Proceedings of 2023 IEEE
International Conference on Mechatronics and Automation
August 6 - 9, Harbin, Heilongjiang, China

Research on Target Encirclement Strategy of Amphibious Spherical Robots in 3D Underwater Environments

Dan Yang[1,3], Liwei Shi[1,3,*], Shuxiang Guo[1,2,3,4], Shilong Feng[2,3]

[1] *School of Medical Technology, Beijing Institute of Technology, Beijing, 100081, China*
[2] *School of Life Science, Beijing Institute of Technology, Beijing, 100081, China*
[3] *Key Laboratory of Convergence Medical Engineering System and Healthcare Technology(Beijing Institute of Technology), Ministry of Industry and Information Technology, Beijing, 100081, China*
[4] *The Department of Electronic and Electrical Engineering, Southern University of Science and Technology, Shenzhen, Guangdong 518055, China*

* Corresponding author: shiliwei@bit.edu.cn

***Abstract* - The cooperative trapping of multiple amphibious robots in a 3D underwater environment is a comprehensive research topic that mainly includes the generation of dynamic alliances for trapping and path planning. To address this problem, this paper first proposes a "trap point occupancy strategy" to realize the generation of dynamic alliances among multiple robots during the trapping process. Secondly, based on this, an improved gray wolf algorithm is proposed in this paper to implement robot target guidance and adaptive obstacle avoidance functions, used for the path planning of the robot's arrival at the trapping point during the trapping process, thus completing the entire trapping task. Then, this paper conducts simulation experiments in a 3D underwater environment with static obstacles, and the results confirm that the proposed cooperative trapping algorithm has high efficiency and robustness. Finally, the innovations and all the work in the article are summarized, and prospects are put forward.**

Index Terms - Keywords: Amphibious spherical robot, multi-robot collaborative encirclement, encirclement point occupation algorithm, improved gray wolf algorithm

I. Introduction

With the development of technology and the advancement of productivity levels, the global population is growing rapidly, which also leads to further depletion of land resources. In recent years, people have begun to turn their attention to the ocean, and autonomous underwater vehicles (AUVs) as intelligent robots that can autonomously complete underwater operations without real-time operation have been widely used in the exploration and development of marine resources [1-5]. Currently, research on single AUV systems at home and abroad is relatively mature. However, in complex 3D underwater environments, a single robot is difficult to complete the complex tasks assigned to it, so multi-AUV systems are widely researched. The advantage of the multi-AUV system is that it does not rely on the high performance of a single robot in the system, and the failure of a single robot will not affect the operation of the entire system, which greatly reduces the risk of task failure.

In the widespread application of multi-AUV systems, collaborative encirclement technology has attracted extensive attention from researchers in the field of robotics. The implementation of a complete encirclement task includes two main aspects [6-7]: (1) the allocation of the encirclement point location, that is, planning encirclement points that can successfully encircle the evading AUV, and assigning these encirclement points to the encircling AUVs in an optimal way to minimize the total path distance to reach the encirclement point. (2) Robot path planning, in order to enable each AUV to reach the assigned encirclement point along the shortest path while successfully avoiding obstacles, it is necessary to plan the traveling path.

In response to the encirclement problem with multiple AUVs, there are currently various strategies available. Chen and NI [1,8] applied the bio-inspired neural network (BNN) algorithm to multi-robot encirclement problems. Although their proposed method can effectively find the optimal path, the communication and computational costs are relatively high, making it unsuitable for small underwater robots.

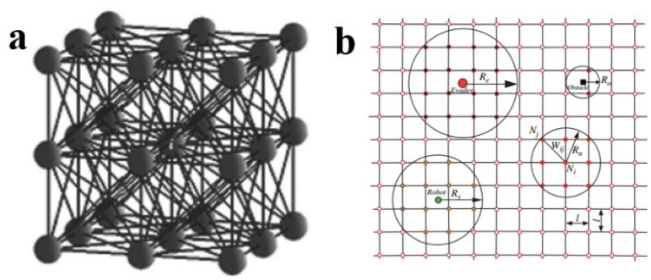


Fig. 1 Neural network model: a. 3D connected neural network [1]; b. Neural network model for the hunting task [8]

