

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot



Uncertain moving obstacles avoiding method in 3D arbitrary path planning for a spherical underwater robot



Ruochen An^a, Shuxiang Guo^{b,c,*}, Liang Zheng^a, Hideyuki Hirata^c, Shuoxin Gu^d

^a Graduate School of Engineering, Kagawa University, Takamatsu, Kagawa 760-0396, Japan

^b Key Laboratory of Convergence Medical Engineering System and Healthcare Technology, The Ministry of Industry and Information Technology,

School of Life Science and Technology, Beijing Institute of Technology, Beijing, China

^c Department of Intelligent Mechanical Systems Engineering, Kagawa University, Takamatsu, Kagawa 760-0396, Japan

^d School of Control Engineering, Chengdu University of Information Technology, Chengdu 610225, China

ARTICLE INFO

Article history: Received 1 December 2020 Received in revised form 8 November 2021 Accepted 31 December 2021 Available online 19 January 2022

Keywords:

Spherical underwater robot Uncertain dynamic obstacles avoiding 3D arbitrary path planning Inertial measurement unit and depth sensor localization

ABSTRACT

In order to avoid the risk of obstacles collision during the spherical underwater robot (SUR) move to target points in 3D arbitrary path planning, an underwater obstacle avoiding method was studied. Considering the uncertainty of the movement of obstacles in the actual environment, we present an uncertain moving obstacle avoiding method based on the improved velocity obstacle method. In addition, to reduce the distance and time of obstacle avoidance, the concept of the time of obstacle avoiding was designed. First, the size and velocity information of obstacles are obtained through the camera, which can provide an accurate decision basis for obstacles avoidance in the next step. Then, according to the time when the robot collides with the obstacle, the time of start and end of the obstacle avoidance is determined. The movement direction and velocity of the robot are obtained based on the improved velocity obstacle method and the movement characteristics of SUR. Finally, a detailed 3D arbitrary path planning analysis based on an improved ant colony algorithm was conducted. A series of experiments were carried out in the pool that validates the proposed methods are also presented. **(E)** 2022 Elsevier B.V. All rights reserved.

1. Introduction

Due to the underwater environment's dangers and limited human diving depth, underwater robots are increasingly becoming an essential tool for developing the ocean environment [1]. In the ocean environment that has a large number of obstacles, apart from static obstacles, underwater robots also face the threat of moving obstacles such as underwater floating objects and other underwater vehicles. Uncertain moving obstacles avoiding method in 3D arbitrary path planning that improves one aspect of the autonomous vehicle's performance is a fundamental requirement for the underwater robots [2,3].

Over the past few years, many studies have been dedicated to the uncertain moving obstacles avoiding method in 3D arbitrary path planning for the underwater robots. The information of static obstacles is easy to predict, but most of them are sudden moving obstacles in the actual ocean, and the information of obstacles is difficult to predict. The common collision avoidance methods include the artificial aperture method (APF) [4,5], dynamic window method (DWA) [6], and behavior method [7]. In addition, in [8], the authors improved the artificial potential field method that the

https://doi.org/10.1016/j.robot.2021.104011 0921-8890/© 2022 Elsevier B.V. All rights reserved. weighting matrix accepts obstacle avoidance terms to enhance the detection performance. However, this control system is not suitable for AUV due to the difference in motor control accuracy. A new nonlinear controller using the sliding mode approach for the AUVs in [9]. The sliding mode is commonly utilized in underwater adaptive strategy. However, the "chattering" effect is the main drawback associated with the sliding mode. The greater the switching amplitude of the controlled amount, the more obvious the jitter will be. A dynamic bioinspired neural network is proposed [10], a target attractor is introduced into the neuron activity updating equation to improve the computing efficiency. However, the inability to use any prior knowledge results in a decrease in overall efficiency. Most of the abovementioned research on obstacle avoidance methods has achieved good results, but these methods have low accuracy for avoiding moving obstacles. To solve the problem of the poor accuracy of AUV avoiding obstacles in the actual environment in the existing methods, the velocity obstacle method is ideal [11-13]. The velocity obstacle method can realize the active obstacle avoidance of the AUV during the movement process by transforming the uncertain movement of the moving obstacle into an uncertain position.

Path planning is a necessary part of the underwater robot control system. It not only determines the ability of autonomy of

^{*} Corresponding author. E-mail address: s19d501@stu.kagawa-u.ac.jp (S. Guo).

underwater robots and the premise of the reliability of a mission and the likelihood of success. Autonomous path-planning has the capability of planning a trajectory that safely leads the robot from the initial or current position to its destination. Path planning has various objectives according to the different underwater tasks, such as the total travel time, length, energy consumption, etc. Various solution approaches have been developed to determine the time- or length-optimal paths in the current fields. Autonomous path-planning methods for underwater robots are divided into global [14,15] and local path planning [16,17]. Cao et al. [18] have adopted the particle swarm optimization (PSO) algorithm for the trajectory optimization of AUV path searching functions. Simulation experiments show that this algorithm has higher efficiency and adaptability and has excellent robustness and a fast convergence speed. However, the convergence speed of the particle swarm optimization algorithm in the initial stage of the search is fast, and the convergence speed in the late stage of the investigation is slow. Yao et al. [19] proposed an improved Genetic Algorithm (GA) for path planning. The simulation shows that the convergence speed is faster. However, this method is challenging to apply in real environments with obstacles. Yan Ma et al. [20] worked on the path planning by improving the traditional Ant Colony Optimization (ACO) for underwater robots. The simulation results obtained from the proposed algorithm compared with the traditional ACO give improved results and can quickly find an optimal solution. ACO is a heuristic global optimization algorithm. Compared with other algorithms, the ACO has low requirements for selecting initial and is highly robust. The advantage of ACO is that it can apply it to the underwater 3D path planning problem [21]. However, the ant colony algorithm has the issue of a slow convergence rate and is easily trapped in a local optimum. Population-optimization algorithms have an essential and influential role in path planning in a complex dynamic environment. For this reason, population-optimization algorithms are suitable for the path planning of AUVs. However, the algorithm has widespread problems, such as poor real-time capability, slow processing speed, easily falling into a local optimum, and demanding applications in a real environment.

In our previous research, a spherical underwater robot was proposed and designed [22]. We have improved four-generation spherical underwater robots. Up to now, the fourth spherical underwater robot was achieved the 6 degrees of freedom (DoF) movement [23] and the movement of switching speed [24]. Our lab also developed an amphibious spherical robot [25-27]. Similarly, the amphibious spherical robot realized multi-degree-offreedom motion [28]. The positioning function of robots was evaluated [29], and communication function between robots was also realized [30,31]. Different tasks require different shapes and sizes of AUVs [32,33]. The core feature of spherical robots refers to more flexibility and can achieve multiple degrees of freedom motion compared with autonomous underwater vehicles (AUVs). Therefore, these spherical shapes robots have been receiving increasing interest from academia. Most researchers focus on prototype design [34] for spherical robots and drive system control methods [35]. However, few studies are conducted on closed-loop motion control (such as uncertain dyskinesia avoidance methods and path planning, etc.). At the present state of development of previous research, the functionalization of the SUR is necessary to create a fully self-contained, intelligent, and decision-making underwater robot. The propulsion model and shape of the bionic spherical robot are different from other AUVs. Its fluid dynamics are complex, making it difficult to realize the trajectory tracking of the spherical robot through experiments. This paper focuses on the achievement of uncertain moving obstacles avoided in 3D arbitrary path planning. Above all, a dynamic obstacle avoiding system based on improved velocity obstacle is proposed in this paper.



Fig. 1. Fourth-generation Spherical Underwater Robot (SUR IV).

This paper focuses on the 3D path planning with uncertain moving obstacles avoided for the SUR are studied. A controller that can improve the stability performance of the underwater robot under disturbances is designed based on the PID control. The stability performance of the underwater robot provided the premise of 3D path planning. In order to ensure safety during the 3D path planning of the underwater robot, an obstacle avoiding method based on the improved velocity obstacle method was presented and considered the movement information of dynamic obstacles to ensure the obstacle avoidance of dynamic obstacles in the planned path. The time of obstacle avoiding of SUR will be introduced and can achieve shorter distance and time obstacle avoidance. It makes the obstacle avoidance of the robot more efficient. The 3D arbitrary path planning is based on an improved ant colony algorithm that is optimized based on the ant colony algorithm, uses the elite ant strategy, and changes the weight of the volatility to reduce the number of iterations of the algorithm. Also, the ant colony algorithm and the particle swarm optimization algorithm are merged to solve that the ant colony algorithm is easy to converge in the local solution. The improved ACO algorithm is used for global path planning can achieve the shortest path in a short time for the underwater robot. Simulation and experimental evaluations of the method proposed in this paper are conducted.

The rest of this paper is organized as follows. The mechanical design, control system modeling and stability unit are introduced in Section 2. In Section 3 reports the principle of dynamic obstacles avoiding and 3D path planning for SUR IV. Stability experiments, dynamic obstacles avoiding, and 3D arbitrary path planning experiments are given in Section 4. Finally, conclusion is summarized in Section 5.

