Journal of Robotics and Control (JRC)
Volume 6, Issue 2, 2025
ISSN: 2715-5072, DOI: 10.18196/jrc.v6i2.23693

Application of an Adaptive Dynamic Sliding Surface Controller with Traction Tracking for a Mecanum Wheel Mobile Robot

Ha Vo Thu [1*], Thuong Than Thi [2], Thanh Nguyen Thi [3], Binh Nguyen Hai [4], Dung Vu Van [5]
[1, 2, 3, 4, 5] Faculty of Electrical, University of Economics Technology for Industries, Hanoi, Vietnam
Email: [1] vtha@uneti.edu.vn, [2] ttthuong.dien@uneti.edu.vn, [3] ntthanh.ddt@uneti.edu.vn, [4] nhbinh@uneti.edu.vn, [5] vvdung@uneti.edu.vn
*Corresponding Author

***Abstract*—The paper introduces an algorithm application of an adaptive Dynamic Sliding Surface Controller that integrates neural networks and fuzzy logic systems with traction tracking for a Mecanum Wheel Mobile Robot. In this framework, neural networks are employed to approximate the uncertain nonlinear numerical aspects of MWMR, while fuzzy logic systems are utilized to adaptively. The stability of the closed-loop system is investigated using the Lyapunov criterion. The proposed controller is verified by numerical simulation. The simulation results show that the proposed controller performs better than the backstepping sliding controller in the case of uncertain model parameters and when there is an impact disturbance.**

Keywords*—*Mecanum Wheeled Autonomous Robot (WMMR); Sliding Mode Control (SMC); Fuzzy Logic System (FLS); Neural Network (NN).

I. INTRODUCTION

Recently, omnidirectional autonomous robots have been improved to increase manoeuvrability and payload, aiming at specific industrial applications, including changing the Omni wheel structure to using Mecanum wheels. Mecanum wheels have two common types: α = 45o and α = 90o [1]. When transmitting torque to the wheel, the rollers on the wheel in contact with the floor will form two velocity components, such as the velocity in the direction of wheel movement and the velocity perpendicular to the roller axis, and the direction depends on the direction of the torque. Therefore, controlling the motor's coordination, starting with the driving wheels, will create a force vector that pushes the self-propelled robot in different directions, increasing the robot's flexibility. Therefore, in recent years, this type of wheel has also been applied in the design of autonomous robot models for logistics and transportation with small areas that are not enough to design a turning path for the robot, such as L. Schulze et al. [2] created an omnidirectional robot using Mecanum wheels with two functions of transporting and pulling shelves, or Michael Göller et al. [3][4] applied Mecanum wheels to design robot models to serve supermarket. In addition, there are several other studies [5][6] for industrial production, agriculture [7][8], industrial production transportation [9]-[11], and space exploration [12].

Nowadays, due to the complex working environment and small and narrow space, the design of the turning radius for the robot is not enough, so MWMR has been improved and upgraded to meet the requirements of intelligent logistics systems and modern industrial systems [13]. MWMR is integrated with the IoT-based manufacturing system in the factory's internal logistics automation system [14].

Furthermore, MWMR is integrated with a controller to perform a variety of tasks, including large-scale wind power plant blade processing [15], Super Proton Synaptic acceleration tracking robot [16], aromatherapy robot providing essential oils or medicinal products to avoid or alleviate COVID-19 infection [17], and so on. To facilitate the design and application of control algorithms as in [18]-[27] or trajectory algorithms [28][29] and image processing, vision systems for robots in [30], and [31], robots are modelled using kinematic models and dynamic models, based on the Euler-Lagrange principle but differ in the mechanical structure, transmission of WMRM such as using three wheels, four wheels, of which typical are [32]-[40]. With the nature of nonlinear models, the estimation or omission of model parameters significantly affects the control quality of MWMR [41][42] are two works that have been carefully and qualitatively studied in building kinematic and dynamic models for WMRM, with the most scientific and complete evaluation and parameter estimation for WMRM models. There are also a few studies that have focused on examining the navigation position of autonomous robots as well as control according to the kinematic model of WMRM, shown in [43]-[46]. Due to the limitations of using only the dynamic model with nonlinear components, including friction, vibration, wheel slip, etc., for the motion control of MWMR, the dynamic equation is considered to obtain a more effective method for improving the control quality. The dynamic model is constructed in [47] and [48], followed by some tracking control algorithms for this full model in [49] and [50].

Therefore, many controllers have been proposed to control the trajectory tracking motion for MWMR, among which the PID controller is still the typical controller [51]-[53]. However, the PID controller cannot control the system effectively due to its low accuracy when the motion trajectory is complex and the desired velocity changes over time. To overcome the disadvantages of PID, the PID controller has been improved to respond to MWMR with uncertain parameters, such as the fuzzy PID controller [54][55]. On the other hand, the algorithm uses a genetic algorithm to find near-optimal motion trajectories in logistics planning for mobile robots [56]. The controller architecture is based on a sliding model that includes connections like non-contact senses for precise route tracking [57][58] etc. is also a new approach to improve the control quality, and this algorithm also gives results with good tracking quality, maximum deviation of about 0.08 (m). To ensure the control quality when considering nonlinear components, the Backstepping feedback method is a feasible solution to solve the back-propagation nonlinear mathematical models [59][60].

However, for high-order nonlinear systems, the computational volume is large and complex and takes a lot of time due to the need to calculate the derivative in each iteration step. Next, the sliding mode controller (SMC) was then utilized [61]-[65] in the event of a disturbance-prone system. However, the SMC method has a restriction in the form of chattering, which can be reduced with an accurate object model. This contradicts the robot model's characteristics and parameter uncertainty. The Dynamic Sliding Surface Controller (DSC) is an effective alternative control method for nonlinear systems such as MWMR to improve the control quality and limit some disadvantages of the Backstepping and Sliding controller. Research [66] has presented the structure and construction method of the DSC controller. When the system contains uncertain components, the study focuses on improving the Backstepping controller and developing a Multi-Sliding Surface Controller (MSSC). Considering the system stability based on the Lyapunov control function. However, to avoid taking derivatives in the iteration steps of the virtual controller, DSC has added a low-pass filter [67]. In recent years, there has been a trend to use DSC because of its advantages [68], [69]. For MWMR, it is difficult to build an accurate mathematical model because factors such as friction, changing loads, and changing environmental conditions are unpredictable. Therefore, modern, practical design methods, in this case, are to use adaptive algorithms to tune the controller parameters or approximate the uncertain parameters of the object. Many research works in this direction use fuzzy logic systems as adaptive tuning mechanisms for nonlinear controllers [70]-[74].

However, when the system comprises a large number of uncertain nonlinear components, the system model deviates significantly. As a result, an algorithm is required to anticipate and estimate the unknown components in order to develop an adaptive controller for the system. With their ability to learn and estimate nonlinear functions with great accuracy, NNs have piqued researchers' interest in adaptive control system applications. The radial neural network (NN) is a technique for estimating unknown parameters in the controller [75]-[78].

Furthermore, the NN is paired with a fuzzy controller [79][80], establishing a new viable research direction for MWMR. That is also the research direction chosen in the article. The results of this investigation will be presented in four sections. Sections 1 and 2 outline the research objectives and mathematical modelling equations for MWMR. Section 3 discusses the algorithmic structure of the controller, as well as simulations performed to evaluate and validate the proposed controller. Section 4 concludes and provides directions for further research.

II. The Four Mecanum Wheel Mobile Robot

A. *The Kinematics of WMMR*

Consider a Mecanum-wheels omnidirectional mobile robot as Fig. 1.

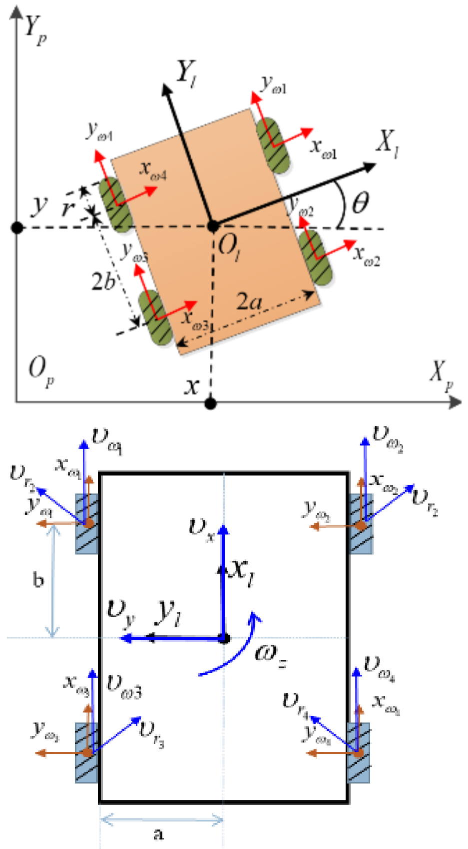

Fig. 1. Kinematic relationship of Mecanum-wheels mobile robot

The velocity vector is parallel to the axis and at an angle of 45 degrees as shown in Fig. 1.

