

Lập kế hoạch chuyển động thông minh cho robot di động ba bánh bằng điều khiển dự báo theo mô hình phi tuyến

Intelligent motion planning of three-wheeled mobile robots via nonlinear model predictive control

Quang Duy Do Nguyen^{1,2}, Xuan Minh Dinh^{3,5}, Thanh Loan Pham², Thanh Trung Duong²,
Xuan Hai Le^{4*}, Huy Tung Pham⁶

¹Trường Đại học Công nghiệp Hà Nội, Hà Nội, Việt Nam

²Trường Đại học Mô địa chất, Hà Nội, Việt Nam

³Khoa Cơ khí - Cơ điện tử, Đại học Phenikaa, Yên Nghĩa, Hà Đông, Hà Nội, Việt Nam

⁴Trường Quốc tế, Đại học Quốc gia Hà Nội, 144 Xuân Thủy, Cầu Giấy, Hà Nội, Việt Nam

⁵Trung tâm Nghiên cứu Phát triển và Xưởng sản xuất, Công ty Cổ phần Đầu tư và Công nghệ HTI, Hà Nội, Việt Nam

⁶Trường Đại học Hàng hải Việt Nam, Hải Phòng, Việt Nam

*Corresponding author E-mail: hailx@vnu.edu.vn

DOI: <https://doi.org/10.64032/mca.v29i4.350>

Abstract

This study proposes an intelligent, safe, and efficient motion planning strategy for a Three-Wheeled Mobile Robots, based on the integration of Nonlinear Model Predictive Control (NMPC) and Control Barrier Functions (CBF), collectively referred to as NMPC-CBF. The approach is implemented in MATLAB R2023b using the CasADi toolkit and the IPOPT solver, and evaluated through two challenging simulation scenarios involving both static and dynamic obstacles. The results demonstrate that NMPC-CBF not only enables effective collision avoidance but also generates shorter trajectories, smaller position and orientation errors, and smoother, more stable control signals compared to conventional NMPC. Thanks to its flexible response and consistent control performance in highly dynamic environments, the proposed method shows promising potential for practical applications in robotic systems operating under complex conditions.

Keywords: Nonlinear Model Predictive Control, Control Barrier Functions, Motion Planning, Mobile robots, CasADi toolkit

Các chữ viết tắt

3WMRs	Three-Wheeled Mobile Robots
NMPC	Nonlinear Model Predictive Control
GP	Global Planning
LC	Local Planning
RRT	Rapidly-exploring Random Tree
APF	Artificial Potential Field
DRL	Deep Reinforcement Learning
DWA	Dynamic Window Approach
TEB	Timed Elastic Band
ROS	Robot Operating System
MPC	Model Predictive Control
NMPC	Nonlinear Model Predictive Control
RK4	Fourth Order Runge-Kutta
OCF	Optimal Control Problem
CBF	Control Barrier Function
BT	Bug-Type
IPOPT	Interior Point OPTimizer
NLP	Nonlinear Programming Problem

Tóm tắt

Nghiên cứu này đề xuất một chiến lược lập kế hoạch chuyển động thông minh, an toàn và hiệu quả cho robot di động ba bánh, dựa trên sự tích hợp giữa phương pháp điều khiển dự báo theo mô hình phi tuyến (NMPC) và hàm rào điều khiển (CBF), gọi chung là NMPC-

CBF. Phương pháp được triển khai trong MATLAB R2023b sử dụng công cụ CasADi và bộ giải IPOPT, và được đánh giá qua hai kịch bản mô phỏng đầy thách thức liên quan đến cả các vật cản cố định và vật cản chuyển động. Các kết quả thu được cho thấy rằng NMPC-CBF không chỉ cho phép robot tránh va chạm hiệu quả mà còn tạo ra các quỹ đạo ngắn hơn, các sai số vị trí và hướng nhỏ hơn, cùng các tín hiệu điều khiển mượt mà, và ổn định hơn so với phương pháp NMPC truyền thống. Nhờ khả năng phản ứng linh hoạt và duy trì hiệu suất điều khiển cao trong môi trường có nhiều biến động, phương pháp được đề xuất cho thấy tiềm năng đầy hứa hẹn cho các ứng dụng thực tiễn trong các hệ thống robot hoạt động dưới các điều kiện vận hành phức tạp.

1. Giới thiệu

Trong những năm gần đây, robot di động ngày càng thể hiện được tầm quan trọng trong nhiều lĩnh vực của đời sống hiện đại. Nhờ vào khả năng di chuyển linh hoạt và thực hiện các nhiệm vụ tự động, chúng đã được ứng dụng rộng rãi trong công nghiệp, dịch vụ thông minh, và an ninh và quốc phòng. Bởi vậy, robot di động đã trở thành chủ đề thu hút sự quan tâm lớn từ cộng đồng các nhà khoa học và kỹ sư, đặc biệt là trong các nghiên cứu về robot và tự động hóa [1]. Trong bối cảnh này, các robot di động dạng ba bánh (3WMRs) được xem là một lựa chọn hoàn hảo cho nghiên cứu và triển khai thực tế nhờ thiết kế đơn giản, chi phí thấp, nhưng vẫn đảm bảo đủ độ linh hoạt để mô phỏng hành vi của các hệ thống robot phức tạp. Chính vì vậy, 3WMRs rất phù hợp để nghiên cứu và kiểm thử các thuật toán điều khiển và lập kế hoạch quỹ đạo.

Để giúp 3WMRs có thể vận hành đáng tin cậy trong môi trường thực địa, việc phát triển các phương pháp lập kế hoạch đường đi thông minh là vô cùng quan trọng. Bài toán này không chỉ dừng lại ở việc tìm một đường đi khả thi từ vị trí ban đầu đến vị trí mục tiêu, mà còn bao gồm các yêu cầu quan trọng như tránh va chạm với vật cản, tối ưu hóa quãng đường di chuyển, đảm bảo sự trơn tru trong chuyển động, và tuân thủ các ràng buộc động học của hệ thống.

Trong những năm qua, nhiều giải pháp điều hướng chuyển động đã được nghiên cứu và đề xuất, nhằm giải quyết các yêu cầu nhiệm vụ khắt khe đối với 3WMRs. Các thuật toán lập kế hoạch chuyển động cho robot có thể được phân chia theo nguyên lý hoạt động thành hai nhóm chính bao gồm các thuật toán truyền thống và các thuật toán dựa trên học máy [2]. Ngoài ra, nó cũng có thể được phân loại theo phạm vi hoạt động thành hai nhóm bao gồm các giải thuật lập kế hoạch đường đi toàn cục (GP), và các giải thuật lập kế hoạch đường đi cục bộ (LP) [3].

Trong nhóm thuật toán phục vụ lập kế hoạch toàn cục, các phương pháp tìm kiếm dựa trên đồ thị như A* [4] và Dijkstra [5] đã được ứng dụng rộng rãi, nhờ khả năng cung cấp các lời giải tối ưu về mặt hình học, giúp tìm đường đi ngắn nhất trên đồ thị có trọng số. Đặc biệt, A* là thuật toán mở rộng của Dijkstra khi kết hợp thêm thông tin Heuristic để dẫn hướng tìm kiếm, giúp cải thiện đáng kể thời gian tính toán. Tuy nhiên, các phương pháp này không xét đến ràng buộc động học và động lực học của robot, và không tính tới khả năng tránh vật cản động. Ngoài ra, khi áp dụng các phương pháp tìm kiếm dựa trên đồ thị, việc mở rộng kích thước lưới có thể ảnh hưởng tới hiệu quả tính toán. Đối với nhóm phương pháp dựa trên lấy mẫu, phương pháp cây ngẫu nhiên (RRT) [6] là một lựa chọn điển hình nhờ cấu trúc đơn giản và tốc độ xử lý nhanh, được ứng dụng rộng rãi trong điều hướng robot di động. Tuy nhiên, thuật toán này vẫn tồn tại một số hạn chế như hiệu suất lập kế hoạch thấp, tính ngẫu nhiên cao và chất lượng đường đi chưa tối ưu. Đặc biệt, RRT gặp hạn chế về khả năng xử lý hiệu quả các chướng ngại vật động, khiến nó khó được áp dụng trực tiếp trong các hệ thống robot hoạt động trong môi trường thực tế.

Để khắc phục những hạn chế của các phương pháp lập kế hoạch đường đi toàn cục, các thuật toán lập kế hoạch đường đi cục bộ đã được nghiên cứu, giúp robot phản ứng nhanh với các chướng ngại vật chưa được biết trước. Nghiên cứu [7] đã đề xuất sử dụng thuật toán trường tiềm năng nhân tạo (APF) để tránh vật cản cho robot. Phương pháp này được xây dựng dựa trên cơ sở vật lý, các chướng ngại vật trong môi trường tạo ra lực đẩy đối với robot, trong khi các điểm mục tiêu tạo ra lực hấp dẫn kéo robot, và robot di chuyển theo hướng có năng lượng tiềm năng nhỏ nhất dưới tác động của lực tổng hợp. Ưu điểm của APF là dễ triển khai trong thời gian thực và tính toán nhanh. Tuy nhiên, theo nghiên cứu [8], kỹ thuật APF bộc lộ hạn chế khi dễ rơi vào các điểm tối thiểu cục bộ, khiến APF chủ yếu được sử dụng cho lập kế hoạch đường đi trong môi trường tĩnh và không áp dụng được khi gặp chướng ngại vật động. Một cách tiếp cận khác tại nghiên cứu [9] đã đề xuất sử dụng logic mờ Mamdani để tránh chướng ngại vật với chi phí thấp, kết quả cho thấy robot thành công tới mục tiêu mà không va phải chướng ngại vật nào xung quanh. Tuy nhiên, theo nghiên cứu [10], việc sử dụng logic mờ để điều khiển trong thời gian thực đòi hỏi phải thiết kế luật một cách tỉ mỉ, và điều này phụ thuộc nhiều vào kinh nghiệm của người thiết

kế. Các luật mờ được tinh chỉnh không tốt có thể ảnh hưởng tiêu cực đến hiệu suất điều khiển, đặc biệt trong các tình huống vận hành phức tạp. Nghiên cứu [11] sử dụng học tăng cường sâu (DRL) để điều hướng tự động robot trong môi trường chưa biết trước, kết quả cho thấy robot có thể điều hướng mượt mà và có tiềm năng mở rộng sang các ứng dụng thực tế. Tuy nhiên, DRL phụ thuộc vào dữ liệu huấn luyện để hình thành các chính sách điều khiển tối ưu, do đó nếu dữ liệu chưa đủ tốt thì sẽ ảnh hưởng tới khả năng phản ứng của robot khi gặp các tình huống chưa từng thấy trong quá trình huấn luyện. Một cách tiếp cận lập kế hoạch đường đi cục bộ khác là sử dụng phương pháp tiếp cận của số động (DWA) [12], dựa trên việc lấy mẫu các vận tốc khả thi của robot di động trong một khoảng thời gian ngắn. Tuy nhiên, theo nghiên cứu [13], khi robot phải đối mặt với các chướng ngại vật di chuyển, thuật toán DWA có thể không phản ứng kịp thời nếu vận tốc của chướng ngại vật quá cao, từ đó làm tăng nguy cơ va chạm. Một số phương pháp cục bộ tiên tiến hơn là dựa trên dải đàn hồi thời gian (TEB) [14,15], trong đó quỹ đạo điều hướng được tối ưu hóa theo thời gian thực bằng cách biến đổi thành một bài toán tối ưu hóa phi tuyến nhiều ràng buộc. Hiện nay, TEB đã được tích hợp sẵn trong hệ điều hành robot (ROS) và được ứng dụng rộng rãi trong các hệ thống robot di động thực tế. Tuy nhiên, TEB thường cần được kết hợp với một thuật toán lập kế hoạch đường đi toàn cục, như A* được đề xuất trong [15], để đảm bảo khả năng điều hướng hiệu quả. Việc tích hợp với A* hoặc các thuật toán lập kế hoạch toàn cục dựa trên đồ thị nói chung đòi hỏi phải có sẵn bản đồ môi trường hoạt động, điều này làm hạn chế khả năng ứng dụng trong các tình huống môi trường xuất hiện nhiều vật cản ngẫu nhiên cố định hoặc di chuyển chưa biết trước.

