

Article

An Improved A-Star Ship Path-Planning Algorithm Considering Current, Water Depth, and Traffic Separation Rules

Rong Zhen ^{*}, Qiyong Gu, Ziqiang Shi and Yongfeng Suo

Navigation College, Jimei University, Xiamen 361021, China

* Correspondence: zrandsea@163.com; Tel.: +86-15080331250

Abstract: The influence of the maritime environment such as water currents, water depth, and traffic separation rules should be considered when conducting ship path planning. Additionally, the maneuverability constraints of the ship play a crucial role in navigation. Addressing the limitations of the traditional A-star algorithm in ship path planning, this paper proposes an improved A-star algorithm. Specifically, this paper examines the factors influencing ship navigation safety, and develops a risk model that takes into account water currents, water depth, and obstacles. The goal is to mitigate the total risk of ship collisions and grounding. Secondly, a traffic model is designed to ensure that the planned path adheres to the traffic separation rules and reduces the risk of collision with incoming ships. Then, a turning model and smoothing method are designed to make the generated path easy to track and control for the ship. To validate the effectiveness of the proposed A-star ship path-planning algorithm, three cases are studied in simulations and representative operational scenarios. The results of the cases demonstrate that the proposed A-star ship path-planning algorithm can better control the distance to obstacles, effectively avoid shallow water areas, and comply with traffic separation rules. The safety level of the path is effectively improved.

Keywords: ship global path planning; A-star algorithm; navigational safety; path optimization



Citation: Zhen, R.; Gu, Q.; Shi, Z.; Suo, Y. An Improved A-Star Ship Path-Planning Algorithm Considering Current, Water Depth, and Traffic Separation Rules. *J. Mar. Sci. Eng.* **2023**, *11*, 1439. <https://doi.org/10.3390/jmse11071439>

Academic Editor: Markel Penalba

Received: 30 June 2023

Revised: 14 July 2023

Accepted: 17 July 2023

Published: 18 July 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Ships are essential carriers in maritime transportation and play a crucial role in the transportation network [1]. Additionally, the integration of intelligent technologies, such as autonomous systems and advanced data analytics, into ships has become an inevitable advancement in maritime transportation [2]. In recent years, there has been rapid development in the technology of intelligent ships, which has garnered unprecedented attention in both military and commercial sectors [3,4]. The utilization of intelligent ships offers numerous benefits, such as reduced labor costs, energy savings, and fewer accidents [5]. Furthermore, these ships can effectively perform complex and hazardous engineering tasks in specific waters [6]. Research on intelligent ship autonomy has gained increasing attention as they heavily rely on highly autonomous systems.

Path planning plays a vital role in the development of autonomous systems for ships. It serves as the foundation for ship systems [7]. A safe and efficient path is essential for ships to ensure routine safe navigation, dynamic collision avoidance, and avoidance of grounding areas [8]. Path planning entails determining a secure and effective route from the starting point to the destination while considering specific requirements such as path length, risk factors, and rule constraints [9,10]. Path planning can be broadly classified into global path planning and local (real-time) path planning [11]. Global path planning calculates paths in advance in a static environment with stationary obstacles, whereas local path planning computes navigational paths in real time in a dynamic environment with both moving and stationary obstacles. Research on path planning is crucial for the advancement of autonomous systems in ships and serves to enhance their autonomy.

Despite the numerous proposed improvements in ship path-planning algorithms, many of these methods fail to sufficiently account for the impact of the marine environment on ship navigation [12,13]. The majority of existing path-planning algorithms prioritize enhancing algorithm performance and broadening their application scope. While safety-oriented path planning for unmanned vehicles has placed emphasis on assessing obstacle hazards, only some studies have thoroughly evaluated the risks associated with diverse factors in marine environments. As a result, there is a pressing need to develop a path-planning algorithm that considers various navigation factors to comprehensively enhance path safety.

The A-star algorithm has been extensively studied and utilized in the field of path planning due to its high performance and efficiency in most cases. At the same time, the A-star algorithm offers the advantage of heuristic search, which reduces the number of search nodes while ensuring an optimal path, thus improving search efficiency. However, the conventional A-star algorithm only takes into account the path length, making it unsuitable for ship navigation in complex sea environments. To effectively utilize the A-star algorithm for ship path planning, it is crucial to expand its functionality to incorporate multiple navigation safety factors, thus enhancing path safety. Additionally, the planned path must accommodate the dynamic motion characteristics of the ship. In global path planning, the minimum turning radius of the ship holds significant importance as a parameter of the ship's dynamic motion characteristics; therefore, it must be given due consideration. This paper aims to propose an improved A-star algorithm that enhances the security of global path planning. The contributions of this paper are as follows:

- (1) By analyzing the key factors that impact the safe navigation of ships, this paper establishes a risk model that comprehensively considers factors such as water current, water depths, and obstacle distances. The model aims to reduce the risk of collision with obstacles and prevent grounding.
- (2) This paper quantifies the traffic separation rules to establish a traffic model. The model enables the ship to adhere to traffic separation rules, reducing the risk of collision with incoming ships.
- (3) This paper proposes a turn model and a smooth method to enhance the smoothness of the path. The model optimizes the path on the basis of the ship's minimum turning radius to make it easier for the ship to track.

The remainder of this article is structured as follows: Section 2 introduces recent research on path-planning algorithms, including a detailed overview of the A-star algorithm. Section 3 explains the established methods used in the risk, traffic, and turn models. Section 4 presents the principles and specific methods for improving the A-star algorithm. Section 5 showcases three case studies. Lastly, Section 6 summarizes the conclusions and proposes directions for future research.

2. Literature Review

Ship path planning differs from path planning for robots, roads, and other applications because it considers the influence of the water environment and the maneuvering restrictions of the ship [14,15]. This section offers a concise overview of recent research advancements in ship path-planning algorithms and provides a detailed examination of the research conducted on the A-star algorithm in the context of ship path planning.

2.1. Research Progress of Ship Path Planning

Ship path planning plays a crucial role in achieving autonomous navigation [16]. Over the years, global path-planning methods have been primarily categorized into four groups: search algorithms based on existing map information, random expansion algorithms, intelligent bionic algorithms, and deep reinforcement learning algorithms. The Dijkstra algorithm [17] is a typical search algorithm that utilizes existing map information to determine the shortest path between two points. The A-star algorithm, which enhances the

efficiency of the Dijkstra algorithm by incorporating a heuristic function, has been widely employed in ship path planning [18].

The most commonly used random expansion algorithms in ship path planning are the rapidly exploring random trees (RRT) algorithm and the probabilistic road map (PRM) algorithm [19]. These algorithms obtain feasible paths by randomly sampling path nodes, which allows them to effectively solve path-planning problems with complex constraints without requiring accurate environmental modeling [20]. However, paths planned by these algorithms may not be suitable for tracking, and the convergence speed of the algorithms can be slow. To address the slow convergence of the RRT algorithm, Dong et al. [21] developed an environment framework that provides an initial path and guides the algorithm's expansion. Furthermore, Cao et al. [22] proposed an RRT algorithm enhanced with path shearing and smoothing modules to mitigate navigational risks in inland rivers. However, these algorithms do not consider the specific navigation characteristics of the ship and cannot guarantee that the path is the optimal solution.

Intelligent bionic algorithms, derived from bionic research, are employed to address path-planning challenges in complex environments. Commonly used algorithms include genetic algorithm (GA), ant colony optimization (ACO), and particle swarm optimization (PSO) [23]. Although these algorithms are robust, they often get stuck in local optima and are slow in planning. Thus, they are frequently used in combination with other algorithms [24]. To overcome the problem of GA getting trapped in the local optimum, Long et al. [25] introduced a new population initialization method with adaptive mutation and crossover probabilities to escape from the local optimal solution. Zhang et al. [26] suggested a differential evolution particle velocity approach, which effectively resolved the issue of PSO falling into the local optimum. Implementing these algorithms typically requires extensive environmental modeling, substantial computational resources, and parameter tuning. These requirements can pose challenges when applying them to practical path-planning scenarios.

The rapid advancement of autonomous ship navigation has rendered traditional algorithms insufficient to meet the demands of ship operations. As a result, researchers have increasingly turned to deep learning and reinforcement learning techniques for ship path planning [27]. Li et al. [28] enhanced the action space and reward function by incorporating marine collision avoidance rules into the reward function, and by introducing an artificial potential field. Simulation experiments demonstrated that the trained ship model can autonomously avoid collisions. Liu et al. [29] improved the efficiency and generalization capability of an algorithm by utilizing continuous multi-time target ship information and a redesigned reward function. Their approach, based on the TD3 algorithm, resulted in a smoother and more stable path. However, it is important to note that reinforcement learning algorithms require extensive training of ships to effectively plan paths. In reality, ship accidents are relatively rare, and available data are limited. Therefore, the application of such algorithms to real-world scenarios remains challenging.

