

A METHOD OF OBSTACLE AVOIDANCE FOR AMR ROBOT IN WAREHOUSE AUTOMATION

Nong Minh Ngoc^{1*}, Do Nang Toan², Vu Vinh Quang³, Ngo Ngoc Vu⁴

¹Thai Nguyen University, ²Institute of Information Technology - Vietnam Academy of Science and Technology

³TNU - University of Information and Communication Technology, ⁴TNU - University of Technology

ARTICLE INFO		ABSTRACT
Received:	05/12/2022	Currently, automation models are born to solve all the unnecessary risks of traditional models and in the supply chain, warehouse automation has brought great efficiency in warehouse management. This paper approaches the obstacle avoidance method for AMR robot in warehouse automation proposed based on the calculation of the probability of collision that may occur between the robot and the objects when a robot is on its trajectory, and the adjusted trajectory during the movement processing. The calculation of the collision were provided to support the robot to choose a safe area to move through and over obstacles. An obstacle avoidance algorithm based on complement of probability is proposed. The results of the probability calculation and the robot path finding algorithm are tested and simulated.
Revised:	26/12/2022	
Published:	26/12/2022	
KEYWORDS		
Warehouse automation		
Avoidance obstacle		
Collision probability		
Collision prediction		
AMR robot		

MỘT TIẾP CẬN TRÁNH VẬT CẢN CHO ROBOT AMR TRONG TỰ ĐỘNG HÓA NHÀ KHO

Nông Minh Ngọc^{1*}, Đỗ Năng Toàn², Vũ Vinh Quang³, Ngô Ngọc Vũ⁴

¹Đại học Thái Nguyên, ²Viện Công nghệ thông tin - Viện Hàn lâm KH và CN Việt Nam

³Trường Đại học Công nghệ thông tin và Truyền thông - ĐHTN

⁴Trường Đại học Kỹ thuật Công nghiệp - ĐHTN

THÔNG TIN BÀI BÁO	TÓM TẮT	
Ngày nhận bài: 05/12/2022	Hiện nay, các mô hình tự động được ra đời để khắc phục tất cả những rủi ro không đáng có của những mô hình truyền thống và trong chuỗi cung ứng, tự động hóa kho hàng ra đời đã đem lại hiệu quả rất lớn trong công tác quản lý kho. Bài báo này tiếp cận phương pháp tránh vật cản cho robot tự hành AMR ứng dụng trong tự động hóa nhà kho (warehouse automation) được đề xuất dựa trên tính toán xác suất va chạm có thể xảy ra giữa robot và các đối tượng khi robot di chuyển trong quỹ đạo cho trước cũng như quỹ đạo điều chỉnh theo thời gian thực. Việc tính toán xác suất là cơ sở cho robot ra quyết định cho robot lựa chọn vùng an toàn để di chuyển xuyên qua và tránh chướng ngại vật. Phương pháp tránh vật cản dựa trên phân bổ phần bù xác suất được đề xuất, kết quả tính toán xác suất và thuật toán tìm đường cho robot được thử nghiệm và mô phỏng.	
Ngày hoàn thiện: 26/12/2022		
Ngày đăng: 26/12/2022		
TỪ KHÓA		
Tự động hóa nhà kho		
Tránh chướng ngại vật		
Xác suất va chạm		
Dự đoán va chạm		
Robot AMR		

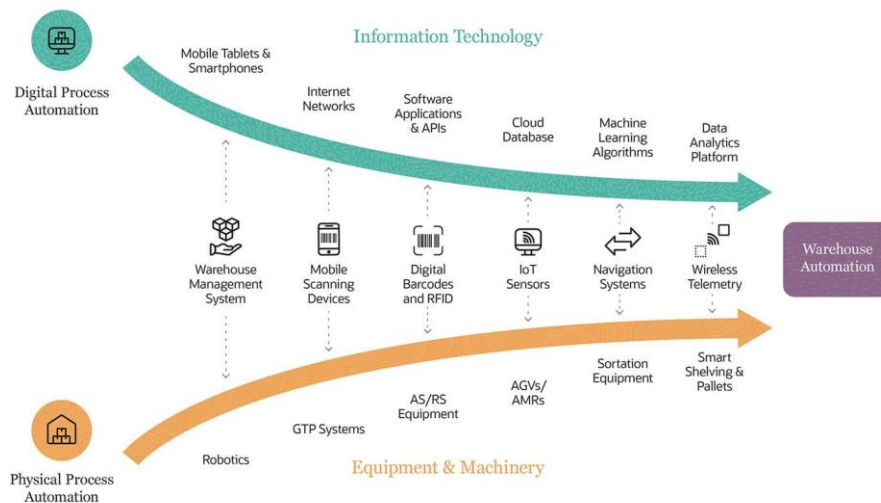
DOI: <https://doi.org/10.34238/tnu-jst.7049>

