OBSTACLE AVOIDANCE PATH OF WHEELED AGRICULTURAL HANDLING ROBOTS IN WAREHOUSE BASED ON IMPROVED ACO-DWA ALGORITHM
/
基于改进 ACO-DWA 算法的轮式农业机器人仓库搬运避障路径研究

Kan ZHANG*)
School of Computer Engineering, Shanxi Vocational University of Engineering Science and Technology, Jinzhong, Shanxi / China
Tel: +8613834246680; E-mail: zk321672024@126.com
Corresponding author: Kan Zhang
DOI: https://doi.org/10.35633/inmateh-74-51

***Keywords:** Artificial potential field; Ant colony algorithm; Agricultural robot; Path optimization; Obstacle avoidance*

ABSTRACT
The existing obstacle avoidance control algorithms for wheeled agricultural warehouse handling robots are prone to the local optimal solution in the process of path optimization and collision can easily occur during multi-robot simultaneous operation. Given this, the obstacle avoidance of wheeled agricultural warehouse handling robots was explored in this study, and an obstacle avoidance path planning algorithm for wheeled agricultural handling robots in warehouse based on improved ACO-DWA algorithm was proposed. Then, the moving trajectory of agricultural warehouse handling robots during handling process was studied, and their spatial kinematics equation was given. Next, the real-time pose of agricultural warehouse handling robots was detected, and their motion path was planned considering the real-time position of obstacles and the target locations of handling. In addition, the obstacle avoidance controlling quantity of agricultural warehouse handling robots was calculated according to the deviation between the pose of robots and the planned path. Supported by a controller, the obstacle avoidance control work of agricultural warehouse handling robots was realized. It was concluded through the effect experiment that compared with the traditional method, the improved ACO-DWA algorithm designed in this study significantly reduced the number of collisions between agricultural robots, and through practical application, the proposed algorithm can meet the needs of improving warehouse logistics management efficiency.

摘要
针对现有轮式农业机器人仓储搬运避障控制算法存在的路径寻优易陷入局部最优解，及多机器人同时作业易发生碰撞等问题，对轮式农业机器人的仓库避障控制进行了研究，并提出提出了一种基于改进 ACO-DWA 算法的轮式农业机器人仓库搬运避障路径规划算法；对农业机器人仓储机器人搬运过程中的移动轨迹进行了研究，给出了机器人空间运动学方程。检测农业农业机器人仓储搬运实时位姿，考虑障碍物实时位置和搬运目标地点，规划机器人移动路径。根据农业机器人位姿与规划路径之间的偏差，计算农业机器人仓储搬运避障控制量，在控制器的支持下，实现农业机器人仓储搬运避障控制工作。通过效果测试实验得出结论：与传统方法相比，本文所设计的改进 ACO-DWA 算法，农业机器人之间发生碰撞的次数也显著少于传统避障控制算法，经实际应用能够满足提升仓储物流管理效率的需求。

INTRODUCTION
Agricultural warehouse handling robots capable of automatic navigation, intelligent identification and efficient cargo handling have been increasingly widely applied to the fields of agricultural products warehousing and agricultural logistics, helping enterprises to realize automatic, intelligent and efficient agricultural warehouse management *(Hao et al., 2022)*. In dynamic and complicated warehouse environments, the autonomous navigation and obstacle avoidance of agricultural warehouse handling robots are the key to ensuring efficient and safe operations. When it comes to the obstacle avoidance of agricultural warehouse handling robots, they perceive the surrounding environment and identify obstacles in a real-time manner through sensors and plan a collision-free path according to a certain algorithm, so as to successfully evade obstacles and reach the destination *(Guo et al., 2022)*.

Obstacle avoidance control methods for agricultural warehouse handling robots have been proposed in order to ensure the obstacle avoidance effect and enhance the safety of robots during movement to the greatest extent on the premise of ensuring the handling work quality. Agricultural handling robots, which integrate various key technologies, such as deep visual sensor positioning, obstacle detection & path planning, energy consumption and thermal management and system integration & optimization, have experienced rapid development and application in the military and civilian field *(Dang, 2021)*. Especially with the modernization, informatization and intelligent development of agricultural production, plant protection robots that can operate autonomously in fields and on edges of fields possess broad development prospects, which, on the one hand, reduce the labor intensity of agricultural production, and on the other hand, improve working performance and quality *(Zheng et al., 2021)*.

Agricultural handling robots have gradually replaced manual labor to conduct such logistics operations as handling and sorting in real production, promoting the intelligent upgrading of agricultural warehouses and driving them to develop towards intelligent and unmanned directions. The research on the obstacle avoidance path of agricultural warehouse handling robots has been successively carried out, but the order of task allocation, the path planning of robots, and multi-robot collaborative handling remain to be deeply explored to ensure the practical application and popularization of handling robots *(Li et al., 2021)*. Meanwhile, the current research difficulty lies in how to coordinate handling robots with other automation equipment in the warehouse and how to conduct path planning for handling robots under the interference of other movable automation equipment to ensure the most efficient operation of the warehouse.