$$\begin{aligned} v_{x1} &= v_{\omega 1} + \frac{v_{r1}}{\sqrt{2}},\ v_{y1} = \frac{v_{r1}}{\sqrt{2}};\\ v_{x2} &= v_{\omega 2} + \frac{v_{r2}}{\sqrt{2}},\ v_{y2} = \frac{v_{r2}}{\sqrt{2}};\\ v_{x3} &= v_{\omega 3} + \frac{v_{r3}}{\sqrt{2}},\ v_{y3} = \frac{v_{r3}}{\sqrt{2}};\\ v_{x4} &= v_{\omega 4} + \frac{v_{r4}}{\sqrt{2}},\ v_{y4} = \frac{v_{r4}}{\sqrt{2}}; \end{aligned} \tag{1}$$

The wheel speed is calculated according to the robot's.

$$\begin{aligned} v_{x1} = v_x - a\omega_z, v_{y1} = v_y + b\omega_z; v_{x2} = v_x + a\omega_z, v_{y2} \\ = v_y + b\omega_z \\ v_{x3} = v_x - a\omega_z, v_{y3} = v_y - b\omega_z; v_{x4} = v_x + a\omega_z, v_{y4} \\ = v_y - b\omega_z \end{aligned} \tag{2}$$

From (1) and (2) we have:

$$\begin{aligned} v_{\omega_1} = \dot{\theta}_1 = v_x - v_y - (b+a)\omega_z, v_{\omega_2} = \dot{\theta}_2 \\ = v_x + v_y + (b+a)\omega_z, \\ v_{\omega_3} = \dot{\theta}_3 = v_x + v_y - (b+a)\omega_z, v_{\omega_4} = \dot{\theta}_4 \\ = v_x - v_y + (b+a)\omega_z \end{aligned} \tag{3}$$

From formula (3):

$$v_\omega = Jv_l \tag{4}$$

With: $v_\omega = [\dot\theta_1, \dot\theta_2, \dot\theta_3, \dot\theta_4]^T$; $v_l = [v_{xl}, v_{yl}, \omega_z]^T = [\dot x_l, \dot y_l, \dot\theta_l]^T$

$$J = \begin{bmatrix} 1 & -1 & -(a+b) \\ 1 & 1 & (a+b) \\ 1 & 1 & -(a+b) \\ 1 & -1 & (a+b) \end{bmatrix} \tag{5}$$

From (4) and (5) we deduce:

$$v_l = \begin{bmatrix} \dot x_l \\ \dot y_l \\ \dot\theta_l \end{bmatrix} = \frac{r}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -\frac{1}{a+b} & \frac{1}{a+b} & -\frac{1}{a+b} & \frac{1}{a+b} \end{bmatrix}\begin{bmatrix} \dot\theta_1 \\ \dot\theta_2 \\ \dot\theta_3 \\ \dot\theta_4 \end{bmatrix} \tag{6}$$

The equation representing this relationship is also the robot's kinematic equation.

$$\dot q = \begin{bmatrix} \dot x \\ \dot y \\ \dot\theta \end{bmatrix} = R(\theta)v_l = \begin{bmatrix} C\theta & -S\,\theta & 0 \\ S\,\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} \dot x_l \\ \dot y_l \\ \dot\theta_l \end{bmatrix} \tag{7}$$

With: $R(\theta) = \begin{bmatrix} \mathrm{Cos}\theta & -Sin\,\theta & 0 \\ Sin\,\theta & \mathrm{Cos}\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Formula (7) yields the robot system's kinematic equation:

$$v_l = R(\theta)^{-1}\dot q \tag{8}$$

Instead of (4) we have:

$$v_\omega = Jv_l = JR(\theta)^{-1}\dot q \tag{9}$$

B. *Dynamics of WMMR*

The kinetic energy of a self-propelled robot is calculated by the kinetic energy of the robot body plus the kinetic energy on the 4 wheels:

$$T_l = \frac{1}{2}m_l v_l^2 + \frac{1}{2}I_l{\omega_z}^2 \tag{10}$$

$$\begin{aligned} T_{\omega 1} &= \frac{1}{2}m_\omega v_{\omega 1}^2 + \frac{1}{2}I_m{\omega_z}^2 + \frac{1}{2}I_\omega\dot\theta_1^2 \\ T_{\omega 2} &= \frac{1}{2}m_\omega v_{\omega 2}^2 + \frac{1}{2}I_m{\omega_z}^2 + \frac{1}{2}I_\omega{\dot\theta_2}^2 \end{aligned} \tag{11}$$

$$\begin{aligned} T_{\omega 3} &= \frac{1}{2}m_\omega v_{\omega 3}^2 + \frac{1}{2}I_m{\omega_z}^2 + \frac{1}{2}I_\omega\dot\theta_3^2; \\ T_{\omega 4} &= \frac{1}{2}m_\omega v_{\omega 4}^2 + \frac{1}{2}I_m{\omega_z}^2 + \frac{1}{2}I_\omega{\dot\theta_4}^2 \end{aligned} \tag{12}$$

T_l: Kinetic energy of the robot;$T_{\omega i}$ Kinetic energy of the wheels (i=1, 2, 3, 4).

From formulas (3), (10), (11), (12) we have the total kinetic energy:

$$T = \frac{1}{2}m_t({\dot x_l}^2 + {\dot y_l}^2) + \frac{1}{2}I{\omega_z}^2 + \frac{1}{2}I_\omega(\dot\theta_1^2 + \dot\theta_2^2 + \dot\theta_3^2 + \dot\theta_4^2) \tag{13}$$

With: $m_t = m_l + 4m_\omega$; $I = I_l + 4m_\omega(b^2 + a^2) + I_m$

According to formula (3) we have:

$$v_{xl} = \dot x_l = \frac{r}{4}(\dot\theta_1 + \dot\theta_2 + \dot\theta_3 + \dot\theta_4); v_{yl} = \dot y_l = \frac{r}{4}(-\dot\theta_1 + \dot\theta_2 + \dot\theta_3 - \dot\theta_4)$$

$$\begin{aligned} \dot\theta_l &= \omega_z = \frac{r}{4(a+b)}(-\dot\theta_1 + \dot\theta_2 - \dot\theta_3 + \dot\theta_4); \\ v_\omega &= [\dot\varphi_1, \dot\varphi_2, \dot\varphi_3, \dot\varphi_4]^T = [r\dot\theta_1 \quad r\dot\theta_2 \quad r\dot\theta_3 \quad r\dot\theta_4]^T \end{aligned} \tag{14}$$

From formulas (13) and (14) we have:

$$\begin{aligned} T &= \frac{1}{2}\left(\frac{m_t r^2}{8} + \frac{Ir^2}{16(b+a)^2} + I_\omega\right)(\dot\theta_1^2 + \dot\theta_2^2 + \dot\theta_3^2 + \dot\theta_4^2) \\ &+ \left(\frac{m_t r^2}{8} - \frac{Ir^2}{16(b+a)^2}\right)(\dot\theta_1\dot\theta_3 - \dot\theta_2\dot\theta_4) \\ &- \frac{Ir^2}{16(b+a)^2}(\dot\theta_1\dot\theta_2 - \dot\theta_1\dot\theta_4 - \dot\theta_1\dot\theta_3 + \dot\theta_3\dot\theta_4) \end{aligned} \tag{15}$$

With: $A = \frac{m_t r^2}{8}$; $B = \frac{Jr^2}{16(L+d)^2}$;

Equation (15) will be written:

$$\begin{aligned} T = \frac{1}{2}(A + B + I_\omega)(\dot\theta_1^2 + \dot\theta_2^2 + \dot\theta_3^2 + \dot\theta_4^2) + (A - B)(\dot\theta_1\dot\theta_4 + \dot\theta_2\dot\theta_3) \\ - B(\dot\theta_1\dot\theta_2 - \dot\theta_1\dot\theta_3 - \dot\theta_2\dot\theta_4 + \dot\theta_3\dot\theta_4) \end{aligned} \tag{16}$$

With: $L = T - V$ is the Lagrangian function. Using the Euler-Lagrange equation, we have:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\dot\theta}\right) - \frac{\partial L}{\partial\theta} = \tau - F(\dot\theta) \tag{17}$$

