

THIẾT KẾ VÀ PHÁT TRIỂN ĐIỀU KHIỂN BẮM QUỸ ĐẠO CHO ROBOT DI ĐỘNG DỰA TRÊN MẠNG NƠN CHỊU TÁC ĐỘNG CÁC THÔNG SỐ BẤT ĐỊNH VÀ NHIỄU LOẠN

DESIGN AND DEVELOPMENT OF TRACTION TRACKING CONTROL FOR MOBILE ROBOTS
BASED ON NEURAL NETWORKS SUBJECT TO UNCERTAINTY PARAMETERS AND DISTURBANCES

Nguyễn Thị Hiền¹, Võ Thu Hà^{1,*},
Võ Quang Lạp¹, Bùi Huy Hải²

DOI: <http://doi.org/10.57001/huiv5804.2024.286>

TÓM TẮT

Bài báo trình bày việc thiết kế và phát triển các bộ điều khiển theo dõi quỹ đạo robot di động dựa vào mạng nơon chịu tác động của các tham số bất định và nhiễu loạn. Thuật toán có cấu trúc điều khiển tích hợp bộ điều khiển Backstepping và bộ điều khiển nơon cho robot di động. Hệ thống robot di động đã được mô hình hóa, bao gồm mô hình động học, động lực học và cơ cấu chấp hành của robot. Mô hình động lực học được tính bằng phương pháp Lagrangian cho các hệ thống phi tuyến. Khi các thông số động lực học của robot chưa xác định hoặc chưa xác định chính xác, mạng nơon được chọn là mạng nơon hai lớp ẩn và với giả thiết các hàm kích hoạt tuyến tính cho lớp đầu ra. Đầu ra của mỗi nơon trong lớp ẩn có thể được tính toán dựa trên đầu vào mạng, trọng số của lớp đầu tiên và hàm kích hoạt của lớp ẩn. Tính toán để đảm bảo đưa ra tín hiệu sai số bằng 0, tính ổn định được chứng minh bằng phương pháp Lyapunov. Các bộ điều khiển khác nhau của robot di động đã được mô phỏng bằng Matlab/Simulink. Kết quả là, việc theo dõi quỹ đạo của robot di động của bộ điều khiển sử dụng mạng nơon (NN) cho kết quả bám quỹ đạo tốt hơn so với bộ điều khiển PD-Backstepping khi có các tham số bất định và nhiễu loạn mô hình.

Từ khóa: Wheel Mobile Robot (WMR); bộ điều khiển PD; bộ điều khiển PD kết hợp với bộ điều khiển dựa vào kỹ thuật Backstepping; bộ điều khiển nơon; bộ điều khiển PD kết hợp với bộ điều khiển dựa vào kỹ thuật Backstepping kết hợp bộ điều khiển nơon.

ABSTRACT

The paper presents the design and development of mobile robot trajectory tracking controllers based on neural networks subject to uncertain parameters and disturbances. The algorithm has a control structure that integrates a backstepping controller and a neural controller for mobile robots. The mobile robot system has been modelled, including the robot's kinematic, dynamic, and actuator models. The dynamic model is calculated using the Lagrangian method for nonlinear systems. When the dynamic parameters of the robot are unknown or not precisely determined, the neural network is chosen as a two-hidden layer neural network with the assumption of linear activation functions for the output layer. The output of each neuron in the hidden layer can be calculated based on the network input, the weight of the first layer and the activation function of the hidden layer. The calculation ensures that the error signal is zero and that the Lyapunov method proves the stability. Different mobile robot controllers have been simulated using Matlab/Simulink. As a result, the trajectory tracking of mobile robots by the controller using a neural network (NN) gives better trajectory tracking results than the PD-Backstepping controller in the presence of uncertain parameters and model disturbances.

Keywords: Wheel Mobile Robot (WMR); PD Controller; PD Controller combined with Backstepping based controller; Neural Controller; PD Controller combined with Backstepping based controller combined with Neural Controller.

¹Khoa Điện - Tự động hóa, Trường Đại học Kinh tế - Kỹ thuật công nghiệp

²Khoa Điện tử và Kỹ thuật máy tính, Trường Đại học Kinh tế - Kỹ thuật công nghiệp

*Email: vtha@uneti.edu.vn

Ngày nhận bài: 20/4/2024

Ngày nhận bài sửa sau phản biện: 10/6/2024

Ngày chấp nhận đăng: 27/9/2024

KÝ HIỆU

Ký hiệu	Ý nghĩa
K_P	Hệ số tỉ lệ
K_D	Hệ số vi phân
C	Cos
S	Sin

CHỮ VIẾT TẮT

NN	: Bộ điều khiển nơron
PD	: Bộ điều khiển PD
PD Backstepping	: Bộ điều khiển PD Backstepping
WRM	: Robot di động
PD Backstepping NN	: Bộ điều khiển PD Backstepping nơron

1. GIỚI THIỆU

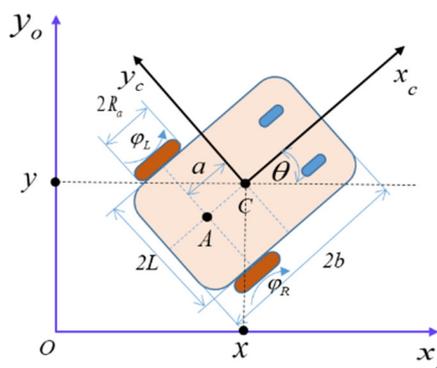
Trong những năm gần đây, bài toán điều khiển chuyển động cho robot di động đã thu hút sự chú ý của các nhà khoa học trên khắp thế giới. Robot di động là một trong số các hệ thống chịu ràng buộc nonholonomic [1]. Và nó cũng là một hệ thống phi tuyến nhiều vào - nhiều ra [2]. Đa số các công bố trước là sử dụng cấu trúc hai mạch vòng điều khiển: mạch vòng động học bên ngoài sử dụng hàm Lyapunov tổng hợp bộ điều khiển bám vị trí, mạch vòng động lực học bên trong điều khiển bám tốc độ. Nhờ có sự tiến bộ của lý thuyết cũng như kỹ thuật điều khiển, đã có rất nhiều phương pháp điều khiển khác nhau được áp dụng để thiết kế các luật điều khiển cho rô bốt di động như: điều khiển trượt [3, 4], điều khiển bền vững [5], điều khiển thích nghi [6-8], điều khiển backstepping [9, 10], tuyến tính hóa phản hồi đầu ra [11, 12].... Các luật điều khiển này đã được thiết kế với giả thiết "bánh xe chỉ lăn mà không trượt". Tuy nhiên, trong thực tiễn ứng dụng, điều kiện các bánh xe chỉ lăn mà không trượt lại có thể thường xuyên không được đảm bảo nên đã có hiện tượng trượt bánh xe [13, 14]. Có rất nhiều nguyên nhân gây ra hiện tượng này như robot di chuyển trên mặt sàn có lực ma sát yếu, lực ly tâm khi robot chuyển động theo đường vòng cung,... Trong các tình huống như vậy, nếu muốn cải thiện chất lượng điều khiển, thì cần phải thiết kế một bộ điều khiển có khả năng bù trượt bánh xe. Phương pháp điều khiển thích nghi nơron, trong tài liệu [15] tác giả S.J.Yoo đã sử dụng mạng nơron để xấp xỉ các thành phần bất định của mô hình và giá trị chặn của thành phần tốc độ trượt bánh xe. Trong [16] đề xuất thuật toán điều khiển thích nghi nơron dựa

