American Journal of Traffic and Transportation Engineering
2023; 8(6): 158-165
http://www.sciencepublishinggroup.com/j/ajtte
doi: 10.11648/j.ajtte.20230806.15
ISSN: 2578-8582 (Print); ISSN: 2578-8604 (Online)

Methodology Article

A Two-Stage Algorithm for Coastal Complex Water Route Planning Based on A* and Recursion

Dongkui Wu, Fan Yang, Lixiong Chen*

Merchant Marine College, Shanghai Maritime University, Shanghai, China

Email address:

wudk@shmtu.edu.cn (Dongkui Wu), 879209294@qq.com (Fan Yang), chenlixiong2002@163.com (Lixiong Chen)

*Corresponding author

To cite this article:

Dongkui Wu, Fan Yang, Lixiong Chen. (2023). A Two-Stage Algorithm for Coastal Complex Water Route Planning Based on A* and Recursion. *American Journal of Traffic and Transportation Engineering*, *8*(6), 158-165. https://doi.org/10.11648/j.ajtte.20230806.15

Received: November 16, 2023; **Accepted**: December 6, 2023; **Published**: December 22, 2023

Abstract: Route planning is a crucial task in maritime navigation, as it ensures the safety of navigation and reduces fuel consumption. In order to further enhance the safety of vessel navigation in complex waterways, reduce sailing distances, and improve the convenience of vessel operations, this paper addresses the shortest path problem for vessel navigation in complex waterways. This problem aims to find a route with the shortest sailing distance and the fewest number of ship turning maneuvers in a complex water area, while ensuring navigation safety from the starting point to the destination. By abstracting the actual sailing process of vessels, the paper initially divides the navigation area into grids, and then differentiates between impassable and passable grids by inflating the boundaries of navigational obstacles. An integer programming model based on grid partitioning is formulated to describe this problem, and the optimality of the routes is analyzed. A two-stage algorithm is proposed to solve this problem, wherein the first stage utilizes the A* algorithm to find the shortest path from the starting point to the destination, and the second stage uses a recursive algorithm based on the path generated in the first stage to adjust turning points and further reduce the sailing distance. Finally, a simulation platform is built using Python, and the above algorithms are employed for experimentation. The experimental results demonstrate the effectiveness of the proposed model and algorithms.

Keywords: Grid Method, A* Algorithm, Recursive Algorithm, Route Design

1. Introduction

Ship route planning is an important reference factor for ensuring ship navigation safety and developing green shipping. Guo Dongdong et al. [1] have highlighted that fuel consumption not only represents a primary operational cost for vessels but also serves as a major source of pollutant gas emissions. Therefore, research and application related to optimizing ship routes provide an effective approach to reducing fuel consumption. Furthermore, Yan Xinping et al. [2] have underscored that optimizing ship routes is a pivotal step within autonomous navigation technology for vessels.

In ship route design, the planning of routes along complex coastal waters has always been a challenging task. These areas often encompass unpredictable maritime conditions, shoals, islands, and various obstacles, posing considerable difficulties for safe navigation of vessels. To overcome these challenges, this paper proposes an innovative route planning method that combines the A* search algorithm and recursive strategy, segmented into two key stages, aimed at effectively charting a safe navigational path for vessels in complex coastal waters.

The initial stage employs the A* algorithm, establishing appropriate state space and heuristic functions to seek the shortest path from the starting point to the destination. In this phase, the A* algorithm efficiently explores feasible paths within the navigational area, considering marine environmental conditions, vessel characteristics, and the necessity to circumvent obstacles, to generate a preliminary route. Research by Lv Chao et al. [3] suggests that, in route planning, the A* algorithm demonstrates an approximately 40% efficiency improvement compared to the Dijkstra algorithm.

Subsequently, the second stage utilizes a recursive strategy to optimize and refine the initial route. By considering additional environmental information and navigational constraints near the route, the recursive algorithm further enhances the path based on the initial route, adapting to changing conditions in complex waterways. This phase emphasizes maximizing route efficiency and convenience while ensuring safe navigation.

The two-stage route planning method proposed in this paper fully leverages the search efficiency of the A* algorithm and the path optimization capabilities of the recursive strategy. It presents a novel solution for charting safe and efficient routes for vessels navigating in complex coastal waters. This method not only addresses the diverse maritime environmental and topographical conditions but also maximizes vessel navigation efficiency and convenience while ensuring safety, heralding significant technological advancements and innovation in the maritime transportation sector.

2. Basic Methods and General Principles of Coastal Complex Water Planning

Safety is the foremost consideration when designing general routes [4]. Each country's offshore areas have their established standard routes, which can be located in the "Route Guide". Route design is the responsibility of the second officer, and during the planning process, the second officer should take into account the captain's voyage requirements. In route design, it's essential to plot the entire planned route on the master chart and provide annotations for each segmented part on the navigational chart. Care must be taken to avoid any overlapping notations on the navigational chart. Furthermore, it is important to adjust the spacing between hazardous objects and indicate longitude, latitude, and turning direction details at turning points. Once the route design is completed, it should be submitted for the captain's review. Only when the captain approves and accepts the plan can it be put into action.