Using equations (15) and (16) with the Lagrange method (17), the robot's motion equation is described by the system of equations:

$$\begin{cases} (A + B + I_\omega)\ddot\theta_1 - B\ddot\theta_2 + B\ddot\theta_3 + (A - B)\ddot\theta_4 = \tau_1 - f_{c1}\,sgn(\dot\theta_1) \\ -B\ddot\theta_1 + (A + B + I_\omega)\ddot\theta_2 + (A - B)\ddot\theta_3 + B\ddot\theta_4 = \tau_2 - f_{c2}\,sgn(\dot\theta_2) \\ B\ddot\theta_1 + (A - B)\ddot\theta_2 + (A + B + I_\omega)\ddot\theta_3 - B\ddot\theta_4 = \tau_3 - f_{c3}\,sgn(\dot\theta_3) \\ (A - B)\ddot\theta_1 + B\ddot\theta_2 - B\ddot\theta_3 + (A + B + I_\omega)\ddot\theta_4 = \tau_4 - f_{c1}\,sgn(\dot\theta_4) \end{cases} \tag{18}$$

$$\Leftrightarrow \tau = M\ddot\theta + G\,sgn(\dot\theta) + \tau_d \tag{19}$$

With: $\tau = [\tau_1, \tau_2, \tau_3, \tau_4]^T$; $\theta = [\theta_1, \theta_2, \theta_3, \theta_4]^T$;

$$F(\dot\theta) = \left[f_{c1}\,sgn(\dot\theta_1), f_{c2}\,sgn(\dot\theta_2), f_{c3}\,sgn(\dot\theta_3), f_{c4}\,sgn(\dot\theta_4)\right]^T \tag{20}$$

$\tau_d$is the external noise component:

$$M = \begin{bmatrix} (A + B + I_\omega) & -B & B & (A - B) \\ -B & (A + B + I_\omega) & (A - B) & B \\ B & (A - B) & (A + B + I_\omega) & -B \\ (A - B) & B & -B & (A + B + I_\omega) \end{bmatrix}$$

But according to the kinetic equation (9)

$$\begin{aligned} v_\omega = Jv_l = JR(\theta)^{-1}\dot q \;\Leftrightarrow \dot\theta = JR(\theta)^{-1}\dot q \\ \Rightarrow \ddot\theta = \dot JR(\theta)^{-1}\dot q + J\dot R(\theta)^{-1}\dot q + JR(\theta)^{-1}\ddot q \end{aligned}$$

So the dynamic equation (20) will be written:

$$\begin{aligned} \tau &= M(J\dot R(\theta)^{-1}\dot q + JR(\theta)^{-1}\ddot q) + G\,sgn(JR(\theta)^{-1}\dot q) + \tau_d \\ \Rightarrow \tau &= MJR(\theta)^{-1}\ddot q + MJ\dot R(\theta)^{-1}\dot q + G\,sgn(JR(\theta)^{-1}\dot q) + \tau_d \end{aligned} \tag{21}$$

Adaptive Sliding Surface Traction Tracking for Mecanum Wheel Mobile Robot

C. *Sliding Controller*

To simplify the calculation and demonstrate the stability of the control system, the state variables of the system are set as follows:

$$\begin{cases} x_1 = q = [x \quad y \quad \phi]^T \\ x_2 = v_R = [V_{Gx} \quad V_{Gy} \quad \Omega]^T \end{cases} \tag{22}$$

$$\Rightarrow \begin{cases} \dot{x}_1 = Rx_2 \\ MJ\dot{x}_2 + MJx_2 + GJ\,sgn(x_2) + \tau_d = E\tau \end{cases} \quad (23)$$

Assuming no external noise components, the MWMR model has the following form:

$$\begin{cases} \dot{x}_1 = Rx_2 \\ MJ\dot{x}_2 + MJx_2 + GJ\,sgn(x_2) = E\tau \end{cases} \quad (24)$$

Set: $e_1 = x_1 - x_{1d}$, $x_{1d} = q_d = [x_d\ y_d\ \varphi_d]$

$$\Rightarrow \dot{e}_1 = \dot{x}_1 - \dot{x}_{1d} = Rx_2 - \dot{x}_{1d} \quad (25)$$

β is defined as:

$$\beta = -R^{-1}(c_1 e_1 - \dot{x}_{1d}) \quad (26)$$

With $c_1 = \begin{bmatrix} c_{1x} & 0 & 0 \\ 0 & c_{1y} & 0 \\ 0 & 0 & c_{1\varphi} \end{bmatrix}$

β is fed via a first-order low-pass filter.

$$T\dot{\beta}_f + \beta_f = \beta \quad (27)$$

T was chosen to be minimal enough that it did not increase DSC computation time.

$$\beta_f(s) = \frac{\beta(s)}{Ts+1}; \dot{\beta}_f = \frac{\beta - \beta_f}{T} \quad (28)$$

Prove the stability of the virtual controller. The proposed lyapunov function:

$$V_1 = \frac{1}{2} e_1^T e_1 \quad (29)$$

$$\dot{V}_1 = e_1^T \dot{e}_1 = \dot{e}_1(Rx_2 - \dot{x}_{1d}) = -e_1^T c_1 e_1 + e_1^T(c_1 e_1 + Rx_2 - \dot{x}_{1d}) \quad (30)$$

With $x_2 = \beta$ then $\dot{V}_1 = -e_1^T c_1 e_1 + e_1^T \dot{c}_1 (e_1\text{-}e_1) = -e_1^T c_1 e_1$

From (30) and (26) we can deduce that:

$$\dot{V}_1 = \text{-}e_1^T c_1 e_1 \le 0 \quad (31)$$

Let e_2 be the virtual control signal error and is determined by:

$$e_2 = x_2 - \beta_f \quad (32)$$

Select slide surface:

$$S = \lambda e_1 + N\, e_2 \quad (33)$$

With $\lambda > 0$ is the coefficient of the slip surface.

$$\begin{aligned} &\dot{S} = \lambda \dot{e}_1 + N\dot{e}_2 + \dot{N}e_2 = \lambda \dot{e}_1 + \dot{N}e_2 + N(\dot{x}_2 - \dot{\beta}_f) \\ &\Rightarrow \dot{S} = \lambda \dot{e}_1 + \dot{N}e_2 + N[J^{-1}M^{-1}(MJx_2 - GJ\,sgn(x_2) + E\tau) - \dot{\beta}_f] \end{aligned} \quad (34)$$

Next, to improve the control quality, we consider the system stability as follows: The second Lyapunov candidate function is chosen to ensure the system's stability and calculate the control signa.

$$V_2 = \frac{1}{2} S^T S \quad (35)$$

The control signal of the system will be calculated in the form of a sliding mode controller to increase the system's robustness to disturbances. Therefore, the control signal consists of 2 components as follows: τ_{eq} is defined as follows:

$$\tau_{eq} = -E^T(EE^T)^{-1} \begin{pmatrix} MJ(N^{-1}(\lambda \dot{e}_1 + \dot{N}e_2) - \dot{x}_{2d}) \\ -MJx_2 - GJ\,sgn(x_2) \end{pmatrix} \quad (36)$$

τ_{sw} is defined as follows:

$$\tau_{sw} = -E^T(EE^T)^{-1} MJN^{-1}(c_2\,sgn(S) + c_3 S) \quad (37)$$

with $c_2 = \begin{bmatrix} c_{2x} & 0 & 0 \\ 0 & c_{2y} & 0 \\ 0 & 0 & c_{2\varphi} \end{bmatrix}$; $c_3 = \begin{bmatrix} c_{3x} & 0 & 0 \\ 0 & c_{3y} & 0 \\ 0 & 0 & c_{3\varphi} \end{bmatrix}$

The system's control signal is the sum of τ_{eq}, τ_{sw}:

$$\tau = \tau_{eq} + \tau_{sw} \quad (38)$$

$$\Rightarrow \dot{V}_2 = S^T\dot{S} == S^T \begin{Bmatrix} \lambda \dot{e}_1 + \dot{N}e_2 + \\ N[J^{-1}M^{-1}(-MJx_2 - GJ\,sgn(x_2) + E\tau) - \dot{\beta}_f] \end{Bmatrix}$$

With (36) and (31) then $\dot{x}_{2d} = \dot{\beta}_f$:

$$\Rightarrow \dot{V}_2 = -S^T c_2\,sgn(S) - S^T c_3 S_1 \quad (39)$$

With c_2 and c_3 are the coefficients, so:

$$\dot{V}_2 = -S^T c_2\,sgn(S) - S^T c_3 S \le 0 \quad (40)$$

This satisfies the Lyapunov stability criterion.

- Simulation results using the DSC control algorithm in the absence of impact disturbances. DCS controller parameters are determined: $k_{11} = k_{12} = k_{13} = k = 13$.

