Hindawi
Mathematical Problems in Engineering
Volume 2019, Article ID 2902170, 10 pages
https://doi.org/10.1155/2019/2902170

Research Article

Global Path Planning for Unmanned Surface Vehicle Based on Improved Quantum Ant Colony Algorithm

Guoqing Xia, Zhiwei Han, Bo Zhao, Caiyun Liu, and Xinwei Wang

College of Automation, Harbin Engineering University, Harbin 150001, China

Correspondence should be addressed to Zhiwei Han; hanzhiwei@hrbeu.edu.cn

Received 17 December 2018; Revised 19 March 2019; Accepted 14 April 2019; Published 24 April 2019

Academic Editor: Mahmoud Mesbah

As a tool to monitor marine environments and to perform dangerous tasks instead of manned vessels, unmanned surface vehicles (USVs) have extensive applications. Because most path planning algorithms have difficulty meeting the mission requirements of USVs, the purpose of this study was to plan a global path with multiple objectives, such as path length, energy consumption, path smoothness, and path safety, for USV in marine environments. A global path planning algorithm based on an improved quantum ant colony algorithm (IQACA) is proposed. The improved quantum ant colony algorithm is an algorithm that benefits from the high efficiency of quantum computing and the optimization ability of the ant colony algorithm. The proposed algorithm can plan a path considering multiple objectives simultaneously. The simulation results show that the proposed algorithm's obtained minimum was 2.1–6.5% lower than those of the quantum ant colony algorithm (QACA) and ant colony algorithm (ACA), and the number of iterations required to converge to the minimum was 11.2–24.5% lower than those of the QACA and ACA. In addition, the optimized path for the USV was obtained effectively and efficiently.

1. Introduction

An unmanned surface vehicle (USV) is a kind of autonomous marine vehicle. Determining the path of a USV is an important problem associated with its safety and efficiency [1]. Depending on whether the environmental information is obtained from a digital map or sensors, path planning is divided into global and local stages [2]. In this paper, a USV global path planning study is presented. Global path planning is the process of planning a path to connect the starting and destination points under a given planning space from a digital map and constraints according to the mission requirements. The indices for evaluating a path can be path length, energy consumption, path smoothness, and path safety.

Obtaining a short path from the starting point to the destination point is one of the main objectives of global path planning. Planning the shortest path is an NP-hard problem [3]. Existing methods take the path length as a single objective of the path planning, and neither energy consumption nor other indices are considered.

The energy consumption during sailing determines the USV's endurance and the duration of the mission. Since the environmental loads such as wind, waves, and ocean currents influence the performance of the USV, the calculation of the USV's energy consumption is complex. Niu et al. [4] considered the effect of the ocean current on the energy consumption of USVs. Lee et al. [5] found a more economical path by considering the shallow water effect as well as tidal currents and wind for surface ship navigation. Most calculations of energy consumption have considered the effects of ocean currents on the USV without considering wind and waves.

The smoothness of a path depends on the size and number of the turns that the USV makes while sailing along the planned path. The smoother path allows the USV to make fewer turns along the path, which reduces the mechanical wear on the steering actuators, such as rudders. Smooth paths can reduce unnecessary curvature discontinuities and possible stops. In a previous report [6], the smoothness of a path was evaluated by summing the angles of each turn on the path that the vehicle follows. Ma et al. [7] evaluated the turn angle set by adopting the maximum value of the turn angle set to assess the path smoothness for the USV.

Obstacles such as islands and reefs affect the safety of the USV. Path safety means that the USV cannot collide with any obstacles while sailing. Ma et al. [7] used circles that just covered the obstacles to identify the safe area.

Since USV global path planning involves optimization algorithms, environmental models, and marine craft hydrodynamics, existing path planning algorithms have difficulty meeting the mission requirements. Intelligent optimization algorithms are widely used in global path planning, such as the genetic algorithm [8], particle swarm algorithm [9], NSGA-II [10], and ant colony algorithm [11]. With the development of quantum technology, the idea of combining quantum computing with intelligent optimization algorithms has been developed. Narayanan and Moore combined quantum mechanics principles and evolutionary computing methods for the first time [12]. A quantum bit and superposition of states were proposed to solve the knapsack problem by a quantum-inspired evolutionary algorithm (QEA) [13]. Based on the QEA with a quantum rotation gate strategy, an adaptive evolution-based quantum-inspired evolutionary algorithm (AEQEA) introduces an adaptive evolution mechanism [14]. A new improved quantum evolution algorithm (IQEA) with a mixed local search procedure was proposed [15]. Li et al. [16] proposed a quantum ant colony algorithm (QACA) that combined quantum computing and the ant colony algorithm for continuous space optimization. You et al. [17] proposed a novel parallel ant colony optimization algorithm based on a quantum dynamics mechanism (PQACO). An improved quantum ant colony algorithm was proposed for the optimization of evacuation paths from dangerous areas to safe areas [18]. The quantum ant colony algorithm was used to determine campus path navigation [19].

In this paper, a global path planning algorithm for USV based on the improved quantum ant colony algorithm (IQACA) is proposed. The main contributions of the proposed approach are as follows:

(1) At present, most USV global path planning algorithms only search for a feasible path for one objective [3–6]. In this paper, path planning was considered with multiple simultaneous objectives, which were path length, energy consumption, path smoothness, and path safety.

(2) The IQACA is a new optimization algorithm that combines quantum-inspired computing with the ant colony algorithm (ACA). The quantum bit (Q-bit) is used to encode the pheromone in the ACA to obtain the quantum pheromone, and the ant movement is determined based on the concentration of the quantum pheromone on the path. Compared to the existing QACA [16–19], the phase of the quantum ant colony is transformed by an adaptive quantum rotation gate, and the quantum pheromone is updated by local and global update rules in the IQACA.

Simulation experiments in a complex environment with wind, waves, and ocean currents verified the effectiveness of the objective model, and we obtained a desired path based on the IQACA.

The paper is organized as follows. In Section 2, the USV path planning problem is established, and the USV kinetic model, environmental loads, and cost function of the path planning are described. In Section 3, the principles of the IQACA are provided, and we apply the IQACA to USV global path planning. In Section 4, the simulations for USV global path planning using the IQACA are presented. Conclusions are provided in Section 5.