Path planning, a key issue in the technical field of agricultural warehouse handling robots, determines how robots travel to the designated target, involving the perception and calculation of environmental information. At present, the algorithms used in path node search mainly include intelligent algorithms (ant colony optimization (ACO) algorithm *(Rosemann et al., 2012)*, genetic algorithm *(Sarkar et al., 2012)*, algorithms based on graph search (A**(Ornek et al., 2022)*, Hybrid A**(Zhai et al., 2022)*). A* algorithm has been widely used, but when searching in complex unstructured scenes, it is characterized by a large calculated quantity, serious memory consumption and narrow channels, which will easily generate oscillation. Aiming at the slow operation speed of A* algorithm, *Qian et al. (2019)* designed a tracking controller with global convergence by following the idea of backstepping, which can avoid the local stability problem caused by the linearization method when handling nonlinearity problems. *Ji et al. (2021)* reduced the number of inflection points using the differential method, which, however, increased the calculated quantity. *Ding et al. (2022)* proposed the jump point search method, but this method failed to guarantee the global optimal path in complex irregular maps. *Hossain et al. (2022)* used the fourth-order Bezier curve to express the trajectory shape, whose plasticity, however, was restricted by the limitations of the Bezier curve, making it inapplicable to the complex road surface. During the actual operation process in orchard environments, the kinematical constraints of robots should be ensured in addition to obstacle avoidance constraints and distance cost. Therefore, the methods combining path planning and trajectory optimization have been proposed by some scholars, and the planned paths have been subjected to trajectory optimization.

Kobayashi and Motoi (2022) applied the differential flatness technique to the controller design, and expressed the state variables and input variables with the output and its derivatives of the system. By selecting reasonable output variables, the system can be effectively reduced in dimension, which is convenient for the design and solution of the controller. *Wang et al. (2022)* successfully designed a trajectory tracking controller for nonholonomic systems with the help of the input-output feedback local linearization method together with the dynamic expansion method, and achieved a good trajectory tracking effect by means of simulation.

The existing control algorithms of warehouse logistics robots correct travel paths and evade obstacles on the paths based on high-definition cameras, radar sensors, and distance sensors. For example, *Zohaib et al. (2014)* proposed an Automated Guided Vehicle (AGV), which can realize autonomous navigation and obstacle avoidance on the basis of manually setting the control program. However, AGV robots have some problems in global path optimization, and they are prone to local optimum so as to select noneconomical paths *(Zohaib et al., 2014)*. *Hichri et al. (2022)* put forward a path optimization algorithm based on Q-learning on the basis of neural network model and machine learning algorithm. According to the instructions, the existing economical control algorithms for warehouse logistics robots have been selected, and the correction of travel paths and the obstacle avoidance along the paths have been realized mostly based on high-definition cameras, radar sensors and distance sensing. For example, *Tan et al. (2022)* used the backstepping method to construct two kinds of motion controllers from two situations: local minor error and global arbitrary error, which successfully solved the trajectory tracking problem of mobile robots and extended them to nonholonomic systems with simple dynamic models.

Wang et al. (2022) proposed an Automated Guided Vehicle (AGV), which can realize autonomous navigation and obstacle avoidance on the basis of manually setting the control program. However, AGV robots have some problems in global path optimization, and they are prone to local optimum so as to select noneconomical paths *(Wang et al., 2022). Zhou et al. (2022)* proposed a path optimization algorithm based on Q-learning on the basis of neural network model and machine learning algorithm, which can select a more economical path according to instructions and avoid obstacles along the path. *Zhang et al.* (2021) divided different priority sequences for handling robots according to different handling tasks, and low-priority robots should proactively give way to high-priority robots under this sequence condition.

Based on the above analysis, an obstacle avoidance path planning algorithm for wheeled agricultural warehouse handling robots based on improved ACO-DWA algorithm was put forward in this study. Under dynamic environments, image acquisition and preprocessing were performed, dynamic obstacles in the video were positioned according to feature recognition, and the obstacle avoidance path of agricultural warehouse handling robots was planned based on obstacle positioning results. Then, the travel cost of agricultural robots was fused into the objective function of search nodes using the improved ACO-DWA algorithm, path planning was conducted online according to the environmental map, and the acquired path planning effect was evidently better than that obtained by the traditional DWA algorithm in path length, obstacle avoidance ability and calculation time.

MATERIALS AND METHODS

The basic principle for optimally designing the obstacle avoidance control method of agricultural warehouse handling robots is described as follows: the dynamic obstacles in the agricultural warehouse were identified and positioned through image acquisition, image preprocessing, feature extraction and feature recognition. Then, the obstacle avoidance path of agricultural warehouse handling robots was planned according to obstacle positioning results. Next, the controlled quantity of obstacle avoidance was calculated considering the current handling position of agricultural robots, and the agricultural robot's task of obstacle avoidance control was completed under the support of a controller.