Fig. 2 and Fig. 3 illustrate that in the first stage, when the robot's location is not yet in orbit, the DSC controller acts to advance the robot quickly toward the orbit, with a low transient component. The orbit tracking quality is good, with low deviation:

$$x_e = 0.0152(m); y_e = 0.0623(m); \theta_e = 0.0361(rad),$$

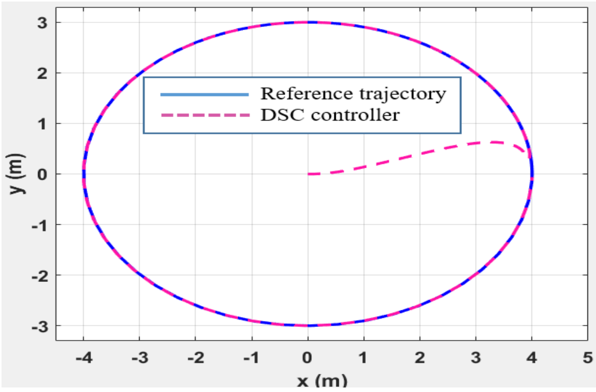

Fig. 2. Representation of circular trajectory with DCS controller

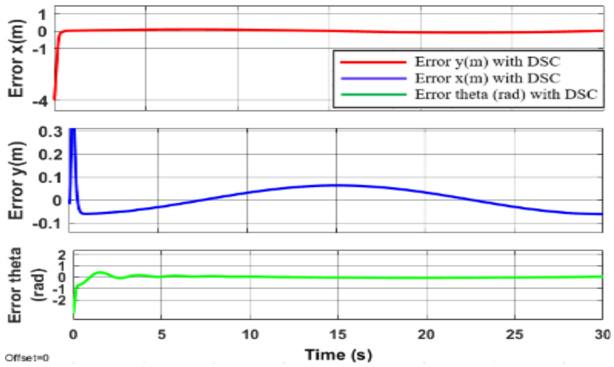

Fig. 3. Representation of circle components tracking error x_e, y_e, θ_e

when the system is in a steady state:

- Simulation results using the DSC control algorithm have impact disturbances.

Suppose the disturbances are random values that affect the system and satisfy the condition $|\tau_d| \leq \ell$. DCS controller parameters are determined: $k_{11} = k_{12} = k_{13} = k = 13$.

Comment on the results: With the DSC controller (shown in Fig. 4 and Fig. 5), the chattering phenomenon is significantly reduced, and the influence of noise is also reduced. The response time is faster when using the input low-pass filter; the tracking error is eliminated. However, to reduce the chattering phenomenon caused by noise, the effect of the sign() function will be approximated by an online artificial NN and compensated in the control law.

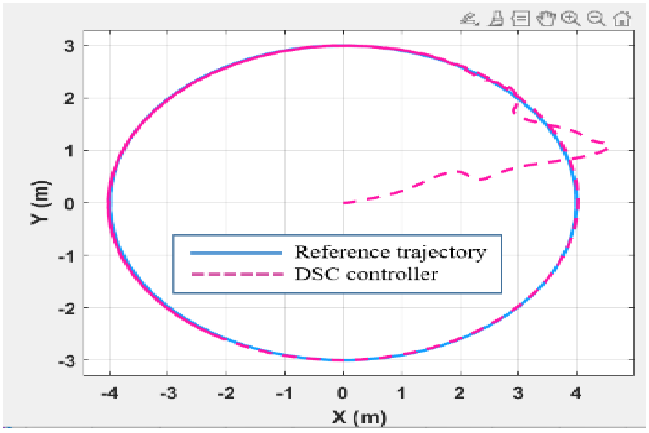

Fig. 4. Representation of circular trajectory with DCS controller

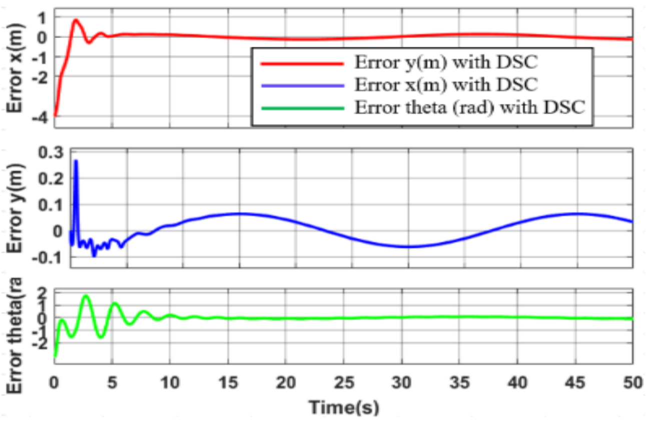

Fig. 5. Representation of circle components tracking error x_e, y_e, θ_e

B. Adaptive Fuzzy Neural Network Dynamic Surface Controller for MWMR (AFNNDCS)

1) Approximating the MWMR model uncertainty using radial neural networks.

In the event of considerable model variation, control quality is no longer guaranteed. As a result, reviewing model deviation and making adjustments in the control component will help to improve the controller's quality. Thus, altering the controller values combined with online unknown control will surely enhance the performance of the MWMR system of controls. Fig. 6 presents a framework diagram for the AFNNDSC measuring control system.

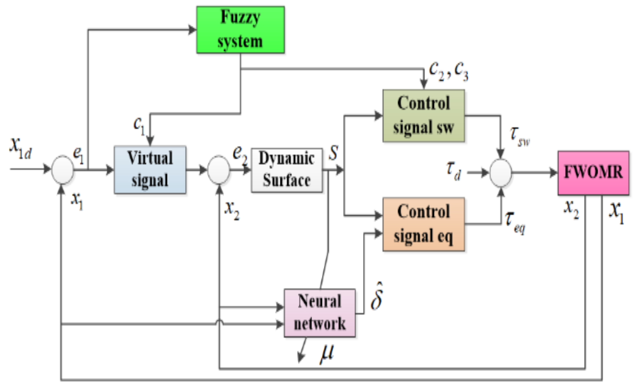

Fig. 6. Controller structure diagram AFNNDCS

The uncertain components are represented by the expression:

$$\Phi = -M^{-1}J^{-1}(MJx_2 - G\,sgn(x_2) + \tau_d) \tag{41}$$

Φ is a 3×1 vector value that contains the MWMR's uncertain components.

$$\Rightarrow \begin{cases} \dot{x}_1 = Nx_2 \\ \dot{x}_2 = \Phi + M^{-1}J^{-1}E\tau \end{cases} \tag{42}$$

The derivative of the sliding surface is calculated using the same techniques as in the preceding section for the controller design:

$$\dot{S} = \lambda\dot{e}_1 + N\dot{e}_2 + \dot{N}e_2 = \lambda\dot{e}_1 + Re_2 + N(\Phi + M^{-1}J^{-1}E\tau - \dot{x}_{2d}) \tag{43}$$

System control signals:

$$\tau = \tau_{eq} + \tau_{sw} \tag{44}$$

$$\tau_{eq} = -E^T(EE^T)^{-1}MJ\left(N^{-1}\left(\lambda\dot{e}_1 + \dot{N}e_2\right) - \dot{x}_{2d} + \hat{\Phi}\right) \tag{45}$$

$$\tau_{sw} = -E^T(EE^T)^{-1}MJ^{-1}(c_2\,sgn(S) + c_3 S) \tag{46}$$

with $\hat{\Phi} = \left[\hat{\Phi}_x \quad \hat{\Phi}_y \quad \hat{\Phi}_\theta\right]^T$ is the output vector of the neural network trained online to approximate the uncertain components of the system.

$x_1 = [x \quad y \quad \theta]^T$ and $x_2 = [v_x \quad v_y \quad \omega \quad \eta]^T$ are the position and velocity vectors of the robot. The proposed NN network is shown in Fig. 7.

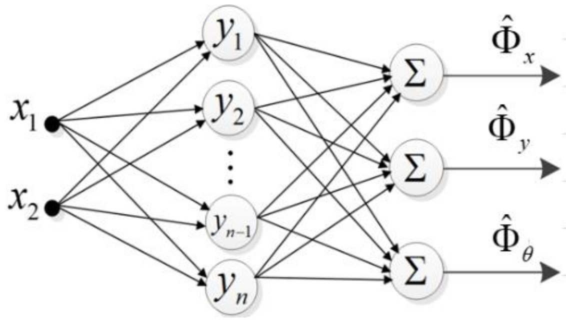

Fig. 7. Radial neural network

$\mu = \begin{bmatrix} \mu_{11} & \mu_{12} & \mu_{13} \\ \mu_{21} & \mu_{22} & \mu_{23} \\ \vdots & \vdots & \vdots \\ \mu_{n1} & \mu_{n2} & \mu_{n3} \end{bmatrix}$, $\gamma = [\gamma_1 \quad \gamma_2 \quad \cdots \quad \gamma_n]^T$ is a vector that contains the neuron nuclei's output values.

The adaptive law of $\widehat{\Phi}$ (47):

$$\Phi = \mu^T\gamma + \varepsilon \text{ and } \widehat{\Phi} = \hat{\mu}^T\gamma \tag{47}$$

Then $\tilde{\mu} = \mu - \hat{\mu}$.