trên học tăng cường (Reinforcement Learning), sử dụng mạng nơron để xấp xỉ các thành phần ma sát trượt, thành phần bất định mô hình.

Trong bài báo này, nhóm nghiên cứu đã xây dựng một cách có hệ thống phương pháp mô hình hóa hệ động học và động lực học của robot 3 các tham số hằng bất định của mô hình, chịu tác động nhiễu ngoài, trượt ngang bánh xe. Các mô hình này sẽ được sử dụng để thiết kế các luật điều khiển bù trượt bánh xe, bất định mô hình, và nhiễu ngoài. Đó là các thuật toán điều khiển được phát triển trên nền tảng kỹ thuật Backstepping. Vậy nội dung chính của bài báo là đề xuất bộ điều khiển phi tuyến động học theo kỹ thuật Backstepping kết hợp mạng nơron. Bài báo gồm 4 phần, phần 1 giới thiệu, phần 2 thiết kế bộ điều khiển phi tuyến động học dựa vào kỹ thuật Backstepping kết hợp mạng nơron, phần 3 mô phỏng kiểm chứng thuật toán và phần 4 đưa ra kết luận và kiến nghị.

2. THIẾT KẾ BỘ ĐIỀU KHIỂN ĐỘNG HỌC PHI TUYẾN DỰA VÀO KỸ THUẬT BACKSTEPPING KẾT HỢP MẠNG NƠN

Xét một Whele Mobile Robot (WMR) trong trường hợp có hiện tượng trượt ngang của robot tự hành tự trị kiểu vi sai biểu diễn như hình 1.



Hình 1. Mối quan hệ động học của robot tự hành tự trị kiểu vi sai

Trong đó: A là trọng tâm trục bánh xe chủ động; C là trọng tâm vủa robot WRM; a là khoảng cách giữa tọa độ trọng tâm và trục bánh xe; R_a là bán kính bánh xe chủ động; 2L là khoảng cách giữa hai bánh xe chủ động. x_o, y_o là hệ trục tọa độ tham chiếu, x_c, y_c là hệ trục tọa độ gắn trên robot. θ là góc điều hướng; ϕ_L, ϕ_R là tốc độ bánh xe bên trái, bên phải.

Vị trí của WMR được xác định bởi vector $Q = [X, Y, \theta]^T$ (X, Y, θ) là tọa độ xe trong hệ trục tọa độ tham chiếu.

Theo như tài liệu tham khảo [1] ta có phương trình động học thuận của robot sẽ được viết như phương trình (1) sau:

$$\dot{\mathbf{Q}} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} C(\theta) & -aS(\theta) \\ S(\theta) & aC(\theta) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (1)$$

Trong đó, v là vận tốc dài; ω là vận tốc góc của robot di động.

Theo tài liệu [1, 2], phương trình động lực học của WMR có thể được mô tả là:

$$M(Q)\ddot{Q} + V(Q, \dot{Q}) + F(\dot{Q}) + G(Q) + \tau_d = B(Q)\tau - A^T(Q)\lambda \quad (2)$$

Trong đó: $M(Q)$ là ma trận quán tính xác định dương; $V(Q, \dot{Q})$ là ma trận hướng tâm và ma trận Coriolis; τ_d là thành phần nhiễu không xác định của hệ thống; $B(Q)$ là ma trận đầu vào; λ là nhân tử Lagrange; τ là mô men điều khiển chuyển động cho robot; $A^T(Q)$ là ma trận ràng buộc; \dot{Q} và \ddot{Q} lần lượt biểu thị các vectơ vận tốc và gia tốc tổng quát.

Với: $M(Q) = \begin{bmatrix} m & 0 & maS(\theta) \\ 0 & m & -maC(\theta) \\ maS(\theta) & -maC(\theta) & I_c + 2ma^2 \end{bmatrix}$; $\tau = \begin{bmatrix} \tau_R \\ \tau_L \end{bmatrix}$;

$$V(Q, \dot{Q}) = \begin{bmatrix} ma\dot{\theta}C(\theta) \\ ma\dot{\theta}S(\theta) \\ 0 \end{bmatrix}; B(Q) = \frac{1}{R_a} \begin{bmatrix} \cos(\theta) & C(\theta) \\ S(\theta) & S(\theta) \\ L & -L \end{bmatrix};$$

$$A(Q) = \begin{bmatrix} -S(\theta) \\ C(\theta) \\ -a \end{bmatrix}; \lambda = -m(\dot{x}C(\theta) + \dot{y}S(\theta))\dot{\theta}.$$

Trong đó: m là khối lượng của WMR; I_c là quán tính theo trọng tâm C của robot; τ_L và τ_R lần lượt là mô men bánh trái và bánh phải.

