

Ship Path Planning in Ice Based on an Ant Colony-Dynamic Window Hybrid Algorithm

Yanzhuo Xue¹, Xinyue Zhang¹, Wenbo Liu², Yang Lu¹, Menghang Liu¹ and Yuejun Liu¹

Received: 22 March 2025 / Accepted: 30 April 2025

© Harbin Engineering University and Springer-Verlag GmbH Germany, part of Springer Nature 2026

Abstract

In this study, a hybrid path planning algorithm for ships in ice areas is proposed. First, the operation space of ships in ice areas is established in accordance with the raster method. Second, improvement strategies are introduced to address the slow iteration speed and poor solution quality of the traditional ant colony algorithm, and the Ant Colony Optimization (ACO) algorithm is used to plan the global path in a static environment. Then, the node deletion algorithm is proposed to optimize the global path. Finally, the dynamic window approach (DWA) algorithm is optimized, and the subnodes of the preprocessed path are considered the subobjectives to be optimized. The final path in the dynamic environment is solved using the improved DWA algorithm. The simulation results show that: the ACO algorithm can plan a high-quality global path with a high iteration speed, the node deletion algorithm can effectively eliminate redundant nodes and reduce the computational complexity of the subsequent algorithms, and the improved DWA algorithm can efficiently avoid dynamic obstacles in the environment on an original global path and solve the path that meets the motion performance of the ship in ice areas.

Keywords Polar navigation; Path optimization; Ant colony algorithm; Dynamic window approach; Hybrid algorithm

1 Introduction

1.1 Background

The greenhouse effect has led to increased global warming and the melting of Arctic ice at an ever-increasing rate. Based on long-range statistical projections, Diebold and Rudebusch (2022) suggest that there is a 60% probability that the Arctic Ocean will be ice-free in the 2030s. Ships serve as an important tool for exploring, developing, and governing the Arctic. Rapidly declining Arctic sea ice has improved navigation in the Arctic shipping lanes (Wang et al., 2024); however, the waters remain fraught with dan-

gers (Qian et al., 2020), such as extreme climatic conditions, complex hydrographic environments, and unstable ice conditions. Especially, besetting of ships in ice or ship–ice collision may lead to ship damage, property loss, environmental hazards, and casualties (Liu et al., 2024). Therefore, the planning of practical and reliable ship paths plays an important role in promoting the sustainable development of polar resources.

1.2 Path planning algorithm

With the continuous development of artificial intelligence and Internet of Things technology, path planning algorithms have transitioned into one of the focal topics in the field of control research, with applications covering numerous fields, such as robot navigation, autonomous driving, aerospace, and navigation.

1.2.1 Global path planning algorithm

Global path planning algorithms are suitable for the selection of optimal paths based on certain evaluation metrics in a wide range of environments.

Charalambopoulos et al. (2023) used the improved Dijkstra's algorithm to plan ship paths under environmental conditions and verified the feasibility of applying the algorithm in the Aegean archipelagos and the Mediterranean Sea. Based on the genetic algorithm, Zhang and Gao (2024) introduced new operators, optimization strategies, and heuristic strategies and used the improved algorithm to solve

Article Highlights

- An improved ant colony algorithm is introduced for global path planning, complemented by a node deletion algorithm for post-processing optimization.
- An improved dynamic window approach algorithm is developed for local path planning and dynamic obstacle avoidance
- A hybrid path planning algorithm for ships in ice areas is proposed, integrating the above two algorithms

✉ Yang Lu
luyang@hrbeu.edu.cn

¹ College of Shipbuilding Engineering, Harbin Engineering University, Harbin 150001, China

² Design Institute of Offshore Oil Engineering Co., Ltd, Tianjin 300451, China

the multi-objective optimization problem, and the results showed that the improved algorithm reduces the search space while improving efficiency. Huo et al. (2024) introduced the distance and steering-angle penalty factors on the basis of the ant colony algorithm (ACA) and enhanced the updating mechanism of pheromone, and the findings revealed that the algorithm can plan the paths to satisfy the motion constraints at a faster speed.

1.2.2 Local path planning algorithm

The local path planning algorithm exhibits better performance for unknown and dynamic obstacles. The current position of the moving target and the state of the nearby environment usually need to be considered in the solution.

Sun et al. (2024) proposed the artificial potential field (APF) grid method to solve the problems of local oscillation and susceptibility to local minima by modifying the potential field function on the mesh surface and introducing virtual target points; in addition, they planned the path of an unmanned underwater vehicle (UUV). Li et al. (2024) used the collision avoidance dynamic parameters between ships as the main indexes to create a repulsive potential field and designed a collision-avoidance decision-making scheme based on the nonlinear model predictive control

algorithm. Song et al. (2024) enriched the knowledge maps (KM) based on the International Regulations for Preventing Collisions at Sea and then proposed a KM-dynamic window approach (DWA) algorithm, which renders the moving target with a collision avoidance buffer, and verified the effectiveness of the improved algorithm by comparing it in single- and multivessel scenarios.

1.2.3 Hybrid path planning algorithm

The single path planning algorithm has many limitations and is more practical to use than a hybrid path planning algorithm.

Cai et al. (2024) improved the initial pheromone distribution, heuristic function, and pheromone update method to prevent the Ant Colony Optimization (ACO) algorithm from falling into local optima and combined the algorithm with the improved APF method to improve its performance. The problem of the ACO algorithm failing to converge was solved. Tang et al. (2024) proposed a dynamic window model prediction APF algorithm to address the poor overlap between the actual and predicted trajectories of underactuated planning craft; the algorithm can better match the trajectory of the target and is suitable for path planning in obstructed waters.

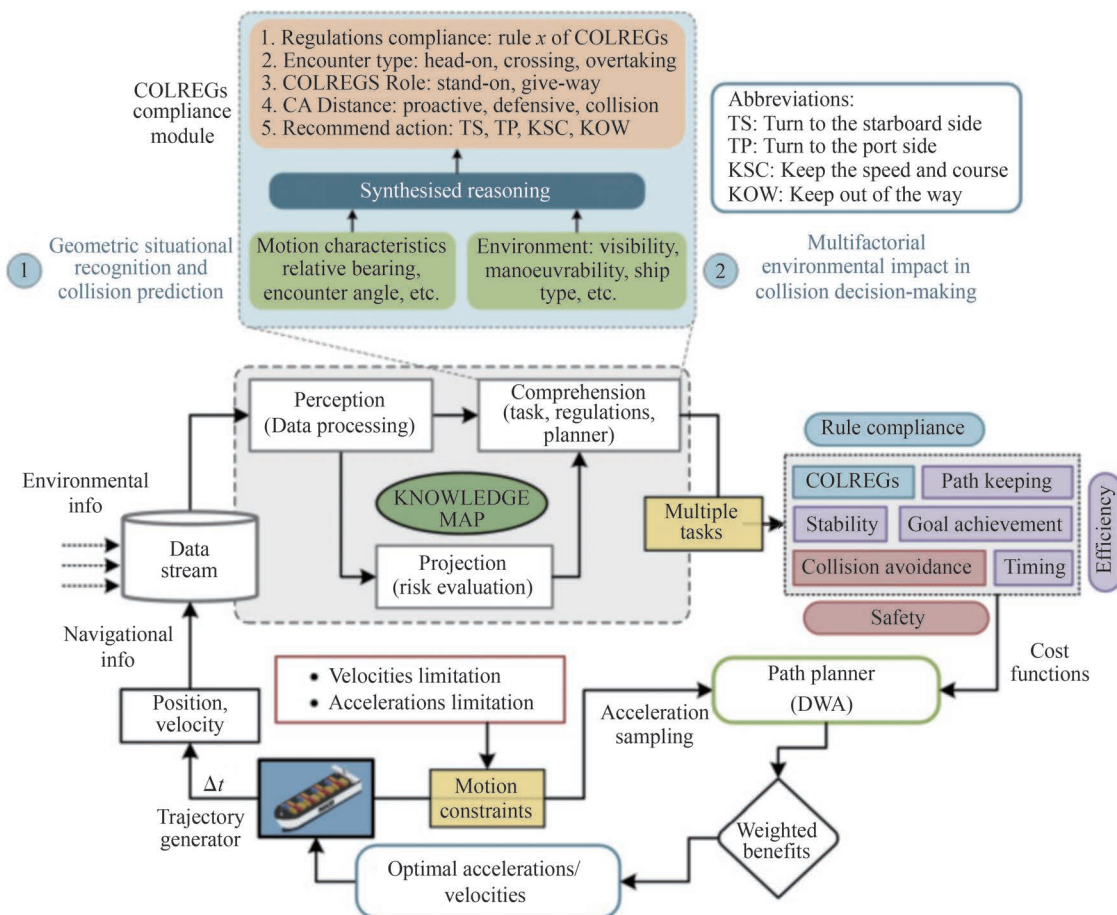


Figure 1 Diagram of the KM-DWA decision-making framework based on the KM and DWA algorithm (Song et al., 2024)

1.3 Path planning of a ship in ice

The field of ship path planning in ice areas mainly focuses on the application of complex and variable ice conditions and various optimization algorithms. Shu et al. (2023) established the objective function regarding navigation efficiency, navigation cost, icebreaking cost, and safety in breakable and unbreakable sea ice conditions and solved the multi-objective optimization functions using dynamic programming and numerical simulation. Zhang et al. (2022) developed a multi-objective ice routing model while considering time-varying ice data and proposed a three-dimensional ACA to determine a ship's passing waypoints and the sailing speeds along each waypoint. Choi et al. (2015) developed a stochastic path-optimization model for the stochastic domain in ice thickness and concentration and proposed an uncertainty-based path planning model for determining an optimal path under time-varying stochastic conditions. Lee et al. (2021) proposed a ship path planning optimization model considering the risk assessment of ice regime. The ice conditions were assessed based on the polar operational limit assessment risk indexing system; the arrival time, avoidance of land, criteria for the ice regime, and restriction of design variables were considered constraints in the search for the optimal path using the seed genetic algorithm.

From the above papers, for the shortcomings of global and local path planning algorithms, scholars have put forward a number of improvement strategies to broaden their application scenarios, attain a better algorithmic performance, and strengthen their robustness. However, in the face of the Arctic Sea as a special environment, the corresponding research needs to be further developed.

