

ĐIỀU KHIỂN ROBOT ĐA HƯỚNG BA BÁNH BẰNG PHƯƠNG PHÁP DỰ BÁO MÔ HÌNH

CONTROLLING THREE-WHEELED OMNIDIRECTIONAL ROBOTS USING MODEL PREDICTIVE CONTROL

Nguyễn Hoàng Long, Huỳnh Hồ Trung Tịnh, Nguyễn Duy Ánh*

Khoa Cơ khí, Trường Đại học Bách Khoa, Đại học Quốc gia Thành phố Hồ Chí Minh

*Email: ndanhcdt@hcmut.edu.vn

TÓM TẮT

Bài báo này trình bày một phương pháp điều khiển dự báo mô hình (MPC) cho robot đa hướng ba bánh. Phương pháp đề xuất sử dụng chiến lược tìm kiếm tối ưu đơn giản để giảm thiểu độ phức tạp tính toán, đồng thời duy trì khả năng theo dõi quỹ đạo chính xác. Các thí nghiệm mô phỏng trên các quỹ đạo khác nhau cho thấy phương pháp MPC đạt được sai số trung bình thấp và xử lý tốt các yếu tố phi tuyến như trượt bánh xe và nhiễu cảm biến IMU. Kết quả nghiên cứu cung cấp cơ sở khoa học cho việc áp dụng phương pháp MPC trong các ứng dụng robot đa hướng yêu cầu độ chính xác cao.

Từ khóa: Robot đa hướng; Điều khiển dự báo mô hình; Bánh xe omni; Điều khiển phi tuyến; Theo dõi quỹ đạo.

ABSTRACT

This paper presents a Model Predictive Control (MPC) method for three-wheeled omnidirectional robots. The proposed MPC method employs a simplified optimization search strategy to reduce computational complexity while maintaining accurate trajectory tracking capabilities. Simulation experiments on various trajectories (circle, square, and figure-8) demonstrate that the MPC method achieves low average error and effectively handles nonlinear factors such as wheel slip and IMU sensor noise. The research results provide a scientific basis for applying MPC in omnidirectional robot applications requiring high precision.

Keywords: Omnidirectional robot; Model predictive control; Omni wheel; Nonlinear control; Trajectory tracking.



1. ĐẶT VẤN ĐỀ

Robot đa hướng đang ngày càng được sử dụng rộng rãi trong nhiều lĩnh vực từ sản xuất công nghiệp đến dịch vụ và giải trí, nhờ khả năng di chuyển linh hoạt trong các không gian hạn chế [1]. Khác với các robot di động thông thường, robot đa hướng có thể di chuyển theo bất kỳ hướng nào mà không cần thay đổi hướng của khung xe. Đặc điểm này mang lại lợi thế lớn trong các ứng dụng yêu cầu độ chính xác và linh hoạt cao [2].

Trong số các cấu hình robot đa hướng, thiết kế ba bánh omni được ưa chuộng nhờ tính đơn giản và hiệu quả [3]. Tuy nhiên, việc điều khiển loại robot này gặp nhiều thách thức do tính phi tuyến trong mô hình động học và các yếu tố bất định như trượt bánh xe, ma sát và nhiễu từ cảm biến. Các phương pháp điều khiển truyền thống như PID thường không đủ khả năng xử lý hiệu quả các yếu tố phức tạp này [4].

Điều khiển dự báo mô hình (Model Predictive Control - MPC) là một phương pháp điều khiển tiên tiến có khả năng dự đoán hành vi tương lai của hệ thống và tối ưu hóa các đầu vào điều khiển [5-7]. Tuy nhiên, việc triển khai MPC thường đòi hỏi các công cụ tối ưu hóa phức tạp, gây khó khăn trong việc áp dụng trên các hệ thống nhúng với tài nguyên tính toán hạn chế.

Trong nghiên cứu này, nội dung trình bày một phương pháp MPC đơn giản cho robot đa hướng ba bánh, sử dụng chiến lược tìm kiếm tối ưu hiệu quả để giảm thiểu độ phức tạp tính toán. Mục tiêu chính của nghiên cứu bao gồm: (1) Phát triển một phương pháp MPC đơn giản và hiệu quả cho robot đa hướng ba bánh; (2) Mô phỏng và đánh giá hiệu suất của phương pháp trên các quỹ đạo khác nhau; (3) Phân tích

khả năng xử lý các yếu tố bất định như trượt bánh xe và nhiễu cảm biến.