2.2. Research Progress of A-Star Algorithm

The A-star algorithm is widely utilized in path planning due to its high level of completeness and optimality. However, when used in isolation, it can lead to paths that overlook environmental factors, exhibit poor algorithm efficiency, and do not align with real conditions [30]. To tackle these challenges, researchers have put forward various enhancements to the A-star algorithm.

Complex environments and large maps exponentially increase the time complexity of the A-star algorithm [31]. To tackle this challenge, researchers have made improvements to the search method of the A-star algorithm. Duchon et al. [32] enhanced the algorithm's efficiency by modifying the traditional eight-neighbor search to an omnidirectional search and incorporating the hop search algorithm into the path-planning process. Chen et al. [33] introduced a bidirectional A-star algorithm that utilizes a dynamic window to search for paths simultaneously from both the start and the goal positions. The dynamic window

significantly improved the algorithm's efficiency, resulting in a substantial reduction in search time. Fernandes et al. [34] improved the algorithm's efficiency by limiting node expansion and constructing 16 directional layers that only visit adjacent layers when searching for the lowest cost. Similarly, Zhang et al. [35] proposed an expansion method that prioritizes nodes in the same direction as the target node, reducing the number of expanded nodes in the A-star algorithm.

The A-star algorithm often produces paths with many turns and lacks smoothness [36]. Researchers have proposed various cost functions to smoothen paths and several post-smoothing algorithms for generated paths to address these issues. Thaker et al. [37] tackled the problem of excessive turning points in A-star algorithm paths by increasing the offset distance of obstacles using a buffer area centered on the robot. Experimental results confirmed that this method resulted in smoother paths. Lu et al. [38] introduced a path-planning algorithm that combined the A-star algorithm with the Floyd algorithm to reduce the sharpness of breakpoints in the planned path. Gunawan et al. [39] addressed the issue of large turning point angles in the traditional A-star algorithm paths by incorporating angle information between nodes to minimize the turning angle, thereby avoiding potential safety hazards associated with sudden turns of unmanned vehicles. Sun et al. [40] combined the A-star algorithm with the variable weight evaluation function of the artificial potential field (APF) method and optimized the relevant parameters using the PSO algorithm, resulting in smoother planned paths and enabling dynamic obstacle avoidance.

Environmental factors have a significant impact on global route planning for ships. Recognizing the limitation of the A-star algorithm in considering only path length, researchers have endeavored to incorporate various environmental factors in the maritime area that affect navigation. To address the issue of the A-star algorithm's path being too close to obstacles, Shu et al. [41] proposed the utilization of an obstacle detection method to select safer nodes. In their study, Liu et al. [42] improved the algorithm's cost function by considering the risk of water currents, presenting an A-star algorithm that addresses the combination problem of the normal path and the berthing path. Additionally, Liu [43] integrated environmental water depth interpolation into the algorithm and analyzed the ship motion characteristics to devise an A-star algorithm that takes into account water depth risk, effectively reducing the risk associated with water depth along the path.

In summary, researchers have made significant improvements to the A-star algorithm, resulting in increased efficiency and practicality. However, these improved algorithms often fail to guarantee safe navigation due to their limited consideration of environmental factors, ship turning restrictions, and traffic separation rules. While these algorithms may offer shorter paths, they also introduce higher navigation risks, making them unsuitable for intelligent navigation [44]. To address these limitations, this paper proposes an improved A-star algorithm that incorporates various navigation factors. The algorithm comprises risk model, traffic model, and turn model. The risk model takes into account the influence of current, water depth, and obstacle distance on navigation risks, thereby avoiding grounding areas and maintaining a safe distance from obstacles. The traffic model ensures that ships comply with maritime rules, reducing the risk of collision with incoming ships. Furthermore, to better align with the dynamic motion characteristics of the ship during the path-planning stage, the algorithm utilizes a geometric smoothing method. This method optimizes the path to accommodate the ship's turning radius, ensuring that it is in line with the ship's maneuverability. By considering these factors, the proposed algorithm enhances both the safety and the practicality of ship navigation.

3. Model Design

3.1. Overview of the Model

When planning a ship's path, it is crucial to consider the length of the path and various risks that can impact navigation safety [45]. Turbulent currents, grounding, ship encounters, and improper maneuvering are among the factors that can compromise the safety of navigation [46,47]. The influence of water currents on ship movement is particularly

significant in the ocean. The speed and direction of currents can have a profound impact on a ship’s course. Neglecting the influence of water currents during a voyage can lead to deviations from the intended path, resulting in unnecessary delays or increased risks. Water depth is another critical factor that affects navigation safety. Insufficient water depth can lead to grounding or damage to the ship. Therefore, when planning a ship’s path, it is essential to consider the limitations imposed by water depth to avoid potential safety issues. Furthermore, adherence to traffic separation rules is necessary in busy waterways to ensure safe and orderly navigation. Failure to comply with these rules can result in hazardous situations such as traffic congestion or collisions. Considering these factors, path planning should incorporate risk assessment to enhance navigation safety. By accounting for turbulent currents, water depth limitations, and traffic separation rules, the path can be optimized to minimize potential risks and ensure a safe and efficient voyage.

To focus on the given problem, certain assumptions were made in this paper. The overall map was assumed to represent a confined ocean environment near a harbor, resulting in a relatively short total travel distance and no significant environmental changes during the ship’s journey. Assuming a static environment is reasonable for short-duration travel of 1–2 h. This is because weather information is typically forecasted and updated on an hourly basis, with minimal changes expected during this period. Therefore, if the travel time is <1 h, it can be safely assumed that the environment is static.

To tackle these challenges, three models were developed: the risk model, the traffic model, and the turn model. Figure 1 illustrates the relationship between the risk factors and the respective models. By integrating these models into the path-planning process, we can prioritize safety and ensure the utmost security for ship navigation.

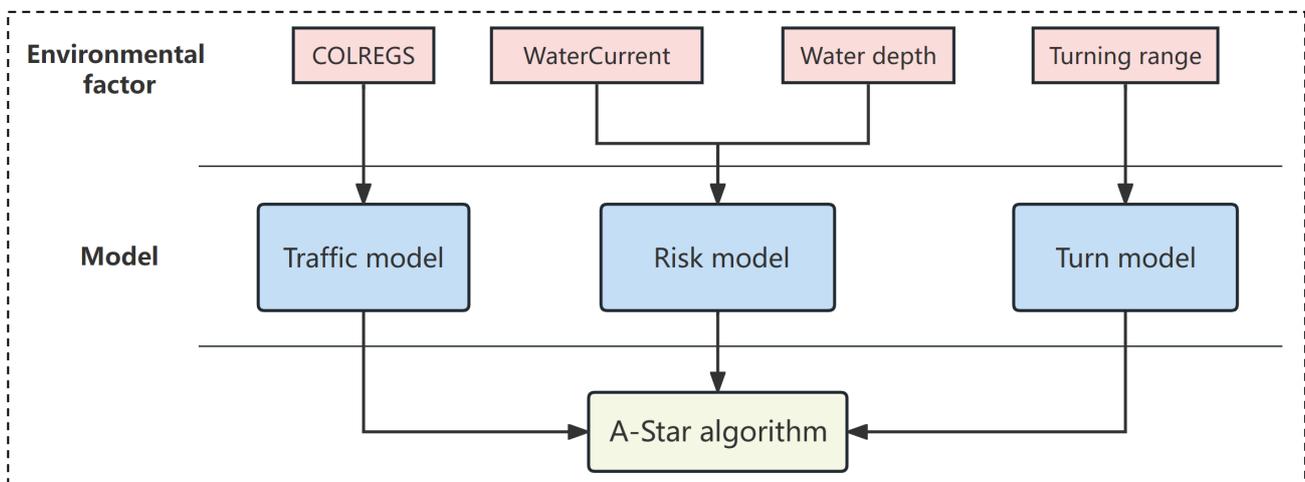


Figure 1. Correspondence between risk factors and models.

3.2. Risk Model

Although the traditional A-star algorithm can identify the shortest path, the generated path often remains very close to obstacles. This approach significantly increases the navigation risk due to the inherent time lag and inertia in ship movements. During navigation, if the water currents push the ship toward obstacles, the risk of collision further escalates. Furthermore, water depth plays a crucial role in safe navigation as it directly affects the risk of grounding. Therefore, in path planning, it is essential to simultaneously consider the proximity to obstacles, water depth, and the influence of water and wind currents.