1.4 Motivation and contribution

In this paper, a hybrid ACO-DWA algorithm is proposed to provide a decision for path planning of ships in ice areas, and its main contributions include the following:

- 1) The traditional ACA has problems, such as slow convergence and poor solution quality in high-complexity models. Four improvement strategies are proposed, and the performance of the improved algorithm in static environment models of different levels of complexity is verified.
- 2) Restricted by the search limitation of the ACA, several redundant nodes are found in the global path. This paper proposes a node deletion algorithm aimed at deleting redundant paths.
- 3) With the nodes of the preprocessed routes considered the sub-objectives to be optimized, the improved DWA algorithm is used to plan a path that meets the motion performance of ships in the ice area.

The remainder of this paper is organized as follows. Section 2 establishes the environment model applied to the subsequent algorithms. Section 3 introduces the basic models of the ACA and DWA algorithms. Section 4 describes

the improvement strategies of the algorithms and the execution framework of the hybrid algorithm. Section 5 presents the results and analyses of simulations. Section 6 concludes the paper with a summary and outlook.

2 Polar environment model

Based on ice conditions in the navigation area and the ice class of the ship, the obstacles to ship navigation are classified into two categories: static and dynamic.

2.1 Static environment model

The navigation process of ships in an ice area can be regarded as the motion of a mass point in a two-dimensional plane. The raster method is implemented to establish the environmental model. The environmental information on the Arctic is projected in a plane, and the grid is divided by selecting the appropriate accuracy. In the model, real glaciers, ice caps, tundra, etc., are mapped as black static obstacles and the navigable area as a blank part (Figure 2).

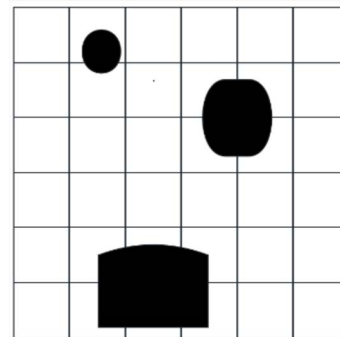


Figure 2 Example of the projection of environmental information in the model

For navigational safety, the hybrid grid needs to be converted into an obstacle grid, and its positional indexes are identified through the ordinal method, with increments from left to right and from top to bottom (Figure 3).

If the grid consists of L rows and columns, and each grid cell has a side length of 1 unit, then the coordinates of the center point of the grid cell with serial number N can be calculated as follows:

$$\begin{cases} x = \text{mod}(N, L) - 0.5 \\ \text{if } x = -0.5, \text{ then } x = L - 0.5 \\ y = L + 0.5 - \text{ceil}\left(\frac{L}{N}\right) \end{cases} \quad (1)$$

2.2 Dynamic environmental model

Under the influence of wind, waves, and currents, the Arctic experiences dynamic obstacles, such as moving layer ice

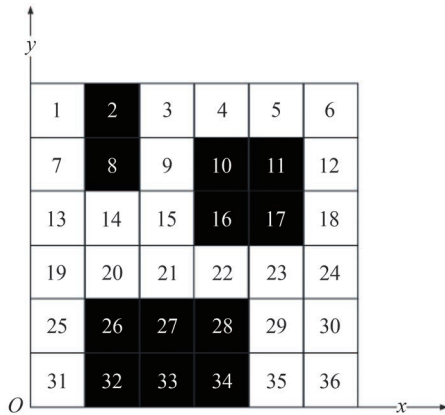


Figure 3 Conversion of the hybrid grid into an obstacle grid based on Figure 2

and drift ice groups. The state and position of the dynamic obstacles at moment t can be expressed as follows: the length of dynamic obstacles L_d , width of dynamic obstacles W_d , center-of-mass coordinates x_d^t along the axis X at moment t , center-of-mass coordinates y_d^t along the axis Y at moment t , course angle θ_d^t , linear velocity v_d^t , and angular velocity ω_d^t at the current moment:

$$s_d^t = (L_d, W_d, x_d^t, y_d^t, \theta_d^t, v_d^t, \omega_d^t) \quad (2)$$

Based on the various types of motion, the dynamic obstacles considered in this paper are divided into linear and circumferential motion models. The linear motion model constantly moves along a straight line with a constant velocity and an angular velocity equal to 0. The circumferential motion model consistently moves with a constant linear velocity and angular velocity.

3 Models of algorithms

3.1 Ant colony optimization

Ant colony algorithm refers to a bionic algorithm inspired by the group collaboration among ants through pheromone exchange. As an intelligent multisubject system, the operation mechanism of ACA mainly includes the following: distributed operation, self-organization, and positive feedback regulation. In each iteration, the ACA sends out M ants to select a path based on the heuristic information and pheromones of the current location. The probability that the k th ($1 \leq k \leq M$) ant proceeds to node j from node i at moment t in each iteration is denoted as $P_{ij}^k(t)$:

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha (\eta_{ij})^\beta}{\sum_{j \in a} [\tau_{ij}(t)]^\alpha (\eta_{ij})^\beta}, & j \in a \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

$$\eta_{ij} = \frac{1}{d_{ij}} \quad (4)$$

where $\tau_{ij}(t)$ refers to the pheromone content of the k th ant from node i to node j at moment t , η_{ij} denotes the desirability from node i to node j , α indicates the parameter of the pheromone to control the influence of $\tau_{ij}(t)$, β means the parameter of the desirability to control the influence of η_{ij} , a indicates the set of nodes that are next reachable by the k th ant at moment t .

The pheromone needs to be updated when all ants have completed the path search in a certain iteration. The pheromone updating mechanism is as follows: A part of the original pheromone is evaporated, and a part is added:

$$\tau_{ij}(t+1) = (1 - \rho) \tau_{ij}(t) + \Delta \tau_{ij}(t) \quad (5)$$

$$\Delta \tau_{ij}(t) = \sum_{k=1}^M \Delta \tau_{ij}^k(t) \quad (6)$$

where ρ denotes the volatilisation coefficient, indicates the rate of pheromone evaporation, and $\Delta \tau_{ij}(t)$ means the amount of pheromone from node i to node j released by the ants at moment t .

Based on the difference in $\Delta \tau_{ij}^k(t)$, the pheromone updating model can be classified into three types: ant-cycle, ant-quantity, and ant-density models. The ant-cycle model, which is a global updating mechanism, is more suitable for solving the global search problem. Its updating mechanism is as follows:

$$\Delta \tau_{ij}^k(t) = \begin{cases} \frac{Q}{L_k}, & \text{the ant passes through the path } (i, j) \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

where Q refers to the increasing strength coefficient of the pheromone, and L_k indicates the path length, which is sought by the k th ant in that iteration. Figure 4 shows the basic flowchart of the ACA.

3.2 Dynamic window approach

The DWA algorithm is a suboptimal method based on predictive control theory, with the aim of path planning for real-time obstacle avoidance. The core idea of this algorithm is the transformation of a problem into sub-problems, which is to be optimized based on the nonholonomic constraints, dynamics constraints, and local environment information of the motion model. Deriving the local path, moving the motion target, and updating the environment information, the algorithm combines the optimization and feedback process. The motion target moves toward the end point while avoiding obstacles by periodically executing the above steps.

The common models used in this algorithm are omnidi-

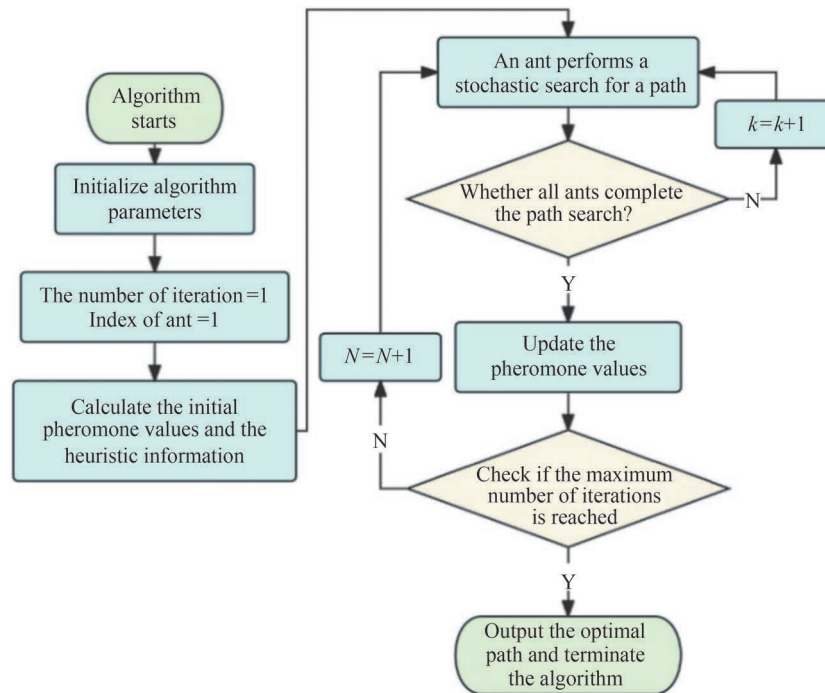


Figure 4 Basic flowchart of the ant colony algorithm

rectional and nonomnidirectional motion models. The former can generate angular velocity around the axis and linear velocities in two directions, but it cannot accurately simulate the movement of ships. Therefore, we select the 2-degree-of-freedom (DoF) nonomnidirectional motion model to simulate the acceleration and deceleration and the rudder control behavior of the ship, which enables the generation of surging and yawing (Figure 5).

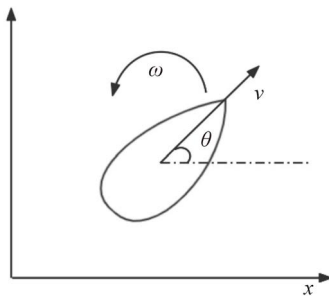


Figure 5 2-DoF nonomnidirectional motion model

The state after a short period Δt is as follows:

$$\begin{cases} x' = x + v\Delta t \cos\theta \\ y' = y + v\Delta t \sin\theta \\ \theta' = \theta + \omega\Delta t \end{cases} \quad (8)$$

Once the state of the target is known, the subsequent path can be deduced using the linear and angular velocities. A sampleable velocity space can be generated by sub-

jecting the target to the kinematic properties of the target and obstacles in the space in which it is located.