2. Problem Statement

2.1. USV Kinetic Model. The kinetic model of a USV accounts for the forces, such as the control force and environmental loads, which cause USV motion. For the USV, the control force is mainly the thrust of each propeller. The environmental loads on the USV are generated by wind, waves, and ocean currents. The kinetic model of the USV, which was proposed previously [20], is as follows:

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu = \tau_{env} + \tau_{thr} \tag{1}$$

$$\tau_{env} = \tau_{wind} + \tau_{wave} + \tau_{current} \tag{2}$$

where M is the system inertia matrix, $C(\nu)$ is the Coriolis-centripetal matrix, $D(\nu) \in R^{3\times3}$ is damping matrix. τ_{wind}, τ_{wave}, and $\tau_{current}$ are wind, wave, and ocean current forces acting on the USV, respectively, and τ_{thr} is the thrust generated by the USV propulsion system. The generalized velocity $\nu = [u, v, r]^T$ is obtained by (1), where the first two components (u, v) are the linear velocities of the surge and sway, and r is the angular velocity of the yaw.

2.2. Models of Environmental Loads. When planning a global path for USVs, it is necessary to consider the environmental effects on the vehicles. Thus, we need to analyze the impacts of wind, waves, and ocean currents on the USV. The planned area is a confined sea with some static obstacles, and the mission execution time is short. Therefore, it can be assumed that the environmental loads are basically stable in limited time and space.

2.2.1. Wind Forces. The wind acts directly on the superstructure of the hull. As reported previously [21], the wind forces are written as follows:

$$\begin{aligned}
\tau_{windX} &= \frac{1}{2}\rho_a A_f V_w^2 C_{wx}(\alpha_R) \\
\tau_{windY} &= \frac{1}{2}\rho_a A_s V_w^2 C_{wy}(\alpha_R) \\
\tau_{windN} &= \frac{1}{2}\rho_a A_s V_w^2 C_{wn}(\alpha_R) \cdot L
\end{aligned} \tag{3}$$

where ρ_a is the density of air, A_f and A_s are the frontal and lateral projected areas, $C_{wx}(\alpha_R)$, $C_{wy}(\alpha_R)$, and $C_{wn}(\alpha_R)$ are the empirical force coefficients, α_R is the angle between the wind and the heading of the vessel, L is the length of the vessel, V_w is the relative wind speed, and τ_{windX}, τ_{windY}, and τ_{windN} are the wind forces during the surge, sway, and yaw, respectively [22].

2.2.2. Wave Forces. When a vehicle is sailing on the sea, the interference of wave forces is complicated. The wave forces acting on the hull are first- and second-order wave forces. The second-order wave forces, which impact the heading and path of the USV, are proportional to the square of the wave height [22]. The wave forces are simplified as follows:

$$\tau_{waveX} = \frac{K_{w1}s}{s^2 + 2\lambda_1\omega_{e1}s + \omega_{e1}^2}w_1 + d_1$$
$$\tau_{waveY} = \frac{K_{w2}s}{s^2 + 2\lambda_2\omega_{e2}s + \omega_{e2}^2}w_2 + d_2 \quad (4)$$
$$\tau_{waveN} = \frac{K_{w3}s}{s^2 + 2\lambda_3\omega_{e3}s + \omega_{e3}^2}w_3 + d_3$$

where w_i $(i = 1, 2, 3)$ are Gaussian white noise processes, and τ_{waveX}, τ_{waveY}, and τ_{waveN} are the wave forces during the surge, sway, and yaw, respectively. The amplitudes of τ_{waveX}, τ_{waveY}, and τ_{waveN} are adjusted by choosing the constants K_{wi} $(i = 1, 2, 3)$, while the spectra are parameterized in terms of the pairs λ_i and ω_{ei} $(i = 1, 2, 3)$. The wave drift forces d_i $(i = 1, 2, 3)$ are usually modeled as slowly varying bias terms:

$$\dot{d}_1 = w_4$$
$$\dot{d}_2 = w_5 \quad (5)$$
$$\dot{d}_3 = w_6$$

where w_i $(i = 4, 5, 6)$ are Gaussian white noise processes [22].

2.2.3. Ocean Current Forces. The ocean currents cause vessels sailing on the sea to change their positions and postures. The ocean current forces are given as follows:

$$\tau_{currentX} = \frac{1}{2}\rho A_f V_c^2 C_X(\beta)$$
$$\tau_{currentY} = \frac{1}{2}\rho A_s V_c^2 C_Y(\beta) \quad (6)$$
$$\tau_{currentN} = \frac{1}{2}\rho A_s V_c^2 C_N(\beta) \cdot L$$

where ρ is the density of the seawater, A_f and A_s are the frontal and lateral projected areas below the waterline, respectively, C_X, C_Y, and C_N are the empirical force coefficients, V_c is the relative current speed, β is the angle between the ocean current and the heading of the vessel, L is the length of the vessel, and $\tau_{currentX}$, $\tau_{currentY}$, and $\tau_{currentN}$ are the ocean current forces during surge, sway, and yaw, respectively [22].

2.3. Path Representation by Grids. The real task area is partitioned to reduce the modeling complexity. Visibility graphs [23], Voronoi diagrams [24], and grid maps [25] are the most commonly used path planning algorithms. The grid map-based path planning algorithm is powerful in that it generates a path with the shortest computation time [25]. To facilitate the calculation, the planned path is represented on grids. The area under consideration is discretized into grids. The information, such as the relative speed and direction of the wind, the amplitude and direction of the waves, the relative speed and direction of ocean current, and the position of the obstacles, is discretized in each grid. Stationary obstacles are encoded in a binary format on the grids. We assigned weights of 1 to all obstacle grids and weights of 0 to all free neighbor grids of them.

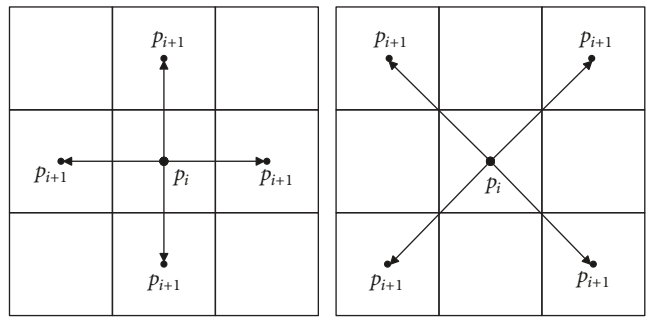

Figure 1: The positions of p_i and p_{i+1}.

2.4. Objectives of USV Global Path Planning. Since USV global path planning is a multiobjective optimization problem, we should analyze the interrelated objectives and discuss the importance of each objective based on the requirements of the mission. A cost function can be constructed as a weighted sum of the objective functions. Finally, the cost function is used to evaluate the quality of the planned path.