Researchers at Shandong University, led by YU, have improved the RTT* algorithm and proposed a Cyl-HRRT* algorithm [9]. By biasing the sampling and expansion towards a subset of cylindrical states, this algorithm improves current solutions and provides better paths for Autonomous Underwater Vehicles (AUVs). Although this path planning method has been proven to have good obstacle avoidance effects in different obstacle environments, the Cyl-HRRT* algorithm requires complex branch-and-bound search. Considering the mechanical structure and kinematics constraints of small spherical robots, using the Cyl-HRRT* algorithm will instead increase the algorithm complexity, leading to decreased computational efficiency. Chen [10] proposed a capture strategy based on Bug2 and angle-first, which guides the encircling AUV to the target location by directing it to move in a straight line towards the target location or along the edge of obstacles. Although this method is simple and easy to implement, it only plans for straight-line and along-the-edge movements, and does not necessarily obtain the shortest and optimal path to reach the target location.

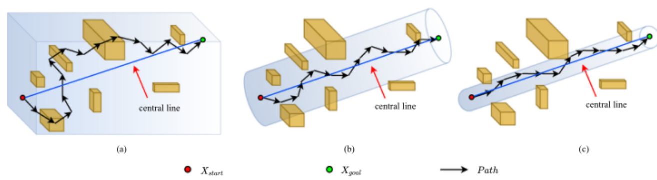


Fig. 2 Using the Cyl-HRRT* to perform subset contraction during search [9]

To address this problem, Yin and Cao [11-12] proposed a multi-AUV cooperative target search strategy based on improved potential fields. However, their experimental results only confirmed the effectiveness of this method in a two-dimensional environment, and its applicability in a three-dimensional environment needs further research and verification.

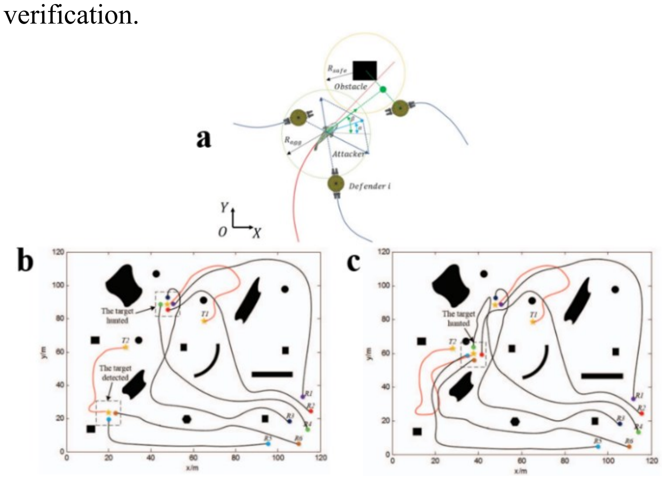


Fig. 3 Artificial potential field planning: a. The generation mechanism of resultant force based on artificial potential field [11]; b-c. Simulation of hunting process with two targets [12]

To address the problems identified in the literature above, this paper proposes a target point allocation strategy based on "capture point occupancy" [1] and an improved wolf pack algorithm for path planning, in order to efficiently encircle and capture escaping AUVs in a three-dimensional environment. The results prove that the strategy requires fewer capture AUVs, has lower computational costs, and generates shorter trajectory paths, making it highly applicable in a three-dimensional environment.

The algorithm advancements and breakthroughs in this article can be summed up as follows: First, in order to successfully prevent the issue of robots duplicating encirclement locations, we assigned the positions of the encirclement points prior to conducting the encirclement; Second, to better determine the fitness value of each position, we improved the heuristic function of the conventional wolf pack algorithm and used more appropriate evaluation criteria. We also carefully considered the Euclidean distance between the robot position, obstacles, and the target point. Then, we introduced the size information of the robot, and when judging whether a collision occurs, we no longer simply consider the collision between the path and the obstacles, but also whether the edges of the spherical robot will collide, making the algorithm more suitable for practical problems; Finally, we introduced an adaptive parameter control strategy, which improves the robustness and adaptability of the algorithm.