On the basis of the analysis conducted earlier, this paper presents the following definition of the risk model:

$$r_s(m, n) = r_{obs}(m, n) + r_{depth}(m), \tag{1}$$

where $r_{obs}(m, n)$ denotes the risk from obstacles, and $r_{depth}(m)$ denotes the stranding risk from shallow water area. The purpose of designing the model $r_{obs}(m, n)$ is to maintain a certain safe distance from obstacles in the environment and prevent collisions between the ship and obstacles caused by wind, water currents, and other factors. The purpose of designing the model $r_{depth}(m)$ is to ensure that the ship can avoid grounding or potentially grounding areas. The expression for $r_{obs}(m, n)$ is as follows:

$$r_{obs}(m, n) = e^{-d}(1 + v_{cur}h_c), \tag{2}$$

where d is the distance from a navigable node $N[m]$ to the obstacle node $O[n]$. Furthermore, v_{cur} is the current velocity. It should be noted that this article replaced obstacle expansion with the distance function in Equation (2). This was to prevent the algorithm from being trapped in a locally optimal solution in a narrow area. Lastly, h_c denotes the direction coefficient, expressed as follows:

$$h_c = \begin{cases} -v_U \cos(\theta_{cur} - \theta_{mn}), & \cos(\theta_{cur} - \theta_{mn}) < 0 \\ 0, & \cos(\theta_{cur} - \theta_{mn}) \geq 0 \end{cases} \tag{3}$$

where v_U is the ship velocity, and θ_{cur} represents the angle between the water current v_{cur} and the horizontal direction. As shown in Figure 2, θ_{mn} is the angle between vector $\overrightarrow{N[m]O[n]}$ and the horizontal direction, while $\overrightarrow{N[m]O[n]}$ represents the angle formed between the current position of the ship $N[m]$ and the obstacle $O[n]$.

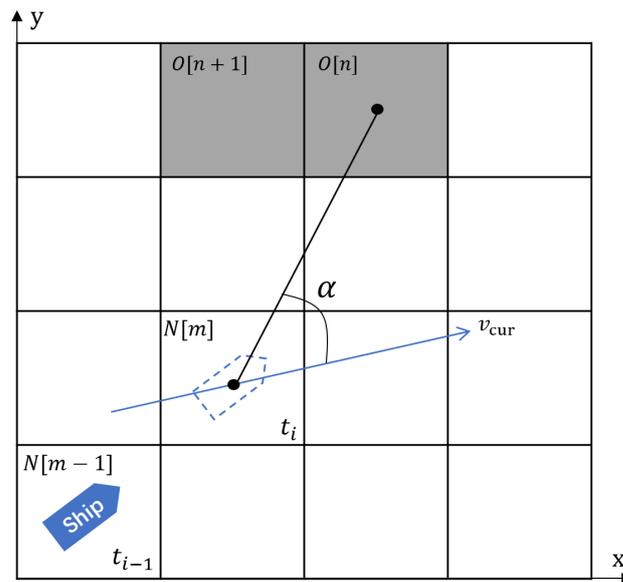


Figure 2. Schematic diagram of calculating risk.

In Figure 2, assuming that node $N[m]$ is the node where the ship sails at time t_i , it can be observed that at this moment the ship has collision risks with obstacles $O[n]$ and $O[n+1]$. α represents the angle between the vector $\overrightarrow{N[m]O[n]}$ and the water current v_{cur} . By calculating α , the risk value at node $N[m]$ can be computed using Equation (2).

To ensure that the ship can avoid grounding areas, we define $r_{depth}(m)$ as follows:

$$r_{depth}(m) = \begin{cases} \frac{S_{min}}{D(m)}, & S_{min} < D(m) \\ inf, & D(m) \leq S_{min} \end{cases} \tag{4}$$

where $D[m]$ is the water depth at node $N[m]$, and S_{min} is the maximum draft of the ship, expressed as follows:

$$S_{min} = z_{max} + 0.5L \tan \theta_{max} + \bar{T} + e_{enc}, \tag{5}$$

where z_{max} is the maximum settlement amplitude of ships at different velocity, L is the length of ships, θ_{max} is the maximum pitch angle, \bar{T} is the average draft under the mission load, and e_{enc} is the calculation error.

According to Equation (1), there will be k obstacle nodes near a feasible node $N[m]$, corresponding to k different risk values. In this case, the maximum value is taken as the final risk degree.

$$r_s(m, n) = \max\{r_s(m, 1), r_s(m, 2), r_s(m, 3), \dots, r_s(m, k)\}. \tag{6}$$

In addition, a local search is used instead of a global search to save time. The search range is set to d_g , as shown in Figure 3.

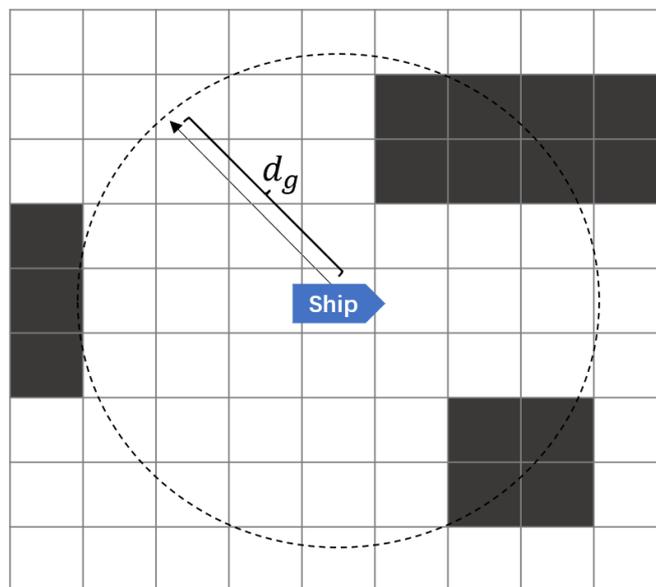


Figure 3. Schematic diagram of local search.

3.3. Traffic Model

Similar to road traffic, ship navigation is also governed by traffic rules at sea. The International Regulations for Preventing Collisions at Sea (COLREGS) mandate that ships should cross separation zones in a direction perpendicular to the separation zone whenever possible. This rule aims to clarify the intention of crossing, minimize the risk of collision with other ships within the separation zone, and enhance navigation efficiency. However, the conventional A-star algorithm does not inherently ensure that the planned path aligns with the specified direction required by the traffic separation rules. To address this issue, we define the following traffic model on the basis of the rules:

$$r_{tra}(m) = \begin{cases} 1 - \cos(\theta_t - \theta_{ship}), & \text{if } \cos(\theta_t - \theta_{ship}) \geq 0 \\ \text{math.inf}, & \text{if } \cos(\theta_t - \theta_{ship}) < 0 \end{cases} \tag{7}$$

where $r_{tra}(m)$ is the traffic separation cost at node $N[m]$, θ_t is the direction of traffic rules toward the true north, $N[m].f$ is the parent node of $N[m]$, and θ_{ship} is the direction of path vector $\overrightarrow{N[m]O[m].f}$ toward the true north, which is the ship's driving direction. As shown in Figure 4, the orange horizontal line denotes the traffic separation zone. The orange arrow is the driving direction specified by the traffic rules, where $\alpha = \theta_t - \theta_U$ is used to calculate

whether the currently planned path conforms to the specified direction. For example, if the ship continues to follow the path in Figure 4 at time t_{i+1} , the ship will violate the traffic rules at sea.

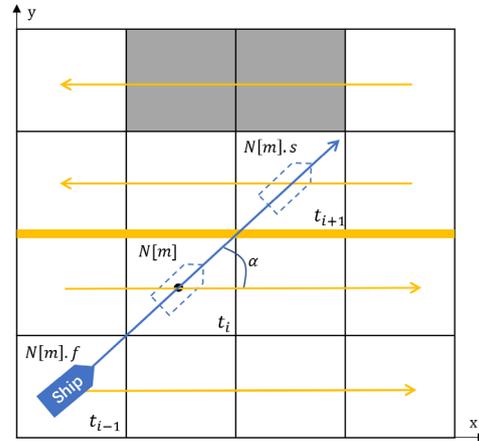


Figure 4. Schematic diagram of the traffic model.

3.4. Turn Model

In actual ship navigation, frequent turning can increase the difficulty of the captain’s maneuvering and pose a risk to navigation. However, traditional A-star path-planning algorithms only consider the path length and often include many turning nodes. To address this issue, this paper presents a turn model to reduce the number of turns in the planned path. The turn model is defined as follows:

$$r_{turn}(m) = \arccos \left(\frac{\overrightarrow{N[m]N[m].s}}{\overrightarrow{N[m].fN[m]}} \right), \tag{8}$$

where $r_{turn}(m)$ is the turning cost at node $N[m]$, $N[m].s$ is the child node of node $N[m]$, and $N[m].f$ is the parent node of node $N[m]$. As shown in Figure 5, the ship is at node $N[m]$ at time t_i . The turning cost here is estimated by calculating the angle β_i . β_i is the angle of vector $\overrightarrow{N[m]N[m].s_i}$ toward the vector $\overrightarrow{N[m].fN[m]}$. A greater turning range at node $N[m]$ attracts a greater turning cost. This model not only reduces the number of turns but also limits the scope of turns to a certain extent.