2. Design of the spherical underwater robot

2.1. Mechanical design

The 3D model of the spherical underwater robot that was developed for operation in an underwater environment was shown in Fig. 1. Considering the stability requirements of the underwater robot's uncertain moving obstacles avoiding task, a novel fairing device was designed, as illustrated in Fig. 1, based on the previous robot. The fairing is used to adjust the water flow and reduce



Fig. 2. The waterproof structure of the Open MV camera.



(a) Contours of static pressure.

(b) Velocity vector.

Fig. 3. DOFs of SUR IV in up motion.

the resistance of the underwater robot in the water. This device adopts three parts: a fixed frame, fairing ring, and spiry weight. The streamlined shape can reduce water-flow resistance and movement stability when the robot is in the water. Meanwhile, the fixed frame can reduce the inclusion of impurities (gravel, etc.) in the thruster system and extend its service life. The fairing system was 3D-printed, which effectively reduces the weight of the fairing. The fairing is made by 3D printing using polylactic acid (PLA) material and its total weight is 108 g. The weight of the robot is 7.9 kg, so the weight of the fairing is negligible.

Furthermore, an Open MV4 camera which can capture RGB565 images with 640 \times 480 resolution at 25 fps–50 fps is installed on the SUR IV, which is used for perceiving the obstacles information. The camera includes STM32H743II ARM Cortex M7 processor, 480 MHz, 1MB RAM and 2MB flash. The height angle of view of the camera is 115°, and the horizontal angle of view is 90°. A customized waterproof hull as shown in Fig. 2, is made using 3D printing technology and mounted middle of the robot by the extended back cover. An optical glass is fixed in the front of the camera lens.

2.2. Hydrodynamic analysis

Hydrodynamic analysis is necessary to evaluate the SUR IV performance. To analyze the hydrodynamic characteristics of the SUR IV, some simulation analysis is carried out in ANSYS-FLUENT. To estimate the parameters of dynamic model of robot, we assumed that the robot was a static sphere, and the environment of flow field was 20 °C without external disturbances. Due to the grid and flow field of the spherical robot are two key factors for the effectiveness and computational complexity of fluid dynamic analysis. In the hydrodynamic analysis, the flow field used in this simulation is established based on the 3D model of the SUR IV. Hyper mesh and ANSYS fluent meshing were used to process the surface mesh and volume mesh of the model, respectively, the total number of meshes was 285410 and 13246083, the mesh



(a) Contours of static pressure.

(b) Velocity vector.

Fig. 4. DOFs of SUR IV in forward motion.



Fig. 5. Control system of SUR IV.

quality was 0.86773919, and the meshing results of the robot 3D model and flow field.

After performing the meshing operation and flow field of the SUR IV, import the meshing result file into ANSYS FLUENT. The inlet of the flow field is set as the velocity inlet, and the outlet of the flow field is set as the outlet. The inlet velocity of the forward motion and up and down motion is 0.15 m/s. The turbulence model is a k- ε model. Using this method, we determined that the wall of the spherical underwater robot is static, and the flow field moves at a certain speed. To simulate the motion of SUR IV underwater, we consider the robot moves relative to the flow field. The maximum relative velocity that the propulsive forces generated by the hybrid propeller and the velocity of the flow field was 0.15 m/s. SUR IV is designed to have six independent actuators for the surge, sway, heave, vaw, roll and pitch directions. In this part, we mainly consider two types of SUR IV motion are forward motion and up motion and we focus on the hydrodynamic analysis of the novel fairing device. After hydrodynamic analysis simulations, the pressure affected by the flow field and the upmotion velocity streamline as show in Fig. 3a, b, respectively. In Fig. 4a, b, was shown the forward motion of SUR IV. In particular, the effect of the novel fairing device was not obvious, and the water-flow trajectories are stable.

2.3. Control design

To improve the efficiency of the control system, we divided it into three parts: power supply, propulsion system control, and data collection. The control circuit is designed as shown in Fig. 5. The ATMega2560 is employed as the control center communicate with senor module to obtain feedbacks underwater of SUR IV's, so the servo motors and DC motors can adjust the robot's attitude in time for responding feedback. In order to further improve the robot's control accuracy and stability, the robot's control strategy was optimized. Since IMU is susceptible



Fig. 6. PID control overall flow chart.



Fig. 7. Virtual environment of motion. (a) 3D model of SUR IV. (b) The trajectory of SUR IV under interference.

to the effect of duration, we have corrected the IMU to improve the accuracy of it in previous research [36]. The stability motion control of the underwater robot is the premise of path planning. We used a proportional-integral-derivative (PID) controller that introduced in the previous research [23] to control the direction and magnitude of the thrusters and enhance their flexibility.

Controlled by a PID self-control system, SUR IV can resist external disturbances, the overall flow chart of PID control as shown in Fig. 6. The input quantity of PID control is based on the error between the feedback quantity and the desired value, and the data of attitude solution was calculated by 9-DOFs IMU. To demonstrate that this PID control is suitable to SUR IV, some simulations are conducted. We modeled a virtual prototype of the spherical underwater robot under the Automatic Dynamic Analysis of Mechanical Systems (ADAMS) environment. We established a 3D model of the robot in SolidWorks and then the model was imported into ADAMS. The model was simplified in ADAMS and set the material properties of the robot, such as weight and the center of gravity. In addition, some related constraints and forces must be applied to the virtual prototype model. Randomly generate a series of random forces using excel and import them into Adams as random interference forces to simulate the interference of water flow on the stability of the robot when the robot is moving underwater. The PID control model is established in the MATLAB environment, and the data exchange parameters are realized through the ADAMS/ Control module, and finally the joint simulation of MATLAB and ADAMS is realized. By applying PID control to the virtual prototype model, the moving trajectory of the centroid of the robot as shown in Fig. 7 was obtained and also demonstrate the velocity and stability of the robot. Through the post-processing function of ADAMS/ View to simulation other basic motion. Consequently, jumping, and sudden changes not appeared, and the robot can implement stable motion. At the same time, we verified the stability of the course and depth of the robot through experiments. The relevant experiments will be explained in detail in Chapter 4. Based on the stability control of SUR IV, precise path planning can be realized.

3. Principle of uncertain moving obstacles avoiding method in 3D arbitrary path planning

3.1. Analysis of uncertain dynamic avoiding

Combined with the camera and IMU of SUR, we will employ the target detection method presented in paper [37]. This method



Fig. 8. The velocity relationship between the SUR IV and obstacles.

can identify the obstacle properties in the SUR working environment and distinguish between static obstacles and dynamic obstacles. The division of ranging points is not enough to acquire the movement properties of the obstacles, so comparison of the difference between the adjacent time grid maps is necessary, and it can detect the movement of obstacles (movement or static). The velocity obstacle method defines a relative velocity obstacle region. When the relative velocity falls into this area, it is considered that there will be navigation conflict between the robot and the obstacle in the limited time. In order to solve the conflict, the relative velocity is released out of the conflict area along the shortest path. In Fig. 8, B is defined as SUR, and D is a dynamic obstacle. The SUR and obstacles are expanded into the round moving body, and their velocity are $V_{\rm B}$ and $V_{\rm D}$ respectively.

Definition 1. Relative collision cone (RCC), i.e. the set of relative velocities $V_{\rm R} = VB - V_{\rm Dn}$ for underwater vehicles.

$$RCC = \{V_R | l_{RO} \cap \odot D \neq \varnothing\}$$
⁽¹⁾

where: LRO is the straight line of relative velocity; \odot D is the safety protection zone of robot.