Determination of obstacle position in the environment via visual positioning technology

(1) Generation of dynamic environmental images during handling in warehouse

In the optimally designed obstacle avoidance control method for agricultural warehouse handling robots, the dynamic environment of agricultural warehouse handling robots was acquired through a single visual sensor, and the generation result of obstacle images at any moment is displayed as follows:

$$I\left(x_i\left(t\right),y_i\left(t\right)\right)\begin{cases}k_c x_i \cdot f \cdot \cos\alpha \\ k_c y_i \cdot f \cdot \sin\alpha\end{cases} \tag{1}$$

Where k_c is the imaging coefficient of the single visual sensor; f represents the focal length; α stands for the imaging angle of an obstacle, i.e., the included angle between the obstacle and the built-in sensor in the agricultural warehouse handling robot *(Dang, 2021)*. To follow the rule of "everything looks small in the distance and big on the contrary" in the obstacle image, the distance between the imaging sensor and the obstacle should be measured as follows:

$$d = \frac{H\left(k_1 f - h\tan\alpha\right)}{k_1 f\tan\alpha + h} \tag{2}$$

In Formula (2), H and h represent the height of the imaging equipment and the height of the imaging plane, respectively; k_1 is the constant coefficient whose value depends on the work parameters of the single visual sensor. Using the above method, the generation result of dynamic obstacle images can be obtained. Since the optimally designed obstacle avoidance control method operates under dynamic environments, the acquisition frequency of dynamic obstacle images should be set. Controlled by the drive program, obstacle images are continuously acquired, completing the task of image acquisition.

(2) Environmental image preprocessing

In order to provide effective reference for the obstacle avoidance work of agricultural warehouse handling robots, the initial obstacle image was taken as the processing object, and the preprocessing work of the initial obstacle image was completed through image filtering, image enhancement, image correction, obstacle target extraction, etc. Image filtering aimed to filter out the noise information in the obstacle image. Median filtering and mean filtering were combined to ensure the denoising effect on the image. The principle of median filtering is to replace the value of each pixel in the image with the median value of the pixel in the neighborhood. Mean filtering can facilitate the image pixel to change more smoothly, belonging to a linear filtering method. The filtering results of the initially acquired obstacle image in the warehouse environment of agricultural handling robots are exhibited as follows:

$$\begin{cases} g_1(x,y) = \text{med}\{I(x-k,y-1),(k,l \in U)\} \\ g_2(x,y) = \dfrac{1}{m}\sum I(x,y) \end{cases} \tag{3}$$

Where U is the 2D template of the image; m represents the number of pixel points contained in the initial image; $g_1(x,y)$ and $g_2(x,y)$ correspond to the median and mean filtering results, respectively. Under the irradiation of natural light, an object will generate a shade at the bottom or on the surface of another object. The shades of two objects may be overlapped if they are close to each other, leading to the failure to independently extract the image of the obstacle with shade crossing and making it necessary to eliminate the shaded area of the image and complete the image enhancement. The processing results are depicted as below:

$$g_z(x,y) = R(i,j) \times G(i,j) \tag{4}$$

Where $R(i,j)$ and $G(i,j)$ are the pixel value in the initial image and the grey level histogram, respectively. G(i,j) is calculated as per the following formula:

$$\text{G}(\text{i},\text{j}) = f_{sum}\left(\frac{I(x,y)}{m}\right) \tag{5}$$

Where $f_{sum}()$ is the sum function. Image calibration mainly aims at the deviation of light to improve the light distribution of the image so that the image can be more uniform and natural. This technology is applicable to such scenarios as images with rich details in the shaded part, backlighting scenarios and images with color deviations. The light calibration process of images can be quantitatively expressed as below:

$$g_j = k_c \cdot I^{\kappa_{\text{gamma}}}(x,y) \tag{6}$$

Where k_c is the constant coefficient and k_{gamma} denotes the Gamma coefficient. On this basis, the obstacle objects in the initially acquired image were extracted by means of background subtraction. Background subtraction is a preprocessing method that performs the differential operation on the image to be processed and the background reference image by taking the background of one frame in the image sequence as the reference and obtains the binarized image of the object through threshold division. This method can effectively extract objects from the background. The extraction results of obstacle objects are as follows:

$$Z(x,y) = \begin{cases} 1, | \ I(x,y) - B(x,y) | < \delta \\ 0, \quad \text{otherwise} \end{cases} \tag{7}$$

where $B(x,y)$ is the visual image generation background of agricultural warehouse handling robots; δ represents the background difference threshold. Repeating the above operation, the preprocessing results for the obstacle images of agricultural warehouse handling robots can be obtained.

(3) Identification of obstacles in dynamic environment

Whether any obstacle exists in the dynamic environment corresponding to the current visual field of the agricultural warehouse handling robot is judged by means of feature matching. In the process of obstacle identification, the corresponding matching standard, marked as τ_B, should be set first according to the features of possible obstacles in the warehouse environment. Then, the matching result between the extracted environmental image feature and the feature standard is:

$$\zeta(i) = \frac{\tau_T \cdot \tau_B(i)}{\|\tau_T\| \cdot \|\tau_B(i)\|} \tag{8}$$

where $\tau_B(i)$ is the edge feature standard of the i-th type of obstacles. The relevant data are substituted into Formula 10. If the calculated feature matching degree is higher than the threshold ζ_0, i-th type of obstacles exist in the current dynamic environment, or otherwise, such obstacles do not exist *(Ornek et al., 2022)*. The specific value of variable i is adjusted according to the above method and the operation is repeated until completing the obstacle identification in the dynamic environment.

