SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I4P107

Volume 11 Issue 4, 55-63, April 2024

Original Article

An Effective Trajectory Tracking Control Method for Delta Robot Based on Backstepping Sliding Mode Control Techniques

Hung Pham Van[1], Xuan Minh Dinh[2], Minh Pham Van[3], Nam Pham Van[4], Hai Le Xuan[5], Nam Khanh Nguyen[6]

[1,2,3,4,6]*Ha Noi University of Industry, Ha Noi, Viet Nam.*
[5]*VNU - International School, Ha Noi, Viet Nam.*

[1]*Corresponding Author : phamvanhung@haui.edu.vn*

Received: 06 February 2024 Revised: 06 March 2024 Accepted: 05 April 2024 Published: 30 April 2024

Abstract *- In this paper, we propose the application of the backstepping sliding mode control method for the Delta robot, aiming to achieve desired motion trajectories in a short time and maintain system stability, even in the presence of unknown external disturbances affecting joint torques. The stability of the system is rigorously demonstrated based on the Lyapunov stability theory. The performance of the controller is evaluated through numerical simulations using the Matlab&Simulink tool, with a figure-eight trajectory, and compared with two other control algorithms, including classical PD and Backstepping. The results show that the proposed controller exhibits excellent performance, allowing for precise control of the Delta robot's motion along the desired trajectory with minimal error, short response time, and improved stability compared to both PD and backstepping controllers, especially when the system is subjected to unknown external disturbances.*

Keywords *- Delta robot, Asymptotic stability, Backstepping sliding mode control, Tracking control.*

1. Introduction

The Delta 3-DOF (Degrees of Freedom) robot is a type of parallel robot with high rigidity, flexibility, speed, cost-effectiveness, and precision. As of the current time, Delta robot systems have become popular and are being used in various fields such as medicine, education, and particularly in the manufacturing industry. Consequently, the Delta 3-DOF robot is attracting considerable attention from researchers, especially in the control, mechanical, and robotics fields.

Currently, trajectory tracking control problems of Delta robots have been extensively studied worldwide, with various algorithms being applied to improve control quality. Some notable methods include the classical PID (Proportional-Integral-Derivative) control mentioned in [1], nonlinear control methods such as Backstepping [2], sliding mode control [3], and intelligent control methods [4-7], as well as Hedge algebras method [8].

In addition, for the field of industrial robot control in general, other advanced control methods such as optimization [9], Model Predictive Control (MPC) [10, 11], reinforcement learning [12], deep learning [13], and deep reinforcement learning [14] have also been applied. The classical PID control method [1] has advantages in terms of simple and flexible computation. However, tuning the control parameters in the PID controller can be relatively complex, especially for nonlinear objects like the Delta robot. Therefore, some studies [4, 5] have proposed the use of Recurrent Fuzzy Neural Networks (RFNN) to automatically adjust the PID controller parameters, thereby improving the trajectory tracking quality of the system.

Another adaptive control approach [6] utilizes Radial Basis Function (RBF) neural networks with online learning capability to estimate unknown nonlinear components of the model, aiming to enhance control performance. In [7], the authors employed an adaptive Artificial B-Spline Neural Network (BSNN) to achieve online training and improve trajectory tracking of the Delta 3-DOF robot. However, in general, neural network-based methods require a deep understanding of network architectures.

Control techniques based on Hedge Algebras [8] require experienced designers to provide parameter adjustment rule tables. Control methods based on optimization techniques [9], Model Predictive Control (MPC) [10, 11], and machine learning techniques [12, 14] for robot control design do not rely on a model but instead rely on machine learning techniques, enabling automatic adjustment and improvement of control quality over time. However, these methods may require a large amount of training data, making them more complex to implement and tune compared to conventional control methods.

Based on the above analysis, we propose applying the Backstepping sliding mode control method to the Delta robot to help achieve the desired motion trajectory in a short time and maintain system stability, even when influenced by unknown external disturbances affecting joint torques. This method stands out for its robustness due to the advantages of sliding mode control and simplicity in design through the use of the recursive Lyapunov function of the Backstepping technique. We have verified this method through simulation and found that our proposed method significantly outperforms pure Backstepping [2] and classic PD control with manually selected proportional and derivative parameters [4], especially in the presence of disturbances.

Furthermore, this method requires fewer computations compared to optimization methods [9] and MPC [10, 11]. Moreover, it does not rely on the designer's experience like Hedge algebras-based research [8]. Additionally, the Backstepping sliding mode controller does not rely on specific data or require the same level of complexity as machine learning-based methods [12-14] and neural networks [4-7]. The remaining parts of the study are organized as follows: Section 2 presents the mathematical model of the Delta parallel robot. Section 3 proposes the Backstepping sliding mode controller. Section 4 provides simulation results and discussions. Finally, Section 5 concludes and suggests future directions.

2. The Mathematical Model of the Delta Parallel Robot

The mathematical model of the 3-DOF industrial parallel Delta robot has been extensively presented in prior studies [1-5]. The robot's structure, as depicted in Figures 1 and 2, consists of a fixed base A, a moving platform B, three arms $A_iB_i (i = 1,2,3)$ and three fore-arms $B_iE_i (i = 1,2,3)$, with each forearm having a parallelogram structure. To simplify the model, these structures are replaced by equivalent rigid bars with corresponding lengths.

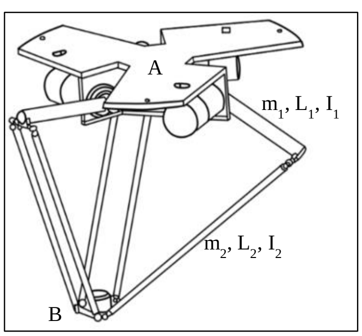

Fig. 1 3-DOF parallel delta robot [6]

The two points B_i and E_i, each with a mass of m_b are considered in the dynamic model shown in Figure 2. The dynamic model consists of four rigid bodies, as follows: the link A_iB_i rotates around axes perpendicular to the OA_iB_i plane, a mass of m_1 at point tại A_iB_i, three-point masses $m_b = 0.5m_2$ at point B_i and the moving platform with a mass of $m_p + 3m_b$. Here, m_P represents the mass of the moving platform. Additionally, moment forces $\tau_i (i = 1,2,3)$ are applied on the link A_iB_i, corresponding to the motor torques.