Trong những năm gần đây, các phương pháp điều khiển dựa trên điều khiển mô hình dự báo (MPC) đã trở thành một xu hướng nổi bật trong lĩnh vực điều hướng robot, nhờ khả năng xử lý trực tiếp và hiệu quả các ràng buộc vật lý, đồng thời tối ưu hóa các chỉ tiêu chất lượng của hệ thống. Theo nghiên cứu [16], MPC được xem là một trong những kỹ thuật điều khiển tiên tiến, khi cho phép dự đoán hành vi tương lai của hệ thống thông qua mô hình, từ đó tối ưu hóa điều khiển theo thời gian thực. So với các bộ điều khiển truyền thống, MPC có ưu thế vượt trội trong việc đảm bảo cả hiệu suất, các ràng buộc, và an toàn vận hành. Hiện nay, nhờ sự phát triển của các thiết bị nhúng giá rẻ nhưng có hiệu năng cao, cùng với nguồn tài liệu về các mô hình vật lý của đối tượng ngày càng chính xác và phong phú, khiến rào cản kỹ thuật trong việc ứng dụng MPC vào các hệ thống robot thực tế đã được giảm thiểu đáng kể. Nghiên cứu [17] đã triển khai điều khiển dự đoán theo mô hình phi tuyến (NMPC) cho bài toán lập kế hoạch đường đi thông minh, nhằm giúp robot di động tránh vật cản động trong quá trình di chuyển. Kết quả thực nghiệm được trình bày tại nghiên cứu [18] đã chứng minh rằng NMPC có khả năng điều hướng robot một cách mượt mà, chính xác và an toàn trong thời gian thực. Tuy nhiên, các thử nghiệm này chủ yếu tập trung vào các tình huống đơn giản, trong đó robot chỉ gặp một vật cản ngẫu nhiên và cố định trên đường tới mục tiêu. Về mặt kỹ thuật tại [17,18], mô hình động học của robot được rời rạc hóa bằng phương pháp Runge-Kutta bậc 4 (RK4), từ đó thiết lập thành một bài toán điều khiển tối ưu (OCP) có ràng buộc. Hàm mục tiêu trong OCP được xây dựng bao gồm chi phí chạy dọc theo thời gian, phản ánh sai lệch giữa trạng thái hiện tại và tham chiếu, và chi phí điểm cuối, nhằm đảm

bảo ổn định và hội tụ về đích. Trong nghiên cứu này, việc tránh va chạm được thực hiện thông qua ràng buộc khoảng cách hình học giữa robot và vật cản tại từng bước tối ưu. Mặc dù cách tiếp cận này đơn giản và dễ tích hợp vào bài toán tối ưu, song nó có thể chưa đủ hiệu quả trong các môi trường có yếu tố vật cản phức tạp, chẳng hạn như khi robot phải đối mặt với nhiều vật cản cố định, vật cản di động, hoặc các tình huống kết hợp cả hai loại chướng ngại vật. Nghiên cứu [19, 20] đã áp dụng NMPC cho bài toán bám quỹ đạo kết hợp tránh vật cản cho robot di động và đã chứng minh được tính hiệu quả của các phương pháp. Cụ thể, nghiên cứu [19] đề xuất một khung điều khiển NMPC nhằm bám theo quỹ đạo toàn cục được tạo ra bởi thuật toán A*, và cho thấy hiệu năng vượt trội so với thuật toán TEB trong môi trường có vật cản tĩnh. Tuy nhiên, cách tiếp cận này chưa phù hợp với bài toán điều hướng trong môi trường có vật cản động, do chưa tích hợp các ràng buộc hoặc thành phần chi phí liên quan đến việc tránh va chạm với vật cản di chuyển. Ngoài ra, việc áp dụng NMPC kết hợp với A* đòi hỏi phải có sẵn bản đồ môi trường hoạt động, điều này làm giảm tính linh hoạt của hệ thống trong các tình huống thay đổi hoặc chưa biết trước môi trường vận hành.

Một số nghiên cứu gần đây [20, 21] đã đề xuất cách tiếp cận khác với các phương pháp truyền thống [17,18] sử dụng ràng buộc cứng để tránh va chạm, bằng cách chuyển điều kiện tránh va chạm thành ràng buộc mềm thông qua việc đưa các hàm phạt vào hàm mục tiêu của bài toán tối ưu. Phương pháp này giúp tăng khả năng tìm được nghiệm khả thi cho bài toán tối ưu trong các môi trường phức tạp hoặc khi robot bị giới hạn điều khiển nghiêm ngặt. Tuy nhiên, so với phương pháp sử dụng hàm rào (CBF) [22], ràng buộc mềm không đảm bảo tuyệt đối an toàn trong mọi tình huống, đặc biệt khi hệ số phạt không được lựa chọn phù hợp. Về mặt lý thuyết, CBF có thể đảm bảo chặt chẽ tính không xâm phạm nhờ vào việc biểu diễn điều kiện tránh va chạm dưới dạng ràng buộc bất đẳng thức nghiêm ngặt, áp dụng trực tiếp lên tín hiệu điều khiển theo thời gian thực. Tuy nhiên, trong thực tế mô phỏng và triển khai, CBF vẫn có thể xảy ra xâm phạm nhẹ do ảnh hưởng của sai số tính toán, ràng buộc điều khiển giới hạn, hoặc nhiều cảm biến, khiến robot tạm thời vượt qua biên an toàn trong một số tình huống ngắn hạn.

Từ những phân tích chi tiết đã trình bày, bài báo đề xuất cách tiếp cận kết hợp giữa điều khiển dự đoán mô hình phi tuyến (NMPC) và hàm rào điều khiển (CBF), gọi là NMPC-CBF, nhằm giải quyết bài toán lập kế hoạch đường đi thông minh cho robot di động ba bánh. Khác với các phương pháp NMPC truyền thống vốn sử dụng ràng buộc cứng hoặc ràng buộc mềm thông qua các hàm phạt, cách tiếp cận của chúng tôi tận dụng ưu điểm của CBF để đảm bảo rằng các ràng buộc an toàn luôn được thỏa mãn tại mọi thời điểm lấy mẫu. Trong khi đó, NMPC đảm nhiệm vai trò tối ưu hóa quỹ đạo di chuyển của robot thông qua việc cực tiểu hóa một hàm mục tiêu bao gồm chi phí theo thời gian, chi phí tại điểm cuối và độ mượt của tín hiệu điều khiển đầu ra, đồng thời tuân thủ chặt chẽ các ràng buộc động học và giới hạn vật lý của hệ thống. Khác với các thuật toán lập kế hoạch chuyển động toàn cục như A* [4], Dijkstra [5] và RRT [6], NMPC-CBF không phụ thuộc vào kích thước lưới và xử lý hiệu quả các chướng ngại vật động. So với một số thuật toán lập kế hoạch chuyển động cục bộ như APF [7], logic mờ Mamdani [9], DRL [11], NMPC-CBF không yêu cầu thiết kế luật thủ công dựa trên kinh nghiệm như trong logic mờ, không phụ thuộc vào dữ liệu huấn luyện như

DRL, và đặc biệt có thể hoạt động ổn định cả trong môi trường động thay vì chỉ phù hợp trong môi trường tĩnh như APF. So với các cách tiếp cận sử dụng NMPC trong [19-21], NMPC-CBF ưu việt hơn [19] khi vừa phù hợp với cả vật cản tĩnh lẫn vật cản động, về mặt lý thuyết nó vừa đảm bảo tính an toàn đủ nghiêm ngặt thông qua ràng buộc sử dụng CBF, thay vì sử dụng ràng buộc mềm như trong [20], [21].

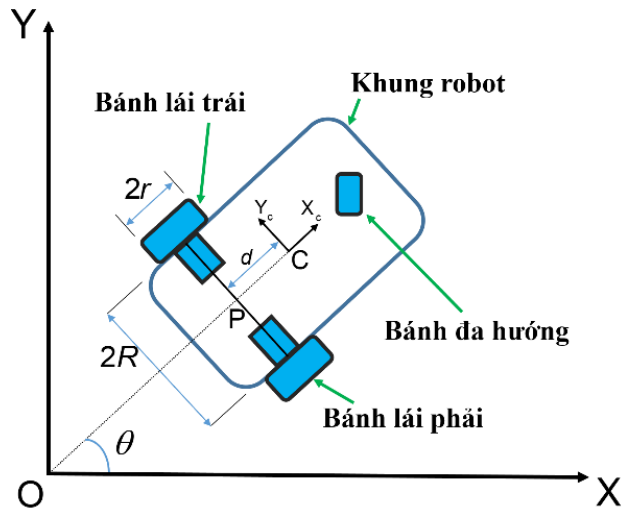
Để đánh giá hiệu quả của chiến lược điều khiển NMPC-CBF được đề xuất, chúng tôi đã thiết lập hai kịch bản mô phỏng riêng biệt trên phần mềm MATLAB R2023b, bao gồm môi trường có vật cản cố định và môi trường có vật cản chuyển động với độ phức tạp cao. Trong cả hai tình huống, robot đều phải đối mặt với nhiều vật cản được bố trí gần nhau, nhằm chỉ ra chuyển động với tốc độ khác nhau, nhằm mô phỏng những thách thức thực tế. Kết quả cho thấy NMPC-CBF giúp robot tạo ra quỹ đạo ngắn hơn, mượt mà và an toàn hơn, với tín hiệu điều khiển ổn định và phản ứng linh hoạt trước các thay đổi của môi trường. So với phương pháp NMPC truyền thống tại [17,18], chiến lược NMPC-CBF duy trì được khoảng cách an toàn tốt hơn và tránh được va chạm trong các tình huống phức tạp - điều đặc biệt quan trọng khi triển khai trong môi trường thực, nơi mà nguy cơ xảy ra va chạm có thể xuất hiện bất kỳ lúc nào trong quá trình di chuyển.

Ngoài ra, so với nghiên cứu [23], chúng tôi đã bổ sung thêm độ mịn của tín hiệu điều khiển vào hàm mục tiêu trong OCP nhằm làm mịn chuỗi điều khiển đầu ra. Việc này giúp giảm các thay đổi đột ngột trong tín hiệu điều khiển, nâng cao độ ổn định và độ bền của hệ thống cơ khí khi triển khai trong thực tế do đầu vào điều khiển được làm mịn. Đáng chú ý, các nghiên cứu [17,18] đều đã triển khai NMPC trên hệ thống robot thực và chứng minh được khả năng vận hành trong thời gian thực. Do đó, với đặc điểm tương đồng về cấu trúc bài toán giữa phương pháp của chúng tôi và nghiên cứu [17,18], có thể khẳng định rằng phương pháp NMPC-CBF do chúng tôi đề xuất cũng hoàn toàn phù hợp để triển khai trên hệ thống thực tế, với tiềm năng cải thiện hơn nữa về độ an toàn và độ chính xác trong các tình huống phức tạp hơn. Trong tương lai, các kết quả thử nghiệm thực tế sẽ được triển khai để khẳng định thêm tính hiệu quả của chiến lược NMPC-CBF do chúng tôi đề xuất.

Các nội dung còn lại của nghiên cứu được sắp xếp như sau: Phần 2 trình bày mô hình động học của robot di động ba bánh; Phần 3 trình bày chi tiết chiến lược lập kế hoạch đường đi thông minh do nhóm tác giả đề xuất. Kết quả mô phỏng và phân tích định lượng được trình bày rõ trong Phần 4. Cuối cùng, Phần 5 nêu các kết luận rút ra từ nghiên cứu và đề xuất định hướng phát triển trong tương lai.