There are two general principles in the design of offshore routes: safety and economy. The design of the route should ensure that ships arrive at the destination port safely and on time, and positive impacts such as downstream and windward can also be utilized[4]. It can be seen that the best route should be to make reasonable use of hydrology and meteorology, avoid danger as much as possible, and obtain a safe, efficient, comfortable, and ideal route with minimal cargo damage.

The design of coastal complex water routes, in a sense, is the planning problem of the shortest path. The shortest path problem is a problem in the field of combinatorial optimization. In recent years, the optimization problem of the shortest path has also developed rapidly in areas such as drones, map navigation, vehicle loading, and operations research. Zhou Yinfei utilizes electronic nautical charts to identify obstructed areas, pre-generate waypoints, extract in sequence the largest feasible navigable window sequences that do not intersect with obstructed areas from the pre-generated waypoints, and construct a navigable constraint space [5]. Pan Mingyang, Liu Yisai, Li Qi, Li Chao, and Chen Zhitai have designed a directed topological network for an inland waterway system based on navigational conditions. They enhanced the A* algorithm by integrating waterway-related constraints, optimizing cost functions, and evaluating functions. This improvement allows for optimized route planning that caters to diverse vessel parameters [6]. In 2022, Cui Kangjing, Zheng Yuanzhou, Chen Guocheng, Hu Weidong and Others used artificial potential field method and simulated annealing algorithm, considering the pre-conditions of wind speed and wave height, it effectively avoids large wind and wave areas and reduces voyage time [7]. Chen Xiao, Dai Ran, Zhao Yanpeng, Zhang Chaoyue and Others studied the use of fish school algorithms for shallow beach avoidance in route design [8].

For the optimization of the shortest path, commonly used path algorithms include single object path finding algorithm, Dijkstra algorithm, genetic algorithm, and other algorithms. Liu Zhifang used the adaptive search ability and parameter optimization techniques of genetic algorithms to optimize the accuracy of the controller in "Research on Genetic Algorithms in Ship Route Adaptive Controllers" [9]. In "Route Planning Based on Maklink Graph and Ant Colony Algorithm", Chen Xiao proposed using Dijkstra algorithm to obtain the initial path and then using ant colony algorithm for optimization. Compared with traditional paper chart route design, it has the characteristics of short time consumption, low economic cost, etc [10]. Zhu Qing used ant colony algorithm for path optimization in "Research on Ship Route Design Method Based on Ant Colony Algorithm". And conducted sea state simulation on environmental modeling to verify the convergence of the algorithm [11]. Yao Xiaoxiao et al. employed the Douglas-Peucker algorithm to compress extensive trajectory data. Subsequently, they applied the DBSCAN algorithm to cluster the processed AIS trajectory point data and extract key turning points in the routes [12]. Xie Xinlian conducted an analysis of buffer zones based on Predicted Area of Danger (PAD) theory for avoiding target vessels, drawing upon relevant achievements in the maritime field. This was followed by environmental modeling of complex waterways using the tangent graph method. Subsequently, the Dijkstra algorithm was employed to compute an initial path that circumvents all navigational hazards [13]. Yu Bixiu improves the A* algorithm to enable unmanned hydrographic survey vessels to return to the preset route more quickly after avoiding obstacles. [14]. Wang Lipeng designed a secondary optimization path planning method that integrates ship motion characteristics to address automated route planning tasks in complex navigational environments [15]. Tong Bangyu improved the Ant Colony Algorithm to enhance the planning effectiveness of ship routes in navigating through icy regions, considering the particularities of ship route selection in icy navigational environments [16].

3. Problem Description and Integer Programming Model

This section first describes the shortest path problem for vessel navigation and provides a detailed explanation of grid partitioning and obstacle inflation within navigable waters.

Subsequently, we present a 0-1 integer programming model for this problem and analyze the optimality of the path based on this mathematical model.

During vessel navigation, the surrounding maritime areas can be monitored using radar and other equipment, such as the circular area in Figure 1 (solid circle centered at o). The destination terminal of the vessel is located at point D in Figure 1, where ideally the vessel should travel along line segment OD. However, due to the complexity of navigation at sea and considering the detection range of the radar, the ideal destination for the vessel is point B, which is the intersection of line segment OD and circle O. Consequently, a rectangular area OABC with OB as the diagonal is constructed to represent the vessel's navigable area. The problem under consideration in this paper is to consider various potential navigational obstacles within the water area OABC and plan a route with the shortest sailing distance and minimal ship turning maneuvers from the current point O to the destination point B.

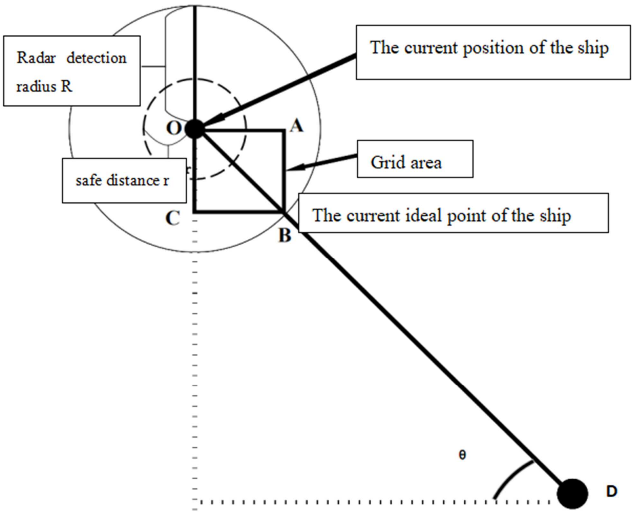


Figure 1. *The ship's radar range.*

As the navigation area is determined, the grid partitioning and obstacle inflation handling should be done consequently. Firstly, the navigable area is partitioned into grids based on a specified scale, followed by the determination of the positions of various obstacles within the navigable area. Secondly, the allocation of grids occupied by these obstacles based on their boundaries. Thirdly, the original boundaries of the obstacles are inflated based on the safety distance associated with different obstacles, resulting in new boundaries. Finally, the grids occupied by the obstacles are determined based on these inflated obstacle boundaries that consider the safety distance.