2. VẬT LIỆU VÀ PHƯƠNG PHÁP NGHIÊN CỨU

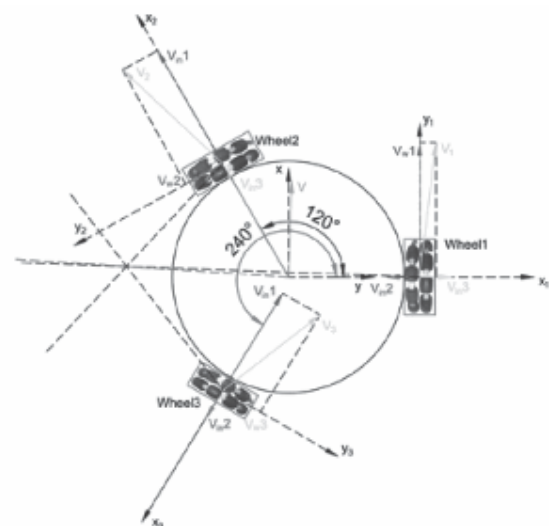
2.1. Mô hình Robot đa hướng ba bánh

2.1.1. Cấu trúc vật lý

Robot đa hướng ba bánh được nghiên cứu có dạng hình tròn với ba bánh xe omni được đặt cách đều nhau 120° trên đường tròn (Hình 1). Mỗi bánh xe có thể quay quanh trục của nó và có khả năng di chuyển vuông góc với hướng quay, cho phép robot di chuyển theo bất kỳ hướng nào trong mặt phẳng.

Các thông số vật lý chính của robot bao gồm:

- Bán kính robot (R): 0,2m;
- Bán kính bánh xe (r): 0,05m;
- Khoảng cách từ tâm robot đến tâm bánh xe (L): 0,15m;
- Vận tốc bánh xe tối đa: 40 rad/s.



Hình 1. Cấu trúc vật lý của robot đa hướng ba bánh, thể hiện vị trí và hướng của các bánh xe

2.1.2. Mô hình động học

Động học thuận (Forward Kinematics): động học thuận chuyển đổi từ vận tốc bánh xe thành vận tốc của robot trong không gian Cartesian:

$$\begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} = \frac{r}{3} \begin{bmatrix} -\sin\left(\theta + \frac{\pi}{3}\right) & \sin\left(\theta + \frac{\pi}{3}\right) & -\sin\left(\theta + \frac{5\pi}{3}\right) \\ \cos\left(\theta + \frac{\pi}{3}\right) & \cos\left(\theta + \frac{\pi}{3}\right) & \cos\left(\theta + \frac{5\pi}{3}\right) \\ \frac{1}{L} & \frac{1}{L} & \frac{1}{L} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}$$

Trong đó:

- $\omega, \omega_2, \omega_3$ là vận tốc góc của ba bánh xe.
- θ là hướng của robot trong hệ tọa độ toàn cục.
- v_x, v_y là vận tốc dịch chuyển trong hệ tọa độ toàn cục.
- ω là vận tốc góc của robot.
- r là bán kính bánh xe.
- L là khoảng cách từ tâm robot đến tâm bánh xe.

Động học ngược (Inverse Kinematics): Động học ngược chuyển đổi từ vận tốc mong muốn của robot trong không gian Cartesian (vận tốc dịch chuyển v_x, v_y và vận tốc góc ω) thành vận tốc góc của mỗi bánh xe. Mỗi quan hệ này được mô tả bởi phương trình:

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} -\sin\left(\theta + \frac{\pi}{3}\right) & \cos\left(\theta + \frac{\pi}{3}\right) & L \\ -\sin\left(\theta + \pi\right) & \cos\left(\theta + \pi\right) & L \\ -\sin\left(\theta + \frac{5\pi}{3}\right) & \cos\left(\theta + \frac{5\pi}{3}\right) & L \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix}$$

Các phương trình động học này là cơ sở để phát triển bộ điều khiển MPC và mô phỏng hành vi của robot.