The model only considers the position relationship and the current state between the robot and the obstacle. Taking D as the reference point, robot makes relative motion. If the relative motion trajectory of D and B intersects, there will be motion conflict between robot and obstacle, otherwise there will be no motion conflict. BD is the line between the underwater robot and the dynamic obstacle. α is the angle between the line BD and the boundary of the relative collision cone. β is the angle between the



Fig. 9. The relationship between the SUR IV and obstacles when changing the depth level.

line BD and the relative speed. Therefore, the following judgment can be made on the motion conflict between the robot and the obstacle: when $\alpha > \beta$, there is motion conflict between the SUR and obstacles; when $\alpha \le \beta$, there is no motion conflict between the two one. The values of α and β can be calculated as following:

$$\sin \alpha = \frac{d_1}{\left| \overrightarrow{BD} \right|} \tag{2}$$

$$\cos \beta = \cos \left(\angle \left(v_R, \overrightarrow{BD} \right) \right) = \frac{v_R \cdot BD}{|v_R| \left| \overrightarrow{BD} \right|}$$
(3)

where d_1 is the radius of the relative collision cone. The robot and dynamic obstacle are in the same depth level. If robot wants to surpass dynamic obstacle, it can adopt the strategy of changing the depth level to solve the motion conflict, as shown in Fig. 9. In the process of changing the depth level, it is necessary to keep at least one interval between the robot and the obstacle in the horizontal and vertical directions, which is the radius of the obstacle. In order to simplify the calculation process, taking D as the reference frame, the relative velocity of B is $V_R = VB - V_D$. The floating/diving rate of robot is given as $V \perp DB \rightarrow DD$: combined with the speed of the robot and obstacle and the safety interval, the track in this section can be obtained:

$$\begin{cases} t_{P1} = \frac{d_V}{v_\perp} \\ l = v_R t + d_1 \end{cases} \Rightarrow l = \frac{d_V v_R}{v_\perp} + d_{P1} \tag{4}$$

In order to maintain a safe distance between the robot and the dynamic obstacle, the robot moves from P2 to P3. When the distance between the robot and obstacle in the horizontal direction is l, the robot will change the depth. That is to say, the time required for robot movement in this stage is

$$t_{P2} = \frac{2a_1}{v_R} \tag{5}$$

For the period of P3 to P4, robot diving to the original depth according to the given diving rate. The position relationship between the robot and the obstacle must meet the following requirements when using the depth release,

$$\sqrt{(x_O - x_B)^2 + (y_O - y_B)^2} \ge \frac{d_V v_R}{v_\perp} + d_{P1}$$
(6)

3.2. The time of obstacle avoiding

In an environment with unknown obstacles, there are usually unpredictable obstacles. The underwater robot needs to autonomously complete obstacle avoidance in the case of significant changes in the environment. The most straightforward idea is to move along the line between the starting point and the target point. The robot moves around the edge of the obstacles when



Fig. 10. Circular tangent.

the robot encounters obstacles. This method is more suitable for static obstacles avoidance. Also, it increases the travel length and time of the robot. It means increasing the energy consumption of the robot in the same task and reducing the use efficiency of the robot. To minimize the distance and time of obstacle avoidance, it is necessary to calculate when to start avoiding obstacles and end obstacle avoiding. The characteristics of SUR and motion characteristics of obstacles when calculating the collision time that should to be consider.

According to the distance between the unmanned surface craft and the obstacle, the area around the unmanned surface craft is divided into three areas: safe, conventional obstacle avoidance and emergency obstacle avoidance areas. Among them, the obstacles in the safe area can be determined to be safe. It will pose a threat to unmanned surface craft; at the same time, due to the large safe area, there may be a large number of obstacles. Therefore, in order to reduce the amount of calculation, obstacle avoidance calculation processing is not performed on the obstacles in the safe area. Obstacles in the emergency obstacle avoidance area will pose a serious threat to the unmanned surface craft. For this reason, the unmanned surface craft needs to take emergency measures. This work mainly introduces the obstacle avoidance of the underwater robot in the conventional obstacle avoidance area. When the SUR performs its mission, the velocity and direction are planned through the trajectory tracking algorithm to ensure that the unmanned surface craft can travel along the trajectory calculated by the global path planning. In this study, an improved ant colony algorithm will be used. The algorithm will be explained in detail in chapter 3.3.

We established the coordinate system of the underwater robot and the obstacle coordinate system. $B = (O_B, R_B)$ and V_B are the coordinate system and velocity of the SUR respectively. $O = (O_0, R_0)$ and V_0 as for the obstacle as shown in Fig. 10. The solid line is the position of the underwater robot and the dynamic obstacle at the current time t_0 . We assume that the collision time between the underwater robot and the dynamic obstacle is $t_{\rm C}$, and the dynamic obstacle reaches the position of the yellow dashed line at time t_c . The radius of the dynamic obstacle is expanded according to the radius of the underwater robot, and the radius is expanded to R_e at this time. Because the underwater robot needs some time to move to a specific angle. In order to ensure the safety of the underwater robot, the radius of the obstacle area is expanded to $O_a = \{ O_a, R_a \}, R_{a=} R_B + R_0 + R_T$. Assuming that the speed of the robot is kept constant, obstacle avoidance is performed by changing the direction of the robot's motion. When the robot turns to the left with a constant angular acceleration at a certain time t, it is tangent to the circle along a straight line and point $O_{\rm C}L$. At this time, the tangent distance is the minimum distance between the underwater robot and the dynamic obstacle at the moment of collision.

$$\begin{cases} x = x_t + \int_0^{t_L} v\cos(\theta_0 + w_0 t - \frac{1}{2}at^2)dt \\ y = y_t + \int_0^{t_L} v\cos(\theta_0 + w_0 t - \frac{1}{2}at^2)dt \\ z = z_t + \int_0^{t_L} v\cos(\theta_0 + w_0 t - \frac{1}{2}at^2)dt \end{cases}$$
(7)

where (x,y,z) and (x_t , y_t , z_t) are the coordinates of point $O_{\rm CL}$ and point B in the underwater robot coordinate system. And v, θ_0 , w_0 , a are the current speed, the heading angular, angular velocity and angular acceleration of the underwater robot. The same is true for the robot turning right with a constant angular acceleration. When the robot turns to the right with a constant angular acceleration at a certain time t, it is tangent to the circle along a straight line and point $O_{\rm C}R$. Define the starting time to avoid obstacles as $t_{\rm avoid}$,

$$t_{avoid} = k \max(t_L, t_R) \tag{8}$$

In the formula, k is a coefficient greater than 1, to ensure the safety of the unmanned surface craft and the gentle turn. The underwater robot starts to avoid obstacles at the moment of t_{avoid} , and it can avoid obstacles safely and smoothly.

When the underwater robot starts to avoid obstacles, it is necessary to determine the end time of obstacle avoiding. In this study, the underwater robot needs to travel along the planned trajectory to the target point when performing tasks. Therefore, as long as the underwater robot can safely follow the desired speed and the desired path. In this time, the underwater robot can safely travel in the direction of the target point, the obstacle avoidance can be ended. This means that the underwater robot can end the obstacle avoidance as following,

$$v_{desired} \notin RCC, v \in v_{desired}$$
 (9)

where $v_{desired}$ is velocity vector of the desired path.

3.3. 3D arbitrary path-planning

This paper focuses on optimizing the traveling path and time of underwater robots. Based on the advantages of the ant colony algorithm (ACO) [38–40], this paper designs and implements the path planning algorithm based on the ant colony algorithm and proves the feasibility through corresponding simulation and swimming pool experiments. This paper uses the elite ant strategy and changes the weight of the volatility, reducing the number of iterations of the algorithm. Then, the ant colony algorithm and the particle swarm optimization (PSO) algorithm are merged to solve that the ant colony algorithm is easy to converge in the local solution. Final, the improved algorithm is compared with the traditional ant colony algorithm, and the simulation results prove the feasibility and superiority of the improved algorithm in improving the convergence speed and avoiding the ant colony algorithm falling into local optimization. The improved ant colony algorithm was applied to the path planning of the robot. Through simulation and experimental verification, good results were obtained, which further verified the effectiveness and practicability of the improved algorithm.

The underwater robot should return to the original launch site after completing the underwater mission. Therefore, in this study, the robot is mainly realized after reaching the designated target point and returning to the starting point. As mentioned above, the motion of SUR IV s is very limited, and its path is a straight line for the moving target. The path planning algorithm is used for path planning to ensure that the underwater vehicle can travel along the track line calculated by the global path planning. The following path planning is required,

- (1) From the start point to the target points, and the end point is the same as the start point that is assumed to be $[X_{Start}, Y_{Start}, Z_{Strat}] = [X_{End}, Y_{End}, Z_{End}] = [0,0,0]$. The overall threedimensional environment model is 250 cm × 150 cm × 30 cm.
- (2) After completing the above tasks, return to the start point.
- (3) The shortest path and the lowest energy consumption of the robot.