Sau khi loại bỏ ràng buộc $A^T(Q)\lambda$ ta có phương trình động lực học mới như sau:

$$\bar{M}(q)\dot{V}_a(t) + \bar{V}_m(Q, \dot{Q})V_a(t) + \bar{F}(\dot{Q}) + \bar{G}(Q) + \bar{\tau}_d = \bar{B}(Q)\tau \quad (3)$$

Trong đó: $V_a = \begin{bmatrix} v_a \\ \omega_a \end{bmatrix}$ là vận tốc thực của robot;

$$\bar{\tau}_d = S^T(Q)\tau_d; \bar{F}(\dot{Q}) = S^T(Q)F(\dot{Q}); \bar{M}(Q) = S^T(Q)M(Q)S(Q);$$

$$\bar{V}_m(Q, \dot{Q}) = S^T(Q)[C(Q, \dot{Q})S(Q) + S(Q)]; \bar{B}(Q) = S^T(Q)B(Q);$$

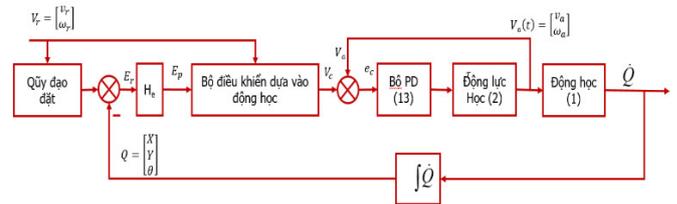
$$\bar{G}(Q) = S^T(Q)G(Q).$$

Nhận xét: Từ phương trình (3) các thành phần bất định và nhiễu trong hệ bao gồm: Khối lượng xe, momen quán tính là bất định, do đó ma trận quán tính được xem là bất định. Khi xe di chuyển trên mặt sàn khác nhau, đặc biệt là mặt sàn trơn và ẩm ướt dễ xảy ra hiện tượng trượt bánh xe

ảnh hưởng đến quỹ đạo đường đi, hoặc khi xe di chuyển tốc độ nhanh vào các vòng cua thì ma sát giữa bánh xe và mặt sàn sẽ thay đổi, là nhiễu làm ảnh hưởng nhiều đến vị trí và góc hướng của xe. Trong nghiên cứu này, chúng tôi sử dụng mô hình động học thuận (1), động lực học (3) của robot di động làm đối tượng điều khiển sao cho robot di động này bám theo một quỹ đạo cho trước.

2.1. Nội dung thuật toán

Bộ điều khiển dựa trên động học hay còn gọi là bộ điều khiển bước lùi được đề xuất bởi Kanayama năm 1992 là một quy tắc điều khiển theo dõi ổn định cho một robot di động và có cấu trúc điều khiển như hình 2.



Hình 2. Sơ đồ cấu trúc bộ điều khiển PD-Backstepping

Trong đó: $E_p = T_e(Q_r - Q)$ là sai lệch quỹ đạo đặt và quỹ đạo thực theo [18-20].

+ H_e được xác định từ kết quả phương trình động học [17]:

$$H_e = \begin{bmatrix} C(\theta) & S(\theta) & 0 \\ -S(\theta) & C(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

+) Quỹ đạo đặt là $Q_r = [X_r, Y_r, \theta_r]^T$ nên có thể viết:

$$\dot{Q}_r = [\dot{X}_r, \dot{Y}_r, \dot{\theta}_r]^T = S(Q)V_r(t) \quad (5)$$

+) Quỹ đạo thực $Q = [X, Y, \theta]$ nên có thể viết:

$$\dot{Q} = [\dot{X}, \dot{Y}, \dot{\theta}]^T = S(Q)V_a(t) \quad (6)$$

Nên sai lệch vị trí có thể biểu diễn như sau:

$$E_p = \begin{bmatrix} E_Y \\ E_Y \\ E_\theta \end{bmatrix} = H_e(Q_r - Q) = \begin{bmatrix} C(\theta) & S(\theta) & 0 \\ -S(\theta) & C(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_r - X \\ Y_r - Y \\ \theta_r - \theta \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} \dot{E}_Y \\ \dot{E}_Y \\ \dot{E}_\theta \end{bmatrix} = \dot{E}_p = f(t, E_p) = \begin{bmatrix} \omega_a \cdot (E_Y) + v_r \cdot C(E_\theta) - v_a \\ -\omega_a \cdot E_X + v_r \cdot S(E_\theta) \\ \dot{\theta}_r - \dot{\theta} \end{bmatrix} \quad (7)$$

+) Bài toán điều khiển trong trường hợp này sẽ là tính toán một quy tắc điều khiển cho xe, trong đó mục tiêu tính toán vận tốc $V_c = f(E_p, V_r, k)$ làm cho hệ thống ổn

$$+) \frac{\partial \mathbf{Q}_c}{\partial \mathbf{V}_c} = J_{cv} = \begin{bmatrix} \frac{\partial X}{\partial V_c} & \frac{\partial X}{\partial \omega_c} \\ \frac{\partial Y}{\partial V_c} & \frac{\partial Y}{\partial \omega_c} \\ \frac{\partial \theta}{\partial V_c} & \frac{\partial \theta}{\partial \omega_c} \end{bmatrix} = \begin{bmatrix} J_{cv11} & J_{cv12} \\ J_{cv21} & J_{cv22} \\ J_{cv31} & J_{cv32} \end{bmatrix} \quad (17)$$

$$+) \frac{\partial \mathbf{V}_c}{\partial \mathbf{a}} = \begin{bmatrix} \frac{\partial V_c}{\partial k_x} & \frac{\partial V_c}{\partial k_y} & \frac{\partial V_c}{\partial k_\theta} \\ \frac{\partial \omega_c}{\partial k_x} & \frac{\partial \omega_c}{\partial k_y} & \frac{\partial \omega_c}{\partial k_\theta} \end{bmatrix} = \begin{bmatrix} E_x & 0 & 0 \\ 0 & v_r E_y & v_r S(E_\theta) \end{bmatrix} \quad (18)$$

Thay (17), (18) vào (16):

$$\frac{\partial \mathbf{Q}_c}{\partial \mathbf{a}} = \frac{\partial \mathbf{Q}_c}{\partial \mathbf{V}_c} \cdot \frac{\partial \mathbf{V}_c}{\partial \mathbf{a}} = J_{cv} \begin{bmatrix} E_x & 0 & 0 \\ 0 & v_r E_y & v_r S(E_\theta) \end{bmatrix} \quad (19)$$

Thay (19) vào (15):

$$\frac{\partial \mathbf{J}_m}{\partial \mathbf{a}} = \begin{bmatrix} \frac{\partial J_m}{\partial k_x} & \frac{\partial J_m}{\partial k_y} & \frac{\partial J_m}{\partial k_\theta} \end{bmatrix} = \gamma E_p^T H_e J_{cv} \begin{bmatrix} E_x & 0 & 0 \\ 0 & v_r E_y & v_r S(E_\theta) \end{bmatrix} \quad (20)$$

Trong đó:

$$\gamma = \begin{bmatrix} \gamma_x & 0 & 0 \\ 0 & \gamma_y & 0 \\ 0 & 0 & \gamma_\theta \end{bmatrix}; E_p^T = [E_x \ E_y \ E_\theta];$$

$$J_{cv} = \begin{bmatrix} \frac{\partial X}{\partial V_c} & \frac{\partial X}{\partial \omega_c} \\ \frac{\partial Y}{\partial V_c} & \frac{\partial Y}{\partial \omega_c} \\ \frac{\partial \theta}{\partial V_c} & \frac{\partial \theta}{\partial \omega_c} \end{bmatrix} = \begin{bmatrix} J_{cv11} & J_{cv12} \\ J_{cv21} & J_{cv22} \\ J_{cv31} & J_{cv32} \end{bmatrix}$$

