Journal of
Marine Science and Engineering

MDPI

Article

Research on Autonomous Underwater Vehicle Path Optimization Using a Field Theory-Guided A* Algorithm

Zhiyuan Xu [1], Yong Shen [2], Zhexue Xie [2] and Yihua Liu [1,*]

[1] Merchant Marine College, Shanghai Maritime University, Shanghai 201306, China; 19963846305@163.com
[2] Ningbo Dagang Pilotage Co., Ltd., Ningbo 315812, China; sheny@nbport.com.cn (Y.S.); xiezx@nbport.com.cn (Z.X.)
* Correspondence: liuyh@shmtu.edu.cn

Abstract: Autonomous Underwater Vehicles (AUVs) have become indispensable tools in the fields of ocean exploration, resource exploitation, and environmental monitoring. Path planning and obstacle avoidance are crucial to improve the operational capabilities of AUVs. However, most algorithms focus only on macro-global or micro-local path planning and rarely address both problems simultaneously. This study extends the classical A* algorithm by integrating field theory principles. The resulting Field Theory Augmented A* (FT-A*) algorithm combines the constraints in the AUV's dynamics and the threats posed by obstacles to ensure a safe navigation distance. The paths planned by the FT-A* algorithm were subsequently re-optimised in conjunction with Dubins curves, taking into account path smoothness and redundant node problems. Simulation experiments confirm that the improved algorithm can effectively help AUVs navigate safely around obstacles, which greatly improves navigation safety and increases the arithmetic power and navigation efficiency. The proposed FT-A* algorithm provides a robust solution for underwater path planning and demonstrates great practical value for AUV operation in complex marine environments.

Keywords: Autonomous Underwater Vehicle; A* algorithm; FT-A* algorithm; path planning; field theory

check for updates

Citation: Xu, Z.; Shen, Y.; Xie, Z.; Liu, Y. Research on Autonomous Underwater Vehicle Path Optimization Using a Field Theory-Guided A* Algorithm. *J. Mar. Sci. Eng.* **2024**, *12*, 1815. https://doi.org/10.3390/jmse12101815

Academic Editor: Rafael Morales

Received: 9 September 2024
Revised: 4 October 2024
Accepted: 9 October 2024
Published: 11 October 2024

1. Introduction

Path planning for Autonomous Underwater Vehicles (AUVs) has been a hot topic of interest and challenge. This is attributed to its great potential in areas such as scientific research and underwater operations. In recent years, due to the advancement of science and technology as well as the growing demand for marine exploration, underwater submarines have been playing an increasing role in various marine applications, such as mine clearance, pipeline maintenance, and marine environment monitoring [1–3]. Path planning is essential for AUV navigation. The underwater collision avoidance problem is fundamentally a path planning challenge, requiring AUVs to find safe routes from start to finish. Given the complexity and uncertainty of the underwater environment, AUVs must be sufficiently intelligent to navigate autonomously in such conditions [4].

In recent years, various solution methods have been developed and applied to the collision-free path planning problem for underwater robots, and these methods can be broadly classified into two main categories, i.e., local path planning algorithms and global path planning algorithms [5].

Micro-local planning algorithms such as rapidly exploring random trees (RRTs), fuzzy logic algorithms, and algorithms for neural networks (ANNs) are widely used for AUV navigation. The RRT algorithm searches for paths in a high-dimensional state space and performs superiorly for non-convex obstacles and complex environments. Its stochastic and stepwise growth characteristics make it suitable for most continuous state space planning problems. However, the paths generated by RRTs are usually not optimal solutions, with poor real-time performance and stability [6–8]. The fuzzy logic algorithm takes into account the uncertainty of the environment, such as sensor noise or uncertainty of obstacle locations due to the dynamics of the environment. The algorithm enhances the ability of the system to adapt to the environment; however, the algorithm suffers from drawbacks such as insufficiently accurate mathematical models and lack of systematicity [9,10]. Neural network algorithms mainly learn the features of the environment and optimal paths by training neural networks to achieve intelligent path planning [11]. Common algorithms are deep reinforcement learning (DRL) [12] and convolutional neural networks (CNN). However, applying these algorithms in path planning usually requires a large amount of training data and the design of complex neural network structures to achieve good results [13].

Macro-global path planning algorithms include the genetic algorithm (GA), particle swarm optimization (PSO), ant colony optimization (ACO), and A* algorithms. Genetic algorithms have a powerful global search capability, but convergence is slower and less stable [14]. The particle swarm algorithm has the advantage of a fast search time, but is prone to fall into local optimum [15]. The ant colony algorithm has the advantage of a high search capability and is highly efficient, but is characterised by slow pre-convergence when used for path planning [16]. The traditional A* algorithm, as a heuristic algorithm, has the characteristics of simplicity, high efficiency, and an optimal search path, which is the research hotspot of path planning today [17–19]. Considering the plasticity of the traditional A* algorithm itself, several scholars have improved it. In order to shorten the search time, Szczerba and Cai et al. [20,21] proposed a sparse A* search algorithm, which uses the maximum steering angle and the maximum path length as the constraints of the basic search algorithm. Subsequently, Chen et al. [22] constructed an obstacle search space by randomly distributing points based on the sparse A* search algorithm, combined with a visibility detection method under the constraint of the maximum turning radius to make AUV paths smoother. Li and Zhang [23] designed a multi-directional search A* algorithm, which reduces the number of search nodes in order to obtain the optimal path. Traditional A* algorithms often simplify the object of study to a point mass, ignoring the key feature that AUVs have a physical volume. This simplification fails to take into account the complexity of the dynamic and variable underwater environment and ignores the fact that AUVs have six degrees of freedom of motion underwater. Therefore, it lacks practical significance to simply consider AUVs as particles and apply them directly to underwater path planning problems. In contrast, this study focuses on refining the traditional A* algorithm by incorporating the concept of field theory. The conventional A* algorithm excels in macroscopic global path planning, efficiently identifying the optimal route from the starting point to the destination. Field theory, on the other hand, offers robust support at the microscopic level, enabling the newly planned path to effectively avoid collisions within local areas, thereby ensuring both the safety and efficiency of navigation. To summarise, the constraints that must be taken into account when AUVs perform the path planning problem include the following two points.

- Limitations of kinematic characteristics

Due to the complex underwater environment and the limitation of an AUV's own kinematic characteristics, an AUV does not have the flexible steering ability and acceleration of cars on the road. Considering the kinematic characteristics of AUVs in path planning will greatly improve safety during navigation.

- Threat cost of obstacles

Considering the shape, position, and other information of the obstacles, as well as the complexity of the seabed environment, a suitable safety distance threshold is determined, so as to ensure that the planned path has stronger practical significance.

In light of these constraints, this paper proposes an innovative approach by integrating field theory with the traditional A* algorithm to address the specific challenges in AUV path planning. The main contributions of this work can be summarised as follows:

- A hybrid optimisation algorithm (FT-A*) is formed by combining field theory and the A* algorithm to compensate for the lack of local obstacle avoidance ability of the traditional A* algorithm in microscopic conditions.
- Redefining G and H as weighted combinations of safety potential energy values and Euclidean distances not only takes into account the kinematic properties of the AUV itself, but also incorporates the threat cost of obstacles into path planning.
- The redundant node removal algorithm is proposed to perform redundant node removal to improve the arithmetic power and search efficiency while ensuring the security of the planned path.

The rest of this paper is organised as follows: Section 2 introduces the kinematics and dynamics model of the AUV. Section 3 introduces the concept of a potential energy field, proposes the FT-A* algorithm on the basis of the A* algorithm, and further improves the FT-A* algorithm for the problems of redundant nodes and path smoothness. In Section 4, simulation experiments are conducted to analyse and compare the operation results of various algorithms, which prove that the improved FT-A* algorithm has considerable robustness. Finally, the whole paper is summarised.

2. Mathematical Modelling of an AUV

2.1. Coordinate Framework and Motion Parameter Specification

The dynamics study for an AUV is usually described in two coordinate systems, i.e., an Earth-fixed coordinate system and body coordinate system, as shown in Figure 1. $O_E - x_E y_E z_E$ is the Earth-fixed coordinate system, the origin of which is a point on the horizontal plane, and it has three axes in three directions, namely x_E, y_E, z_E, which is also known as the static coordinate system. $O_B - x_B y_B z_B$ is the body coordinate system, the origin of which is usually chosen in the center of gravity of the AUV, and it has three axes in three directions, respectively x_B, y_B, z_B. This coordinate system is often referred to as the moving coordinate system because it changes with the moving state of the AUV. The motion parameters and symbols used to describe these dynamics are listed in Table 1.

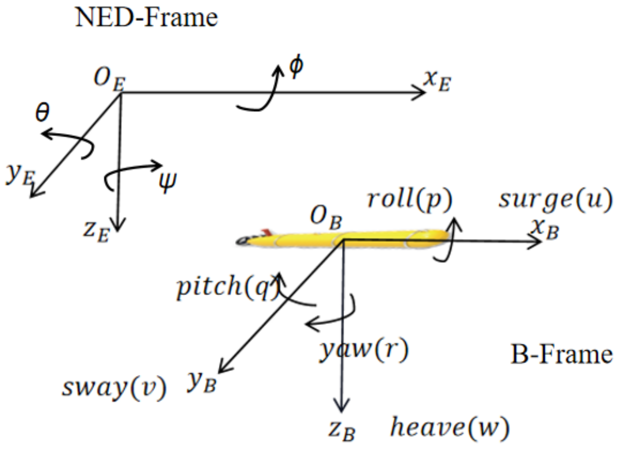


Figure 1. Earth-fixed and body coordinate systems.

Table 1. Motion parameters and symbols for AUV.

	Position and Angle	**L-Vel and A-Vel ***	**Force and Torque**
coordinate system	NED-Frame	B-Frame	B-Frame
Surge	x_E	u	X
Sway	y_E	v	Y
Heave	z_E	w	Z
Roll	ϕ	p	K
Pitch	θ	q	M
Yaw	ψ	r	N

* L-vel and A-vel are abbreviations for linear velocity and angular velocity.