Based on the above problem, we provide a detailed mathematical description of this issue. Let set N denote the set of grid cells that represent the navigational area. For any cell $i \in N$, each cell can only be connected to its adjacent cells $j \in N_i$. The distance between the center of two adjacent cells is denoted as d_{ij}. The line connecting two adjacent cells (i, j) is a navigable section in the navigational area (also referred to as an edge). Since navigational obstacles exist in the navigational area, as described in Section 3.1, all the occupied grids obtained by inflating the obstacles in the area are defined as no-go zones and represented as set F, where $F \subseteq N$. Therefore, for any edge (i, j), if i or j is in a no-go zone F, the ship cannot navigate on that edge, and the edge is referred to as an infeasible edge. The navigable edges are referred to as feasible edges. Additionally, considering the need to account for ship turning in the navigation process, the true north direction is taken as the positive direction, and the acute angle between any feasible edge and the positive direction is defined as the direction of that feasible edge. Therefore, for any grid cell $i \in N$ that is not in a no-go zone, we only need to consider whether the angles of the entering edge (j, i) and the exiting edge (i, k) in that cell are consistent to determine whether a turning operation should be performed at that point. Thus, our problem can be transformed into finding a set of feasible edges for the ship to travel from the starting grid to the destination grid with the fewest path segments and turning maneuvers. Next, we present the assumptions and mathematical model of this problem.

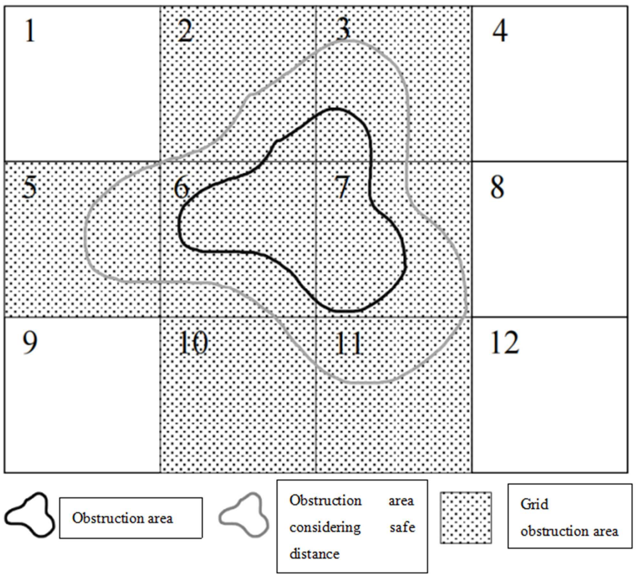


Figure 2. *Obstruction expansion diagram.*

Assumption 1. Since the observable range of radar is relatively small compared to the radius of the Earth, the distance between any two points is considered as the Euclidean distance.

Assumption 2. To ensure navigation safety, for any grid cell, as long as there is any obstacle in it regardless of size, it will be regarded as an no-go area.

4. Two Stages Algorithm Based on A* and Recursive Algorithm

In order to generate the optimal sailing route based on real-time sea conditions during the ship's journey, this section proposes a two-stage algorithm to solve the aforementioned problem. The first stage uses the A* algorithm to find the shortest path between the starting point and the end point. In the second stage, on the path generated in the first stage, the influence of grid inflation described in 3.1 is removed. While ensuring safety, a recursive algorithm is used to continuously adjust the number of turns on the path to minimize the number of turning points. Figure 3 illustrates the process of our algorithm.

4.1. A Algorithm*

The A* algorithm combines the advantages and characteristics of breadth first algorithm, Dijkstra algorithm, best first algorithm, and other algorithms. Through a certain heuristic function, the search algorithm is based on the principle of combining evaluation functions to provide guidance for expanding nodes and selecting the best node to meet the computational requirements of the algorithm. Through a certain heuristic function, the search algorithm is based on the principle of combining evaluation functions to provide guidance for expanding nodes and selecting the best node to meet the computational requirements of the algorithm. $h(n)$ represents the distance from the current point to the starting point and $g(n)$ represents the distance from the current point to the endpoint. Therefore, this function considers two objectives: 1) how much the current node has advanced from the starting point; 2) How far is the current node from the endpoint. Therefore, each grid which rasterized (excluding prohibited navigation areas) can be considered as a node. Below, we provide the detailed steps of the A * algorithm:

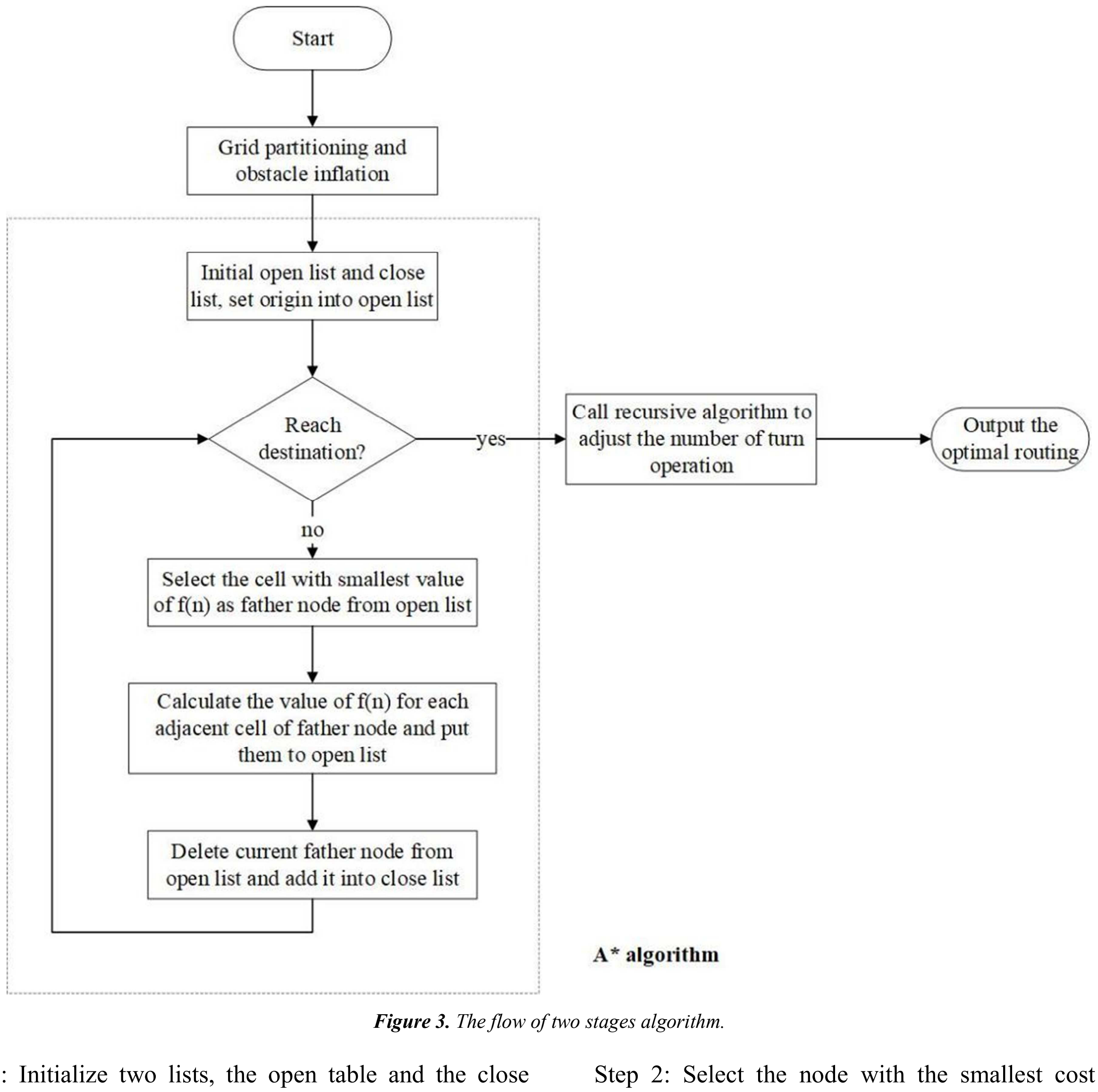


Figure 3. *The flow of two stages algorithm.*

Step 1: Initialize two lists, the open table and the close table. The open list is used to store nodes to be explored, and its significance is that the next node to be explored can be selected from it. The close list represents nodes that have been explored. During initialization, as there are no explored nodes, only the starting point is stored in the open list, and the close list is empty.

Step 2: Select the node with the smallest cost function from the open list to explore, place it in the close list, calculate the cost of adjacent nodes that are not in the open list, and store it in the open table.

If the currently selected node is the endpoint, the algorithm terminates; otherwise, repeat step 2.

From the above steps, it can be seen that the A * algorithm always finds a node with the lowest cost to explore, so it can be determined that the final path is the shortest path.

4.2. Recursive Algorithm

After the A* algorithm computes an optimal path, it may not necessarily have the minimum number of turning points. Additionally, the reduction in dimensionality through grid inflation may result in some feasible routes not being considered. As shown in Figure 4, route 1 is the optimal route generated by the first stage, although, route 2 cross the expanded cell, it keeps a distance from the boundary of the obstacle. However, route 3 and route 4 are infeasible, because they intersect or touch the boundary of the original obstacle or the boundary considering safety distance. Therefore, one of the most crucial determinations is whether the adjusted edge will intersect or touch obstacles. If the adjusted edge does not intersect with obstacles, we have designed an algorithm based on recursion to adjust the turning points and optimize the path. The pseudocode for the recursive algorithm is as follows. For ease of algorithmic expression, we define the following sets, parameters, and operators: Let set I represent the set of all turning points (including the starting point and the end point) computed by the A* algorithm. Define a decision operator $cross(i, j)$ to indicate whether the updated edge (i, j) intersects or touches obstacles. When $cross(i, j) = 1$, it indicates that the updated edge intersects or touches the current obstacle, and when $cross(i, j) = 0$, it indicates no intersection or touch.