The structure of the remaining parts of this paper is as follows: Section 2 describes the problem addressed in this paper. Section 3 introduces the algorithm principles and implementation process of the proposed collaborative trapping strategy. In order to evaluate the performance of the proposed algorithm, Section 4 conducts simulation experiments and provides experimental results. Section 5 summarizes and outlooks the work of the entire paper, and points out the innovations of this paper.

II. PROBLEM DESCRIPTION

The cooperative encirclement problem in multi-robot systems refers to the task of assigning one or more robots to capture one or more moving targets, and how to plan the robot's path to maximize the success rate of the encirclement. In underwater environments, multi-robot encirclement problems have higher complexity because robot movement and communication are influenced by environmental factors such as water flow and obstacles, while target movement is also more complex. Therefore, before analyzing the problem, it is necessary to construct an equivalent simulation environment in underwater terrain.

The most common method for modeling threats and obstacles in complex underwater environments is to represent them as mountainous terrain [13]. Figure 4 shows a schematic diagram of randomly generated mountain-shaped terrain used to simulate underwater terrain, and the actual environment has been proportionally reduced. Subsequent problem-solving is based on this environment.

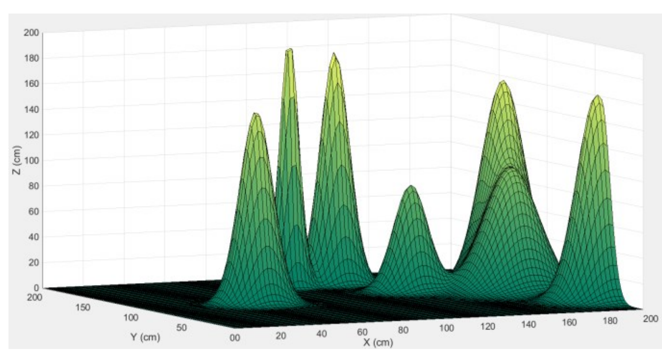


Fig. 4 Three-dimensional underwater environment simulation diagram

Consider a multi-robot system consisting of N encircling robots and one target robot, the position and velocity of the $i-th$ robot at time t can be represented as:

$$P_i(t) = [x_i(t), y_i(t), z_i(t)] \in R^3 \quad (1)$$
$$v_i(t) = [u_i(t), w_i(t), r_i(t)] \quad (2)$$

Among them, $x_i(t)$, $y_i(t)$, $z_i(t)$ represents the component size of robot i at time t along the x-axis, y-axis, and z-axis directions, $u_i(t)$, $w_i(t)$, $r_i(t)$ represents the component size of the velocity.

The position and velocity of the target robot at time t can be represented as:

$$P_a(t) = [x_a(t), y_a(t), z_a(t)] \in R^3 \quad (3)$$
$$v_a(t) = [u_a(t), w_a(t), r_a(t)] \quad (4)$$

Among them, $x_i(t)$, $y_i(t)$, $z_i(t)$ represents the component size of target i at time t along the x-axis, y-axis, and z-axis directions, $u_i(t)$, $w_i(t)$, $r_i(t)$ represents the component size of the velocity.

Therefore, the distance vector between the $i-th$ robot and the target robot can be represented as $R_{ai}(t) = |P_i(t) - P_a(t)|$. In the three-dimensional collaborative encirclement problem, there must be at least six defenders to form the most basic "encirclement formation" [6-7], where defenders can also be obstacles.

The judgment condition for a successful capture is that the encircling robots are uniformly distributed in the six directions of front, back, left, right, up, and down at a distance of r from the target robot through diving and surfacing operations, while surrounding the target at its geometric center. Figure 5 shows a schematic diagram of a successful capture. This successful capture condition can be expressed by the following equation:

$$P_a(t) = \frac{1}{N}\sum_{i=1}^{N} P_i(t) \quad (5)$$
$$R_{ai}(t) = |P_i(t) - P_a(t)| \leq r \quad (6)$$

Where r is the encirclement radius. Therefore, the key to the multi-robot encirclement problem lies in how to plan the paths of the robots so that they can approach the target as quickly as possible and avoid conflicts between robots.

