(b)

Figure 3. Connection diagram mecanum robot controller. (a)The actual four-mecanum wheeled mobile robot; (b) The system diagram.

C. Kinematic Modeling

The robot model uses four mecanum wheels that are independently driven by 4 separate motors. The arrangement of the position of each wheel is done as shown in Fig. 3.

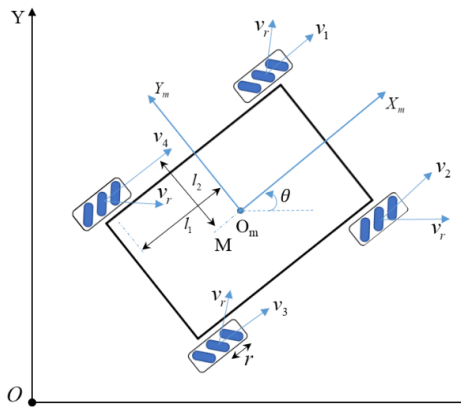


Figure 4. The four mecanum wheeled robot in the world frame.

where:

XOY is the base coordinate frame, $X_mO_mY_m$ is the coordinate frame of the robot, the direction angle θ

v_i : speed of the wheels (m/s)

v_r : passive roller speed (m/s)

r : wheel radius (m), middle angle (v_i, v_r) is $\pi/4$

I : moment of inertia of the robot ($kg \times m^2$)

I_b : wheel moment of inertia ($kg \times m^2$)

Total speed of the wheel in the direction X_m and Y_m of the robot coordinate frame:

$$v_1 = v_{m1x} - v_{m1y} \tag{3}$$

$$v_2 = v_{m2x} + v_{m2y} \tag{4}$$

$$v_3 = v_{m3x} - v_{m3y} \tag{5}$$

$$v_4 = v_{m4x} + v_{m4y} \tag{6}$$

The speed of the wheel can also be expressed in terms of translational and angular velocities as follows:

$$v_{m1x} = \dot{x}_m - \dot{\theta}_m l_2 \qquad v_{m1y} = \dot{y}_m + \dot{\theta}_m l_1$$

Substituting Eqs. (3)–(6), we get combined $\omega_i = \frac{v_i}{r}$:

$$\begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{\theta}_m \end{bmatrix} = \frac{r}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ \frac{-1}{l_1+l_2} & \frac{1}{l_1+l_2} & \frac{1}{l_1+l_2} & \frac{-1}{l_1+l_2} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} \tag{7}$$

Inverse kinematics in the global coordinate system:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = Rot(z,\theta) \begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{\theta}_m \end{bmatrix} \tag{8}$$

Kinematic equations for MWMR:

$$\begin{bmatrix} \dot{x}_m \\ \dot{y}_m \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \tag{9}$$

D. Dynamic Modeling

The kinetic energy is given by [6], [10], [13]:

$$E = \frac{1}{2} m(\dot{x}_m{}^2 + \dot{y}_m{}^2) + \frac{1}{2} I\dot{\theta}^2 + \frac{1}{2} I_b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \tag{10}$$

The system moves on the ground, the gravitational potential energy of the system $T = 0$. The sum of the kinetic and potential energies of the system is:

$$L = E + T = E \tag{11}$$

$$L=\frac{1}{2}\left(m\frac{r^2}{8}+I\frac{r^2}{16(l_1+l_2)^2}+I_b\right)\left(\omega_1^2+\omega_2^2+\omega_3^2+\omega_4^2\right)+\left(\frac{mr^2}{8}-\frac{Ir^2}{16(l_1+l_2)^2}\right)(\omega_1\omega_3+\omega_2\omega_4)-\frac{Ir^2}{16(l_1+l_2)^2}(\omega_1\omega_2-\omega_1\omega_4-\omega_2\omega_3+\omega_3\omega_4) \quad (12)$$

The dynamics model can then be derived using the Lagrange's equations:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\varphi}_i}\right)-\frac{\partial L}{\partial \varphi_i}=Q_i \quad (13)$$