* Corresponding author. Email: ngocnm@tnu.edu.vn

1. Giới thiệu

Tự động hóa cơ giới là một cách sử dụng công nghệ để giảm thiểu sự di chuyển của nhân viên và xây dựng quy trình làm việc hiệu quả hơn. Đồng thời, cũng là một phần của dự án tự động hóa doanh nghiệp giúp loại bỏ việc sử dụng nhiều lao động đến các công việc có tính chất lặp đi lặp lại (nhập/xuất dữ liệu hay phân tích dữ liệu thủ công).

Hình 1 mô tả các thành phần chủ yếu của hệ thống tự động nhà kho gồm: 1- Tự động hóa kỹ thuật số (Digital Process Automation) sử dụng dữ liệu và phần mềm để giảm thiểu các quy trình làm việc thủ công. Ví dụ như: công nghệ nhận dạng và thu thập dữ liệu tự động (AIDC), như mã vạch di động...; 2- Tự động hóa vật lý (Physical Process Automation) là một cách sử dụng công nghệ để giảm thiểu sự di chuyển của nhân viên và xây dựng quy trình làm việc hiệu quả hơn. Bao gồm: sử dụng hệ thống robot AGV, AMR, hệ thống băng tải thông minh...



Hình 1. Hệ thống tự động nhà kho [1]

Công nghệ tự động hóa kho bao gồm:

1. **Giao hàng tận nơi (GTP):** là một trong những phương pháp phổ biến nhất để tăng hiệu quả và giảm tắc nghẽn. Danh mục này bao gồm băng tải, băng chuyền và hệ thống nâng thẳng đứng. Khi được áp dụng đúng cách, các hệ thống GTP có thể tăng gấp đôi hoặc gấp ba tốc độ lấy hàng trong kho.

2. **Hệ thống lưu trữ và truy xuất tự động (AS/RS):** công nghệ thực hiện GTP gồm các hệ thống và thiết bị tự động như xe chở vật liệu, xe đưa đón tote và máy xúc lật mini để lưu trữ và lấy vật liệu hoặc sản phẩm.

3. **Xe có hướng dẫn tự động (AGV):** những xe này sử dụng dải từ tính, dây điện hoặc cảm biến để điều hướng một con đường cố định qua nhà kho.

4. **Robot di động tự động (AMR):** AMR sử dụng bộ cảm biến tinh vi, hệ thống GPS, trí tuệ nhân tạo, máy học và điện toán để lập kế hoạch đường đi nhằm diễn giải và điều hướng trong một trường làm.

5. Ngoài ra, hệ thống còn bao gồm: Hệ thống Pick-to-Light và Put-to-Light, Chọn bằng giọng nói và nhiệm vụ, Hệ thống phân loại tự động... nhằm tối ưu hóa khả năng vận hành.

Trong tự động hóa nhà kho, robot AMR (Autonomous Mobile Robot) là một thành phần quan trọng góp phần tự động hóa các quy trình vận hành [2], [3], mang lại hiệu quả về nguồn lực, kinh tế và thay thế con người làm việc tại môi trường độc hại, nguy hiểm. Bởi vậy, lớp bài toán tìm đường, dẫn đường cho các robot này là mục tiêu nghiên cứu tập trung vào việc tìm đường đi ngắn nhất, tránh va chạm với các chướng ngại vật cố định, chướng ngại vật đang di chuyển. Hình 2, cho thấy hệ thống các robot AMR hoạt động trong nhà kho thông minh với nhiệm vụ nhận/phát/sắp xếp hàng hóa trên hệ thống kệ hàng.