The definitions of displacement vector η, velocity vector v, and thrust vector τ of the AUV are shown in Equations (1)–(3). The external force acting on the AUV can be represented by the external force vector $\tau_1 = [X, Y, Z]^T$, where X, Y, Z are the forces received by the AUV in the directions $O_B - x_B$, $O_B - y_B O_B - z_B$, and $O_B - z_B$, respectively. And the external moments to which the AUV is subjected can be expressed by the external moment vector $\tau_2 = [K, M, N]^T$, where K, M, N represent the roll, pitch, and yaw moments to which the AUV is subjected, respectively.

$$\eta = [x, y, z, \phi, \theta, \psi]^T \tag{1}$$

$$v = [u, v, w, p, q, r]^T \tag{2}$$

$$\tau = [X, Y, Z, K, M, N]^T \tag{3}$$

2.2. The Kinetic Modelling of AUVs

By considering the various forces and moments of the AUV, comprehensive equations covering attitude and position variations can be derived to analyse in depth the complex kinematic properties of the AUV. The general description of the six-degree-of-freedom dynamical equations for the motion of the AUV is provided as follows [24]:

$$\begin{cases} \dot{\eta} = J(\eta)v \\ M\dot{v} + C(v)v + D(v)v + g(\eta) + \tau_w = \tau \end{cases} \tag{4}$$

where $\eta = [x, y, z, \phi, \theta, \psi]^T$ is the displacement vector; $\boldsymbol{v} = [u, v, w, p, q, r]^T$ is the velocity vector; x, y, z are positions along surge, sway, and heave directions; ϕ, θ, ψ denote the Euler angles of the underwater vehicle in the Earth coordinate system; u, v, w denote linear velocities along surge, sway, and heave directions; p, q, r denote rotational velocities in roll, pitch, and yaw motions; $J(\eta)$ denotes the Jacobi transformation matrix; M denotes the system inertia matrix; $C(v)$ is the Coriolis-centripetal matrix; $D(v)$ is the hydrodynamic damping matrix; $g(\eta)$ denotes the restoring force vector from gravity and buoyancy; and τ denotes the propulsive and rotational torques.

By selecting the spindle, the system inertia matrix is defined as:

$$\boldsymbol{M} = \begin{bmatrix} m & 0 & 0 & 0 \\ 0 & m & 0 & 0 \\ 0 & 0 & m & 0 \\ 0 & 0 & 0 & I_z \end{bmatrix} \tag{5}$$

The Coriolis-centripetal matrix is defined as:

$$\boldsymbol{C} = \begin{bmatrix} 0 & 0 & 0 & -mv \\ 0 & 0 & 0 & mu \\ 0 & 0 & 0 & 0 \\ mv & -mu & 0 & 0 \end{bmatrix} \tag{6}$$

The hydrodynamic damping matrix is defined as:

$$\boldsymbol{D}(\boldsymbol{\nu}) = \begin{bmatrix} X_u + X_{u|u|}|u| & 0 & 0 & 0 \\ 0 & Y_v + Y_{\nu|y|v|}|v| & 0 & 0 \\ 0 & 0 & Z_w + Z_{w|w|w|} & 0 \\ 0 & 0 & 0 & N_r + N_{r|r|r|}|r| \end{bmatrix} \tag{7}$$

$\tau = [T_u, T_v, T_w, 0, 0, T_r]^T$, where T_u, T_v, T_w T_r are the variables of the AUV in the direction of longitudinal surge, longitudinal sway, heave, and yaw, respectively. The kinematic equations of the AUV are shown in Equation (8):

$$\begin{cases} \dot{x} = ucos\psi - vsin\psi \\ \dot{y} = usin\psi + vcos\psi \\ \dot{z} = w \\ \dot{\psi} = r \\ m\dot{u} - mvr + X_u u + X_{|u|u}|u|u = T_u \\ m\dot{v} + mur + Y_v v + Y_{|v|v}|v|v = T_v \\ m\dot{w} + Z_w w + Z_{|w|w}|w|w = T_w \\ I_z\dot{r} + N_r r + N_{|r|r}|r|r = T_r \end{cases} \tag{8}$$

In order to explore the collision avoidance problem of AUVs in underwater navigation, a PX-210 AUV with a weight of 70 kg is used as an example, and its navigation speed ranges from 0 to 4 knots. Without considering special cases, we make the following assumptions: (1) the centre of mass (CM) coincides with the centre of gravity (CG) and the centre of buoyancy (CB); and (2) since the AUV has a small amount of inertia in the water and its maximum travel speed is only 4 knots, the effects of small-angle rolls and pitches (e.g., 1 to 5 degrees) on its travel path are negligible. Based on the fact that the motions of the AUV in the direction of roll and pitch have a minimal effect on its overall dynamics ($\phi = \theta = 0; p = q = 0$), these motions can be considered negligible factors in practice [25].

2.3. Kinematic Equations for AUVs

The kinematic model characterizes the AUV's motion by describing its position, velocity, and acceleration, while excluding the forces or moments responsible for generating these movements.

Since the displacement vector η and the velocity vector v are represented in two different coordinate systems, a coordinate transformation is required to relate the parameter information in different coordinate systems.

In particular, the transformation relationship between the position parameter information under the B-Frame and the position parameter information under the NED-Frame is as follows:

$$\begin{bmatrix} x_E \\ y_E \\ z_E \end{bmatrix} = R \begin{bmatrix} x_B \\ y_B \\ z_B \end{bmatrix} \tag{9}$$

where R is a passive linear velocity transformation matrix used to convert the velocity vector of the AUV in the B-Frame to the velocity vector in the NED-Frame. This conversion is based on the attitude angles of the AUV (i.e., roll, pitch, and yaw) and is realized by matrix multiplication to transform the velocity in the coordinate system.

$$R = \begin{bmatrix} \cos\phi\cos\theta & \cos\phi\sin\theta\sin\varphi - \sin\phi\cos\varphi & \cos\phi\sin\theta\cos\varphi + \sin\phi\sin\varphi \\ \sin\phi\cos\theta & \sin\phi\sin\theta\sin\varphi + \cos\phi\cos\varphi & \sin\phi\sin\theta\cos\varphi - \cos\phi\sin\varphi \\ -\sin\theta & \cos\theta\sin\varphi & \cos\theta\cos\varphi \end{bmatrix} \tag{10}$$

The velocity transformation relationship between the two coordinate systems adheres to the aforementioned transformation rules. However, it is important to note that this transformation is only applicable to the velocity at the centre of mass (CM). If the velocity at other points on the AUV is considered, additional factors from rigid body dynamics, such as angular velocity, must be taken into account. The transformation relationship between the velocity parameter information under the B-Frame and the NED-Frame is as follows:

$$\begin{bmatrix} \dot{x_E} \\ \dot{y_E} \\ \dot{z_E} \end{bmatrix} = R \begin{bmatrix} \dot{x_B} \\ \dot{y_B} \\ \dot{z_B} \end{bmatrix} = R \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{11}$$

The conversion relationship between the angular velocity under the B-Frame and the angular velocity under the NED-Frame is as follows:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = T \begin{bmatrix} \dot{x_B} \\ \dot{y_B} \\ \dot{z_B} \end{bmatrix} \tag{12}$$

where T is the angular velocity conversion matrix.

This is obtained by combining Equations (9)–(12):

$$\begin{bmatrix} \dot{x_E} \\ \dot{y_E} \\ \dot{z_E} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} R & 0_{3\times3} \\ 0_{3\times3} & T \end{bmatrix} \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \end{bmatrix} \tag{13}$$

3. Methodology

3.1. Traditional A* Path Planning Algorithm

The A* algorithm is a well-established heuristic search algorithm for solving the path planning problem for ground-based AUVs. It continuously evaluates the cost of neighbouring nodes and obtains the best neighbouring nodes until the goal is reached [18]. The algorithm uses heuristics to search for the goal location, where n denotes the current node; $F(n)$ is the total distance cost from the starting point to the end point; G(n) is the actual distance cost from the starting point to the current node; and H(n) is the estimated distance cost from the current node to the goal point (usually expressed in terms of Euclidean distance). Overall, the A* algorithm is used to minimise the total distance cost $F(n)$:

$$F(n) = G(n) + H(n) \tag{14}$$

The A* algorithm is a widely used pathfinding and graph traversal algorithm that combines the best features of Dijkstra's algorithm and greedy best-first-search algorithm. It is designed to find the shortest path from a starting node to a target node in a weighted graph, while also being efficient by using heuristics to guide the search.

The A* algorithm consists of two main data structures: the OPEN list and the CLOSE list. The OPEN list stores nodes that have been generated but not yet examined, while the CLOSE list stores nodes that have already been visited and examined. The algorithm iteratively expands nodes, evaluates their neighbours, and updates the lists until the target node is found.

1. Put the starting point as the 1st node into the CLOSE list and calculate the evaluation function of up to 26 surrounding nodes.
2. Put these nodes and node information into the OPEN list. If the current node already exists in the OPEN list, compare the evaluation function value of the node and update the node information of the node with the smaller evaluation function value in the OPEN list.
3. In the OPEN list, select the node with the smallest evaluation function value as the child node of the current node.
4. Take the selected child node as the parent node and, at the same time, put its node information into the CLOSE list, use the new parent node and then child node expansion and calculate the evaluation function value of each child node.
5. Compare the evaluation function value to the new child node and keep updating the OPEN and CLOSE list until the target node is searched.

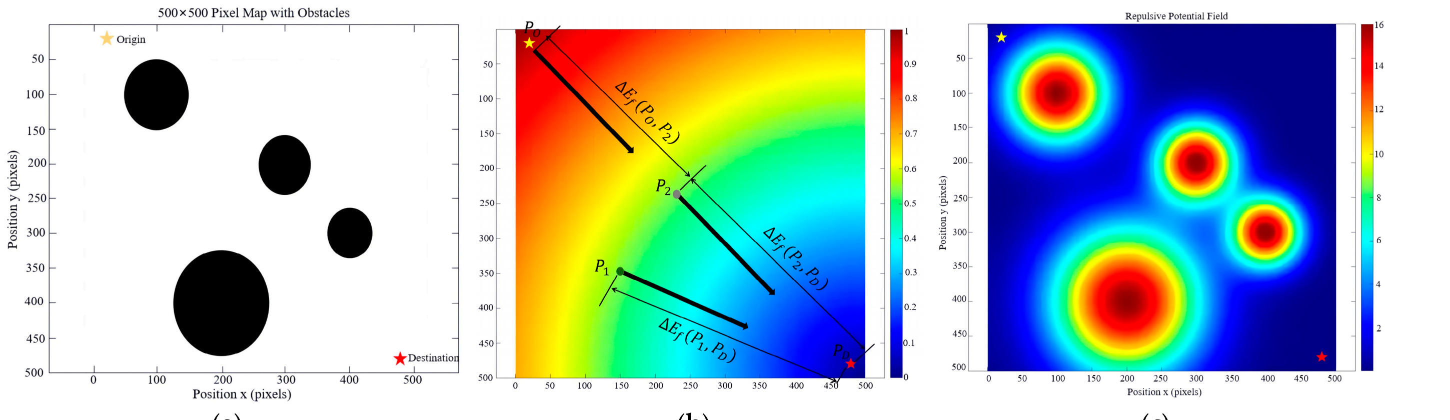
The A* algorithm, as a classical global path planning algorithm, has been widely used in the field of autonomous driving. However, although the paths planned according to this algorithm have the shortest distances, the limitations of the algorithm itself lead to more constraints when applying it directly to the path planning problem of AUVs.