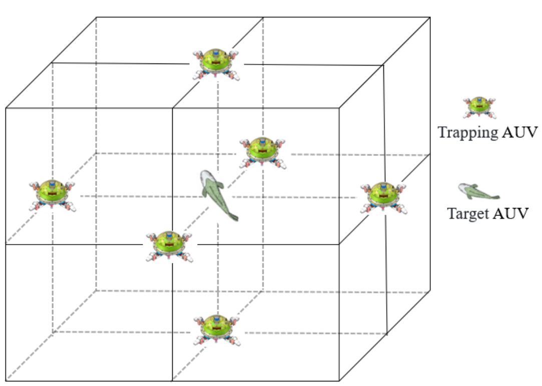


Fig. 5 Illustration of successful trapping in a 3D environment

III. SOLUTION

In this section, a collaborative encirclement strategy is proposed for a multi-water-land amphibious spherical robot system, to achieve efficient target encirclement by the amphibious spherical robots in a three-dimensional underwater static obstacle environment.

A. Dynamic alliance generation based on "encirclement point occupancy"

This paper proposes a dynamic "encirclement point occupancy" algorithm, which is used to assign encirclement points to capture AUVs when capturing an escaping target, in order to maximize the encirclement and control of the target. The implementation principle of the algorithm is based on the shortest total travel time of the encircling AUVs.

Figure 6 shows the implementation process of the "encirclement point occupancy" algorithm. This algorithm ensures that each AUV is assigned to the best encirclement point and solves the problem of duplicate assignment from the root cause through the one-to-one correspondence between AUVs and encirclement points.

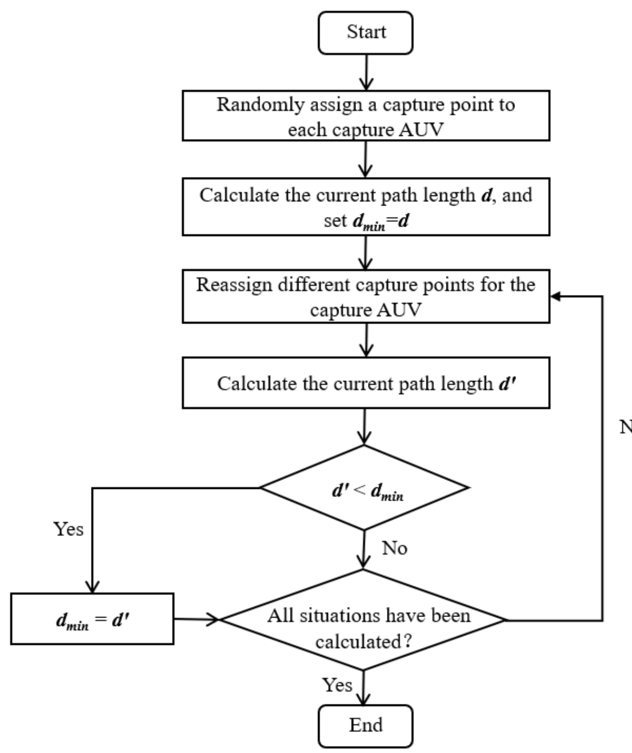


Fig. 6 Flowchart of encirclement point allocation strategy algorithm

B. Path planning based on improved wolf pack algorithm

The Improved Wolf Pack Algorithm (IWPA) is an optimization algorithm based on the behavior of wolves in nature. It solves optimization problems by simulating the hunting behavior of wolf packs. In this paper, we propose using the IWPA algorithm to solve the path planning problem of spherical underwater robots by introducing hunting, dispersal, and migration behaviors from the hunting process of wolf packs. Hunting behavior refers to the behavior of wolves searching for prey within their territory, dispersal behavior refers to the behavior of wolves randomly wandering, and migration behavior refers to the behavior of wolves moving to other areas to avoid local optima. There are many related introductions on the behavior of wolf pack hunting [13-16], and this paper will not repeat them. Below we will focus on introducing our innovation points.

The objective function of IWPA is the shortest path length for the underwater robot to reach the trapping point and avoid obstacles. Therefore, this paper introduces an improved heuristic function in the search part of the IWPA algorithm to effectively improve the search efficiency of the algorithm and optimize the planned path. The designed heuristic function is as follows:

$$\hbar_{i,j} = w_1 * dist(i,j) + w_2 * \min(dist(i,o)) \quad (7)$$

This function considers both the distance between the current position and the target point and the distance between obstacles comprehensively. Here, i and j represent the coordinates of the current state and the target point, o represents the set of all obstacles, $dist(i,j)$ represents the Euclidean distance between i and j, and $\min(dist(i,o))$ represents the distance between i and the nearest obstacle, w_1 and w_2 are distance weighting parameters.