Hình 2. Robot AMR trong tự động hóa kho hàng

Có nhiều phương pháp tìm kiếm đường đi cho robot AMR được nghiên cứu: thuật toán dựa trên đồ thị như thuật toán Dijkstra hay thuật toán A* được đề xuất bởi Guo và cộng sự (2012) [4]; Liu và cộng sự (2019) [5]. Hiệu quả của những thuật toán này phụ thuộc vào việc xây dựng thông tin bản đồ (đồ thị). Tuy nhiên, do sự thay đổi thiết kế của hệ thống nhà kho xảy ra thường xuyên, nên việc thay đổi thông tin bản đồ cũng phải thay đổi thường xuyên theo, không có tính kế thừa, linh hoạt trong cập nhật. Phương pháp Simultaneous Localization and Mapping (SLAM) đề xuất bổ sung việc cập nhật tọa độ bản đồ gần như tự động [6]. Tuy nhiên, để tạo ra bản đồ thực tế, cần phải lập thủ công các nút chính trên tọa độ dẫn đến tốn thời gian tính toán.

Phương pháp Model Predictive Control (MPC) được đề xuất bởi Li và cộng sự (2019) [7]; Xu và cộng sự (2020) [8], các phương pháp này tham chiếu đến đường dẫn được thuật toán A* tạo trước. Bởi vậy, các phương pháp này cũng đòi hỏi phải tạo bản đồ mỗi khi thay đổi bố cục của kho hàng. Shini và cộng sự (2022) [9], đã đề xuất một phương pháp để thực hiện quy hoạch đường dẫn cho nhiều robot chỉ sử dụng MPC linh hoạt hơn, đáp ứng với những thay đổi trong việc bố trí kho hàng. Đồng thời, tác giả giới thiệu một số thay đổi trong thiết kế của hàm mục tiêu của MPC trong điều kiện MPC được sử dụng một mình và robot không thể di chuyển do có sự tồn tại của giải pháp cục bộ tối ưu.

Hầu hết các phương pháp trên đều dựa vào hệ thống bản đồ cho trước, tuy nhiên trong tìm đường, việc phát sinh các tình huống thực tế của robot đối với các vật cản (robot khác, con người, máy móc) là không thể tránh khỏi. Bởi vậy, ứng với bài toán thực tế khi robot hoạt động trong môi trường động (dynamic environment) thì các phương pháp này còn hạn chế. Do vậy, chúng tôi đề xuất một phương pháp tránh vật cản cho robot AMR khi gặp chướng ngại vật dựa trên tính toán xác suất va chạm có thể xảy ra.

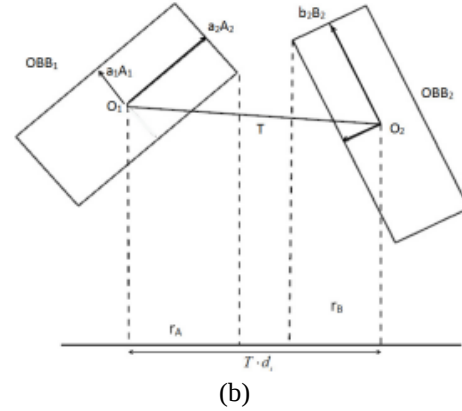
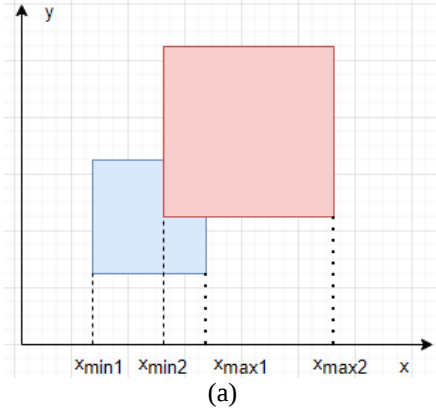
2. Tính toán va chạm cho robot tự hành

Trong điều khiển robot, việc tránh va chạm với các vật thể trong vùng làm việc là hết sức quan trọng, vấn đề mấu chốt là làm sao đánh giá được khả năng xảy ra va chạm của robot với các chuyển động trong không gian có quỹ đạo phức tạp và có các chướng ngại cũng chuyển động.