2. Mô hình động học của robot di động 3 bánh

Mô hình của một robot di động vi sai 3 bánh như được thể hiện trên **Hình 1**, bao gồm hai bánh đẩy phía sau được điều khiển bởi hai động cơ một chiều độc lập và một bánh tự lựa hướng phía trước. Trong trường hợp tổng quát, giả sử tâm của robot tại C cách trung điểm P của hai bánh một khoảng cách d . Trong bài báo này, chúng tôi giả thiết rằng trọng tâm của robot được thiết kế trùng với điểm P , tức là làm cho khoảng cách $d=0$. Ngoài ra, r là bán kính của mỗi bánh xe và R biểu thị một nửa khoảng cách giữa 2 bánh đằng sau.



Hình 1: Robot di động 3 bánh di chuyển trong không gian 2 chiều

Trạng thái của một robot di động vi sai ba bánh trong không gian hai chiều được xác định bởi véc tơ $\chi = [x, y, \theta]^T \in \mathbb{R}^{3 \times 1}$, trong đó (x, y) là tọa độ vị trí của robot được tính theo đơn vị mét (m), và θ là góc định hướng của robot được tính theo đơn vị radian (rad). Đầu vào điều khiển trong mô hình động học được chúng tôi mô tả bởi véc tơ $u = [v, \omega]^T \in \mathbb{R}^{2 \times 1}$ với v (m/s) là vận tốc tuyến tính của robot, và ω (rad/s) là vận tốc góc của robot. Theo nghiên cứu [23,24], trong trường hợp lý tưởng khi robot không chịu ảnh hưởng của các bất định tham số, mô hình động học của một robot di động ba bánh được thiết lập như sau:

$$\dot{\chi} = f(\chi, u) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (1)$$

Bên cạnh đó, phương trình thể hiện ràng buộc của robot liên quan tới điều kiện không bị trượt ngang được thiết lập như sau:

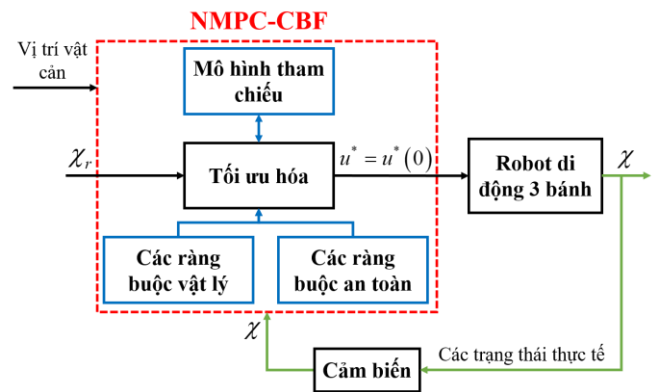
$$\dot{x} \sin(\theta) - \dot{y} \cos(\theta) = 0 \quad (2)$$

Nhận xét 1: Mô hình động học (1) với các điều kiện ràng buộc (2), đã mô tả mối quan hệ giữa đầu vào điều khiển $u = [v, \omega]^T$ và trạng thái $\chi = [x, y, \theta]^T$ của robot di động ba bánh, đồng thời phản ánh rõ bản chất phi tuyến của hệ thống. Đối với các mục tiêu điều khiển ở mức cao như lập kế hoạch chuyển động thông minh, mô hình động học là đủ để triển khai. Trong khi đó, mô hình động lực học thường chỉ cần thiết ở tầng điều khiển cấp thấp, nơi nó giúp điều khiển robot bám theo sát quỹ đạo đã được thiết lập từ bộ điều khiển cấp cao.

Nhận xét 2: Mô hình động học (1) đơn giản và dễ xác định trong thực tế hơn so với mô hình động lực học, bởi các biến như vận tốc tuyến tính, vận tốc góc, và góc định hướng đều có thể đo trực tiếp hoặc gián tiếp, bằng các cảm biến được sử dụng rộng rãi trong công nghiệp như bộ mã hóa Encoder có độ phân giải cao và IMU độ trôi thấp. Điều này giúp mô hình động học phù hợp cho các ứng dụng điều khiển sử dụng NMPC.

3. Chiến lược lập kế hoạch đường đi thông minh đề xuất

Hình 2 minh họa cấu trúc tổng thể của chiến lược lập kế hoạch đường đi thông minh NMPC-CBF cho robot di động ba bánh. Trong cấu trúc này, mô hình tham chiếu đóng vai trò là khối dự đoán, sử dụng mô hình động học của robot để dự đoán trạng thái tương lai χ trong một khoảng thời gian hữu hạn dựa trên chuỗi điều khiển ứng viên. Trong đó, chuỗi điều khiển ứng viên là tập hợp các tín hiệu điều khiển khả thi được tạo ra để mô phỏng trước hành vi của robot trong tương lai. Từ đó, bộ tối ưu hóa sử dụng thông tin này để tìm ra tín hiệu điều khiển tối ưu $u^* = u^*(0)$, là phần tử đầu tiên trong chuỗi tín hiệu điều khiển tối ưu được tạo ra sau khi giải bài toán tối ưu hóa, sao cho robot bám sát vị trí mục tiêu χ_r , đồng thời thỏa mãn các ràng buộc vật lý và ràng buộc an toàn được biểu diễn bằng CBF. Đây là tín hiệu duy nhất được áp dụng ngay cho robot ở mỗi chu kỳ lấy mẫu điều khiển, còn các tín hiệu còn lại sẽ bị bỏ qua và tính toán lại ở bước thời gian tiếp theo để thích ứng với các trạng thái mới thực tế. Tín hiệu $u^*(0)$ sau khi được áp dụng cho robot di động ba bánh sẽ tạo ra phản hồi, trạng thái thực tế của robot được cập nhật và gửi lại về bộ điều khiển NMPC-CBF thông qua cảm biến để thực hiện chu trình tiếp theo.



Hình 2: Sơ đồ cấu trúc của chiến lược NMPC-CBF

3.1 Điều khiển dự báo mô hình phi tuyến

Thông thường các trạng thái tiếp theo của robot được tính toán sử dụng phương pháp rời rạc hóa Euler áp dụng cho mô hình động học tại phương trình (1) như sau:

$$\begin{bmatrix} x(k+1) \\ y(k+1) \\ \theta(k+1) \end{bmatrix} = \begin{bmatrix} x(k) \\ y(k) \\ \theta(k) \end{bmatrix} + \Delta_t \begin{bmatrix} v(k) \cos(\theta(k)) \\ v(k) \sin(\theta(k)) \\ \omega(k) \end{bmatrix} \quad (3)$$

Trong đó: $x(k)$, $y(k)$ và $\theta(k)$ lần lượt là tọa độ vị trí hiện tại của robot và góc định hướng của robot tại thời điểm k . $v(k)$ và $\omega(k)$ biểu thị vận tốc tuyến tính và vận tốc góc của robot tại thời điểm k . Δ_t là bước thời gian rời rạc tức khoảng thời gian giữa hai lần lấy mẫu liên tiếp. Cuối cùng, $x(k+1)$, $y(k+1)$, và $\theta(k+1)$ là các trạng thái dự đoán tiếp theo của robot tại thời điểm $k+1$.

Phương trình (3) có thể được biểu diễn tương đương như sau:

$$\chi_{k+1} = \chi_k + \Delta_T f(\chi_k, u_k) \quad (4)$$

Nhận xét 3: Theo nghiên cứu [17], tuy phương pháp Euler tại phương trình (3) mang lại ưu điểm là đơn giản và rất dễ triển khai, nhưng lại chỉ sử dụng một độ dốc duy nhất, nên phương pháp này không cung cấp nghiệm xấp xỉ tốt cho các phương trình vi phân phi tuyến khi thời gian lấy mẫu Δ_T lớn. Chính vì vậy trong nghiên cứu này, thay vì sử dụng phương pháp Euler chúng tôi sử dụng RK4. Phương pháp RK4 sử dụng bốn ước lượng độ dốc tại các điểm khác nhau trong khoảng lấy mẫu để đưa ra dự đoán chính xác hơn về trạng thái tiếp theo của hệ thống, đặc biệt phù hợp với các hệ phi tuyến như 3WMRs.

Theo nghiên cứu [19], trạng thái χ_{k+1} của robot được tính toán theo RK4 như sau:

$$\chi_{k+1} = \chi_k + \frac{\Delta_T}{6} (R_1 + 2R_2 + 2R_3 + R_4) \quad (5)$$

Trong đó: $R_i (i=1, \dots, 4)$ là ước lượng độ dốc tại các điểm khác nhau trong một bước lấy mẫu, dùng để tính trạng thái tiếp theo của hệ phi tuyến.

Cụ thể, theo [19] các giá trị ước lượng độ dốc trong phương trình (5) được xác định như sau:

$$\begin{cases} R_1 = f(\chi_k, u_k) \\ R_2 = f\left(\chi_k + \frac{\Delta_T}{2} R_1, u_k\right) \\ R_3 = f\left(\chi_k + \frac{\Delta_T}{2} R_2, u_k\right) \\ R_4 = f(\chi_k + \Delta_T R_3, u_k) \end{cases} \quad (6)$$

Phương trình (5) có thể được viết thành:

$$\chi_{k+1} = f(\chi_k, u_k) \quad (7)$$

Sau khi xây dựng mô hình rời rạc hóa bằng phương pháp RK4, chúng tôi tiến hành thiết lập bài toán NMPC nhằm tìm ra chuỗi điều khiển tối ưu cho robot di động ba bánh trong một khoảng thời gian dự đoán hữu hạn N . Cụ thể, với mô hình rời rạc thu được từ RK4 tại phương trình (5), chúng tôi xây dựng bài toán điều khiển tối ưu như sau:

$$\min_u (J_N(\chi_0, u)) \quad (8)$$

Các ràng buộc đẳng thức và bất đẳng thức ban đầu được xây dựng như sau:

$$\begin{cases} \chi_k(0) = \chi_0 \\ \chi_{k+1} = f(\chi_k, u_k); \forall k = [0, N-1] \\ \chi_{\min} \leq \chi_k \leq \chi_{\max}; \forall k = [0, N] \\ u_{\min} \leq u_k \leq u_{\max}; \forall k = [0, N-1] \end{cases} \quad (9)$$

Trong đó: $\chi_{\min}$ là giới hạn dưới của trạng thái, $\chi_{\max}$ là giới hạn trên của trạng thái, $u_{\min}$ là giới hạn dưới của tín hiệu điều khiển, $u_{\max}$ là giới hạn trên của tín hiệu điều khiển,

Cụ thể, trong phương trình (8):

$$J_N(\chi_0, u) = \sum_{k=0}^{N-1} (\|\chi_k - \chi_r\|_Q^2 + \|u_k - u_r\|_K^2 + \|\Delta u_k\|_S^2) + \|\chi_N - \chi_r\|_P^2 \quad (10)$$

Trong đó: χ_k là trạng thái của robot tại thời điểm k , u_k là tín hiệu điều khiển của robot tại thời điểm k , χ_r là trạng

thái tham chiếu, u_r là tín hiệu điều khiển tham chiếu, và $\Delta u_k = u_k - u_{k-1}$ đại diện cho độ biến thiên tín hiệu điều khiển.

Các giá trị $Q \in \mathbf{R}^{3 \times 3}$, $K \in \mathbf{R}^{2 \times 2}$, $S \in \mathbf{R}^{2 \times 2}$, $P \in \mathbf{R}^{3 \times 3}$ là các ma trận trọng số phản ánh mức độ ưu tiên giữa bám điểm tham chiếu để giảm sai số trạng thái, giảm sai lệch điều khiển để tiết kiệm năng lượng, và làm mịn tín hiệu điều khiển để làm mượt chuyển động của robot.