Obstacle avoidance path planning of agricultural warehouse handling robots

When the real-time position of dynamic obstacles in the dynamic warehouse handling environment, the motion path of robots is planned, ensuring that the planned path will not be overlapped with the obstacle at the same moment. The real-time pose detection result of the agricultural warehouse handling robot is taken as the initial position. Considering the target location of the warehouse handling task, the initial motion path is generated, expressed as follows:

$$L_0 = \frac{x_r\left(y_m - y_r\right)}{x_m - x_r} + y_r \tag{9}$$

In Formula 9, (x_m, y_m) stands for the coordinates of the final target position for the agricultural warehouse handling robot to execute the handling task, and the initially generated motion path is a straight path. According to the movement speed of the robot, its actual node position at any moment can be acquired and compared with the visual positioning information of the dynamic obstacle at the corresponding moment, and then the relationship between their distance and the threshold is judged. If the calculated distance is higher than the threshold, there is no collision risk between the robot and the obstacle, making it unnecessary to adjust the position of this path node, or otherwise, this path node should be adjusted through the following method:

$$\begin{cases} x_{r'}(t) = x_r(t) + \Delta x \\ y_{r'}(t) = y_r(t) + \Delta y \end{cases}, \left(x_r(t), y_r(t)\right) \in L_0 \tag{10}$$

where Δx and Δy are the adjustment quantities of the agricultural warehouse handling robot in horizontal and vertical directions, respectively. Each node on the initially generated motion path of the robot is adjusted according to the above method and connected with two continuous front and real nodes, the adjusted path is subjected to smoothing, and thus the path planning of the robot is completed. Since both the robot and obstacle are under dynamic changes, it is necessary to dynamically update the planned path according to the pose detection result of the robot and the visual positioning result of the obstacle.

ALGORITHM DESIGN

ACO algorithm

(1) State transfer rate. At time t, the state transfer state of the ant from the state node i to the adjacent state node *j* can be defined as below:

$$P_{ij}^m(t) = \begin{cases} \dfrac{\tau_{ij}^{\alpha}(t) \cdot \eta_{ij}^{\beta}(t)}{\sum\limits_{S \in U^m} \tau_{is}^{\alpha}(t) \cdot \eta_{is}^{\beta}(t)}, \text{if} \in U^m \\ 0, \text{else} \end{cases} \tag{11}$$

where $P_{ij}^m(t)$ is the state transfer probability of the *m*-th ant from the state node *i* to the state node *j* at time *t*; $\tau_{ij}^{\alpha}(t)$ is the pheromone concentration the path (*i,j*), in which *α* is the information heuristic factor, reflecting the influence of pheromone on the path selection of ants; $\eta_{ij}^{\beta}(t)$ represents the heuristic function for the *m*-th ant to choose the adjacent state node *j* at the state node *i*, in which *β* is the expected heuristic factor, indicating the importance of heuristic information in guiding the search process of the ant colony; U^m denotes the set of next nodes not accessed by the ant; s is the set of selectable nodes adjacent to the current position; $\tau_{is}^{\alpha}(t)$ is the pheromone concentration of the *m*-th ant between the current state node *i* and each adjacent state node; $\eta_{is}^{\beta}(t)$ is the heuristic function of the *m*-th ant between the current node *i* and each adjacent state node.

The heuristic function $\eta_{ij}(t)$ can be expressed as below:

$$\eta_{ij}(t) = 1 / D_{ij} \tag{12}$$

where D_{ij} is the distance between the state nodes *i* and *j.*

(2) Pheromone concentration updating model

The currently common pheromone concentration updating models include ant-density system (ADS), ant-quantity system (AQS) and ant-cycle system (ACS). ADS and AQS models adopt the local updating strategy while ACS model applies the global updating strategy. Considering the algorithm solving speed and obstacle avoidance ability, AQS model was taken as the prototype in this study. Assuming that the set of state nodes on the path *(i, j)* passed by the *m*-th ant in the current cycle is $X\{(i,j)\,|\,i=1,2,...,n;j=1,2,...,n\}$, then:

$$\Delta\tau_{ij}^{m}(t)=\begin{cases}\dfrac{Q}{D_{ij}},\text{if}(i,j)\in X\\ 0,\text{else}\end{cases} \tag{13}$$

where $\Delta\tau_{ij}^{m}(t)$ is the pheromone concentration increment on the path of the *m*-th ant moving from the state node *i* to the adjacent state node *j* at time *t*; *Q* is the pheromone intensity, which is a constant greater than 0.