2.4.1. Path Length. Since the task area is modeled by grids, the planned path is represented on a rectangular grid. The path passes the centers of the grids. Thus, the distance $L_{i,i+1}$ between two adjacent waypoints $p_i = (x_i, y_i)$ and $p_{i+1} = (x_{i+1}, y_{i+1})$ is equal to the Euclidean distance between the centers of the grids as follows:

$$L_{i,i+1} = \begin{cases} 1, & x_i = x_{i+1} \text{ or } y_i = y_{i+1} \\ \sqrt{2}, & \text{otherwise} \end{cases} \quad (7)$$

The positions of p_i and p_{i+1} are shown in Figure 1. If p_i and p_{i+1} are adjacent in the horizontal or vertical direction, $L_{i,i+1} = 1$. If p_i and p_{i+1} are adjacent in the diagonal direction, $L_{i,i+1} = \sqrt{2}$.

Therefore, the total length of the path L is the sum of the distances between the adjacent waypoints:

$$L = \sum_{i=1}^{m} L_{i,i+1} \quad (8)$$

where m is the number of path segments.

2.4.2. Energy Consumption. In this paper, the energy consumption of the USV while sailing is derived from the propulsion system. Thus, E is the sum of the energy consumption of each segment along the entire path:

$$E = \sum_{i=1}^{m} E_{i,i+1} \tag{9}$$

Supposing that the USV is sailing at a constant velocity between p_i and p_{i+1}, the energy consumption $E_{i,i+1}$ between p_i and p_{i+1} equals the work done by the propulsion system to overcome the environmental loads, such that

$$E_{i,i+1} = \tau_{env} \cdot \left|\overrightarrow{v}_{usv}\right| \cdot t \tag{10}$$

where t is the time for the USV to sail in $L_{i,i+1}$.

$$t = \frac{L_{i,i+1}}{\left|\overrightarrow{v}_{out}\right|} \tag{11}$$

where $|\overrightarrow{v}_{usv}|$ is the magnitude of the velocity $\overrightarrow{v}_{usv}$ generated by the USV propulsion system, τ_{env} is the resultant force of the environmental loads, and $|\overrightarrow{v}_{out}|$ is the magnitude of the velocity $\overrightarrow{v}_{out}$ of the USV moving in the horizontal plane. Since the headings of the USV in the grid are several fixed values, as shown in Figure 1, the angular velocity r caused by the yaw motion can be ignored when solving $|\overrightarrow{v}_{out}|$. Hence, $|\overrightarrow{v}_{out}|$ is equal to

$$\left|\overrightarrow{v}_{out}\right| = \sqrt{u^2 + v^2} \tag{12}$$

where u and v are obtained by (1).

It is known from (10) and (11) that the energy consumption is proportional to $|\overrightarrow{v}_{usv}|$ and $1/|\overrightarrow{v}_{out}|$, when the thrust τ_{thr} generated by the propulsion system is a fixed value. To reduce the energy consumption, it is necessary to adjust the USV's heading to take advantage of the environmental loads to increase $|\overrightarrow{v}_{out}|$.

2.4.3. Path Smoothness. It is assumed that the current waypoint of the USV is $p_i = (x_i, y_i)$, the previous waypoint is $p_{i-1} = (x_{i-1}, y_{i-1})$, and the next waypoint is $p_{i+1} = (x_{i+1}, y_{i+1})$. Thus, the angle θ_{i+1} of the vector $\overrightarrow{p_i p_{i+1}}$ and the angle θ_i of the vector $\overrightarrow{p_{i-1} p_i}$ are

$$\theta_{i+1} = \arctan\left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i}\right) \tag{13}$$

$$\theta_i = \arctan\left(\frac{y_i - y_{i-1}}{x_i - x_{i-1}}\right) \tag{14}$$

The difference ψ_i between θ_{i+1} and θ_i is

$$\psi_i = abs\left(\theta_{i+1} - \theta_i\right) \tag{15}$$

θ_i, θ_{i+1}, and ψ_i, are shown in Figure 2. Therefore, the cost function of the path smoothness J_{smooth} is

$$J_{smooth} = \sum_{i=1}^{N} \psi_i \tag{16}$$

where N is the number of differences ψ_i.

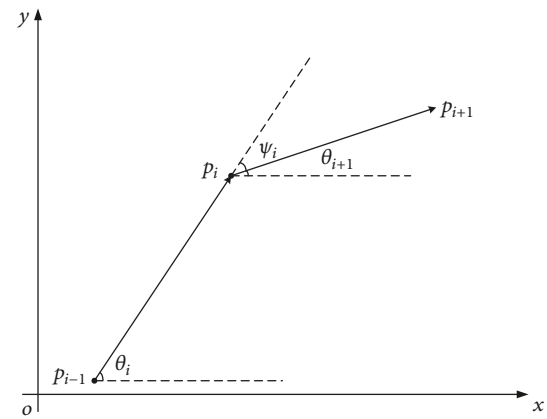

Figure 2: The angles of the path segments.

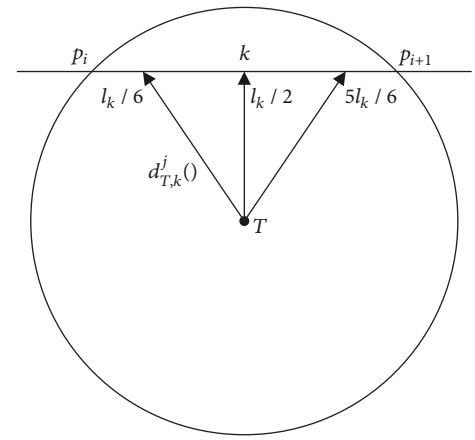

Figure 3: Schematic diagram of the calculation of the path safety.

2.4.4. Path Safety. Using the safety cost of the nodes on the grids cannot accurately represent the threat impact of each path segment. First, three sampling points are selected on a path segment and the average Euclidean distance between the three sampling points and the center of the obstacle is calculated. The schematic diagram of the calculation of the path safety is shown in Figure 3. l_k is the length of the kth path segment between the waypoint p_i and p_{i+1}. For the kth path segment, three sampling points are taken at $l_k/6$, $l_k/2$, and $5l_k/6$, respectively. The average Euclidean distance between the three sampling points and the center of the obstacle is

$$D_{T,k} = \frac{1}{3}\left[d^j_{T,k}\left(\frac{l_k}{6}\right) + d^j_{T,k}\left(\frac{l_k}{2}\right) + d^j_{T,k}\left(\frac{5l_k}{6}\right)\right] \tag{17}$$

where $d^j_{T,k}()$ is the Euclidean distance from the sampling point on the kth path segment to the center of the obstacle T_j.