A formula for the subsection to maximum and minimum velocity constraints is defined as follows:

$$V_m = \{v \in [v_{\min}, v_{\max}], \omega \in [\omega_{\min}, \omega_{\max}]\} \quad (9)$$

A functional formula for subsection to the maximum acceleration achievable by the motion target is as follows:

$$V_a = \left\{ \begin{aligned} v_{t+1} &\in [v_t - \dot{v}\Delta t, v_t + \dot{v}\Delta t] \\ \omega_{t+1} &\in [\omega_t - \dot{\omega}\Delta t, \omega_t + \dot{\omega}\Delta t] \end{aligned} \right\} \quad (10)$$

The functional formula below is used when the target is subjected to the space of the environment in which it is located:

$$V_d = \left\{ \begin{aligned} v &\leq \sqrt{2 \times d(v, \omega) \times \dot{v}} \\ \omega &\leq \sqrt{2 \times d(v, \omega) \times \dot{\omega}} \end{aligned} \right\} \quad (11)$$

where $d(v, \omega)$ denotes the distance from the simulated target path to the nearest obstacle at the current velocity (v, ω) .

Thus, the sampleable velocity space is an intersection computed as follows:

$$V_c = V_m \cap V_a \cap V_d \quad (12)$$

The sampleable velocity space can be discretized based on the linear and angular velocity resolutions. The path after the corresponding predicted simulation time T can be solved based on the model for each velocity (v_i, ω_i) , and the optimal sampling point can be selected depending on the score of the evaluation function.

The algorithm expects the motion target to move toward the aim along a safe path as fast as possible, and thus, the evaluation function is defined as follows:

$$G = w_1 H(v, \omega) + w_2 D(v, \omega) + w_3 V(v, \omega) \quad (13)$$

where $H(v, \omega)$ refers to the evaluation subfunction of the course angle for the simulation of the angle between the connecting line from the end position of the path to the target and the course; $D(v, \omega)$ denotes the evaluation subfunction of distance for the simulation of the distance between the end position of the path and the nearest obstacle, whose value is not more than the threshold parameter L ; $V(v, \omega)$ indicates the evaluation of the subfunction of velocity, whose numerical value is equal to the sampling linear speed. The highest scoring path can be selected by weighting coefficients after normalizing each subfunction.

Figure 6 shows the basic flowchart of the DWA algorithm.

4 Path planning framework in polar marine environments

In this chapter, a ship path planning model for the ice area is proposed based on the hybrid algorithm. The model con-

siders the kinematic performance, characteristics of the working environment, and the time and safety cost comprehensively.

4.1 Global path planning with advanced ACO

Although the ACA exhibits high reliability and robustness, the search at the start of iteration easily falls into the local optima and is inapplicable to complex scenes. Based on the above problems, four improvement strategies are proposed in this work.

4.1.1 Improvement of the initial pheromone distribution

In the traditional ACA, no difference exists in the initial pheromone distribution on various paths, and the search lacks guidance at the beginning of iteration. The APF functions by creating a virtual force field that acts on a moving target, which in turn creates a tendency for the target to proceed toward an attractive force. Inspired by the APF, the initial pheromone distribution can be adjusted based on the relative position of the feasible node to the target end point:

$$\tau'_{ij} = \tau_{ij} \times \left(1 + \frac{\frac{1}{d_{je}}}{\sum_{j \in a} \frac{1}{d_{je}}} \right) \quad (14)$$

where τ_{ij} denotes the original initial pheromone, τ'_{ij} corresponds to the improved initial pheromone, a indicates the set of next feasible nodes, and d_{je} represents the distance between the next feasible node and the target end point.

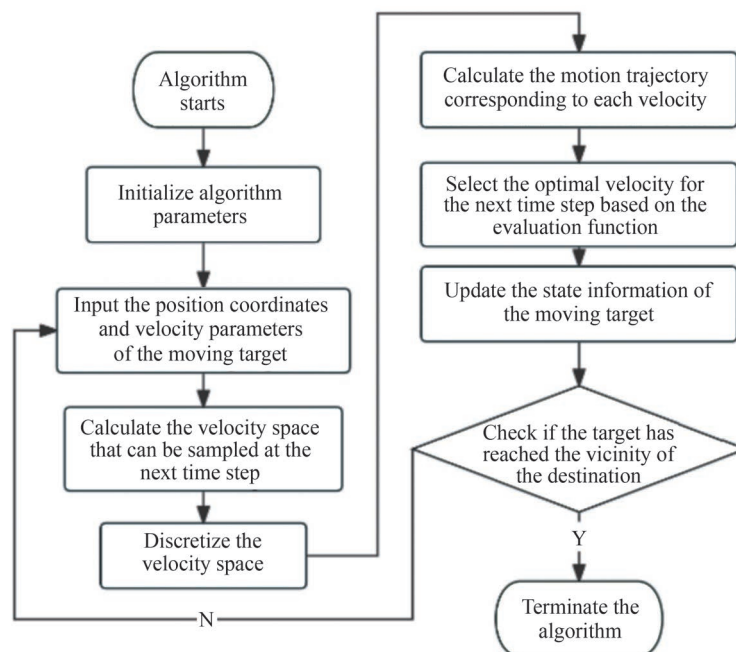


Figure 6 Basic flowchart of the DWA algorithm

Figure 7 shows the relative positions of the nodes.

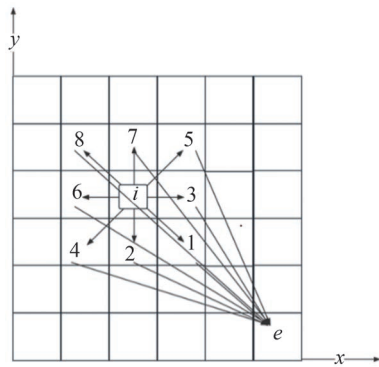


Figure 7 Relative position of the nodes

4.1.2 Design of adaptive heuristic functions

In the field of operations optimization, ACA was initially used to solve the traveler's problem, which requires the optimization model to search for the shortest loop traversing each node. Given that the prediction of the distance between the next node and the end point is not possible, the heuristic function of Eq. (4) is used to attain a better performance.

When searching for paths in the raster environment, following the traditional heuristic function will lead to an insignificant difference in the heuristic information on various optional paths and poor priori performance of the algorithm. To address this problem, we propose an improved heuristic function by introducing the positional relationship between the current node i , the next feasible node j , and the end point e :

$$\eta_{ij} = \frac{1}{d_{ij} + d_{je}} \quad (15)$$

The improved heuristic function enhances the purposefulness of the ants' search paths. However, the differences in the heuristic information on different paths are extremely small when the map size is large and the ants are close to the starting point. Meanwhile, when the ants are closer to the end point, the over-differentiated heuristic information prompts the ants to select a single path during their convergences to the end point. Therefore, in this paper, a constant P is introduced to further optimize the heuristic function:

$$\eta_{ij} = \frac{1}{d_{ij} + d_{je} - d_{ie} + P} \quad (16)$$

4.1.3 Designing an adaptive pheromone volatilisation coefficient

In the traditional algorithm, the pheromone volatilization coefficient is a constant. If ρ is extremely small, the algorithm falls into the local optima, and conversely, an excessively large ρ leads to the reduction of the convergence

speed and operation efficiency of the algorithm. A constant pheromone volatilisation coefficient evidently cannot meet the needs of the algorithm at various stages. The pheromone volatilization coefficient is defined as follows:

$$\rho = n \frac{\lambda^N}{N!} e^{-\lambda} + m, N = 0, 1, 2 \dots \quad (17)$$

where N refers to the number of iterations, and n , λ , and m are parameters. The pheromone volatilization coefficient obeying the Poisson distribution takes a small value at the beginning of iteration, which is conducive to pheromone accumulation. The coefficient takes a larger value at the middle stage of iteration, which improves the algorithm's global searching capability, and a smaller value at the later stage, which helps the algorithm to converge quickly.

Empirical evaluations demonstrate the relatively stable performance of the adaptive volatilization strategy within reasonable parameter ranges, indicating strong robustness.

4.1.4 Optimization of pheromone updating mechanisms

Traditional ACAs consider the path that successfully reaches the end point as a feasible solution to the problem to be optimized and update the pheromones on the paths. However, this approach ignores the differences between various feasible solutions. Some ants can search for feasible paths, but the paths are longer. Updating the pheromone on longer paths will cut down the influence of the pheromone on the shorter paths, which will result in a misleading effect on the subsequent ant colony.

The wolf pack algorithm represents a swarm intelligence algorithm that simulates a wolf pack hunting. This algorithm adopts the allocation rule of "performance related", and thus, the strongest individuals can secure more resources to better continue the population. Inspired by this algorithm, this study introduces the threshold value $\mu \bar{L}$, where μ is the threshold parameter, and $\bar{L}$ represents the average of all feasible path lengths in the iteration. The paths with lengths less than $\mu \bar{L}$ are regarded as high-quality paths, and only pheromones on high-quality paths are updated after each iteration.

4.1.5 Advanced ACO framework

To solve the high-quality global paths of ships in the ice area quickly, based on the four optimization strategies, we determine the algorithm flow of the improved ACO algorithm (Figure 8). The red part is the optimized module.