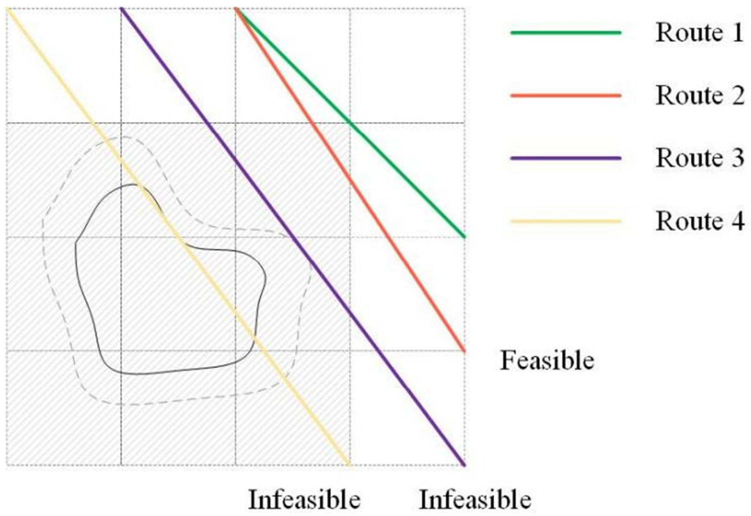


Figure 4. *Examples for feasible and infeasible routes.*

Table 1. *Recursive algorithm.*

Algorithm 1:	$\boldsymbol{Recursive\ (S, T, I)}$
Input:	Start point S, destination T and all turning points I
Output:	Adjusted turning points I
	if $\lvert I \rvert = 3$:
	if $cross(S, T) = 1$: Return I
	else return $\{S, T\}$
	else:
	for $i = S$ to $\lvert I \rvert - 2$ do:
	for $j = i + 2$ to $\lvert I \rvert$ do:
	if $cross(i, j) = 0$
	$I \leftarrow I \backslash \{i, i + 1, \ldots j\}$
	$Recursive\ (S, T, I)$

5. Simulation and Solution

After the A* algorithm calculates an optimal path, it does not necessarily meet the minimum turning points. Meanwhile, due to the use of grid based dimensionality reduction, some feasible paths are not taken into account. Therefore, the most important judgment of whether an edge can be adjusted is whether any adjusted edge will intersect or be tangent to the obstacle. If the adjusted edge does not intersect with the obstacle, we have designed a recursive algorithm based adjustment turning point and optimization path.

5.1. Parameter Description

First, consider a radar with a detection radius of 12 nautical miles. Let the ship have a clear current position S and a destination D. Draw a circle with a radius of 12 nautical miles centered at the current position to determine the range currently detected by the radar. Then, obtain the intersection point of the detection range circle and the starting point, which is referred to as the current ideal destination point T. The line connecting the starting point within the circle is referred to as L, and the length of L is equal to the radar's detection radius of 12 nautical miles. A rectangular area is then created using L as the diagonal, representing the area of the nautical chart that needs to be gridded. This area is then divided into blocks to obtain a 20×20 grid image. Since the diagonal of the gridded area is 12 nautical miles, the length and width of the divided grids are approximately 800 meters. Assuming the ship length is 150 meters, which is relatively small compared to the grid size, the impact of the ship's length can be ignored.

Next, mark the obstacles on the nautical chart and the expanded area. Any region that cannot be connected to the feasible area on any side is identified as a restricted grid. Label the grid points occupied by the obstacles as shown in

Figure 5, which illustrates 8 common obstacles encountered during navigation.

Subsequently, the A* algorithm is used to analyze and optimize the path. The heuristic function of the A* algorithm is crucial, and it significantly influences the estimated cost $f(n)$. We utilize the commonly used Euclidean distance as the heuristic function in the A* algorithm. The distance of any two cell can be calculated by, where (x_i, y_i) is the position of each cell.