Taking the derivative and expanding the formula we get:

$$\begin{bmatrix} m+\frac{4I_b}{r^2} & 0 & 0 \\ 0 & m+\frac{4I_b}{r^2} & 0 \\ 0 & 0 & I+\frac{4I_b(l_1+l_2)^2}{r^2} \end{bmatrix}\begin{bmatrix}\ddot{x}\\ \ddot{y}\\ \ddot{\theta}\end{bmatrix}+\begin{bmatrix}(m+\frac{4I_b}{r^2})\dot{y}\dot{\theta}\\ -(m+\frac{4I_b}{r^2})\dot{x}\dot{\theta}\\ 0\end{bmatrix}=\frac{1}{r}\begin{bmatrix}\cos\theta+\sin\theta & \cos\theta-\sin\theta & \cos\theta+\sin\theta & \cos\theta-\sin\theta\\ -\cos\theta+\sin\theta & \cos\theta+\sin\theta & -\cos\theta+\sin\theta & \cos\theta+\sin\theta\\ -(l_1+l_2) & l_1+l_2 & l_1+l_2 & -(l_1+l_2)\end{bmatrix}\begin{bmatrix}Q_1\\Q_2\\Q_3\\Q_4\end{bmatrix} \quad (14)$$

The dynamic equations for MWMR:

$$M\ddot{q}+C(q,\dot{q})+B\xi=B\tau \quad (15)$$

where:

$$M=\begin{bmatrix} m+\frac{4I_b}{r^2} & 0 & 0 \\ 0 & m+\frac{4I_b}{r^2} & 0 \\ 0 & 0 & I+\frac{4I_b(l_1+l_2)^2}{r^2} \end{bmatrix};$$

$$\tau=\begin{bmatrix}\tau_1\\ \tau_2\\ \tau_3\\ \tau_4\end{bmatrix};\ C(q,\dot{q})=\begin{bmatrix}(m+\frac{4I_b}{r^2})\dot{y}\dot{\theta}\\ -(m+\frac{4I_b}{r^2})\dot{x}\dot{\theta}\\ 0\end{bmatrix};$$

$$\xi=\left[\frac{\mu \mathrm{rmg}}{4}\mathrm{sign}(\omega_i)\right];$$

$$B=\frac{1}{r}\begin{bmatrix}\cos\theta+\sin\theta & -\cos\theta+\sin\theta & -(l_1+l_2)\\ \cos\theta-\sin\theta & \cos\theta+\sin\theta & (l_1+l_2)\\ \cos\theta+\sin\theta & -\cos\theta+\sin\theta & (l_1+l_2)\\ \cos\theta-\sin\theta & \cos\theta+\sin\theta & -(l_1+l_2)\end{bmatrix}^T$$

III. Controller Design

A. Dynamic Surface Control

From the dynamic Eq. (15), the dynamical model can be summarized as follows:

$$\begin{cases} x_1=q \\ \dot{x}_1=x_2 \\ \dot{x}_2=u+f(x)\end{cases} \quad (16)$$

Considering the sliding surface: $S_1=x_1-x_{1d}$

Choose: $S_1=x_1-x_{1d}$

The virtual control signal x_{2d} is followed $\bar{x}_2$ through a filter that is the first-order inertial link:

$$\varsigma\dot{x}_{2d}+x_{2d}=\bar{x}_2, x_{2d}(0)=\bar{x}_2(0) \quad (17)$$

The sliding: $S_2=x_2-x_{2d}$

Derivative S_1, S_2 and combination Eq. (15) we choose:

$$u=\dot{x}_{2d}-f(x)-K_2S_2=\frac{\bar{x}_2-x_{2d}}{\varsigma}-f(x)-K_2S_2$$

From this we get the control signal:

$$\tau=-B^T(BB^T)^{-1}[M(-\frac{\dot{x}_{1d}-K_1S_1-x_{2d}}{\varsigma}+K_2S_2)-C-B\xi] \quad (18)$$

Define: $e_1=q-q_d$; $\chi_2=x_{2d}-\bar{x}_2$

We get: $\dot{S}_1=-K_1S_1+S_2+\chi_2$

Proof. Considering the Lyapunov function is

$$V=\frac{S_1^TS_1+S_2^TS_2+\chi_2^T\chi_2}{2} \quad (19)$$

Taking the derivative:

$$\dot{V}=S_1^T\dot{S}_1+S_2^T\dot{S}_2+\chi_2^T\dot{\chi}=S_1^T(-K_1S_1+S_2+\chi_2)+S_2^T(-K_2S_2)+\chi_2^T(-\frac{\chi_2}{\varsigma}+\eta_2) \quad (20)$$