2.1. Tính toán va chạm dựa trên kỹ thuật hộp bao

Tính toán va chạm dựa vào hộp bao AABB đã được đề cập bởi S. Kockara và cộng sự [10]. Trong đó, việc phát hiện va chạm giữa hai hộp bao AABB xác định bởi $[C_1, a_0, a_1, a_2]$ và $[C_2, b_0, b_1, b_2]$ với giả sử $a_i > 0, b_j > 0, i, j = 0, 1, 2$ được xác định thông qua tọa độ cao nhất và thấp nhất của

mỗi hộp bao. Kí hiệu $(x_{\min 1}, y_{\min 1}, z_{\min 1})$, $(x_{\max 1}, y_{\max 1}, z_{\max 1})$ là tọa độ thấp nhất và cao nhất của hộp bao có tâm C_1 được mô tả trong Hình 3^a.



Hình 3. Kỹ thuật phát hiện va chạm

(a) Tính toán va chạm theo AABB

(b) Tính toán va chạm theo OBB

$$\begin{aligned} x_{\min 1} &= C_0 - \frac{a_0}{2}, & x_{\max 1} &= C_0 + \frac{a_0}{2} \\ y_{\min 1} &= C_0 - \frac{b_0}{2}, & y_{\max 1} &= C_0 + \frac{b_0}{2} \\ z_{\min 1} &= C_0 - \frac{c_0}{2}, & z_{\max 1} &= C_0 + \frac{c_0}{2} \end{aligned} \quad (1)$$

Tương tự, tính được $(x_{\min 2}, y_{\min 2}, z_{\min 2})$, $(x_{\max 2}, y_{\max 2}, z_{\max 2})$ là tọa độ thấp nhất và cao nhất của hộp bao xác định bởi tâm C_2 . Hai hộp bao AABB va chạm nhau nếu xảy ra một trong bốn điều kiện sau:

$$\begin{aligned} (x_{\min 1}, y_{\min 1}, z_{\min 1}) &\in [(x_{\min 2}, y_{\min 2}, z_{\min 2}), (x_{\max 2}, y_{\max 2}, z_{\max 2})] \\ (x_{\max 1}, y_{\max 1}, z_{\max 1}) &\in [(x_{\min 2}, y_{\min 2}, z_{\min 2}), (x_{\max 2}, y_{\max 2}, z_{\max 2})] \\ (x_{\min 2}, y_{\min 2}, z_{\min 2}) &\in [(x_{\min 1}, y_{\min 1}, z_{\min 1}), (x_{\max 1}, y_{\max 1}, z_{\max 1})] \\ (x_{\max 2}, y_{\max 2}, z_{\max 2}) &\in [(x_{\min 1}, y_{\min 1}, z_{\min 1}), (x_{\max 1}, y_{\max 1}, z_{\max 1})] \end{aligned} \quad (2)$$

Để tìm điểm va chạm, ta chọn điểm va chạm là đỉnh tương ứng với một trong bốn trường hợp trên. Phương pháp này tính toán đơn giản, dễ cài đặt xong độ chính xác không cao.

Tính toán va chạm dựa vào hộp bao OBB (Oriented Bounding Boxes): được David Eberly [11] đề cập áp dụng phát hiện va chạm giữa các đối tượng trên mô hình không gian 3D. Trong đó, hình hộp OBB (Hình 3b) bao gồm một tâm C , ba vector $\vec{A}_0, \vec{A}_1, \vec{A}_2$ chỉ hướng của hình hộp và 3 hệ số độ dài tương ứng với kích thước của hình hộp là $a_0 > 0, a_1 > 0, a_2 > 0$. Khi đó, 8 đỉnh của hình hộp sẽ được xác định như sau:

$$C + \sum_{i=0}^2 s_i a_i * \vec{A}_i \quad |s_i| = 1, i = 0, 1, 2. \quad (3)$$

Việc tính toán điểm va chạm giữa hai OBBs được thực hiện bằng việc tìm chính xác điểm va chạm. Đối với hai hộp bao, nếu chúng va chạm với nhau ở dạng *đỉnh - đỉnh*, *đỉnh - cạnh*, *cạnh - cạnh*, *đỉnh - mặt* thì điểm tiếp xúc là duy nhất. Nhưng nếu chúng va chạm với nhau ở dạng *mặt - mặt*, *cạnh - mặt* thì sẽ có vô số điểm tiếp xúc, khi đó chỉ cần đưa ra một điểm bất kỳ là được.

2.2. Tính toán va chạm dựa trên xác suất

Các nghiên cứu về tính toán xác suất va chạm giữa các đối tượng được ứng dụng không chỉ giải quyết vấn đề tránh vật cản [12]-[14] (các tàu vận tải, thiết bị tự lái, robot tự hành,...) mà còn trong khoa học quân sự như: tính toán điểm chính xác, vùng chính xác tạo va chạm (đánh chặn, tính toán điểm mục tiêu của vũ khí và tên lửa,...)