The output of the hidden layer is defined as:

$$\gamma_i = exp\left(-\frac{\|X_1 - \partial_{1i}\|^2 + \|X_2 - \partial_{2i}\|^2}{\psi_i^2}\right) \tag{48}$$

The update rule has the following form:

$$\dot{\hat{\mu}} = \Gamma(\gamma S^T N - \varsigma\|S\|\hat{\mu}) \tag{49}$$

With: Γ is a positive definite square matrix of order; ς is the learning rate of the NN. For (23), (44) and (49), and satisfies:

$$\|S\| \geq \frac{\varepsilon_N + \varsigma\frac{\|\mu\|_F^2}{4}}{c_{3min}} \tag{50}$$

Choose the Lyapunov function:

$$V_2 = \frac{1}{2}S^TS + \frac{1}{2}tr\left(\tilde{\mu}^T\Gamma^{-1}\tilde{\mu}\right) \tag{51}$$

$$\begin{aligned}\Rightarrow \dot{V}_2 &= S^T\dot{S} + tr\left(\tilde{\mu}^T\Gamma^{-1}\dot{\tilde{\mu}}\right)\\ &= S^T\dot{S} + tr\left(\tilde{\mu}^T\Gamma^{-1}(\dot{\mu} - \dot{\hat{\mu}})\right) = S^T\dot{S}\\ &\quad - tr\left(\tilde{\mu}^T\Gamma^{-1}\dot{\hat{\mu}}\right) \quad (\dot{\mu} = 0)\end{aligned} \tag{52}$$

$$\begin{aligned}\Rightarrow \dot{V}_2 = -S^Tc_2\,sgn(S) - S^Tc_3S + S^TN(\Phi - \widehat{\Phi})\\ - tr\left(\tilde{\mu}^T\Gamma^{-1}\dot{\hat{\mu}}\right)\end{aligned} \tag{53}$$

$$\begin{aligned}\Rightarrow \dot{V}_2 = -S^Tc_2\,sgn(S) - S^Tc_3S + S^TN\varepsilon + S^TN\widetilde{\Phi}^T\gamma\\ - tr\left(\tilde{\mu}^T\Gamma^{-1}\dot{\hat{\mu}}\right)\end{aligned} \tag{54}$$

$$\begin{aligned}\Rightarrow \dot{V}_2 = -S^Tc_2\,sgn(S) - S^Tc_3S + S^TN\varepsilon\\ + \varsigma\|S\|\,tr(\tilde{\mu}^T(\mu - \tilde{\mu}))\end{aligned} \tag{55}$$

Apply the inequality Cauchy-Schwarz

$$Tr(\tilde{\mu}^T(\mu - \tilde{\mu})) \leq \|\tilde{\mu}\|_F\|\mu\|_F - \|\tilde{\mu}\|_F^2 \tag{56}$$

$$\begin{aligned}\Rightarrow \dot{V}_2 &\leq -S^Tc_2\,sgn(S) - S^Tc_3S + S^TN\varepsilon\\ &\quad + \varsigma\|S\|(\|\tilde{\mu}\|_F\|\mu\|_F - \|\tilde{\mu}\|_F^2)\\ &\leq -S^Tc_2\,sgn(S) - S^Tc_3S\\ &\quad + \|S\|\varepsilon_N\\ &\quad + \varsigma\|S\|(\|\tilde{\mu}\|_F\|\mu\|_F - \|\tilde{\mu}\|_F^2)\end{aligned} \tag{57}$$

With blocking condition (54), $\dot{V}_2$ become:

$$\dot{V}_2 \leq -S^Tc_2\,sgn(S) - \varsigma\|S\|\left(\|\tilde{\mu}\|_F - \frac{1}{2}\|\mu\|_F\right)^2 \tag{58}$$

2) Fuzzy rule construction for AFNNDSC

The parameter $C_{1i}(i = x, y, \theta)$ is the first output of the fuzzy set and is also a parameter of the sliding surface. The parameters $C_{2i}(i = x, y, \theta)$ and $C_{3i}(i = x, y, \theta)$. Are the second and third outputs of the fuzzy set.

The inputs of the fuzzy regularizer $e_{1x}, e_{1y}, e_{1\theta}$ and derivatives. The input fuzzy sets and output constants are selected through experimentation. The fuzzy sets for the input linguistic variables are shown in Fig. 8 and Fig. 9. The output values of the fuzzy regularizer are shown in Table I.

Table I represents the basic inference rule of the fuzzy controller for two outputs.

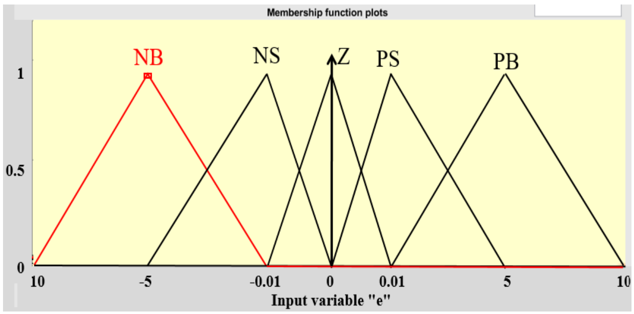

Fig. 8. Fuzzy sets for input e_1

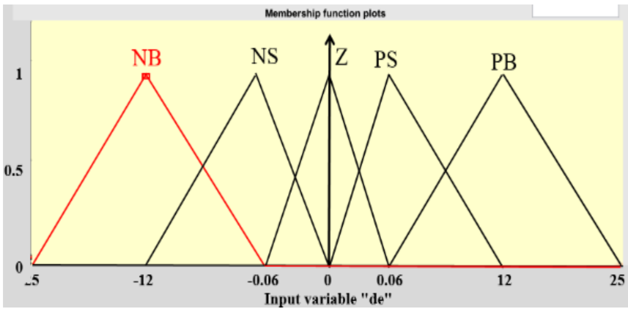

Fig. 9. Fuzzy sets for input $\dot{e}_1$

TABLE I. OUTPUT VALUES C_1, C_2, C_3

Variable output language	C_1	C_2 and C_3
VS	5	25
S	6.5	27.5
M	8	30
B	9.25	32.5
VB	10	35

In this situation, external disturbance torques, specifically Gauss noise (shown in Fig. 10), have a direct effect on the robot's motors, while friction ignores the interference. The coefficient for the chosen sliding surface is: $\lambda = diag(8,8,8)$.

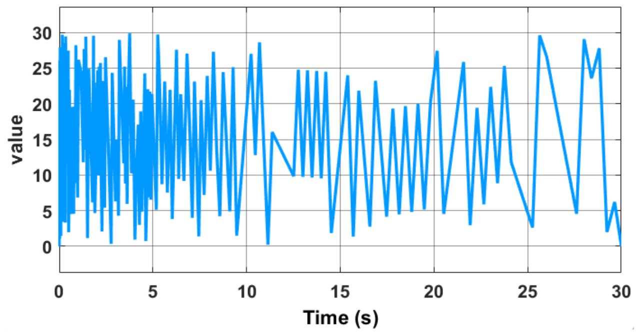

Fig. 10. Torque noise (Nm)

Below are the simulation results in Fig. 11 to Fig. 21:

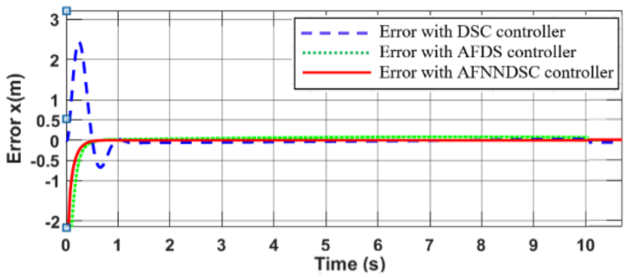

Fig. 11. Represent error on the x-axis

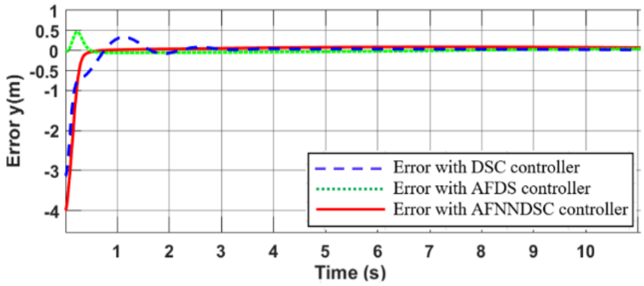

Fig. 12. Represent error on the y-axis

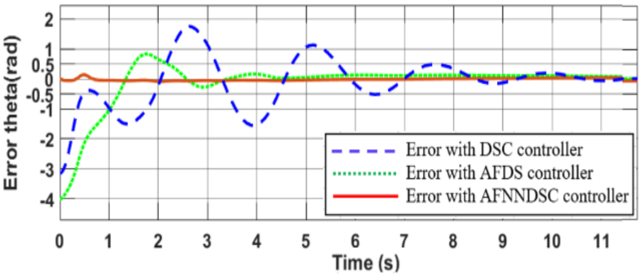

Fig. 13. Representation of deviation angle

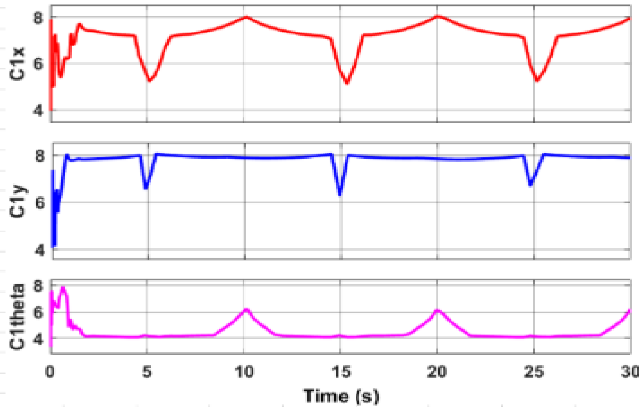

Fig. 14. Representation of parameter C_1

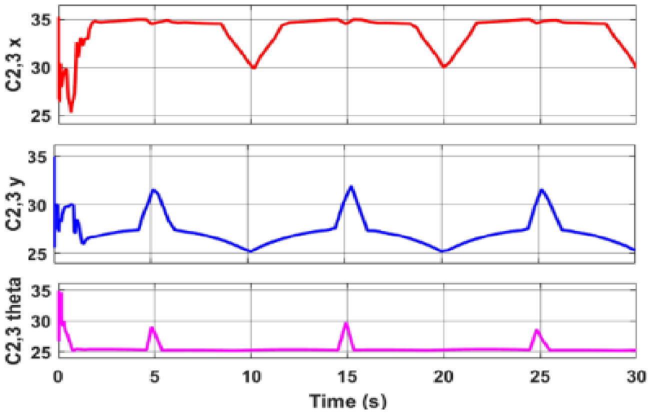

Fig. 15. Representation of parameter C_2, C_3

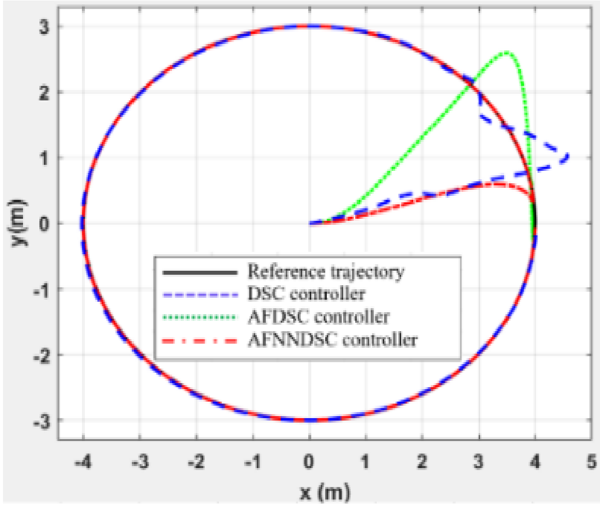

Fig. 16. Circular orbital motion of the robot

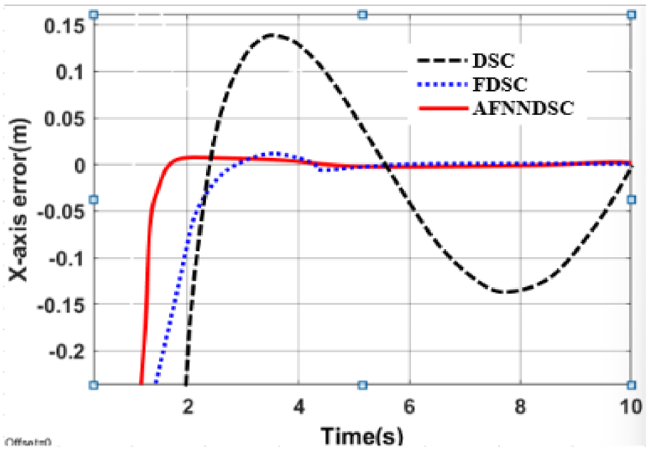

Fig. 17. Represent error on the x-axis

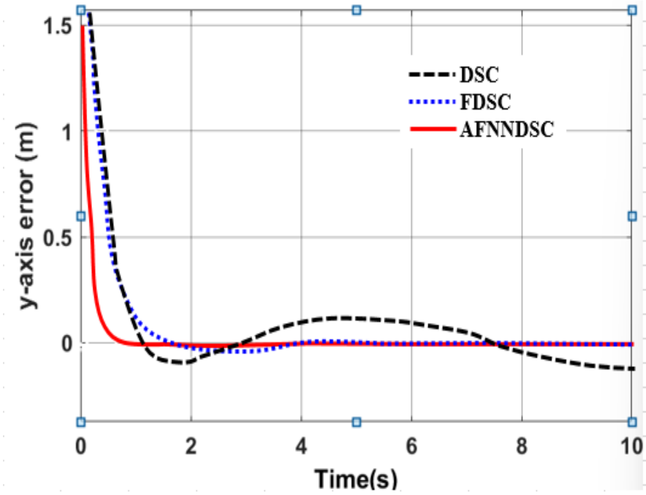

Fig. 18. Represent error on the y-axis

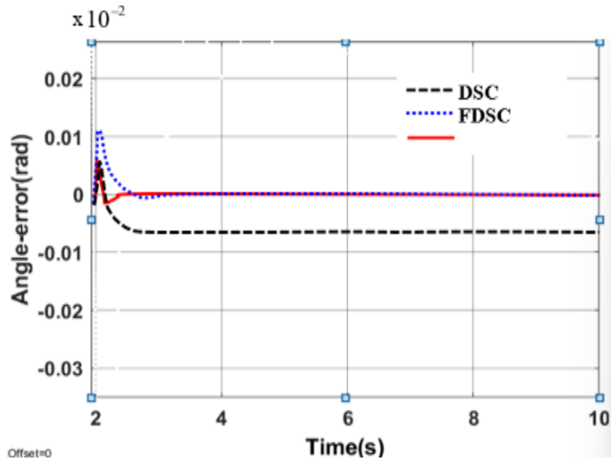

Fig. 19. Representation of deviation angle

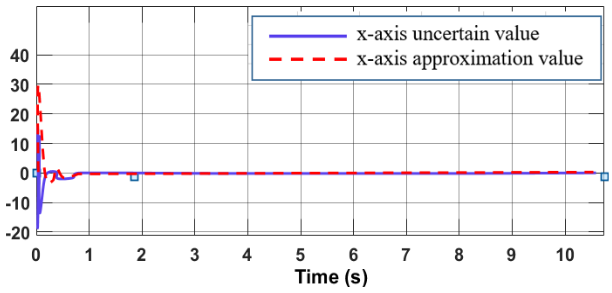

Fig. 20. The value of $\hat{\Phi}_x$ compared to Φ_x

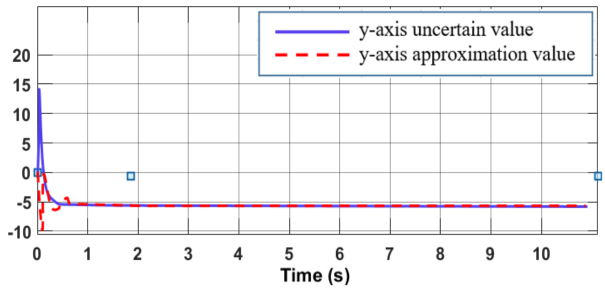

Fig. 21. The value of $\hat{\Phi}_y$ compared Φ_y

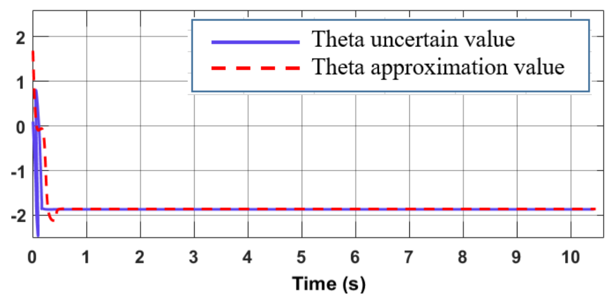

Fig. 22. The value of $\hat{\Phi}_\theta$ compared Φ_θ

The orbital motion error of the three converters is shown in the Table II.

TABLE II. MAXIMUM DEVIATION VALUE WHEN ROBOT FOLLOWS TRAJECTORY

Controllers	Representation of error values		
	x(m)	y(m)	θ (rad)
DCS	0.1352	0.1583	0.00552
AFDCS	0.00126	0.00315	0.000252
AFNNDCS	0.000472	0.000423	0.000294

III. CONCLUSION

The primary purpose of this research is to investigate and propose an artificial adaptive fuzzy sliding surface controlled by adaptive strategies for tracking the motion of MWMR with uncertain constant parameters and external disturbances that alter the mathematical model (1). This algorithm is based on the DSC and the adaptive control structure combines NN and FLS, with NN used to approximate the unknowable nonlinear component of MWMR and the FLS used to change the DSC parameters continually. The closed-loop system's stability was evaluated using Lyapunov principles. Simulation findings demonstrate the mathematical model's reliability, the proposed control system's effectiveness, and the feasibility of practical application. In the near future, the authors will research and test the MWMR autonomous vehicle model in order to test the controllers. This product can be used in practical training for students and graduate students in robotics, automation, control and mechatronics, which are currently in great demand in Vietnam. The development direction is towards practical applications such as manufacturing autonomous robots in warehouses, factories, logistics, environmental monitoring robots in places with toxic ecological conditions, robots serving medical.