Improved DWA algorithm design

The classical DWA algorithm converts the position control of robots into speed control and describes the obstacle avoidance problem into the optimization problem of robot speed space with constraints, including speed, heading direction and surrounding environmental obstacles' position constraints. The speed set that depends on the physical constraints of obstacles around the robot's moving trajectory nodes and is composed of the robot's longitudinal speed and yaw rate limitations must meet the following formula:

$$U_S=\{(u_{a,}\omega_r)\,|\,0\le u_a\le u_{a\max},\ -\omega_{r\max}\le\omega_r\le\omega_{r\max}\} \tag{14}$$

It can be considered that the robot's traveling trajectory is composed of n broken line segments with n time frames, and it is believed that the connection point between broken line segments is close to the obstacle position to the greatest extent on the premise of meeting the expansion size limitation of the obstacle. To protect the robot from colliding with any obstacle during the movement, it can be obtained the speed set after time dt must meet the following formula according to the limitation of kinematic conditions:

$$U_a=\{(u_a,\omega_r)\,|\,u_a\le\sqrt{2\cdot dist(u_a,\omega_r)\cdot\dot{u}_a},\omega_r\le\sqrt{2\cdot dist(u_a,\omega_r)\cdot\dot{\omega}_r}\} \tag{15}$$

where $dist(u_a,\omega_r)$ is the straight-line distance between the robot and the obstacle at the next moment.

Assuming that the speed set of the robot at the present moment is $u_{acurr}+\omega_{a\max}$, then the speed set U_d at the next moment must satisfy the following condition:

$$U_d=(u_{ad},\omega_{rd})=\begin{cases}u_{acurr}-u_{a\max}d_t\le u_a\le u_{acurr}+u_{a\max}dt\\ \omega_{rcurr}-\omega_{r\max}d_t\le\omega_r\le\omega_{rcurr}+\omega_{r\max}dt\end{cases} \tag{16}$$

The final speed set U can be expressed as below:

$$U=U_S\cap U_a\cap U_d \tag{17}$$

The speed set at the next moment is predicted through the objective function. The objective function defined in this study comprehensively considers the movement speed, driving direction and collision safety, as follows:

$$G(u_{ad},\omega_{rd})=l\cdot\theta+m\cdot dist(u_a,\omega_r)+n\cdot u_{a\max} \tag{18}$$

where θ represents the included angle between the robot's driving direction and the target line; $dist(u_a,\omega_r)$ indicates the shortest distance between the robot position and the obstacle; l, m, and n are the angle, distance and vehicle speed weight coefficients, respectively.

A greater value of the objective function indicates the greater excellence of the speed set. During the path planning process, the robot needs to acquire position and speed information from multiple sensors, while these signals are usually not always continuous, and the evaluation result may have a specific error. To reduce the error, the above 3 weight coefficients are generally normalized into a number within [0, 1].

Out of the above considerations, an improved ACO-DWA algorithm was proposed in this study based on the advantages of the ACO algorithm in global optimization and continuous convergence of search process. The weight coefficient of the objective function was made self-adaptive by the ACO algorithm. This algorithm significantly reduces the bypassing distance and path planning time of the robot and improves the through capacity and safety in scenarios with dense obstacles. Hereby the specific dynamic updating process of the above weight coefficients according to the ACO algorithm will be detailed.

DAW algorithm improvement fusing ACO algorithm

It is assumed that at time t, there are obstacles of a certain density in the area of the robot's driving direction. If the number of obstacles in this area is *K*, the shortest distance between the robot and the *i*-th obstacle is D_i and the azimuth angle is θ_i. When *M* is greater than the threshold, this area is distributed with dense obstacles. The shortest distance D_{ij} between the *i*-th obstacle and the *j*-th obstacle is defined as follows:

$$D_{ij} = \sqrt{D_i^2 + D_j^2 - D_i D_j \cos(\theta_i - \theta_j)}, \theta_i >> \theta_j \tag{19}$$

Considering the safety and maneuverability of the robot when passing through obstacles and in order to measure the through capacity of the robot between two obstacles, the number of its pass-through functions D_S is defined as below:

$$D_s = a \cdot \frac{\theta_{\max}}{\omega_{r\max}} + b \cdot \frac{u_{\max}}{\dot{u}_a} \tag{20}$$

where $\omega_{r\max}$ is the maximum value in ω_r ; $\theta_{\max}$ is the maximum value in θ ; coefficient a reflects the influence of the robot's deviation in direction on the through capacity and it is taken as 0.6 according to the modeling experience of grid maps. Coefficient b indicates the influence of the robot's speed on the through capacity, and it is taken as 0.4 according to the modeling experience of grid maps, too.

Introducing the expansion radius σ of obstacles, the condition for the robot to pass through two obstacles safely is:

$$D_s > \frac{D_{ij}}{\sigma} \tag{21}$$

where σ is set to 0.3 according to the modeling experience of grid maps. The updating model for dynamic pheromone is as follows:

$$\Delta\tau_{ij}^m = \begin{cases} \dfrac{\sigma \cdot D_{s\max} - D_{ij}}{D_{ij} - \sigma \cdot D_{s\min}}, \delta > \varepsilon \\ \dfrac{D_{ij} - \sigma \cdot D_{s\max}}{D_{ij} - \sigma \cdot D_{s\min}}, \delta \le \varepsilon \end{cases} \tag{22}$$

where $\delta = D_{\max} - D_{\min}$, in which e is the acceptable error in the *n*-th iteration, being a constant; D_{max} refers to the maximum number of pass-through functions of the robot when moving between any two obstacles after traveling to the local obstacle avoidance area; D_{min} is the minimum number of pass-through functions of the robot when moving between any two obstacles after traveling to the local obstacle avoidance area.