Do đó, độ lợi của bộ điều khiển động học sẽ thay đổi và thích ứng để làm cho hàm chi phí bằng 0 theo độ dốc giảm dần sau:

$$K_x = k_x + \Delta K_x; K_y = k_y + \Delta K_y; K_\theta = k_\theta + \Delta K_\theta \quad (21)$$

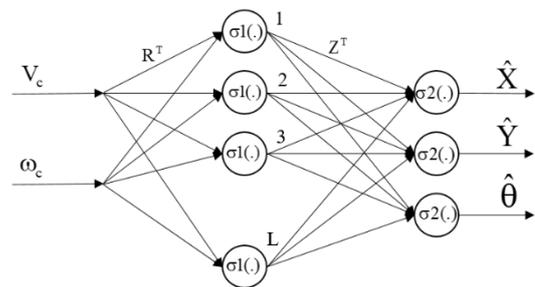
Trong đó, sự biến thiên của các thông số $\Delta K_x, \Delta K_y, \Delta K_\theta$ sẽ được tính theo các phương trình:

$$\Delta K_x = -\eta_{k_x} \frac{\partial J_{cv}}{\partial K_x}; \Delta K_y = -\eta_{k_y} \frac{\partial J_{cv}}{\partial K_y}; \Delta K_\theta = -\eta_{k_\theta} \frac{\partial J_{cv}}{\partial K_\theta} \quad (22)$$

Với các hệ số $\eta_{k_x}, \eta_{k_y}, \eta_{k_\theta}$ là tốc độ học của thuật toán cập nhật.

Phần quan trọng của việc tính đạo hàm của phương trình (20) là cách tính ma trận Jacobian J_{cv} của hệ. Ma trận J_{cv} có thể được tính toán bằng cách sử dụng các phương trình chính xác của hệ thống hoặc có thể được cung cấp bởi mô hình trực tiếp mạng nơron. Phương pháp tính ma trận J_{cv} bởi mô hình trực tiếp mạng nơron để cập nhật trực tuyến giá trị cho ma trận J_{cv} . Mạng nơron mô hình trực tiếp có cấu trúc như hình 4. Khi đó việc tính ma trận Jacobian sẽ được cập nhật trực tiếp bằng cách luôn cập nhật giá trị $\hat{\mathbf{Q}} = [\hat{X} \ \hat{Y} \ \hat{\theta}]^T$ cho biểu thức tính ma trận

Jacobian $J_{cv} = \frac{\partial \hat{\mathbf{Q}}}{\partial \mathbf{V}_c}$. Khi đó đầu vào của mạng nơron.



Hình 4. Mạng nơron 2 lớp ẩn

Ma trận Jacobian có thể được tìm thấy từ mạng nơron ở trên là:

$$\frac{\partial \hat{\mathbf{Q}}}{\partial \mathbf{V}_c} = J_{cv} = \begin{bmatrix} \frac{\partial \hat{X}}{\partial V_c} & \frac{\partial \hat{X}}{\partial \omega_c} \\ \frac{\partial \hat{Y}}{\partial V_c} & \frac{\partial \hat{Y}}{\partial \omega_c} \\ \frac{\partial \hat{\theta}}{\partial V_c} & \frac{\partial \hat{\theta}}{\partial \omega_c} \end{bmatrix} = \begin{bmatrix} J_{cv11} & J_{cv12} \\ J_{cv21} & J_{cv22} \\ J_{cv31} & J_{cv32} \end{bmatrix} \quad (23)$$

Với đầu ra của mạng nơron sẽ được tính:

$$y_i = \sigma \left(\sum_{j=1}^m Z_{ij} \sigma \left(\sum_{j=1}^n R_{ij} x_j + R_{i0} \right) + Z_{i0} \right), \quad (24)$$

$i = 1, 2, 3, \dots, m; j = 1, 2, 3, \dots, n$

Trong đó: R_{ij} là ma trận trọng số lớp thứ nhất; Z_{ij} là ma trận trọng số cho lớp thứ hai.

Đầu vào của lớp thứ nhất là $V_c = \begin{bmatrix} v_c \\ \omega_c \end{bmatrix} = x_j$ và đầu ra của lớp này sẽ là đầu vào của lớp thứ hai.

$$T_i = \sigma \left(\sum_{j=1}^n R_{ij} x_j + R_{i0} \right) \quad (25)$$

Các giá trị ban đầu $x_0 = 1, T_0 = 1$, khi đó ta có thể tính:

$$y_i = \sigma\left(\sum_{l=1}^L Z_{il} T_l\right) = \sigma u_i^2 \quad (26)$$

$$T_l = \sigma\left(\sum_{j=1}^n R_{lj} x_j\right) = \sigma u_l^1 \quad (27)$$

Khi đó: $\frac{\partial Q_c}{\partial V_c} = \frac{\partial y_i}{\partial x_j} = \frac{\partial y_i}{\partial T_l} \frac{\partial T_l}{\partial x_j}$ (28)

Theo phương trình (26) và (27) ta có:

$$\frac{\partial y_i}{\partial T_l} = W_{il} \sigma' \left(\sum_{l=1}^L Z_{il} T_l\right) = Z_{il} \sigma'(u_i^2) \quad (29)$$

$$\frac{\partial T_l}{\partial x_j} = R_{lj} \sigma' \left(\sum_{j=1}^n R_{lj} x_j\right) = R_{lj} \sigma'(u_l^1) \quad (30)$$

Thay (29), (30) vào (28) tính được ma trận Jacobian theo mạng nơron sẽ theo công thức:

$$\frac{\partial Q_c}{\partial V_c} = \frac{\partial y_i}{\partial x_j} = \frac{\partial y_i}{\partial T_l} \cdot \frac{\partial T_l}{\partial x_j} = Z_{il} \cdot \sigma' \left(\sum_{l=1}^L Z_{il} T_l\right) \cdot R_{lj} \cdot \sigma' \left(\sum_{j=1}^n R_{lj} x_j\right) \quad (31)$$

Mặt khác từ phương trình (26) và (27) có thể tính:

$$\frac{\partial y_i}{\partial Z_{il}} = \sigma' \left(\sum_{l=1}^L Z_{il} T_l\right) T_l = \sigma'(u_i^2) T_l \quad (32)$$

$$\frac{\partial T_l}{\partial R_{lj}} = \sigma' \left(\sum_{j=1}^n R_{lj} x_j\right) x_j = \sigma'(u_l^1) x_j \quad (33)$$

Trong đó: $\sigma'()$ là đạo hàm của hàm kích hoạt.