In order to achieve the requirements of the robot, 3D path planning in this paper adopts the ant colony algorithm. Ant colony algorithm is a simple and easy method of achieving 3D path planning underwater practically. At the beginning of the algorithm, m ants are randomly placed in n cities, and the first element of tabu table is set as the current city. At this time, the number of pheromones on each path is equal. Let, (C is a small constant), each ant chooses the next city independently according to the remaining pheromones and heuristic information (distance between the two cities) on the path. The probability is as follows,

$$p(i,j) = \begin{cases} \frac{[\tau(i,j)]^{\alpha} [\eta(i,j)]^{\beta}}{\sum_{u \in J_k(i)} [\tau(i,u)]^{\alpha} [\eta(i,u)]^{\beta}}, & \text{if } j \in J_k(i) \\ 0, & \text{otherwise} \end{cases}$$
(10)

where $\tau(i, j)$ is the amount of pheromone on the path (i, j) at time t; $J_k(i)$ is a city sequence that can be reached directly from the city and is not visited by antsCity collection in R_k ; α , β is a parameter of its own choice, representing the importance of pheromone concentration and heuristic information respectively. $\eta(i, j)$ is a heuristic message that the expression is as formula (6),

$$\eta\left(i,j\right) = 1/d_{ii} \tag{11}$$

where d_{ij} is the distance between adjacent nodes. The common random greedy algorithm that is an algorithm to choose the shortest path among the currently available paths will be obtained when $\alpha = 0$ in formula (9). Due to ants determine the path completely according to pheromones, the algorithm will converge quickly. The constructed path is often quite different from the actual path, and the performance of the algorithm is not good. In order to solve this problem, the improved algorithm proposed in this paper is to dynamically adjust the values of α and β instead of the fixed values of α and β in the traditional ant colony algorithm. Thereby improving the algorithm convergence speed and escaping from the local optimum, thereby achieving the shortest distance and lowest energy consumption path. In the improved algorithm, we initialize the pheromone η_0 and multiply all the pheromone by the parameter less than ρ . This means that all pheromones are volatile, and pheromone volatility is an inherent characteristic. It can avoid the infinite accumulation of pheromone in the algorithm, and make the algorithm quickly discard the bad path constructed previously. Here is the updated formula,

$$\tau(i,j) = (1-\rho) \tau(i,j) + \sum_{k=1}^{m} \Delta \tau_k(i,j)$$
(12)

$$\Delta \tau_k (i, j) = \begin{cases} \sum_{k=0}^m \Delta \tau_{ij}^k + e \Delta \tau_{ij}^{bs}, \\ 0, other wise \end{cases}$$
(13)

Where m is the number of elite ants in our ants, which accounts for a certain proportion of ants α . For the parameter $\rho(0 < \rho \le 1)$ that is the rate of pheromones evaporate is an adaptive improvement,

$$\rho(t) = \begin{cases} \rho \\ \rho \times \frac{t}{T} \end{cases}$$
(14)

 $\Delta \tau_{ij}^{bs}$ is the pheromone quantity released by the K ant on his side, which is equal to the reciprocal of the length of the path constructed by the K ant in this round and the parameter is given by the following formula

$$\Delta \tau_{ij}^{bs} = \begin{cases} \frac{1}{L_{bs}}, (i,j) \in T^{bs} \\ 0, other wise \end{cases}$$
(15)

Of which L_{bs} is the length of path T_{bs} , we can adjustment the parameters affect weight by this formula. In order to verify the feasibility and effectiveness of the improved algorithm mentioned in the article, the algorithm proposed in this paper is verified. In this article, since the specified target point is reached in a known environment, we import a specific distance matrix. Here is the specific distance matrix,

	Γ 0	3.0000	5.0000	2.0000	3.7500	5.7500	2.7500 -
	3.0000	0	2.0000	5.0000	0.7500	2.7500	5.7500
	5.0000	2.0000	0	3.0000	2.7500	0.7500	3.7500
A =	2.0000	5.0000	3.0000	0	5.7500	3.7500	0.7500
	3.7500	0.7500	2.7500	5.7500	0	2.0000	5.0000
	5.7500	2.7500	0.7500	3.7500	2.0000	0	3.0000
	2.7500	5.7500	3.7500	0.7500	5.0000	3.0000	0 _

After determining the target point determined in this article, the position matrix and coordinate matrix can be obtained. First obtain the number of ants in the ant colony m, and initialize parameters such as ρ and Q. The ants are selected according to the transition probability. The values of m and T_{bs} are dynamically adjusted according to Eqs. (12) and (14). Record the path that has been traversed and add the visited node to the taboo list. Record the foraging path and path length of each ant in each generation. Record the path and path length of the ant iterated once in the ant colony and write it into the cell storage structure CELL. Update the pheromone according to the length of the ants. When the number of iterations of the algorithm is greater than the set maximum number of iterations or the optimal solution given by the algorithm meets the target condition, the program is exited, and the optimal solution is output. This paper adopts elite ant strategies and changes in volatility to improve the ant colony algorithm. In order to further improve the solution speed and obtain a relatively good quality solution.

In order to further improve the solution speed and obtain a relatively good quality solution. Ant colony algorithm is easy

to combine with a variety of heuristic algorithms to improve the performance of the algorithm. PSO has a fairly fast speed of approaching the optimal solution, which can effectively optimize the parameters of the system. The hybrid models of different algorithms are mainly divided into two categories: (1) the global optimization algorithm is mixed with the local optimization algorithm; (2) the global optimization algorithm is mixed with the global optimization algorithm. Looking at all kinds of hybrid algorithms related to PSO algorithm, most of them adopt one strategy to improve it, either with other algorithms or by adding mutation operation, while there are few hybrid algorithms using two strategies at the same time. Although Che proposed an improved ant colony optimization algorithm based on particle swarm optimization (PSO) in previous research [41], the algorithm is mainly aimed at the limited visual domain. Moreover, the algorithm is aimed at finding the best path between the starting point and the target point. In actual situations, the underwater robot usually needs to return to the starting point after completing the task. Therefore, the path planning task in this paper requires the shortest path and the lowest energy consumption path for the underwater robot to return to the starting point after reaching the target points.

We merge the ant colony algorithm and the particle swarm algorithm and use the particle swarm to perform a large-scale search in the overall regional space to ensure that it is not easy to fall into a local solution. At the same time, the optimal solution obtained by the particle swarm algorithm search in the overall regional space is used to update the pheromone of the ant colony and guide the ant colony path search. At the same time, the ant colony algorithm is based on a small range of pheromone search to ensure rapid convergence. In order to allow individuals with high fitness to be preserved, the fitness is defined as,

$$fitness = \frac{1}{distance}$$
(16)

where distance is the distance of each individual. It can be known from Eq. (16) that the shortest distance of each individual has the highest fitness, which is more conducive to passing on to the offspring. Then, update the particle velocity,

$$v_t = w v_{t-1} + r_1 \left(p_{t-1} - x_{t-1} \right) + r_1 \left(g_{t-1} - x_{t-1} \right)$$
(17)

where w is the weight coefficient, v_i is the velocity at the current moment and v_{t-1} is the velocity at the previous time. x_{t-1} , P_{t-1} and g_{t-1} are the individual position, the individual optimal position, and the global optimal position respectively at the previous moment. Based on this, by combining PSO particle swarm optimization algorithm with ant colony algorithm, PSO particle swarm optimization algorithm is used to train the pheromones in ant colony algorithm before searching for the optimal path, so as to obtain a relatively suitable value of pheromones. The pheromone update method of the basic ant colony algorithm has positive feedback. Positive feedback is conducive to the rapid convergence of the algorithm, but there will be more pheromone accumulation on the better path in the later stage of optimization, which will affect the diversity of the population. The algorithm is easy to fall into the local optimum and stagnate, which will eventually lead to a decline in the quality of the solution. Therefore, this paper introduces the probability formula in the algorithm

$$\Delta \tau_{ij}^{k} = \begin{cases} \left(rand < \frac{C_{pbest}}{C} \right) QC_{gbest}^{k} / C^{k} \\ 0 \end{cases}$$
(18)

where C^k represents the total length of the path constructed by the *k*th ant, and Q is a parameter. C_{pbest} and C_{gbest} are respectively the local optimal path length and the global optimal



Fig. 11. The process of the 3D arbitrary path planning.

path length. $\left(rand < \frac{C_{pbest}}{C}\right)$ represents the probability of $\frac{C_{pbest}}{C}$ to update the pheromone. Increasing C_{gbest}^k reduces the proportion of the current path pheromone and can slow down the volatilization ratio of historical pheromone. The two methods together improve the overall ant colony algorithm Search ability. In summary, the process of the 3D arbitrary path planning as shown in Fig. 11.

3.4. Performance comparison of proposed algorithm and existing algorithms

Under the MATLAB environment, the simulation experiments of the three-dimensional path planning of the underwater robot are programmed. The experimental environment is set to a threedimensional environment of 250 cm \times 150 cm \times 30 cm. In the three-dimensional environment, from the starting point to the target point and back to the starting point. The starting point is (0, 0, 0), and the target point is a random 100 points in a three-dimensional environment. The particle swarm algorithm parameters are set to both r₁ and r₂ is 0.1. The simulation results are shown in Fig. 12. The algorithm proposed in this paper can achieve the requirements of robot path planning.

The population optimization algorithm is an algorithm invented by people through bionics research, and it is often used to deal with path planning problems.

Among here, the commonly used algorithms in underwater environments include GA, PSO, GWO, and ACO algorithm [42], which has strong robustness. The PSO algorithm [43] has fast search speed and relatively simple calculations. As for the GWO algorithm [44], it has strong convergence performance and is easy to implement. The advantage of the ACO algorithm [45] is that it has low requirements for the selection of initial conditions and is highly robust. Next, we compare the proposed algorithm with



Fig. 12. The simulation result of improved ACO algorithm. (a) Path planning with improved ACO algorithm. (b) Changing trend of best fitness with improved ACO algorithm.



Fig. 13. Comparison of travel length of the traditional ACO, PSO, GA, and GWO, and improved ACO algorithm.

the following four algorithms: GA, PSO, GA, and traditional ACO. Figs. 13 and 14 correspond the travel times and travel lengths of the 3D path by the five different algorithms in the same underwater environments with a record of per five target points. The numerals along the *x*-axis indicate the number of the target points in the 3D underwater environments.