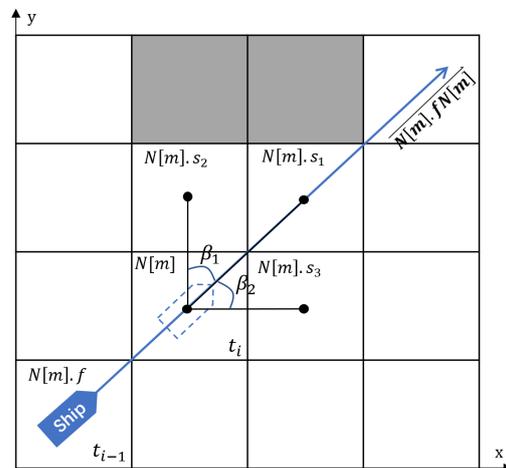


Figure 5. Schematic diagram of the turn model.

4. A-Star Algorithm and Improvements

This section begins by introducing the rules of environmental modeling, which can help to ensure that the A-star algorithm plans a path that is safer, better suits ship tracking, and adheres to traffic separation rules. Next, we propose an improved A-star algorithm based on the model presented in Section 3.

4.1. Environment Modeling

To accurately represent the path-planning process, it is necessary to create a two-dimensional environment map. There are several conventional methods for environmental modeling, including the grid, geometric information, and view methods [48]. The grid method involves dividing a planar map into a series of grids to create a grid map. This method is efficient in representing the characteristics of the actual environment while optimizing time and space consumption. It is also simple and direct, reducing the path search time and simplifying programming. Therefore, the grid method is utilized in this study to model the navigation environment.

(1) Water area and water depth division standard

The division of grids into navigable and non-navigable areas can be based on the boundaries of the environment. A grid that contains objects is considered non-navigable and denoted in black, representing non-navigable waters. On the other hand, a grid that does not contain any objects is considered navigable and denoted in white, representing navigable water areas, as illustrated in Figure 6a. Obstructions that occupy less than one grid are expanded to ensure grid regularity and facilitate subsequent simulation implementation. The navigable and non-navigable waters after expansion are depicted in Figure 6b. The method for establishing the water depth environment follows the same process as above, as shown in Figure 6c,d.

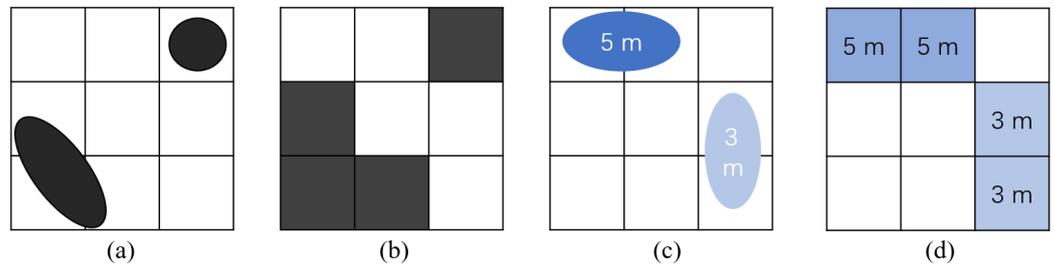


Figure 6. Obstacles division and water depth standards.

(2) Movement rules of ships in the grid environment

Ships can only move within the white grids representing feasible water areas and cannot cross or appear in the black grids. At the same time, the ship can move in eight directions within the grid environment, as illustrated in Figure 7.

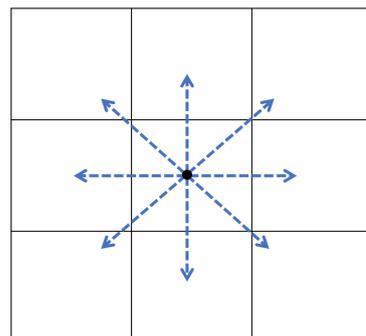


Figure 7. Schematic diagram of eight neighborhoods.

It is worth noting that the algorithm proposed in this paper is also applicable to the Electronic Nautical Chart (ENC) environment. The specific steps are to import the ENC data into the computer for preprocessing and establish the environmental model of the electronic chart on the basis of the processed ENC data. The environmental model is established using the grid map method, with each grid containing data on water depth, water currents, wind currents, and traffic separation rules. Lastly, path planning is performed using the constructed environment and the improved algorithm, and the resulting path nodes are displayed on the electronic chart [49].

4.2. Improved A-Star Algorithm

Given the limitations of traditional A-star algorithms in ship path planning, this section explores an improved A-star algorithm based on the risk models developed in Section 3.

4.2.1. Traditional A-Star Algorithm

The A-star algorithm is a heuristic algorithm based on the graph method, and it is also the most effective direct search algorithm for finding the shortest path in static road networks. Due to its high accuracy and efficiency, it is widely used in global path planning. The algorithm searches for the path by calculating the cost function of each node in the field, which has the following form:

$$F(N[i]) = g(N[i]) + h(N[i]), \tag{9}$$

where $g(N[i])$ is the actual cost function from $N[start]$ to $N[i]$, $h(N[i])$ is the cost estimation function from $N[i]$ to $N[target]$, and $F(N[i])$ is the cost estimation function from $N[start]$ to $N[target]$ through $N[i]$. Here, $h(N[i])$ has several forms, which are expressed by Euclidean distance in this paper, as follows:

$$h(N[i]) = \sqrt{(x_i - x_g)^2 + (y_i - y_g)^2}, \tag{10}$$

where (x_i, y_i) and (x_g, y_g) are the current node $N[i]$ and target node $N[target]$ position coordinates, respectively.

4.2.2. Improved A-Star Algorithm

To incorporate the risk model, traffic model, and turn model into the A-star algorithm, this paper introduces a redesigned cost function. The improved cost function, denoted as $F(N[i])$, is defined as follows:

$$F(N[i]) = g(N[i])' + \epsilon h(N[i]), \tag{11}$$

$$g(N[i])' = g(N[i]) + \pi r_n(N[i]), \tag{12}$$

where $\epsilon > 0$ is a constant coefficient. ϵ is used to balance the weight between $g(n)'$ and $h(n)$. $\pi > 0$ is a constant coefficient. $g(N[i])$ is defined as

$$g(N[i]) = \sum_{k=0}^{t-1} d(k), \tag{13}$$

where t is the node amount from $N[start]$ to $N[i]$. $r_n(N[i])$ is the risk function, which is defined as follows:

$$r_n(N[i]) = r_s(N[i]) + r_{tra}(N[i]) + r_{turn}(N[i]), \tag{14}$$

where $r_s(n)$, $r_{tra}(n)$, and $r_{turn}(n)$ were defined in Equations (1), (7) and (8).

The improved A-star algorithm was designed to consider various factors that affect navigation safety so that the planned path can meet several key characteristics. Firstly, it can maintain a safe distance from obstacles in the environment. Secondly, it can avoid shallow water areas and reduce the risks associated with currents. Thirdly, it can comply with traffic separation rules. Lastly, it can minimize the number of turns required. The pseudocode of Algorithm 1 demonstrates the implementation of the improved A-star algorithm.

Algorithm 1: Improved A-star algorithm

```

1:  Mark  $N[start]$  as openlist
2:  if not traffic separation rule then
3:     $r_{tra} = 0$  at any time
4:  While openlist  $\neq \emptyset$  do
5:    Select openlist  $N[i]$  whose value of  $F_n[i]$  is the smallest
6:    if  $N[i]=N[goal]$  then
7:      return "path  $P_n$  is found"
8:    else
9:      Mark  $N[i]$  as closelist
10:     if successor  $N_i[j]$  of  $N[i]$  not in closelist or openlist then
11:       Mark  $N_i[j]$  as openlist
12:       Calculate  $r_s(j)$ ,  $r_{tra}(j)$ ,  $r_{turn}(j)$  by (1), (7), and (8), respectively.
13:       if  $N_i[j]$  in openlist and  $F_{new}(N_i[j])$  is smaller than  $F_{old}(N_i[j])$  then
14:          $F_{old}(N_i[j]) = F_{new}(N_i[j])$  and set parent node of  $N[j]$  as  $N[i]$ 
15:     return "path  $P_n$  is not found"

```

4.2.3. Smooth Paths with Geometry

The map is modeled using a grid-based approach, and the planned path consists of line segments formed by the grid. However, the vertices of this path are not conducive to the tracking and smooth navigation of the ship. Since the dynamic motion characteristics of the ship play a crucial role in path planning, and the turning radius of the ship is an important parameter of its dynamic motion characteristics in global path planning, the planned path must be within the range of the ship’s maneuverability. To address this issue, this paper adopts the geometric smoothing path method to replace the vertices of the planned path with curve segments, taking into account the ship’s minimum turning radius as an important reference parameter. The optimization process is shown in Figure 8.