Ngoài ra, các thành phần cụ thể trong phương trình (10) được tính như sau:

$$\begin{cases} \|\chi_k - \chi_r\|_Q^2 = (\chi_k - \chi_r)^T Q (\chi_k - \chi_r) \\ \|u_k - u_r\|_K^2 = (u_k - u_r)^T K (u_k - u_r) \\ \|\Delta u_k\|_S^2 = (\Delta u_k)^T S (\Delta u_k) \\ \|\chi_N - \chi_r\|_P^2 = (\chi_N - \chi_r)^T P (\chi_N - \chi_r) \end{cases} \quad (11)$$

Nhận xét 4: Bài toán điều khiển tối ưu trong phương trình (8), và các ràng buộc đẳng thức, bất đẳng thức trong phương trình (9) chưa bao gồm khả năng tránh vật cản cho robot. Nội dung bên dưới trình bày thuật toán đảm bảo an toàn cho robot sử dụng CBF.

3.2 Thuật toán đảm bảo an toàn cho robot

Các nghiên cứu [17,18] đã thiết lập bài toán tránh va chạm cho robot, được thực hiện thông qua việc bổ sung ràng buộc khoảng cách vào phương trình (9) dựa trên mối quan hệ hình học giữa robot và vật cản tại từng bước tối ưu như sau:

$$\sqrt{(x(k) - x_{obs})^2 + (y(k) - y_{obs})^2} \geq (r_{obs} + r_{rob}) \quad (12)$$

Trong đó: (x_{obs}, y_{obs}) là vị trí của vật cản, r_{obs} là bán kính của vật cản, và r_{rob} là bán kính của robot.

Nhận xét 5: Có thể thấy rằng ràng buộc bất đẳng thức (12) thuộc nhóm các thuật toán BT, vốn đặc biệt phù hợp với NMPC [23] trong các bài toán tránh vật cản. Cụ thể, phương pháp này yêu cầu robot di chuyển tiếp tuyến theo chu vi vật cản và sau đó quay lại lộ trình ban đầu sau khi hoàn tất việc đi vòng qua vật cản. Tuy nhiên, cách tiếp cận này chủ yếu mang tính hình học và không đảm bảo được tính bất biến tiến của tập an toàn trong suốt quá trình điều khiển. Hơn nữa, việc xử lý căn bậc hai trong ràng buộc (12) gây ra khó khăn cho các bộ giải tối ưu hóa. Khác với cách tiếp cận tại [17,18], trong nghiên cứu này, chúng tôi được truyền cảm hứng từ các công trình tại [22,23] để áp dụng CBF cho việc đảm bảo an toàn cho robot trong bài toán tối ưu NMPC. Phương pháp CBF không chỉ đảm bảo tính khả thi của ràng buộc an toàn trong miền liên tục và rời rạc, mà còn duy trì sự ổn định và khả năng mở rộng cho nhiều vật cản trong môi trường động. Do đó, CBF được tích hợp như một ràng buộc bổ sung trong khung NMPC của chúng tôi thiết kế, nhằm đảm bảo tính an toàn toàn cục của hệ thống. Từ đó tạo ra chiến lược lập kế hoạch đường đi thông minh NMPC-CBF cho 3WMRs.

Thuật toán này đảm bảo vị trí $\chi_{x,y} = [x, y]^T \in \mathbf{R}^{2 \times 1}$ trong không gian của robot luôn nằm trong một tập an toàn $C \in \mathbf{R}^{2 \times 1}$ là tập bất biến tiến. Theo các nghiên cứu [22,23], tập C được định nghĩa như sau:

$$\begin{aligned}
C &= \{ \chi_{x,y} \in \mathbf{R}^{2 \times 1} : h(\chi_{x,y}) \geq 0 \} \\
\partial C &= \{ \chi_{x,y} \in \mathbf{R}^{2 \times 1} : h(\chi_{x,y}) = 0 \} \\
Int(C) &= \{ \chi_{x,y} \in \mathbf{R}^{2 \times 1} : h(\chi_{x,y}) > 0 \}
\end{aligned} \quad (13)$$

Trong đó: ∂C là biên của tập an toàn, và $Int(C)$ là tập các vị trí của robot nằm hoàn toàn bên trong vùng an toàn, cách xa biên.

Có thể thấy rằng CBF cho phép xác định một tập hợp các trạng thái an toàn thỏa mãn các ràng buộc khoảng cách:

$$C = \{ \chi_{x,y} \in \mathbf{R}^{2 \times 1} : h(\chi_{x,y}) \geq 0 \} \quad (14)$$

Nếu tồn tại một tín hiệu điều khiển thỏa mãn [22,23]:

$$h(\chi_{x,y}, u) \geq -\gamma h(\chi_{x,y}); \forall \gamma \in (0,1] \quad (15)$$

Tuy nhiên, trong miền không gian rời rạc bất phương trình (15) có thể được viết lại thành:

$$\Delta h(\chi_{x,y}(k), u_k) \geq -\gamma h(\chi_{x,y}(k)); \forall \gamma \in (0,1] \quad (16)$$

Trong đó: $\Delta h(\chi_{x,y}(k), u_k) := h(\chi_{x,y}(k+1)) - h(\chi_{x,y}(k))$, và γ là hằng số để điều chỉnh mức độ mềm của ràng buộc an toàn.

Bất phương trình (16) cũng có thể được viết lại như sau:

$$h(\chi_{x,y}(k+1)) \geq (1-\gamma)h(\chi_{x,y}(k)); \forall \gamma \in (0,1] \quad (17)$$

CBF có thể tích hợp trực tiếp vào bài toán tối ưu trong NMPC, giúp xây dựng ràng buộc an toàn mềm dẻo nhưng vẫn đảm bảo tính chặt chẽ. Trong bài toán tránh vật cản, các vật cản được mô hình hóa là vùng cấm, còn khu vực xung quanh robot là vùng an toàn. Lúc này, ràng buộc đẳng thức tại (17) được bổ sung vào (9) trong bài toán NMPC-CBF của chúng tôi như sau:

$$\begin{cases}
\chi_k(0) = \chi_0 \\
\chi_{k+1} = f(\chi_k, u_k); \forall k = [0, N-1] \\
\chi_{\min} \leq \chi_k \leq \chi_{\max}; \forall k = [0, N] \\
u_{\min} \leq u_k \leq u_{\max}; \forall k = [0, N-1] \\
h(\chi_{x,y}(k+1)) \geq (1-\gamma)h(\chi_{x,y}(k)); \forall \gamma \in (0,1]
\end{cases} \quad (18)$$

Kết hợp cả ý tưởng của nghiên cứu [17,18] tại phương trình (12), hàm rào CBF được chúng tôi thiết kế dựa trên khoảng cách giữa robot và vật cản như sau:

$$h(\chi_{x,y}(k)) = (x(k) - x_{obs})^2 + (y(k) - y_{obs})^2 - (r_{obs} + r_{rob})^2 \quad (19)$$

Trong trường hợp là các vật cản động, vị trí của chúng tại mỗi bước thời gian k sẽ thay đổi và thu được thông qua các hệ thống cảm biến. Khi đó, phương trình (20) được điều chỉnh lại để mang dạng tổng quát cho cả vật cản cố định và di chuyển như sau:

$$\begin{aligned}
h(\chi_{x,y}(k)) \\
= (x(k) - x_{obs}(k))^2 + (y(k) - y_{obs}(k))^2 - (r_{obs} + r_{rob})^2
\end{aligned} \quad (20)$$

Trong đó: $(x_{obs}(k), y_{obs}(k))$ là vị trí của vật cản tại các bước thời gian lấy mẫu k được thu thập thông qua các hệ thống cảm biến.

Có thể thấy, lời giải chung của bài toán điều khiển tối ưu tại phương trình (8) là một chuỗi tín hiệu điều khiển tối ưu có dạng [17]:

$$u^* = (u^*(0), u^*(1), \dots, u^*(N-1)) \quad (21)$$

Trong khuôn khổ của phương pháp NMPC-CBF, chỉ phần tử đầu tiên $u^* = u^*(0)$ được áp dụng để điều khiển robot, trong khi các phần tử còn lại bị loại bỏ và sẽ được tính toán lại tại bước thời gian kế tiếp.

Nhận xét 6 (tổng kết bài toán NMPC-CBF): Tổng kết lại, phương pháp lập kế hoạch đường đi thông minh NMPC-CBF của chúng tôi được như sau: “Sau khi xây dựng mô hình rời rạc thu được tại phương trình (5) sử dụng RK4, chúng tôi xây dựng bài toán điều khiển tối ưu tại (8) với hàm mục tiêu được mô tả trong (10), kết hợp với các ràng buộc vật lý và ràng buộc an toàn dựa trên hàm rào CBF như đã trình bày trong (20). Phương pháp NMPC-CBF tạo ra chuỗi tín hiệu điều khiển tối ưu trong (21), áp dụng phần tử đầu tiên trong chuỗi này để điều khiển robot, cho phép robot di chuyển từ vị trí ban đầu đến mục tiêu, đồng thời đảm bảo duy trì an toàn trước các vật cản, kể cả trong môi trường động.”

4. Các kết quả mô phỏng và phân tích

4.1 Thiết lập mô phỏng

Các chương trình mô phỏng được chúng tôi viết và triển khai trên cùng một máy tính có cấu hình Intel® Core™ i7-11850H (8 nhân, 2.50 GHz), RAM 32 GB, chạy phần mềm MATLAB R2023b, đảm bảo chương trình hoạt động ổn định. Trong nghiên cứu này, chiến lược NMPC-CBF được thực hiện thông qua việc xây dựng và giải bài toán điều khiển tối ưu bằng công cụ mã nguồn mở CasADi [25], được tích hợp dưới dạng thư viện vào môi trường MATLAB R2023b. Tương tự như trong [17], bài toán điều khiển tối ưu (OCP) được chuyển thành một bài toán lập trình phi tuyến (NLP) sử dụng phương pháp Multiple Shooting [26], trong đó cả trạng thái và các biến điều khiển đều được coi là các tham số tối ưu hóa. CasADi sau đó được kết nối với bộ giải IPOPT [27] nhằm tìm ra lời giải tối ưu cho bài toán.

Trong nghiên cứu này, một số giả định kỹ thuật được chúng tôi đặt ra nhằm đơn giản hóa quá trình mô phỏng trong việc đánh giá hiệu quả của NMPC-CBF. Cụ thể, robot được giả định di chuyển trên mặt phẳng hai chiều, không có hiện tượng trượt bánh và không chịu ảnh hưởng bởi nhiễu môi trường hay sai số đo lường. Đây là các giả định phổ biến thường thấy tại các nghiên cứu trong giai đoạn đánh giá hiệu quả của thuật toán điều khiển, đặc biệt khi tập trung vào khả năng lập kế hoạch chuyển động và tránh va chạm. Các vật cản trong mô phỏng được mô hình hóa dưới dạng hình tròn với bán kính không đổi, giúp đơn giản hóa quá trình tính toán và va chạm mà vẫn phản ánh tương đối chính xác các tình huống trong thực tế như người đi bộ, cột trụ hay chướng ngại vật dạng tròn. Vận tốc của các vật cản chuyển động được giả định là không đổi trong từng bước lấy mẫu và không bị ảnh hưởng bởi các yếu tố bên ngoài. Ngoài ra, vị trí mục tiêu của robot được giả định là cố định và đã biết trước, phù hợp với các ứng dụng phổ biến như định vị tới trạm sạc, điểm giao hàng hay khu vực mục tiêu đã định trước trong hệ thống điều hướng tự động.