$$\rho = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{1}$$

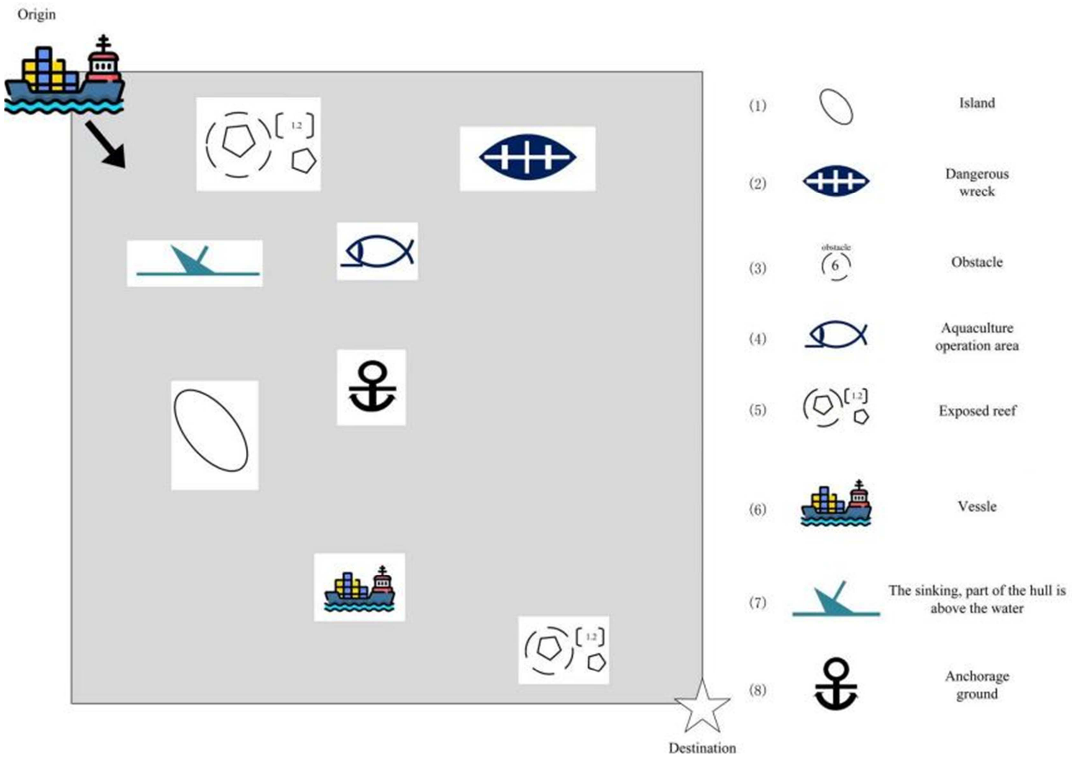


Figure 5. *Simulated Chart.*

5.2. Simulation

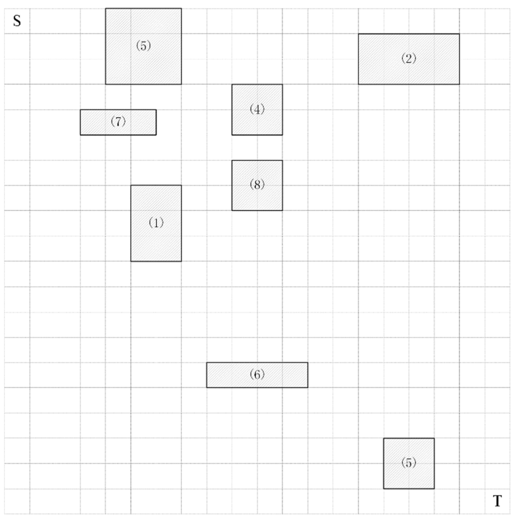
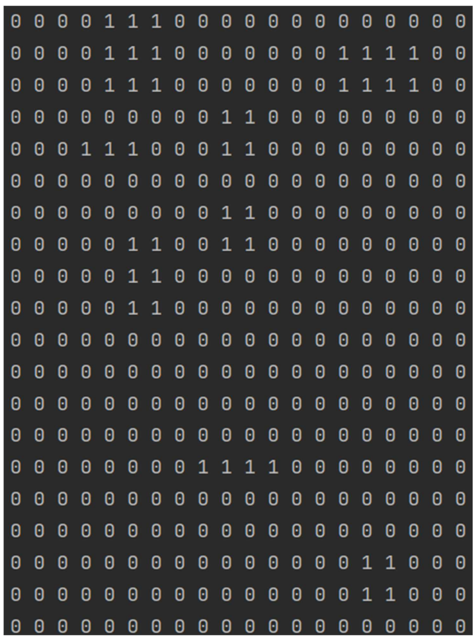