- Due to the complex underwater environment and the limitation of AUV's own kinematic characteristics, AUV cannot steer and accelerate as flexibly as land vehicles. Therefore, fully considering the kinematic characteristics of AUVs in path planning will significantly improve the safety of planned paths.
- The shape and position of obstacles and the complexity of the undersea environment are considered comprehensively to determine the appropriate safety distance threshold, so as to ensure that the planned path has stronger practical significance.
- Although the A* algorithm follows the shortest path principle, it still needs to improve the smoothness of the path and solve the problem of too many steering points.

3.2. Introduction and Analysis of the FT-A Algorithm*

Field is initially a basic concept in physics, and with the development of the idea of field theory, it has been abstracted into a mathematical concept for describing the distribution law of a certain physical quantity or mathematical function in space, i.e., there is a correspondence between any point in a region of space and a definite physical quantity [26–28]. In two-dimensional traffic, a microscopic traffic flow model based on the potential energy field theory can well characterise the driving risk of a vehicle (ship) in motion [29]. Applying this theory to the path planning problem of an underwater vehicle can be a good solution to the kinematic characterisation constraints inherent in the movement of an AUV as well as the threat cost constraints of obstacles.

In this paper, field theory is applied to analyse the entire process of AUV navigation underwater. This approach helps in understanding both the macroscopic and microscopic aspects of AUV's movement and interaction with the environment. From the macroscopic point of view, the AUV always sails towards the destination, so there is a traction potential field between the destination and the AUV, which constantly pulls it to the destination to avoid aimless sailing underwater. From the microscopic point of view, the AUV colliding with obstacles will bring about a great loss, so it is necessary to avoid obstacles during the sailing process, thus there is a safety potential field between the AUV and obstacles, which guides the AUV to travel in the safety area and finally reach the destination. The schematic is shown in (Figure 2), where (a) is the environment constructed, (b) is the traction potential field, and (c) is the safety potential field.



(**a**) (**b**) (**c**)

Figure 2. Schematic diagram of potential energy field construction. (**a**) Schematic diagram of environment modeling and obstacle distribution; (**b**) Schematic of the traction potential field; (**c**) Schematic diagram of the safety potential field.

3.2.1. The Gravitational Field

The destination exerts a gravitational force on the AUV, guiding the AUV towards its constant motion (similar to the heuristic function h in the A* algorithm). The AUV will always be affected by the traction potential energy field until it reaches the destination. Let the destination be the zero potential energy point of the traction potential energy field, $E_f(destination) = 0$, and the starting point has a traction potential energy value of 1, $E_f(origin) = 1$. Points at different locations have varying values of traction potential energy, and in the case where the AUV does not arrive at the destination, the difference in the traction potential energy that exists between the point at the location of the AUV and the destination triggers the AUV's motion, which moves from the high potential energy point to the low potential energy point, as shown in (Figure 3). Once the AUV reaches the destination, it is no longer subjected to traction and thus stops moving. The position of the starting point is $p_o = [x_o, y_o, z_o]$ and the position of the destination point is $p_d = [x_d, y_d, z_d]$.

Where $\Delta E_f(p_1, p_d)$ denotes the potential energy difference between point p_1 and the end point, $\Delta E_f(p_2, p_d)$ denotes the potential energy difference between point p_2 and the end point, and $\Delta E_f(p_o, p_2)$ denotes the potential energy difference between the starting point and point p_2. Within the framework of the traction potential energy field, $\Delta E_f(p_1, p_d)$ denotes the potential energy difference between point p_1 and the destination point p_d. Similarly, $\Delta E_f(p_2, p_d)$ represents the potential energy difference between point p_2 and the destination. Additionally, $\Delta E_f(p_o, p_2)$ captures the potential energy variation between the starting point p_o and point p_2. These potential energy differences, as defined by the traction potential energy field, are crucial for evaluating the overall path cost by reflecting the energetic interactions between key points along the planned trajectory.

The traction potential energy function at any p-point underwater is as follows:

$$E_f(x_p, y_p, z_p) = \frac{\sqrt{(x_p - x_\mathrm{d})^2 + (y_p - y_\mathrm{d})^2 + (z_p - z_\mathrm{d})^2}}{\sqrt{(x_\mathrm{o} - x_\mathrm{d})^2 + (y_\mathrm{o} - y_\mathrm{d})^2 + (z_o - z_\mathrm{d})^2}} \tag{15}$$

The potential energy difference function at any two points p_1, p_2 underwater is as follows:

$$\Delta E_f(p_1, p_2) = E_f(x_{P1}, y_{P1}, z_{P1}) - E_f(x_{P2}, y_{P2}, z_{P2}) \tag{16}$$

The AUV always moves from the direction of the high potential energy gradient to the direction of the geopotential energy gradient in the direction of the potential energy difference:

$$\underset{\Delta E_f(p_1,p_2)}{\longrightarrow} = \frac{(x_{i+1} - x_i, y_{i+1} - y_i, z_{i+1} - z_i)}{\sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2}} \tag{17}$$

In order to visualise the value of the traction potential energy at each point and the potential energy relationship, the potential energy formula is analysed as shown in Table 2.

Table 2. Analytical table of potential energy formulas.

Point	Value of the Traction Potential Energy	Potential Difference
Origin (p_o)	$E_f(origin) = 1$	
p_1	$E_f(x_{p_1}, y_{p_1}, z_{p_1}) = \frac{\sqrt{(x_{p_1}-x_\mathrm{d})^2+(y_{p_1}-y_\mathrm{d})^2+(z_{p_1}-z_\mathrm{d})^2}}{\sqrt{(x_\mathrm{o}-x_\mathrm{d})^2+(y_\mathrm{o}-y_\mathrm{d})^2+(z_o-z_\mathrm{d})^2}}$	$\left\lvert\underset{\Delta E_f(p_1,p_d)}{\longrightarrow}\right\rvert = \left\lvert\underset{\Delta E_f(p_2,p_d)}{\longrightarrow}\right\rvert$ $\underset{\Delta E_f(p_1,p_d)}{\longrightarrow} \neq \underset{\Delta E_f(p_2,p_d)}{\longrightarrow}$
p_2	$E_f(x_{p_2}, y_{p_2}, z_{p_2}) = \frac{\sqrt{(x_{p_2}-x_\mathrm{d})^2+(y_{p_2}-y_\mathrm{d})^2+(z_{p_2}-z_\mathrm{d})^2}}{\sqrt{(x_\mathrm{o}-x_\mathrm{d})^2+(y_\mathrm{o}-y_\mathrm{d})^2+(z_o-z_\mathrm{d})^2}}$	$\left\lvert\underset{\Delta E_f(p_2,p_o)}{\longrightarrow}\right\rvert = \left\lvert\underset{\Delta E_f(p_2,p_d)}{\longrightarrow}\right\rvert$ $\underset{\Delta E_f(p_2,p_o)}{\longrightarrow} = \underset{\Delta E_f(p_2,p_d)}{\longrightarrow}$
Destination (p_d)	$E_f(destination) = 0$	

3.2.2. The Safety Potential Field

When AUV navigates underwater, if it collides with an obstacle, the loss may be incalculable, so how to prevent the collision from occurring is a problem worth studying. The traditional A* algorithm has the characteristics of simplicity, high efficiency, and optimal search path in the path finding process. However, the algorithm regards the target as a prime point, and when it is directly applied to the underwater path planning problem, it ignores the threat cost of the surrounding obstacles and its own dynamic constraints. To improve this algorithm, we apply field theory to establish the safety potential energy field of the AUV, through which the effect of obstacles on the AUV's navigation on the seabed is represented.

The size of the range of the AUV's own safety potential energy field is determined by a variety of factors. For example, under the influence of navigational environment conditions, if navigating in an area with complicated terrain and more undercurrents, the AUV needs a larger safety potential energy field to ensure the safety of navigation. Due to the limitation of its own performance and dynamics, the AUV with more powerful propulsion system and more sensitive control system can adopt a smaller safety potential energy field. In addition, different mission requirements and changes in water depth also affect the size of the potential energy field. Figure 3 shows the safety potential energy field of an AUV when navigating underwater.

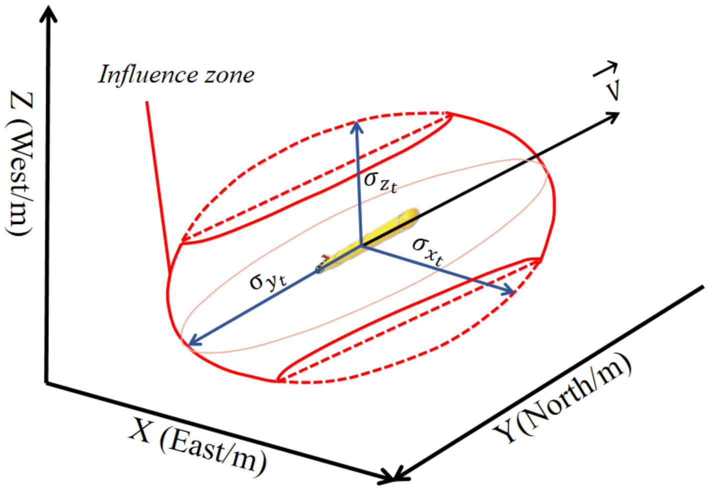


Figure 3. Schematic diagram of the safety potential energy field of an AUV.