2.1.3. Mô hình trượt và ma sát

Để mô phỏng chính xác hơn hành vi

của robot trong môi trường thực tế, các yếu tố ảnh hưởng sau được đưa vào mô hình:

1) Ma sát: Nghiên cứu xem xét hai loại ma sát ảnh hưởng đến hoạt động robot: ma sát mặt đất (hệ số 0.01) phát sinh tại điểm tiếp xúc giữa bánh xe và bề mặt, và ma sát khớp (hệ số 0.005) trong các cơ cấu truyền động. Việc mô hình hóa cả hai loại ma sát giúp mô phỏng chính xác hơn hành vi robot trong môi trường thực tế và hỗ trợ bộ điều khiển MPC dự đoán, bù trừ hiệu quả các ảnh hưởng động lực học phức tạp.

2) Hiệu ứng trượt bánh xe: Nghiên cứu mô hình hóa hai loại trượt bánh xe: trượt dọc (theo hướng quay - Longitudinal slip) với phân phối Gaussian (trung bình 0.01, độ lệch chuẩn 0.005) và trượt ngang (vuông góc với hướng quay - Lateral slip) cũng với phân phối Gaussian (trung bình 0.005, độ lệch chuẩn 0.002). Ảnh hưởng của trượt được mô hình hóa bằng cách giảm vận tốc thực tế so với vận tốc lý thuyết của bánh xe. Việc mô phỏng cả hai hiệu ứng này giúp bộ điều khiển MPC dự đoán và thích ứng tốt hơn với các điều kiện bất định trong môi trường thực tế. Ảnh hưởng của trượt bánh xe được mô hình hóa bằng cách giảm vận tốc thực tế của bánh xe so với vận tốc lý thuyết:

$$\omega_{\text{actual}} = \omega_{\text{command}} \times (1 - \text{slip})$$

Với góc trượt ngang, chúng tôi sử dụng mô hình:

$$\begin{aligned} v_x &= v_{\text{magnitude}} \times \cos(\text{slip_angle}) \times (1 - \text{lateral_slip_factor} \times \sin^2(\text{slip_angle})^2) \\ v_y &= v_{\text{magnitude}} \times \sin(\text{slip_angle}) \times (1 - \text{lateral_slip_factor} \times \cos^2(\text{slip_angle})^2) \end{aligned}$$

2.2. Phương pháp điều khiển dự báo mô hình (MPC)

2.2.1. Nguyên lý hoạt động

Điều khiển dự báo mô hình (MPC) là một phương pháp điều khiển tiên tiến dựa trên 

việc dự đoán hành vi tương lai của hệ thống và tối ưu hóa các đầu vào điều khiển. Phương pháp MPC hoạt động theo các bước cơ bản sau:

1) Đo trạng thái hiện tại: Xác định vị trí (x, y) và hướng (θ) hiện tại của robot.

2) Dự đoán trạng thái tương lai: Sử dụng mô hình động học để dự đoán vị trí và hướng trong N bước tiếp theo (horizon N).

3) Xác định quỹ đạo mục tiêu: Xác định các điểm mục tiêu trên quỹ đạo mong muốn cho N bước tiếp theo.

4) Tối ưu hóa điều khiển: Tính toán các đầu vào điều khiển tối ưu để giảm thiểu sai số giữa quỹ đạo dự đoán và quỹ đạo mục tiêu.

5) Áp dụng điều khiển: Chỉ áp dụng đầu vào điều khiển cho bước đầu tiên.

6) Lặp lại chu trình: Cập nhật trạng thái hiện tại và lặp lại các bước trên.

2.2.2. Dự đoán trạng thái

Để dự đoán trạng thái tương lai của robot, nghiên cứu sử dụng mô hình động học đơn giản hóa. Cụ thể, tại thời điểm k và với đầu vào điều khiển là vận tốc bánh xe $\omega_1, \omega_2, \omega_3$, trạng thái tiếp theo của robot được dự đoán như sau:

1) Tính toán vận tốc robot từ vận tốc bánh xe sử dụng phương trình động học thuận.

2) Cập nhật hướng của robot: $\theta_{k+1} = \theta_k + \omega \times dt$. Với dt là khoảng thời gian giữa hai bước liên tiếp.

3) Cập nhật vị trí của robot:

$$x_{k+1} = x_k + (v_x \cos(\theta_k) - v_y \sin(\theta_k)) \times dt$$

$$y_{k+1} = y_k + (v_x \sin(\theta_k) + v_y \cos(\theta_k)) \times dt$$

Với $dt = 0,05s$ là khoảng thời gian giữa hai bước liên tiếp.