Algorithm flow

Step 1: The starting point and target point of the robot's movement are positioned after acquiring the environmental map information through a laser radar; the information of all state nodes in space are acquired and the adjacency matrix and heuristic information matrix are calculated;

Step 2: Parameter initialization. The number of iterations is initialized as *N*, the ant colony scale as *M*, the information heuristic factor as α , the expected heuristic factor as β , pheromone evaporation coefficient as ρ , and the pheromone concentration as τ ;

Step 3: The density of obstacles, the actual distance between the robot and each obstacle and their orientations are calculated in a real-time manner;

Step 4: Path selection and updating. The adjacency matrix is inquired, the feasible node set for the robot to move from the current node i to the next node is acquired, and the probability for the m-th ant to select adjacent nodes is solved. Whether the ant enters an area with dense obstacles is judged with such information as the density of obstacles, the actual distance between the robot and each obstacle and their orientations according to the captured map information during node updating; if yes, turn to Steps 5-7; if not, all weight coefficients in Step 5 are made constant;

Step 5: The node set greater than the threshold $dist(u_a, \omega_r)$ is eliminated from the planned global path nodes, the ant number is updated, the distance between obstacles and the number of the robot's pass-through functions D_S are calculated via the vehicle-mounted processor, the values of D_{max} and D_{min} are acquired, the pheromone is calculated and updated, and the dynamically updated weight coefficients l, m and n are obtained;

Step 6: The alternative speed space $U(u_a, \omega_r)$ is solved, and the 3 weight coefficients are respectively normalized to obtain the optimal speed set of the robot at time *t+1;*

Step 7: This speed is executed, and whether the target point is reached is judged; if yes, end the iteration process; if not, return to Step 1.

RESULTS

Experimental environment and parameter settings

A square area (side length: 30m×30m) was chosen as the experiment site for the agricultural warehouse handling robot, which was subjected to grid partitioning. The layout of shelves and the scene picture are displayed in Fig. 1.

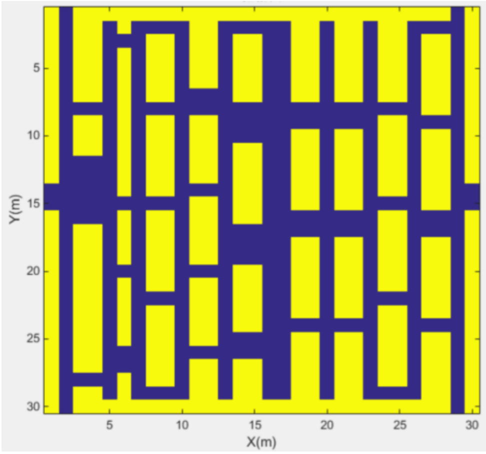

Fig. 1 - Warehouse site diagram based on grid method arrangement

In Fig. 2, the yellow cube indicates the immovable shelf, and the blue area is the passage. The length, width and height of the agricultural handling robot used in the experiment are 0.7, 0.5 and 0.4 m respectively, and the warehouse logistics handling robot itself has the telescopic lifting function, with the maximum bearing capacity of 100 kg.

At the same time, the robot is also equipped with a set of sensor suites, including a wireless communication module and a visual radar system, which are used to receive the background instruction information, deal with emergencies and avoid collisions. And the best travel path was chosen using the improved ACO-DWA algorithm from the global perspective. The speed and angular velocity of the agricultural handling robot are 2 m/s and 2.5 rad/s, respectively, the maximum number of iterations of the improved ACO-DWA algorithm is 800, and the value range for the core parameter of the policy gradient algorithm is 0.7-2.2. The obstacle avoidance path for the wheeled agricultural warehouse handling robot acquired through the improved ACO-DWA algorithm is exhibited in Fig. 2.

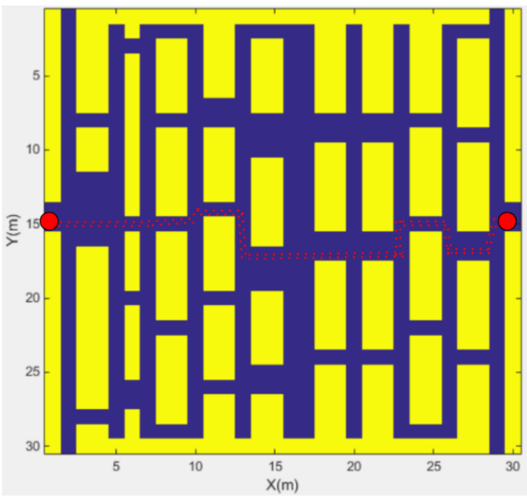

Fig. 2 - The travel path selected improved by ACO-DWA algorithm

In Fig. 3, the improved ACO-DWA algorithm proposed in this study planned the travel path from the perspectives of global planning and collision avoidance of moving obstacles, and chose a more economical path; the number of iterations of the improved ACO-DWA algorithm in global optimization was analyzed, and the algorithm iteration efficiency was higher under general circumstances, proving that the algorithm possesses stronger data training ability. If the number of algorithm iterations was too great and even exceeded the maximum number of iterations, the algorithm would be easily stuck in local optimum, failing to realize the optimization within a global scope. In the process of path optimization, the maximum number of iterations of each algorithm was set to 800, the improved ACO-DWA algorithm gained the optimal result, and the agricultural handling robot turned for 10 times in the warehouse, with a driving distance of 42 m.