The implementation process of this algorithm is shown in Figure 7. By executing the algorithm process, the optimal path for trapping AUV to the assigned trapping point can be realized. This path can avoid obstacles, has the shortest length, and takes less time.

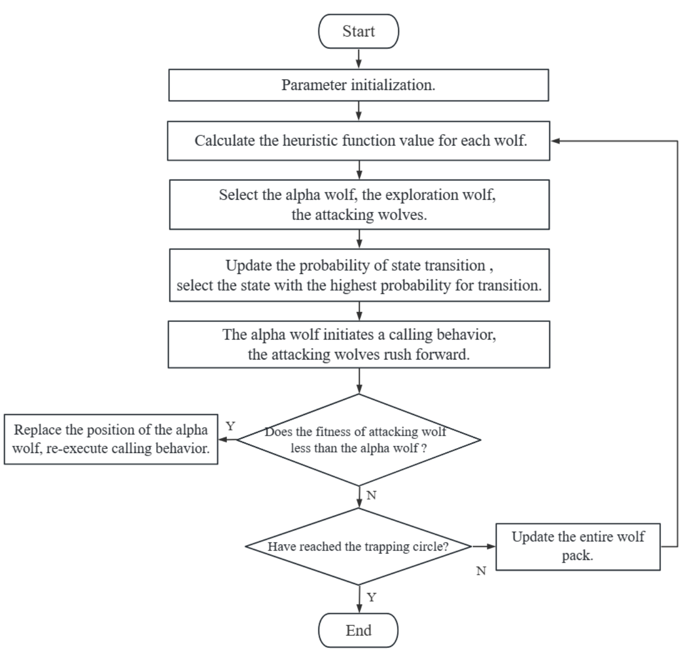


Fig. 7 Flowchart for implementing the wolf pack algorithm for path planning

IV. Simulation and Results

A. Building the environment model

In this paper, the terrain mapping method is used to model the 3D underwater environment, where six custom peaks are used to simulate underwater obstacles. The center point, area, and height of the peaks can be manually set. The size of the underwater environment is 200*200*200 cm^3.

In order to make the simulation effect closer to the actual situation, this paper proportionally reduces the size of the spherical robot to $5cm$. When performing obstacle collision detection, the volume of the robot is fully considered rather than simply ignoring the volume as a point. Figure 8 shows the effect of the built environment model.

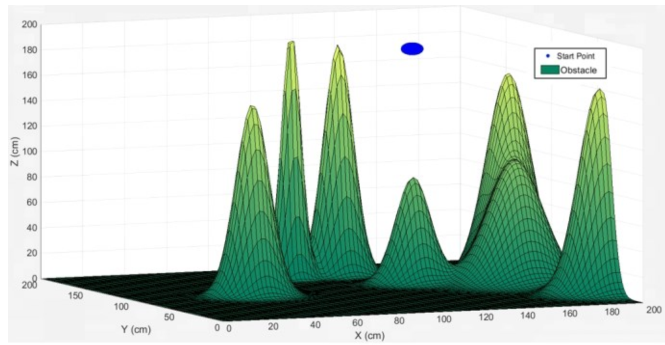


Fig. 8 Schematic diagram of the environment model

B. Trap point allocation based on the "trap point occupancy" algorithm

We let the program randomly generate the coordinates of six trap robots, and set the coordinate of the target point (i.e., the initial position of the escaping robot) to (50, 50, 80). According to the implementation principle of the trap point occupancy algorithm described in Part II, the six positions around the escaping robot are assigned to the six trap robots, while ensuring the shortest total path. Figure 9 shows the effect of the trap point occupancy algorithm's position assignment.

