

Underwater Robot Path Planning Based on Improved Ant Colony Algorithm

Xiaolin Luan¹

*College of Information Science And Engineering
Ocean University of China
Qingdao, 266100, China*

Yuting Sun²

*Institute of Oceanographic Instrumentation
Shandong Academy of Sciences
Qingdao, 266100 China*

Feixiang Gong³, **Zhiqing Wei**, **Bo Yin**, **Yanpin Cong**

*College of Information Science And Engineering
Ocean University of China
Qingdao, 266100, China
E-mail: gongfeixiang@gmail.com*

MengDie Xu

*School of Government Management, Beijing normal university,
Beijing, 100875 China*

Against the background that an efficient modeling method is much needed to describe the 3D of Marine environment, three-dimensional modeling serves as a precondition for the path planning. This paper introduces a method for modeling three-dimensional with Ant Colony Algorithm which is characterized by strong robustness and lower selection criteria for the initial line. In addition, the Ant colony algorithm during its operation process does not need manual adjustment, and its parameters are relatively small. Simulations and experiments in this paper show that Ant colony algorithm has a good performance in modeling three-dimensional path planning, and the path is shorter than other alternatives. It proves that the method is effective both in establishing the corresponding data space and combining with the data for the path planning.

*CENet2015
12-13 September 2015
Shanghai, China*

¹Speaker

²This research is supported by the Qingdao innovation and entrepreneurship leading talent project (13-cx-2), Qingdao strategic industry development project(13-4-1-15-HY) and Shandong province science and technology project (2013GHY11519).

³Corresponding Author

1. Introduction

As the science and technology gains further momentum, the marine exploration attracts wide attention of scholars and experts in China and beyond. To improve the efficiency of marine exploration, an effective modeling method is much needed to describe the 3D of Marine environment. The 3D spatial modeling is a computing method, which puts the real three-dimensional space into a coordinate form in a certain way. Three-dimensional modeling, as a precondition of the three-dimensional path planning, affects the efficiency of the path planning to a great degree.

Over the past years, some algorithms have been adopted in an attempt to solve the problem of path planning in three dimensional space:

There are two sub-problems in the AUV(Autonomous Underwater Vehicle) global path planning, the environment modeling and its path searching algorithm corresponding to the modeling. The modeling methods mainly include visibility graph, grid method, free space method and topological method, which are mainly applied in two-dimensional space. To overcome the shortcomings brought by the local optimum of the traditional algorithms, that is, A* searching algorithm and artificial potential field method, experts in recent years have introduced the genetic algorithm [1,2], artificial neural network algorithm[3], and ant colony algorithm to work out the path planning. Nevertheless, there exist such weak points as more undesirable iterations, less time efficiency and lower precision in those mentioned algorithms. The paper presents an improved ant colony algorithm and advances the efficiency of pheromone expression and update [4]. The results of stimulation experiment verify the suitability and effectiveness of the proposed algorithm.

2. Environment Modeling

The 3D spatial modeling is a computing method, which displays the real three-dimensional space in a coordinate form. Three-dimensional modeling, as a precondition of the three-dimensional path planning, affects the efficiency of the path planning to a great degree [5].

The working area of robots is limited within the coordinate system O-XYZ where the O1, O2, O3 respectively denote the spherical obstacles whose radiuses are already known. Thus, the AUV global path planning aims to find the shortest path from the starting point to the goal while keeping away from the three obstacles.

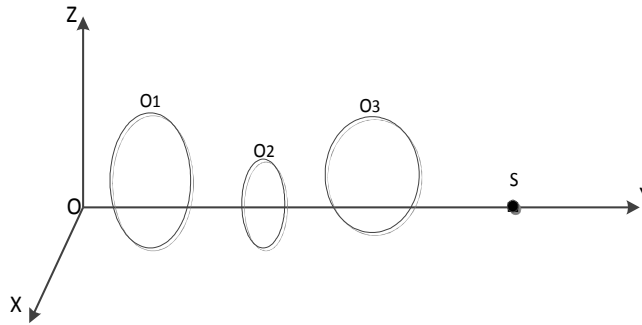


Figure 1: Obstacles in the coordinate system O-XYZ

It is assumed that the original point O is the starting point of the mobile robot and the S is the goal, and the line OS is on the axis Y. To generate a cube ABCD-EFGH in the coordinate system O-XYZ, we set the plane ABCD as a 21×21 square which is located on the plane X-OZ. The lines AB and BC are parallel to the axes X and Y respectively, with origin O as the center point of plane ABCD. In addition, the value of the line AE is h. The figure 2 illustrates that we divide the space ABCD-EFGH equally into $n+1$ portion at the direction of OS, connecting each