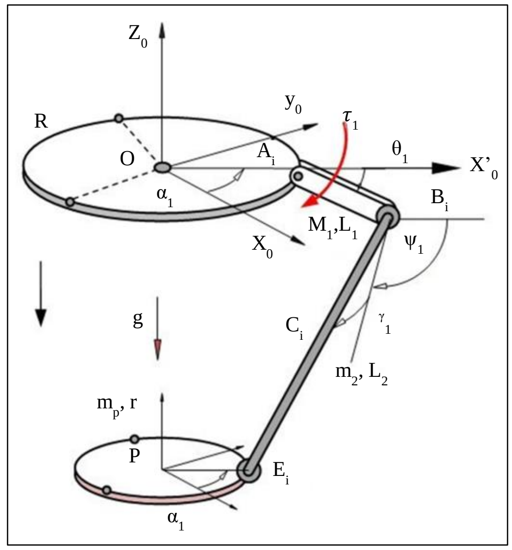

Fig. 2 Dynamic model of the parallel delta robot [4, 5]

The selected state vector for establishing the motion equations of the Delta robot is:

$$s = [s_a^T \quad s_P{}^T]^T = [\theta_1 \quad \theta_2 \quad \theta_3 \quad x_P \quad y_P \quad z_P]^T \quad (1)$$

Where:
$s_a = [\theta_1 \quad \theta_2 \quad \theta_3]^T$ denotes the coordinates of the active joints (angle of the arm of the robot).
$s_P = [x_P \quad y_P \quad z_P]^T$ represents the coordinates of the center of the moving platform.

The system of kinematic equations for the parallel Delta robot is established according to the research [4, 5] as follows:

$$\begin{aligned} f_1 &= L_2^2 - (cos\alpha_1(R - r) + L_1 cos\alpha_1 cos\theta_1 - x_P)^2 \\ &- (sin\alpha_1(R - r) + L_1 sin\alpha_1 cos\theta_1 - y_P)^2 \\ &-(L_1 sin\theta_1 + z_P)^2 = 0 \\[1em] f_2 &= L_2^2 - (cos\alpha_2(R - r) + L_1 cos\alpha_2 cos\theta_2 - x_P)^2 \\ &- (sin\alpha_2(R - r) + L_1 sin\alpha_2 cos\theta_2 - y_P)^2 \qquad (2) \\ &-(L_1 sin\theta_2 + z_P)^2 = 0 \\[1em] f_3 &= L_2^2 - (cos\alpha_3(R - r) + L_1 cos\alpha_3 cos\theta_3 - x_P)^2 \\ &- (sin\alpha_3(R - r) + L_1 sin\alpha_3 cos\theta_3 - y_P)^2 \\ &-(L_1 sin\theta_3 + z_P)^2 = 0 \end{aligned}$$

The system of motion equations for the parallel Delta robot is established based on the research [4, 5].

$$(I_{I_y}+m_bL_1^2)\ddot{\theta}_1 = gL_1\left(\frac{1}{2}m_1+m_b\right)cos\theta_1+\tau_1-2\lambda_1L_1(sin\theta_1(R-r)-cos\alpha_1sin\theta_1x_P-sin\alpha_1sin\theta_1y_P-cos\theta_1z_P) \quad (3)$$

$$(I_{I_y}+m_bL_1^2)\ddot{\theta}_2 = gL_1\left(\frac{1}{2}m_1+m_b\right)cos\theta_2+\tau_2 -2\lambda_2L_1(sin\theta_2(R-r)-cos\alpha_2sin\theta_2x_P -sin\alpha_2sin\theta_2y_P-cos\theta_2z_P) \quad (4)$$

$$(I_{I_y}+m_bL_1^2)\ddot{\theta}_3 = gL_1\left(\frac{1}{2}m_1+m_b\right)cos\theta_3+\tau_3 -2\lambda_3L_1(sin\theta_3(R-r)-cos\alpha_3sin\theta_3x_P -sin\alpha_3sin\theta_3y_P-cos\theta_3z_P) \quad (5)$$

$$(m_P+3m_b)\ddot{x}_P = -2\lambda_1\begin{pmatrix}cos\alpha_1(R-r)\\+L_1cos\alpha_1cos\theta_1-x_P\end{pmatrix} -2\lambda_2(cos\alpha_2(R-r)+L_1cos\alpha_2cos\theta_2-x_P) -2\lambda_3(cos\alpha_3(R-r)+L_1cos\alpha_3cos\theta_3-x_P) \quad (6)$$

$$(m_P+3m_b)\ddot{y}_P = -2\lambda_1\begin{pmatrix}sin\alpha_1(R-r)\\+L_1sin\alpha_1cos\theta_1-y_P\end{pmatrix} -2\lambda_2(sin\alpha_2(R-r)+L_1sin\alpha_2cos\theta_2-y_P) -2\lambda_3(sin\alpha_3(R-r)+L_1sin\alpha_3cos\theta_3-y_P) \quad (7)$$

$$(m_P+3m_b)\ddot{z}_P = -(m_P+3m_b)g+2\lambda_1(z_P+L_1sin\theta_1) +2\lambda_2(z_P+L_1sin\theta_2)+2\lambda_3(z_P+L_1sin\theta_3) \quad (8)$$

With R as the upper disc radius, r as the lower disc radius, L_1 as the upper arm length, L_2 as the lower arm length and $\alpha_1, \alpha_2, \alpha_3$ respectively equal to 0, 120, and 240 degrees. By combining Equations 3 to 8, we obtain the dynamic equation of the Delta robot in matrix form, expressed as follows:

$$M(s)\ddot{s}+C(s,\dot{s})\dot{s}+g(s)+\Phi_s^T(s)\lambda+d(s,\dot{s})=\tau \quad (9)$$

$Where,\ \lambda \in \mathbb{R}^{3\times1}$ is the Lagrange multiplier vector, s is the system state vector, $M(s) \in \mathbb{R}^{6\times6}$ is the positive definite symmetric generalized inertia matrix, $C(s) \in \mathbb{R}^{6\times6}$ is the centrifugal and Coriolis matrix, $g(s) \in \mathbb{R}^{6\times1}$ is the composite vector of the coupling, centrifugal, and gravity components.