Using the inequalities:

$$S_1^TS_2\le\frac{S_1^TS_1+S_2^TS_2}{2};\quad S_1^T\chi_2\le\frac{S_1^TS_1+\chi_2^T\chi_2}{2};\quad \chi_2^T\eta_2\le\frac{3(\chi_2^T\chi_2)(\eta_2^T\eta_2)}{2}+\frac{\varepsilon}{3} \quad (21)$$

Hence, we have:

$$\dot{V}\le\frac{2S_1^TS_1+S_2^TS_2+\chi_2^T\chi_2}{2}-(2+K)(S_1^TS_1+S_2^TS_2)+\varepsilon+[-\chi_2^T\chi_2(1+\frac{M^2}{2\varepsilon}+K)+\frac{M^2(\chi_2^T\chi_2)}{2\varepsilon}\frac{(\eta_2^T\eta_2)}{M^2}] \quad (22)$$
$$\le-2KV+\varepsilon-(1-\frac{\eta_2^T\eta_2}{M^2})\frac{M^2(\chi_2^T\chi_2)}{2\varepsilon}$$

With the condition $V>\varepsilon/2K$ gives $\dot{V}<0$, the Lyapunov stability can be satisfied. Since the constant ε can be arbitrarily small, the errors of the system can always be limited to an acceptable level.

B. Adaptive Fuzzy Dynamic Surface Control

It can be obsered that by using the DSC algorithm for MWMR, the quality of the system depends on the value of (K_1, K_2). Therefore, we propose the Addaptive Fuzzy Dynamics Surface Control (AFDSC) which can solve this issue. In particular, the structure diagram of the proposed AFDSC algorithm is shown in Fig. 5.

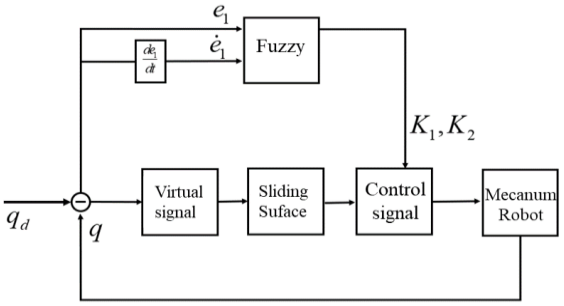


Figure 5. Structure of AFDSC controller.

The input of fuzzy block includes tracking error e_1 and its derivative $\dot{e}_1$. The member ship functions are shown in Fig. 6 and Fig. 7.

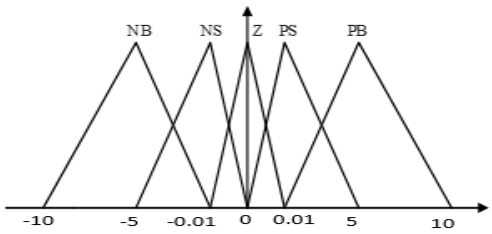


Figure 6. Member ship functions for input $\dot{e}_1$.

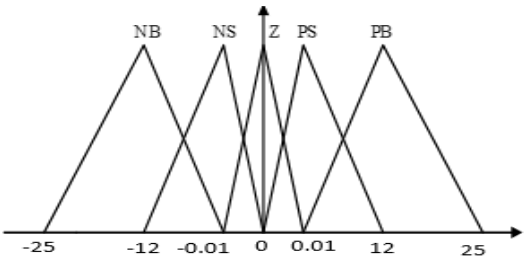


Figure 7. Member ship functions for input $\dot{e}_1$.

After formulate language variable and domain, we must determine membership of the fuzzy language variables. Considering the design should be simple and meet real-time requirements, the system adopts triangular membership function for variables e_1 and $\dot{e}_1$. Table I shows the membership function names and their meanings.

TABLE. I. INPUT VALUES

e_1	Language variable $\dot{e}_1$	Meaning
NB	NB	Negative big
NS	NS	Negative small
Z	Z	Zero
PS	PS	Positive small
PB	PB	Positive big

Outputs (K_1, K_2) are selected as Table II and the base inference rules of the fuzzy tuner for these two outputs are shown in Tables III and IV.