REFERENCES

[1] T. T. K. Ly, T. N. HongThai, N. H. Q. Dzung, and N. T. Thanh, "Determination of Kinematic Control Parameters of Omnidirectional AGV Robot with Mecanum Wheels Track the Reference Trajectory and Velocity," *In Advances in Engineering Research and Application. ICERA 2020.*, vol. 178, pp. 319-328, 2021.

[2] L. Schulze, S. Behling, and S. Buhrs, "Development of a Micro Drive - Under Tractor Research and Application," *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 2, pp. 16-28, 2011.

[3] M. Göller, T. Kerscher, J. M. Zöllner, R. Dillmann, M. Devy, T. Germa and F. Lerasle, "Setup and control architecture for an interactive Shopping Cart in human all day environments," *International Conference on Advanced Robotics*, pp. 1-6, 2009.

[4] M. Göller, T. Kerscher, M. Ziegenmeyer, A. Ronnau, J. M. Zöllner, and R. Dillmann, "Haptic Control for the Interactive Behavior Operated Shopping Trolley InBOT," *International Conference on Advanced Robotics*, pp. 1-8, 2009.

[5] D. P. d. Oliveira, W. P. N. d. Reis, and O. M. Junior, "A Qualitative Analysis of a USB Camera for AGV Control," *Sensors*, vol. 4111, no. 19, pp. 1-31, 2019.

[6] A. Gfrerrer, "Geometry and kinematics of the Mecanum wheel," *Computer Aided Geometric Design*, vol. 25, pp. 784–791 , 2008.

[7] M. Szeremeta and M. Szuster, "Neural Tracking Control of a Four-Wheeled Mobile Robot with Mecanum Wheels," *Applied Sciences*, vol. 12, no. 11, pp. 1-21, 2022.

[8] J. Qian, B. Zi, D. Wang, Y. Ma, and D. Zhang, "The Design and Development of an Omni-Directional obile Robot Oriented to an Intelligent Manufacturing System," *Sensors*, vol. 17, no. 9, pp. 1-30, 2017.

[9] C.-C. Tsai, H.-L. Wu, and Y.-R. Lee, "Intelligent Adaptive Motion Controller Design for Mecanum Wheeled Omnidirectional Robots with Parameter Variations," *International Journal of Nonlinear Sciences and Numerical Simulation*, vol. 11, pp. 91-96, 2010.

[10] X. Lu, X. Zhang, G. Zhang, J. Fan, and S. Jia, "Neural network adaptive sliding mode control for omnidirectional vehicle with uncertainties," *ISA Transactions*, vol. 86, pp. 201–214, 2019.

[11] Z. Sun, S. Hu, D. He, W. Zhu, H. Xie, and J. Zheng, "Trajectorytracking control of Mecanum-wheeled omnidirectional mobile robots using adaptive integral terminal sliding mode," *Computers & Electrical Engineering*, vol. 96, no. 107500, pp. 116-128, 2021.

[12] V. Alakshendra and S. S. Chiddarwar, "Adaptive robust control of Mecanum-wheeled mobile robot with uncertainties," *Nonlinear Dynamics*, vol. 87, no. 4, pp. 2147–2169, 2017.

[13] T. Zhao, X. Zou, and S. Dian, "Fixed-time observer-based adaptive fuzzy tracking control for Mecanum-wheel mobile robots with guaranteed transient performance," *Nonlinear Dynamics*, vol. 107, no. 1, pp. 921–937, 2022.

[14] D. U. Rijalusalam and I. Iswanto, "Implementation kinematics modeling and odometry of four omni wheel mobile robot on the trajectory planning and motion control based microcontroller," *Journal of Robotics and Control (JRC)*, vol. 2, no. 5, pp. 448-455, 2021.

[15] D. Mišković, L. Milić, A. Čilag, T. Berisavljević, A. Gottscheber, and M. Raković, " Implementation of Robots Integration in Scaled Laboratory Environment for Factory Automation," *Applied Sciences*, vol. 12, no. 3, pp. 1-21, 2022.

[16] B. Tao, X. Zhao, and H. Ding, "Kinematic modeling and control of mobile robot for large-scale workpiece machining," *Part B: Journal of Engineering Manufacture*, vol. 236, no. 2, pp. 29-38, 2022.

[17] C. P. Sesmero, L. R. Buonocore, and M. D. Castro, "Omnidirectional Robotic Platform for Surveillance of Particle Accelerator Environments with Limited Space Areas," *Applied Sciences*, vol. 11, no. 14, 2021.

[18] I. Iswanto, A. Ma'arif, N. M. Raharja, G. Supangkat, F. Arofiati, R. Sekhar, and D. U. Rijalusalam, "PID-based with Odometry for Trajectory Tracking Control on Four-wheel Omnidirectional Covid-19 Aromatherapy Robot," *Emerging Science Journal*, vol. 5, pp. 157-181, 2021.

[19] P. N. Paraskevopoulos. *Modern Control Engineering*. Engineering & Technology: Marcel Dekker, 2002.

[20] J.-Jacques, E. Slotine, and W. Li. *Applied Nonlinear Control*. Prentice –Hall International Inc: Inc, 1991.

[21] P. A. Iounou and J. Sun. *Robust and Adaptive Control*. Prentice – Hall International: Inc, 1996.

[22] L. Sciavicco and B. Siciliano. *Modeling and control of Robot Manipunator*. International Editions, 1996.

[23] J. J. Craig. *Introduction to Robotics Mechanics & Control*. Addison – Wesley Publishing Company, 1986.

[24] J. Somló, B. Lantos, and T. C. Pham. *Advanced robot control*. Akadémiai Kiadó, 1997.

[25] M. W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Dynamics and Control*. Addison – Wesley Publishing Company, 2004.

[26] F. L. Lewis, D. M. Dawson, and C. T. Abdallah. *Robot manipulator control: theory and practice*. CRC Press, 2003.

[27] J. Angeles, *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms*. Springer, 2013.

[28] D. B. Marghtu, *Mechanisms and Robots Analysis with MATLAB*, Springer, 2009.

[29] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Force control* (pp. 363-405). Springer London, 2009.

[30] S. Jabin, *Robot Learning*, Sciyo, 2010.

[31] P. Corke. *Robotics and control: fundamental algorithms in MATLAB®*. springer Nature, 2021.

[32] P. I. Corke, W. Jachimczyk, and R. Pillat. *Robotics, vision and control: fundamental algorithms in MATLAB* (vol. 73, p. 2). Berlin: Springer, 2011.

[33] T. Wang, C. Tsai, and D. Wang, "Dynamic Control of An Omnidirectional Mobile Platform," *Engineering, Computer Science*, vol. 7, no. 1, pp. 9-18, 2010.

[34] X. Wu, Z. Chen, W. Chen and a. W. Wang, "Research on the Design of Educational Robot with Four-wheel Omni-direction Chassis," *J. Comput*, vol. 29, no. 4, pp. 284–294, 2018.

[35] M. J. Jung, H. S. Kim, S. Kim, and J. H. Kim, "Omnidirectional mobile base OK-II," *Proceeding of the 2000 IEEE International Conference on Robotics and Automation, San Francisco*, pp. 3449-3454, 2000.

[36] M. M. Olsen and H. G. Petersen, "A new method for estimating parameters of a dynamic Robot model," *IEEE Transactions on Robot ics and Automation*, vol. 17, no. 1, pp. 95-100, 2001.

[37] Y. P. Leow, K. H. Low, and W. K. Loh, "Kinematic modelling and analysis of mobile robots with omni-directional wheels," *7th International Conference on Control, Automation, Robotics and Vision, 2002. ICARCV 2002.*, pp. 820-825, 2002.

[38] R. L. Williams, B. E. Carter, P. Gallina, and G. Rosati, "Dynamic model with slip for wheeled Omnidirectional Robots," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 3, pp. 285-293, 2002.

[39] W. K. Loh, K. H. Low, and Y. P. Leow, "Mechatronics design and kinematic modelling of a singularityless Omni-directional wheeled mobile Robot," *Robot ics and Automation, Proceedings. ICRA 03. IEEE International Conference*, vol. 3, pp. 3237-3242, 2003.

[40] Y. Liu, X. Wu, J. J. Zhu, and J. Lew, "Omni-directional mobile Robot controller design by trajectory linearization," *Proceedings of the 2003 American Control Conference*, pp. 3423–3428, 2004.

[41] J. Salih, M. Rizon, S. Yaacob, A. Adom, and M. Mamat, "Designing Omni- Directional Mobile Robot with Mecanum Wheel," *American Journal of Applied Sciences*, vol. 3, no. 5, pp. 1831-1835, 2006.

[42] H. P. Oliveira, A. J. Sousa, P. Moreira, and P. J. Costa, " Precise Modeling of a Four Wheeled Omni-directional Robot," *Proc. 8th Conf. Auton. Robot Syst. Compet*, pp. 57–62, 2008.