$\Phi_s(s) = \frac{\partial f}{\partial s} \in \mathbb{R}^{3\times6}$ is the Jacobian matrix of the kinematic equations, $d(s,\dot{s}) \in \mathbb{R}^{6\times1}$ is the vector containing the unknown force components, $\tau = \begin{bmatrix}\tau_a\\ \Theta_{3\times1}\end{bmatrix} \in \mathbb{R}^{6\times1}$ is the control signal vector, with $\tau_a = [\tau_1, \tau_2, \tau_3]^T$ is the vector representing the moment forces generated by the three motors connected to the robot's joints and $\Theta_{3\times1} \in \mathbb{R}^{3\times1}$, is the vector containing all non-zero elements. Additionally, the symbol $f = [f_1 \quad f_2 \quad f_3]^T$ represents the vector representing the motion coupling of the Delta robot, I_{Iy} is the inertia tensor in Equations 3 to 8.

Specifically, in this case, the components in Equation (9) take the following values:

$$M(s) = diag\begin{pmatrix}I_{Iy}+m_bL_1^2, I_{Iy}+m_bL_1^2, I_{Iy}+m_bL_1^2,\\ ,m_P+3m_b, m_P+3m_b, m_P+3m_b\end{pmatrix} \quad (10)$$

$$C(s,\dot{s}) = \Theta_{6\times6}, \lambda = [\lambda_1 \quad \lambda_2 \quad \lambda_3]^T \quad (11)$$

$$\Phi_s(s) = \frac{\partial f}{\partial s} = \begin{bmatrix}\frac{\partial f_1}{\partial \theta_1} & \cdots & \frac{\partial f_1}{\partial z_P}\\ \vdots & \ddots & \vdots\\ \frac{\partial f_3}{\partial \theta_1} & \cdots & \frac{\partial f_3}{\partial z_P}\end{bmatrix}, \quad (12)$$

$$\tau = [\tau_1 \quad \tau_2 \quad \tau_3 \quad 0 \quad 0 \quad 0]^T$$

$$g(s) = \begin{bmatrix}-gL_1\left(\frac{m_1}{2}+m_b\right)cos\theta_1,\\ -gL_1\left(\frac{m_1}{2}+m_b\right)cos\theta_2,\\ -gL_1\left(\frac{m_1}{2}+m_b\right)cos\theta_3,\\ 0,0,(m_P+3m_b)g\end{bmatrix}^T \quad (13)$$

3. Backstepping Sliding Mode Controller

In this study, we propose to apply the Backstepping sliding mode control method to the Delta robot to achieve the desired motion trajectory of the moving platform within a short time and maintain system stability. This method stands out for its robustness due to the advantages of Sliding Mode Control (SMC) and its simplicity in design by utilizing the recursive Lyapunov function approach of the Backstepping technique. To implement this method, firstly, in Section (3.1), we restructured the dynamic model of the Delta robot in Equation (2) into a strict-feedback canonical form. Then, in Section (3.2), by constructing feedback control laws for each subsystem, we establish a unified control signal for the entire system. The general structure diagram of the control system is illustrated in Figure 3.

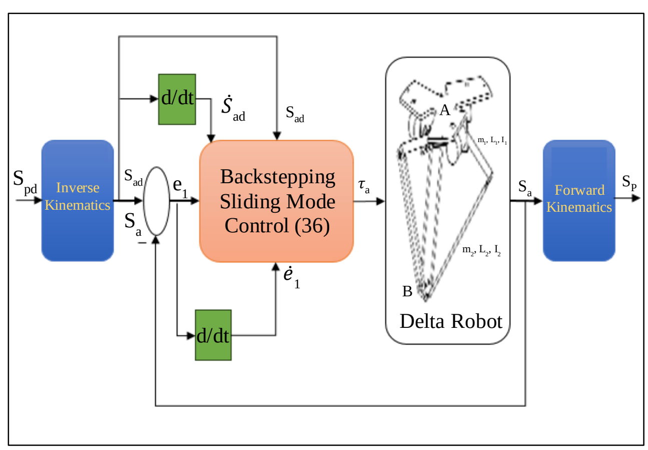

Fig. 3 General structure diagram of the backstepping sliding mode control system

3.1. Restructuring the System Model of the Parallel Delta Robot

Let Φ_a be the Jacobian matrix corresponding to $s_a = [\theta_1 \quad \theta_2 \quad \theta_3]^T$ and Φ_P be the Jacobian matrix corresponding to $s_P = [x_P \quad y_P \quad z_P]^T$, from Equation (12), we have:

$$\Phi_\alpha = \begin{bmatrix} \frac{\partial f_1}{\partial \theta_1} & \cdots & \frac{\partial f_1}{\partial \theta_3} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_3}{\partial \theta_1} & \cdots & \frac{\partial f_3}{\partial \theta_3} \end{bmatrix}, \Phi_\alpha = \begin{bmatrix} \frac{\partial f_1}{\partial x_P} & \frac{\partial f_1}{\partial y_P} & \frac{\partial f_1}{\partial z_P} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_3}{\partial x_P} & \cdots & \frac{\partial f_3}{\partial z_P} \end{bmatrix} \quad (14)$$

Let $R = \begin{bmatrix} I_3 \\ -\Phi_p^{-1}\Phi_a \end{bmatrix} \in \mathbb{R}^{6\times 3}$ with $I_3 \in \mathbb{R}^{3\times 3}$ being the identity matrix. Multiplying R^T both sides of Equation (9) by, we obtain:

$$R^T\big(M(s)\ddot{s} + C(s,\dot{s})\dot{s} + g(s) + \Phi_s^T(s)\lambda + d(s,\dot{s})\big) = R^T\tau = \tau_a \quad (15)$$

From the system of kinematic equations in (2) of the system, we have:

$$\dot{f} = \Phi_a \dot{s}_a + \Phi_p \dot{s}_P = 0 \quad (16)$$

From (16), with Φ_P the non-singular matrix, we have $\dot{s}_p = -\Phi_P^{-1}\Phi_a\dot{s}_a$. Therefore, we can deduce:

$$\dot{s} = R\dot{s}_a \text{ and } \ddot{s} = R\ddot{s}_a + \dot{R}\dot{s}_a \quad (17)$$

Substituting (17) into (15), we obtain:

$$R^T\big(M(s)\big(R\ddot{s}_a + \dot{R}\dot{s}_a\big) + C(s,\dot{s})\dot{s} + g(s) + \Phi_s^T(s)\lambda + d(s,\dot{s})\big) = \tau_a \quad (18)$$