Các thông số vật lý của 3WMRs bao gồm: $v \in [-0.5, 0.5]$ (m/s), $\omega \in \left[\frac{-\pi}{4}, \frac{\pi}{4} \right]$ (rad/s), và $r_{rob} = R = 0.3$ (m). Ngoài ra, trong tình huống mô phỏng này vị trí của robot bị giới hạn

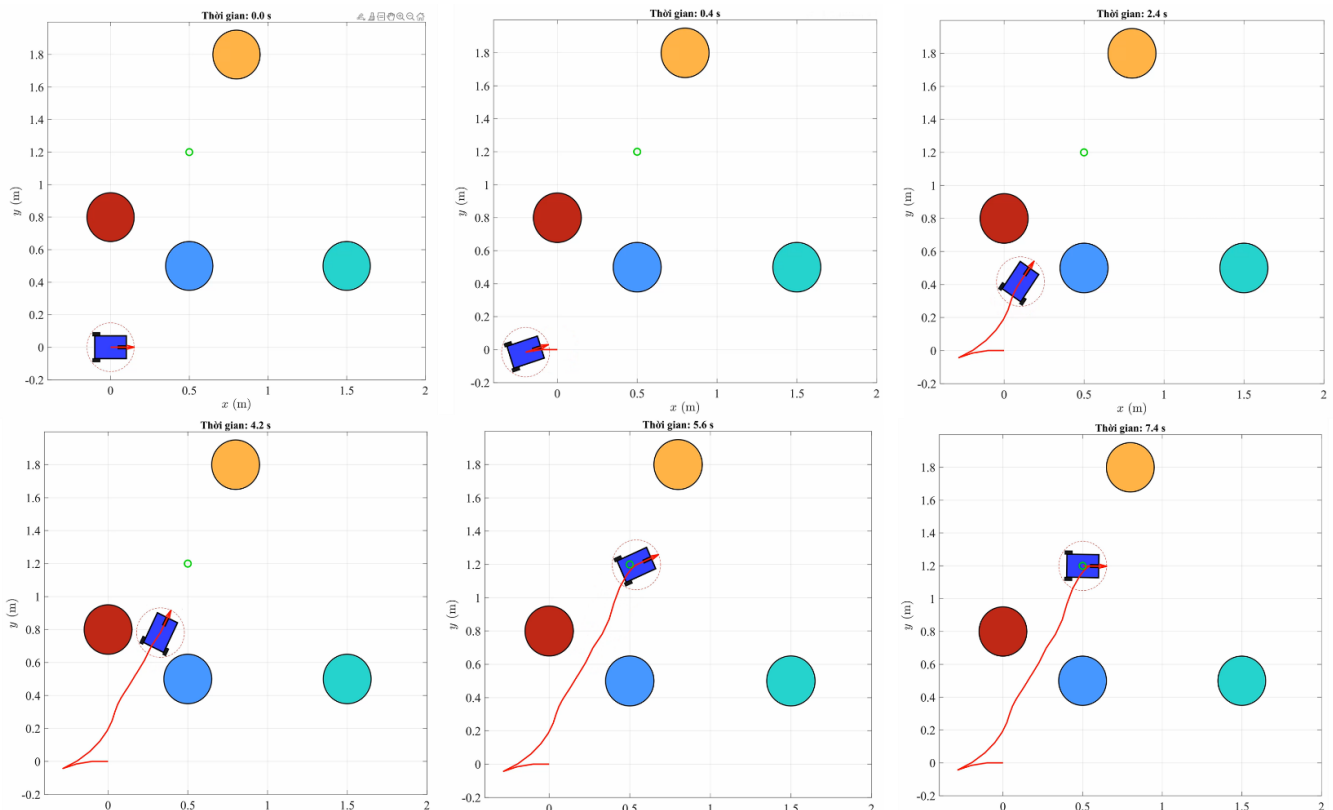
trong phạm vi: $x \in [-2, 2]$ (m) và $y \in [-2, 2]$ (m). Chu kỳ lấy mẫu trong chương trình mô phỏng được thiết lập là $\Delta_r = 0.2$ (s). Robot được thiết lập để xuất phát tại vị trí ban đầu là $\chi_0 = [0, 0, 0]^T$. Các thông số của NMPC-CBF được chúng tôi lựa chọn kỹ lưỡng dựa trên kinh nghiệm và nhiều lần thử sai, được thể hiện trên **Bảng 1**.

Bảng 1: Thông số bộ điều khiển NMPC-CBF

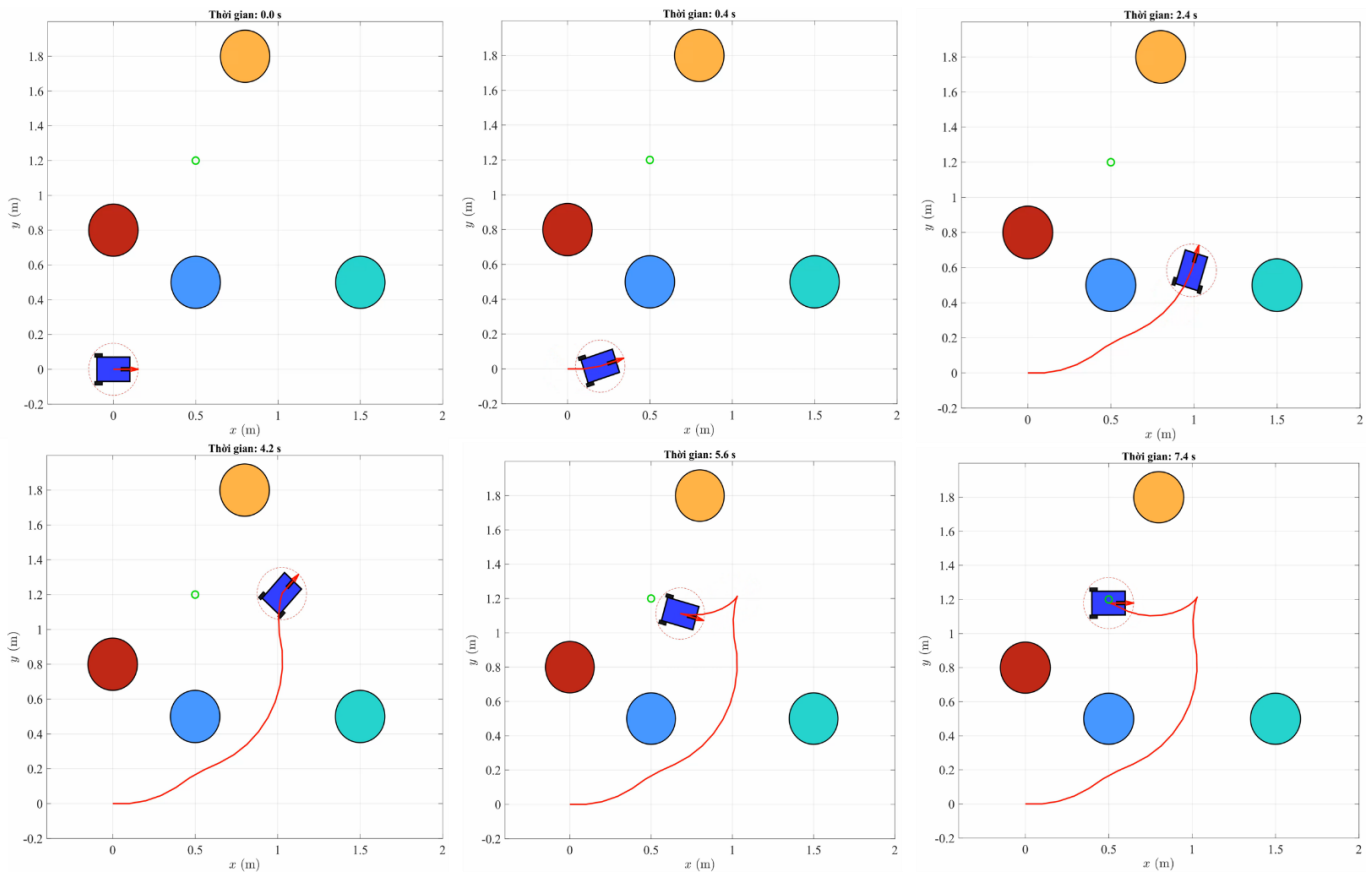
Các tham số	Các giá trị
Q	$\text{diag}([1, 5, 0.1])$
K	$\text{diag}([0.5, 0.001])$
S	$\text{diag}([0.01, 0.001])$
P	$\text{diag}([150, 150, 150])$
γ	0.25
N	10

Trong nội dung mô phỏng này, chúng tôi đã thiết lập hai kịch bản riêng biệt nhằm kiểm tra khả năng vận hành của robot di động sử dụng chiến lược NMPC-CBF trong những tình huống phức tạp. Trong kịch bản thứ nhất, robot hoạt động trong môi trường có sự xuất hiện của các vật cản cố định. Đối với kịch bản thứ hai, robot phải di chuyển an toàn giữa các vật cản chuyển động. Ở cả hai trường hợp, tổng cộng bốn vật cản với bán kính không đổi là $r_{obs} = 0.3$ (m) được đưa vào mô phỏng. Đối với kịch bản có các vật cản chuyển động, các vật thể được thiết lập di chuyển với tốc độ khác nhau: vật cản 1 di chuyển với tốc độ tối đa 0.2 m/s, vật cản 2 là 0.3 m/s, vật

cản 3 là 0.35 m/s, và vật cản 4 là 0.15 m/s. Ngoài ra, để tăng mức độ phức tạp cho bài toán điều khiển, mỗi vật cản được gán một quỹ đạo chuyển động riêng biệt, phản ánh những tình huống thực tế, nơi robot phải đối mặt với các vật thể chuyển động theo những hướng và tốc độ không đồng nhất như sau: vật cản 1 ($x = 0.2t, y = 0.8$) (m,m), vật cản 2 ($x = 1.5, y = 0.5 + 0.3t$) (m,m), vật cản 3 ($x = 0.5, y = 0.5 + 0.35t$) (m,m), vật cản 4 ($x = 0.8, y = 1.8 - 0.15t$) (m,m). Có thể thấy, trong tình huống các vật cản cố định thì vị trí của chúng dễ dàng được xác định như sau: vật cản 1 ($x = 0.0, y = 0.8$), vật cản 2 ($x = 1.5, y = 0.5$), vật cản 3 ($x = 0.5, y = 0.5$), và vật cản 4 ($x = 0.8, y = 1.8$). Vị trí mục tiêu của robot được thiết lập như sau: $\chi_r = [0.5, 1.2, 0]^T$. Bên cạnh đó, chúng tôi cũng so sánh phương pháp NMPC-CBF đề xuất với thuật toán NMPC tại [17] nhằm đánh giá mức độ cải thiện về khả năng lập quỹ đạo và tránh va chạm của robot trong cùng điều kiện thí nghiệm. Đáng chú ý, trong tình huống mô phỏng vật cản cố định, khoảng cách giữa vật cản 1 và vật cản 3 khá hẹp, gần xấp xỉ bán kính vùng an toàn của robot, đây được xem là một trường hợp hiếm gặp trong thực tế. Tuy nhiên, tình huống này được đưa vào mô phỏng nhằm kiểm tra khả năng phản ứng và giới hạn điều khiển của hệ thống lập kế hoạch chuyển động trong điều kiện không gian chật hẹp. Kết quả cho thấy NMPC-CBF vẫn có thể điều hướng qua khe hẹp đó, chấp nhận xâm nhập nhẹ vào vùng an toàn, thể hiện khả năng cân bằng hợp lý giữa tính an toàn và hiệu quả quỹ đạo trong môi trường phức tạp. Các kết quả thu được và phân tích chi tiết hơn được thể hiện ở nội dung bên dưới.