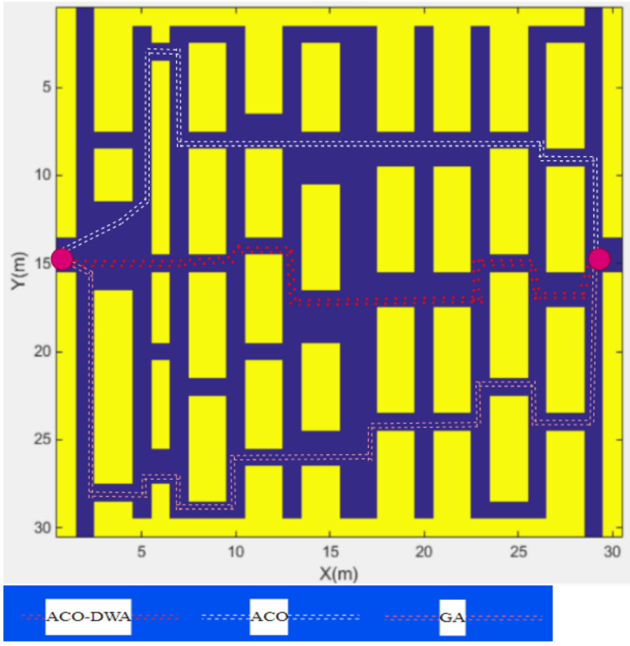

Fig. 3 - The travel path selected by each algorithm

Comparison of experimental data

To verify the improved ACO-DWA algorithm proposed in this study, path planning was performed from two angles: global planning and collision avoidance of moving obstacles, and a relatively economical path was chosen; as for the traditional ACO algorithm and genetic algorithm, turning occurred for 9 and 16 times, respectively, for the sake of avoiding other warehouse handling robots, which affected the overall travel speed. The paths obtained by the 3 algorithms for completing this handling task are exhibited in Figure 3.

Since the improved ACO-DWA algorithm selected a near path and can perceive and judge other moving obstacles around based on policy gradient and plan a new local path in advance, it spent the shortest time; for the other two traditional path planning algorithms, retracing occurred for avoiding obstacles, which influenced the final time consumption.

From the convergence speed of the algorithm, the convergence time of the improved ACO-DWA algorithm returned to zero after the 413th iteration, while the ACO algorithm and genetic algorithm completed the convergence only after approaching the maximum number of iterations, being 726 and 755, respectively. The iterative performance of the algorithm determines the efficiency of the algorithm and also has an important impact on the obstacle avoidance ability of the algorithm for selecting the optimal travel path. The experimental data showed that when a relatively complicated agricultural warehouse handling scenario, turning took place for 9 and 16 times, respectively, under obstacle avoidance planning based on the traditional ACO algorithm and genetic algorithm; under the control of the improved ACO-DWA algorithm, the policy gradient-based algorithm effectively avoided collisions during the simultaneous operation of multiple warehouse handling robots, and turning only occurred for 10 times. Besides, the policy gradient-based algorithm would adjust the direction of motion and speed of the robot timely according to the local obstacle distribution on site, which not only ensured traveling according to the path planned by the improved ACO-DWA algorithm as a whole but also effectively realized the avoidance of local moving obstacles.

CONCLUSIONS

In the dynamic warehouse environment, the obstacle avoidance control method of agricultural warehouse handling robots based on visual positioning and dynamic rectification is an efficient, accurate and highly adaptable method, which is of great practical significance for strengthening warehousing operation efficiency and safety. Visual positioning technology enables the robot to acquire and process environmental information in real time, and avoid obstacles through the real-time rectification of the robot's trajectory. With the increasing complexity of the working environment for warehouse handling robots, robots should not only avoid static obstacles but also evade other robots on the path, that is, multiple robots working simultaneously are dynamic obstacles. In the path planning of logistics robots, the overall path of robots should be planned first, and the dynamic obstacles in local areas should be avoided by dynamic real-time rectification, which proposes higher requirements for the real-time communication of robots and the real-time selection of local paths. Therefore, the dynamic real-time rectification and obstacle avoidance based on the spatial coordinate transformation of robots will become one of the main development trends of warehouse logistics robots in the future.

In this study, a path planning method for wheeled agricultural warehouse handling robots was proposed. The improved DWA algorithm integrating the ACO algorithm can effectively improve the operation efficiency of the algorithm, reduce the bypassing distance around obstacles, shorten the path planning time and enhance the driving safety. The effectiveness and universality of the improved ACO-DWA algorithm were verified through simulation tests. This method evidently accelerates the optimization and convergence rate of the global path for agricultural warehouse handling robots. The applicability of the algorithm under scenarios with dynamic obstacles will be further explored on this basis.

REFERENCES

[1] Dang, X. (2021). Automatic obstacle avoidance control method of packaging robot based on gravitational potential field (基于引力势场的包装机器人自动避障控制方法). *Packaging Engineering*, Vol.42, pp. 254-259, Chongqing/China.