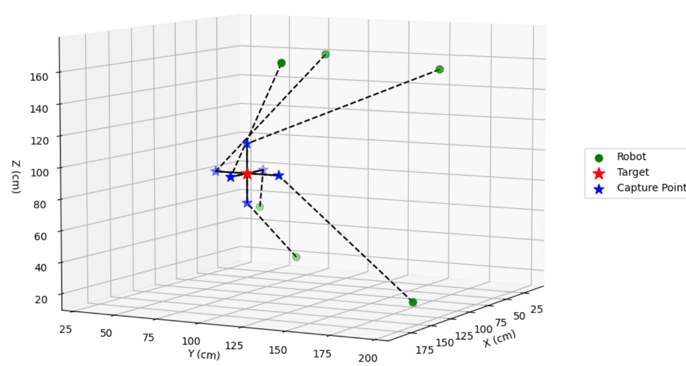
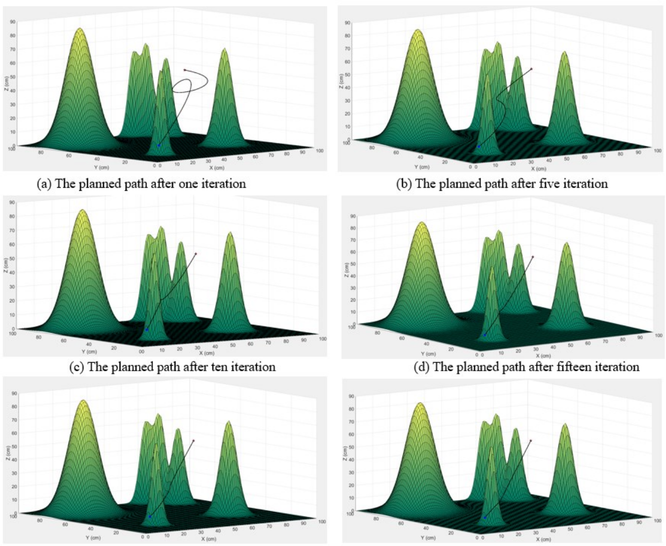


Fig. 9 The effect of the trap point occupancy algorithm's position allocation

As shown in the figure, the circles represent the positions of the trap robots, and the red pentagram in the middle represents the position of the escaping robot. The six positions around the escaping robot, including front, back, left, right, up, and down, are the trap points that are at a distance of radius from the target point. In this article, the trapping radius is set to $20cm$.

C. Path planning situation based on "improved wolf pack algorithm"

Firstly, we take one robot and one target location as an example to verify the path planning effectiveness of the "improved wolf pack algorithm" proposed in this paper.



(a) The planned path after one iteration (b) The planned path after five iteration

(c) The planned path after ten iteration (d) The planned path after fifteen iteration

(e) The planned path after twenty iteration (f) The planned path after twenty-five iteration

Fig. 10 The path planning effect of the improved wolf pack algorithm

Figures (a) - (f) show the iterations of the algorithm after 1, 5, 10, 15, 20, and 25 iterations, respectively. It can be seen that as the number of iterations increases, the path becomes shorter and shorter. After twenty iterations, the path change is very small, and it can be considered that the shortest path has been basically obtained and tends to be optimal. In addition, this article designs an obstacle (i.e. a mountain peak) on the straight line connecting the starting point and the target point. From the above results, it is evident that the algorithm perfectly avoids obstacles at every step of the path.

D. Performance assessment of the algorithm

In order to evaluate the performance of the algorithm, we compared the improved wolf pack algorithm proposed in this article with the traditional Wolf Pack Algorithm (WPA) [17,18] and two classic swarm intelligence algorithms, Particle Swarm Optimization (PSO) [19,20] and Ant Colony Optimization (ACO) [19,21]. The maximum iteration times for the algorithm were set to 100 times (at which the optimal path could be reached), and the starting point, ending point, and obstacle positions of the paths were consistent.

To make the comparison results more clear, we conducted a quantitative analysis of the path planning performance from two aspects: path length and algorithm running time. Table 1 shows the comparative results of the three algorithms.