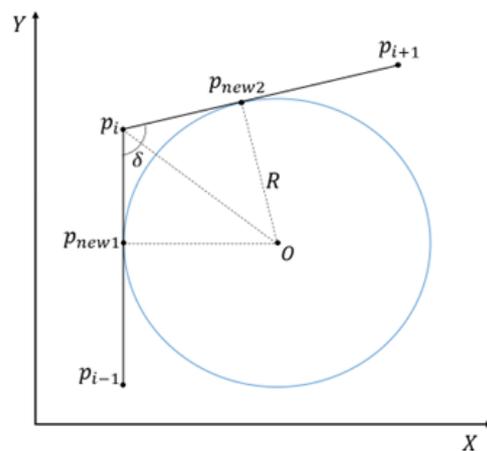


Figure 8. Principle of geometrically smooth paths.

As shown in Figure 8, $p_{i-1}(x_{i-1}, y_{i-1})$, $p_i(x_i, y_i)$, and $p_{i+1}(x_{i+1}, y_{i+1})$ are polyline segments on the planned path. We use a circle of radius R such that $p_{i-1}p_i$ and $p_i p_{i+1}$ are

tangents to the circle, intersecting at $p_{new1}(x_{new1}, y_{new1})$ and $p_{new2}(x_{new2}, y_{new2})$. The calculation process of p_{new1}, p_{new2} is as follows:

$$\delta = \arccos\left(\frac{\overrightarrow{P_{i-1}P_i}}{\overrightarrow{P_{i+1}P_i}}\right), \tag{15}$$

$$p_i p_{new1} = p_i p_{new2} = R \times \cot\left(\frac{\delta}{2}\right). \tag{16}$$

The radius of the circle is set to $R = p_{new1}O$, and the coordinate scale coefficient φ of the tangent point is set as follows:

$$\varphi_1 = \frac{p_i p_{new1}}{p_{i-1} p_i}, \quad \varphi_2 = \frac{p_i p_{new2}}{p_i p_{i+1}}. \tag{17}$$

From $p_{i-1}(x_{i-1}, y_{i-1}), p_i(x_i, y_i), p_{i+1}(x_{i+1}, y_{i+1})$, and the coordinate scale coefficient φ , we get

$$p_{new1} = (\varphi_1 \times x_{i-1} + (1 - \varphi_1) \times x_i, \varphi_1 \times y_{i-1} + (1 - \varphi_1) \times y_i), \tag{18}$$

$$p_{new2} = (\varphi_2 \times x_{i+1} + (1 - \varphi_2) \times x_i, \varphi_2 \times y_{i+1} + (1 - \varphi_2) \times y_i). \tag{19}$$

The slopes and formulas of the straight line $p_{i-1}p_i$ and the straight line $p_i p_{i+1}$ are

$$k_1 = \frac{y_i - y_{i-1}}{x_i - x_{i-1}}, \quad y_1 = k_1 \cdot x - \frac{x_{i-1}y_i - x_i y_{i-1}}{x_i - x_{i-1}}, \tag{20}$$

$$k_2 = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}, \quad y_2 = k_2 \cdot x - \frac{x_i y_{i+1} - x_{i+1} y_i}{x_{i+1} - x_i}. \tag{21}$$

From the slope k_1 and the point p_{new1} , the vertical straight line y_1 and the straight line y_1' passing through the point p_{new1} can be obtained. In the same way, y_2' can be obtained (y_2' passes through p_{new2} and is perpendicular to y_2). The intersection of the straight lines y_1' and y_2' is the center of the circle, and the center of the circle is set to $O(x_o, y_o)$; then, the function expression of the curve after smooth geometric optimization is as follows:

$$f(x) = \begin{cases} y_1, & | x_{i-1} \leq x \leq x_{new1} \\ \pm\sqrt{R^2 - (x - x_o)^2} + y_o, & | x_{new1} < x < x_{new2} \\ y_2, & | x_{new2} \leq x \leq x_{i+1} \end{cases} \tag{22}$$

In practical applications, we can adjust the curvature of the smooth curve by changing the size of R , so that the curvature of the optimized curve satisfies the minimum turning radius of ships.

5. Case Study

Simulation experiments were conducted in this section to validate the effectiveness of the proposed improved A-star path-planning method. First, the path-planning performance of the proposed A-star method was compared with traditional A-star methods, considering obstacles, water depth, water currents, and traffic separation rules. Then, the proposed improved A-star method was tested in real scenes in Zhoushan and Hainan ports. It should be noted that the environmental data for Case 2 and Case 3 were obtained from shipxy.com. In case 2, nautical charts were used for path planning instead of satellite maps to illustrate the bathymetric boundaries and marine traffic diversion areas. In case 1 and case 2, the China-made autonomous cargo ship, 'Jindouyun 0', was selected for the simulation experiment. The ship's key parameters are shown in Table 1. The minimum safe navigation depth was calculated on the basis of the ship's key parameters.

Table 1. Ship key parameters.

Parameters	Value
Ship length	12.86 m
Ship width	3.8 m
Design maximum draft	1 m
Minimum turning radius R_0	36 m
Design velocity	8 knots

5.1. Case 1: Path Planning in Complex Simulation Environment

For this study, we conducted simulation experiments on the Python 3.8 platform and used a 50×40 grid map for the experiment. In order to simulate realistic environmental conditions, we created a complex environment that included current velocity, shallow water areas, and traffic separation zones.

5.1.1. Setup

The relevant model parameters need to be determined to establish the proposed model. The velocity of the current is set to $v_m = 2\text{m/s}$. The direction of the current is due east. The ship velocity is set to $v_U = 4\text{m/s}$. $N[start]$ is set to (5, 8), and $N[target]$ is set to (40, 35). The grid length is set to 20 m. The depth of shallow water is set to 1 m. Other parameters of the simulation experiment are shown in Table 2. In order to verify the effectiveness of each model, this section designs a path for each model. Using the model established in Section 3, the following four different cost functions are used for simulation experiments: (1) Path1 is the traditional A-star algorithm; (2) Path2 verifies the validity of the risk model, where $r_n(N[i]) = r_s(N[i]) + r_{turn}(N[i])$; (3) Path3 verifies the validity of the traffic model, where $r_n(N[i]) = r_{tra}(N[i]) + r_{turn}(N[i])$; (4) Path4 simultaneously considers all models in Section 3, i.e., $r_n(N[i]) = r_s(N[i]) + r_{tra}(N[i]) + r_{turn}(N[i])$.

Table 2. Parameter initialization.

Parameters	Value
Node range of x-axis	[1, 50]
Node range of y-axis	[1, 40]
Grid length l	20 m
Minimum radius of ship R_0	36 m
π	0.2
ϵ	0.5

5.1.2. Results

Figure 9 compares the four different path-planning results. Path1 is located close to the obstacle and passes through a shallow water area, violating traffic rules and increasing the risk of grounding. Moreover, Path1 has many sharp turns that could lead to ship collisions and groundings. Path2 starts planning the path toward the upper left direction from the starting point, aiming to move away from obstacles and reduce the risk of water currents pushing the ship toward them. This demonstrates the effectiveness of the proposed risk model in this paper. However, Path2 does not comply with the rule of separated traffic, which could increase the risk of collision with oncoming ships when navigating in the opposite direction. Path3 follows the separation rule by driving to the right in the lower part of the map and crossing the separation zone perpendicularly in the upper part of the map to reach the destination. This demonstrates the effectiveness of the traffic model proposed in this paper. However, Path3 is still too close to the obstacle, which is due to the lack of consideration of the risk model. Path4 is a path planned using the improved algorithm proposed in this paper that considers multiple risk factors. This path keeps a safe distance from the obstacle, avoids shallow water areas, and reduces the risk of ship navigation. Path4 also adheres to traffic separation rules, with fewer turns and a smoother trajectory.

With the aid of geometrically smooth curves, the optimized path ensures that the curvature of turns remains within the maneuverability range of the ship, making it more suitable for safe navigation and tracking. The proposed algorithm comprehensively improves the safety of path planning in complex marine environments, making the planned route highly suitable for ship navigation, and demonstrating its superiority over the traditional methods.

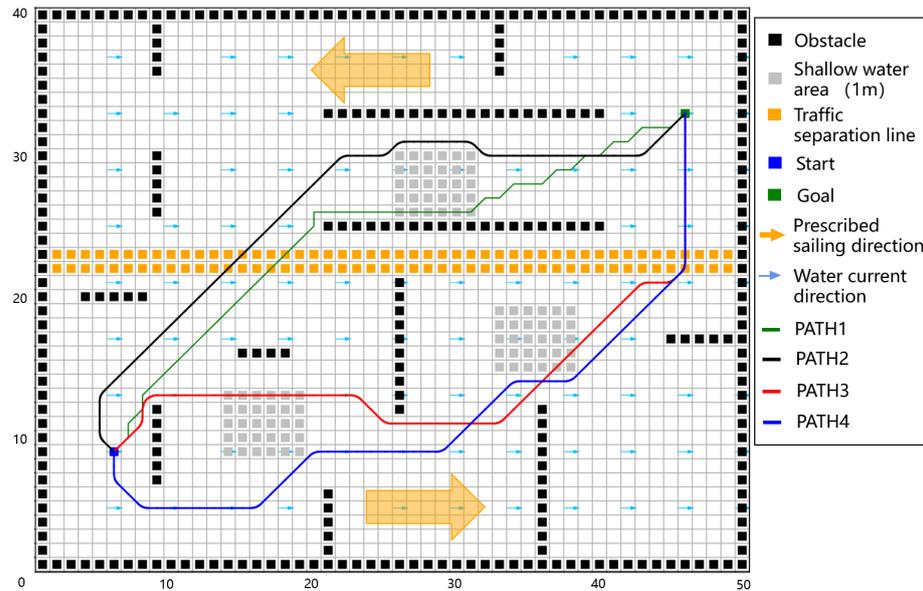


Figure 9. Simulation results of four different cost functions.