Bằng cách lặp lại quá trình này trạng thái của robot trong N bước tiếp theo ($N = 15$ trong nghiên cứu này, tương đương với 0,75 giây) có thể được dự đoán.

2.2.3. Hàm chi phí

Hàm chi phí được thiết kế để cân bằng giữa độ chính xác theo dõi quỹ đạo và nỗ lực điều khiển. Hàm chi phí tổng thể bao gồm ba thành phần chính:

1) Chi phí vị trí: Phạt sai lệch giữa vị trí dự đoán và vị trí mục tiêu:

$$J_{\text{position}} = w_{\text{pos}} \times \sum_{i=1}^N \gamma^i \times \|(x_i, y_i) - (x_{\text{target},i}, y_{\text{target},i})\|^2$$

2) Chi phí hướng: Phạt sai lệch giữa vị trí dự đoán và hướng mong muốn:

$$J_{\text{orientation}} = w_{\text{orient}} \times \sum_{i=1}^N \gamma^i \times |\theta_i - \theta_{\text{desired},i}|$$

3) Chi phí điều khiển: Phạt nỗ lực điều khiển để tránh thay đổi đột ngột:

$$J_{\text{control}} = w_{\text{control}} \times \sum_{i=1}^N \gamma^i \times \|\omega_i\|^2$$

Trong đó:

$w_{\text{pos}}, w_{\text{orient}}, w_{\text{control}}$ là các trọng số tương ứng với từng thành phần.

$\gamma \in (0, 1]$ là hệ số giảm (discount factor) để ưu tiên các hành động sớm hơn trong horizon.

$\|\cdot\|$ ký hiệu chuẩn Euclidean.

Trong nghiên cứu này, các giá trị được sử dụng bao gồm $w_{pos} = 40.0$, $w_{orient} = 5.0$, $w_{control} = 0.05$ và $\gamma = 0.95$. Những tham số này được lựa chọn cẩn thận để cân bằng giữa độ chính xác theo dõi quỹ đạo và nỗ lực điều khiển, đồng thời tạo ra phản ứng điều khiển mượt mà và ổn định cho robot đa hướng ba bánh.

2.2.4. Phương pháp tối ưu hóa đơn giản

Thay vì sử dụng các công cụ tối ưu hóa phức tạp, nghiên cứu này đề xuất một phương pháp tìm kiếm hiệu quả trong không gian điều khiển. Phương pháp này bao gồm các bước sau:

1) Tính toán điều khiển cơ sở: Dựa trên lỗi vị trí và hướng hiện tại, tính toán một đầu vào điều khiển cơ sở: $v_{linear} = \min(K_p \times \|e_{pos}\|, v_{max}) \omega = K_\omega \times e_\theta$. Trong đó: e_{pos} là lỗi vị trí hiện tại, e_θ là lỗi hướng hiện tại, K_p , K_ω là các hệ số tỉ lệ, v_{max} là giới hạn vận tốc.

2) Chuyển đổi sang vận tốc bánh xe: Sử dụng động học ngược để chuyển đổi vận tốc mong muốn thành vận tốc bánh xe cơ sở.

3) Tạo các biến thể điều khiển: Tạo một tập hợp các biến thể của điều khiển cơ sở bằng cách thay đổi các hệ số v_x , v_y , ω trong một phạm vi xác định:

$$\omega_{test} = \begin{bmatrix} \omega_{1,base} \times v_{xvar} \\ \omega_{2,base} \times v_{yvar} \\ \omega_{3,base} \times \omega_{var} \end{bmatrix}$$

4) Đánh giá chi phí: Tính toán chi phí của mỗi biến thể bằng cách mô phỏng phản ứng của robot và tính tổng chi phí trong horizon.

5) Chọn điều khiển tối ưu: Chọn biến thể có chi phí thấp nhất làm đầu vào điều khiển.