We observe that $R^T\Phi_s^T(s) = 0$, so we introduce new variables:

$$\bar{M} = R^T M(s) R, \bar{C} = R^T\big(M(s)\dot{R} + C(s,\dot{s})R\big), \\ ,\bar{d} = R^T d(s,\dot{s}), \bar{g} = R^T g(s) \quad (19)$$

In this case, Equation 18 can be rewritten as:

$$\bar{M}\ddot{s}_a + \bar{C}\dot{s}_a + \bar{g} + \bar{d} = \tau_a \quad (20)$$

From the dynamic model of the parallel Delta robot expressed in (20), we can deduce:

$$\ddot{s}_a = \bar{M}^{-1}\big(-\bar{g} - \bar{d} - \bar{C}\dot{s}_a\big) + \bar{M}^{-1}\tau_a \quad (21)$$

Now, we proceed to introduce the following variables:

$$x_1 = s_a, x_2 = \dot{s}_a, F = \bar{M}^{-1}\big(-\bar{g} - \bar{d} - \bar{C}\dot{s}_a\big) \quad (22)$$

From here, we can represent the system model in the form of strict-feedback as follows:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = F + \bar{M}^{-1}\tau_a \end{cases} \quad (23)$$

3.2. Designing the Backstepping Sliding Mode Controller

Step 1: Constructing the virtual control law for the first subsystem $\dot{x}_1 = x_2$.

Firstly, we introduce new state variables:

$$e_1 = x_1 - x_{1d} \quad (24)$$

Where, x_{1d} is a vector representing the desired coordinates of the active joints. By differentiating Equation (24) with respect to time, we obtain:

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

Next, we select a Lyapunov function for the first subsystem as follows:

$$V_1 = \frac{1}{2}{e_1}^T e_1 \quad (26)$$

Taking the derivative of Equation (26) with respect to time and combining it with Equation (25), we have:

$$\dot{V}_1 = {e_1}^T\dot{e}_1 = {e_1}^T(\dot{x}_1 - \dot{x}_{1d}) = {e_1}^T(x_2 - \dot{x}_{1d}) \quad (27)$$

At this point, we introduce additional state variables:

$$e_2 = x_2 - \alpha_1 \quad (28)$$

Where α_1 represents the virtual control signal for the first subsystem. Choosing the virtual control signal α_1 such that:

$$\alpha_1 = -c_1 e_1 + \dot{x}_{1d} \quad (29)$$

Where $c_1 \in \mathbb{R}^{3\times 3}$ is a positive definite symmetric constant matrix. By differentiating Equation (28) with respect to time, we obtain:

$$\dot{e}_2 = \dot{x}_2 - \dot{\alpha}_1 \quad (30)$$

Step 2: Design the control law for the second subsystem using the sliding mode control method. This control law is also applicable to the entire system (23).

Defining the total sliding surface for the entire system as follows:

$$S_2 = \mu e_1 + e_2 \quad (31)$$

Where $\mu \in \mathbb{R}^{3\times3}$ is a positive definite constant matrix.

By differentiating Equation 31 with respect to time, we obtain, we get:

$$\dot{S}_2 = \mu \dot{e}_1 + \dot{e}_2 \quad (32)$$

Choosing a Lyapunov function for the entire system that satisfies:

$$V_2 = V_1 + \frac{1}{2} {S_2}^T S_2 \quad (33)$$

Taking the derivative of Equation (33) with respect to time and combining it with Equation (27), we have:

$$\dot{V}_2 = {e_1}^T \dot{e}_1 + {S_2}^T \dot{S}_2 \quad (34)$$

Substituting (30), (32) into (34) and combining (23) and (31), we obtain:

$$\dot{V}_2 = -{e_1}^T c_1 e_1 + {e_1}^T e_2 + {S_2}^T (\mu \dot{e}_1 + F + \bar{M}^{-1} \tau_a - \dot{\alpha}_1) \quad (35)$$

The control signal for the entire system is designed to satisfy:

$$\tau_a = \bar{M}(-S_2({e_1}^T e_2 + {e_2}^T c_2 e_2) - \eta sgn(S_2) + \dot{\alpha}_1 - \mu \dot{e}_1 - F) \quad (36)$$

To mitigate the chattering phenomenon, the function $sgn(S_2)$ in (36) is replaced by $\mathrm{sat}(S_2)$. In this case, Equation (36) can be rewritten as:

$$\tau_a = \bar{M}(-S_2({e_1}^T e_2 + {e_2}^T c_2 e_2) - \eta sat(S_2) + \dot{\alpha}_1 - \mu \dot{e}_1 - F)$$

With the unified control signal for the entire system (36), the derivative of the Lyapunov function now satisfies:

$$\dot{V}_2 = -{e_1}^T c_1 e_1 - {e_2}^T c_2 e_2 - {S_2}^T \eta sgn(S_2) \leq 0 \quad (37)$$

Where $c_2 \in \mathbb{R}^{3\times3}$ is a positive definite symmetric constant matrix. From Equation (37), the system satisfies stability according to the Lyapunov principle.

4. The Simulation Results

In this study, we present numerical simulation results using the Matlab/Simulink tool to validate the effectiveness of the proposed Backstepping sliding mode control applied to the trajectory tracking problem of a parallel Delta robot. The algorithm has been verified and compared with other control methods, including Backstepping control in [2] and the classic Proportional Derivative (PD) control with gain parameters selected according to [4]. We conducted these simulations with two different scenarios using an 8-shaped trajectory. The specific simulation scenarios are presented as follows:

Scenario 1: This scenario is regarded as the simplest, assuming that the dynamic parameters of the robot are accurately known and that the system is not affected by any external disturbances.

Scenario 2: In this scenario, the system had to face significant challenges as it was subjected to unknown external disturbances that affected the joint torques of the system. These disturbances varied over time and are depicted in Figure 4.