TABLE I
COMPARISON OF IMPROVED WOLF PACK ALGORITHM WITH OTHER SWARM INTELLIGENCE ALGORITHMS

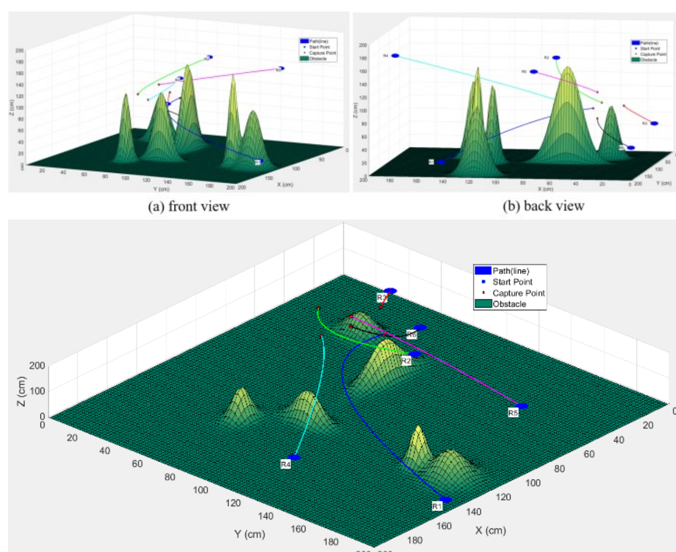
Algorithm	IWPA	WPA	PSO	ACO
Path length (cm)	102.33	142.42	102.72	155.44
Execution time (times / s)	0.63	0.54	0.82	1.14

From the data in Table 1, it can be seen that the length of the UAV's trajectory planned by WPA, PSO, and ACO algorithms are 142.42 cm, 102.72 cm, and 155.44 cm respectively, while the length of the UAV's trajectory planned by the IWPA algorithm is 102.33 cm. Thus, WPA and ACO algorithms got stuck in local optimal trajectories during the search process. Although there is not a notable difference between the travel path length planned by the IWPA algorithm and the PSO algorithm, the IWPA algorithm has a relatively shorter average running time.

The comparative results show that the proposed IWPA algorithm in this article has superior performance and can effectively solve the path planning problem of small spherical robots in three-dimensional underwater environments.

E. Overall trapping situation

The execution result of the "trap point occupation algorithm", i.e., the allocation of AUV and trap points, is input as initial information to the path planning module. An improved wolf pack algorithm is used to plan the movement path of the AUV for trapping based on this information, and the entire three-dimensional underwater environment trapping process is realized. The following is a schematic diagram of the final trapping situation.



(a) front view (b) back view

(c) top view

Fig.11 Trapping situation

As can be seen from robot R1 in Figure (a) and robot R6 in Figure (b), when there is an obstacle on the line connecting the starting point and the target point, the path planned by the algorithm will bend to avoid the obstacle, ensuring good obstacle avoidance effect. It can be seen from robots R4, R5 in Figure (a) and robot R3 in Figure (b) that when there is no obstacle on the line connecting the starting point and the target point, the path planned by the algorithm is close to a straight line, ensuring optimal path planning in an obstacle-free environment.

From this top view, it can be seen that the designed path takes into account the size information of the robot when avoiding obstacles, and the edges of the spherical robot will not touch the obstacles during the path planning process.

IV. CONCLUSION

The underwater robot encirclement system requires mutual cooperation between multiple robots to achieve a successful encirclement. In this article, the "encirclement point occupancy" algorithm is designed for small robot sensor limitations and the complexity of the three-dimensional underwater environment. This algorithm allocates the six encirclement points around the escaping robot to different encircling robots, and is able to minimize the total travel distance. Based on the allocated encirclement points, we improved the wolf pack algorithm to plan the path for the robots to reach the encirclement positions. The theoretical analysis and simulation experiments prove that our proposed algorithm can achieve a good encirclement effect on the escaping AUV, and has good performance.

Based on the above analysis, our proposed solution can efficiently achieve the robot encirclement task in a three-dimensional underwater static obstacle environment. In future work, we will continue to improve the algorithm to achieve the encirclement task in a dynamic obstacle environment.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (62273042, 61773064).

REFERENCES

[1] Z. Chen, D. Zhu, "Multi-AUV Cooperative Hunting Control with Improved Glasius Bio-inspired Neural Network," *The Journal of Navigation*, vol. 72, no.2, pp.759-776, 2019.