Where $\underset{V}{\rightarrow}$ is the velocity vector of the AUV, σ_{x_t}, σ_{y_t}, σ_{y_t} denote the radius of influence of the AUV at time t, respectively, and the red ellipsoid enclosing the AUV is the safety potential energy field of the AUV itself.

The influence of different locations of obstacles on the AUV's navigation safety is directly reflected in different safety potential energy values. When the obstacles and the AUV are farther away, the influence of the safety potential energy field is smaller and the "repulsive force" is smaller. On the contrary, when they are close to each other, the influence of the safety potential energy field is large and the "repulsive force" on the AUV is stronger. The modelling of the function is expressed as follows:

$$E_0(x_t, y_t, z_t) = \frac{b}{2\pi\sigma'_{x_t}\sigma'_{y_t}\sigma'_{z_t}} e^{-\frac{(x-x_t)^2}{2\sigma'^2_{x_t}} - \frac{(y-y_t)^2}{2\sigma'^2_{y_t}} - \frac{(z-z_t)^2}{2\sigma'^2_{z_t}}} \tag{18}$$

where, $E_0(x, y, z)$ denotes the safe potential energy value of the submerged point (x, y, z), (x_t, y_t, z_t) denotes the position of the obstacle's centre at moment t, and b is a scaling factor used to adjust the potential energy field between the calculated potential energy value and the actual degree of obstacle influence on the AUV to account for the actual influence of the obstacle on the AUV's motion. The radius of influence of the potential energy field σ'_{x_t}, σ'_{y_t}, σ'_{z_t} will be dynamically adjusted according to the velocity and acceleration of the AUV in order to adapt to the actual state of motion of the AUV. We can introduce the velocity vector $v = (u, v, w)$ and the acceleration vector v̇, as well as the influence $\tau + \tau_E$ caused by propulsion and environmental factors.

The radius of influence is adjusted to the following:

$$\sigma'_{x_t},\ \sigma'_{y_t}, \sigma'_{z_t} = \sigma\big(1 + k_v||v|| + k_a\big||\dot{v}\big|| + k_\tau||\tau + \tau_E||\big) \tag{19}$$

where σ is the initial radius of influence and represents the standard sphere of influence of the potential energy field in the absence of external dynamics (e.g., velocity, acceleration, or external forces). k_v, k_a, and k_τ are adjustment coefficients used to adjust the spread of the potential energy field according to the magnitude of velocity, acceleration, and external force.

The FT-A* algorithm proposed in this paper is an improvement on the traditional A* algorithm, which enhances the path planning capability of AUVs in complex environments by introducing a safe potential energy field. In the FT-A* algorithm, instead of characterising the operating cost in the traditional A* algorithm in terms of distance, the G-value is redefined as the sum of the safety potential energy values passing through all nodes on the path. This reflects the potential energy value accumulated in the safety potential energy field for the path from the starting point to the current node. To represent the heuristic function of the G-value using an integral formula, the following equation is presented:

$$G = \int Flag\ d(close\text{-}set) \tag{20}$$

where, $Flag = E(p)$ is the potential energy value of each node in the safe potential energy field, $(close\text{-}set)$ refers to the list of nodes that have been traversed, and $d(close\text{-}set)$ denotes the cumulative summation of each node in the CLOSE list.

In traditional A* algorithms, the H-value is usually based on the Euclidean distance or the Manhattan distance. In the FT-A* algorithm, the H-value is redefined as a weighted combination of the security potential cost and distance from the current node to the target node. Not only the security potential energy value of the target node is considered, but also the spatial distance from the current location to the end point. The expression of the H-value heuristic function is shown below:

$$H = H(D_{euclid}) + H\big(E_{pot}\big) \tag{21}$$

$H\big(E_{pot}\big)$ is the weighted value of the potential energy value of this raster.

$H(D_{euclid})$ is the weighted value of the Euclidean distance from the current node to the end point; the mathematical expression is as follows:

$$D_{euclid} = \sqrt{\left(x_p - x_{end}\right)^2 + \left(y_p - y_{end}\right)^2 + \left(z_p - z_{end}\right)^2} \tag{22}$$

In summary, the value of H is as follows:

$$H = \alpha \cdot \frac{E(P) + E(end)}{2} + \beta \cdot \sqrt{\left(x_p - x_{end}\right)^2 + \left(y_p - y_{end}\right)^2 + \left(z_p - z_{end}\right)^2} \tag{23}$$

α and β are weighting parameters to adjust the relative importance of potential and distance costs in the total cost. When the distance from the target node is far or in an open and relatively safe area, the distance weight β should be increased to make it approach the target node quickly. And in waters closer to obstacles or in more complex underwater environments, α should be adjusted to meet the positional requirements.

We define ω as the ratio of the Euclidean distance from the current node to the target node to the Euclidean distance from the starting point to the target node, using the mathematical expression below:

$$\omega = \frac{D_{euclid}\left(x_p, y_p, z_p, x_{end}, y_{end}, z_{end,}\right)}{D_{Max}} \tag{24}$$

$$\alpha = \begin{cases} \alpha_0, & \omega \le \omega_l \\ \frac{\omega - 2\omega_l + \omega_i}{2(\omega_i - \omega_l)} \times (\alpha_i - \alpha_0) + \alpha_0, & \omega_l < \omega \le \omega_i \\ 0.5\alpha_0, & \omega > \omega_i \end{cases} \tag{25}$$

$$\beta = \begin{cases} \beta_l, & \omega \ge \omega_s \\ \frac{\omega_u - \omega}{\omega_u - \omega_s} \times (\beta_u - \beta_0) + \beta_0, & \omega_s \le \omega < \omega_u \\ \beta_0, & \omega < \omega_u \end{cases} \tag{26}$$

α_0, β_0 are the potential energy weights when the underwater environment is more complex and the obstacles are denser.

α_i, β_u are the potential energy weights when the AUV is farther away from the obstacles.

ω_l and ω_i are two thresholds relative to the potential energy value, which correspond to the boundaries where the potential energy weight transitions from the base weight to increase or decrease, respectively.

ω_u and ω_s correspond to the thresholds where the distance weights transition from the base weights to increase or decrease, respectively.

In areas with more complex or dangerous environments, $\alpha > \beta$ to emphasise safety.

In regions that are open, far from the end point, and relatively safe, $\alpha < \beta$ to optimise the path length and improve navigational efficiency.

As shown in Figure 4, the local traction potential energy field between AUV navigation paths is established, the obstacles are covered with the potential energy field, and the value at the centre of the grid represents the potential energy value $\mathrm{E}_{\mathrm{o(p)}}$ of the network p. The nodes to be traversed are indicated using open-set, and the nodes that have already been traversed are indicated using close-set.

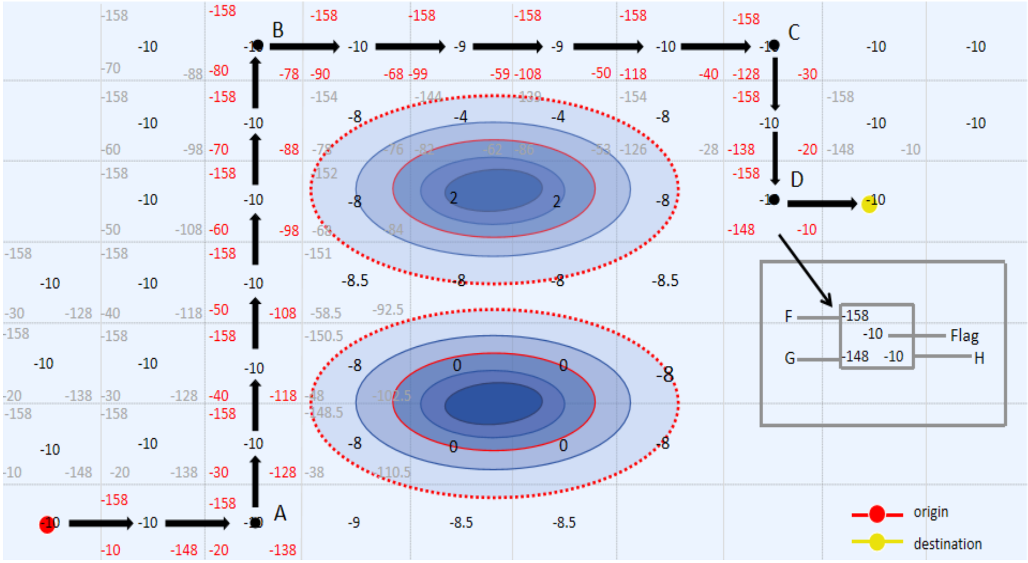


Figure 4. Schematic diagram of FT-A* algorithm path planning.

Taking the data in the grey box on the right side of the picture as an example, −158 denotes the total evaluation cost in the A* algorithm, i.e., F in the A* algorithm; −148 denotes the generation value of all nodes of the route path from the starting point, i.e., G in the A* algorithm; -10 in the lower right corner of the grid denotes the weighted generation value from the raster to the destination, i.e., H in the A* algorithm; and in the centre of the grid, Flag $= E(p)$ denotes that the generation value of the current generation value of the grid is -10.

The FT-A* algorithm has the following innovations compared to the traditional A* algorithm:

(1) The G-value and H-value are redefined. Compared to the traditional algorithm that uses distance as a criterion to evaluate the cost of a path, the FT-A* algorithm uses the weighted value of the potential energy value and distance as the new criterion, which takes into account the kinematic characteristics of the AUV itself as well as the threat cost of the obstacles, and helps to avoid planning a path that passes through the high-risk areas directly even though they may be the shortest possible paths.

(2) The size of the potential energy field is not fixed, but can be dynamically adjusted according to the kinematic characteristics of the AUV (e.g., parameters such as velocity and acceleration). This dynamic adjustment allows the potential energy field to adapt to instantaneous changes in the environment, enabling the AUV to maintain efficient and safe navigation when in different undersea environments.