[2] Ding, H. W., Cao, X. G., Wang, Z. S., Dhiman, G., Hou, P., Wang, J., Li, A.S., & Hu, X. (2022). Velocity clamping-assisted adaptive Salp swarm algorithm: balance analysis and case studies. *Mathematical Biosciences and Engineering*, Vol. 19, pp. 7756-7804, United States.

[3] Guo, H. Q., Fu, L. X., Zhang, Y., & Mao, J. L. (2022). Robot obstacle avoidance path planning method based on dynamic obstacles (基于动态障碍物的机器人避障路径规划方法). *Video Engineering*, Vol. 46, pp. 73-77, Beijing/China.

[4] Hao, K., Zhang, H. J., Li, Z. S., & Liu, Y. L. (2022). Path planning of mobile robot based on improved obstacle avoidance strategy and double optimization ant colony algorithm (基于改进避障策略和双优化蚁群算法的机器人路径规划). *Transactions of the Chinese Society for Agricultural Machinery*, Vol. 58, pp. 303-312, Beijing/China.

[5] Hichri, B., Gallala, A., Giovannini, F., & Kedziora, S. (2022). Mobile robots path planning and mobile multirobots control: a review. *Robotica*, Vol. 40, pp. 4257-4270, United Kingdom.

[6] Hossain, T., Habibullah, H., Islam, R., & Padilla, R. V. (2022). Local path planning for autonomous mobile robots by integrating modified dynamic-window approach and improved follow the gap method. *Journal of Field Robotics*, Vol. 39, pp. 371-386, United States.

[7] Ji, X. Y., Feng, S., Han, Q. D., Yin, H. F., & Yu, S. W. (2021). Improvement and fusion of A∗ algorithm and dynamic window approach considering complex environmental information. A*rabian Journal for Science and Engineering*, Vol. 46, pp. 7445-7459, Saudi Arabia.

[8] Kobayashi, M., & Motoi, N. (2022). Local path planning: dynamic window approach with virtual manipulators considering dynamic obstacles. *IEEE Access*, Vol. 10, pp. 17018-17029, United States.

[9] Li, J., & Qin, W. (2021). Design of mobile robot obstacle-avoidance control system based on vision system (基于视觉的移动机器人避障控制系统设计). *Machine Tool & Hydraulics,* Vol. 49, pp. 24-28, Guangdong/China.

[10] Ornek, B. N., Aydemir, S. B., Duzenli, T., & Ozak, B. (2022). A novel version of slime mould algorithm for global optimization and real world engineering problems. *Mathematics and Computers in Simulation*, Vol. 198, pp. 253-288, Netherlands.

[11] Qian, Q. W., Wu, J. F., & Wang, Z. (2019). Optimal path planning for two-wheeled self-balancing vehicle pendulum robot based on quantum-behaved particle swarm optimization algorithm. *Personal and Ubiquitous Computing*, Vol. 23, pp. 393-403, Germany.

[12] Sarkar, R., Barman, D., & Chowdhury, N. (2022). Domain knowledge based genetic algorithms for mobile robot path planning having single and multiple targets. *Journal of King Saud University-Computer and Information Sciences*, Vol. 34, pp. 4269–4283, Saudi Arabia.

[13] Tan, Z., Liang, H. G., Zhang, D., & Wang, Q. G. (2022). Path planning of surgical needle: a new adaptive intelligent particle swarm optimization method. *Transactions of the Institute of Measurement and Control*, Vol. 44, pp. 766-774, United Kingdom.

[14] Wang, Z., Ding, H., Wang, J., Hou, P., Li, A., Yang, Z., & Hu, X. (2022). Adaptive guided Salp swarm algorithm with velocity clamping mechanism for solving optimization problems. *Journal of Computational Design and Engineering*, Vol. 9, pp. 2196-2234, Netherlands.

[15] Wang, Z., Ding, H., Yang, Z., Li, B., Guan, Z., & Bao, L. (2022). Rank-driven Salp swarm algorithm with orthogonal opposition-based learning for global optimization. *Applied Intelligence*, Vol. 52, pp. 7922-7964, Netherlands.

[16] Zhai, L. Z., & Feng, S. H. (2022). A novel evacuation path planning method based on improved genetic algorithm. *Journal of Intelligent and Fuzzy Systems*, Vol. 42, pp. 1813-1823, Netherlands.

[17] Zhang, L., Zhang, Y. J., & Li, Y. F. (2021). Mobile robot path planning based on improved localized particle swarm optimization. *IEEE Sensors Journal*, Vol. 21, pp. 6962-6972, United States.

[18] Zheng, J., Gao, Y., Lei, Z., Yang, C. J., Wang, C., & Oderman, G. (2021). Real time search of agricultural machinery based on matrix sequence sensor. *INMATEH-Agricultural Engineering*, Vol. 4, pp. 425-433, Romania.

[19] Zohaib, M., Pasha, S. M., Javaid, N., Salaam, A., & Iqbal, J. (2014). An improved algorithm for collision avoidance in environments having U and H shaped obstacles. *Studies in Informatics and Control*, Vol.23, pp. 97-106, Romania.