Figure 6. *Grid Expansion Chart.*

Step 1: The chart in Figure 5 is gridified, and the navigable and non-navigable areas are determined based on obstacles and their safety distances. As shown in Figure 6, the chart is gridified and inflated according to the radar detection range, where the gray area represents the non-navigable area. The starting point is located at the top left corner S, and the end point is located at the bottom right corner T.



```
0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 1 1 0 0 0 0 0 0 0 1 1 1 1 0 0
0 0 0 0 1 1 1 0 0 0 0 0 0 0 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 0 0 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Figure 7. *Simulated grid points via python.*

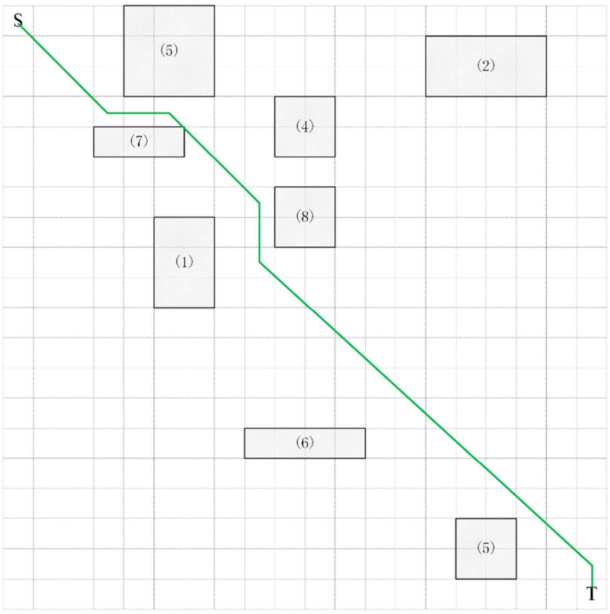


Figure 8. *Route generated by A*.*

Step 2: The chart is digitized and input into the Python simulator, resulting in Figure 7, which illustrates a 20*20 matrix. Each number represents a grid, where '0' indicates a navigable area and '1' indicates a non-navigable area.

Step 3: The A* algorithm described in section 4.1 is invoked for the initial calculation, yielding the shortest path from the starting point to the end point, as shown in Figure 8. It is observed that this path is entirely within the navigable grids.

Step 4: The recursive algorithm described in section 4.2 is utilized to optimize the original path. Figure 9 displays the newly optimized path, where it is evident that both the number of turning points and the travel path have been reduced compared to the original path. The distance has been reduced by 578.49 m, and the turning points have been reduced by 2. Although the new route crosses the inflated non-navigable areas (7) and (5), it can be observed from the right side of Figure 9 that the new route remains in a safe position.

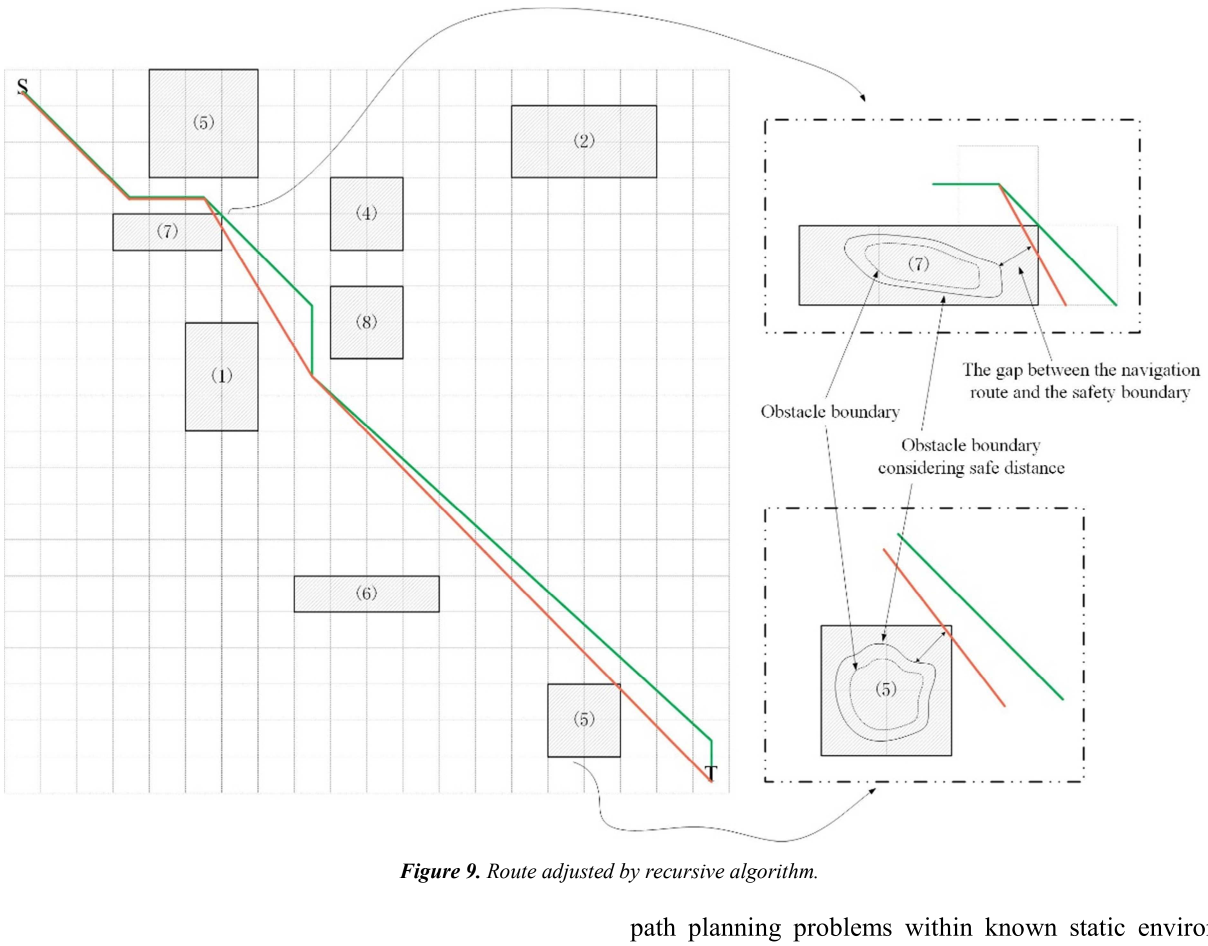


Figure 9. *Route adjusted by recursive algorithm.*

6. Conclusion

This paper primarily addresses the allocation of traversal factors to raster charts to facilitate the application of the A* algorithm and recursive algorithms in solving optimal ship passage routes in two stages. It focuses on the exploration of path planning problems within known static environments, aiming to achieve the shortest route, minimize the number of turning points, and ensure smoothness in offshore routes. Multiple computational modes and global planning methods are employed to eliminate redundant turning points, utilizing a spatiotemporal processing strategy to meet practical requirements.

The paper concludes with an experimental simulation to test the designed path planning method, and the results affirm the effectiveness and feasibility of the improved algorithm. The study delves into route planning for offshore navigation, identifying several areas that warrant further improvement. It explores methods for reducing the dimensionality of paths traversing prohibited areas during rasterization, making them more manageable. The paper proposes global obstacle processing, enabling even small portions of grid points to be marked as prohibited areas. These can be further detailed into grids or labeled as partially passable areas, allowing for selective ship passage using other algorithms. The study is primarily limited to offshore routes and does not fully address the feasibility of navigating areas that require selective low-speed passage, such as shoals.

This paper employs a distinct approach to regional planning, deviating from the commonly used Markov diagram method, by utilizing the grid method. It involves gridding the selected area and designating obstructive objects as no-go zones, thereby enabling a more precise selection and planning of optimal paths. Subsequently, mathematical models are constructed using the A* algorithm and recursive algorithms. To address situations where vessels might be too close to no-go areas, resulting in radar alarms and the like, adjustments to turning points are made. This ensures that obstructive objects are kept at a tangent distance within radar safety limits, thus enhancing safety during the navigation process.