4.2 Global path preprocessing

In the ACA, the target can only move toward 8 adjacent nodes in the raster environment, and thus, the path suffers from a redundancy problem (Figure 9):

If the partial path in the global route is $[\dots, a, b, c, d, e, f, \dots]$, as shown in the dashed part in Figure 9, then nodes b, c, d , and e are redundant nodes. The global route must

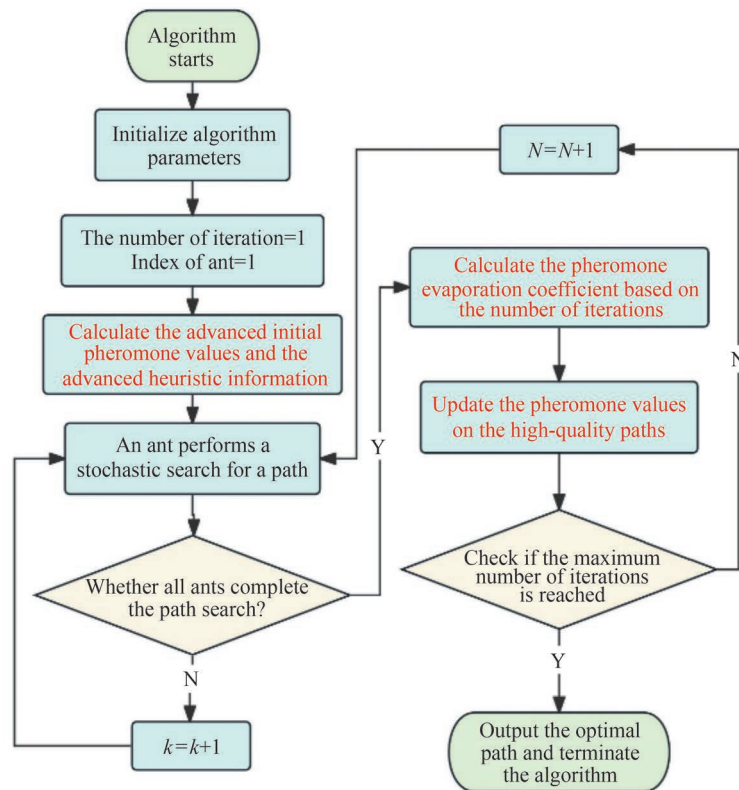


Figure 8 Algorithm flow of the improved ACO algorithm

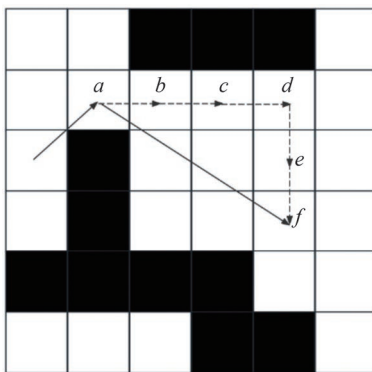


Figure 9 Path from point a to point f with a redundancy problem

be processed to extract the key nodes applied to the subsequent algorithms. The paths after preprocessing should be $[\dots, a, f, \dots]$.

The following example illustrates the method of determining redundant nodes. In a rectangle consisting of 4×3 raster, a line connects nodes a and f (Figure 10). If the distance from the center point of a grid to the connecting line af is less than half of the diagonal length of the square grid, then the grid is regarded as the proximity grid of the path af , i.e., the gray grid in Figure 10. This type of grid is called a free grid. If the adjacent grids of the path are all free grids, then the path is navigable, and the rest of the nodes between the two nodes are redundant nodes that can be deleted.

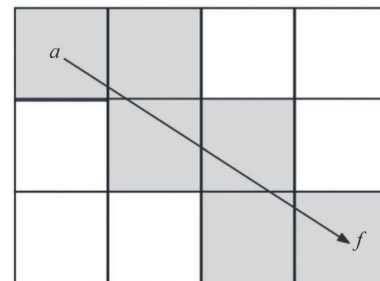


Figure 10 Determining redundant nodes in the raster environment

Figure 11 shows the node deletion algorithm flow.

In the figure, R is the global path, which is a one-dimensional array consisting of each node identified by the ordinal number method and outputted from the ACO algorithm. S denotes the initial point of the local path to be optimized, and C is the index value of the hop count of the local path to be optimized. After the traversal of the algorithm, the index values of the redundant nodes in R are stored in D and deleted together.

4.3 Local path planning with advanced DWA

The traditional DWA algorithm can solve the single-objective point planning problem. The global path after preprocessing is a polyline-path composed of multiple nodes connected in turn. For adaption to the multi-objective point optimization problem and obstacle avoidance needs of

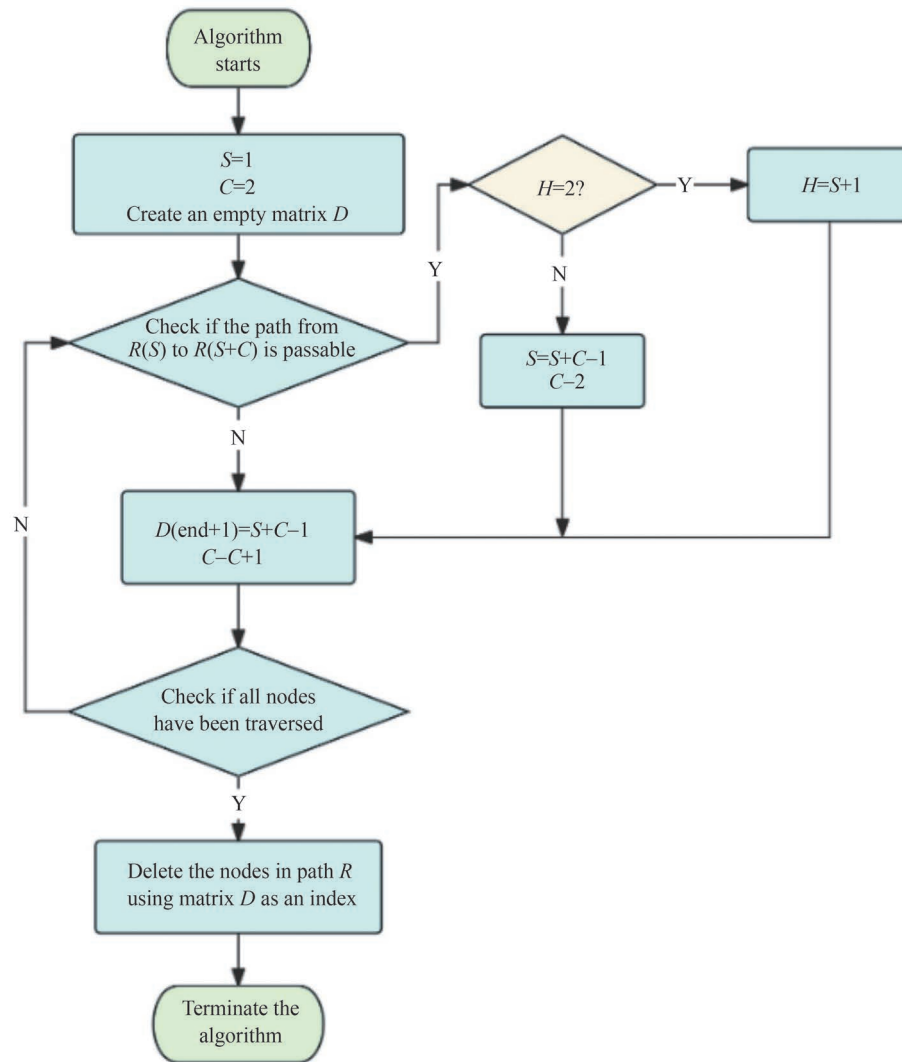


Figure 11 Node deletion algorithm flow

ships in the ice area, we propose improvement strategies to address the deficiencies of the DWA algorithm.

4.3.1 Improving the initial course angle

When the DWA algorithm starts operating, the initial course angle needs to be set. If the initial course angle is a random value generated by the algorithm, when its deviation from the 1st target subnode $R(2)$ is large, the searched path will make a detour in the early stages of the algorithm's operation (Figure 12).

To clarify the forward direction of the search path at the initial stage of the algorithm, we assume that the position coordinate of the starting point is (x_0, y_0) , and that of the first sub-target point $R(2)$ of the global path is (x_1, y_1) . The initial course angle is defined as follows:

$$\theta_0 = \arcsin \left(\frac{y_1 - y_0}{((x_1 - x_0)^2 + (y_1 - y_0)^2)^{\frac{1}{2}}} \right) \quad (18)$$

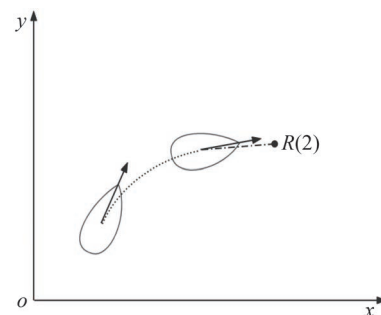


Figure 12 Path making a detour when the initial course angle's deviation from the 1st target subnode is large

4.3.2 Designing an adaptive evaluation function

The traditional algorithm consistently maintains the weights of each subfunction constant during operation. However, in the complex and changeable polar sea, the navigation needs of ships in the ice area change continuously. Evidently, difficulty arises in the implementation of

the traditional evaluation function when planning a path that meets its navigation needs. This problem is addressed through the improvement of the weight coefficients of the evaluation function:

$$\begin{cases} w'_1 = \left(1 + \frac{d(v, \omega)}{L}\right) \times w_1 \\ w'_2 = 2 \times w_2 \\ w'_3 = \left(2 - \frac{d(v, \omega)}{L}\right) \times w_3 \end{cases} \quad (19)$$

This weight adjusts adaptively with the ship's position in the ice area:

1) When the ship is distant from the obstacle, the algorithm expects its course angle to be adjusted to the right direction as soon as possible. In addition, the weight of the course angle subfunction in the evaluation function is higher at this point.

2) When the ship is close to the obstacle, the algorithm expects it to accelerate as quickly as possible to avoid the obstacle. In this case, the weight of the speed subfunction in the evaluation function is higher.

4.3.3 Current target node selection mechanism

The DWA algorithm needs to avoid obstacles in real-time during operation, and thus, its generated paths are occasionally inevitably deorbited from the global path. If the ship navigates past the current target node due to obstacle avoidance, the algorithm will continue to plan based on the original node, which will inevitably result in a bypass problem (Figure 13).

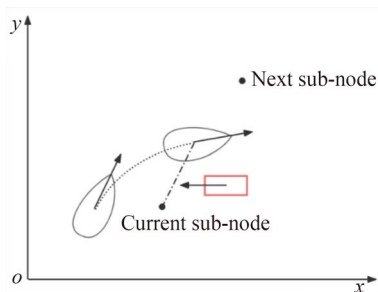


Figure 13 Bypass problem caused by obstacle avoidance

In the global path R , $R(i)$ is the current target subnode, and $R(i + 1)$ is the next target subnode. The angle between the connecting line from node $R(i)$ to the current position of the ship and the connecting line from node $R(i + 1)$ to the same position is constantly monitored during the algorithm operation. The current target node is kept constant at an angle less than 90° ; otherwise, the next target node is selected to continue the path planning. When obstacles force the ship to deviate from the planned global path, this switching mechanism reduces unnecessary path oscillations.

4.3.4 Optimization of frequent braking

In the single-objective point problem, when the DWA algorithm plans the path near the end point, a braking command is sent to the moving target due to the low scores of the course angle evaluation subfunction and distance evaluation subfunction.