3.3. Improved FT-A* Algorithm

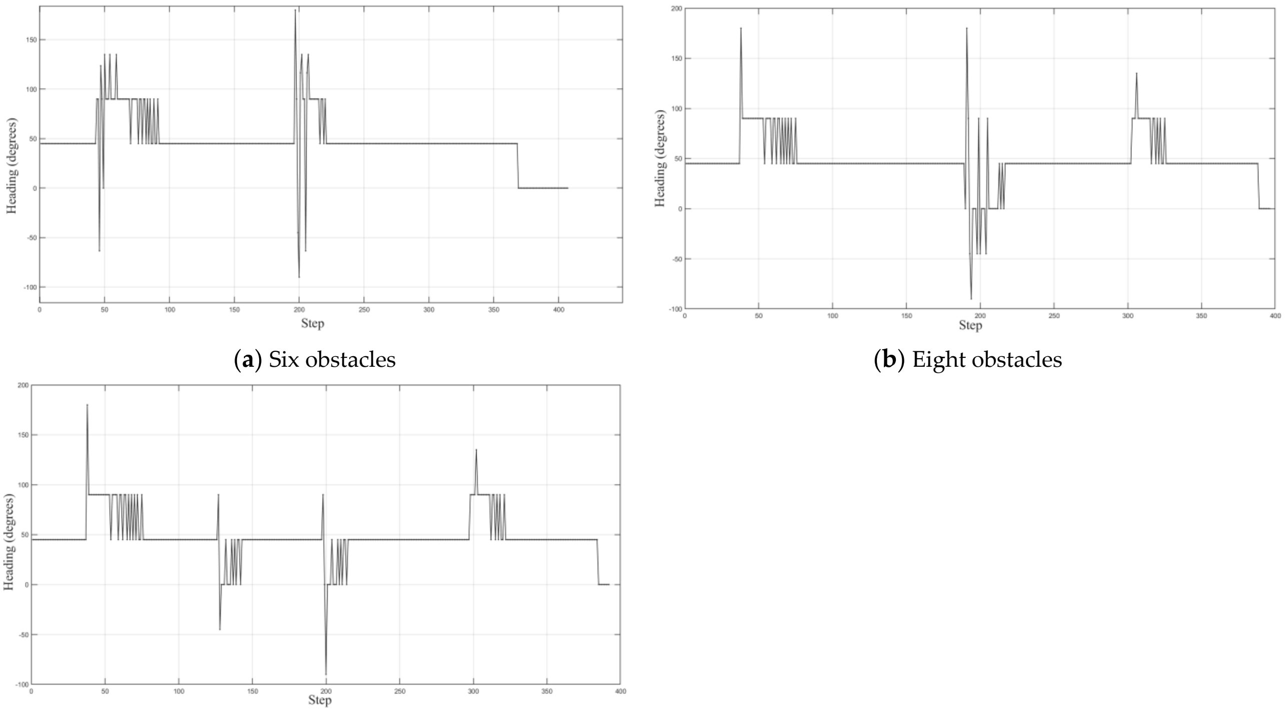
The introduction of potential energy fields solves the problem of searching for local shortest paths and obstacle avoidance for AUVs, but the inherent flaws of the algorithm result in the found paths still not being optimal. The limitations of the algorithm are summarised in this section as follows:

(1) The paths generated by the FT-A* algorithm need to be improved in terms of smoothness and continuity, and the oversteering does not match the actual situation of AUV underwater navigation.

(2) It still cannot completely solve the dynamic constraint problem. Particularly, the steering angle change of AUV during underwater navigation is not considered. As shown in Figure 5, at a certain moment, its steering angle changes more than 180° compared with the previous moment.

(3) The traditional A* algorithm produces too many redundant nodes, leading to inefficient search.

For the path optimisation problem, some current research has considered the dynamic constraint problem of the research object to make the path smoother. For example, Liang et al. (2013) used the maximum steering angle as a constraint for path planning of unmanned surface vehicle (USV) [30]. Zhang et al. used Dubins curves to satisfy the steering angle constraints of the AUV [31]. Petres (2007) considered the AUV's dynamics equations to control the curvature of the planned path [32]. However, the above studies are independent of the position information of the AUV itself, which makes it difficult to effectively improve the navigation efficiency of the AUV. For this reason, to address the above limitations, the algorithm designed in this paper first calculates the potential energy values between nodes through a path simplification method using a mathematical model and removes the redundant nodes to improve the navigation efficiency. Secondly, for the problem of sharp turns in the path, the path is smoothed by combining the Dubins curve to ensure that the path improves smoothness and continuity under the premise of meeting the steering angle limit of the AUV, as shown in Figure 6.



(**a**) Six obstacles

(**b**) Eight obstacles

(**c**) Ten obstacles

Figure 5. Real-time variation of heading angle under different working conditions.

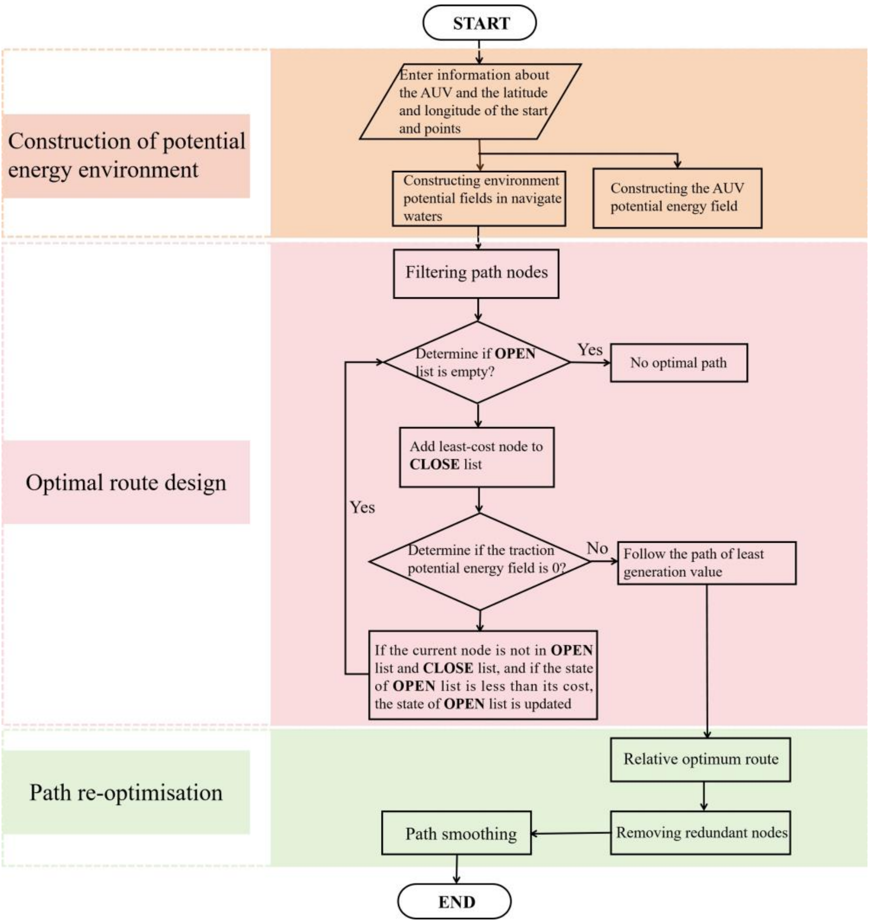

START
Construction of potential energy environment
Enter information about the AUV and the latitude and longitude of the start and points
Constructing environment potential fields in navigate waters
Constructing the AUV potential energy field
Optimal route design
Filtering path nodes
Determine if OPEN list is empty?
Yes
No optimal path
Add least-cost node to CLOSE list
Yes
Determine if the traction potential energy field is 0?
No
Follow the path of least generation value
If the current node is not in OPEN list and CLOSE list, and if the state of OPEN list is less than its cost, the state of OPEN list is updated
Path re-optimisation
Relative optimum route
Removing redundant nodes
Path smoothing
END

Figure 6. Schematic diagram of the improved FT-A* algorithm path planning.

3.3.1. Removal of Redundant Nodes/Path Simplification

In order to improve the efficiency of navigation and to solve the problem of redundant nodes, which often occurs in traditional algorithms, we simplify the planned route (as shown in Figure 7). This simplification aims to reduce unnecessary steering points while maintaining the integrity of the path. The path simplification process is described by the following mathematical model:

(1) Calculation of potential energy value

The safe potential energy value at the spatial coordinates (x, y, z) is $E(x, y, z)$

(2) Defining nodes and paths

Root node: O

Sub nodes: $A, B, C, D \ldots \ldots$

Connect the line segments: $\overline{P_iP_{i+1}}$, where $P_i = (x_i, y_i, z_i)$, $P_{i+1} = (x_{i+1}, x_{i+1}, x_{i+1})$, and all the raster points on the path between P_i and P_{i+1} are the set $S_{i,i+1}$.

(3) Comparison of potential energy between nodes

Let the potential energy values of all grid points through which the connected line segment $\overline{P_iP_{i+1}}$ passes be the set $V_{i,i+1}$, where

$$V_{i,i+1} = \{E(x, y, z) | (x, y, z) \in S_{i,i+1}\} \tag{27}$$

(4) Decision-making rules

If $V_{i,i-1} = V_{i,i+1}$, then P_i is viewed as a redundant node and is removed from the path and continues to be checked backward; otherwise, P_i continues to be retained.

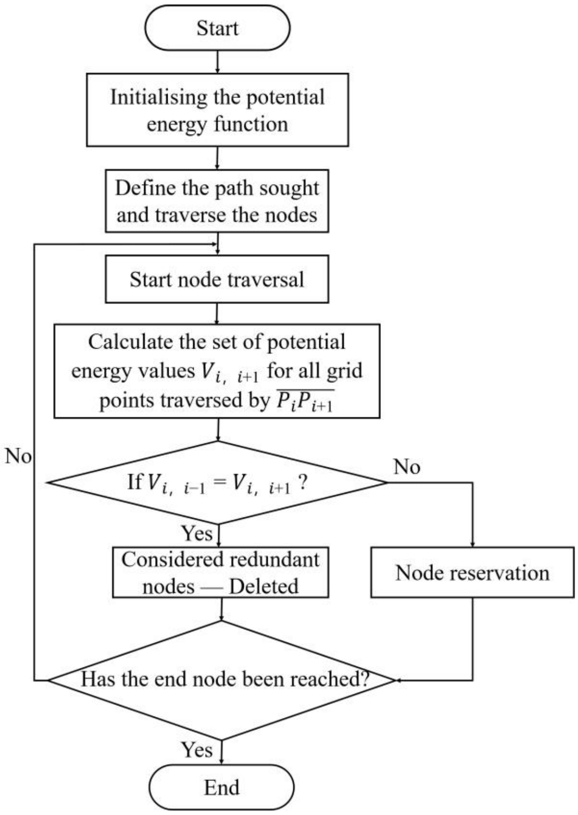

Start
Initialising the potential energy function
Define the path sought and traverse the nodes
Start node traversal
Calculate the set of potential energy values $V_{i,\ i+1}$ for all grid points traversed by $\overline{P_iP_{i+1}}$
No
If $V_{i,\ i-1} = V_{i,\ i+1}$?
No
Yes
Considered redundant nodes — Deleted
Node reservation
Has the end node been reached?
Yes
End

Figure 7. Flowchart for removing redundant nodes.