To further demonstrate the superiority of the improved algorithm, we calculated risk indicators for two paths: navigation risk $r_s = r_s(m, n)$. We compared path length, navigation risk, number of turns, maximum turn angle, and adherence to traffic separation rules among the four paths. The risk calculation was based on Equation (1) of the risk model.

The simulation results of the statistical experiment are presented in Table 3. Comparing the data, we can observe that Path1 planned by the traditional A-star algorithm has the shortest length. However, it has significantly higher risks of ship grounding and collision, as well as more turns, indicating a higher navigation risk compared to the algorithm proposed in this paper. Path2 also considers the risk model and has a navigation risk close to Path4. Path3 complies with the traffic model and conforms to traffic rules, but its proximity to the obstacle increases the risk of ship navigation. Paths2–4 have fewer turns compared to Path1, as they all consider the turn model. Path 4 significantly reduces various risks at the expense of a certain path length. Meanwhile, the path is smooth and conforms to traffic separation rules. The turning radius at the path’s bends is within the ship’s handling capabilities. In summary, the advantages of Path4 prove the effectiveness of the improved A* algorithm.

Table 3. Comparison of experimental simulation data.

	Path Length (m)	Collision Risk r_s	Turn Times (n)	Compliance with Traffic Separation Rules
Path1	516.9	30.11	19	No
Path2	545.2	13.53	7	No
Path3	592.1	18.58	8	Yes
Path4	608.7	9.39	8	Yes

5.2. Case 2: Path Planning in Real Scenes in Zhoushan Port

In this scenario, the effectiveness of the improved A-star method is verified using a real scene of the hidden reef in Zhoushan Harbor. The ship intends to travel from a starting point (29°56.551 N, 122°13.826 E) to a destination point (29°56.96 N, 122°13.60 E). The area is characterized by numerous hidden reefs that obstruct ships from passing through. In addition, several shallow areas in this range affect navigation, as depicted in Figure 10. The grid length is set to 25 m, and the velocity of the water current is set to $v_c = 1$ m/s. The direction of the water current is northwest. The geometric smooth path radius R is set to 50 m. It should be noted that the reef area does not adhere to traffic separation regulations due to the low volume of ship traffic in the area, and traffic separation rules are not taken into account in this case. The remaining parameter settings are the same as in Case 1.

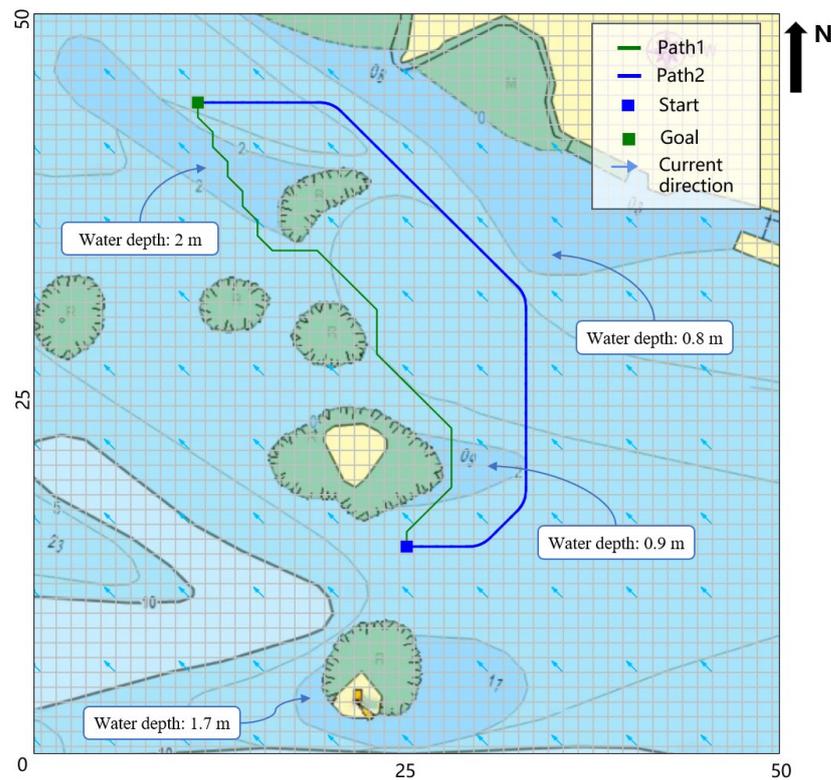


Figure 10. The path planned by the proposed A-star algorithm in Zhoushan port, China.

In Figure 10, Path1 and Path2 represent the paths generated by the traditional and improved A-star algorithms, respectively. From Figure 10, it can be observed that the path distance planned by the traditional A-star algorithm is close to the reefs. In complex marine environments, there is a high probability of collision with reefs due to the influence of water currents, which undermines the assurance of safe ship arrival at the destination. Furthermore, Path1 passes through two shallow water areas comparable to the ship's maximum draft, posing a high risk of grounding. Compared with Path1, Path2 avoids all shallow water areas and keeps a certain distance from obstacles to ensure that the ship will not collide with obstacles due to water currents. Table 4 compares various metrics for the paths. According to the experimental data, Path1 traverses shallow water areas below the maximum draft of the ship, resulting in an infinite risk of grounding. The navigation risk associated with Path1 is extremely high, making it unsuitable for ship tracking. On the other hand, Path2 takes into account multiple safety factors during navigation, exhibiting a higher level of safety and thus being more suitable for ship operations. Moreover, Path2 has a turning radius of 50 m (2 grids), well within the maneuverable range of the ship, ensuring a smooth trajectory. In summary, the improved A-star algorithm significantly reduces ship risks while slightly increasing the path length. Case 2 demonstrates the effectiveness of the

proposed model and highlights its practical significance in the context of complex shallow water ship path planning.

Table 4. Comparison of experimental simulation data.

	Path Length (miles)	Risk r_s	Turn Times (n)
Path1	0.52	inf	16
Path2	0.65	21.8	4

5.3. Case 3: Path Planning in Real Scenes in Hainan Port

In this case study, we choose the real scene of Hainan Port to verify the effectiveness of the improved A-star method. There are multiple traffic separation zones in this area, as shown in the rectangular area circled by the pink dotted line in Figure 11. The sides of the traffic separation zone dictate the opposite direction of travel, and ships that violate these rules run the risk of colliding with incoming ships. Therefore, it is crucial to ensure that planned routes respect traffic segregation regulations. In order to verify the effectiveness of the algorithm in this case, the “Yude Ship” is selected for simulation experiments. The “Yude Ship” has a length of 199 m, a full load draft of 12 m, and a minimum turning radius of twice the length of the ship. The planning start point (20°4.755 N, 110°8.501 E) and end point (20°15.392 N, 110°22.11 E) are set. The area is divided into 100 × 100 grids, and the length of each grid is set to 300 m. The water current velocity v_c is 2 m/s, and the direction is 10° northeast. The smooth path radius R is set to two meshes.

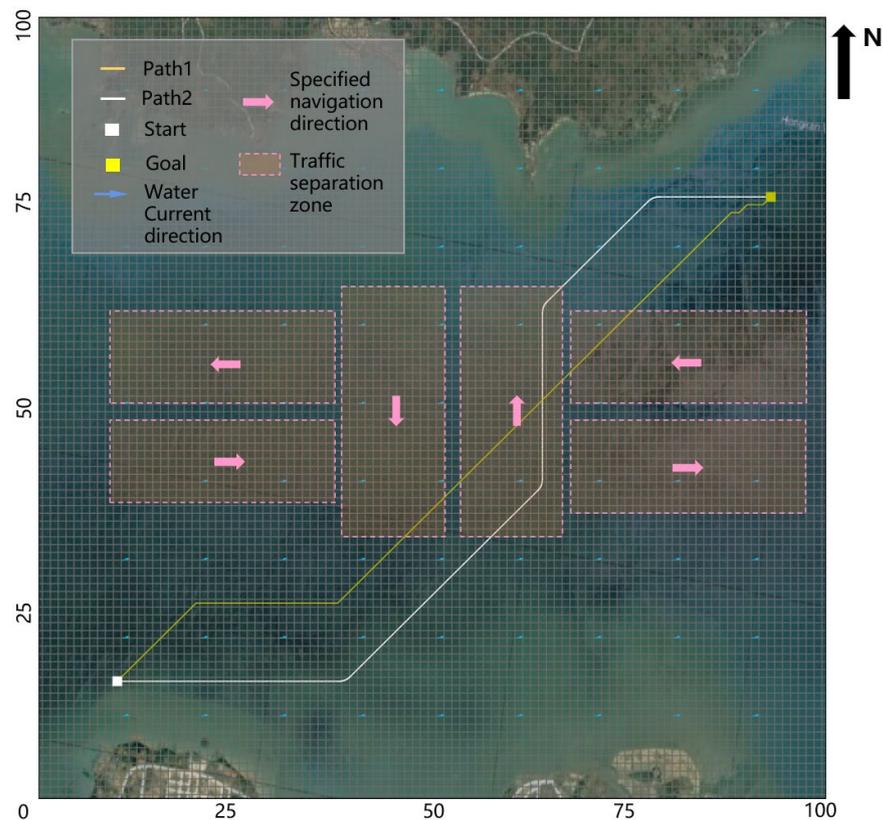


Figure 11. The path planned by the proposed A-star algorithm in Hainan port, China.