Những năm gần đây, đã có nhiều phương pháp tính toán xác suất va chạm đã được đề xuất như: Chen và cộng sự [15] đã nghiên cứu một phương pháp bằng cách sử dụng logic mờ sử dụng lý thuyết thời gian và khoảng cách đến điểm tiếp cận gần nhất (Time to the Closest Point of Approach –TCPA, Distance of the Closest Point of Approach-DCPA) để nhận biết tránh va chạm; Son và cộng sự [16] dựa vào fuzzy logic đã tính toán rủi ro va chạm có sử dụng TCPA và DCPA là hàm thành viên; Namgung [17] năm 2019 sử dụng neuro fuzzy để tính toán chỉ số rủi ro va chạm (Collision Risk Index -CRI); Fulgenzi và cộng sự [18] thực hiện các mô phỏng tránh va chạm trong môi trường động (dynamic environment) bằng cách kết hợp các lưới công suất động được cung cấp bởi cảm biến và xác suất vận tốc của chướng ngại vật (Probabilistic Velocity Obstacles - PVO); Kim và cộng sự [19] năm 2022, sử dụng bộ lọc Kalman kết hợp với bước dự đoán của bộ lọc UKF (Unscented Kalman filter).

Hầu hết các nghiên cứu trên tập trung vào trạng thái hiện tại của đối tượng và tính toán nguy cơ va chạm dựa trên các phương pháp VO, việc xác định quỹ đạo chuyển động của đối tượng tập trung vào việc tránh các vùng có khả năng va chạm, cho nên đường đi của robot tự hành là không tối ưu về quãng đường, tổn kém thời gian.

3. Phương pháp tìm đường cho robot bằng độ đo va chạm

Chúng tôi đề xuất phương pháp tối ưu đường đi cho robot tự hành dựa trên phương pháp tính toán xác suất va chạm, từ đó giúp robot tự hành lựa chọn vùng di chuyển và quỹ đạo chuyển động an toàn.

Xét một vật thể được giới hạn bởi miền Ω . Khi đó đường kính của vật thể được định nghĩa là $2r = \max \rho(M_1; M_2); \forall M_1, M_2 \in \Omega$

Giả sử chúng ta có m vật thể $O(M_i, r_i), i=1 \dots m$ cùng chuyển động trong một không gian quan sát H giới hạn $0 \leq x \leq x_{\max}; 0 \leq y \leq y_{\max}; 0 \leq z \leq z_{\max}$; với các giả thiết sau:

Xét hai vật thể giới hạn bởi hai miền Ω_1, Ω_2 , độ đo khả năng xảy ra va chạm giữa chúng được xác định bởi công thức (4) trong đó V là thể tích các miền tương ứng.

$$p[1, 2, t] = \frac{V(\Omega_1 \cap \Omega_2)}{V(\Omega_1 \cup \Omega_2)} \quad (4)$$

Công thức (4) được coi là công thức tổng quát tính xác suất xảy ra va chạm giữa hai vật thể. Trong trường hợp đặc biệt khi hai vật thể di chuyển trùng khít vào nhau thì xác suất bằng 1 và khi hai vật thể tách rời nhau thì xác suất bằng không. Đồng thời, độ đo càng lớn thì khả năng xảy ra va chạm càng cao.

Xét 2 vật thể được mô tả bằng hai mặt tròn $O(M_1, r_1), O(M_2, r_2)$. Kí hiệu $d = \rho(M_1, M_2)$ là khoảng cách giữa 2 tâm. Khi đó điều kiện để 2 mặt tròn giao nhau là $d < r_1 + r_2$.