3.3.2. Path Smoothing Process

Redundant nodes are eliminated through potential energy comparison, yet issues like sharp turns and inadequate path smoothing persist. To enhance path practicality and navigation efficiency, it is crucial to consider the AUV's kinematic characteristics, particularly the steering angle limitations, during path smoothing. To address this, we integrate Dubins curves with the FT-A* algorithm to refine path smoothness and continuity while adhering to the AUV's kinematic constraints. Dubins' 1975 research identified that the shortest path between two points in a plane, considering minimum turning radius and steering limits, combines arcs and straight lines. This approach not only optimizes path smoothness but also aligns with the AUV's directional and speed requirements underwater.

(1) Dubins path existence determination

When designing the Dubins path, it is essential to find the tangent point between the straight-line segment and the two circular arcs. If such a tangent point cannot be determined, the Dubins path does not exist. This process is illustrated in Figure 8, which shows the geometric relationship between the straight line and the circular arcs. The mathematical expression for this condition is as follows:

$$\begin{cases} |r_t - r_s| < |c| \\ |r_t + r_s| < |c| \end{cases} \tag{28}$$

where r_t denotes the radius of the terminating arc, r_s denotes the radius of the starting arc, and c denotes the length of the line joining the centres of the two arcs.

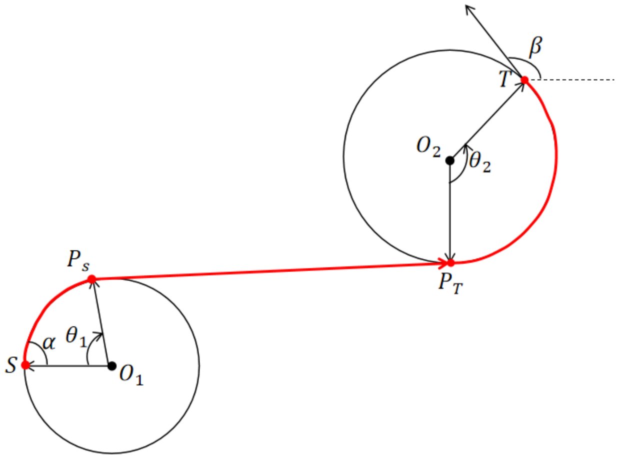


Figure 8. Schematic diagram of Dubins path curve.

(2) Parameter calculation

For the path planning problem with known start position point and end position point, once the existence of Dubins path is verified by Equation (28), then there exists the path designed to improve the FT-A* algorithm by using Dubins curves, and solving this path only requires solving the parameters of the Dubins path, and solving the parameters of the path is performed by the differential vector method, assuming that the start point is S, the end point is T, and the position transformation. The vector $\vec{P}$ is expressed using a mathematical model as follows:

$$\vec{P} = \vec{r_{1s}} - \vec{r_{1t}} + \vec{a} + \vec{r_{2s}} - \vec{r_{2t}} \tag{29}$$

where $\vec{r_{1s}}$, $\vec{r_{1t}}$, $\vec{r_{2s}}$ and $\vec{r_{2t}}$ are the centre vectors of the starting and ending arcs, respectively, $\vec{a}$ is a linear vector, as shown below:

$$\begin{cases} \underset{r_{1s}}{\rightarrow} = r_s \begin{bmatrix} \sin\alpha \\ -\cos\alpha \end{bmatrix} \\ \vec{r_{2s}} = r_s R(\theta_1) \begin{bmatrix} \sin\alpha \\ -\cos\alpha \end{bmatrix} \\ \vec{a} = aR(\theta_1) \begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} \\ \vec{r}_{1t} = -r_t R(\theta_1) \begin{bmatrix} \sin\alpha \\ -\cos\alpha \end{bmatrix} \\ \vec{r}_{2t} = -r_t R(-\theta_2) R(\theta_1) \begin{bmatrix} -\sin\alpha \\ -\cos\alpha \end{bmatrix} \end{cases} \tag{30}$$

where $R(\theta_1)$ is the rotation matrix.

This is obtained by substituting Equation (30) into (29) and simplifying as follows:

$$\frac{C_1}{\sqrt{{C_1}^2 + {C_2}^2}} sin(\alpha - \theta_1) - \frac{C_2}{\sqrt{{C_1}^2 + {C_2}^2}} cos(\alpha - \theta_1) = \frac{-(r_1 + r_2)}{\sqrt{{C_1}^2 + {C_2}^2}} \tag{31}$$

where

$$\begin{cases} C_1 = x_t - x_s - r_1 \sin\alpha - r_2 \sin\beta \\ C_2 = y_t - y_s + r_1 \cos\alpha + r_2 \cos\beta \end{cases} \tag{32}$$

x_s, y_s, x_t, y_t denote the horizontal and vertical coordinates of the start point and the termination point, respectively, and C_1, C_2 are the horizontal and vertical coordinate differences based on the start point and the termination point.

A basic angle value γ_0, calculated based on the coordinate difference between the start point and the termination point, has the mathematical expression shown below:

$$\gamma_0 = arcsin(\frac{C_2}{\sqrt{C_1^2 + C_2^2}}) \tag{33}$$

If the quadrant position in the actual coordinate system is considered, the basic angle value γ is adjusted by the following conditions:

$$\gamma = \begin{cases} \gamma_0 & if\ C_1 > 0, C_2 > 0 \\ 2\pi + \gamma_0 & if\ C_2 < 0, C_1 > 0 \\ \pi - \gamma_0 & if\ C_1 < 0 \end{cases} \tag{34}$$

From Equation (31)

$$\theta_1 = \begin{cases} 2k\pi - \arcsin\frac{-(r_1+r_2)}{\sqrt{C_1^2+C_2^2}} + \alpha - \gamma \\ (2k-1)\pi + \arcsin\frac{-(r_1+r_2)}{\sqrt{C_1^2+C_2^2}} + \alpha - \gamma \end{cases} \tag{35}$$

Since $\alpha - \theta_1 + \theta_2 = \beta$, after solving θ_2, we can solve for the centre of circle O_1 and O_2, as well as the tangent points P_S and P_T of straight line and circular arc (the starting and ending points of straight line portion) by analytical geometry. O_1 and O_2 are the starting and ending arc circumcentric angles of the path obtained by FT-A* algorithm through the smoothing process of Dubins curve, and the complete path curves can be plotted by the known parameters of paths above. The path curve can be plotted by knowing the above path parameters, and the optimised path, according to this method, not only takes into account the steering angle constraints of the AUV, but also solves the problem of insufficient smoothness of the path.

4. Simulation Experiment and Analysis

In order to verify the accuracy of the results, simulation experiments were conducted using MATLAB R2022b to compare the traditional A* algorithm, the FT-A* algorithm, and the improved FT-A* algorithm. Subsequently, in order to further enhance the evaluation quality of this study, two algorithms most suitable for the current experimental validation scenario, the RRT* algorithm and the EACO algorithm, were selected as the control group as a means of verifying the outstanding advantages of the improved FT-A* algorithm.

These two algorithms are chosen as a control group because of their excellent performance in path planning. The RRT algorithm is widely recognized in 3D path planning, especially in multi-obstacle and non-linear environments. However, due to its non-heuristic nature, it is difficult to guarantee globally optimal paths. RRT* introduces a powerful global path planning feature that fills the gap of RRT, thus addressing this limitation. In contrast, ACO (ant colony optimization) is highly praised for its powerful global optimization potential, which allows for dynamic path adjustment in complex, obstacle-laden environments. However, the inherent stochastic nature of ACO makes it susceptible to local optimization, a problem that can be mitigated by EACO (enhanced ACO).

The primary objectives of this simulation study are threefold:

1. Simulation Setup: To thoroughly describe the route planning process of the FT-A* algorithm, establishing the simulation framework for path generation.
2. Comparative Analysis: To conduct a comparative analysis of the planned paths generated by different algorithms, assessing their performance across a variety of conditions.
3. Robustness Testing: To evaluate the robustness of the improved FT-A* algorithm, confirming its ability to generate feasible paths in environments of varying complexity.

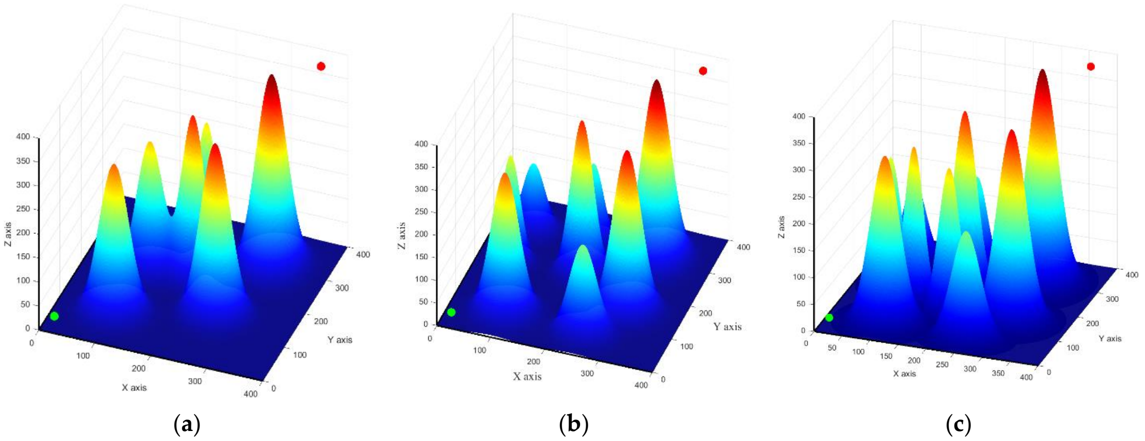
4.1. Simulation Experiment

Firstly, basic parameter information setting was established. As shown in Table 3:

Table 3. Parameterisation.