Hình 3: NMPC-CBF trong trường hợp vật cản cố định

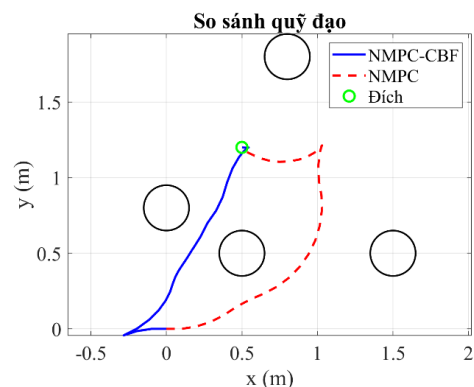


Hình 4: NMPC trong trường hợp vật cản cố định

4.2 Các kết quả thu được và phân tích

4.2.1 Kích bản các vật cản cố định

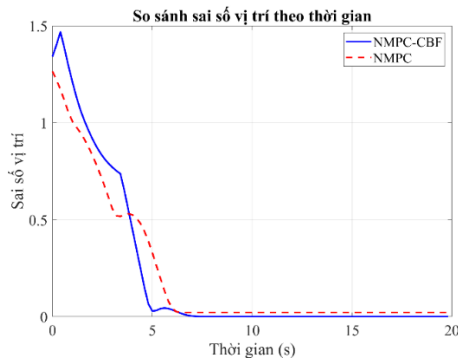
Trong kịch bản mô phỏng với môi trường có nhiều vật cản cố định, hai chiến lược điều khiển NMPC-CBF và NMPC được so sánh nhằm đánh giá khả năng điều hướng an toàn và hiệu quả của robot di động. Các kết quả thu được của NMPC-CBF và NMPC được thể hiện rõ ràng trên Hình 3 và Hình 4 tương ứng, cho thấy NMPC-CBF mang lại quỹ đạo di chuyển mượt mà và tối ưu hơn, nhờ vào việc tích hợp các ràng buộc an toàn dựa trên CBF, giúp robot duy trì khoảng cách an toàn với các vật cản trong suốt quá trình vận hành. Mặc dù quỹ đạo của NMPC-CBF có xâm nhập nhẹ vào vùng an toàn, song mức độ xâm nhập này không đáng kể và không gây ra va chạm, phản ánh sự cân bằng hợp lý giữa yêu cầu an toàn và tối ưu hóa quỹ đạo chuyển động. Ngược lại, NMPC thuần túy tạo ra quỹ đạo điều khiển có xu hướng đi sát mép vật cản, dễ tiềm ẩn nguy cơ va chạm và thể hiện các chuyển động gấp khúc do phải điều chỉnh hướng đột ngột khi gặp chướng ngại vật. Cụ thể, tổng quãng đường di chuyển của robot sử dụng NMPC-CBF là xấp xỉ 1.86 (m) trong khi đó quãng đường di chuyển của robot khi sử dụng NMPC là xấp xỉ 2.42 (m), so sánh quỹ đạo được tạo ra bởi 2 phương pháp được thể hiện rõ trên Hình 5 khẳng định rõ sự tối ưu hơn của NMPC-CBF. Trong đó, phương pháp NMPC-CBF được thể hiện bằng đường nét liền màu xanh dương, còn đối với NMPC được thể hiện bằng đường nét đứt màu đỏ. Vị trí của điểm đích mà robot cần tạo đường đi đến được thể hiện bằng hình tròn nhỏ màu xanh lá cây.



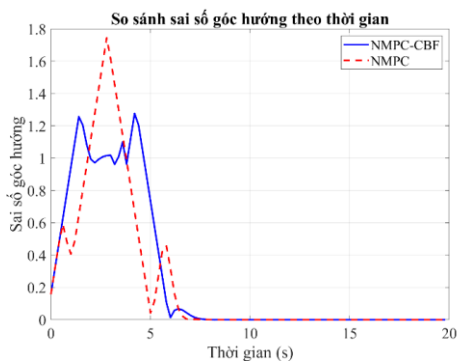
Hình 5: So sánh quỹ đạo được tạo bởi NMPC-CBF và NMPC

Ở thời điểm 0.0 s, cả hai phương pháp đều bắt đầu từ vị trí ban đầu tương đương và cùng lập kế hoạch di chuyển tới một vị trí đích. Tại 0.4 s, quỹ đạo của NMPC-CBF tạo ra giúp robot lùi nhẹ lại để tạo đà và thiết lập hướng di chuyển. Trong khi đó, NMPC vẫn giữ hướng đi khá trực diện và tiếp tục tiến sát vật cản phía trước. Đến 2.4 s, NMPC-CBF thể hiện khả năng tối ưu rõ rệt khi robot đã tiến tới và chuẩn bị lách qua khe hẹp giữa hai vật cản trung tâm với khoảng cách đủ an toàn và tuyến đường ngắn, hướng thẳng về mục tiêu, trong khi đó NMPC buộc phải rẽ góc gấp để điều chỉnh hướng đi khi robot tiến quá gần vật cản, tạo nên một đoạn quỹ đạo gấp khúc và kéo dài quãng đường di chuyển. Điều này tiếp tục được minh chứng tại các thời điểm 4.2 s và 5.6 s, nơi quỹ đạo NMPC-CBF đã tiếp cận gần mục tiêu hơn nhiều, với hướng di chuyển ổn định, trong khi NMPC vẫn phải điều chỉnh để tránh các vật thể phía trên màu cam, khiến robot bị lệch ra khỏi hướng đi tối ưu. Lúc này, robot với NMPC phải lùi lại để đạt được vị

trí mục tiêu như thể hiện tại 5.6 s. Cuối cùng, ở thời điểm 7.4 s, NMPC-CBF đã đưa robot tới gần mục tiêu hơn với khoảng cách ngắn và vị trí thuận lợi để chuẩn bị cho quá trình ổn định với sai số vị trí và góc hướng tiệm cận 0, rồi dừng lại hoàn toàn tại thời điểm xấp xỉ 7.6 s như thể hiện trên **Hình 6** đến **Hình 9**. Quỹ đạo của NMPC-CBF ngắn hơn và ít thay đổi đột ngột hơn so với NMPC. Có thể thấy, chiến lược NMPC-CBF đã giúp tránh sớm các vật cản cố định bằng cách quy hoạch quỹ đạo tối ưu ngay từ đầu thay vì chờ đến gần mới điều chỉnh như phương pháp NMPC. Đáng chú ý, phương pháp NMPC-CBF chấp nhận một mức xâm nhập nhẹ vào vùng an toàn có kiểm soát, nhằm duy trì quỹ đạo ngắn hơn, và đảm bảo khả năng di chuyển mượt mà, không làm mất đi tính an toàn tổng thể của hệ thống.



Hình 6: So sánh sai số vị trí theo thời gian



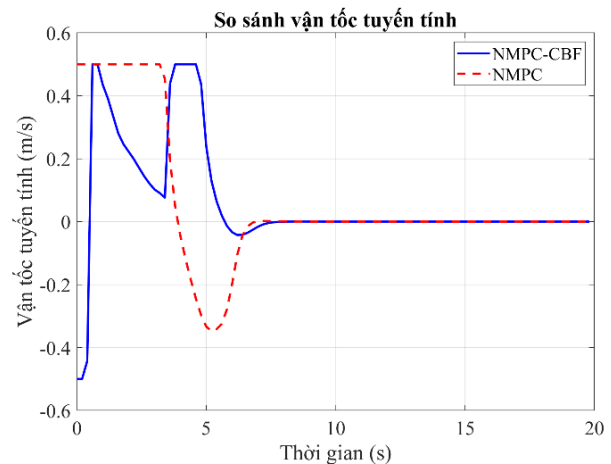
Hình 7: So sánh sai số góc hướng theo thời gian

Chi tiết về sai số vị trí (tính theo đơn vị mét) và sai số góc hướng (tính theo đơn vị radian) lần lượt được thể hiện trong **Hình 6** và **Hình 7**. Cụ thể, sai số vị trí được xác định thông qua chuẩn Norm-2 của độ lệch giữa vị trí thực tế của robot so với điểm mục tiêu. Qua các kết quả mô phỏng, có thể nhận thấy rằng phương pháp NMPC-CBF không chỉ thể hiện ưu thế rõ rệt trong việc tránh vật cản mà còn mang lại hiệu quả vượt trội về độ chính xác trong quá trình hội tụ. Mặc dù, NMPC tạo ra quỹ đạo di chuyển cho robot đạt được sai số góc hướng tiệm cận về 0 khi hệ thống ổn định, thì vẫn tồn tại sai số vị trí còn lại xấp xỉ 0.02 m. Ngược lại, NMPC-CBF cho thấy khả năng điều hướng tối ưu hơn, đưa robot tiến sát mục tiêu với sai số cả về vị trí lẫn góc hướng đều nhỏ hơn, tiệm cận giá trị 0. Ngoài ra, biểu đồ **Hình 8** và **Hình 9** cho thấy sự khác biệt rõ ràng giữa hai phương pháp NMPC-CBF và NMPC thông qua các tín hiệu điều khiển đầu ra, bao gồm vận tốc tuyến tính và vận tốc góc - thường được sử dụng làm tham chiếu cho bộ điều khiển cấp thấp của robot. Việc bám theo các tín hiệu tham chiếu này bằng các giải thuật điều khiển bám giúp robot thực hiện quỹ đạo được tối ưu hóa bởi mỗi phương pháp. Căn cứ

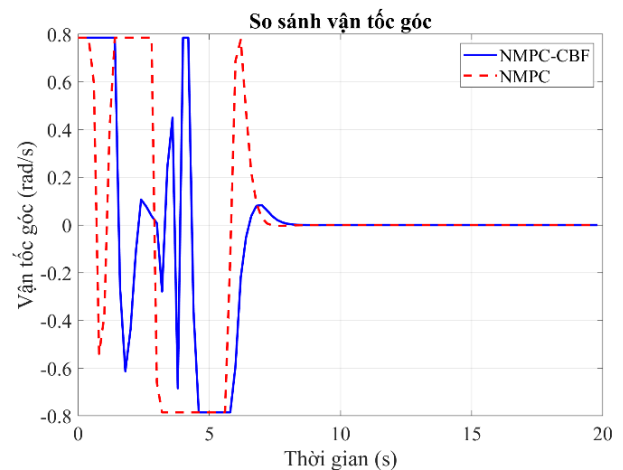
vào các kết quả vận tốc tuyến tính và vận tốc góc được hiển thị trực quan trên hình, có thể thấy rằng cả hai phương pháp NMPC-CBF và NMPC đều tạo ra các quỹ đạo tới điểm đích đồng thời giúp đảm bảo các ràng buộc vật lý đối với vận tốc tuyến tính và vận tốc góc của robot khi thỏa mãn $v \in [-0.5, 0.5]$

(m/s) và $\omega \in \left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$ (rad/s) như đã trình bày ở phần thiết lập

mô phỏng. Bên cạnh đó, khi robot tiến sát đến vị trí mục tiêu, cả hai phương pháp NMPC-CBF và NMPC đều cho thấy khả năng điều chỉnh vận tốc tuyến tính và vận tốc góc hội tụ về giá trị 0. Điều này là cần thiết nhằm đảm bảo robot dừng lại tiệm cận so với vị trí đích mà không phát sinh các chuyển động dư thừa, từ đó tăng tính ổn định và an toàn cho hệ thống, đặc biệt trong các ứng dụng yêu cầu độ chính xác cao.