In the application of the DWA algorithm, segmental optimization of the global path with only each subnode as the target node leads to frequent braking during the voyage. A threshold parameter H is introduced to solve this problem. When the current target node is a subnode of the global path instead of the end point, if the distance between the ship and the subnode is less than H , then the next subnode is considered the target node to continue planning.

Subsequent work related to this paper will consider real navigation scenarios, with a defined maximum ship speed of approximately 6 knots. An operational space with a large scale of 0.1 will be selected, and the simulated navigation time will be in the order of hundreds of seconds.

4.3.5 Advanced DWA framework

For path optimization based on a global path that meets ship performance in the ice area and effectively avoids dynamic obstacles, in accordance with the above strategies, the flow of the improved DWA algorithm is determined (Figure 14).

4.4 Framework of the hybrid algorithm

Ship path planning in ice areas based on the hybrid algorithm is mainly divided into global path planning and local path optimization. The process is described as follows:

- 1) Establish static and dynamic environment models based on real nautical charts;
- 2) Solve the global path in the static environment using the ACO algorithm;
- 3) Preprocess the global path using the node deletion algorithm to obtain key nodes applied to subsequent algorithms;
- 4) Consider the nodes of the preprocessed path as subgoals to be planned, add and delete several obstacles in the environment, and solve the final path using the improved DWA algorithm. When local deviations from the global path occur due to obstacle avoidance, the algorithm continuously monitors the direction toward the next subnode and realigns with the global path as immediately as feasible.

Figure 15 shows the framework of the hybrid algorithm.

5 Simulation and analysis

This chapter analyzes the simulation results. For consistency and reproducibility, the simulation calculation of ship paths in ice areas is based on the method proposed in this article.

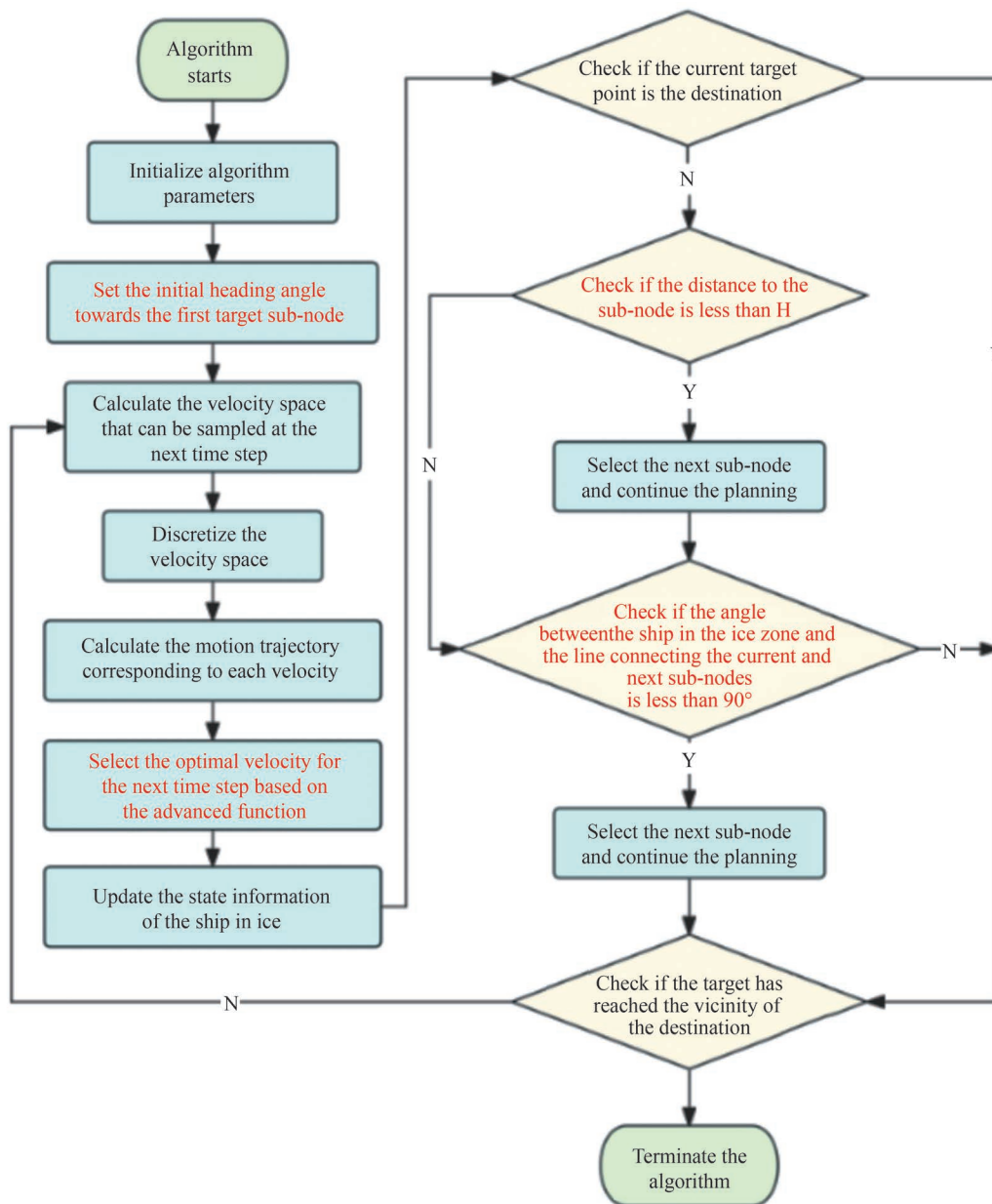


Figure 14 Flow of the improved DWA algorithm

5.1 Ant colony optimization

5.1.1 Simulation in a simplified environment

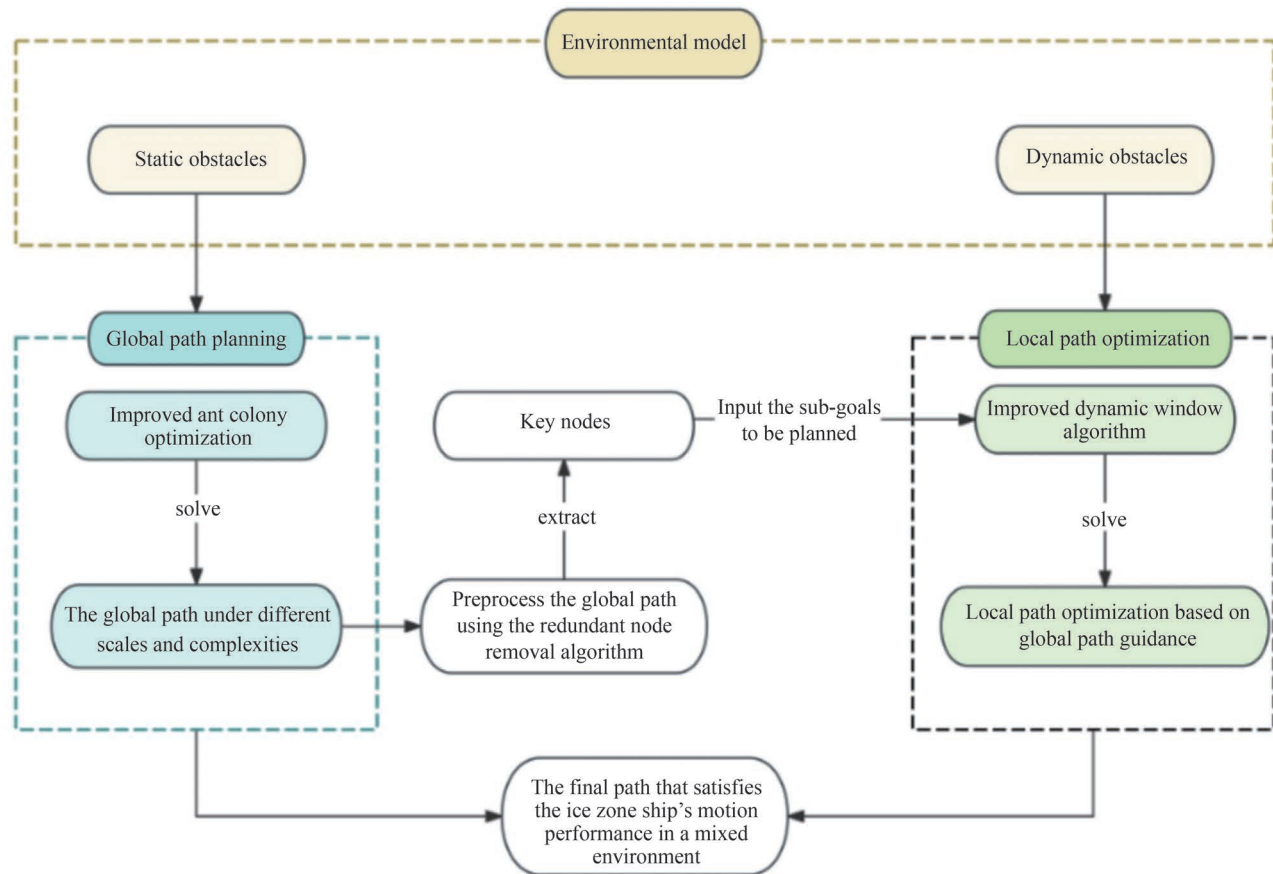
This simulation group selects a 15×15 raster map of the polar sea area. From 1 to 225, each grid in the model is identified in accordance with the serial number method. Table 1 shows each parameter of the algorithm. Figures 16 and 17 show the shortest global path planned and the trend of the convergence curve based on the traditional ACA algorithm, respectively. Figures 18 and 19 show the shortest global path planned and the trend of the convergence curve based on the ACO algorithm, respectively.

5.1.2 Simulation in a complex environment

This group of simulations selects a 30×30 raster map of the polar sea area. Each grid in the model is identified in accordance with the serial number method, with numbers ranging from 1 to 900. Table 2 provides each parameter of the algorithm. Figures 20 and 21 show the shortest global path planned and the trend of the convergence curve based on the traditional ACA algorithm, respectively. Figures 22 and 23 show the shortest global path planned and the trend of the convergence curve based on the ACO algorithm, respectively.