divide to form a plane parallel to ABCD. In this way, there are N planes, that is, Π_j ($j = 0, 1, 2 \dots n$).

As figure 3 shows that we divide any plane Π_j equally into m portion at the directions of $A'B'$ and $A'D'$ respectively, which generates grids with a total number of $m \times m$ and S^* as a set of points that are scattered in the space ABCD-EFGH. Figure 2 and Figure 3 present the robot's working area after the division of space.

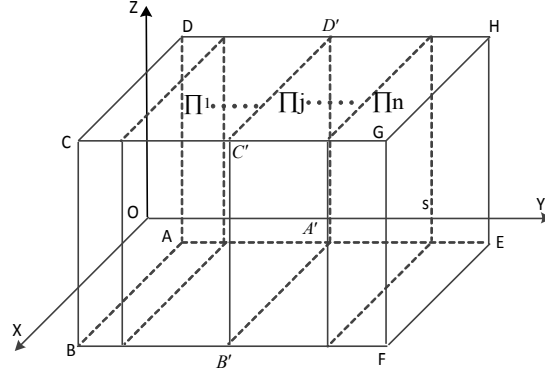


Figure 2: Division of space

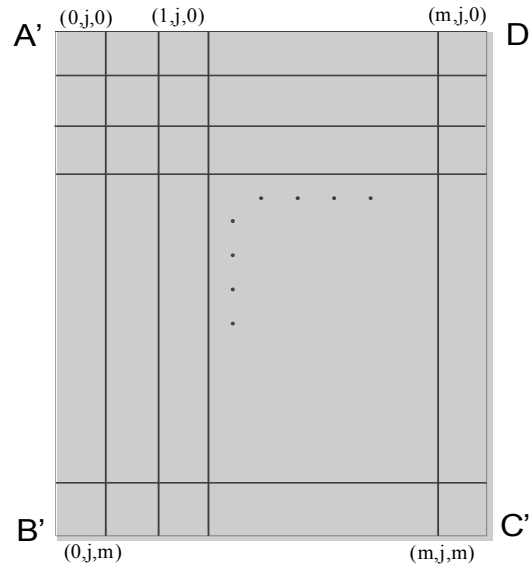


Figure 3: Division of a random plane Π_j

Assume any discrete node is denoted by $P(i, j, k)$, where $i = \{0, 1, 2 \dots m\}$, $j = \{0, 1, 2 \dots n\}$, $k = \{0, 1, 2 \dots m\}$. the point $P(x, y, z)$ in the coordinate system can be expressed as:

$$x = -l + \frac{2l \times i}{m} \quad y = \frac{j \times h}{n} \quad z = -l + \frac{2l \times k}{m} \quad (2.1)$$

In the coordinate system O-XYZ, the robot starts from the point O, arrives at the point $P(i_1, j_1, k_1)$, $i_1, k_1 \in \{0, 1 \dots m\}$, $j_1 \in \{0, 1 \dots n\}$ on the plane Π_1 , then continues to go from $P(i_1, j_1, k_1)$ and reach the point $P(i_2, j_2, k_2)$ on the plane Π_2 , ... , finally arrives at the

$P(i_n, j_n, k_n)$ of the plane Π_n . As a result, we connect the points of $P(i_n, j_n, k_n)$ and the goal S to generate a path between origin O and the goal S.

$$O \rightarrow P_1(i_1, j_1, k_1) \rightarrow P_2(i_2, j_2, k_2) \cdots \rightarrow P_N(i_n, j_n, k_n) .$$

To make a clear description, the definitions are presented as following:

Definition 1: If the line $P_a(i_a, j_a, k_a)P_{a+1}(i_{a+1}, j_{a+1}, k_{a+1})$ does not intersect any obstacles, any point $P_{a+1}(i_{a+1}, j_{a+1}, k_{a+1})$ of the plane Π_{a+1} will be saved in a form that is known as *allowed* $a(i_a, j_a, k_a)$. By this way, all of the points that are allowed to reach by $P_a(i_a, j_a, k_a)$ are collected in the form *allowed* $a(i_a, j_a, k_a)$.