Thuật toán 1: Thuật toán tối ưu hóa đơn giản cho MPC

```

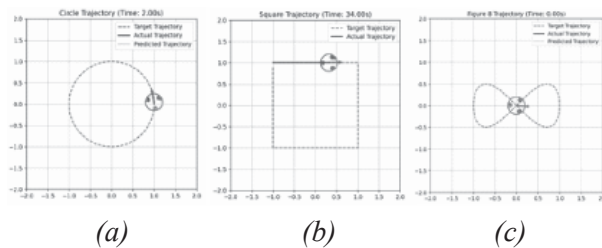
1: procedure OPT-WHEEL-VEL(x, y,  $\theta$ , target_trajectory)
2:   e_pos  $\leftarrow$  compute_position_error(x, y, target_trajectory)
3:   e_theta  $\leftarrow$  compute_orientation_error( $\theta$ , target_trajectory)
4:   v_linear  $\leftarrow$  compute_linear_velocity(e_pos)
5:    $\omega \leftarrow$  compute_angular_velocity(e_theta)
6:   [ $\omega_{1\_base}$ ,  $\omega_{2\_base}$ ,  $\omega_{3\_base}$ ]  $\leftarrow$  kinematic_transform(v_linear,  $\omega$ )
7:   cost_min  $\leftarrow \infty$ 
8:    $\omega_1, \omega_2, \omega_3 \leftarrow 0, 0, 0$ 
9:   for v_x_var  $\in$  [0.7, 1.3] do
10:    for v_y_var  $\in$  [0.7, 1.3] do
11:      for  $\omega_{var} \in$  [0.7, 1.3] do
12:        v_test  $\leftarrow$  v_linear  $\times$  [v_x_var, v_y_var]
13:         $\omega_{test} \leftarrow \omega \times \omega_{var}$ 
14:        [ $\omega_{1\_test}$ ,  $\omega_{2\_test}$ ,  $\omega_{3\_test}$ ]  $\leftarrow$  kinematic_transform(v_test,  $\omega_{test}$ )
15:        [ $\omega_{1\_test}$ ,  $\omega_{2\_test}$ ,  $\omega_{3\_test}$ ]  $\leftarrow$  clip_to_limits( $\omega_{1\_test}$ ,  $\omega_{2\_test}$ ,  $\omega_{3\_test}$ )
16:        cost  $\leftarrow$  simulate_cost([ $\omega_{1\_test}$ ,  $\omega_{2\_test}$ ,  $\omega_{3\_test}$ ], target_trajectory)
17:        if cost < cost_min then
18:          cost_min  $\leftarrow$  cost
19:           $\omega_1, \omega_2, \omega_3 \leftarrow \omega_{1\_test}, \omega_{2\_test}, \omega_{3\_test}$ 
20:        end if
21:      end for
22:    end for
23:  end for
24:  return  $\omega_1, \omega_2, \omega_3$ 
25: end procedure

```



2.3. Thiết lập mô phỏng

Các mô phỏng được tiến hành để đánh giá hiệu suất của bộ điều khiển MPC với thông số: tần số điều khiển 20Hz ($\Delta t = 0.05s$), thời gian mô phỏng 30 giây, horizon dự đoán 15 bước (0.75 giây), và 7 mẫu cho mỗi chiều của biến điều khiển. Ba quỹ đạo tiêu chuẩn được sử dụng: hình tròn (bán kính 1.0m) để đánh giá khả năng duy trì vận tốc góc không đổi; hình vuông (cạnh 1.5m) để thử thách khả năng xử lý chuyển hướng đột ngột; và hình số 8 (2.0m $\times$ 1.0m) để kiểm tra khả năng thích ứng với thay đổi đồng thời vận tốc và hướng (Hình 2). Mô phỏng được thực hiện trong Python với mô hình robot và bộ điều khiển MPC như đã mô tả.



Hình 2. Ba quỹ đạo được sử dụng trong mô phỏng:
(a) Hình tròn, (b) Hình vuông, (c) Hình số 8

3. KẾT QUẢ VÀ THẢO LUẬN

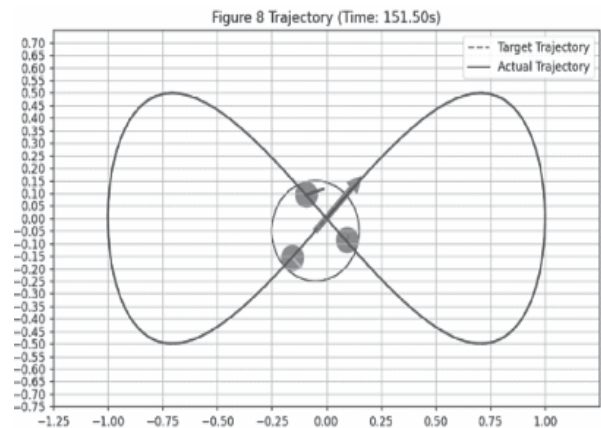
3.1. Hiệu suất theo dõi quỹ đạo

Hiệu suất theo dõi quỹ đạo của phương pháp MPC được đánh giá qua các thí nghiệm mô phỏng trên ba loại quỹ đạo đặc trưng. Bảng 1 trình bày kết quả định lượng về sai số theo dõi quỹ đạo.