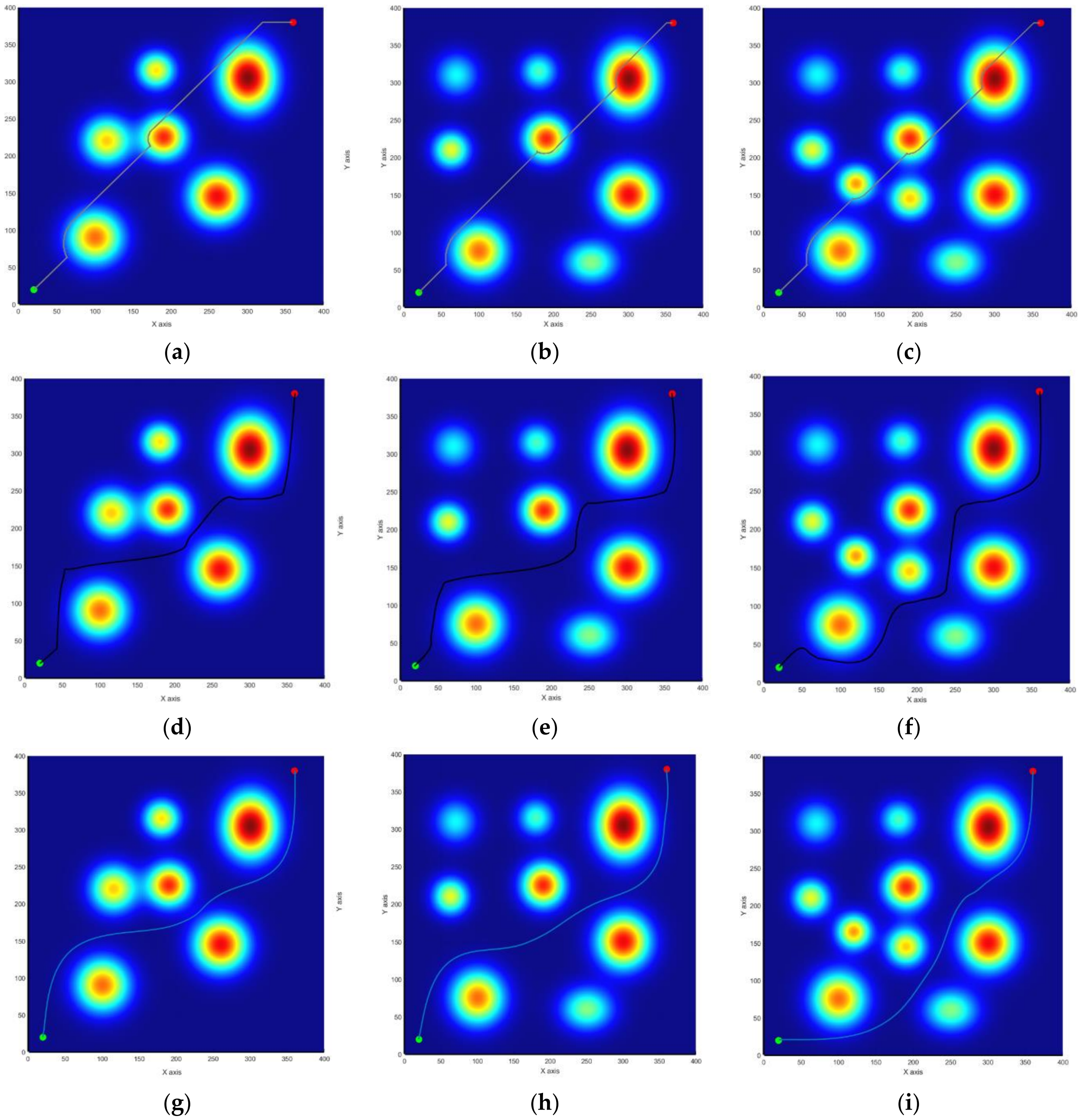
Parameter	Value
Map size	400 × 400 × 400
Origin	(20, 20, 20)
Destination	(480, 480, 480)
Maximum speed	4

This simulation experiment sets up three working conditions, i.e., the implementation of all algorithms under six obstacles, eight obstacles, and ten obstacles. The experimental environment is shown in Figure 9; the results of the run are shown in Figure 10.



(**a**) (**b**) (**c**)

Figure 9. Schematic diagram of working conditions. (**a**) Six-obstacle working condition; (**b**) Eight-obstacle working condition; (**c**) Ten- obstacle working condition.



(**a**) (**b**) (**c**)

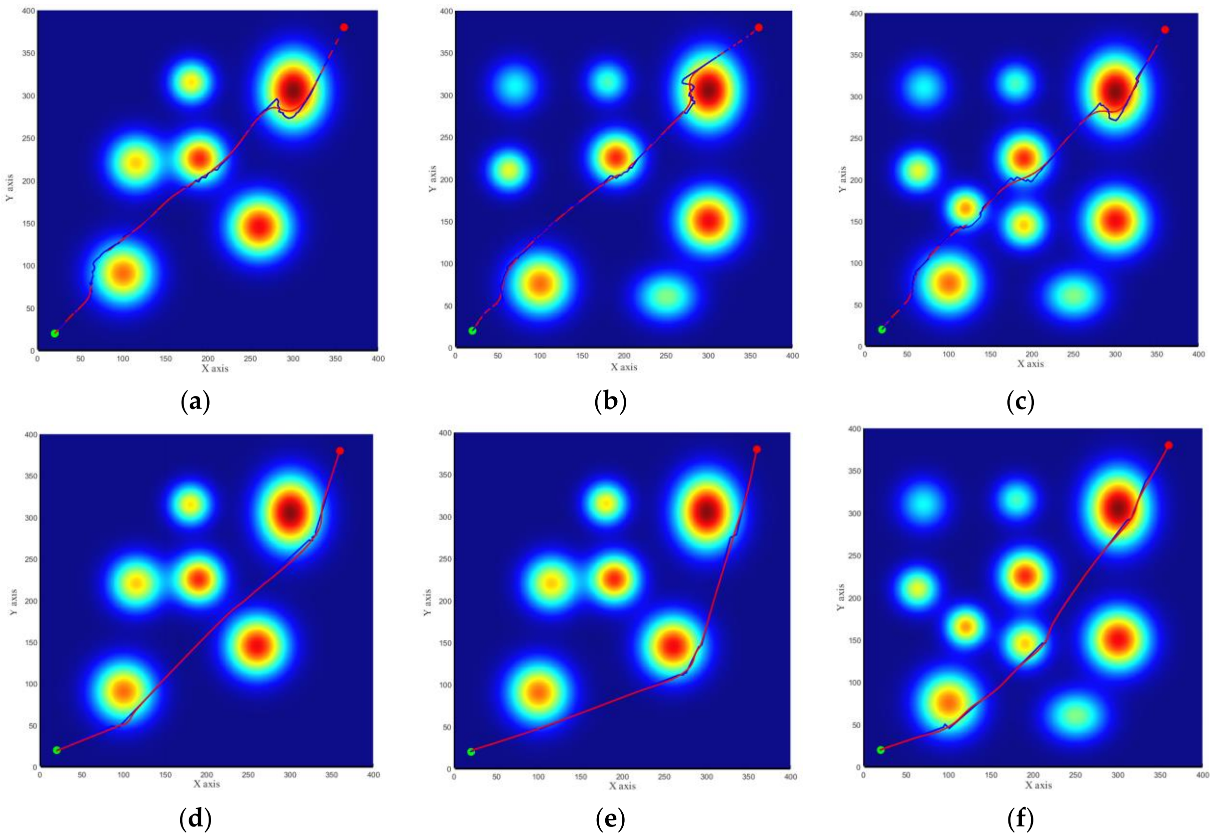
(**d**) (**e**) (**f**)

(**g**) (**h**) (**i**)

Figure 10. Display of experimental results under different algorithms. (**a**–**c**): A* algorithm's operating path under different operating conditions; (**d**–**f**): FT-A* algorithm's operating path under different operating conditions; (**g**–**i**): Improved FT-A* algorithm's operating path under different operating conditions.

Next, the experimental results were considered.

The experimental results for the control group are shown in Figure 11, where the blue lines represent the paths planned by the traditional algorithm and the red lines indicate the paths planned by the improved algorithm.



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 11. Display of experimental results under different algorithms. (**a**–**c**): Schematic diagram of planned paths for RRT and RRT*; (**d**–**f**): Schematic diagram of planned paths for ACO and EACO.

The results of the running statistics are reflected in Table 4:

Table 4. Analysis of operational results.

		A*	**FT-A***	**RRT***	**EACO**	**Im FT-A***
Six obstacles	Distance (m)	648.3	664.7	651.3	653.1	657.4
	Number of nodes	9766	9864	13,916	19,156	6753
	Search time (s)	0.26	0.31	1.28	6.74	0.69
Eight obstacles	Distance (m)	640.7	656.3	644.8	661.6	648.7
	Number of nodes	10,322	10,376	16,483	23,248	6458
	Search time (s)	0.29	0.43	2.06	7.71	0.67
Ten obstacles	Distance (m)	639.5	672.4	647.9	658.4	667.1
	Number of nodes	10,218	10,421	19,042	26,391	6137
	Search time (s)	0.38	0.57	2.97	8.78	0.74

4.2. Analysis of Experimental Results

The results indicate that the improved FT-A* algorithm significantly enhances the efficiency and performance of path planning while ensuring the safety and feasibility of the generated paths. In the control group, both the RRT* and EACO algorithms generate better path distances compared to the improved FT-A* algorithm; however, these algorithms primarily focus on avoiding geometric collisions, aiming to prevent direct impacts without considering the threat levels of obstacles or other environmental factors (e.g., the closer an obstacle is, the greater the associated threat). Consequently, the paths generated by the RRT* algorithm are often positioned dangerously close to obstacles. A similar issue is present in the EACO algorithm, which addresses the shortcomings of the ACO algorithm in terms of path smoothness. However, the stochastic nature of EACO during the exploration phase leads to slow convergence and higher computational demands. Furthermore, EACO fails to model the kinematic characteristics of the AUV (such as steering radius and speed limitations), treating the AUV as a point mass. This results in paths that remain unacceptably close to obstacles, making EACO unsuitable for direct application in AUV path planning.