TABLE. II. OUTPUT VALUES

Values	Meaning	Output K_1	Output K_2
VS	Very small	3	10
S	Small	5	15
M	Medium	10	20
B	Big	12	30
VB	Very big	15	40

TABLE. III. BASIS OF TRADE SYSTEM FOR K1

$\dot{e}_1$	e_1				
	NB	***NS***	***Z***	***PS***	***PB***
NB	*M*	*S*	*VS*	*S*	*M*
NS	*B*	*M*	*S*	*M*	*B*
Z	*VS*	*B*	*M*	*B*	*VS*
PS	*B*	*M*	*S*	*M*	*B*
PB	*M*	*S*	*VS*	*S*	*M*

TABLE. IV. BASE DECISION SYSTEM FOR K2

$\dot{e}_1$	e_1				
	NB	***NS***	***Z***	***PS***	***PB***
NB	M	B	VB	B	M
NS	S	M	B	M	S
Z	VB	S	M	S	VB
PS	S	M	B	M	S
PB	M	B	VB	B	M

IV. SIMULATION AND RESULTS

In this section, we present and discuss simulation results obtained by the proposed controller. The simulation first is implemented on Matlab-Simulink, then on Gazebo and ROS.

A. Simulation on Matlab-Simulink

On Matlab-Simulink software, we consider the desired trajectory of the robot as follows:

$$x = \cos(\frac{2\pi}{15}t);\ y = \sin(\frac{2\pi}{15}t);\ \theta_r = \frac{\pi}{4}$$

Parameters of MWMR and controller are shown in Table V.

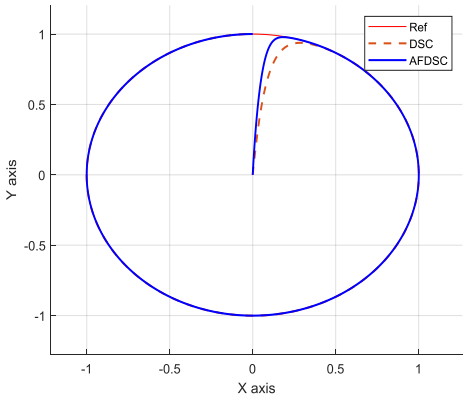
TABLE. V. PARAMETERS OF ROBOT AND CONTROLLER

The parameters in the dynamical model
$m = 20\,kg;$ I=0.56 kgm^2; $r = 0.075\,m,\ l_1 = l_2 = 0.3\,m$
Orbital parameter
$0 < t < 15,\ r_0 = 1m$
Controller parameter
$K_1 = diag(15,15,15); K_2 = diag(40,40,40)$

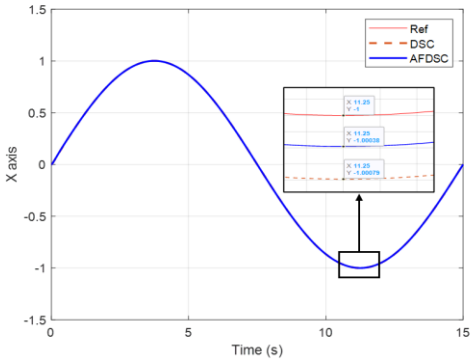
For the performance comparison, we consider the DSC as a baseline algorithm.

First, we discuss the convergence of the algorithms and the tracing trajectory obtained by the algorithms. As shown in Fig. 8(a), the robot can track along the trajectory by using both algorithms. Moreover, the system response is fast and stable even when the robot is not on the desired path. The AFDSC algorithm can track of the desired trajectory after 0.6 s, while the DSC algorithm and track after 1.1 s.