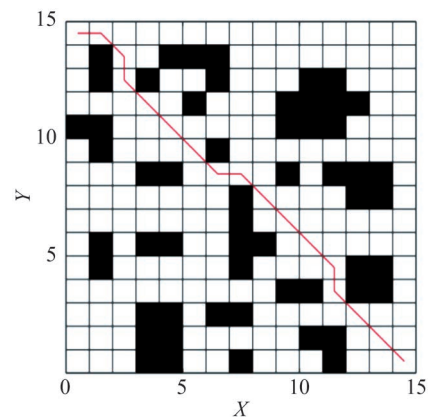
5.1.3 Simulation result analysis

Analysis of the simulation results reveals the following:

**Figure 15** Framework of the hybrid algorithm**Table 1** Parameters in the algorithms

Parameter	Traditional algorithm	Improved algorithm
$N_{\max}$	50	50
M	50	50
α	1	1
β	7	7
ρ	0.3	—
Q	1	1
P	—	5
λ	—	20
n	—	4
m	—	0.2
μ	—	1.3

1) In the case of the traditional algorithm and the improved algorithm with approximately the same parameter selection, both can search for the optimal path in the simple environment. The qualities of planning global paths are approximately the same, and both are 20.97 units in length. In the case of the same level of solution quality, the improved algorithm is more efficient and converges only after the 21st iteration. Meanwhile, the traditional algorithm converges after the 30th iteration.

**Figure 16** Shortest global path planned based on the traditional ACA algorithm

2) In the case of the traditional algorithm and the improved algorithm with approximately the same parameter selection, the path planned by the traditional algorithm in the complex environment shows multiple detours (path length: 48.18 units). The improved algorithm solves a high-quality path with a length of 43.94 units. The improved algorithm exhibits a higher solution quality, runs more stably, and converges only after the 40th iteration. The improved algorithm also displays occasional fluctuations after the 40th

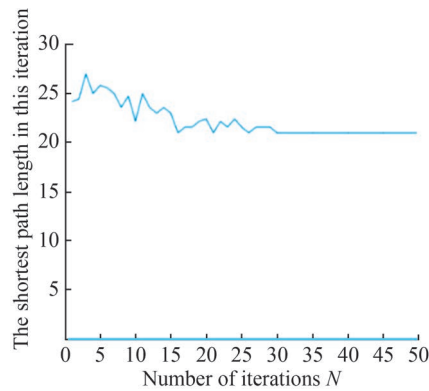


Figure 17 Trend of the convergence curve based on the traditional ACA algorithm

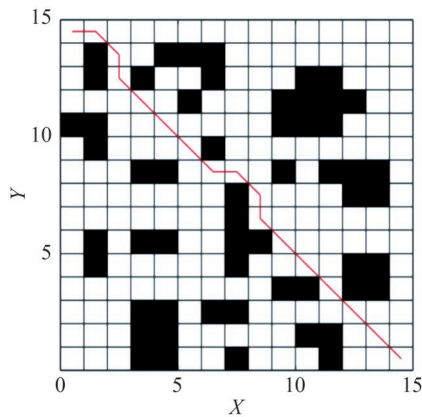


Figure 18 Shortest global path planned based on the ACO algorithm

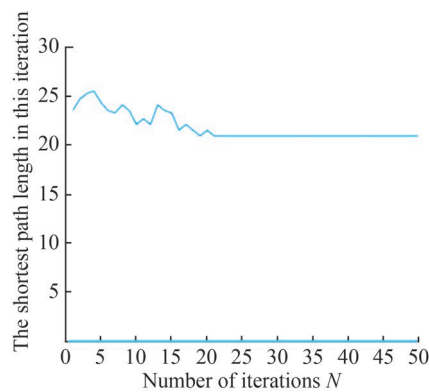


Figure 19 Trend of the convergence curve based on the ACO algorithm

iteration, but the subsequent iterations have a smooth curve. Meanwhile, the traditional algorithm lacks an optimal solution, and thus, the iteration curve still oscillates considerably until the end.

5.2 Node elimination algorithm

5.2.1 Simulation results

Figures 24 and 25 show the schematic of the path in a 15×15 raster model before and after preprocessing, respectively:

Table 2 Parameters in the algorithm

Parameter	Traditional algorithm	Improved algorithm
$N_{\max}$	100	100
M	50	50
α	1	1
β	7	7
ρ	0.3	—
Q	1	1
P	—	7
λ	—	20
n	—	4
m	—	0.15
μ	—	1.5

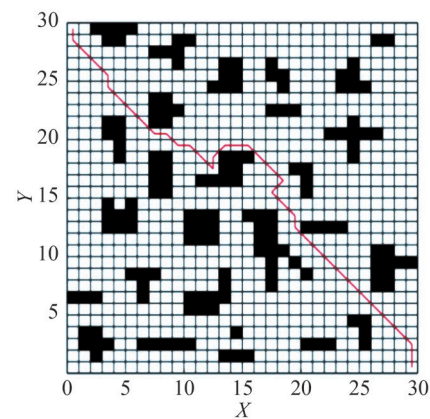


Figure 20 Shortest global path planned based on the traditional ACA algorithm

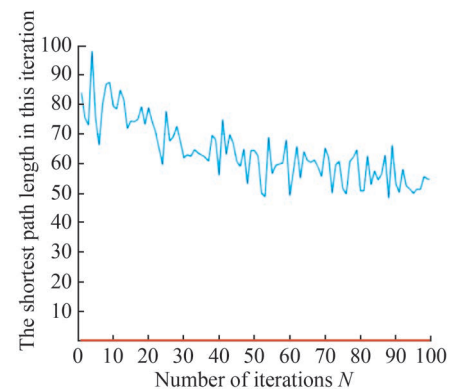


Figure 21 Trend of the convergence curve based on the traditional ACA

Figures 26 and 27 show the schematics of the path in a 15×15 raster model before and after preprocessing, respectively:

5.2.2 Simulation result analysis

From the analysis of the above simulation results, in the simple environment, the global path length before prepro-

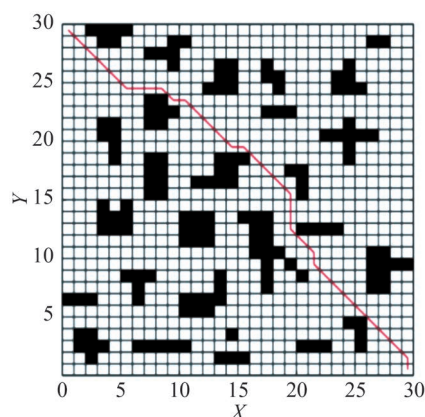


Figure 22 Shortest global path planned based on the ACO algorithm

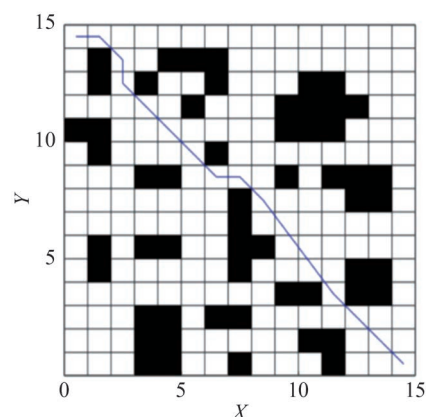


Figure 25 Path after preprocessing

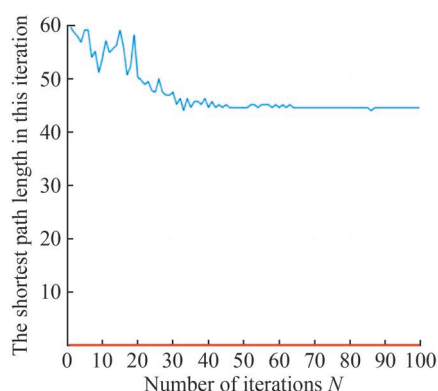


Figure 23 Trend of the convergence curve based on the ACO algorithm

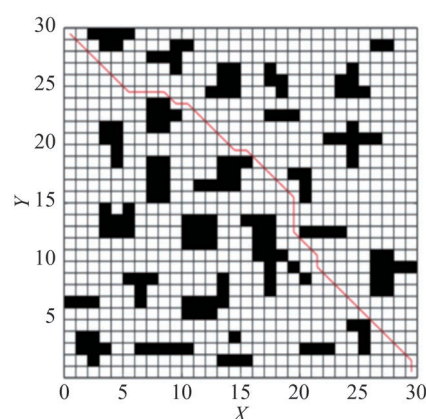


Figure 26 Path before preprocessing

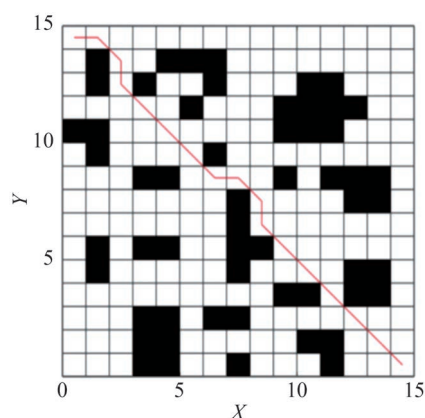


Figure 24 Path before preprocessing

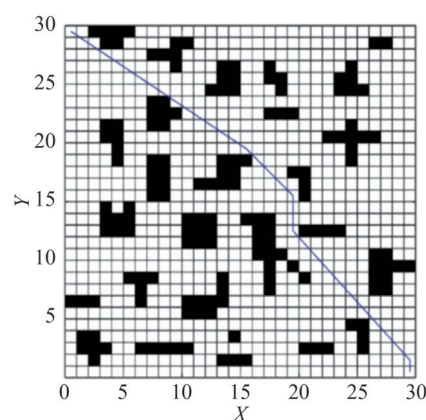


Figure 27 Path after preprocessing

cessing reaches 20.97 units, and the preprocessed path is reduced to a length of 20.72 units. In the complex environment, the global path length before preprocessing is 43.94 units, and the path after preprocessing is reduced to a length of 42.56 units.