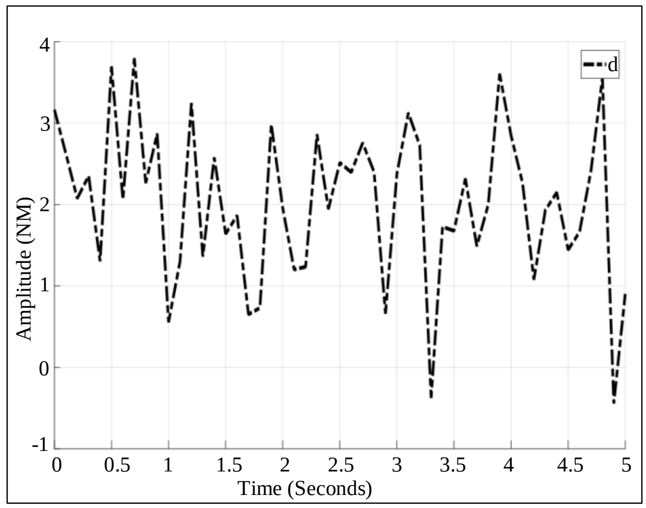

Fig. 4 Unknown external disturbances impacting the joint torques of the system

In order to ensure objectivity in evaluating and comparing the results between control methods, the dynamic model parameters of the Delta robot are taken from the study [6], which include:

$$\alpha_1 = 0(rad), \alpha_2 = 2\pi/3(rad), \alpha_3 = 4\pi/3(rad),$$
$$g = 9.81(m/s^2), m_1 = 0.416(kg), m_b = 0.195(kg),$$
$$m_p = 0.3(kg), R = 0.266(m), r = 0.04(m),$$
$$L_1 = 0.3(m), L_2 = 0.8(m), m_2 = 2 \times 0.195(kg)$$

The controller parameters are selected as follows:

$$\mu = \mathrm{diag}([10,4,4]);\ \eta = \mathrm{diag}([850,150,150]);$$

$$c_1 = \mathrm{diag}([40,5,5]);\ c_2 = \mathrm{diag}([5,0.7,0.7])$$

In addition, the trajectory equation for the 8-shaped motion is given by:

$$x_d = 0.1 + 0.2\cos(2t)\,;\ y_d = 0.1 + 0.2\cos(2t)\sin(2t)\,;$$

$$z_d = -0.7$$

4.1. Simulation Results for Scenario 1

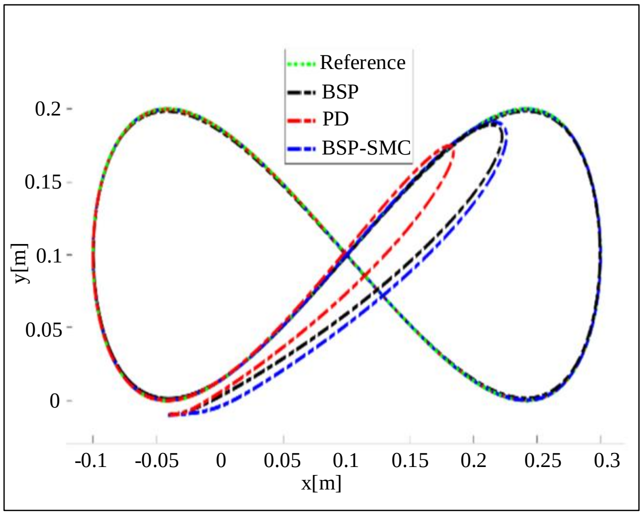

Fig. 5 Trajectory response of the delta robot without disturbances

The simulation results in the case where the system model is not affected by disturbances are shown in Figure 5, Figure 6(a), Figure 6(b), Figure 6(c) and Figure 7(a), Figure 7(b), Figure 7(c). The results demonstrate that the proposed Backstepping Sliding Mode Control algorithm (BSP-SMC) exhibits superior control performance compared to the other two control methods, Backstepping (BSP) and PD.

Specifically, concerning the convergence time for joints 2 and 3, the Backstepping control takes approximately 0.6 seconds. In comparison, the Backstepping sliding mode control achieves a faster response of about 0.1 seconds for joint 1 instead of around 0.3 seconds for Backstepping. The performance of the PD control is also good, but the system's response time is longer, approximately 0.9 seconds for all three joints. It can be observed that the Backstepping sliding mode control not only provides a shorter response time but also exhibits better stability compared to the other two control methods.