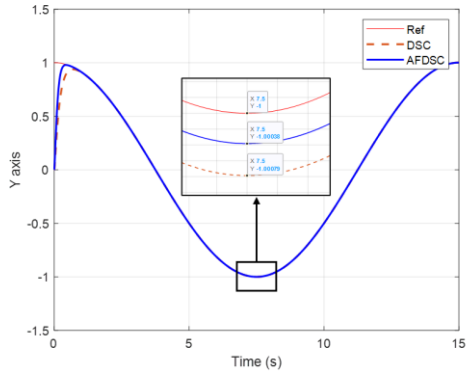
Additionally, the tracking errors in x, y and θ of the reference trajectory are shown in the Figs. 8 (b), (c) and (d), respectively. It is observed that the error is converging to zero by both algorithms. Therefore, the desired trajectory is tracked by both algorithms. However, the tracking error along the x-axis, the y-axis of the AFDSC algorithm, i.e., 0.00038 m, is smaller than that of DSC algorithm, i.e., 0.00079 m. This demonstrates the effectiveness of the proposed controller.



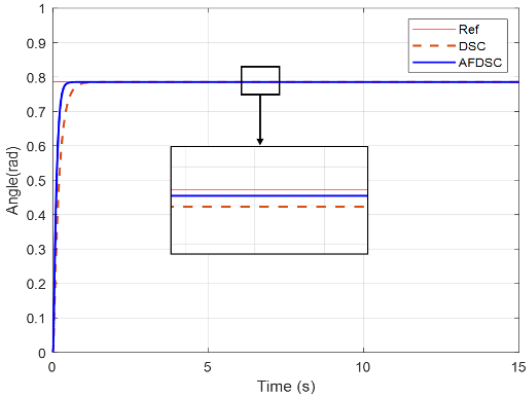
(a)



(b)



(c)



(d)

Figure 8. Tracking performance of DSC and AFDSC (a) trajectory (b) X axis (c) Y axis (d) angle of direction.

Based on the simulation results, it is found that both controllers ensure the MWMR to follow the desired trajectory, but the AFDSC gives better tracking quality. Fuzzy rules are designed to update the parameters of the online DSC every time there is a change in the two fuzzy tuner inputs, the bias and the biased derivative. The proposed AFDSC controller ensures better tracking quality than the DSC controller.

B. Simulation under Gazebo and ROS

The robot model is constructed in Robot Operating System (ROS) platform to verify and confirm the performance of the proposed algorithm and its practical feasibility.

This article uses the Gazebo simulation tool, monitors it with the Rviz visualization tool on the ROS platform to evaluate the results of the selected AFDSC controller. The ROS computation graph of the robot is described in Fig. 9.

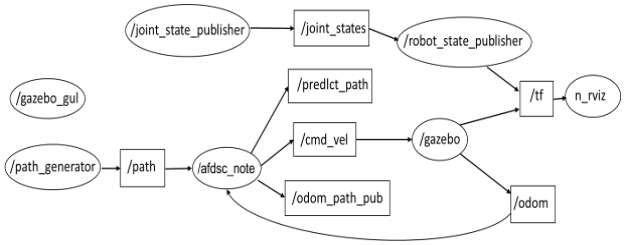


Figure 9. Block diagram of AFDSC controller in ROS.

At the beginning of the movement, the robot was accelerated to obtain a position, direction, and velocity that approximated the desired trajectory. The obtained robot path is described as Fig. 10, with the reference trajectory is the blue line and the robot path is the red line. The origin is (0, 0) the red axis is Ox and the green axis is Oy. The control signal of the robot is shown in Fig. 11. Initially, the ROS executable navigation package has not been run between the 30th and 37th seconds, so the control signal is not available. Approximately from the 37th second is the output control signal response.

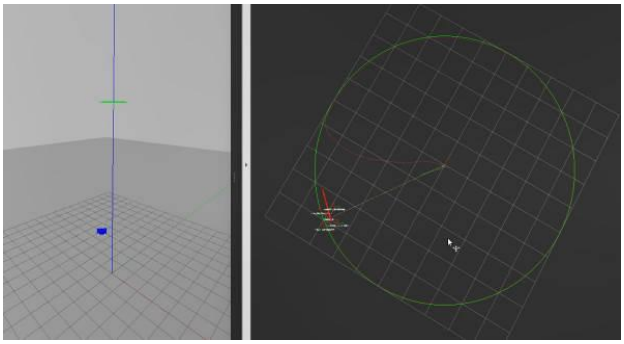


Figure 10. The trajectory of the MWMR along the XY axis (Rviz).