Bảng 1. Hiệu suất theo dõi quỹ đạo của MPC

Quỹ đạo	Sai số trung bình (m)	Sai số tối đa (m)	RMSE (m)
Hình tròn	0,0009	0,0012	0,0009
Hình vuông	0,0009	0,0118	0,0013
Hình số 8	0,0010	0,0021	0,0010

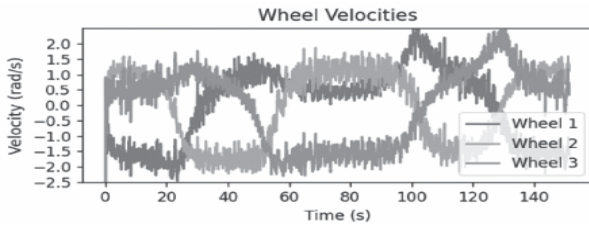
Kết quả cho thấy phương pháp MPC đạt độ chính xác cao với sai số trung bình dưới 1mm trên cả ba quỹ đạo. Quỹ đạo hình tròn cho hiệu suất tốt nhất (sai số tối đa 1,2mm) nhờ tính liên tục trong chuyển động. Quỹ đạo hình vuông đặt ra thách thức lớn hơn với sai số tối đa lên đến 11,8mm tại các góc, nhưng sai số trung bình vẫn duy trì ở mức thấp (0,9mm). Quỹ đạo hình số 8, dù đòi hỏi thay đổi liên tục cả tốc độ và hướng, vẫn đạt hiệu suất tốt với sai số tối đa chỉ 2,1mm. Hình 3 cho thấy quỹ đạo thực tế gần như trùng khớp với quỹ đạo mong muốn, với sai lệch không đáng kể ngay cả tại các điểm uốn cong phức tạp.



Hình 3. Quỹ đạo thực tế của robot khi sử dụng phương pháp MPC trên quỹ đạo hình số 8.

3.2. Phân tích vận tốc bánh xe và lỗi vị trí

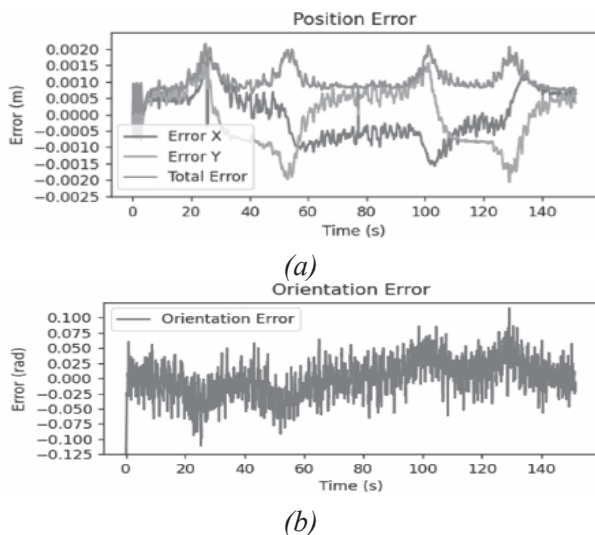
Hình 4 trình bày biến thiên vận tốc của ba bánh xe khi robot theo dõi quỹ đạo hình số 8. Quan sát cho thấy bộ điều khiển MPC tạo ra hồ sơ vận tốc mượt mà cho cả ba bánh xe, với biên độ dao động nằm trong khoảng -2,6 rad/s đến 3,0 rad/s. Đặc biệt, tại các điểm chuyển hướng, sự thay đổi vận tốc diễn ra liên tục mà không có đột biến, góp phần giảm thiểu hiện tượng rung lắc và tăng độ ổn định trong quá trình di chuyển.



Hình 4. Vận tốc của ba bánh xe khi robot theo dõi quỹ đạo hình số 8.

Hình 5(a) thể hiện sai số vị trí theo trục x và trục y khi robot theo dõi quỹ đạo hình số 8. Sai số dao động với biên độ nhỏ và hội tụ nhanh về không sau các điểm chuyển hướng, với giá trị tối đa không vượt quá 2,1mm. Điều này chứng tỏ khả năng thích ứng nhanh của bộ điều khiển MPC với các thay đổi trong quỹ đạo mục tiêu.