Để thấy rằng:

$$\begin{aligned} S(O_1 \cap O_2) &= r_1^2 \arccos\left(\frac{r_1 - h_1}{r_1}\right) - (r_1 - h_1) \times \sqrt{r_1^2 - (r_1 - h_1)^2} \\ &\quad + r_2^2 \arccos\left(\frac{r_2 - h_2}{r_2}\right) - (r_2 - h_2) \times \sqrt{r_2^2 - (r_2 - h_2)^2} \\ S(O_1 \cup O_2) &= \pi r_1^2 + \pi r_2^2 - S(O_1 \cap O_2) \end{aligned}$$

Trong đó, $h_1 = r_1 - \frac{r_1^2 + d^2 - r_2^2}{2d}$; $h_2 = r_2 - \frac{r_2^2 + d^2 - r_1^2}{2d}$;

Và công thức tổng quát xác định độ đo va chạm giữa 2 vật được xác định:

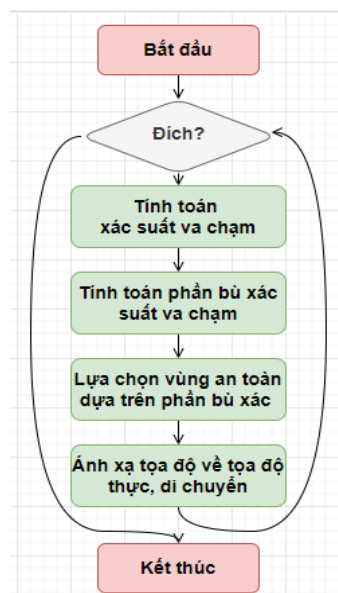
$$p[1, 2, t] = \begin{cases} \frac{S(O_1 \cap O_2)}{S(O_1 \cup O_2)}; & |r_1 - r_2| \leq d < r_1 + r_2; \\ 0; & d \geq r_1 + r_2; \\ \frac{\min(\pi r_1^2, \pi r_2^2)}{\max(\pi r_1^2, \pi r_2^2)}; & d < |r_1 - r_2|. \end{cases} \quad (5)$$

Với không gian 3 chiều, công thức tổng quát xác định độ đo va chạm giữa 2 vật là:

$$p[1, 2, t] = \begin{cases} \frac{\pi h_1^2 (r_1 - \frac{h_1}{3}) + \pi h_2^2 (r_2 - \frac{h_2}{3})}{\frac{4}{3} \pi (r_1^3 + r_2^3) - \pi h_1^2 (r_1 - \frac{h_1}{3}) - \pi h_2^2 (r_2 - \frac{h_2}{3})}; & |r_1 - r_2| \leq d < r_1 + r_2; \\ 0; & d \geq r_1 + r_2; \\ \frac{\min\left(\frac{4}{3} \pi r_1^3, \frac{4}{3} \pi r_2^3\right)}{\max\left(\frac{4}{3} \pi r_1^3, \frac{4}{3} \pi r_2^3\right)}; & d < |r_1 - r_2|. \end{cases} \quad (6)$$

Trong trường hợp tổng quát, với m vật thể được xác giới hạn bởi các miền $\Omega_i, i=1, 2, \dots, m$ thì công thức xác định khả năng xảy ra va chạm giữa m vật thể tại thời điểm (t) là.

$$p[1, 2, \dots, m, t] = \frac{\sum_{i \neq j}^m V(\Omega_i \cap \Omega_j)}{\sum_{i \neq j}^m V(\Omega_i \cup \Omega_j)}. \quad (7)$$