The experiment was conducted five times. The simulation results show that the average value under the same conditions, the GA has the longest time, followed by the genetic algorithm, and the PSO and the traditional ACO have similar times. Before the number of target points is 30 points, the time required for the



Fig. 14. Comparison of travel times of the traditional ACO, PSO, GA, and GWO, and improved ACO algorithm.

Table 1

Comparison of path length (unit: cm).

	Traditional ACO	PSO	GA	GWO	Improved ACO
1 time	4361.83	8276.13	2993.34	4328.71	2080.73
2 times	4432.54	8156.34	3069.58	4318.56	2030.56
3 times	4499.68	8296.19	2998.25	4458.67	2107.58
4 times	4642.57	8294.56	3032.47	4279.49	2067.09
5 times	4500.11	8128.38	2984.26	4358.19	2003.40
Average	4487.35	8230.32	3015.58	4348.72	2057.87

Table 2

Comparison of the time (unit:s).

	Traditional ACO	PSO	GA	GW0	Improved ACO
1 time	22.23	29.61	39.48	51.32	14.29
2 times	21.78	28.73	41.85	51.58	13.71
3 times	22.39	30.32	40.98	52.98	15.11
4 times	22.11	29.67	38.96	50.37	14.76
5 times	22.25	28.59	39.63	51.84	13.48
Average	22.15	29.38	40.18	51.62	14.27

proposed algorithm and the traditional ACO is not much different. When the number of targets gradually increases, the algorithm proposed in this paper shows a more significant advantage in the travel time. As the target point is 30 points, compared with the traditional ACO, the travel time is increased by 17.87%. Compared with the PSO, GA, and GWO, the travel time is increased by 30.48%, 46.3%, and 65.04%, respectively. The traditional ACO, PSO, GA, and GWO, when the target number is 100 points, the increase compared with the proposed algorithm is 34.72%, 39.41%, 54.56%, and 68.79%, respectively. The target point is before 50 points in terms of travel length, and the travel length required by the five algorithms is not much different. After 50 target points, the algorithm proposed in this paper shows advantages compared with PSO, traditional ACO, and GWO. When the number of target points gradually increased to 80 points, the increase compared to the proposed algorithm of the traditional ACO, PSO, GA, and GWO was 50.97%, 62.08%, 32.51%, and 49.47%.

Compared with the traditional ACO, PSO, GA, and GWO, the algorithm proposed in this paper can avoid premature convergence and quickly realize the shortest path searched by mobile robots. The planning path lengths and time of the traditional ACO, PSO, GA, GWO, and improved algorithms are shown in Tables 1 and 2. The simulation results show that the average path length is 2057.87 cm with the method in this paper and the average path length is 4487.35 cm, 8230.32 cm, 3015.58 cm, and 4348.72 cm



Fig. 15. Comparison of best individual adaptation trend of the traditional ACO, PSO, GA, and GWO, and improved ACO algorithm.

with the method of the traditional ACO, PSO, GA, and GWO, respectively. As for the time, the average time is 14.27 s with the method in this paper and the average time is 22.15 s, 29.38 s, 40.18 s, and 51.62 s with the method of the traditional ACO, PSO, GA, and GWO, respectively. The average path length and average time with the method in this paper are significantly shorter than that of the traditional ACO PSO, GA, and GWO. The simulation results of the trend of best fitness are shown in Fig. 15, and it can be seen that the fusion algorithm has a smaller fitness value during path planning and a faster response time. Thus, considering all aspects, the path produced by the proposed algorithm performs better. The improved algorithm exhibits shortened traveling time and length of underwater robots.

3.5. Localization system

The idea of receiver the position of the robot separates heading navigation and depth control. By processing the data acquired from the IMU, it is possible to track the position and orientation of the device, and, thus, in this way, localization and navigation of the robot can be realized in a given environment. Bases on the location system, a controller was designed to track the guidance commands, Fig. 16 shows the guidance closed loop for the proposed system. Specifically, the pose, body velocity, and acceleration of SUR IV is estimated by fusing data from the onboard sensors via a complementary filter. Subsequently, SUR IV's positions are fed to the motion planning system. The heading autopilot and depth controller process the obtained location information $(x_{\rm C}, y_{\rm C})$ and $z_{\rm C}$ and is responsible for computing the direction of movement approaching the target point. Finally, SUR IV's power system receives the input data to control the robot's movement. Location information is obtained with a 9 DOF-IMU and depth sensor as shown in Fig. 17. We define an inertial frame $E = \{X_E, Y_E, Z_E\}$ located at a fixed position O_E at the vertex of the Y_{B}, Z_{B} attached to SUR IV's center of gravity. The aforementioned JY901 unit used in this research is assumed to be fixed on SUR IV with no extra dynamic included. Thus, we define the sensor frame B= { X_B , Y_B , Z_B } attached at the center of the camera with a fixed frame. Therefore, the 9-DOF IMU can obtain the displacement from the initial position to the current position, thereby obtaining the motion state of the robot. For the posture obtained by the JY901 unit, the coordinates of the sensor are the same as those of the robot. After known the horizontal information, *z*-axis location information will obtain by depth sensor. Then compared the current parameters with the target parameters. If the current



Fig. 16. The framework of location system.



Fig. 17. Location system of SUR IV.

positions were not equal to the target point, the robot attempted to reduce the differences through motion by calculating a suitable trajectory. Robot approaches the target-point position on the x axis, y axis and z axis, respectively. When the offset occurs, the robot will correct the position again.

The workflow of the proposed uncertain moving obstacles avoiding method in 3D arbitrary path planning is summarized as follows.

Step 1. Initialize the system; namely, get the location of the SUR IV.

Step 2. Calculate the location by the SUR IV sensors (IMU and depth sensor).

Step 3. Judge whether there are any dynamic obstacles in the space of the robot.

Step 4. Adjust the robot moving direction based on the uncertain moving obstacles avoiding method, if there are any dynamic obstacles found.

Step 5. The SUR IV moves to the next target point if the target point was occupied.

Step 6. Navigate the SUR IV movement based on the proposed approach.

Step 7. The task is ended if the end point is reached; otherwise go to Step 2. Above all, the simplified flowchart of the software structure for SUR IV as shown in Fig. 18.

4. SUR IV characteristic evaluation experiment

To evaluate the performance of the SUR IV, several experiments were carried out in a pool. The pool is 60 cm in deep with a surface area of 300 cm \times 200 cm. The swimming pool coordinate system is the geodetic coordinate system and the robot's coordinate system is consistent with the 9-DOF IMU. Due to the Open MV camera and 9-DOF IMU were adopted to realize the perceive the obstacles information and localization for SUR in submarine environment. The camera this research is also assumed to be fixed on SUR IV with no extra dynamic included. With the camera and IMU fusion, the obstacles information is received. Before the camera with the waterproof hull is used in the task of perceive



Fig. 18. Simplified flowchart of the software structure for SUR IV.

the obstacles information, the stability of the SUR, the waypoints tracking, and the test of the camera are carried out urgently.

4.1. Stability experiment

The SUR IV can adjust itself to a correct direction range by the use of 9-DoF IMU. The 9-DoF IMU is used to receive the attitudes including yaw, pitch, and roll. For evaluating the stability performance of SUR IV, the experiment is conducted. The forward motion took 18 s and the total distance is 270 cm. Due to the influence of the wind, the continuous interference during the advance motion of the SUR IV (under conditions of a southeast wind with a speed of 6 m/s). The experimental resulting yaw angles and trajectories comparison of the robot in Figs. 19 and 20. From the experimental results, we can see that compared with no PID controller, the application of the PID controller proposed in this paper, the underwater robot can maintain stability in the presence of continuous interference.

4.2. Performance of waypoints tracking experiment

The position of the robot speed was estimated by the 9-DoF IMU and the depth sensors. However, the precision of the 9-DoF IMU was affected by the duration of the evaluation. Considering that the robot needs to locate in the underwater environment for a long time, we corrected the IMU in the previous study. We get the calculation of distance and relative angle. For more details, please refer to our related work [18,28]. Based on the robot's



Fig. 19. Yawing angle comparison of the robot of stability experiment.



Fig. 20. Trajectory comparison of the robot of stability experiment.



Fig. 21. Value evaluated (a)The evaluated of distance.

estimated distanced from the starting point and the angle γ relative to the starting position, the current position can be obtained. Fig. 21 shows the estimated distance and angle of the robot when swimming. The experiment was carried out as follows. First, an effective landmark is set 250 cm ahead of the robot's direction of motion. The robot then swims to the landmark at an approximate speed of 0.15 m/s at t = 15 s. Five experiments are conducted to determine the accuracy of d and the accuracy of d is 3.52 cm \pm 1.2 cm, and the accuracy of φ is 2.36° \pm 0.96° while the robot is moving.