5.3 Dynamic window approach

Each grid in the raster model has a default size of $1 \text{ m} \times 1 \text{ m}$, scaled by the scale. Table 3 contains the parameters

of the DWA algorithm:

For the clear visualization of the clusters of trajectories generated by the algorithm at various moments, a larger scale needs to be selected. This selection probably leads to a linear velocity that is extremely large to turn for the avoidance of obstacles. Therefore, the angular velocity-related parameters need to be increased to improve the slewing performance. Table 4 shows the motion parameters of the ship in the ice area:

Table 3 Parameters of the DWA algorithm

Parameter	Value
Scale	0.1
Predictive simulation time T	0.5 s
Simulation period Δt	0.1 s
Linear velocity resolution Δv	0.1 m/s
Angular velocity resolution $\Delta \omega$	10/s
Weight of the course angle evaluation subfunction w_1	9
Weight of the safe distance evaluation subfunction w_2	3.5
Weight of speed evaluation subfunction w_3	3
Threshold parameter of safe distance evaluation subfunction L	15 m
Threshold parameter of the selection of subnode H	8 m

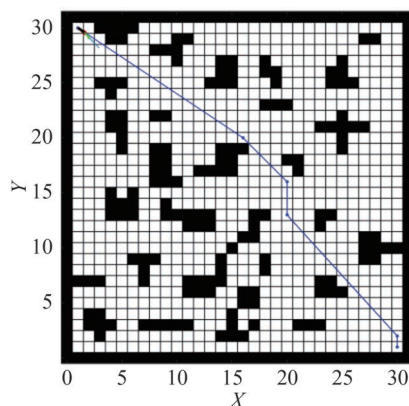
Table 4 Motion parameters of the ship in the ice area

Motion parameter	Value
Minimum linear velocity $v_{\min}$	0 m/s
Maximum linear velocity $v_{\max}$	3 m/s
Maximum linear acceleration $\dot{v}$	1.5 m/s ²
Minimum angular velocity $\omega_{\min}$	-1.75 rad/s
Maximum angular velocity $\omega_{\max}$	1.75 rad/s
Maximum angular acceleration $\dot{\omega}$	2.5 rad/s ²

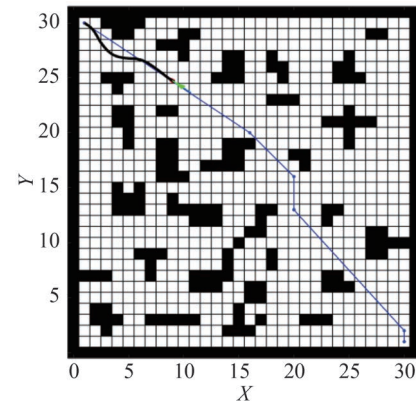
5.3.1 Simulation in a static environment

Given the limitation of chart accuracy, some discrepancies are observed between the actual sea area and the raster model. Considering the unknown static obstacles, some obstacles are added and deleted in the 30×30 raster model, and the path is planned based on the preprocessed global path.

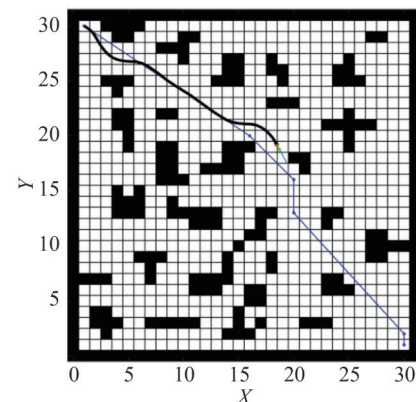
Figure 28 shows the effect diagram when the DWA algorithm starts functioning. At this point, the ship in the ice area sails from the starting point toward the first subtarget. The green part represents the trajectory cluster simulated forward at the current moment. Given that the initial moment linear speed is 0, and the ship is in the acceleration process, the trajectory space is small.

**Figure 28** Effect diagram when the DWA algorithm starts operating

As shown in Figure 29, the global path disregards the distance to obstacles. Therefore, the algorithm is deorbited from the global path during the local optimization process to avoid known and unknown obstacles and then returns to the global path when the navigation environment ahead is safe.

**Figure 29** Global path that disregards the distance to obstacles

As displayed in Figure 30, the target subnode toward which the algorithm is directed lies directly above the obstacle. A diversion is selected to maintain a safe distance from the obstacle. At the end of the diversion, the current node is passed, and the next subnode is selected as the target point for continued planning.

**Figure 30** Target subnode toward which the algorithm is directed is directly above the obstacle

As revealed in Figure 31, the ship in the ice area continues its navigation under the guidance of the global path, which avoids all obstacles in the environment.

As shown in Figure 32, when the ship in the ice area has reached the vicinity of the end point, the algorithm ceases running.

Figures 33 and 34 show the trends of angular and linear velocities, respectively.

5.3.2 Simulation in a mixed environment

Ships are also threatened by dynamic obstacles in polar

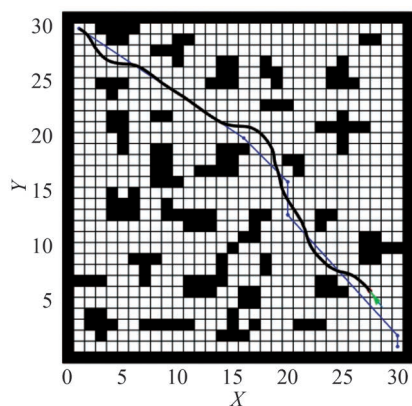


Figure 31 Ship in the ice area continuing its navigation under the guidance of the global path

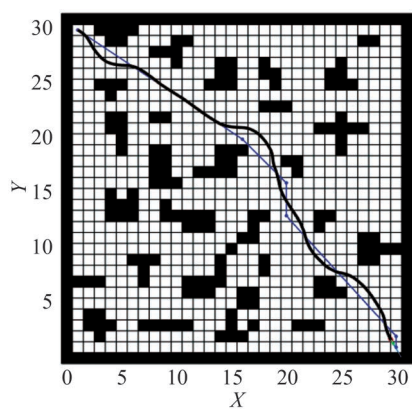


Figure 32 Ship in the ice area after reaching the vicinity of the end point

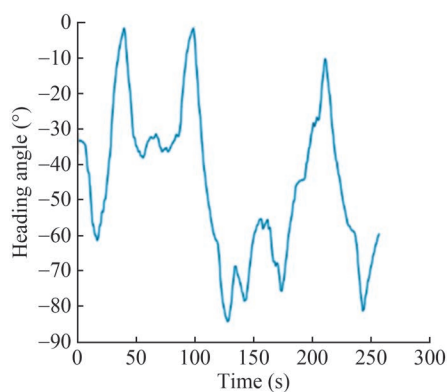


Figure 33 Trend of angular velocity

waters, and thus, two dynamic obstacles have been added to the previous chapter:

$$s_1^{t=0} = (10 \text{ m}, 10 \text{ m}, 24.5, 14.5, \pi, 0.31 \text{ m/s}, 0 \text{ rad/s}) \quad (20)$$

$$s_2^{t=0} = \left(15 \text{ m}, 10 \text{ m}, 26.5, 5.5, \frac{\pi}{2}, 0.3 \text{ m/s}, 0.47 \text{ rad/s}\right) \quad (21)$$

Figure 35 shows the positions of two dynamic obstacles when the algorithm starts running.

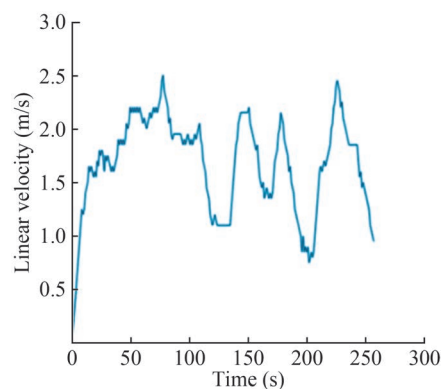


Figure 34 Trend of linear velocity

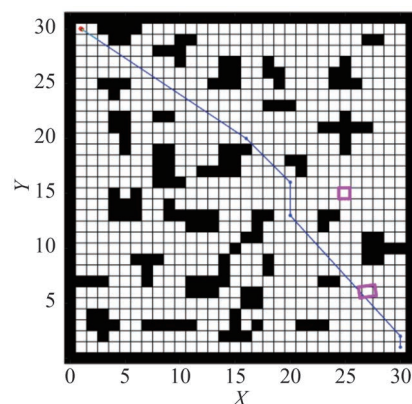


Figure 35 Positions of two dynamic obstacles when the algorithm starts running

In the prometaphase of the algorithm's operation, the ships in the ice area are distant from the dynamic obstacles, and thus, the paths planned in the first half are the same as those in the previous chapter (Figure 36):

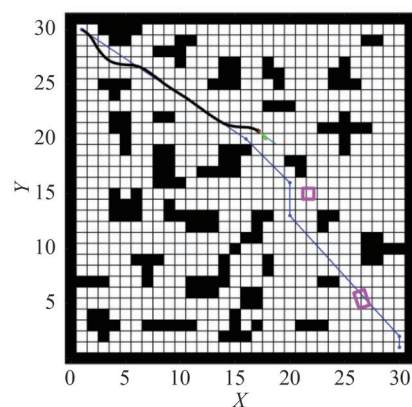


Figure 36 Global path at the prometaphase

Figure 37 illustrates the encounter of the ship in the ice area with a linear motion obstacle.

Figure 38 displays the avoidance of the ship in the ice area with a linear motion obstacle.

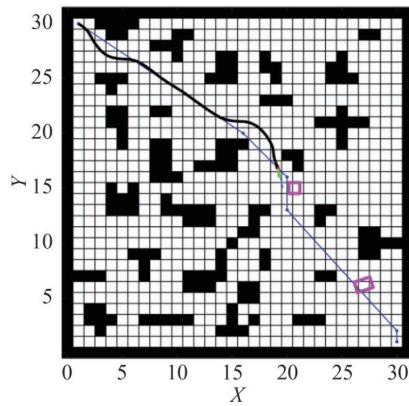


Figure 37 Encounter of the ship in the ice area with a linear motion obstacle

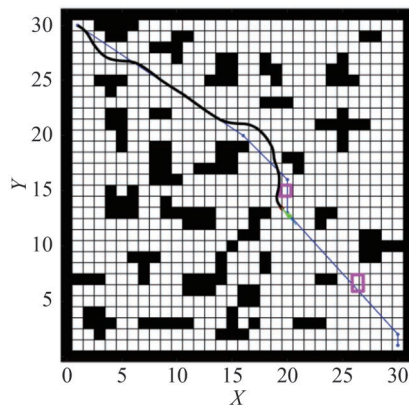


Figure 38 Avoidance of the ship in the ice area with a linear motion obstacle

Figure 39 unveils the encounter of the ship in the ice area with a circumferential motion obstacle

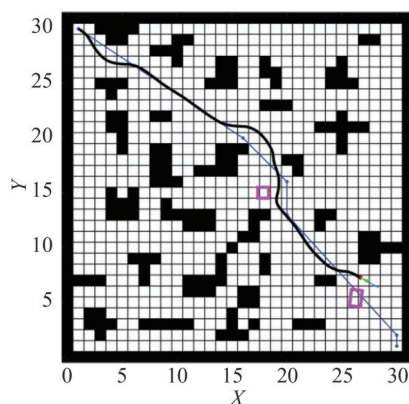


Figure 39 Encounter of the ship in the ice area with a circumferential motion obstacle

Figure 40 shows the avoidance of the ship in the ice area with a circumferential motion obstacle.