The path safety cost between waypoint p_i and p_{i+1} denoted as $J_{safe}^{i,i+1}$ is

$$J_{safe}^{i,i+1} = \begin{cases} 0, & d > d_{safe\_\max} \\ \dfrac{1}{D_{T,k}^{i,i+1}}, & d_{safe\_\min} \le d \le d_{safe\_\max} \\ 1, & d < d_{safe\_\min} \end{cases} \tag{18}$$

where d is the distance between the USV and the obstacle's center, $d_{safe\_\max}$ is the radius of the obstacle's affected area, and $d_{safe\_\min}$ is the radius of the no-sail zone. $J_{T,k}^{i,i+1}$ is obtained using (17).

Thus, the entire path safety cost function J_{safe} is

$$J_{safe} = \sum_{i=1}^{N} J_{safe}^{i,i+1} \tag{19}$$

where N is the number of the waypoints of the planned path.

2.4.5. Cost Function. In summary, the cost function of the USV global path planning was established as

$$\min J = w_1 \cdot L + w_2 \cdot E + w_3 \cdot J_{smooth} + w_4 \cdot J_{safe} \tag{20}$$

where L, E, J_{smooth}, and J_{safe} are obtained by (8), (9), (16), and (19), respectively. w_1, w_2, w_3, and w_4 represent the weights of the path length, energy consumption, path smoothness, and path safety in the cost function, respectively, subject to

$$\begin{aligned} & L \le L_{\max} \\ & 0 \le \left|\overrightarrow{v}_{out}\right| \le \left|v_{\max}\right| \end{aligned} \tag{21}$$

where $L_{\max}$ is the maximum voyage distance of the USV and $|v_{\max}|$ is the maximum speed of the USV.

3. Optimization Algorithm

In this section, we will introduce the optimization algorithm for the USV global path planning—the IQACA. The IQACA is a new optimization algorithm that combines quantum-inspired computing with ant colony optimization algorithm. We will introduce quantum code and a quantum rotation gate from quantum-inspired computing. Some rules based on the ant colony optimization algorithm are presented.

3.1. Quantum Code. The quantum bit (Q-bit) is the basic unit in quantum computing. A Q-bit is a system that has two possible states $|0\rangle$ and $|1\rangle$. The state of a Q-bit $|\varphi\rangle$ is expressed as

$$\left|\varphi\right\rangle = \alpha\left|0\right\rangle + \beta\left|1\right\rangle \tag{22}$$

where α and β are the probability amplitudes, which satisfy $|\alpha|^2 + |\beta|^2 = 1$. $|\alpha|^2$ and $|\beta|^2$ are the probabilities in states $|0\rangle$ and $|1\rangle$, respectively. Thus, the state of the Q-bit $|\varphi\rangle$ is an uncertain superposition state between $|0\rangle$ and $|1\rangle$. When the number of Q-bits of an individual X_i is n, X_i is expressed as

$$X_i = \left[\begin{array}{c|c|c|c} \alpha_{i1} & \alpha_{i2} & \cdots & \alpha_{in} \\ \beta_{i1} & \beta_{i2} & \cdots & \beta_{in} \end{array}\right] \tag{23}$$

where $X_{ic} = (\alpha_{i1}, \cdots, \alpha_{in})$ and $X_{is} = (\beta_{i1}, \cdots, \beta_{in})$ are the two sets of solutions for individual X_i. Therefore, after quantum coding, every individual has two sets of solutions and the search space is doubled.

In the IQACA, the quantum pheromone is obtained by encoding the pheromone left by the ants on the path in the ACA by the Q-bits. The transfer direction of the ants is selected by the quantum pheromone concentration on the path. Thus, the quantum pheromone concentration value τ_{ij}^t of the ith ant on the jth point in the tth iteration is expressed as

$$\tau_{ij}^t = \begin{bmatrix} \alpha_{ij}^t \\ \beta_{ij}^t \end{bmatrix} \tag{24}$$

3.2. Adaptive Quantum Rotation Gate. In the quantum optimization algorithm, a quantum rotation gate is used to update the Q-bits. The update rule of a Q-bit is as follows:

$$\begin{bmatrix} \alpha_{ij}^{t+1} \\ \beta_{ij}^{t+1} \end{bmatrix} = U\left(\theta_t\right) \begin{bmatrix} \alpha_{ij}^t \\ \beta_{ij}^t \end{bmatrix} \tag{25}$$

where $[\alpha_{ij}^t, \beta_{ij}^t]^T$ represents the probability amplitude of the Q-bits in the tth iteration. $U(\theta_t)$ is the quantum rotation gate in the tth iteration

$$U\left(\theta_t\right) = \begin{bmatrix} \cos\theta_t & -\sin\theta_t \\ \sin\theta_t & \cos\theta_t \end{bmatrix} \tag{26}$$

where θ_t is the rotation angle in the tth iteration. In a previous paper [13], the rotation angle was obtained by looking it up in a table. In another paper [26], the local and global updates of the pheromone concentration increments in the ACA were added to the rotation angle step function. In the IQACA, an adaptive adjustment strategy for the rotation angle is obtained by comparing the current solution and the global optimal solution currently being searched. Thus, the rotation angle θ_t in the tth iteration is

$$\theta_t = -\operatorname{sgn}\left(A_i\right) \cdot \Delta\theta_i \tag{27}$$

where $-\operatorname{sgn}(A_i)$ is the direction of the rotation angle and $\Delta\theta_i$ is the size of the rotation angle. A_i is

$$A_i = \begin{vmatrix} \alpha_0 & \alpha_1 \\ \beta_0 & \beta_1 \end{vmatrix} \tag{28}$$

where α_0 and β_0 are the probability amplitudes of the quantum pheromone corresponding to the global optimal solution currently searched and α_1 and β_1 are the probability

amplitudes of the quantum pheromone corresponding to the current solution. $\Delta\theta_i$ is

$$\Delta\theta_i = J_e - \frac{J_e - J_k}{N_{\max}} \cdot t \tag{29}$$

where J_k is the cost value of ant k in the current solution, J_e is the cost value of the global optimal solution currently searched, and $N_{\max}$ is the maximum number of iterations.