The research acknowledges that new challenges continually emerge in path planning, mirroring the endless challenges ships face at sea. As technology advances, more advanced planning methods are expected to revolutionize route planning, ultimately enhancing maritime transportation.

Conflicts of Interest

This research paper is based on the research task titled "Coastal Complex Waterway Route Planning and Navigation Safety Assessment" conducted by the Port and Shipping Demonstration Research Team at Shanghai Maritime University. Authors Dongkui Wu and Lixiong Chen participated in on-site investigations and research activities related to coastal ports as members of the research team. Additionally, Fan Yang, as a team member, contributed to the relevant research work, designing and calculating on the Python platform, providing assistance. The research findings are based on objective and impartial analysis. The study was conducted independently and scientifically, with no identified potential conflicts of interest.

References

[1] GUO D D, YIN Y, XIAO F B. Overview of intelligent ship route optimization methods [J]. Chinese Journal of Ship Research, 2023, 18(4): 151–161

[2] YAN X P, WANG S W, MA F. Review and prospect for intelligent cargo ships [J]. Chinese Journal of Ship Research, 2021

[3] Lv Chao, Hu Qinyou, Xiang Zhe, Zhou Jia. Algorithm and Implementation of Recommended Routes for Major Coastal Ports in China [J]. Journal of Shanghai Maritime University, 2015.

[4] DU Zhixiao GUO Kun MA Shuo WANG Yuxi. Design of an Intelligent Navigation Shore-based Meteorological Service System [J]. Ship Electronic Engineering, 2019.

[5] Zhou Yinfei, Zhang Lihua, Jia Shuaidong, Dai Zeyuan, Dong Jian, Ma Mengkai. Autonomous Navigation Route Planning Method of Unmanned Ship Based on Bessel Curves Constrained by Maximum Navigable Window Sequence [J]. Geomatics and Information Science of Wuhan University, 2022.

[6] Pan Mingyang, Liu Yisai, Li Qi, Li Chao, and Chen Zhitai. Inland Waterway Route Planning and Application based on Improved A* Algorithm [J]. "Journal of Shanghai Maritime University" 2020.

[7] Cui Kangjing, Zheng Yuanzhou, Chen Guocheng, Hu Weidong. Optimization Design of Ship Meteorological Routes in Heavy Wind and Wave Environments [J/OL]. Journal of Wuhan University of Technology (Transportation Science and Engineering Edition), 2022.

[8] Chen Xiao, Dai Ran, Zhao Yanpeng, Zhang Chaoyue. Design of Ship Shallow Avoidance Route Based on Fish Swarm Algorithm [J]. China Navigation, 2019.

[9] Liu Zhifang. Application of Genetic Algorithm in Ship Route Adaptive Controller [J]. Ship Science and Technology, 2017.

[10] Chen Xiao, Dai Ran, Chen Changyuan. Route Planning Based on Maklink Graph and Ant Colony Algorithm [J]. China Navigation, 2017.

[11] Zhu Qing, Research on Ship Route Design Method Based on Ant Colony Algorithm [D]. Harbin Engineering University, 2011.

[12] Yao Xiaoxiao, Hu Qinyou, Yang Chun. Ship Route Planning Based on Ant Colony Algorithm and Massive AIS Data [J]. Transportation Information and Safety, 2019.

[13] Xie Xinlian, He Ping, He Ao, Xin Jianying. Collision Avoidance Path Planning of Ships in Complicated Water Areas [J]. Journal of Chongqing Jiaotong University (Natural Science), 2019.

[14] Yu Bixiu, Chu Xiumin, Liu Chenguang, Zhang Hao, Mao Qingzhou. A Path Planning Method for Unmanned Waterway Survey Ships Based on Improved A* Algorithm [J]. Geomatics and Information Science of Wuhan University, 2019.

[15] Wang Lipeng, Zhang Zhi, Ma Shan, Wang Xuewu. Improved Genetic Algorithm for Route Planning Considering Ship Maneuverability Constraints [J]. Journal of Harbin Engineering University, 2021.

[16] Tong Bangyu, Hu Jiankun. Ship Route Planning in Ice-Covered Areas Based on Improved Ant Colony Algorithm [J]. Navigation of China, 2020."