In Figure 11, Path1 and Path2 represent paths generated by the traditional A-star algorithm and the improved A-star algorithm, respectively. The area framed by the dotted line of the pink rectangle indicates the traffic control area where ships are required to obey the traffic separation rules. Pink arrows indicate the navigation direction for each

area. As shown in Figure 11, although both paths avoid all navigation markers and reach the destination safely, their planned paths are quite different. Path2 obeys the traffic separation rules, while Path1 violates the rules, and there is a path traveling in the opposite direction. Table 5 compares the experimental result data of the two paths. According to the experimental data, the reverse driving of Path1 violates the traffic rules, resulting in an infinite risk value. Path2 sacrifices a certain path length to make the planned path comply with traffic separation rules. The path planned by the improved A* algorithm improves the safety of the path and enables the ship to follow the prescribed channel. In summary, Case 3 verifies the effectiveness of the improved A-star algorithm.

Table 5. Comparison of experimental simulation data.

	Path Length (miles)	Risk r_s	Turn Times (n)	Compliance with Traffic Separation Rules
Path1	17.38	inf	6	No
Path2	19.54	25.8	4	Yes

6. Conclusions

This paper presented a ship path-planning approach that considers multiple safety factors. The proposed algorithm aims to enhance ship navigation safety by considering environmental effects, traffic regulations, and ship maneuvering constraints. It considers environmental factors such as water currents and water depth, as well as traffic regulations and the minimum turning radius of the ship. The effectiveness of the proposed A-star algorithm was demonstrated through three cases. The simulation results showed that the algorithm effectively considers multiple risk factors during navigation, maximizing the safety of the voyage. The planned paths not only comply with traffic regulations but also remain within the ship’s maneuvering capabilities, ensuring safe and efficient navigation. Additionally, the algorithm strikes a balance between path length and navigation safety, reducing the risks of ship collisions and groundings. These improvements to the A-star algorithm have significant potential for enhancing path-planning safety during ship navigation. The findings contribute to the field of ship navigation safety, benefiting the maritime industry and mitigating the risks associated with ship collisions and groundings.

With the rapid development of meteorology and measurement technology, more accurate environmental information can be forecasted. By utilizing this information, safer and more precise paths can be generated. The proposed path-planning system is generally applicable to ships of any size, as most parameters such as ship dimensions, maneuvering constraints, and water depth are used as input parameters.

A major drawback of the proposed path-planning method is the increased time complexity due to the increased number of nodes and computational burden. This can be reduced through code optimization, and the time used for a priori path planning before actual navigation would not substantially affect the operation of the ship. Second, some assumptions and simplifications in this article may differ from reality. In future research, the consideration of factors such as weather can be explored to plan paths that avoid adverse weather conditions.

Author Contributions: Conceptualization, R.Z. and Q.G.; methodology, Q.G.; software, Z.S.; validation, Q.G. and Z.S.; formal analysis, Y.S.; resources, R.Z.; data curation, Q.G.; writing—original draft preparation, Q.G.; writing—review and editing, R.Z.; visualization, Y.S.; project administration, R.Z.; funding acquisition, R.Z. All authors read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (grant number 52001134), the Fujian Provincial Natural Science Foundation (grant numbers 2020J01661, 2020J01660 and 2023J01326), and the Fuzhou–Xiamen–Quanzhou Independent Innovation Region Cooperated Special Foundation (grant number 3502ZCQXT2021007).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created in this study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Silveira, P.; Teixeira, A.; Figueira, J.R.; Soares, C.G. A multicriteria outranking approach for ship collision risk assessment. *Reliab. Eng. Syst. Saf.* **2021**, *214*, 107789. [[CrossRef](#)]
2. Liu, J.; Yan, X.; Liu, C.; Fan, A.; Ma, F. Developments and Applications of Green and Intelligent Inland Vessels in China. *J. Mar. Sci. Eng.* **2023**, *11*, 318. [[CrossRef](#)]
3. Jorge, V.A.M.; Granada, R.; Maidana, R.G.; Jurak, D.A.; Heck, G.; Negreiros, A.P.F.; dos Santos, D.H.; Goncalves, L.M.G.; Amory, A.M. A Survey on Unmanned Surface Vehicles for Disaster Robotics: Main Challenges and Directions. *Sensors* **2019**, *19*, 702. [[CrossRef](#)] [[PubMed](#)]
4. Wang, L.; Li, S.; Liu, J.; Hu, Y.; Wu, Q. Design and implementation of a testing platform for ship control: A case study on the optimal switching controller for ship motion. *Adv. Eng. Softw.* **2023**, *178*, 103427. [[CrossRef](#)]
5. Felski, A.; Zwolak, K. The ocean-going autonomous ship—Challenges and threats. *J. Mar. Sci. Eng.* **2020**, *8*, 41. [[CrossRef](#)]
6. Zhen, R.; Shi, Z.; Liu, J.; Shao, Z. A novel arena-based regional collision risk assessment method of multi-ship encounter situation in complex waters. *Ocean Eng.* **2022**, *246*, 110531. [[CrossRef](#)]
7. Wang, N.; Sun, Z.; Yin, J.; Zou, Z.; Su, S.-F. Fuzzy unknown observer-based robust adaptive path following control of underactuated surface vehicles subject to multiple unknowns. *Ocean Eng.* **2019**, *176*, 57–64. [[CrossRef](#)]
8. Zhen, R.; Shi, Z.; Shao, Z.; Liu, J. A novel regional collision risk assessment method considering aggregation density under multi-ship encounter situations. *J. Navig.* **2022**, *75*, 76–94. [[CrossRef](#)]
9. Zhou, C.; Gu, S.; Wen, Y.; Du, Z.; Xiao, C.; Huang, L.; Zhu, M. The review unmanned surface vehicle path planning: Based on multi-modality constraint. *Ocean Eng.* **2020**, *200*, 107043. [[CrossRef](#)]
10. Li, S.; Xu, C.; Liu, J.; Han, B. Data-driven docking control of autonomous double-ended ferries based on iterative learning model predictive control. *Ocean Eng.* **2023**, *273*, 113994. [[CrossRef](#)]
11. Liu, J.; Wang, J. Overview of obstacle avoidance path planning algorithms for unmanned surface vehicle. *Comput. Appl. Softw.* **2020**, *37*, 1–10+20.
12. Cheng, T.; Utne, I.B.; Wu, B.; Wu, Q. A novel system-theoretic approach for human-system collaboration safety: Case studies on two degrees of autonomy for autonomous ships. *Reliab. Eng. Syst. Saf.* **2023**, *237*, 109388. [[CrossRef](#)]
13. Chen, X.; Liu, S.; Liu, R.W.; Wu, H.; Han, B.; Zhao, J. Quantifying Arctic oil spilling event risk by integrating an analytic network process and a fuzzy comprehensive evaluation model. *Ocean Coast. Manag.* **2022**, *228*, 106326. [[CrossRef](#)]
14. Liu, Z.; Zhang, Y.; Yu, X.; Yuan, C. Unmanned surface vehicles: An overview of developments and challenges. *Annu. Rev. Control* **2016**, *41*, 71–93. [[CrossRef](#)]
15. Chen, X.; Wu, S.; Shi, C.; Huang, Y.; Yang, Y.; Ke, R.; Zhao, J. Sensing data supported traffic flow prediction via denoising schemes and ANN: A comparison. *IEEE Sens. J.* **2020**, *20*, 14317–14328. [[CrossRef](#)]
16. Liu, Y.C.; Bucknall, R. Path planning algorithm for unmanned surface vehicle formations in a practical maritime environment. *Ocean Eng.* **2015**, *97*, 126–144. [[CrossRef](#)]
17. Singh, Y.; Sharma, S.; Sutton, R.; Hatton, D.; Khan, A. Feasibility study of a constrained Dijkstra approach for optimal path planning of an unmanned surface vehicle in a dynamic maritime environment. In Proceedings of the 2018 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), Torres Vedras, Portugal, 25–27 April 2018; pp. 117–122.
18. Liang, C.L.; Zhang, X.K.; Watanabe, Y.; Deng, Y.J. Autonomous Collision Avoidance of Unmanned Surface Vehicles Based on Improved a Star and Minimum Course Alteration Algorithms. *Appl. Ocean Res.* **2021**, *113*, 102755. [[CrossRef](#)]
19. Gu, Q.; Zhen, R.; Liu, J.; Li, C. An improved RRT algorithm based on prior AIS information and DP compression for ship path planning. *Ocean Eng.* **2023**, *279*, 114595. [[CrossRef](#)]
20. Chi, W.; Ding, Z.; Wang, J.; Chen, G.; Sun, L. A Generalized Voronoi Diagram-Based Efficient Heuristic Path Planning Method for RRTs in Mobile Robots. *IEEE Trans. Ind. Electron.* **2021**, *69*, 4926–4937. [[CrossRef](#)]
21. Dong, Y.; Camci, E.; Kayacan, E. Faster RRT-based nonholonomic path planning in 2D building environments using skeleton-constrained path biasing. *J. Intell. Robot. Syst.* **2018**, *89*, 387–401. [[CrossRef](#)]
22. Cao, S.; Fan, P.; Yan, T.; Xie, C.; Deng, J.; Xu, F.; Shu, Y. Inland waterway ship path planning based on improved RRT algorithm. *J. Mar. Sci. Eng.* **2022**, *10*, 1460. [[CrossRef](#)]
23. Öztürk, Ü.; Akdağ, M.; Ayabakan, T. A review of path planning algorithms in maritime autonomous surface ships: Navigation safety perspective. *Ocean Eng.* **2022**, *251*, 111010. [[CrossRef](#)]
24. Lin, H.X.; Xiang, D.; Ou, Y.J.; Lan, X.D. Review of Path Planning Algorithms for Mobile Robots. *Comput. Eng. Appl.* **2021**, *57*, 38–48.
25. Long, Y.; Su, Y.; Zhang, H.; Li, M. Application of improved genetic algorithm to unmanned surface vehicle path planning. In Proceedings of the 2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS), Enshi, China, 25–27 May 2018; pp. 209–212.