3.3. Transfer Rule and Transition Probability. The ant colony optimization algorithm is a bionic intelligent algorithm inspired by the foraging behavior of ant colonies [27]. During the foraging, ants produce a substance called a pheromone. The concentration of the pheromone, which is related to the path length, will determine the movement of other ants. If the path is shorter, the concentration of the pheromone left on the path is larger.

To achieve multiobjective path planning, multiple pieces of heuristic information are used to determine the ant's transfer rules and transition probabilities. The transfer rule of ant k from point i to point j is

$$s = \begin{cases} \arg\max\limits_{s \in S} \left\{ \left[\tau_{ij}^k(t)\right]^\alpha \left[\eta_{ij}^k(t)\right]^\beta \left[\varepsilon_{ij}(t)\right]^\gamma \right\} & q \le q_0 \\ \widetilde{s} & q > q_0 \end{cases} \tag{30}$$

where q is a random number in the range $[0, 1]$. q_0 is a constant within $[0, 1]$. S is the set of points that ant k may reach by point i. $\widetilde{s}$ is the target waypoint selected by the following equation:

$$p_{ij}^k(t) = \frac{\left[\tau_{ij}^k(t)\right]^\alpha \left[\eta_{ij}^k(t)\right]^\beta \left[\mu_{ij}^k(t)\right]^\gamma}{\sum_{s \in allowed(i)} \left[\tau_{ij}^k(t)\right]^\alpha \left[\eta_{ij}^k(t)\right]^\beta \left[\mu_{ij}^k(t)\right]^\gamma} \tag{31}$$

where $\tau_{ij}^k(t)$ is the pheromone on the path from point i to point j in the tth iteration and α ($\alpha > 0$) is the pheromone index. $\eta_{ij}^k(t)$ is the multiple inspiration information on the path from point i to point j in the tth iteration, β ($\beta > 0$) is the index of multiple inspiration information, $\mu_{ij}^k(t)$ is the quantum information strength on the path from point i to point j in the tth iteration, which is expressed as $\mu_{ij}^k(t) = 1/|\alpha_{ij}^k(t)|^2$, and γ ($\gamma > 0$) is the index of the quantum information strength.

The multiple pieces of heuristic information include the path length heuristic information $\phi_{ij}^k(t)$, energy consumption heuristic information $\varepsilon_{ij}^k(t)$, path smoothness heuristic information $\varphi_{ij}^k(t)$, and path safety heuristic information $\zeta_{ij}^k(t)$.

$$\left[\eta_{ij}^k(t)\right]^\beta = \left[\phi_{ij}^k(t)\right]^a \left[\varepsilon_{ij}^k(t)\right]^b \left[\varphi_{ij}^k(t)\right]^c \left[\zeta_{ij}^k(t)\right]^d \tag{32}$$

$$\begin{aligned} \phi_{ij}(t) &= \frac{1}{L_{ij}} \\ \varepsilon_{ij}(t) &= \frac{1}{E_{ij}} \\ \varphi_{ij}(t) &= \frac{1}{J_{smooth}^{ij}} \\ \zeta_{ij}(t) &= \frac{1}{J_{safe}^{ij}} \end{aligned} \tag{33}$$

where L_{ij}, E_{ij}, J_{smooth}^{ij}, and J_{safe}^{ij} are obtained by (8), (9), (16), and (19), respectively. a, b, c, and d are the indices of the path length heuristic information, energy consumption heuristic information, path smoothness heuristic information, and path safety heuristic information, respectively.

3.4. Update Rules of Pheromone. After every ant completes a one-transfer, the pheromone on the path it passes is locally updated to avoid falling into a local optimum. When the current point of the ant is p_i and the next point is p_j, the pheromone local updating rule is

$$\tau(p_j) = (1 - \rho_1) \cdot \tau(p_i) + \rho_1 \cdot \Delta\tau_{ij} \tag{34}$$

where $\tau(p_i)$ is the pheromone of the current point, $\tau(p_j)$ is the pheromone of the next point, ρ_1 ($0 < \rho_1 < 1$) is the pheromone local updating coefficient, and $\Delta\tau_{ij}$ is the pheromone that every ant leaves on the path from p_i to p_j in this iteration, expressed as follows:

$$\Delta\tau_{ij} = \sum_{k=1}^{n} \Delta\tau_{ij}^k \tag{35}$$

$$\Delta\tau_{ij}^k = \begin{cases} \dfrac{Q}{J_k}, & \text{the path segment of the } k\text{th ant} \\ 0, & \text{else} \end{cases} \tag{36}$$

where Q is a constant and J_k is the cost value of the kth ant's path.

After all the ants complete an iteration, the pheromone is globally updated to increase the pheromone concentration on the optimized path. The rules are as follows:

$$\tau(p_j) = \begin{cases} (1 - \rho_2) \cdot \tau(p_j) + \rho_2 \cdot \Delta\tau_{best}, & p_j = \widetilde{s} \\ (1 - \rho_2) \cdot \tau(p_j), & p_j \ne \widetilde{s} \end{cases} \tag{37}$$

$$\Delta\tau_{best} = \begin{cases} \dfrac{Q}{J_e}, & \text{if } (i, j) \text{ belongs to the optimal path in this cycle} \\ 0, & \text{else} \end{cases} \tag{38}$$

where Q is a constant, J_e is the cost value of the optimal path in this iteration, ρ_2 ($0 < \rho_2 < 1$) is the pheromone global updating coefficient, and $\widetilde{s}$ is the global optimal solution currently being searched.

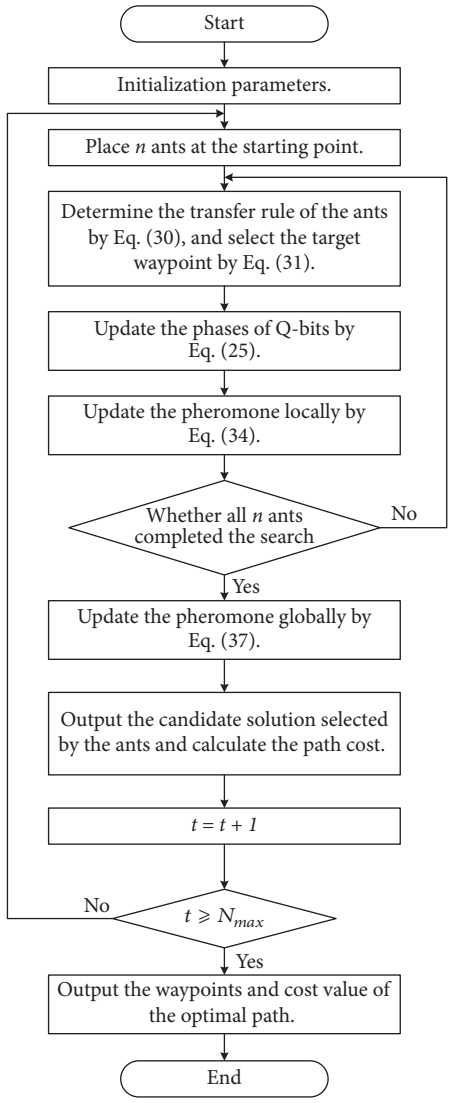

Figure 4: The flowchart of the global path planning algorithm based on the IQACA.