[2] S. Guo, S. Pan, X. Li, L. Shi, P. Zhang, P. Guo and Y. He, "A system on chip-based real-time tracking system for amphibious spherical robots," *International Journal of Advanced Robotic Systems*, Jul. 2017, pp. 1-19.

[3] H. Xing, S. Guo, L. Shi, et al. "A Novel Small-scale Turtle-inspired Amphibious Spherical Robot," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS*), Nov. 2019, pp. 1702-1707.

[4] L. Shi, S. Guo and K. Asaka, "A novel butterfly-inspired underwater microrobot with pectoral fins," *IEEE International Conference on Mechatronics and Automation*, Aug. 2011, pp. 853-858.

[5] L. Shi, S. Guo and K. Asaka, "A novel multifunctional underwater microrobot," *IEEE International Conference on Robotics and Biomimetics*, Dec. 2010, pp. 873-878.

[6] X. Jiang, D. Zhu, et al, "Multi-AUV Encirclement Path Planning Algorithm in 3D Underwater Environment," *Modern Electronics Technique*, vol. 44, no. 11, pp. 155-159, 2021.

[7] X. Jiang, W. Gan, "A new multi-AUV underwater trapping path planning algorithm," *Computer Simulation*, vol. 38, no. 9, pp. 376-380+428, 2021.

[8] J. Ni, S. Yang, "Bio-inspired neural network for real-time cooperative hunting by multi - robots in unknown environments," *IEEE transactions on neural networks*, Jan. 2011, pp. 2062-2077.

[9] F. Yu, H. Shang, Q. Zhu, et al, "An efficient RRT-based motion planning algorithm for autonomous underwater vehicles under cylindrical sampling constraints," *Autonomous Robots*, vol. 47, pp. 281–297, 2023.

[10] S. Chen, H. Chen, et al, "Multi-AUV roundup strategy based on Bug2 and angle priority," *Automation and Information Engineering*, vol. 42, no. 1, pp. 18-21, 2021.

[11] H. Yin, S. Guo, M. Liu, A. Li and M. Zhou, "Collaborative Hunting Strategy for Multi-Amphibious Spherical Robots in Obstacle Environments," *IEEE International Conference on Mechatronics and Automation (ICMA)*, Aug. 2022, pp. 727-732.

[12] X. Cao, C. Sun, "Multi-AUV cooperative target hunting based on improved potential field in underwater environment," *Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, Jul. 2018, pp.118-122.

[13] Y. Liu, W. Li, et al, "UAV trajectory planning based on wolf pack algorithm," *Journal of System Simulation*, vol. 27, no. 8, pp. 1838-1843, 2015.

[14] Z. Liu, L. He, et al, "Mobile robot path planning using improved grey wolf algorithm," *Journal of Xi'an Jiaotong University*, vol. 56, no. 10, pp. 49-60, 2022.

[15] Y. Liu, H. Zhu, W. Fang, "Unmanned aerial vehicle path planning using improved grey wolf algorithm," *Optics and Control*, pp. 1-8, 2023

[16] M. Cao, W. Zhao, Z. Chen, "Robot path planning based on the fusion of particle swarm optimization algorithm and improved grey wolf optimization algorithm," *Journal of System Simulation*, pp.1-9, 2023.

[17] X. Liu, Y. Tian, Y. Tian, "*A Review of Swarm Intelligence Algorithms. Information and Computers (Theoretical Edition)*," vol. 33, no. 24, pp. 63-69, 2021,

[18] M. Cao, W. Zhao, Z. Chen, "Robot Path Planning by Combining Particle Swarm Optimization Algorithm with Improved Grey Wolf Algorithm," *Journal of System Simulation*, pp.1-9, 2023.

[19] H. Wang, L. Wang, "Review of Path Planning Algorithms for Robots," *Journal of Guilin University of Technology*, pp.1-15, 2023.

[20] S. Guo, X. Sun, "Mobile Robot Path Planning Based on Improved Particle Swarm Optimization Algorithm," *Electronic Measurement Technology*, vol. 42, no. 3, pp. 54-58, 2019.

[21] X. He, Z. Li, B. Li, et al, "A Study on Mobile Robot Path Planning Based on Improved Ant Colony Optimization Algorithm," *Modern Manufacturing Engineering*, vol. 509, no. 2, pp. 36-43, 2023.