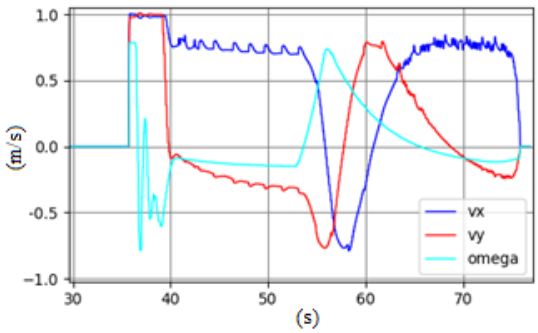


Figure 11. Robot control signals.

The error of tracking the robot's trajectory has a big change because the initial value of the robot's position is far from the set trajectory, then the robot encounters a transition that takes about 4 seconds to stick to the reference trajectory, then the trajectory moves of the robot converges with the reference after the time from 37 seconds. During the tracking process, there is still slight oscillation of the robot around the set orbit due to the delay time of the controller.

V. Conclusion

In this paper, we have proposed an AFDSC algorithm for MWMR to deal with the trajectory tracking problem. In particular, by combining combining the dynamic surface control and fuzzy logic, the parameters of controller were adjusting to adaptive with the changing of the dynamics of the MWMR. The designed control algorithm was implemented and evaluate on a MWMR platform. The simulation results demonstrate the effectiveness of the proposed algorithm, and the robot can achieved superior trajectory tracking accuracy. Our future efforts are aimed at investigating the tracking control and obstacle avoidance problems for multi-robot subject to random disturbances.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Dong Nguyen Minh, Hiep Do Quang, Tien Ngo Manh and Duy Nam Bui designed the proposed algorithm; Dong Nguyen Minh and Tien Ngo Manh implemented and conducted the experiment; Dong Nguyen Minh and Duy Nam Bui wrote the paper; Nam Dao Phuong and Tien Ngo Manh reviewed the paper; all authors had approved the final version.

Funding

This research was funded by Vietnam's National project "Research, develop an intelligent mobile robot using different types of sensing technology and IoT platform, AI, and implemented in radioactive environment monitoring application", code: DTDLCN.19/23 of the CT1187 Physics development program in the period 2021- 2025.

References

[1] D. Q. Hiep, L. T. Thang, N. M. Tien, N. M. Cuong, N. N. Toan, and B. D. Nam, "Design a nonlinear MPC controller for autonomous mobile robot navigation system based on ROS," *International Journal of Mechanical Engineering and Robotics Research*, vol. 11, no. 6, pp. 379-388, June 2022.

[2] I. Zeidis and K. Zimmermann, "Dynamics of a four-wheeled mobile robot with Mecanum wheels," *Z Angew Math Mech.2019;99:e201900173*, October 2019.

[3] T. Hamid and X. Z. Chun, "Omnidirectional mobile robots, mechanisms and navigation approaches," *Mechanism and Machine Theory*, vol. 153. 2020.

[4] Z. Yuan, Y. Tian, Y. Yin, S. Wang, J. Liu, and L. Wu, "Trajectory tracking control of a four mecanum wheeled mobile platform: An extended state observer-based sliding mode approach," *IET Control Theory Appl.*, vol. 14, pp. 415-426, 2020.

[5] Z. Sun, S. Hu, D. He, W. Zhu, H. Xie, and J. Zheng, "Trajectory-tracking control of mecanum-wheeled omnidirectional mobile robots using adaptive integral terminal sliding mode," *Computers & Electrical Engineering*, vol. 96, pp. 107500, Dec. 2021.

[6] J. E. Normey-Rico, I. Alcalá, J. Gómez-Ortega, and E. F. Camacho, "Mobile robot path tracking using a robust PID controller," *Control Engineering Practice*, vol. 9, no. 11, pp. 1209–1214, Nov. 2001.

[7] N. H. Thai, T. T. K. Ly, and L. Q. Dzung, "Trajectory tracking control for mecanum wheel mobile robot by time-varying parameter PID controller," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 4., pp. 1902–1910, Aug., 2022.

[8] I. Carlucho, M. De Paula, and G. G. Acosta, "An adaptive deep reinforcement learning approach for MIMO PID control of mobile robots," *ISA Transactions*, vol. 102, pp. 280–294, July 2020.