3.5. Global Path Planning Algorithm Based on IQACA. The flowchart of global path planning algorithm based on the IQACA is shown in Figure 4.

The main steps are as follows:

Step 1 (initialize the parameters). The number of the ants in the colony is n. The maximum number of iterations is $N_{\max}$. The initial quantum pheromone concentration value of the ith ant on the jth waypoint is expressed as $\tau_{ij}^t = [\alpha_{ij}^t, \beta_{ij}^t]^T = [1/\sqrt{2}, 1/\sqrt{2}]^T$, where $t = 0$;

Step 2. The ants are placed at the starting point. The transfer rule of the ants is determined by (30), and the target waypoint is selected by (31).

Step 3. The phases of the Q-bits are updated by (25).

Table 1: The obtained values from the simulations.

Algorithm	Length	Iteration
ACA	8447	167
QACA	8062	142
IQACA	7891	126

Step 4. The pheromone is locally updated by (34).

Step 5. After all the ants have passed by all the points in an iteration, the pheromone is globally updated by (37).

Step 6. The candidate solution selected by the ants is output and the path cost is calculated.

Step 7. If the iteration $t > N_{\max}$, the algorithm moves to Step 8; otherwise, it returns to Step 2.

Step 8. The waypoints of the optimized solution and the cost value of the path are output. The global optimized path is obtained by the waypoints of the optimized solution.

Step 9. The algorithm ends.

4. Simulation Studies

In this section, the effectiveness and efficiency of the IQACA are validated. The section consists of two parts. The first subsection compares the performance of the ACA, QACA, and IQACA with the Traveling Salesman Problem (TSP). The second subsection deals with the USV global path planning based on the IQACA. To validate the proposed algorithm, simulations were conducted.

4.1. Performance Evaluation of IQACA. To validate the effectiveness of the IQACA presented in this paper, we compared the algorithm performance between ACA, QACA, and IQACA with the TSP. In this paper, RAND100 was selected from the TSPLIB standard library for the simulations. The maximum number of iterations $N_{\max} = 200$, the number of the ants $n = 100$, $\alpha = 3$, $\beta = 1$, $\gamma = 2$, and $\rho_1 = \rho_2 = 0.8$. The obtained values are shown in Table 1, and the iterations are shown in Figure 5.

The known optimal value of RAND100 is 7891. From Table 1, it was concluded that the path length of RAND100 obtained by the IQACA was 2.12% lower than that obtained by the QACA and 6.58% lower than that obtained by the ACA. The number of iterations required for the IQACA to converge to the minimum was 11.27% lower than the QACA and 24.55% lower than the ACA. The results show that the IQACA was superior to the QACA and the ACA in both the path length and iteration number. Since the algorithm uses the pheromone local and global updates, and the phases of the Q-bits are updated by the adaptive quantum rotation gate, the IQACA can avoid the local optimal solution. Because the pheromone is encoded by Q-bits, the search space is doubled, and the convergence speed is faster. Thus, the IQACA is an effective and efficient algorithm.

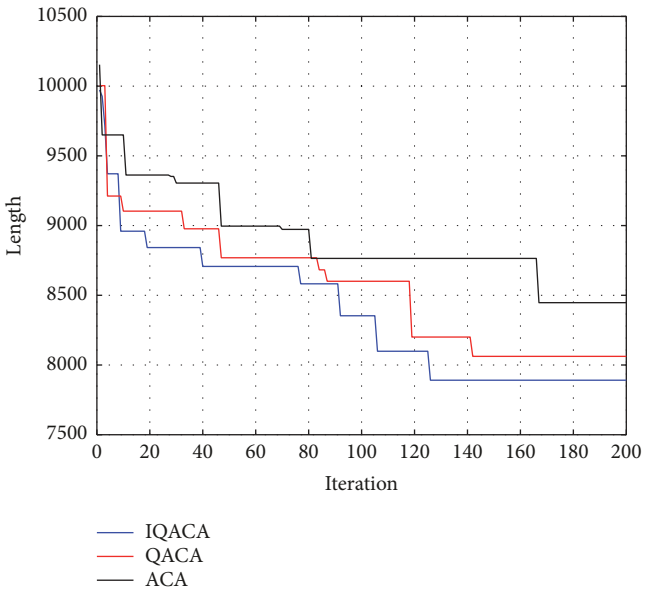

Figure 5: Convergence of the IQACA, QACA, and ACA.

Table 2: Coefficients of USV.

Property	Value
Length overall (m)	80.8
Breadth (m)	18.2
Draught (m)	5.0
Area of frontal projection above the waterline (m^2)	330.9
Area of lateral projection above the waterline (m^2)	874.8
Area of frontal projection below the waterline (m^2)	91.0
Area of lateral projection below the waterline (m^2)	323.4

4.2. USV Global Path Planning Based on IQACA. In this subsection, we will show a simulation of USV global path planning based on the proposed algorithm. Obstacles are black-colored. The coordinates of the starting point are $(0.5, 24.5)$, and the coordinates of the destination point are $(29.5, 0.5)$. The length of the side of a grid is 1 km. We assumed that the wind, waves, and ocean current act on the vessel from the same direction, since in most cases the ocean current is the most significant environmental disturbance on the vessel. In this simulation, the direction of the disturbances is assumed as 240° in the Northeast coordinate system. The relative wind speed was 7.5 m/s. The wave height was 2.5 m. The relative ocean current was 2.0 m/s. The thrust of the USV propulsion system was $500kN$. The coefficients of the USV in this simulation are shown in Table 2. The maximum number of iterations was 500, the number of the ants was 100, $\alpha = 3$, $\gamma = 2$, and $\rho_1 = \rho_2 = 0.8$. The safety boundary of the obstacle was represented by a dotted red circle whose radius was 1.3 times the distance between the vertex and the center of the obstacle. The optimal path is represented by a solid blue line. The objectives of the global path planning were determined based on their respective weights. We considered four scenarios in this simulation. In the first scenario, we highly weighted the path length, so that the weights $[w_1, w_2, w_3, w_4]$ in the cost function (20) were $[0.5, 0.1, 0.3, 0.1]$. In the other three scenario, we weighted the energy consumption, path smoothness, and path safety, respectively, such that the weights $[w_1, w_2, w_3, w_4]$ were $[0.1, 0.5, 0.1, 0.3]$, $[0.3, 0.1, 0.5, 0.1]$, and $[0.1, 0.3, 0.1, 0.5]$, respectively.