26. Zhang, J.; Chen, X.; Hu, H.; Zhang, J.; Wang, L. Global path planning algorithm based on the IPSO algorithm for USVs. In Proceedings of the 2018 Chinese Control and Decision Conference (CCDC), Shenyang, China, 9–11 June 2018; pp. 4901–4906.
27. Woo, J.; Kim, N. Collision avoidance for an unmanned surface vehicle using deep reinforcement learning. *Ocean Eng.* **2020**, *199*, 107001. [[CrossRef](#)]
28. Li, L.; Wu, D.; Huang, Y.; Yuan, Z.-M. A path planning strategy unified with a COLREGS collision avoidance function based on deep reinforcement learning and artificial potential field. *Appl. Ocean Res.* **2021**, *113*, 102759. [[CrossRef](#)]
29. Liu, Z.; Zhou, Z.Z.; Zhang, M.Y.; Liu, J.X. A Twin Delayed Deep Deterministic Policy Gradient Method for Collision Avoidance of Autonomous Ships. *J. Transp. Inf. Saf.* **2022**, *40*, 60–74.
30. Ayawli, B.B.K.; Chellali, R.; Appiah, A.Y.; Kyeremeh, F. An Overview of Nature-Inspired, Conventional, and Hybrid Methods of Autonomous Vehicle Path Planning. *J. Adv. Transp.* **2018**, 8269698. [[CrossRef](#)]
31. Guo, B.; Kuang, Z.; Guan, J.; Hu, M.; Rao, L.; Sun, X. An Improved A-Star Algorithm for Complete Coverage Path Planning of Unmanned Ships. *Int. J. Pattern Recognit. Artif. Intell.* **2022**, *36*, 2259009. [[CrossRef](#)]
32. Duchon, F.; Babinec, A.; Kajan, M.; Beno, P.; Florek, M.; Fico, T.; Jurisica, L. Path planning with modified A star algorithm for a mobile robot. In Proceedings of the 6th Conference on Modelling of Mechanical and Mechatronic Systems (MMaMS), Vysoke Tatry, Slovakia, 25–27 November 2014; pp. 59–69.
33. Chen, D.T.; Liu, X.D.; Liu, S.W. Improved A* algorithm based on two-way search for path planning of automated guided vehicle. *J. Comput. Appl.* **2021**, *41*, 309–313.
34. Fernandes, E.; Costa, P.; Lima, J.; Veiga, G. Towards an Orientation Enhanced A star Algorithm for Robotic Navigation. In Proceedings of the IEEE International Conference on Industrial Technology (ICIT), Seville, Spain, 17–19 March 2015; pp. 3320–3325.
35. Zhang, J.G.; Zhang, F.; Chen, L.G.; Jiang, Q.; Zhu, W. Path planning of automatic guided vehicle based on improved A* algorithm. *Foreign Electron. Meas. Technol.* **2022**, *41*, 123–128. [[CrossRef](#)]
36. Zhang, Y.; Li, L.-l.; Lin, H.-C.; Ma, Z.; Zhao, J. Development of Path Planning Approach Based on Improved A-Star Algorithm in AGV System. In Proceedings of the 3rd European-Alliance-for-Innovation (EAI) International Conference on IoT as a Service (IoTaaS), Taichung, Taiwan, 20–22 September 2017; pp. 276–279.
37. Nayl, T.; Mohammed, M.Q.; Muhamed, S.Q. Obstacles Avoidance for an Articulated Robot Using Modified Smooth Path Planning. In Proceedings of the International Conference on Computer and Applications (ICCA), Doha, United Arab Emirates, 6–7 September 2017; pp. 185–189.
38. Lu, W.L.; Lei, J.S.; Shao, Y. Path Planning for Mobile Robot Based on an Improved A* Algorithm. *J. Ordnance Equip. Eng.* **2019**, *40*, 197–201.
39. Gunawan, S.A.; Pratama, G.N.; Cahyadi, A.I.; Winduratna, B.; Yuwono, Y.C.; Wahyunggoro, O. Smoothed A-star algorithm for nonholonomic mobile robot path planning. In Proceedings of the 2019 International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, 24–25 July 2019; pp. 654–658.
40. Sun, J.G.; Sun, T. Research on UAV path planning based on fusion A* algorithm. *Electron. Meas. Technol.* **2022**, *45*, 82–91. [[CrossRef](#)]
41. Shu, W.N.; Zhao, J.S.; Xie, Z.X.; Zhang, X.S.; Ma, X. Path planning for unmanned surface vessels based on improved A* algorithm. *J. Shanghai Marit. Univ.* **2022**, *43*, 1–6. [[CrossRef](#)]
42. Liu, C.G.; Mao, Q.Z.; Chu, X.M.; Xie, S. An Improved A-Star Algorithm Considering Water Current, Traffic Separation and Berthing for Vessel Path Planning. *Appl. Sci.* **2019**, *9*, 1057. [[CrossRef](#)]
43. Liu, S. Study on the Path Planning Algorithm of Unmanned Surface Vehicles in Complex Marine Environment. Master's Thesis, Tianjin University, Tianjin, China, 2019.
44. Lee, T.; Kim, H.; Chung, H.; Bang, Y.; Myung, H. Energy efficient path planning for a marine surface vehicle considering heading angle. *Ocean Eng.* **2015**, *107*, 118–131. [[CrossRef](#)]
45. Razgallah, I.; Kaidi, S.; Smaoui, H.; Sergeant, P. The impact of free surface modelling on hydrodynamic forces for ship navigating in inland waterways: Water depth, drift angle, and ship speed effect. *J. Mar. Sci. Technol.* **2019**, *24*, 620–641. [[CrossRef](#)]
46. Zhang, X.; Zhang, Q.; Yang, J.; Cong, Z.; Luo, J.; Chen, H. Safety Risk Analysis of Unmanned Ships in Inland Rivers Based on a Fuzzy Bayesian Network. *J. Adv. Transp.* **2019**, *2019*, 4057195. [[CrossRef](#)]
47. Andersen, P. Ship motions and sea loads in restricted water depth. *Ocean Eng.* **1979**, *6*, 557–569. [[CrossRef](#)]
48. Bai, X.E.; Jiang, M.Z.; Xu, X.F.; Sun, D.Y. Ship route planning based on improved ant colony algorithm of two-way search. *China Navig.* **2022**, *45*, 13–20.
49. Wang, Y.; Liang, X.; Li, B.; Yu, X. Research and implementation of global path planning for unmanned surface vehicle based on electronic chart. In *Recent Developments in Mechatronics and Intelligent Robotics: Proceedings of the International Conference on Mechatronics and Intelligent Robotics (ICMIR2017)*; Springer: Berlin/Heidelberg, Germany, 2018; Volume 1, pp. 534–539.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.