As revealed in Figure 41, the algorithm stops running

when the ship in the ice area has reached the vicinity of the end point.

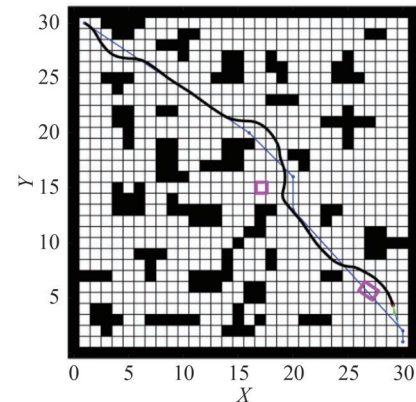


Figure 40 Avoidance of the ship in the ice area with a circumferential motion obstacle

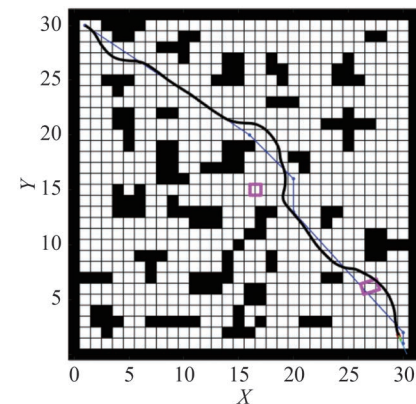


Figure 41 Ship in the ice area reaching the vicinity of the end point

Figures 42 and 43 display the trends of angular and linear velocities.

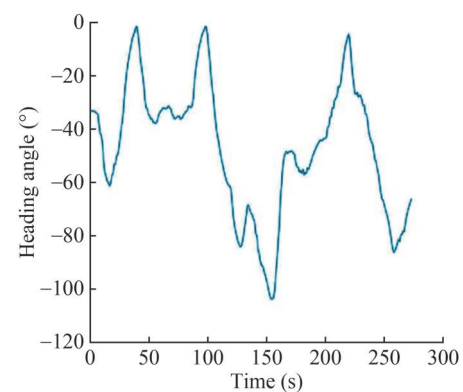


Figure 42 Trend of angular velocity

5.3.3 Simulation result analysis

Analysis of the static environment simulation results reveals that under the guidance of the global path, the

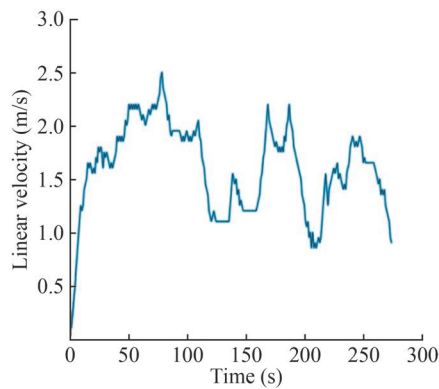


Figure 43 Trend of linear velocity

DWA algorithm can enable the ship in the ice area to avoid all known and unknown static obstacles in the environment. The optimized path is smoother and conforms to the motion performance of ships in the ice area. Furthermore, the moment of acceleration, deceleration, and turning is clearly planned.

Analysis of the simulation results of the hybrid environment indicates that the polar sea can be better simulated after the addition of dynamic obstacles to the original environment model. The algorithm also exhibits excellent avoidance capability for dynamic obstacles in the environment. The planned path can ensure that the ship in the ice area can navigate safely and quickly to its destination.

6 Conclusions

In this study, an innovative method is proposed to plan a reliable path for ships in ice areas using a hybrid ACO-DWA algorithm with the following advantages:

1) Four improvement strategies are proposed for the traditional ACA, and the simulation verification shows that the improved algorithms have higher efficiency and quality, capable of planning a global path that avoids all known static obstacles in the environment.

2) A node deletion algorithm is designed to effectively eliminate unnecessary nodes after preprocessing of the global path, shorten the path length, improve its smoothness simultaneously, and screen out the necessary sub-nodes for the subsequent local path optimization.

3) Considering the navigational space and operational characteristics of ships in the ice area, four improvement strategies are proposed based on the DWA algorithm, with optimization of the local path with the nodes, which are on the preprocessed path, as the sub-targets to be optimized by the improved DWA algorithm. The simulation results reveal that the algorithm can avoid unknown static obstacles and dynamic obstacles in the environment.

Several deficiencies should be addressed, and follow-up work can be carried out in the following two aspects:

1) Real ice area scenarios should be further considered, and ice-breakable areas should be added to the environment model. The objective function should be established by comprehensively considering the navigation time, safety, and other factors to evaluate the execution strategy of ships in ice areas.

2) Ships in ice areas are generally navigated by ice-breakers and special transport ships in a common formation. The obstacle models considered in this paper are limited, and future work can focus on further analysis of multi-objective optimization models related to ship convoy navigation, encounters, and other factors.

3) This paper focuses solely on the analysis of obstacle avoidance operations for ships in ice areas and therefore skips the scenarios involving various speed ranges. Future work can further analyze polar ships navigating in open waters or engaged in icebreaking operations at varying speeds.

Funding This research was supported by the National Key Research and Development Program of China (No. 2021YFC2803400) and the Joint Funds of the National Natural Science Foundation of China (No. U20A20327).

Competing interest The authors have no competing interests to declare that are relevant to the content of this article.

References

- Cai Z, Liu JH, Xu L, Wang JY (2024) Cooperative path planning study of distributed multi-mobile robots based on optimised ACO algorithm. *Robotics and Autonomous Systems* 179: 104748. <https://doi.org/10.1016/j.robot.2024.104748>
- Charalambopoulos N, Xidias E, Nearchou A (2023) Efficient ship weather routing using probabilistic roadmaps. *Ocean Engineering* 273: 114031. <https://doi.org/10.1016/j.oceaneng.2023.114031>
- Choi MJ, Chung H, Yamaguchi H, Nagakawa K (2015) Arctic sea route path planning based on an uncertain ice prediction model. *Cold Regions Science and Technology* 109: 61–69. <https://doi.org/10.1016/j.coldregions.2014.10.001>
- Diebold FX, Rudebusch GD (2022) Probability assessments of an ice-free Arctic: Comparing statistical and climate model projections. *Journal of Econometrics* 231(2): 520–534. <https://doi.org/10.1016/j.jeconom.2020.12.007>
- Huo FC, Zhu S, Dong HL, Ren WJ (2024) A new approach to smooth path planning of Ackerman mobile robot based on improved ACO algorithm and B-spline curve. *Robotics and Autonomous Systems* 175: 104655. <https://doi.org/10.1016/j.robot.2024.104655>
- Lee HW, Roh MI, Kim KS (2021) Ship route planning in Arctic Ocean based on POLARIS. *Ocean Engineering* 234: 109297. <https://doi.org/10.1016/j.oceaneng.2021.109297>
- Li HB, Wang XY, Soares CG, Ni SK (2024) A collision-avoidance decision-making scheme based on artificial potential fields and event-triggered control. *Ocean Engineering* 306: 118101. <https://doi.org/10.1016/j.oceaneng.2024.118101>
- Liu YJ, Xue YZ, Lu Y, Yuan LH, Li F, Li RH (2024) A Dynamic Bayesian Network model for ship navigation risk in the Arctic Northeast Passage. *Ocean Engineering* 312: 119024. <https://doi.org/10.1016/j.oceaneng.2024.119024>

- org/10.1016/j.oceaneng.2024.119024
- Qian H, Zhang R, Zhang YJ (2020) Dynamic risk assessment of natural environment based on Dynamic Bayesian Network for key nodes of the Arctic Northwest Passage. *Ocean Engineering* 203: 107205. <https://doi.org/10.1016/j.oceaneng.2020.107205>
- Song RX, Papadimitriou E, Negenborn RR, van Gelder P (2024) Integrating situation-aware knowledge maps and dynamic window approach for safe path planning by maritime autonomous surface ships. *Ocean Engineering* 311: 118882. <https://doi.org/10.1016/j.oceaneng.2024.118882>
- Shu YQ, Zhu YJ, Xu F, Gan LX, Lee PT W, Yin JC, Chen JH (2023) Path planning for ships assisted by the icebreaker in ice-covered waters in the Northern Sea Route based on optimal control. *Ocean Engineering* 267: 113182. <https://doi.org/10.1016/j.oceaneng.2022.113182>
- Sun MX, Xiao XT, Luan TT, Zhang XS, Wu BQ, Zhen LQ (2024) The path planning algorithm for UUV based on the fusion of grid obstacles of artificial potential field. *Ocean Engineering* 306: 118043. <https://doi.org/10.1016/j.oceaneng.2024.118043>
- Tang CY, Xia JF, Wang WK, Shang HJ, Ji W, Bao QM, Yang DX (2024) A dynamic window model prediction of artificial potential field method for improving the coincidence of actual and predicted trajectory of underactuated planing craft. *Ocean Engineering*. 2024, 313: 119351. <https://doi.org/10.1016/j.oceaneng.2024.119351>
- Wang SY, Yu FJ, Min C, He Y, Pan RR, Shu Q (2024) Projected navigability of Arctic shipping routes based on climate model FIO-ESM v2.1. *Anthropocene* 47: 100445. <https://doi.org/10.1016/j.ancene.2024.100445>
- Zhang C, Zhang D, Zhang MY, Zhang JF, Mao WG (2022) A three-dimensional ant colony algorithm for multi-objective ice routing of a ship in the Arctic area. *Ocean Engineering* 266: 113241. <https://doi.org/10.1016/j.oceaneng.2022.113241>
- Zhang T, Gao FZ (2024) A Traditional Cultural Route Planning and Design System Based on Improved Genetic Algorithm. *Procedia Computer Science* 247: 211-217. <https://doi.org/10.1016/j.procs.2024.10.025>