In contrast, the improved FT-A* algorithm greatly enhances navigation capability in complex underwater environments, reducing computational resource consumption while maintaining excellent path quality. The advantages of the improved algorithm become increasingly apparent in more complex operational environments. For instance, in a simple environment with six obstacles, arithmetic power consumption decreases by 31.1% at the expense of path length, while in a more complex environment with ten obstacles this reduction reaches 37.8%, and both greatly improve navigational safety. This research is expected to be applied to more complex environments.

5. Summary and Outlook

The algorithm proposed in this paper enables accurate and safe path planning for AUVs, thus ensuring safe travelling of AUVs in complex multi-obstacle environments. Firstly, the kinematic and dynamic models of AUVs and their six-degree-of-freedom motions underwater are understood with respect to the basic characteristics of AUVs. On this basis, the concept of field theory is introduced, and the FT-A* algorithm is proposed, which makes up for the shortcomings of traditional A* algorithms that tend to treat the study object as a prime point, ignoring its own dynamic constraints and obstacle threat costs. After solving the navigation safety problem, considering that the algorithm may generate a large number of redundant nodes as well as insufficient path smoothness and continuity, the improved FT-A* algorithm is further proposed to improve the navigation efficiency and remove the redundant nodes. The results of the simulation experiments show that the algorithm has significant advantages in terms of security and executability compared with the traditional A* algorithm and other good path planning algorithms and is expected to support related fields in future practical applications.

However, some limitations of the algorithm need to be noted.

(1) The algorithm proposed in this paper is an empirical algorithm with potential applications and shows good performance under specific conditions, but there is no guarantee that the searched paths are optimal paths, and it may not be suitable for large-scale navigation in unknown environments.

(2) In practical applications, the target area of an AUV is usually a sea area rather than a point, so when the AUV navigates to the end point, it may cause oscillations at the end point boundary due to the influence of the safety potential field of the obstacle.

(3) Although the effectiveness of the algorithm has been validated under ideal conditions, in real applications, the underwater environments in which AUVs are located tend to be complex and variable, with high demands on the accuracy of the environmental maps, which may pose a greater challenge to the implementation of the algorithm.

Author Contributions: The authors acknowledge the following contributions to this paper: Investigation, Z.X. (Zhiyuan Xu); Methodology, Z.X. (Zhiyuan Xu) and Y.L.; Resources, Y.S., Z.X. (Zhexue Xie), and Y.L.; Software, Z.X. (Zhiyuan Xu); Writing—original draft, Z.X. (Zhiyuan Xu); Writing—review & editing, Y.S., Z.X. (Zhexue Xie) and Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Research and Application Demonstration Project of Key Technologies for Safeguarding of Container Vessels in Ningbo Zhoushan Port Based on Intelligent Navigation under grant ZJHG-FW-2024-27, the Shanghai Commission of Science and Technology Project under grants 21DZ1201004 and 2300501900, the Anhui Provincial Department of Transportation Project under grant 2021-KJQD-011, the National Natural Science Foundation of China under grant 51509151, and in part by the Shandong Province Key Research and Development Project under grant 2019JZZY020713.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: Yong Shen and Zhexue Xie were employed by Ningbo Dagang Pilotage Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Ghafoor, N.; Noh, Y. An Overview of Next-Generation Underwater Target Detection and Tracking: An Integrated Underwater Architecture. *IEEE Access* **2019**, *7*, 98841–98853. [CrossRef]
2. Sun, S.; Zhang, X.; Zheng, C.; Fu, J.; Zhao, C. Underwater Acoustical Localization of the Black Box Utilizing Single Autonomous Underwater Vehicle Based on the Second Order Time Difference of Arrival. *IEEE Ocean Eng.* **2020**, *45*, 1268–1279. [CrossRef]
3. Zhao, M.; Gao, Y.Q.; Wu, D.X.; Wang, P.; Zhang, H.G. AUV Global Path Planning Method in Complex Sea Battle Field Environment. *Technology* **2021**, *43*, 41–48.
4. Yan, Z.P.; Zhang, J.Z.; Tang, J.L.; Zeng, J. Three-Dimensional Path Planning for Autonomous Underwater Vehicles Based on a Whale Optimization Algorithm. *Ocean Eng.* **2022**, *250*, 111070. [CrossRef]
5. Cheng, C.X.; Sha, Q.X.; He, B.; Li, G.L. Path Planning and Obstacle Avoidance for AUV: A Review. *Ocean Eng.* **2021**, *235*, 109355. [CrossRef]
6. LaValle, S.M.; Rapidly, R. *Exploring Random Trees: A New Tool Path Planning*; Iowa State University: Ames, IA, USA, 1998.
7. Tu, H.Y.; Deng, Y.Z.; Li, Q.Y.; Song, M.J.; Zheng, X.J. Improved RRT Global Path Planning Algorithm Based on Bridge Test. *Robot. Auton. Syst.* **2023**, *171*, 104570. [CrossRef]
8. Ding, S.; Chen, M.M.; Wang, M. RO Global Path Planning Method Based on RRT* Algorithm. *Ship Sci. Technol.* **2019**, *41*, 66–73.
9. Cordon, O.; Herrera, F.; Lozano, M. Genetic Algorithms and Fuzzy Logic in Control Processes. *Control Sci. Arch.* **1996**, *5*, 135–168.
10. Li, Q.; Zhang, C.; Han, C.W.; Zhang, T.; Zhang, W.C. Fuzzy Logic Algorithm Based Path Planning for Mobile Robot in Dynamic Environment. *J. Cent. South Univ. Nat. Sci. Ed.* **2013**, *44*, 104–108.
11. Duan, Q.J.; Zhang, M.; Zhang, Q. Local Path Planning Method for AUV Based on Fuzzy-Neural Network. *Ship Eng.* **2001**, *1*, 54–58.
12. Arulkumaran, K.; Deisenroth, M.P.; Brundage, M.; Bharath, A.A. Deep Reinforcement Learning: A Brief Survey. *IEEE Signal Process. Mag.* **2017**, *34*, 26–38. [CrossRef]
13. Xu, Z.Z.; Haroutunian, M.; Alan, J.M.; Neasham, J.; Norman, R. A Comparison of Functional Control Strategies for Underwater Vehicles: Theories, Simulations, and Experiments. *Ocean Eng.* **2020**, *215*, 107822. [CrossRef]
14. Haupt, R.L.; Werner, D.H. *Genetic Algorithms in Electromagnetics*; John Wiley and Sons: Hoboken, NJ, USA, 2007.
15. Eberhart, R.; Kennedy, J. A New Optimizer Using Particle Swarm Theory. In Proceedings of the MHS'95. Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October 1995; pp. 39–43.
16. Dorigo, M.; Colorni, A.; Maniezzo, V. Distributed Optimization by Ant Colonies. In Proceedings of the First European Conference on Artificial Life, Paris, France, 11–13 December 1991; pp. 134–142.
17. Ren, Y.; Wang, J.X.; Zhang, X.G. Research on AUV Path Planning Based on Multi-factor Improved A* Algorithm. *Ship Sci. Technol.* **2022**, *44*, 58–62.
18. Ma, Y.H.; Zhang, H.; Qi, L.R. Three-Dimensional UAV Path Planning Based on Improved A* Algorithm. *Electro-Opt. Control* **2019**, *26*, 22–25.
19. Xu, X.; Zeng, J.Z.; Zhao, Y.; Lu, X.S. Research on Global Path Planning Algorithm for Mobile Robots Based on Improved A*. *Expert Syst. Appl.* **2024**, *243*, 122922. [CrossRef]
20. Szczerba, R.J.; Galkowski, P.; Glicktein, I.S.; Ternullo, N. Robust Algorithm for Real-Time Route Planning. *IEEE Trans. Aerosp. Electron. Syst.* **2000**, *36*, 869–878. [CrossRef]
21. Cai, C.J. Research on Collision Hazard of Ships in Waters Based on Field Theory. Master's Thesis, Dalian Maritime University, Dalian, China, 2015.
22. Chen, S.; Liu, C.W.; Huang, Z.P.; Cai, G.S. Global Path Planning for AUV Based on Sparse A* Search Algorithm. *Torpedo Technol.* **2012**, *20*, 271–275.
23. Li, M.C.; Zhang, H.J. AUV 3D Path Planning Based on A* Algorithm. In Proceedings of the 2020 Chinese Automation Congress (CAC), Shanghai, China, 6–8 November 2020.
24. Fossen, T.I. *Guidance and Control of Ocean Vehicles*; John Wiley & Sons: Hoboken, NJ, USA, 1994.
25. Zhuang, Y.; Sharma, S.; Subudhi, B.; Huang, H.; Wan, J. Efficient Collision-Free Path Planning for Autonomous Underwater Vehicles in Dynamic Environments with a Hybrid Optimization Algorithm. *Ocean Eng.* **2016**, *127*, 190–199. [CrossRef]
26. Gan, W.Y.; Li, D.Y.; Wang, J.M. A Hierarchical Clustering Method Based on Data Field. *J. Electron.* **2006**, *34*, 258–262.
27. Lisowski, J. Determining the Optimal Ship Trajectory in Collision Situation. In Proceedings of the IX International Scientific and Technical Conference on Marine Traffic Engineering, Szczecin, Poland, 19–22 October 2001; pp. 192–201.

28. Ni, D.H.; Wang, H. A Unified Perspective on Traffic Flow Theory, Part III: Validation and Benchmarking. *Appl. Math. Sci.* **2013**, *7*, 1965–1982. [CrossRef]
29. Wang, J.; Wu, J.; Li, Y. The Driving Safety Field Based on Driver-Vehicle-Road Interactions. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2203–2214. [CrossRef]
30. Liang, Y.; Zhang, Y.; Lei, J.W. An On-line Fast Route Planning Method Based on Dubins Path. *J. Syst. Simul.* **2013**, *25*, 291–296.
31. Zhang, Y.G.; Zhang, Y.A.; Wu, Y.S. Research on Multi-Aircraft Collaborative Online Route Planning Based on Dubins Path. In *The Sixth Annual Doctoral Conference, Chongqing*; China Science and Technology Press: Beijing, China, 2008.
32. Petres, C.; Pailhas, Y.; Patron, P.; Petillot, Y.; Evans, J.; Lane, D. Path Planning for Autonomous Underwater Vehicles. *IEEE Trans. Robot.* **2007**, *23*, 331–341. [CrossRef]

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.