After that, the 2D waypoints tracking experiment was carried out to evaluate the path-planning system of SUR IV, since precise position control is very important for performing underwater operations and tasks. This experiment mainly tests the PID closed-loop control module as the robot travels along a route until reaching target points and then conducts self-rotation at specific points. The AUV follows the ideal trajectory in the yaw direction efficiently. The target points are assumed to be fixed and located at $[X_A, Y_A] = [125,0], [X_B, Y_B] = [250, 0], [X_C, Y_C]$ = [250, 150], and $[X_D, Y_D] = [125, 150]$. Also, the end point is



Fig. 22. Ideal and actual trajectories of the robot.. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



Fig. 23. Trajectory of the robot of waypoints tracking experiment.

 $[X_F, Y_F] = [0, 150]$. Of these, two points will self-rotate upon arrival, i.e., those located at [X_A, Y_A] and [X_D, Y_D]. The start point of SUR IV are assumed to be (0, 0) in coordinates and there is no initial yaw angle. The radius of acceptance is assumed to be 25 cm (half of SUR IV's circumference) and as mentioned earlier. SUR IV has a constant velocity of 0.15 m/s in the surge direction, and the vehicle is in a horizontal posture. The start point of SUR IV is shown by a triangle, while the target points are marked by stars and the end position by a square. In addition, the ideal trajectory that joins the target points and the start and end points are shown by the red line. 2D path-planning ideal and actual trajectories are illustrated in Fig. 22. The circles of acceptance are depicted therein, as is self-rotation at the target point. These views indicate the good performance of this 2D guidance system. Owing to the depth limitation of the swimming pool, the robot can dive to a maximum depth of 40 cm and remain moving at the depth position. The entire process of the underwater robot's 2D path-planning task is shown in Fig. 23.

4.3. 3D Arbitrary path planning experiment

After the positive results obtained in simulation of 3D arbitrary path planning, we moved to the real-world environment. According to the improved ant colony algorithm proposed in this paper, the shortest path that meets the requirements is obtained. The experimental site as shown in Fig. 24. The end point is the same as the start point that is assumed to be $[X_{Start}, Y_{Start}, Z_{Strat}] = [X_{End}, Y_{End}, Z_{End}] = [0,0,0]$. In addition, due to the limitation



(a) Experimental site.



(b) The prototype of the SUR IV.

Fig. 24. Experimental configuration for SUR IV path planning.

of the experimental site, six target points are selected in the path experiment. The preset target points, that are [XA, YA, ZA] = [250, 0, 0], [XB, YB, ZB] = [250, 0, -30], [XC, YC, ZC] = [250, 150, 0], [XD, YD, ZD] = [250, 150, -30], [XE, YE, ZE] = [0, 150, 0] and [XF, YF, ZF] = [0, 150, -30]. The entire process of the underwater robot's 3D path planning as shown in Fig. 25.

Owing to accelerometers having a cumulative error, the robot will be offset during movement. Because the error is corrected when the acceleration is constant, the robot can approach the target point, and we assume that the radius of acceptance with the start and end points is 10 cm. The marking method is the same as that in the 2D path-planning task. The processes of the 3D path planning task as shown in Fig. 26. The localization errors of start point and end point in the moving direction of Start \rightarrow E \rightarrow F \rightarrow C \rightarrow D \rightarrow B \rightarrow A \rightarrow End occupied 8.5%, 2.5% and 6.3% to the size (54 cm) of the robot in x, y, and z axis. This indicates the good performance of the 3D path planning characteristics of SUR IV, which reached all the target points and, most importantly, did not undergo rapid path changes during motion controlled by the path-planning system proposed in this study.

In order to better verify the advantages of the algorithm proposed in this paper, we conducted comparison experiments. Since the algorithm proposed in this paper is mainly to optimize the travel time and length. In the comparison of simulation results as shown in Figs. 13 and 14, it can be seen that comparing the PSO, GA, GWO algorithms with the algorithm proposed in this paper has a longer travel time. Therefore, only the comparative experiment between the traditional ACO algorithm and the improved



Fig. 25. The trajectories of 3D path planning. (The yellow box is the SUR IV.).



Fig. 26. Processes of the 3D path planning task for six target points.

algorithm is carried out. We conducted underwater experiments with six target points and 12 target points. The experimental results of three-dimensional path planning for six target points are shown in Fig. 26, and the experimental results for 12 target points are shown in Fig. 27. And the processes of the 3D path planning task for 12 target points as sown in Fig. 28. The end point is the same as the start point that is assumed to be $[X_{Start}, Y_{Start}, Z_{Strat}] = [X_{End}, Y_{End}, Z_{End}] = [0,0,0]$. The preset target points, that are [XA, YA, ZA] = [125, 0, 0], [XB, YB, ZB] = [125, 0, -30], [XC, YC, ZC] = [250, 0, -30], [XD, YD, ZD] = [250, 0, 0], [XE, YE, ZE] = [250, 150, 0], [XF, YF, ZF] = [250, 150, -30], [XG, YG, ZG] = [125, 150, -30], [XH, YH, ZH] = [125, 150, 0], [XI, YI, ZI] = [0, 150, -30] and [XK, YK, ZK] = [0, 0, -30].

The average planning path lengths and the motion time of the ACO and improved ACO algorithms are shown in Table 3. The experiment result of 6 target points indicates that the average path length is 1014 cm with the method in this paper. The average path length is 1016 cm with the method of the traditional ACO algorithm. As for the motion time, the average is 71.2 s with the algorithm in this paper, and the average is 72 s with the method of the traditional ACO algorithm. The reduction in length and time of SUR is not significant as there are fewer target points in this task of 3D path planning. As for the experiment result of 12 target points of 3D path planning, the average path length is 1215 cm with the method in this paper, and the average



Fig. 27. Processes of the 3D path planning task for 12 target points. (The yellow box is the SUR IV.).

path length is 1382.6 cm with the method of the traditional ACO algorithm. As for the motion time, the average motion time is 72 s with the method in this paper, and the average motion time is 63.7 s with the method of the traditional ACO algorithm. With the gradual increase in the number of target points, the advantages in reducing the travel length and time of the three-dimensional path algorithm proposed in this paper are more obvious.

4.4. Obstacles avoiding experiments

To verify the validity and accuracy of the dynamic obstacle avoidance method, the cases of dynamic for obstacles avoidance are designed. Obstacle position and motion information are



Fig. 28. Processes of the 3D path planning task for 12 target points.

(a) at 0s (b) at 1s (c) at 14s

Fig. 29. Trajectory of the robot of waypoints tracking experiment.



Fig. 30. Experimental result for SUR IV static obstacles avoiding.

Table 3

Comparison of path length (unit:cm) and the motion time(unit:s).

-				
	Traditional ACO with 6 target points	Improved ACO with 6 target points	Traditional ACO with 12 target points	Improved ACO with 12 target points
Path length	1016	1014	1382.6	1215
Motion time	72	71.2	72	63.7

unknown. Experiments and Results are conducted to verify the effectiveness of dynamic obstacle avoidance. As the front view camera range and distance is limited, the algorithm does not recognize the moving obstacles from the rear of the SUR. In this paper, we assume that there is no obstacle coming from behind SUR. The shape of the static obstacle is the cylinder. The dynamic obstructions are considered as the hemisphere, wherein the diameter of the smallest circumscribed circle of the obstacle is 25 cm. The motion parameters are defined as $\Delta V = 0.1$ m/s.

First, we conducted obstacle avoidance experiments with stationary obstacles. The experimental results are shown in Figs. 29 and 30. The Fig. 29 is the view of camera in the same moment and Fig. 30 is the view of the experimental site. We set up a stationary obstacle in front of the robot. The stationary obstacle



Fig. 31. Experimental result for SUR IV moving obstacles avoiding. (The yellow box is the SUR IV; the red box is the dynamic obstacle.). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

is a hemisphere with a diameter of 25 cm. Assuming that the robot maintains a constant speed and movement direction and successfully recognizes obstacles; the obstacle avoidance starts at the intersection point (P) between the extension line of the robot speed and the tangent line of the obstacle avoidance area. The robot turned slightly to the right and successfully avoided obstacles. When the robot travel along the planned trajectory, the obstacle avoidance of the robot ends. After that, we conducted experiments in which dynamic obstacles from the oblique front at a constant speed of 0.1 m/s, and the movement direction of the obstacle remains unchanged. SUR can find obstacles and successfully avoid them. The robot drives from the starting point to the pre route for 10 s, and the obstacle avoidance process lasts for 3 s. SUR found the obstacle in 4 s. According to the direction and speed of the obstacle, the robot turned right slightly and successfully avoided the obstacle. Fig. 31 show the entire process of SUR IV moving obstacles avoidance task.