Hình 8: So sánh vận tốc tuyến tính theo thời gian



Hình 9: So sánh vận tốc góc theo thời gian

Để thuận tiện cho việc theo dõi và phân tích trực quan về hiệu suất giữa NMPC-CBF và NMPC, Bảng 2 trình bày các thông số đánh giá định lượng quan trọng trong kịch bản các vật cản cố định, bao gồm thời gian tính toán, tổng quãng đường di chuyển, và sai số trạng thái cuối tại vị trí và góc hướng. Cần lưu ý rằng thời gian tính toán được thống kê là kết quả đo bằng cặp lệnh tic/toc trong môi trường MATLAB, phản ánh tổng thời gian CPU xử lý toàn bộ quá trình mô phỏng, tức là khoảng thời gian tích lũy từ khi bắt đầu cho đến khi robot hoàn tất hành trình đến mục tiêu. So sánh các chỉ số trong bảng cho thấy phương pháp NMPC-CBF thể hiện ưu thế

rõ rệt. Mặc dù có thời gian tính toán dài hơn khoảng 5.74 giây so với NMPC thuần, NMPC-CBF lại giúp robot di chuyển theo quỹ đạo ngắn hơn đáng kể (giảm khoảng 23%), đồng thời đạt được trạng thái cuối với sai số vị trí tiệm cận bằng 0 và sai số góc hướng bằng 0. Trong khi đó, NMPC thuần dù vẫn đảm bảo được sự ổn định về góc hướng, nhưng vẫn tồn tại sai số vị trí cuối xấp xỉ 0.02 m, điều này có thể ảnh hưởng đến các tác vụ yêu cầu độ chính xác cao, đặc biệt trong môi trường hẹp hoặc khi cần phối hợp nhiều tác nhân. Việc tăng thời gian tính toán của NMPC-CBF là hoàn toàn hợp lý, do sự bổ sung các ràng buộc an toàn dựa trên CBF vào bài toán tối ưu, dẫn đến bài toán trở nên phức tạp hơn. Tuy nhiên, phần chi phí tính toán tăng thêm này vẫn nằm trong giới hạn chấp nhận được, đặc biệt trong bối cảnh các hệ thống nhúng hiện đại ngày càng mạnh mẽ và có khả năng xử lý thời gian thực tốt hơn. Điều này khẳng định rằng NMPC-CBF không chỉ đảm bảo an toàn mà còn góp phần nâng cao hiệu quả và độ chính xác của hành vi điều hướng.

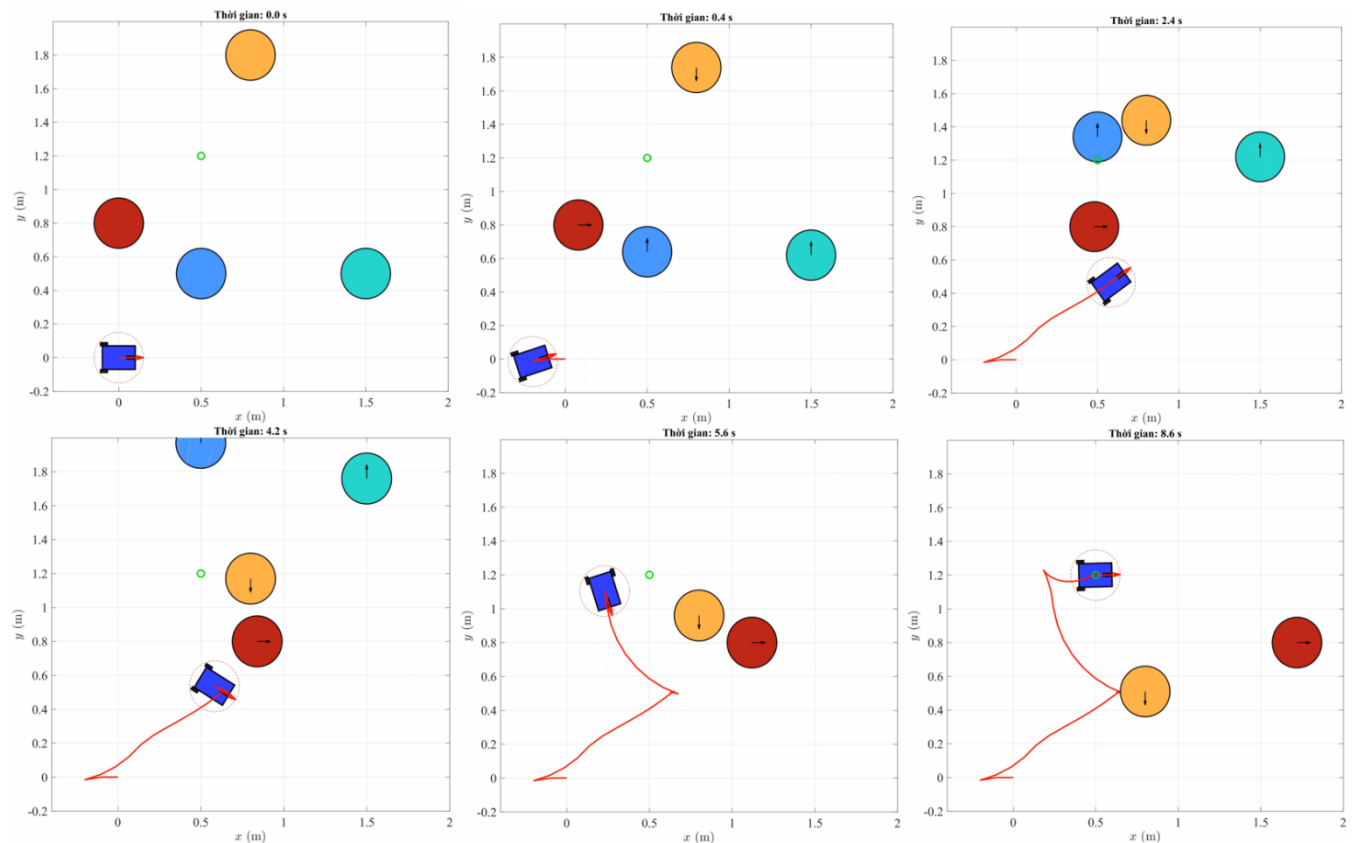
Bảng 2: So sánh hiệu suất giữa NMPC-CBF và NMPC

Các thông số	NMPC-CBF	NMPC
Thời gian tính toán	16.82 (s)	11.08 (s)
Tổng quãng đường	1.86 (m)	2.42 (m)
Sai số vị trí	0 (m)	0.02 (m)
Sai số góc hướng	0 (rad)	0 (rad)

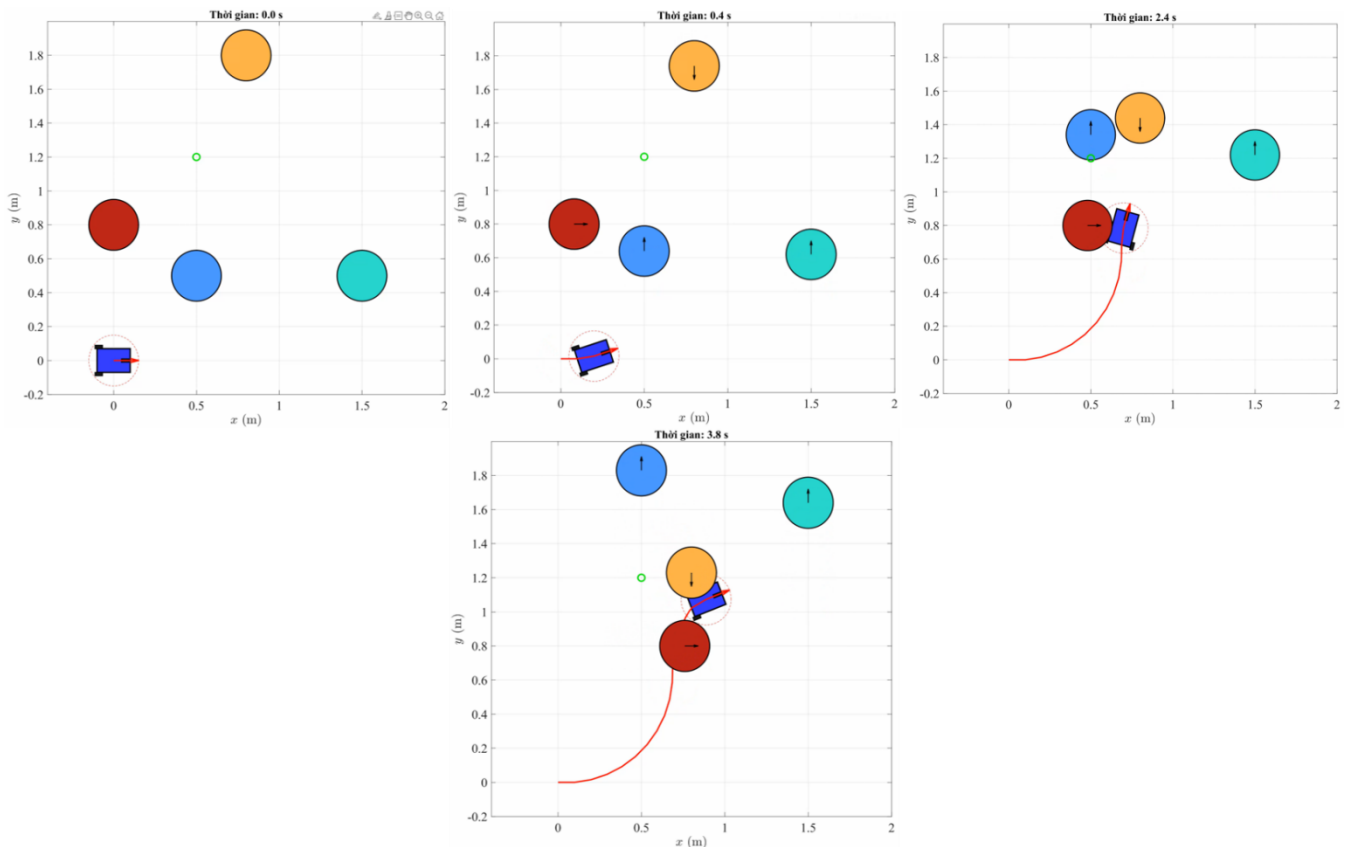
4.2.2 Kịch bản các vật cản di chuyển

Trong kịch bản mô phỏng với các vật cản chuyển động thể hiện trên **Hình 10** và **Hình 11**, lần lượt cho thấy sự khác biệt

giữa hai phương pháp NMPC-CBF và NMPC được thể hiện rõ ràng qua khả năng điều hướng an toàn và hiệu quả của robot. Ở phương pháp NMPC-CBF (**Hình 10**), robot chủ động lựa chọn quỹ đạo mềm mại, ngắn gọn và an toàn hơn, khéo léo lách qua các vật cản đang di chuyển mà vẫn duy trì khoảng cách an toàn tối thiểu, đồng thời nhanh chóng tiếp cận mục tiêu. Tại các thời điểm 2.4 s, 4.2 s, và 5.6 s, robot đã có những chuyển hướng hợp lý để tránh va chạm, chứng minh khả năng thích ứng nhạy bén với sự thay đổi liên tục của môi trường. Đáng chú ý, quỹ đạo chuyển động được thể hiện trong **Hình 10** có độ cong được điều chỉnh hợp lý khi đối mặt với các tình huống nguy hiểm, minh chứng cho quá trình lập kế hoạch chuyển động thông minh, nơi các ràng buộc an toàn từ CBF được tích hợp hiệu quả. Không chỉ tránh được va chạm, robot còn duy trì tính mượt mà trong chuyển động, phản ánh chất lượng điều khiển toàn diện cả về mặt quỹ đạo lẫn hành vi vận động. Ở thời điểm 8.6 s, robot đã tiếp cận thành công vị trí mục tiêu, hoàn tất quá trình dẫn đường một cách hiệu quả và an toàn. Ngược lại, trong phương pháp NMPC (**Hình 11**), robot dường như không nhận thức được mức độ nguy hiểm của các vật cản đang di chuyển, dẫn đến tình huống tiến gần quá mức đến chướng ngại vật. Cụ thể tại thời điểm 2.4 s và 3.8 s, robot đã va chạm với các vật cản màu đỏ và màu cam. Quỹ đạo được tạo ra từ NMPC cho thấy hệ thống lập kế hoạch chuyển động không đủ khả năng điều chỉnh hợp lý trước các tình huống phức tạp, nhất là khi các vật cản thay đổi vị trí liên tục. Những kết quả thu được này đã nhấn mạnh việc tích hợp ràng buộc CBF trong NMPC không chỉ giúp đảm bảo an toàn và tránh va chạm mà còn cải thiện đáng kể hiệu suất dẫn đường và độ tin cậy của hệ thống trong các môi trường động phức tạp. Trong trường hợp này, việc phân tích thêm các thông số như sai số vị trí hoặc tín hiệu điều khiển là không cần thiết, bởi vì



Hình 10: NMPC-CBF trong trường hợp các vật cản di chuyển



Hình 11: NMPC trong trường hợp các vật cản di chuyển

hành vi và chạm trực diện đã trực tiếp chứng minh sự thất bại của NMPC và làm nổi bật ưu điểm vượt trội của NMPC-CBF.