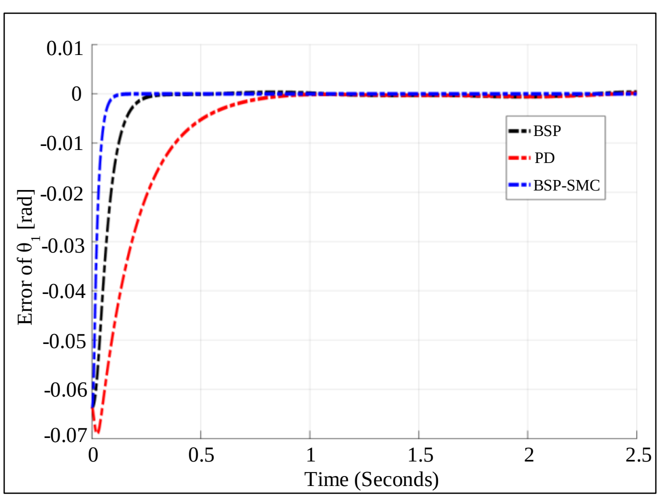

Fig. 6(a) Joint 1 angle error of the delta robot without disturbances

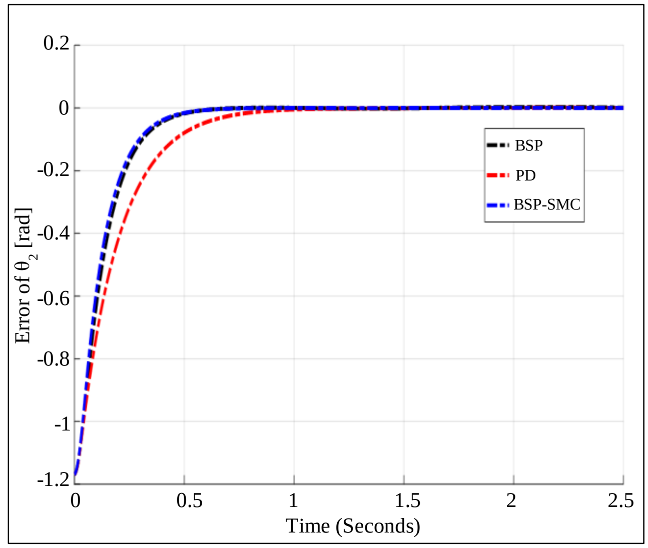

Fig. 6(b) Joint 2 angle error of the delta robot without disturbances

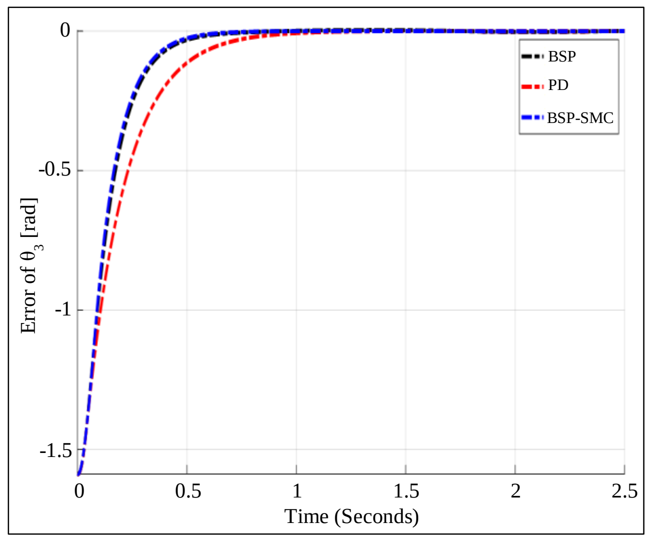

Fig. 6(c) Joint 3 angle error of the delta robot without disturbances

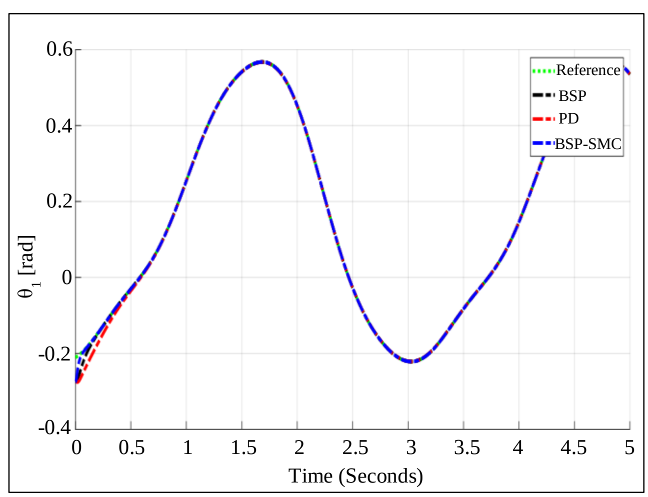

Fig. 7(a) Joint 1 angle trajectory response of the delta robot without disturbances

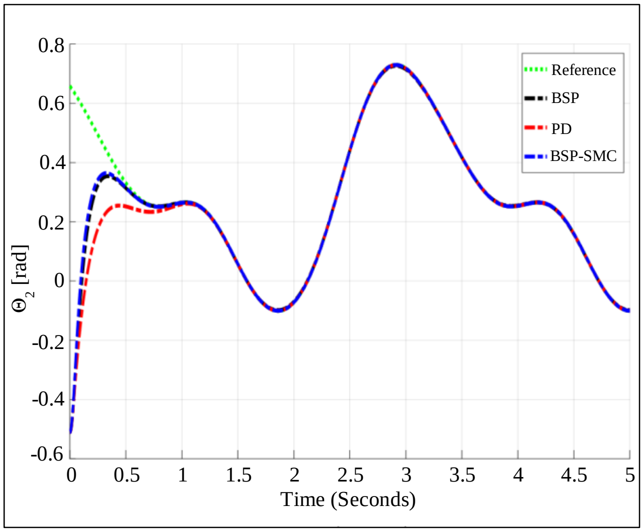

Fig. 7(b) Joint 2 angle trajectory response of the delta robot without disturbances

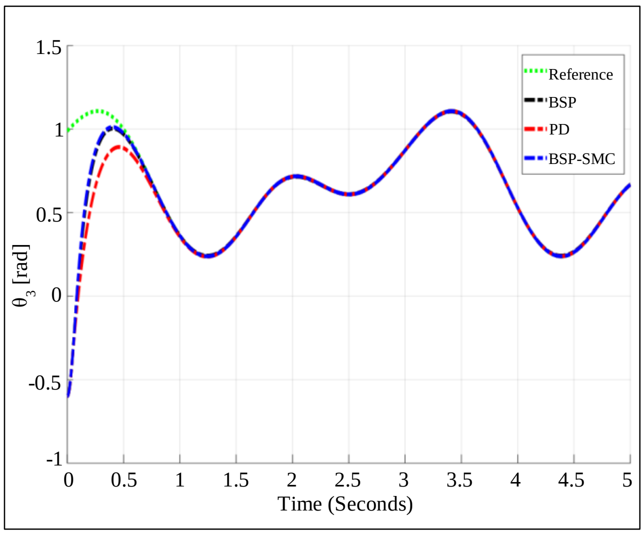

Fig. 7(c) Joint 3 angle trajectory response of the delta robot without disturbances

4.2. Simulation Results for Scenario 2

The simulation results depicted in Figure 8, Figure 9(a), Figure 9(b), Figure 9(c), and Figure 10(a), Figure 10(b), and Figure 10(c) demonstrate the impressive control quality of the Backstepping Sliding Mode Control Algorithm (BSP-SMC) when facing unknown and random disturbances as described in Figure 4, affecting the joint torques. In this case, both Backstepping control and PD control struggle to bring the errors of all three joint angles of the Delta robot to zero, making it challenging to track the desired trajectory.

On the contrary, the proposed Backstepping sliding mode control algorithm exhibits good trajectory tracking capability, with short convergence time maintained at around 0.6 seconds for joints 2 and 3 and approximately 0.1 seconds for joint 1, similar to the simulation results in Scenario 1.