Thuật toán lan truyền ngược là thuật toán điều chỉnh trọng số dựa trên độ dốc giảm dần, do đó trọng số sẽ được điều chỉnh cập nhật như sau:

$$Z_{il}(k+1) = Z_{il}(k) - \eta \frac{\partial E(k)}{\partial Z_{il}} \quad (34)$$

$$R_{lj}(k+1) = R_{lj}(k) - \eta \frac{\partial E(k)}{\partial R_{lj}} \quad (35)$$

Với $E(k)$ là tổng bình phương sai lệch giữa giá trị đầu vào quỹ đạo mong muốn và quỹ đạo đầu ra của mạng nơron.

$$E(k) = \frac{1}{2} e^T(k) e(k) = \frac{1}{2} \sum_{i=1}^m e_i(k)^2 \quad (36)$$

$$e_i(k) = Y_i(k) - y_i(k) \quad (37)$$

Trong đó: $y_i(k)$ được tính bằng cách sử dụng phương trình lan truyền thuận cho mạng nơron với các thành phần của mẫu đầu vào mạng nơron $x_i(k)$.

Độ dốc cần thiết của hàm chi phí $E(k)$ đối với các trọng số có thể dễ dàng được xác định bằng cách sử dụng quy tắc dây chuyền. Cụ thể đối với mức tăng của lớp thứ hai, ta có:

$$\frac{\partial E}{\partial Z_{il}} = \frac{\partial E}{\partial u_i^2} \frac{\partial u_i^2}{\partial Z_{il}} = \left(\frac{\partial E}{\partial e_i} \frac{\partial e_i}{\partial y_i} \frac{\partial y_i}{\partial Z_{il}} \right) \frac{\partial u_i^2}{\partial Z_{il}} \quad (38)$$

Từ công thức (29), (30), (31), (32) và (33) ta có:

$$\frac{\partial E}{\partial u_i^2} = \frac{\partial E}{\partial e_i} \frac{\partial e_i}{\partial y_i} \frac{\partial y_i}{\partial Z_{il}} = -1 \sigma'(u_i^2) e_i = -\sigma'(u_i^2) e_i \quad (39)$$

$$\frac{\partial u_i^2}{\partial Z_{il}} = e_i (-1) \sigma'(u_i^2) T_l = -T_l \sigma'(u_i^2) e_i \quad (40)$$

Tương tự với trọng số lớp đầu tiên:

$$\frac{\partial E}{\partial R_{lj}} = \frac{\partial E}{\partial u_l^1} \frac{\partial u_l^1}{\partial R_{lj}} = \left(\sum_{i=1}^m \frac{\partial E}{\partial u_i^2} \frac{\partial u_i^2}{\partial T_l} \frac{\partial T_l}{\partial u_l^1} \right) \frac{\partial u_l^1}{\partial R_{lj}} \quad (41)$$

Thay công thức (38), (39), (40) vào (41) ta tính được:

$$\frac{\partial E}{\partial R_{lj}} = \left(\sum_{i=1}^m -\sigma'(u_i^2) e_i Z_{il} \sigma'(u_l^1) \right) x_j \sigma'(u_l^1) \quad (42)$$

$$\Leftrightarrow \frac{\partial E}{\partial R_{lj}} = -x_j \sigma'(u_l^1) \sum_{i=1}^m (-\sigma'(u_i^2) e_i) Z_{il} \quad (43)$$

Ta có thể viết đơn giản hóa sai lệch lan truyền ngược cho lớp hai và lớp một tương ứng như sau:

$$\delta_i^2 = \frac{\partial E}{\partial u_i^2} = -\sigma'(u_i^2) e_i \quad (44)$$

$$\delta_l^1 = -\frac{\partial E}{\partial R_{lj}} = \sigma'(u_l^1) \sum_{i=1}^m \delta_i^2 Z_{il} \quad (45)$$

Giả sử các hàm kích hoạt sigmoid được sử dụng, các sai lệch lan truyền ngược có thể được tính như sau:

$$\delta_i^2 = y_i(1 - y_i) e_i \quad (46)$$

$$\delta_l^1 = T_l(1 - T_l) \sum_{i=1}^m \delta_i^2 Z_{il} \quad (47)$$

Kết hợp các phương trình trên, có thể viết công thức thuật toán lan truyền ngược cập nhật các trọng số như sau:

$$Z_{il}(k+1) = Z_{il}(k) - \eta T_l \delta_i^2 \quad (48)$$

$$R_{lj}(k+1) = R_{lj}(k) - \eta x_j \delta_l^1 \quad (49)$$

Vậy từ đó ta sẽ tính được ma trận Jacobian theo công thức (17) với các trọng số cập nhật bởi phương trình (48) và (49) bởi mạng nơron.

3. MÔ PHỎNG KIỂM CHỨNG THUẬT TOÁN ĐIỀU KHIỂN PHI TUYẾN ĐỘNG HỌC DỰA VÀO KỸ THUẬT BACKSTEPPING KẾT HỢP MẠNG NƠN

+) Chọn mô hình robot di động với thông số động học: $m = 10\text{kg}$; $J = 0,56\text{kgm}^2$; $a = 0,05\text{m}$; $R_a = 0,075\text{m}$; $L = 0,2\text{m}$.

+) Chọn quỹ đạo là quỹ đạo hình tròn:
 $x_r = \frac{1}{2} \cos(\frac{\pi}{10,2} t)$; $y_r = \frac{1}{2} \sin(\frac{\pi}{10,5} t)$; $\theta_r = \frac{\pi}{10,2} t$.