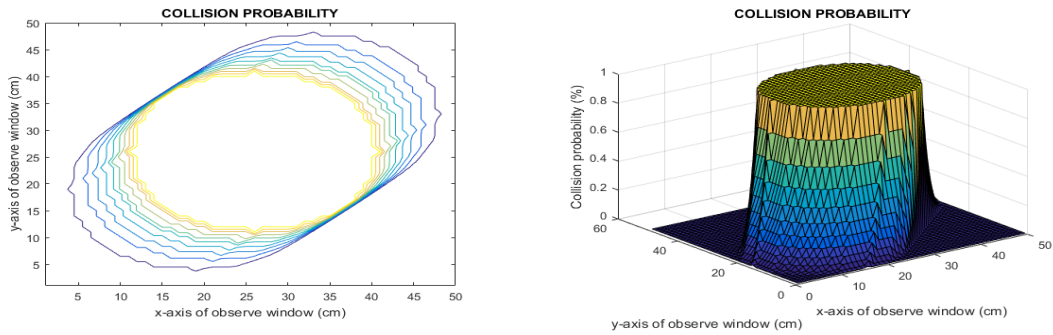
Hình 4. Sơ đồ thuật toán tìm đường

Ý tưởng phương pháp tìm đường cho robot tự hành được đề xuất dựa trên phân bù của xác suất va chạm tính được. Sau khi tính toán được xác suất va chạm, việc xây dựng hướng đi chuyên, quỹ đạo di chuyển của robot tự hành sẽ được ánh xạ lên hệ thống định vị GPS trên cơ sở quỹ đạo tính toán của phân bù xác suất va chạm. Tại mỗi thời điểm $t(i)$ robot tự hành sẽ thực hiện: 1- Thực hiện tính toán đo độ va chạm của xe tự hành; 2- Xác định phân bù của xác suất va chạm; 3- Xây dựng đồ thị của phân bù, lựa chọn điểm di chuyển tiếp theo cho xe tự hành; 4- Ánh xạ ngược với môi trường thực tế hiện có của xe tự hành. Hình 4 mô tả thuật toán tìm đường cho robot tự hành.

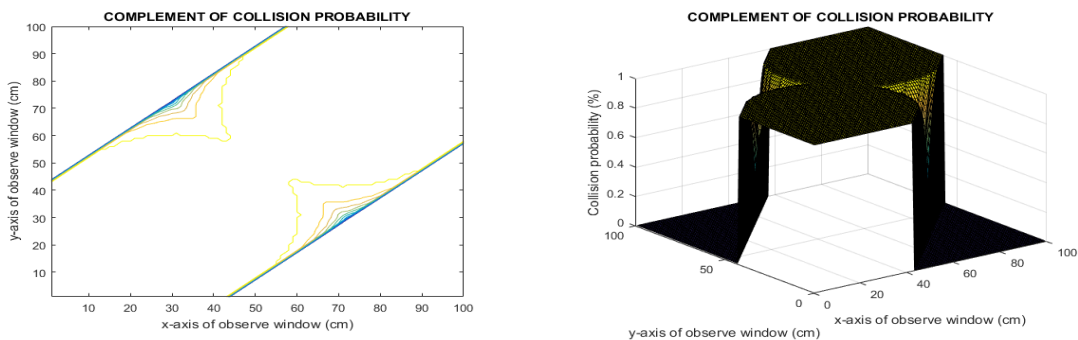
4. Kết quả tính toán và mô phỏng

Với dữ liệu đầu vào thử nghiệm $r_1=50$, $r_2=60$, $t=100$, $n=100$, kết quả tính toán xác suất va chạm theo công thức (5) đã được thể hiện trên hình 5. Trong đó, r_1 , r_2 là bán kính vùng an toàn tương ứng của robot tự hành và vật cản, t là biến thời gian, n là cửa sổ quan sát của mô phỏng.

Đồng thời quỹ đạo tính toán phân bù xác suất va chạm đã được thể hiện ở hình 6. Việc ánh xạ quỹ đạo này như là một gợi ý cho việc xác định thuật toán tránh vật cản cho robot.



Hình 5. Kết quả tính toán xác suất va chạm



Hình 6. Quỹ đạo gợi ý cho robot tự hành tránh vật cản

Dữ liệu được cài đặt trên robot có cấu tạo được mô tả trong Hình 7.



Hình 7. Mô hình robot thực tế phục vụ cài đặt

5. Kết luận

Hướng tiếp cận về một phương pháp tránh vật cản cho robot tự hành AMR dựa trên xác suất va chạm đã được trình bày là cơ sở cho robot ra quyết định di chuyển qua vùng an toàn và xây dựng quỹ đạo cho nó. Kết quả tính toán xác suất va chạm đã kiểm chứng mô hình toán học trong công thức (5) và (6). Tiếp theo, chúng tôi tiếp tục nghiên cứu về bài toán tránh va chạm với các vật thể chuyển động, trong đó có yếu tố môi trường, đồng thời tiến hành phân vùng đồng mức va chạm cho bài toán dự đoán va chạm để robot tự hành đưa ra các quyết định chính xác, an toàn và tin cậy.