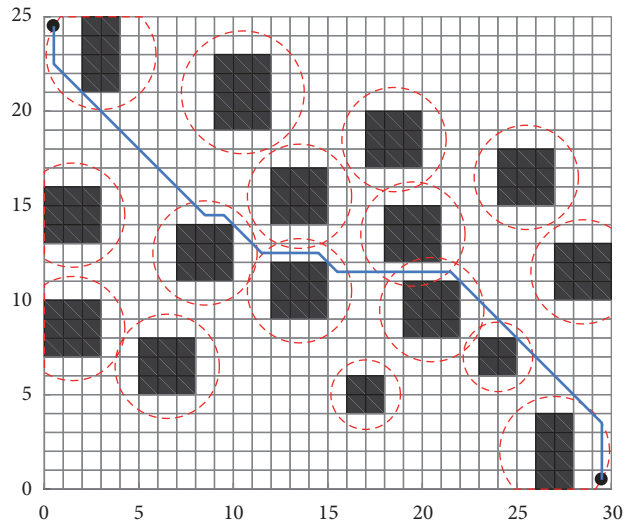

Figure 6: Planned path focusing on the path length.

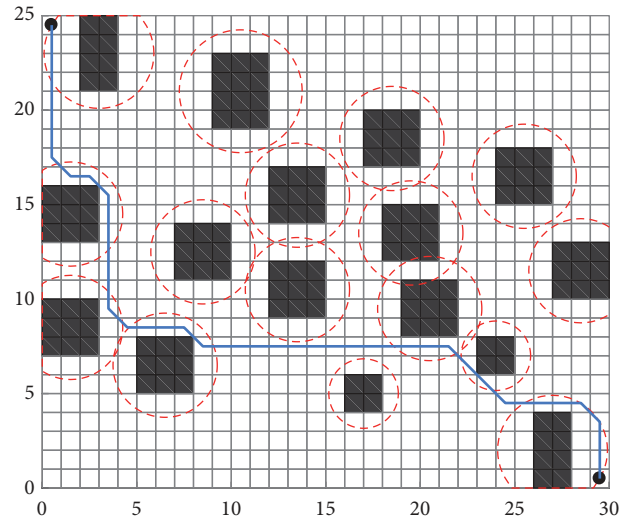

Figure 7: Planned path focusing on the energy consumption.

Figures 6–9 show the planned paths in the four scenarios. The path length, energy consumption, path smoothness, and path safety index for each planned path are listed in Table 3. The results indicate that the proposed algorithm can plan feasible paths for the USV considering different objectives simultaneously. Moreover, by adjusting the weights of different objectives, the proposed algorithm can generate paths for different purposes.

Table 3: Path data for different objectives.

Objective	Path length (km)	Energy consumption (1×10^7 kJ)	Path smoothness (rad)	Path safety index
Path length	**41.87**	12.38	18.85	26.89
Energy consumption	48.31	**10.81**	28.27	27.12
Path smoothness	43.63	11.95	**14.14**	15.51
Path safety	46.80	11.96	28.27	**7.61**

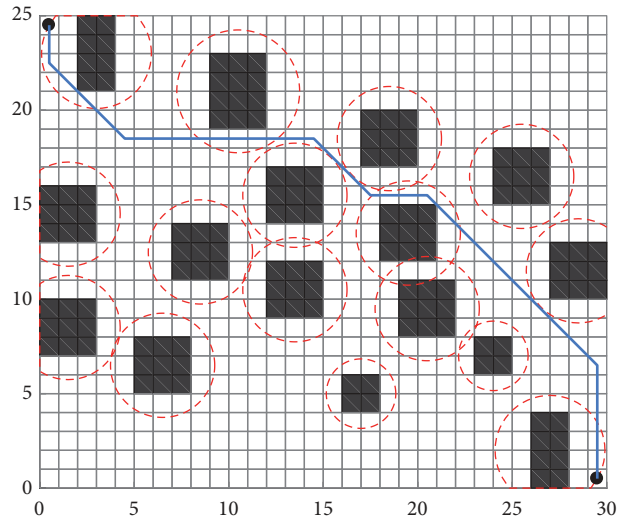

Figure 8: Planned path focusing on the path smoothness.

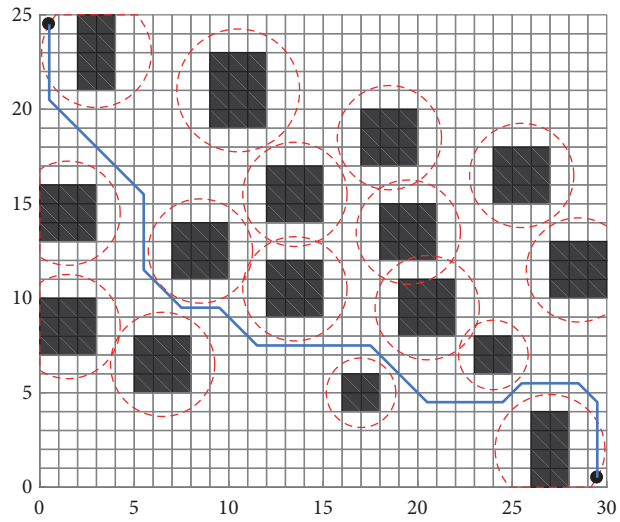

Figure 9: Planned path focusing on the path safety.

5. Conclusion

This paper proposed a global path planning algorithm for the USV based on an improved quantum ant colony algorithm (IQACA).

The IQACA is an optimization algorithm that combines quantum computing with the ACA. In IQACA, using Q-bits to encode the pheromone of the ants, the search space is doubled when the number of the ants is the same. The simulation results show that the proposed algorithm's obtained minimum was 2.1–6.5% lower than those of the quantum ant colony algorithm (QACA) and ant colony algorithm (ACA), and the number of iterations required to converge to the minimum was 11.2–24.5% lower than those of the QACA and ACA. Based on the model of the kinetics of the USV and the marine environment, we defined the objectives of the path planning: the path length, energy consumption, path smoothness, and path safety. The simulation results showed that the proposed algorithm can consider several optimization objectives and generate paths satisfying these requirements.