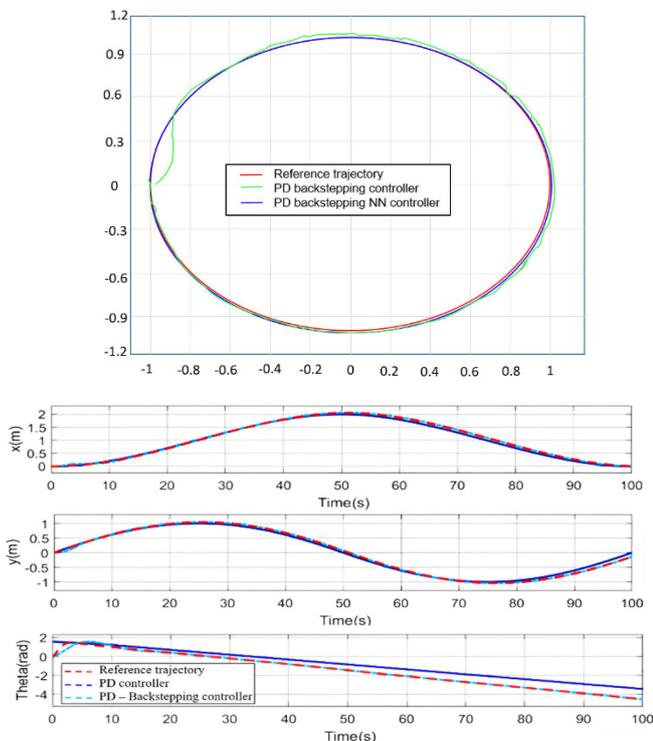
+) Các tham số được sử dụng trong bộ điều khiển:
 $N_h = 20$: Số lượng nơon trong mạng nơon;
 $\eta_{k_x} = \eta_{k_y} = \eta_{k_\theta} = 0,3$: Hệ số tỷ lệ học cho thuật toán lan truyền ngược;
 $\beta_x = \beta_y = \beta_\theta = 0,7$: Hệ số tốc độ động học cho thuật toán lan truyền ngược;
 $N_c = 200$: Số chu kỳ mà bộ dữ liệu đầu vào sẽ lặp lại;
 $F(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$: Hàm sigmoid kích hoạt cho lớp ẩn;
 $f_h(x) = x$: Hàm kích hoạt tuyến tính cho lớp đầu ra.

Thông số của bộ điều khiển được xác định theo bảng 1 Bảng 1. Thông số các bộ điều khiển

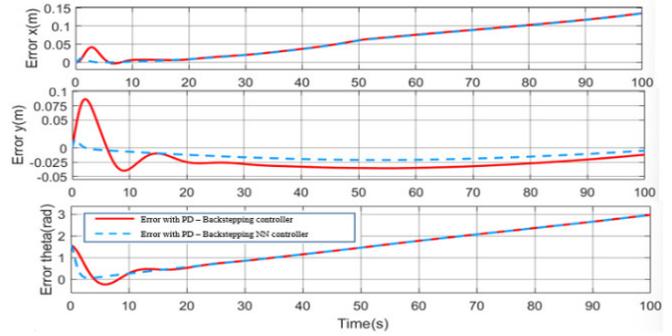
PD Backstepping	PD Backstepping NN
$K_p = \text{diag}[55 \ 55]$;	$K_p = \text{diag}[5]$; $K_i = \text{diag}[11 \ 11]$,
$K_i = \text{diag}[11 \ 11]$,	$K_d = \text{diag}[0,8 \ 0,8]$;
$K_d = \text{diag}[0,8 \ 0,8]$;	$\gamma_x = 1, \gamma_y = 50, \gamma_\theta = 1$
$K_x = 1, K_y = 55, K_\theta = 10$	K_x, K_y, K_θ xác định theo hình 7

3.1. Trường hợp không có nhiễu ngoại tác động

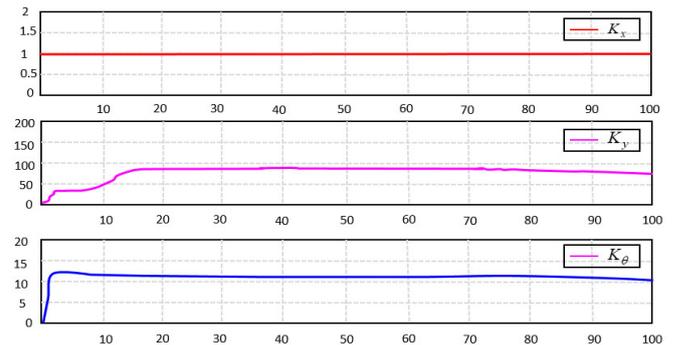
Khi không có nhiễu ngoại tác động kết quả mô phỏng được thể hiện như hình 5, 6 và 7.



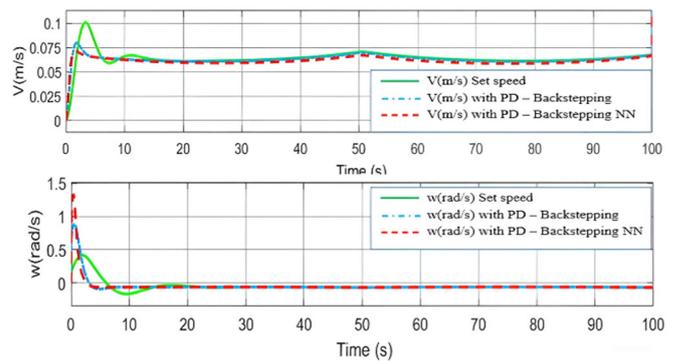
Hình 5. Biểu diễn quỹ đạo bám khi sử dụng PD-Backstepping và PD-Backstepping NN



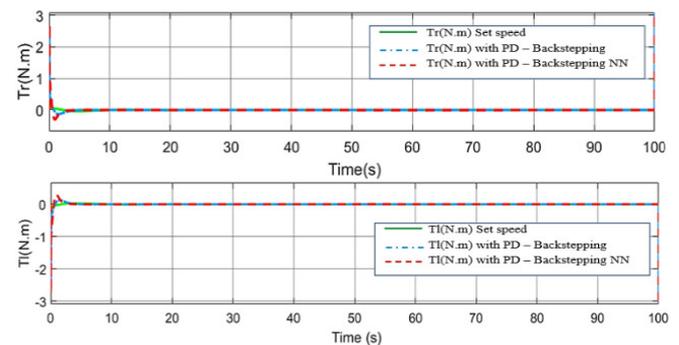
Hình 6. Biểu diễn sai lệch E_x, E_y, E_θ



Hình 7. Biểu diễn các tham số k_x, k_y, k_θ được cập nhật qua mạng NN



Hình 8. Biểu diễn vận tốc V, w



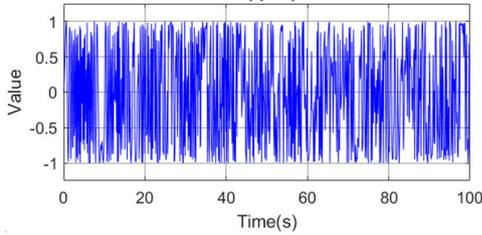
Hình 9. Biểu diễn mô men bánh xe Tr, Tl

Nhận xét: Với bộ điều khiển PD-Backstepping khi không có mạng nơon thì hệ số k_x, k_y, k_θ phải thực hiện qua phương pháp dò nhiều lần để có được các giá trị k_x, k_y, k_θ tối ưu như bảng 1 mới đảm bảo sai lệch bám quỹ đạo E_x, E_y, E_θ tiến tới 0 (hình 6). Trong khi đó bộ điều khiển

PD-Backstepping NN xác định được các tham số k_x, k_y, k_θ được cập nhật như hình 7 và xác định được sai lệch: $E_x = 0,015m; E_y = 0,065m; E_\theta = 0,25rad$.