Ngoài ra, các kết quả định lượng được trình bày trong Bảng 3 cho thấy NMPC-CBF không chỉ giúp robot đảm bảo an toàn mà còn đạt hiệu suất dẫn đường cao, với sai số vị trí và góc hướng cuối bằng 0. Tổng quãng đường chỉ 2.55 m, phản ánh quá trình điều hướng ngắn gọn và hiệu quả. Thời gian tính toán đạt 17.88 s, được đo bằng hàm tic/toc, phản ánh toàn bộ thời gian CPU xử lý quá trình mô phỏng. So với kịch bản vật cản tĩnh với thời gian tính toán là 16.82 s, thời gian tính toán trong kịch bản này tăng lên do hệ thống phải liên tục cập nhật lại ràng buộc CBF theo vị trí tức thời của các vật cản di chuyển, đồng thời giải bài toán tối ưu với điều kiện thay đổi liên tục theo thời gian. Tuy nhiên, mức tăng này vẫn nằm trong giới hạn cho phép của nhiều hệ thống nhúng hiện đại, đặc biệt khi thuật toán có thể được tối ưu hóa và thực thi trên các nền tảng phần cứng chuyên dụng mạnh mẽ.

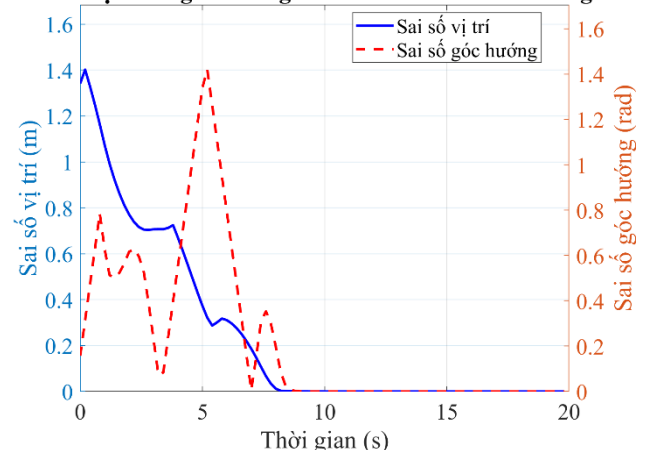
Bảng 3: Hiệu suất của NMPC-CBF

Các thông số	NMPC-CBF
Thời gian tính toán	17.88 (s)
Tổng quãng đường	2.55 (m)
Sai số vị trí	0 (m)
Sai số góc hướng	0 (rad)

Cùng với đó, **Hình 12** cũng cung cấp trực quan hơn khả năng hội tụ nhanh và chính xác của robot về trạng thái mục tiêu khi áp dụng giải thuật lập kế hoạch chuyển động thông minh NMPC-CBF. Cụ thể, quỹ đạo được tính toán bởi NMPC-CBF giúp sai số vị trí và góc hướng của robot giảm về tiệm cận giá trị 0 trong vòng xấp xỉ 8.6 s, cho thấy NMPC-

CBF tạo ra quỹ đạo tối ưu không chỉ đạt độ chính xác cao về vị trí mà còn duy trì đúng hướng. Kết quả này là cơ sở vững chắc khẳng định khả năng lập kế hoạch chuyển động tối ưu và đảm bảo an toàn của thuật toán NMPC-CBF mà chúng tôi đề xuất. Tổng kết lại, từ góc độ thiết kế hệ thống lập kế hoạch chuyển động, sự kết hợp giữa NMPC và CBF có thể xem là một hướng tiếp cận hiệu quả để dung hòa giữa nhiều mục tiêu bao gồm tối ưu hóa hiệu suất dẫn đường, đảm bảo an toàn vận hành, và phù hợp để triển khai thực tế.

Sai số vị trí và góc hướng của NMPC-CBF theo thời gian



Hình 12: Sai số vị trí và góc hướng của NMPC-CBF theo thời gian

5. Các kết luận và hướng phát triển trong tương lai

Phương pháp NMPC-CBF được đề xuất trong nghiên cứu này đã chứng minh khả năng lập kế hoạch chuyển động hiệu quả và an toàn cho robot di động ba bánh trong các môi trường phức tạp có nhiều vật cản cố định và chuyển động. Bằng cách kết hợp hài hòa giữa CBF nhằm đảm bảo các ràng buộc an toàn và NMPC để tối ưu hóa hàm mục tiêu, chiến lược điều khiển này cho thấy hiệu suất vượt trội so với phương pháp NMPC truyền thống. Kết quả mô phỏng cho thấy robot có thể di chuyển với quỹ đạo mượt mà, ổn định, sai số nhỏ và phản ứng nhanh, đặc biệt là khả năng tránh vật cản hiệu quả ngay cả trong những tình huống khó xử lý. Trong nghiên cứu này, các mô phỏng đã được thiết lập trong trường hợp tồn tại khe hẹp giữa hai vật cản nhằm đánh giá giới hạn điều khiển, nhưng các nghiên cứu tiếp theo chúng tôi sẽ tập trung vào các kịch bản thực tế hơn, phản ánh chính xác hơn môi trường vận hành của robot. Trong tương lai, chiến lược NMPC-CBF có tiềm năng được mở rộng để áp dụng cho các hệ thống robot phức tạp hơn, cũng như triển khai trực tiếp trên nền tảng phần cứng thực để đánh giá tính khả thi và độ tin cậy.

Lời cảm ơn

Các tác giả trân trọng cảm ơn sự hỗ trợ từ Trường Đại học Công nghiệp Hà Nội, Trường Đại học Mỏ địa chất, Đại học Quốc gia Hà Nội, Đại học Phenikaa, và HTI Group trong quá trình thực hiện nghiên cứu. Chúng tôi cũng cảm ơn Ban biên tập và các phản biện đã đưa ra những nhận xét hữu ích giúp cải thiện chất lượng bài báo.

Tài liệu tham khảo

- [1] Dang ST, Dinh XM, Nguyen Van T, Pham TN, Nguyen NL, Le HX (2025) *Enhanced Trajectory Tracking for Car-Like Mobile Robots*. In: International Conference on Applied Mathematics and Computer Science. Springer, Cham, pp 421–432
- [2] Zhou C, Huang B, Fränti P (2022) *A review of motion planning algorithms for intelligent robots*. J Intell Manuf 33(2):387–424
- [3] Sánchez-Ibáñez JR, Pérez-del-Pulgar CJ, García-Cerezo A (2021) *Path planning for autonomous mobile robots: A review*. Sensors 21(23):7898
- [4] Fu X, Huang Z, Zhang G, Wang W, Wang J (2025) *Research on path planning of mobile robots based on improved A* algorithm*. PeerJ Comput Sci 11:e2691
- [5] Luo M, Hou X, Yang J (2020) *Surface optimal path planning using an extended Dijkstra algorithm*. IEEE Access 8:147827–147838
- [6] Li X, Tong Y (2023) *Path planning of a mobile robot based on the improved RRT algorithm*. Appl Sci 14(1):25
- [7] Li J, Sun J, Liu L, Xu J (2021) *Model predictive control for the tracking of autonomous mobile robot combined with a local path planning*. Meas Control 54(9–10):1319–1325
- [8] Zhang H, Zhu Y, Liu X, Xu X (2021) *Analysis of obstacle avoidance strategy for dual-arm robot based on speed field with improved artificial potential field algorithm*. Electronics 10(15):1850
- [9] Puriyanto RD, Mustofa AK (2024) *Design and implementation of fuzzy logic for obstacle avoidance in differential drive mobile robot*. J Robot Control (JRC) 5(1):132–141
- [10] Dinh XM, Luong TG, Le XH, Nguyen VT, Kim DT (2025) *Adaptive Fuzzy Dynamic Surface Control with Sliding Mode Control for Enhanced Trajectory Tracking of Delta Robots*. J Control Autom Electr Syst:1–25
- [11] Manh TN, Manh CN, Tien DP, Van MT, Kim DHT, Duc DN (2020) *Autonomous navigation for omnidirectional robot based on deep reinforcement learning*. Int J Mech Eng Robot Res 9(8):1134–1139
- [12] Zhang R, Zhou L, Liu Z (2024) *Dynamic path planning for mobile robot based on RRT* and dynamic window approach*. J Syst Simul 36(4):957–968
- [13] Qin H, Shao S, Wang T, Li Y, Wang N, Yao C (2022) *An improved dynamic window approach for mobile robot dynamic path planning*. In: 2022 12th Int Conf on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER). IEEE, pp 348–353
- [14] Wang Z, Li P, Li Q, Wang Z, Li Z (2023) *Motion planning method for car-like autonomous mobile robots in dynamic obstacle environments*. IEEE Access 11:137387–137400
- [15] Dang ST, Dinh XM, Kim TD, Xuan HL, Ha MH (2023) *Adaptive backstepping hierarchical sliding mode control for 3-wheeled mobile robots based on RBF neural networks*. Electronics 12(11):2345
- [16] Schwenzer M, Ay M, Bergs T, Abel D (2021) *Review on model predictive control: An engineering perspective*. Int J Adv Manuf Technol 117(5):1327–1349
- [17] Sani M, Robu B, Hably A (2021) *Dynamic obstacles avoidance using nonlinear model predictive control*. In: IECON 2021–47th Annual Conference of the IEEE Industrial Electronics Society. IEEE, pp 1–6
- [18] Sani M, Hably A, Robu B, Dumon J, Meslem N (2023) *Real-time Dynamic Obstacle Avoidance For A Non-holonomic Mobile Robot*. In: 2023 IEEE 32nd Int Symposium on Industrial Electronics (ISIE). IEEE, pp 1–6
- [19] Quang HD, Tran TL, Manh TN, Manh CN, Nhu TN, Duy NB (2022) *Design a nonlinear MPC controller for autonomous mobile robot navigation system based on ROS*. Int J Mech Eng Robot Res 11(6)
- [20] Sánchez I, D’Jorge A, Raffo GV, González AH, Ferramosca A (2021) *Nonlinear model predictive path following controller with obstacle avoidance*. J Intell Robot Syst 102:1–18
- [21] Le VA, Tadiparthi V, Chalaki B, Mahjoub HN, D’sa J, Moradi-Pari E, Malikopoulos AA (2024) *Multi-robot cooperative navigation in crowds: A game-theoretic learning-based model predictive control approach*. In: 2024 IEEE Int Conf on Robotics and Automation (ICRA). IEEE, pp 4834–4840
- [22] Zeng J, Li Z, Sreenath K (2021) *Enhancing feasibility and safety of nonlinear model predictive control with discrete-time control barrier functions*. In: 2021 60th IEEE Conf on Decision and Control (CDC). IEEE, pp 6137–6144
- [23] Nguyen MN, Van M, McIlvanna S, Sun Y, Close J, Olayemi K, Jin Y (2024) *Model-free safety critical model predictive control for mobile robot in dynamic environments*. IEEE Trans Intell Veh
- [24] De Carvalho Filho JGN, Carvalho EÁN, Molina L, Freire EO (2019) *The impact of parametric uncertainties on mobile robots velocities and pose estimation*. IEEE Access 7:69070–69086
- [25] Andersson JA, Gillis J, Horn G, Rawlings JB, Diehl M (2019) *CasADi: a software framework for nonlinear optimization and optimal control*. Math Program Comput 11:1–36
- [26] Kirches C (2011) *The direct multiple shooting method for optimal control*. In: Fast Numerical Methods for Mixed-integer Nonlinear Model-Predictive Control. Vieweg+ Teubner Verlag, Wiesbaden, pp 13–29
- [27] Wächter A, Biegler LT (2006) *On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming*. Math Program 106:25–57