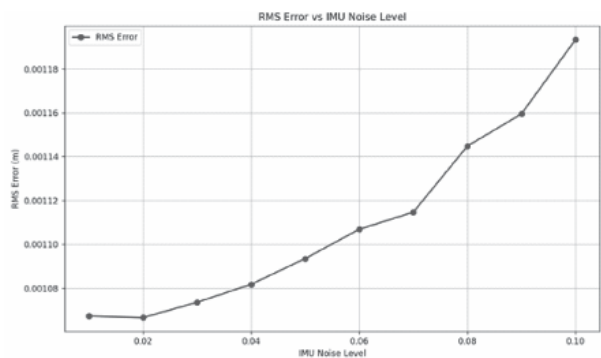
Hình 5(b) minh họa sai số hướng quay của robot, dao động trong khoảng $\pm 0,12$ radian (khoảng $\pm 6,9$ độ). Đáng chú ý, tại các điểm uốn cong của quỹ đạo hình số 8, bộ điều khiển MPC có khả năng dự đoán và điều chỉnh hướng trước, giảm thiểu hiện tượng quá điều chỉnh (overshoot) thường gặp trong các phương pháp điều khiển truyền thống.



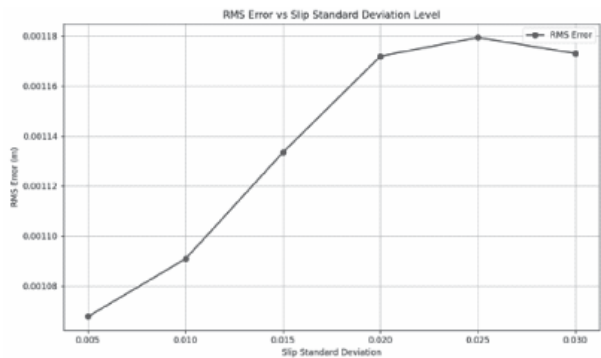
Hình 5. Sai số khi robot theo dõi quỹ đạo hình số 8: (a) Vị trí theo trục x, trục y; (b) Hướng quay.

3.3. Ảnh hưởng của nhiễu và trượt bánh xe

Hình 6 thể hiện ảnh hưởng của nhiễu IMU đến hiệu suất theo dõi quỹ đạo của robot khi di chuyển theo hình số 8. Kết quả cho thấy phương pháp MPC duy trì khả năng chống chịu tốt khi mức độ nhiễu tăng lên. Cụ thể, khi độ lệch chuẩn nhiễu tăng từ 0,01 đến 0,1 radian, sai số trung bình chỉ tăng từ 0,00108m lên 0,00118m, tương đương với mức tăng khoảng 9%.



Hình 6. Sai số trung bình của robot khi theo dõi quỹ đạo hình số 8 với các mức độ nhiễu IMU khác nhau.



Hình 7. Ảnh hưởng của trượt dọc đến hiệu suất theo dõi quỹ đạo.

Tương tự, Hình 7 minh họa ảnh hưởng của hiệu ứng trượt dọc đến hiệu suất theo dõi quỹ đạo. Khi hệ số trượt dọc tăng từ 0,005 đến 0,03, sai số trung bình tăng từ 0,00108m lên khoảng 0,00118m (tăng 9,3%). Kết quả này chứng tỏ bộ điều khiển MPC có khả năng thích

ứng tốt với các điều kiện bất lợi nhờ cơ chế dự đoán và tối ưu hóa điều khiển liên tục.

3.4. Thời gian tính toán

Thời gian tính toán trung bình cho mỗi vòng lặp điều khiển là 5,587ms, đủ để đáp ứng yêu cầu điều khiển thời gian thực với tần số 20Hz (chu kỳ 50ms).

3.5. Thảo luận

3.5.1. Ưu điểm của phương pháp MPC đề xuất

Phương pháp MPC đạt hiệu suất theo dõi quỹ đạo cao với sai số trung bình dưới 2mm, đủ cho nhiều ứng dụng thực tế. Khả năng dự đoán và thích ứng giúp giảm thiểu hiện tượng quá điều chỉnh tại các điểm chuyển hướng. Bộ điều khiển thể hiện khả năng chống chịu tốt với nhiễu và trượt bánh xe, đồng thời duy trì chi phí tính toán hợp lý (5.587ms/chu kỳ) cho phép triển khai thời gian thực.