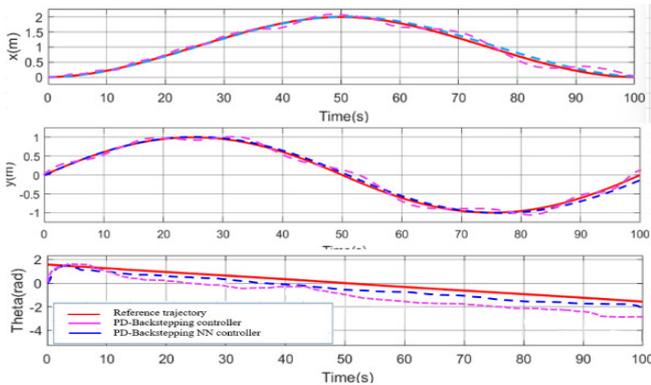
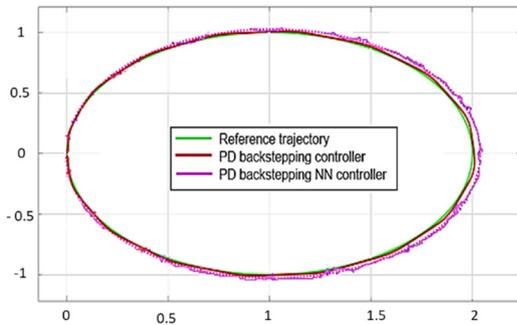
3.2. Trường hợp có nhiễu ngoại tác động

Khi có nhiễu ngoại tác động, đáp ứng nhiễu được thể hiện như hình 10.

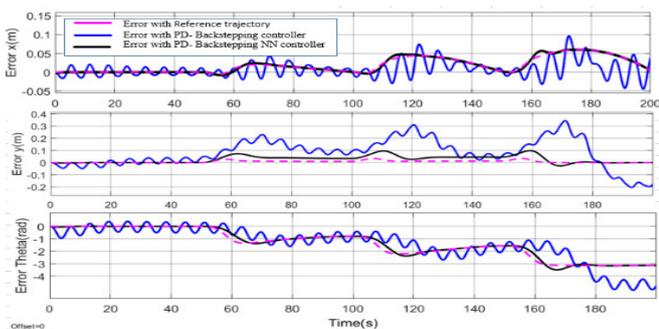


Hình 10. Nhiễu ngoại tác động

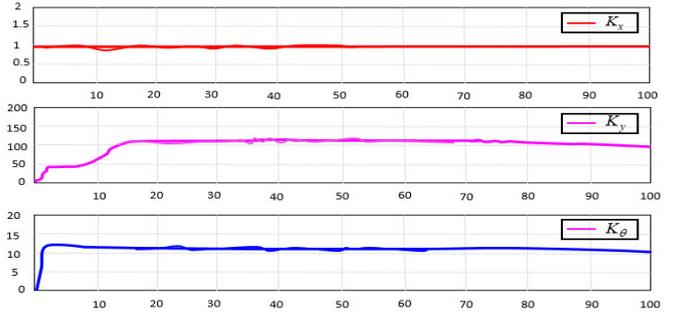
Khi đó có hình 11 theo dõi quỹ đạo khi có nhiễu, hình 12 sai lệch bám quỹ đạo khi có nhiễu.



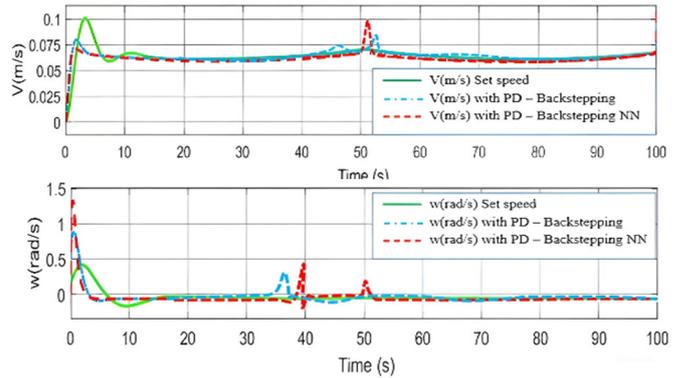
Hình 11. Biểu diễn quỹ đạo bám khi sử dụng PD-Backstepping và PD-Backstepping NN



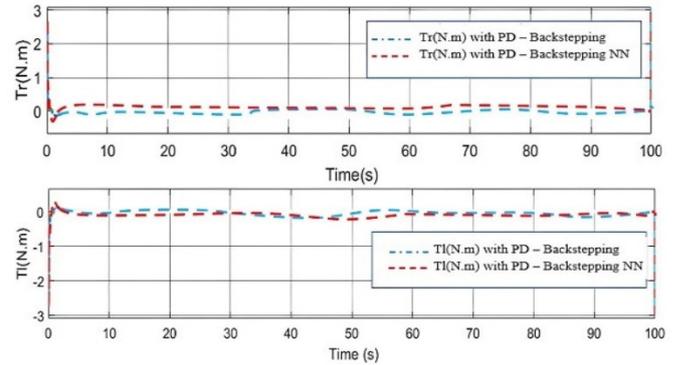
Hình 12. Biểu diễn sai lệch E_x, E_y, E_θ



Hình 13. Biểu diễn các tham số k_x, k_y, k_θ được cập nhật qua mạng NN



Hình 14. Biểu diễn vận tốc V, w khi có nhiễu



Hình 15. Biểu diễn momen bánh xe T_r, T_l khi có nhiễu

Nhận xét: Với bộ điều khiển PD-Backstepping có thành phần nhiễu ngoại tác động thì hệ số k_x, k_y, k_θ cũng phải thực hiện qua phương pháp dò nhiễu lần để có được các tham số k_x, k_y, k_θ nhưng vẫn tồn tại sai lệch lớn so với bộ điều khiển PD-Backstepping NN như hình 12. Bộ điều khiển PD-Backstepping NN xác định được các tham số k_x, k_y, k_θ được cập nhật như hình 13 và xác định được sai lệch: $E_x = 0,001m; E_y = 0,005m; E_\theta = 0,03rad$ khi có nhiễu tác động tốt hơn so với bộ điều khiển PD-Backstepping (hình 12).