We design two static obstacles S1, S2, and three dynamic obstacles D1, D2, D3 in the environment to the environment of the SUR voyage. The experimental site as shown in Fig. 32. The total time of the obstacles' avoidance is 76 s, and the SUR safely arrives at the destination. The distribution of obstacles and the effect of avoidance can be found from this Fig. 33. The size and velocity information of the obstacle is the same as described in Section 4.3. The SUR finds the static obstacle S1, turns left, and avoids successfully. Then it found the dynamic obstacle, turned slightly left, and travel along the planned trajectory to the first target point. And then the SUR can get the next target points. Since the target points, 4 and 5 are occupied by the stationary obstacle S2, the SUR will search for the next target point after detecting in a three-dimensional direction and confirming that it cannot reach the specified point. Before reaching the next target point, the robot again finds the moving obstacle M1 and turns to the left to avoid it again. Finally, the robot reaches the target point. Throughout the process, the robot heading, and speed changes are smaller, we can see that the proposed method was conducted well. The minimum distance between the robot and obstacles is nearly half a SUR (27 cm), which can fully guarantee the safety of SUR. The experimental configuration as shown in Fig. 33.



Fig. 32. Experiment site for SUR IV 3D path planning.

A comparison experiment was conducted to verify the time of obstacle avoidance of the proposed method. Fig. 34 shows the entire process of the 3D path planning task with obstacles avoidance. In the experiment, the SUR moved along the same reference trajectory based on the 3D path planning algorithm proposed in this paper. The triangle represents the starting point of the path, and the triangle represents the target point. Compared with the 70 s required by the 3D arbitrary path planning experiment shown in Fig. 25, the total time of the experiment with introducing the time of obstacle avoidance is only increased by 6 s after the obstacle avoidance task was added. And the vector direction of the robot motion always faces the target point. In the comparison experiment, without introducing the time of obstacle avoidance, the total movement time of the robot is 92 s. As for the travel length of the robot, the total travel length is 1057 cm in the experiment with the concept of the time of obstacle avoidance. And in the experiment without the time of obstacle avoidance, the total travel length is 1394 cm. After introducing the time of obstacle avoidance of the proposed method, the travel time and length were reduced by 17.4% and 24.2%, respectively. The introduced obstacle avoidance time concept can significantly improve the obstacle avoidance performance of the robot. Experiments verify the effectiveness and rationality of the obstacle avoidance algorithm.

5. Conclusions

In this paper, we present uncertain moving obstacles avoiding method in 3D arbitrary path planning for a SUR IV. First, we proposed a stability system that used a PID controller to control the direction and magnitude of the thrusters to ensure the stability of SUR IV with disturbance. The stability controller provided the premise of path planning. And then, the dynamic collision avoidance method is based on improved velocity obstacles that can be solved the real-time collision avoidance problem. By introducing the concept of collision avoidance time, it is possible to obtain the correct time to avoid collisions. After introducing the time of obstacle avoidance of the proposed method, the travel time and length of the SUR have been reduced, and it can significantly improve the obstacle avoidance performance of the robot. Finally, the 3D arbitrary path planning is based on an improved ant colony algorithm that uses the elite ant strategy. Also, the ant colony algorithm and the particle swarm optimization algorithm are merged to solve that the ant colony algorithm is easy to converge in the local solution. The improved ACO algorithm is used for global path planning can achieve the shortest path searched by the mobile robot in a short time.

Robotics and Autonomous Systems 151 (2022) 104011



Fig. 33. Experimental results for SUR IV 3D path planning with obstacles avoiding. (The yellow box is the SUR IV; the red box is the dynamic obstacle.). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Due to the special of the underwater environment, more problems need to be solved when conducting underwater experiments, such as the attitude control of underwater robots and the timeliness of path planning algorithm processing. Therefore, many current three-dimensional path planning algorithms with obstacle avoidance are only evaluated by simulation experiments, and the proposed algorithm is difficult to be applied in real experiments or applied to small underwater robots. For this reason, it is challenging in conducting experiment evaluations. We conducted three-dimensional path planning underwater experiments in uncertain moving obstacles and achieved 3D arbitrary path planning for the underwater robot safely and efficiently. Experiments were conducted to evaluate SUR IV dynamic obstacles avoidance and 3D path-planning characteristics.

Since the 3D path planning algorithm for arbitrary target points based on the ant colony algorithm proposed in this study is only suitable for use in known environments. In the future, on the one hand, we will optimize the proposed path planning algorithm to adapt it to the real environment with complexity and unknown. On the other hand, replacing more accurate cameras and faster processors allows our uncertain dynamic obstacles avoidance algorithm to be used in real environments from full view.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was funded by the SPS KAKENHI (15K2120).



Fig. 34. Processes of the 3D path planning task with obstacles avoiding.

References

- D. Kim, et al., Artificial landmark based underwater localization for AUVs using weighted template matching, Intell. Serv. Robot. 7 (3) (2014) 175–184.
- [2] M. Tim, R.V. Parys, G. Pipeleers, Spline-based motion planning for autonomous guided vehicles in a dynamic environment, IEEE Trans. Control Syst. Technol. 26 (6) (2017) 2182–2189, http://dx.doi.org/10.1109/TCST. 2017.2739706.
- [3] M.G. Mohanan, S. Ambuja, A survey of robotic motion planning in dynamic environments, Robot. Auton. Syst. 100 (2018) 171–185, http://dx.doi.org/ 10.1016/j.robot.2017.10.011.
- [4] R. Conti, E. Meli, A. Ridolfi, B. Allotta, An innovative decentralized strategy for I-AUVs cooperative manipulation tasks, Robot. Auton. Syst. 72 (2015) 261–276, http://dx.doi.org/10.1016/j.robot.2015.06.006.
- [5] M. Oscar, U.O. Rosas, R. Sepúlveda, Path planning for mobile robots using bacterial potential field for avoiding static and dynamic obstacles, Expert Syst. Appl. 42 (12) (2015) 5177–5191, http://dx.doi.org/10.1016/j.eswa. 2015.02.033.
- [6] E.J. Molinos, A. Llamazares, M. Ocaña, Dynamic window-based approaches for avoiding obstacles in moving, Robot. Auton. Syst. 118 (2019) 112–130, http://dx.doi.org/10.1016/j.robot.2019.05.003.
- [7] H. Rezaee, A. Farzaneh, A decentralized cooperative control scheme with obstacle avoidance for a team of mobile robots, IEEE Trans. Ind. Electron. 61 (1) (2013) 347–354, http://dx.doi.org/10.1109/TIE.2013.2245612.