[9] V. Alakshendra and S. S. Chiddarwar, "Adaptive robust control of Mecanum-wheeled mobile robot with uncertainties," *Nonlinear Dynamic*, vol. 87, pp. 2147–2169, 2017.

[10] X. Lu, X. Zhang, G. Zhang, and S. Jia, "Design of adaptive sliding mode controller for four-mecanum wheel mobile robot," in *Proc. 2018 37th Chinese Control Conference (CCC)*, Wuhan, China, pp. 3983-3987, 2018.

[11] Z. Sun, H. Xie, J. Zheng, Z. Man, and D. He, "Path-following control of Mecanum-wheels omnidirectional mobile robots using nonsingular terminal sliding mode," *Mechanical Systems and Signal Processing*, vol. 147, p. 107128, Jan. 2021.

[12] P. S. Yadav, V. Agrawal, J. C. Mohanta, and M. D. Faiyaz Ahmed, "A robust sliding mode control of mecanum wheel-chair for trajectory tracking," *Materials Today: Proceedings*, vol. 56. Elsevier BV, pp. 623–630, 8 April 2022.

[13] G. Cao, X. Zhao, C. Ye, *et al*, "Fuzzy adaptive PID control method for multi-mecanum-wheeled mobile robot," *Journal of Mechanical Science and Technology*, vol. 36, pp. 2019–2029, 2022.

[14] X. Zou, T. Zhao, and S. Dian, "Finite-time adaptive interval type-2 fuzzy tracking control for mecanum-wheel mobile robots," *International Journal of Fuzzy Systems*, vol. 24, no. 3. Springer Science and Business Media LLC, pp. 1570–1585, Nov. 28, 2021.

[15] M. A. Elrehem, A. Hassan, D. Ojo, H. Hassan, and M. Hosny, "The implementation of fuzzy logic controller for the obstacle

avoidance in 3 Mecanum-wheeled Robot," in *Proc. 2022 4th Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, pp. 288-291, Giza, Egypt, 2022.
[16] P. Azizinezhad, R. Sadeghian, and M. Tale Masouleh, "An experimental study on controlling and obstacle avoidance of a four mecanum wheeled robot," in *Proc. 2017 5th RSI International Conference on Robotics and Mechatronics (ICRoM)*, pp. 21-26, Tehran, Iran, 2017.
[17] C. Nguyen Manh, N. T. Nguyen, N. Bui Duy, and T. L. Nguyen, "Adaptive fuzzy Lyapunov-based model predictive control for parallel platform driving simulators," *Transactions of the Institute of Measurement and Control*, vol. 45, no. 5. SAGE Publications, pp. 838–852, Sep. 20, 2022.
[18] D. Swaroop, J. C. Gerdes, P. P. Yip, and J. K. Hedrick, "Dynamic surface control of nonlinear systems," in *Proc. the 1997 American Control Conference (Cat. No.97CH36041)*, Albuquerque, NM, USA, Vol.5, pp. 3028-3034, 1997.
[19] Z. Peng, D. Wang, Z. Chen, X. Hu, and W. Lan, "Adaptive dynamic surface control for formations of autonomous surface vehicles with uncertain dynamics," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 2, pp. 513-520, 2013.
[20] C. Yang and Z. Wu, "Adaptive robust maneuvering control for nonlinear systems via dynamic surface technique," *Nonlinear Dynamics*, vol. 111, no. 9. Springer Science and Business Media LLC, pp. 8369–8381, Feb. 09, 2023.
[21] C. Wang, D. Wang, and Y. Han, "Neural network based adaptive dynamic surface control for omnidirectional mobile robots tracking control with full-state constraints and input saturation," *International Journal of Control, Automation and Systems*, vol. 19, no. 12. Springer Science and Business Media LLC, pp. 4067–4077, Dec. 2021.
[22] P. P. Yip and J. Karl Hedrick, "Adaptive dynamic surface control: A simplified algorithm for adaptive backstepping control of nonlinear systems," *International Journal of Control*, vol. 71, no. 5, pp. 959-979, 1998.
[23] N. Tlale and M. de Villiers, "Kinematics and dynamics modelling of a mecanum wheeled mobile platform," in *Proc. 2008 15th International Conference on Mechatronics and Machine Vision*, pp. 657-662, in Auckland, New Zealand, 2008.