4. KẾT LUẬN

Nội dung bài báo đã trình bày được mô hình động học, động lực học của robot WMR khi xét ảnh hưởng của thành phần trượt ngang bánh xe. Chứng minh được sự ổn định của bộ PD-Backstepping NN có xét đến tham số

hằng bất định mô hình toán học, tác động thành phần nhiễu ngoại và trượt ngang bánh xe lên robot chuyển động với quỹ đạo hình tròn. Tính ổn định tiệm cận của hệ thống được phân tích dựa trên lý thuyết ổn định Lyapunov và chất lượng hoạt động của xe được kiểm nghiệm thông qua mô phỏng kỹ thuật số. Tuy nhiên các kết quả của bài báo mới chỉ dừng lại ở phân tích lý thuyết và khảo sát qua mô phỏng số. Tuy đã cố gắng mô phỏng gần chính xác với đối tượng thực, nhưng để có thể áp dụng được cho WMR còn đòi hỏi phải thực thi cho xe trong thực tế và hoàn thiện hơn nữa thuật toán điều khiển với môi trường chuyển động của xe có vật cản động và tĩnh. Đó là hướng tiếp theo của nhóm nghiên cứu trong thời gian tới.

TÀI LIỆU THAM KHẢO

- [1]. Angga Rusdinar, "Automated Ultraviolet C Light Mobile Robot for Room Sterilization and Disinfection," *IJTech*, 12(4), 854-864, 2021.
- [2]. Li Y., Wang Z., Zhu L., "Adaptive Neural Network PID Sliding Mode Dynamic Control of Nonholonomic Mobile Robot," in *Proceedings of the 2010 IEEE International Conference on Information and Automation, Harbin, China*, 753-757, 2010.
- [3]. Chwa D. K., "Sliding-mode tracking control of nonholonomic wheeled mobile robots in polar coordinates," *IEEE Transactions on Control Systems Technology*, 12(4), 637-644, 2004.
- [4]. Park B., Yoo S., Park J., Choi Y., "Adaptive neural sliding mode control of nonholonomic wheeled mobile robots with model uncertainty," *IEEE Transactions on Control Systems Technology*, 17(1), 207-214, 2009.
- [5]. Z. Jiang, "Robust exponential regulation of nonholonomic systems with uncertainties," *Automatic*, 36(2), 189-209, 2000.
- [6]. Rubio F., Valero F., Llopis-Albert C., "A review of mobile robots: Concepts, methods, theoretical framework, and applications," *Sage Journal*, 10, 2019.
- [7]. Wang H., Wang S., "Trajectory Tracking Control for Nonholonomic Wheeled Mobile Robots with External Disturbances and Parameter Uncertainties," *Int. J. Control Autom. Syst.*, 18, 3015-3022, 2020.
- [8]. Tinh N., Kiem N., Tuan Hung D., Pham T., "Neural Network-based Adaptive Sliding Mode Control Method for Tracking of a Nonholonomic Wheeled Mobile Robot with Unknown Wheel Slips, Model Uncertainties, and Unknown Bounded External Disturbances," *Acta Polytech. Hung.*, 15, 103-123, 2018.
- [9]. Liu J., Wang Z., Zhang L., Walker P., "Sideslip angle estimation of ground vehicles: A comparative study," *IET Control Theory Appl.*, 14, 3490-3505, 2021.
- [10]. Wang C., Wang D., Han Y., "Neural Network Based Adaptive Dynamic Surface Control for Omnidirectional Mobile Robots Tracking Control with Full-State Constraints and Input Saturation," *Int. J. Control Autom. Syst.*, 19, s4067-s4077, 2021.
- [11]. Aldo J., Vicente P., Anand S., Juan D., "Adaptive Fuzzy Velocity Field Control for Navigation of Nonholonomic Mobile Robots," *J. Intell. Robot. Syst.*, 101, 1-12, 2021.
- [12]. Muhammad J.R., Memon A. Y., "Trajectory Tracking and Stabilization of Nonholonomic Wheeled Mobile Robot Using Recursive Integral Backstepping Control," *Electronics*, 10, 1-22, 2021.
- [13]. Liu J., Wang Z., Zhang L., Walker P., "Sideslip angle estimation of ground vehicles: A comparative study," *IET Control Theory Appl.*, 14, 3490-3505, 2021.
- [14]. Zhang J.J., Fang Z.L., Zhang Z.Q., Gao R.Z., Zhang S.B., "Trajectory Tracking Control of Nonholonomic Wheeled Mobile Robots Using Model Predictive Control Subjected to Lyapunov-Based Input Constraints," *Int. J. Control Autom. Syst.*, 20, 1640-1651, 2022.
- [15]. Y. Z. Yue, "Adaptive fuzzy logic-based sliding mode control for a nonholonomic mobile robot in the presence of dynamic uncertainties," in *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 229, 11, 1979-1988, 2015.
- [16]. S. Peng, W. Shi, "Adaptive fuzzy output feedback control of a nonholonomic wheeled mobile robot," *IEEE Access*, 6, 43414-43424, 2018.
- [17]. Ha V.T., Thuong T.T.; Thanh N.T., Vinh V.Q., "Research on some control algorithms to compensate for the negative effects of model uncertainty parameters external interference, and wheeled slip for mobile robot," *Actuators*, 13, 31, 2024.
- [18]. R. W. Brockett, "Asymptotic stability and feedback stabilization," in *Differential Geometric Control Theory*, R. W. Brockett, R. S. Millman, and H. J. Sussmann, Eds. Boston, MA: Birkhauser, 1983
- [19]. Y. Kanayama, et al., "A stable tracking control method for an autonomous mobile robot," in *Proc. IEEE Conf. Robot. Autom.*, 384-389, 1990.
- [20]. A. Bloch, S. Drakunov, "Stabilization of a nonholonomic system via sliding modes," in *Proc. of IEEE international conference on Decision Control*, Madrid, Spain, 2961-2963, 1994.

AUTHORS INFORMATION

Nguyen Thi Hien¹, Vo Thu Ha¹, Vo Quang Lap¹, Bui Huy Hai²

¹Faculty of Electrical and Automation Engineering, University of Economics - Technology for Industries, Vietnam

²Faculty of Electronics and Computer Engineering, University of Economics - Technology for Industries, Vietnam