- [8] M.H. Korayem, S.R. Nekoo, The SDRE control of mobile base cooperative manipulators: Collision free path planning and moving obstacle avoidance, Robot. Auton. Syst. 86 (2016) 86–105, http://dx.doi.org/10.1016/j.robot. 2016.09.003.
- [9] A.K. Khalaji, H. Tourajizadeh, Nonlinear lyapounov based control of an underwater vehicle in presence of uncertainties and obstacles, Ocean Eng. 198 (2020) 106998, http://dx.doi.org/10.1016/j.oceaneng.2020.106998.
- [10] J. Ni, L. Wu, P. Shi, Simon X. Yang, A dynamic bioinspired neural network based real-time path planning method for autonomous underwater vehicles, Comput. Intell. Neurosci. (2017) http://dx.doi.org/10.1155/2017/ 9269742.
- [11] Y. Huang, L. Chen, P.H.A.J.M. Van Gelder, Generalized velocity obstacle algorithm for preventing ship collisions at sea, Ocean Eng. 173 (2019) 142–156.
- [12] J. López, P.S. Vilariño, M.D. Cacho, E.L. Guillénb, Obstacle avoidance in dynamic environments based on velocity space optimization, Robot. Auton. Syst. 131 (2020) 103569, http://dx.doi.org/10.1016/j.robot.2020.103569.
- [13] T. Xu, S. Zhang, et al., Collision avoidance of high-speed obstacles for mobile robots via maximum-speed aware velocity obstacle method, IEEE Access 8 (2020) 138493-138507.
- [14] Y. Ma, et al., Path planning for autonomous underwater vehicles: An ant colony algorithm incorporating alarm pheromone, IEEE Trans. Veh. Technol. 68 (1) (2018) 141–154.
- [15] C. Xiong, D. Chen, D. Lu, et al., Path planning of multiple autonomous marine vehicles for adaptive sampling using Voronoi-based ant colony optimization, Robot. Auton. Syst. 115 (2019) 90–103, http://dx.doi.org/10. 1016/j.robot.2019.02.002.
- [16] Z. Tahir, et al., Potentially guided bidirectionalized RRT* for fast optimal path planning in cluttered environments, Robot. Auton. Syst. 108 (2018) 13–27, http://dx.doi.org/10.1016/j.robot.2018.06.013.
- [17] A.M. Yazdani, et al., IDVD-based trajectory generator for autonomous underwater docking operations, Robot. Auton. Syst. 92 (2017) 12–29, http: //dx.doi.org/10.1016/j.robot.2017.02.001.
- [18] X. Cao, et al., Multi-AUV target search based on bioinspired neurodynamics model in 3-D underwater environments, IEEE Trans. Neural Netw. Learn. Syst. 27 (2016) 2364–2374, http://dx.doi.org/10.1109/TNNLS.2015.2482501.
- [19] P. Yao, S. Zhao, Three-dimensional path planning for AUV based on interfered fluid dynamical system under ocean current, IEEE Access 6 (2018) 42904–42916, http://dx.doi.org/10.1109/ACCESS.2018.2861468.
- [20] Y. Ma, and, et al., Obstacle avoidance path planning of unmanned submarine vehicle in ocean current environment based on improved firework-ant colony algorithm, Comput. Electr. Eng. 87 (2020) 106773, http://dx.doi.org/ 10.1016/j.compeleceng.2020.106773.
- [21] S. Perez-Carabaza, E. Besada-Portas, J.A. Lopez-Orozco, J.M. de la Cruz, Ant colony optimization for multi-UAV minimum time search in uncertain domains, Appl. Soft Comput. 62 (2018) 789–806, http://dx.doi.org/10.1016/ j.asoc.2017.09.009.
- [22] S. Gu, S. Guo, Performance evaluation of a novel propulsion system for the spherical underwater robot (SURIII), Appl. Sci. 7 (11) (2017) 1196, http://dx.doi.org/10.3390/app7111196.
- [23] R. An, S. Guo, S. Gu, L. Zheng, Improvement and evaluation for the stability of mobile spherical underwater robots (SUR III), in: Proceedings of 2019 IEEE International Conference on Mechatronics and Automation, ICMA, Tianjin, China, 2019, pp. 2512–2517, http://dx.doi.org/10.1109/ICMA.2019. 8816247.
- [24] S. Gu, S. Guo, L. Zheng, A highly stable and efficient spherical underwater robot with hybrid propulsion devices, Auton. Robots 44 (5) (2020) 759–771.
- [25] X. Hou, et al., Hydrodynamic analysis-based modeling and experimental verification of a new water-jet thruster for an amphibious spherical robot, Sensors 19 (2) (2019) 259, http://dx.doi.org/10.3390/s19020259.
- [26] X. Hou, et al., Improved model predictive-based underwater trajectory tracking control for the biomimetic spherical robot under constraints, Appl. Sci. 10 (22) (2020) 8106.
- [27] J. Guo, C. Li, S. Guo, A novel step optimal path planning algorithm for the spherical mobile robot based on fuzzy control, IEEE Access 8 (2019) 1394–1405, http://dx.doi.org/10.1109/ACCESS.2019.2962074.
- [28] S. Guo, et al., Modeling and experimental evaluation of an improved amphibious robot with compact structure, Robot. Comput.-Integr. Manuf. 51 (2018) 37–52, http://dx.doi.org/10.1016/j.rcim.2017.11.009.
- [29] H. Xing, et al., Robust RGB-D camera and IMU fusion-based cooperative and relative close-range localization for multiple turtle-inspired amphibious spherical robots, J. Bionic Eng. 16 (3) (2019) 442–454.
- [30] L. Zheng, et al., Collaboration and task planning of turtle-inspired multiple amphibious spherical robots, Micromachines 11 (1) (2020) 71, http://dx. doi.org/10.3390/mi11010071.
- [31] J. Guo, C. Li, S. Guo, Study on the autonomous multirobot collaborative control system based on spherical amphibious robots, IEEE Syst. J. (2020) http://dx.doi.org/10.1109/JSYST.2020.3023727.

- [32] J.S. Jaffe, et al., A swarm of autonomous miniature underwater robot drifters for exploring submesoscale ocean dynamics, Nat. Commun. 8 (1) (2017) 1–8.
- [33] D. Costa, et al., Design of a bio-inspired autonomous underwater robot, J. Intell. Robot Syst. 91 (2) (2018) 181–192.
- [34] A. Mazumdar, A. Fittery, et al., A ball-shaped underwater robot for direct inspection of nuclear reactor and other water-filled infrastructure, in: Proceedings of 2013 IEEE International Conference on Robotics and Automation, 2013, pp. 3415–3422.
- [35] O. Chocron, U. Prieur, L. Pino, A validated feasibility prototype for AUV reconfigurable magnetic coupling thruster, IEEE/ASME Trans. Mechatronics 19 (2) (2013) 642–650, http://dx.doi.org/10.1109/TMECH.2013.2250987.
- [36] R. An, S. Guo, L. Zheng, et al., Modeling and experimental verification of a new spherical underwater robot, in: Proceedings of 2020 IEEE International Conference on Mechatronics and Automation, ICMA, Beijing, China, 2019, pp. 1222–1227, http://dx.doi.org/10.1109/ICMA49215.2020.9233765.
- [37] Z. Chen, Z. Zhang, F. Dai, et al., Monocular vision-based underwater object detection, Sensors 17 (2017) 1784.
- [38] E. R. Conti, et al., An innovative decentralized strategy for I-AUVs cooperative manipulation tasks, Robot. Auton. Syst. 72 (2015) 261–276, http: //dx.doi.org/10.1016/j.robot.2015.06.006.
- [39] W. Zhang, et al., Dynamic obstacle avoidance for unmanned underwater vehicles based on an improved velocity obstacle method, Sensors 17 (2017) 2742.
- [40] E. Kelasidi, et al., Path following, obstacle detection and obstacle avoidance for thrusted underwater snake robots, Front. Robot. AI 6 (2019) 57, http: //dx.doi.org/10.3389/frobt.2019.00057.
- [41] G. Che, et al., An improved ant colony optimization algorithm based on particle swarm optimization algorithm for path planning of autonomous underwater vehicle, J. Ambient Intell. Humaniz. Comput. 11 (2020) 3349–3354.
- [42] B. Matteo, et al., Path planning for underwater information gathering based on genetic algorithms and data stochastic models, J. Mar. Sci. Eng. 9 (11) (2021) 1183, http://dx.doi.org/10.3390/jmse9111183.
- [43] Z. Yan, et al., A real-time path planning algorithm for AUV in unknown underwater environment based on combining PSO and waypoint guidance, Sensors 19 (1) (2019) 20, http://dx.doi.org/10.3390/s19010020.
- [44] M. Panda, et al., Global path planning for multiple AUVs using GWO, Arch. Control Sci. (2020) 77–100.
- [45] C. Qin, et al., A hierarchical information acquisition system for AUV assisted internet of underwater things, IEEE Access 8 (2020) 176089-176100, http: //dx.doi.org/10.1109/ACCESS.2020.3026395.



Ruochen An received the B.S. degree and the M.S. degree at Changchun University of Science and Technology (CUST), China, in 2015 and 2018, respectively. Now, Miss. An is as a Ph.D. student in Intelligent Mechanical Systems Engineering, Kagawa University (KU), Japan. She has published 1 conference papers in recent years. Her current research interests are spherical underwater robotics; bioinspired robotics.



Shuxiang Guo (SM'03-F'21) received the Ph.D. degree in mechanoinformatics and systems from Nagoya University, Nagoya, Japan, in 1995. He is a Full Professor at the Department of Intelligent Mechanical System Engineering, Kagawa University, Takamatsu, Japan, since 2005. He is an IEEE Fellow, since 2020. He is an Engineering Academy of Japan, Fellow (Guest Member), since 2020. He is also the Chair Professor at the Key Laboratory of Convergence Medical Engineering System and Healthcare Technology, Ministry of Industry and Information Technology, Beijing Institute of Technol-

ogy, Beijing, China. He has published about 710 refereed journal and conference papers. He is an Editor-in-Chief of the International Journal of Mechatronics and Automation. His research interests include medical robot systems, microcatheter systems, and biomimetic underwater robots.



Liang Zheng received the B.S. degree and the M.S. degree at Changchun University of Science and Technology (CUST), China, in 2006 and 2010, respectively. Now, Mr. Zheng is as a Ph.D. student in Intelligent Mechanical Systems Engineering, Kagawa University (KU), Japan. He has published 4 conference papers and 2 journal paper in recent years. His current research interests are amphibious spherical underwater robotics; bio-inspired robotics.



Hideyuki Hirata received the B.S. and M.S. degrees from Yokohama National University, Yokohama, Japan, in 1981 and 1983, respectively, and the Ph.D. degree from the Tokyo Institute of Technology, Tokyo, Japan, in 1992, all in mechanical engineering. He is currently Professor of Intelligent Mechanical Systems Engineering with Kagawa University, Takamatsu, Japan. His current research interests include material strength, material design, and microdevices based on computer simulation technology.



Robotics and Autonomous Systems 151 (2022) 104011

Shuoxin Gu received the B.S. degree at Changchun University of Science and Technology (CUST), China, in 2011, then received the M.S. degree at Jilin University (JLU), China, in 2015. Now, Miss. Gu is graduated from the Intelligent Mechanical Systems Engineering at Kagawa University (KU), Japan in 2019. After that, she came to Chengdu University of Information Technology to work as a lecturer, and continued to engage in research on underwater robots. She has published more than ten papers in recent years, including 2 journal papers and 3 conference papers as the first author.

Her current research interests are spherical underwater robotics; bio-inspired robotics.