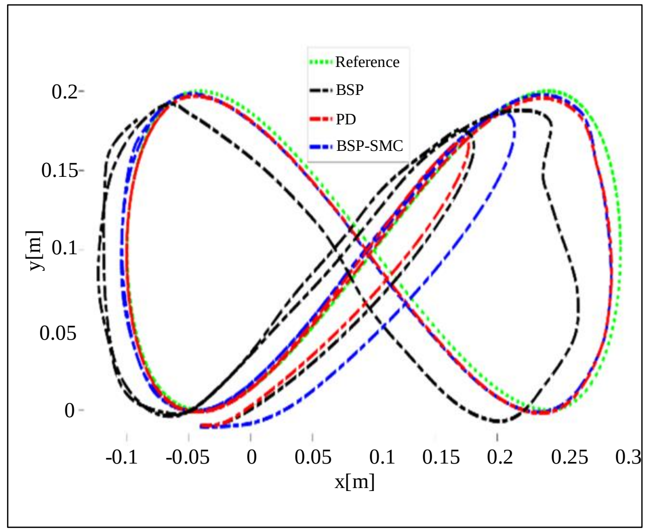

Fig. 8 Trajectory response of the delta robot with unknown external disturbances

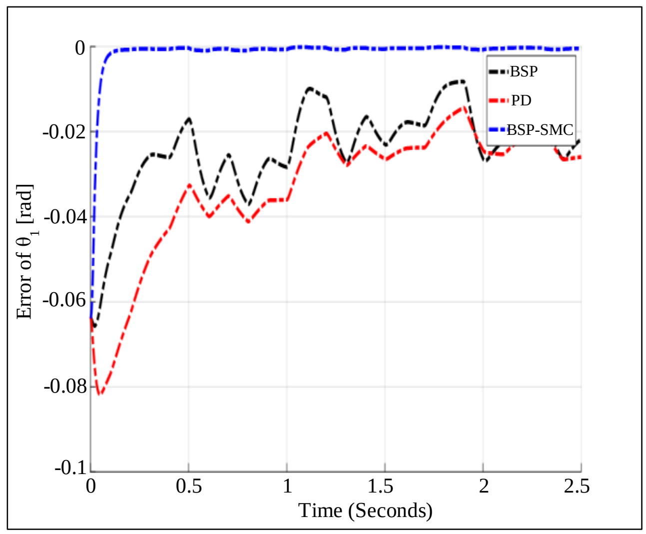

Fig. 9(a) Joint 1 angle error of the delta robot with unknown external disturbances

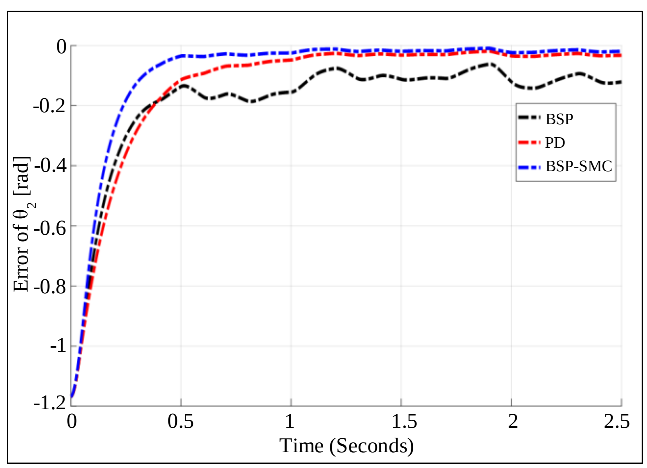

Fig. 9(b) Joint 2 angle error of the delta robot with unknown external disturbances

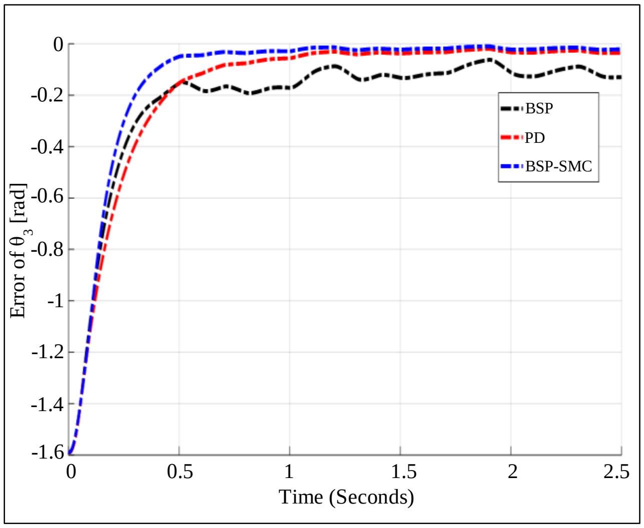

Fig. 9(c) Joint 3 angle error of the delta robot with unknown external disturbances

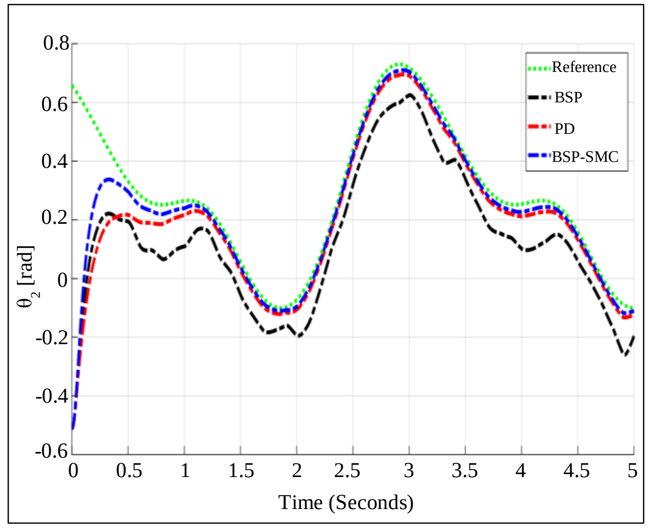

Fig. 10(b) Joint 2 angle trajectory response of the delta robot with unknown external disturbances

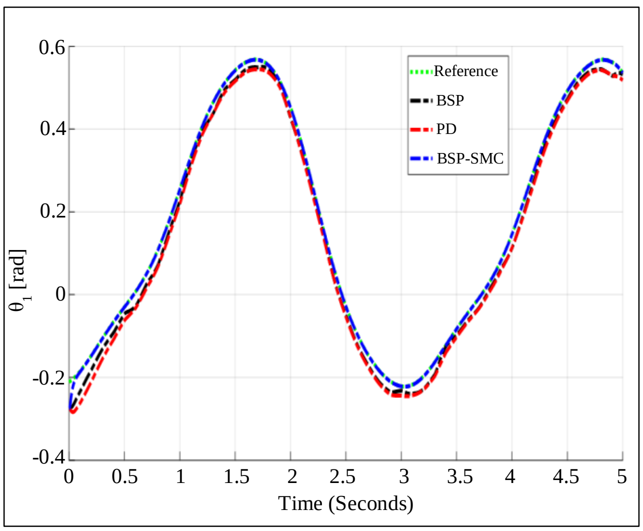

Fig. 10(a) Joint 1 angle trajectory response of the delta robot with unknown external disturbances