TÀI LIỆU THAM KHẢO/ REFERENCES

- [1] A. Jenkins, "Warehouse Automation Explained: Types, Benefits & Best Practices," December 10, 2020. [Online]. Available: <https://www.netsuite.com/portal/resource/articles/inventory-management/warehouse-automation.shtml>. [Accessed November 30, 2022].
- [2] W. Sven and H. G. Eric, "Chapter 3 - Smart warehouses- a sociotechnical perspective," in *The Digital Supply Chain*, Elsevier, 2022, pp. 47-60, doi: 10.1016/B978-0-323-91614-1.00003-4.
- [3] F. Giuseppe, R. D. Kosterb, S. Fabio, and O. S. Jan, "Planning and control of autonomous mobile robots for intralogistics: Literature review and research agenda," *European Journal of Operational Research*, vol. 294, no. 2, pp. 405-426, 2021.
- [4] L. Guo, Q. Yang, and W. Yan, "Intelligent pathplanning for automated guided vehicles system based on topological map," in *IEEE Conference on Control, Systems Industrial Informatics*, 2012, pp. 69-74.
- [5] M. Liu, H. Ma, J. Li, and S. Koenig, "Task and path planning for multi-agent pickup and delivery," in *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2019, pp. 1152-1160.
- [6] Y. Chen, Y. Wu, and H. Xing, "A complete solution for agv slam integrated with navigation in modern warehouse environment," in *Chinese Automation Congress (CAC)*, 2017, pp. 6418-6423.
- [7] J. Li, M. Ran, H. Wang, and L. Xie, "Mpcbased unified trajectory planning and tracking control approach for automated guided vehicles," in *IEEE 15th International Conference on Control and Automation (ICCA)*, 2019, pp. 374-380.
- [8] H. Xu, Z. Yu, X. Lu, S. Wang, S. Li, and S. Wang, "Model predictive control-based path tracking control for automatic guided vehicles," in *4th CAA International Conference on Vehicular Control and Intelligence (CVCI)*, 2020, pp. 627-632.
- [9] S. Ishihara, M. Kanai, R. Narikawa, and T. Ohtsuka, "A Proposal of Path Planning for Robots in Warehouses by Model Predictive Control without Using Global Paths," *IFAC Papers OnLine*, vol. 55, no. 37, pp. 573-578, 2022.
- [10] S. Kockara, T. Halic, and K. Iqbal. "Collision Detection: A Survey," in *IEEE International Conference on Systems, Man and Cybernetics*, 2007, doi: 10.1109/ICSMC.2007.4414258
- [11] D. Eberly, "Dynamic Collision Detection using Oriented Bounding Boxes," 1999. [Online]. Available: <http://www.geometrictools.com>. [Accessed November 30, 2022].
- [12] Y. Hu, A. Zhang, W. Tian, J. Zhang, and Z. Hou, "Multi-ship collision avoidance decision-making based on collision risk index," in *MSC 101/5/5, Interim Guidelines for MASS Trials*, International Maritime Organization (IMO), London, U.K, 2020.
- [13] D.-H. Kim, S.-U. Lee, J.-H. Nam, and Y. Furukawa, "Determination of ship collision avoidance path using deep deterministic policy gradient algorithm," *J. Soc. Naval Arch.Kor.*, vol. 56, no. 1, pp. 58-65, 2019.
- [14] S. Armin, "Predictive Probability Based Collision Avoidance for Unmanned Aerial Vehicles," *Technical Report*, 2019, doi: 10.13140/RG.2.2.15632.97283.
- [15] D. Chen, C. Dai, X. Wan, and J. Mou, "A research on AIS-based embedded system for ship collision avoidance," in *International Conference on Transportation Information and Safety (ICTIS)*, 2015, pp. 512-517.
- [16] N. Son, I. Gong, S. Kim, and C. Lee, "Study on the estimation of collision risk of ship in ship handling simulator using fuzzy algorithm and environmental stress model," *J. Kor. Inst. Navig. Port Res. Kor. Inst. Navig. Port Res.*, vol. 33, no. 1, pp. 43-50, 2009.
- [17] H. Namgung, "Inference rule of collision risk index based on ship near collision via adaptive neuro fuzzy inference system," *Adv. Sci. Technol. Eng. Syst. J.*, vol. 4, no. 4, pp. 152-160, 2019.
- [18] C. Fulgenzi, A. Spalanzani, and C. Laugier, "Combining probabilistic velocity obstacles and occupancy grid for safe navigation in dynamic environments," in *Workshop on safe Navigation, IEEE ICRA*, Rome, Italy, 2007.
- [19] J. H. Kim, S. W. Lee, and E. S. Jin, "Collision avoidance based on predictive probability using Kalman filter," *International Journal of Naval Architecture and Ocean Engineering*, vol. 14, 2022, Art. no. 100438.