3.5.2. Hạn chế và cải tiến tiềm năng

Phương pháp tìm kiếm hiện tại có thể hội tụ đến tối ưu cục bộ thay vì tối ưu toàn cục. Mô hình động học được sử dụng bỏ qua một số hiệu ứng động lực học phức tạp, và phương pháp chưa xử lý các ràng buộc như tránh chướng ngại vật. Các hướng cải tiến bao gồm: cải thiện chiến lược tìm kiếm, tích hợp mô hình học máy để mô phỏng chính xác hơn các hiệu ứng phi tuyến, và phát triển phương pháp xử lý các ràng buộc phức tạp.

3.5.3. Ứng dụng thực tế

Phương pháp MPC đề xuất có tiềm năng ứng dụng trong robot dịch vụ di chuyển trong môi trường phức tạp, AGV trong logistic

tự động, robot kiểm tra và giám sát cơ sở hạ tầng, cũng như nền tảng robot giáo dục chi phí thấp.

4. KẾT LUẬN

Nghiên cứu này đã trình bày một phương pháp điều khiển dự báo mô hình (MPC) đơn giản và hiệu quả cho robot đa hướng ba bánh. Phương pháp sử dụng chiến lược tìm kiếm tối ưu đơn giản nhằm giảm thiểu độ phức tạp tính toán mà vẫn duy trì hiệu suất cao. Kết quả mô phỏng trên ba loại quỹ đạo khác nhau cho thấy bộ điều khiển đạt sai số trung bình dưới 1mm và sai số tối đa dưới 12mm, đồng thời thể hiện khả năng chống chịu tốt với nhiễu IMU và hiệu ứng trượt bánh xe. Với thời gian tính toán trung bình 5.587ms cho mỗi chu kỳ điều khiển, phương pháp phù hợp cho triển khai thời gian thực trên các hệ thống nhúng. Những kết quả này mở ra tiềm năng ứng dụng trong nhiều lĩnh vực từ robot dịch vụ đến logistic tự động, nơi yêu cầu độ chính xác cao trong di chuyển.

Lời cảm ơn:

Nghiên cứu này được tài trợ bởi Trường Đại học Bách Khoa, Đại học Quốc gia Thành phố Hồ Chí Minh thông qua đề tài mã số T-CK-2023-04. Chúng tôi cảm ơn sự hỗ trợ về thời gian và cơ sở vật chất từ Trường Đại học Bách Khoa (HCMUT), Đại học Quốc gia Thành phố Hồ Chí Minh (VNU-HCM). ❖

Ngày nhận bài: **02/4/2025**

Ngày phản biện: **17/4/2025**

Tài liệu tham khảo:

- [1]. Campion, G., Bastin, G., & D'Andréa-Novel, B. (1996), “*Structural properties and classification of kinematic and dynamic models of wheeled mobile robots*”. IEEE Transactions on Robotics and Automation, 12(1), 47-62.
- [2]. Li, X., & Zell, A. (2009), “*Motion control of an omnidirectional mobile robot*”. In Informatics in Control, Automation and Robotics (pp. 181-193). Springer, Berlin, Heidelberg.
- [3]. Doroftei, I., Grosu, V., & Spinu, V. (2007), “*Omnidirectional mobile robot – Design and implementation*”. In Bioinspiration and Robotics Walking and Climbing Robots (pp. 511-528). IntechOpen.
- [4]. Åström, K. J., & Hägglund, T. (2006), “*Advanced PID control*”. ISA-The Instrumentation, Systems, and Automation Society.
- [5]. Killpack, M., Deyle, T., Anderson, C., & Kemp, C. C. (2010, December), “*Visual odometry and control for an omnidirectional mobile robot with a downward-facing camera*”. In 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (pp. 139-146). IEEE.
- [6]. Klancar, G., & Skrjanc, I. (2007), “*Tracking-error model-based predictive control for mobile robots in real time*”. Robotics and Autonomous Systems, 55(6), 460-469.
- [7]. Wang, C., Liu, X., Yang, X., Hu, F., Jiang, A., & Yang, C. (2018), “*Trajectory tracking of an omni-directional wheeled mobile robot using a model predictive control strategy*”. Applied Sciences, 8(2), 231.