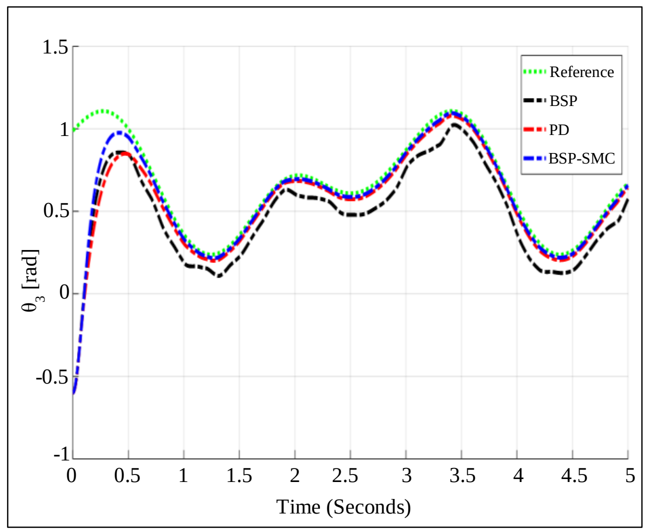

Fig. 10(c) Joint 3 angle trajectory response of the delta robot with unknown external disturbances

However, the Backstepping sliding mode control algorithm still exhibits small oscillations during operation due to the continuous impact of uncertain disturbances on the joint torques of the system.

5. Conclusion

The study proposes the application of the Backstepping sliding mode control method for the Delta robot, enabling the robot to achieve the desired motion trajectory in a short time while maintaining system stability, even in the presence of unknown external disturbances impacting the joint torques. The results demonstrate excellent control performance, as the control algorithm can effectively track the desired motion of the Delta robot and maintain stability with a fast settling time of around 0.6 seconds for joints 2 and 3, approximately 0.1 seconds for joints 1, even with disturbances. The stability and reliability of the Backstepping sliding mode control algorithm provide a solid foundation for further development and real-world applications for Delta robots. In the future, the authors plan to implement the Backstepping sliding mode control algorithm on an experimental model and continue research to improve the algorithm's performance and extend its application to various models of industrial robots.

Acknowledgments

This research work was funded by the Hanoi University of Industry under Project Granted number 32-2022-RD/HĐ-ĐHCN.

References

[1] Luis Angel Castañeda, Alberto Luviano-Juárez, and Isaac Chairez, "Robust Trajectory Tracking of a Delta Robot Through Adaptive Active Disturbance Rejection Control," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1387-1398, 2015. [CrossRef] [Google Scholar] [Publisher link]

[2] Weirong Yang, Weidong Liu, and Wen Tang, "Impedance and Backstepping Controller of Delta Robot Trajectory Tracking," *2022 41st Chinese Control Conference (CCC)*, Hefei, China, pp. 750-755, 2022. [CrossRef] [Google Scholar] [Publisher link]

[3] Tingting Su et al., "Robust Trajectory Tracking of Delta Parallel Robot Using Sliding Mode Control," *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, Xiamen, China, pp. 508-512, 2019. [CrossRef] [Google Scholar] [Publisher link]

[4] Le Minh Thanh et al., "Delta Robot Control Using Single Neuron PID Algorithms Based on Recurrent Fuzzy Neural Network Identifiers," *International Journal of Mechanical Engineering and Robotics Research*, vol. 9, no. 10, pp. 1411-1418, 2020. [CrossRef] [Google Scholar] [Publisher link]

[5] Le Minh Thanh et al., "Evaluating the Quality of Intelligent Controllers for 3-DOF Delta Robot Control," *International Journal of Mechanical Engineering and Robotics Research*, vol. 10, no. 10, pp. 542-552, 2021. [CrossRef] [Google Scholar] [Publisher link]

[6] Nguyen Van Khan et al., "A Comparison Study of Some Control Methods for Delta Spatial Parallel Robot," *Journal of Computer Science and Cybernetics*, vol. 31, no. 1, pp. 71-81, 2015. [CrossRef] [Google Scholar] [Publisher link]

[7] O. Aguilar-Mejia et al., "Adaptive Control of 3-DOF Delta Parallel Robot," *2019 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, Ixtapa, Mexico, pp. 1-6, 2019. [CrossRef] [Google Scholar] [Publisher link]

[8] Thanh-Lam Bui, Thanh-Hung Nguyen, and Xuan-Thuan Nguyen, "A Controller for Delta Parallel Robot Based on Hedge Algebras Method," *Journal of Robotics*, vol. 2023, pp. 1-20, 2023. [CrossRef] [Google Scholar] [Publisher link]

[9] Chandan Choubey, and Jyoti Ohri, "GWO-Based Tuning of LQR-PID Controller for a 3-DOF Parallel Manipulator," *IETE Journal of Research*, vol. 69, no. 7, pp. 4378-4393, 2021. [CrossRef] [Google Scholar] [Publisher link]

[10] João Cavalcanti Santos, Marc Gouttefarde, and Ahmed Chemori, "A Nonlinear Model Predictive Control for the Position Tracking of Cable-Driven Parallel Robots," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2597-2616, 2022. [CrossRef] [Google Scholar] [Publisher link]

[11] Tobias Gold, Andreas Völz, and Knut Graichen, "Model Predictive Interaction Control for Industrial Robots," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 9891-9898, 2020. [CrossRef] [Google Scholar] [Publisher link]

[12] Yazhou Hu et al., "Reinforcement Learning Tracking Control for Robotic Manipulator with Kernel-Based Dynamic Model," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 9, pp. 3570-3578, 2020. [CrossRef] [Google Scholar] [Publisher link]

[13] Xi Chen, and Jan Guhl, "Industrial Robot Control with Object Recognition Based on Deep Learning," *Procedia CIRP*, vol. 76, pp. 149-154, 2018. [CrossRef] [Google Scholar] [Publisher link]

[14] Rongrong Liu et al., "Deep Reinforcement Learning for the Control of Robotic Manipulation: A Focussed Mini-Review," *Robotics*, vol. 10, no. 1, pp. 1-13, 2021. [CrossRef] [Google Scholar] [Publisher link]