[43] Y. Kanayama, Y. Kimura, F. Miyazaki, and T. Noguchi, "A stable tracking control scheme for an autonomous mobile robot," *proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 384-389, 1990.

[44] C.-C. Tsai, L.-B. Jiang, T.-Y. Wang, and T.-S. Wang, "Kinematics Control of an Omnidirectional Mobile Robot," *Proceedings of 2005 CACS Automatic Control Conference Tainan*, pp. 18-19, 2005.

[45] Y.-P. Hsu, C.-C. Tsai, Z.-C. Wang, Y.-J. Feng, and H.-H. Lin, "Hybrid navigation of a four-wheeled tour-guide robot," *2009 ICCAS-SICE*, pp. 4353-4358, 2009.

[46] E. Hashemi, M. G. Jadidi, and O. B. Babarsad, "Trajectory planning optimization with dynamic modeling of four wheeled Omni-directional mobile Robots," *Proc. IEEE Int. Symp. Comput. Intell. Robot. Autom.*, pp. 272–277, 2009.

[47] J. Palacín, E. Rubies, E. Clotet, and D. Martínez, "Evaluation of the path-tracking accuracy of a three-wheeled omnidirectional mobile robot designed as a personal assistant," *Sensors*, vol. 21, no. 21, p. 7216, 2021.

[48] H. P. Oliveira, A. J. Sousa, A. P. Moreira, and P. J. Costa, "Dynamical Models for Omni-Directional Robots With 3 and 4 Wheels," *International Conference on Informatics in Control, Automation and Robotics*, vol. 2, pp. 189–196, 2012.

[49] H. P. Oliveira, A. J. Sousa, A. Moreira, and P. J. Costa, "Modeling and Assessing of Omni-directional Robots with Three and Four Wheels," *Contemporary Robotics: Challenges and Solutions*, pp. 109-138, 2009.

[50] Y. Liu, J. J. Zhu, R. L. Williams, and J. Wu, "Omni-directional mobile Robotcontroller based on trajectory linearization," *Rob. Auton. Syst*, vol. 56, no. 5, pp. 461–479, 2008.

[51] L.-C. Lin and H.-Y. Shih, "Modeling and Adaptive Control of an Omni- Mecanum-Wheeled Robot," *Intell. Control Autom*, vol. 4, no. 2, pp. 166–179, 2013.

[52] C. T. Lee and W. T. Sung, "Controller design of tracking WMR system based on deep reinforcement learning," *Electronics*, vol. 11, no. 6, p. 928, 2022.

[53] M. Korkmaz, Ö. Aydoğdu, and H. Doğan, "Design and Performance Comparison of Variable Parameter Nonlinear PID Controller and Genetic Algorithm Based PID Controller," *Conference Innovations in Intelligent Systems and Applications*, pp. 978-982, 2012.

[54] O. Aydogdu and M. Korkmaz, "A Simple Approach to Design of Variable Parameter Nonlinear PID Controller," *International Conference on Advancements in Information Technology*, vol. 20, pp. 81-85, 2011.

[55] E. Malayjerdi, H. Kalani, and M. Malayjerdi, "Self-Tuning Fuzzy PID Control of a Four-Mecanum Wheel Omni-directional Mobile Platform," *Electrical Engineering (ICEE), Iranian Conference on*, pp. 816-820, 2018.

[56] Q. Jia, C. Chang, S. Liu, L. Zhang, and S. Zhang, "Motion Control of Omnidirectional Mobile Robot Based on Fuzzy PID," *In Chinese Control and Decision Conference (CCDC)*, pp. 5149-515, 2019.

[57] S. Harun and M. F. Ibrahim, "A genetic algorithm based task scheduling system for logisticsservice robots," *Bulletin of Electrical Engineering and Informatics*, vol. 8, no. 1, pp. 206-213, 2019.

[58] 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, pp. 107-128, 221.

[59] X. Wu and Y. Huang, "Adaptive fractional-order non-singular terminal sliding mode control based on fuzzy wavelet neural networks for omnidirectional mobile robot manipulator," *ISA transactions*, vol. 121, pp. 258-267, 2022.

[60] X. Sun, L. Zhang, and J. Gu, "Neural-network based adaptive sliding mode control for Takagi-Sugeno fuzzy systems," *Information Sciences*, vol. 628, pp. 240-253, 2023.

[61] G. G. Rigatos, C. S. Tzafestas, and S. G. Tzafestas, " Mobile Robot motion control in partially unknown environments using a sliding-mode fuzzy-logic controller," *Rob. Auton. Syst*, vol. 33, no. 1, pp. 1–11, 2000.

[62] D. Qian, J. Yi, and D. Zhao, "Control of overhead crane systems by combining sliding mode with fuzzy regulator," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 9320-9325, 2011.

[63] J. Chen, J. Wang, S. Ouyang, and Y. Yang, "Adaptive sliding mode control based on a filter for four-wheel omni-directional mobile robots," *Cybernetics and Information Technologies*, vol. 14, no. 2, pp. 140-153, 2014.

[64] D. Swaroop, J. K. Hedrick, P. P. Yip, and J. C. Gerdes, "Dynamic surface control for a class of nonlinear systems," *IEEE Trans. Automat. Contr*, vol. 45, no. 10, pp. 1893-1899, 2000.

[65] B. Song and J. K. Hedrick. *Dynamic surface control of uncertain nonlinear systems: an LMI approach*. Springer Science & Business Media, 2011.

[66] X.-Y. Luo, Z.-H. Zhu, and X.-P. Guan, "Adaptive Fuzzy Dynamic Surface Control for Uncertain Nonlinear Systems," *International Journal of Automation and Computing*, vol. 6, no. 4, pp. 385-390, 2009.

[67] D. Z. S. Qi and L. W. L. Guo, "Adaptive Dynamic Surface Control of Nonlinear Switched Systems with Prescribed Performance," *J. Dyn. Control Syst*, vol. 24, no. 2, pp. 269–286, 2018.

[68] B. Yoo and W. Ham, "Adaptive fuzzy sliding mode control of nonlinear system," *IEEE Trans. Fuzzy Syst*, vol. 6, no. 2, pp. 315–321, 1998.

[69] S. Dai, C. Wang, and F. Luo, " Identification and Learning Control of Ocean Surface Ship Using Neural Networks," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 4, pp. 801-810, 2012.

[70] H. T. T. Uyen, P. D. Tuan, L. V. Anh, D. Q. True, and P. X. Minh, "PerfOMRance assessment of adaptive neural network dynamic surface controller with adaptive neural network backstepping controller and adaptive neural network sliding mode backstepping controller," *Journal of Military Science and Technology*, 2017.

[71] Zayer, "Fuzzy Logic Control Of Crane System," *The Iraqi Journal For Mechanical And Material Engineering*, vol. 11, no. 3, pp. 437–447, 2011.

[72] E. Hashemi, M. G. Jadidi, and N. G. Jadidi, "Model-based PI fuzzy control of four-wheeled Omni-directional mobile Robots," *Rob. Auton. Syst.*, vol. 59, no. 11, pp. 930–942, 2011.

[73] J. Park and I. W. Sandberg, "Universal Approximation Using Radial-Basis-Function Networks," in *Neural Computation*, vol. 3, no. 2, pp. 246-257, 1991.

[74] B. Miao and T. Li, "Direct adaptive neural network control of a class of nonlinear systems," *2014 International Joint Conference on Neural Networks (IJCNN)*, pp. 2870-2874, 2014.

[75] J. Wang, J. Chen, S. Ouyang, and Y. Yang, "Trajectory tracking control based on adaptive neural dynamics for four-wheel drive Omnidirectional mobile robots," *Engineering Review*, vol. 34, no. 3, pp. 237-249, 2014.

[76] L. Yu, S. Fei, and G. Yang, "A Nơ ron Network Approach for Tracking Control of Uncertain Switched Nonlinear Systems with Unknown Dead-Zone Input," *Circuits, Syst. Signal Process*, vol. 34, no. 8, pp. 2695–2710, 2015.

[77] S. N. Huang, K. K. Tan, and T. H. Lee, "Adaptive motion control using neural network approximations," *Automatica*, vol. 38, no. 2, pp. 227–233, 2002.

[78] J. Xu, M. Zhang, and J. Zhang, "Kinematic model identification of autonomous Mobile Robot using dynamical recurrent neural networks," *2005 IEEE International Conference Mechatronics and Automation*, vol. 3, pp. 1447-1460, 2005.

[79] P. Petrehuş, Z. Lendek, and P. Raica, "Fuzzy modeling and design for a 3D crane," *IFAC Proc*, vol. 46, no. 20, pp. 479–484, 2013.

[80] V. T. Ha, T. T. Thuong, N. T. Thanh, and V. Q. Vinh, "Research on Some Control Algorithms to Compensate for the Negative Effects of Model Uncertainty Parameters, External Interference, and Wheeled Slip for Mobile Robot," *Actuators*, vol. 31, no. 13, pp. 1-31, 2024.