In the future, the following studies should be conducted in depth. First, the correlation between the multiple objectives should be calculated to determine the weight of each objective in the cost function to meet the actual mission requirements. Moreover, the kinetic and kinematic constraints of the USV should be added to the cost function. Finally, more practical environmental loads should be applied to calculate their effects on the path energy consumption of the USV.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities (HEUCFJ180404) and the 7th Generation Ultra Deep Water Drilling Platform (Ship) Innovation Project.

References

[1] H. Kim, S.-H. Kim, M. Jeon, J. Kim, S. Song, and K.-J. Paik, "A study on path optimization method of an unmanned surface vehicle under environmental loads using genetic algorithm," *Ocean Engineering*, vol. 142, pp. 616–624, 2017.

[2] K. Chu, M. Lee, and M. Sunwoo, "Local path planning for off-road autonomous driving with avoidance of static obstacles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1599–1616, 2012.

[3] C.-C. Sun, G. E. Jan, S.-W. Leu, K.-C. Yang, and Y.-C. Chen, "Near-shortest path planning on a quadratic surface with O(n log n) time," *IEEE Sensors Journal*, vol. 15, no. 11, pp. 6079-6080, 2015.

[4] H. Niu, Y. Lu, A. Savvaris, and A. Tsourdos, "An energy-efficient path planning algorithm for unmanned surface vehicles," *Ocean Engineering*, vol. 161, pp. 308–321, 2018.

[5] T. Lee, H. Chung, and H. Myung, "Multi-resolution path planning for marine surface vehicle considering environmental effects," in *Proceedings of the OCEANS 2011 IEEE - Spain*, Spain, June 2011.

[6] F. Ahmed and D. Kalyanmoy, "Multi-objective optimal path planning using elitist non-dominated sorting genetic algorithms," *Soft Computing*, vol. 17, no. 7, pp. 1283–1299, 2013.

[7] Y. Ma, M. Hu, and X. Yan, "Multi-objective path planning for unmanned surface vehicle with currents effects," *ISA Transactions®*, vol. 75, pp. 137–156, 2018.

[8] S. Shao, W. Guan, B. Ran, Z. He, and J. Bi, "Electric vehicle routing problem with charging time and variable travel time," *Mathematical Problems in Engineering*, vol. 2017, Article ID 5098183, 13 pages, 2017.

[9] Y. Liu, X. Zhang, X. Guan, and D. Delahaye, "Potential odor intensity grid based uav path planning algorithm with particle swarm optimization approach," *Mathematical Problems in Engineering*, vol. 2016, Article ID 7802798, 16 pages, 2016.

[10] G. Xia, C. Liu, and X. Chen, "Multi-objective optimization for AUV conceptual design based on NSGA-II," in *Proceedings of the OCEANS 2016*, pp. 1–6, IEEE, Shanghai, China, 2016.

[11] J. Zhao, D. Cheng, and C. Hao, "An improved ant colony algorithm for solving the path planning problem of the omnidirectional mobile vehicle," *Mathematical Problems in Engineering*, vol. 2016, Article ID 7672839, 10 pages, 2016.

[12] A. Narayanan and M. Moore, "Quantum-inspired genetic algorithms," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '96)*, pp. 61–66, Nagoya, Japan, May 1996.

[13] K. H. Han and J. H. Kim, "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 6, pp. 580–593, 2002.

[14] H. Xing, Y. Ji, L. Bai, X. Liu, Z. Qu, and X. Wang, "An adaptive-evolution-based quantum-inspired evolutionary algorithm for QoS multicasting in IP/DWDM networks," *Computer Communications*, vol. 32, no. 6, pp. 1086–1094, 2009.

[15] L. Cui, L. Wang, J. Deng, and J. Zhang, "A new improved quantum evolution algorithm with local search procedure for capacitated vehicle routing problem," *Mathematical Problems in Engineering*, vol. 2013, Article ID 159495, 17 pages, 2013.

[16] P.-C. Li and S.-Y. Li, "Quantum ant colony algorithm for continuous space optimization," *Control Theory and Applications*, vol. 25, no. 2, pp. 237–241, 2008.

[17] X.-M. You, S. Liu, and Y.-M. Wang, "Quantum dynamic mechanism-based parallel ant colony optimization algorithm," *International Journal of Computational Intelligence Systems*, vol. 3, pp. 101–113, 2010.

[18] M. Liu, F. Zhang, Y. Ma, H. R. Pota, and W. Shen, "Evacuation path optimization based on quantum ant colony algorithm," *Advanced Engineering Informatics*, vol. 30, no. 3, pp. 259–267, 2016.

[19] Q. Yong, C. Binbin, and X. YanFang, "A novel quantum ant colony algorithm used for campus path," in *Proceedings of the 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, vol. 2, IEEE, 2017.

[20] T. I. Fossen, *Guidance and Control of Ocean Vehicles*, vol. 199, Wiley, New York, NY, USA, 1994.

[21] R. M. Isherwood, "Wind resistance of merchant ships," *The Royal Institution of Naval Architects*, vol. 115, pp. 327–338, 1972.

[22] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*, John Wiley & Sons, 2011.

[23] H. Han-Pang and C. Shu-Yun, "Dynamic visibility graph for path planning," in *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, pp. 2813–2818, IEEE, Sendai, Japan, 2004.

[24] M. Candeloro, A. M. Lekkas, and A. J. Sørensen, "A Voronoi-diagram-based dynamic path-planning system for underactuated marine vessels," *Control Engineering Practice*, vol. 61, pp. 41–54, 2017.

[25] H. Kim, D. Kim, J.-U. Shin, H. Kim, and H. Myung, "Angular rate-constrained path planning algorithm for unmanned surface vehicles," *Ocean Engineering*, vol. 84, pp. 37–44, 2014.

[26] Jia. Yang et al., "A novel quantum ant colony optimizing algorithm," *Acta Scientiarum Naturalium Universitatis Sunyatseni*, vol. 48, no. 3, pp. 22–27, 2009.

[27] M. Dorigo and G. Di Caro, "Ant colony optimization: a new meta-heuristic," in *Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406)*, vol. 2, pp. 1470–1477, IEEE, July 1999.