Definition 2: the distance of a straight line between P_a and P_{a+1} of any two adjacent planes Π_a and Π_{a+1} is written as $d(P_a, P_{a+1})$. Its computational formula is presented in formula (2.2).

$$d(P_a, P_{a+1}) = \sqrt{(x_{a+1} - x_a)^2 + (y_{a+1} - y_a)^2 + (z_{a+1} - z_a)^2} \quad (2.2)$$

Based on the formula 2, the issue of the optimal path planning of mobile robots boils down to find out the shortest path of sequence of points, as shown in formula (2.3)

$$\begin{aligned} L_{os} &= \sum_{a=0}^{n+1} \sqrt{(x_{a+1} - x_a)^2 + (y_{a+1} - y_a)^2 + (z_{a+1} - z_a)^2} \\ &= \sum_{a=0}^{n+1} \sqrt{(x_{a+1} - x_a)^2 + \left(\frac{h}{n+1}\right)^2 + (z_{a+1} - z_a)^2} \end{aligned} \quad (2.3)$$

3. Solution Process by Ant Colony Algorithm

3.1 Pheromone

The pheromone is the carrier of the past information and directly affects the global convergence and solution efficiency of ant colony algorithm.. During the process of path planning, the adjacent nodes are connected as a pheromone carrier, which is only effective in addressing the questions with less possible solutions. Given that the storage carrier of pheromone is set on the connection of discrete points, the space complexity is too heavy because there are a large number of nodes in the environment modeling of three-dimensional space. According to the mentioned partition method, there will be m^2 points on a plane of

Π_j , in which any points can be connected with the adjacent plane Π_{j+1} in m^2 possible ways. So, for any two planes, there exist m^4 ways of connection, as shown in Figure 4. Suppose that the line between each discrete point, such as $P_j(1, m)P_{j+1}(1, 1)$, serves as storage carrier of pheromone, the space complexity will be increased dramatically as m increases.

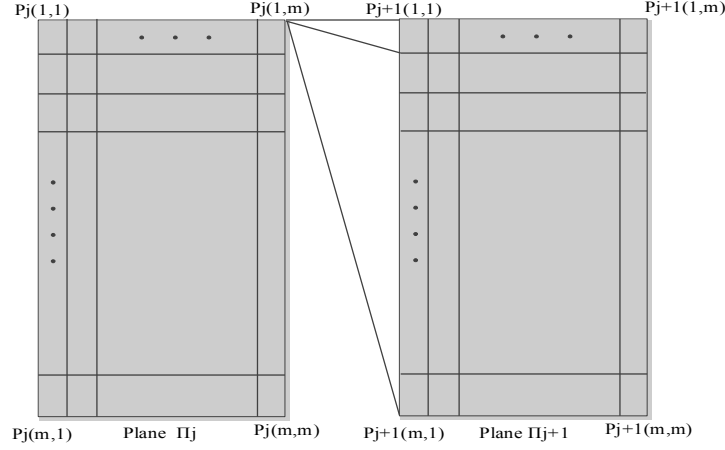


Figure 4: Sketch Map of Traditional Pheromone

Thus, this chapter employs the method of activity on node, in which the pheromone is stored on the discrete points of the environment modeling. For example, for the point $P_j(1,1)$, there are $2m^2$ storage carriers in two adjacent planes, compared to the method connecting points in two adjacent planes as carriers, the storage space is decreased. The value of pheromone corresponding to each point indicates how attractive a discrete point is to ants.

3.2 Transfer Rule

The transfer rule makes a good use of information on new path and prior knowledge. For ant colony algorithm, ants choose the path on the basis of probability. The various decision-making rules adopted by ants are known as the pseudo-random proportional rule, which not only takes advantage of heuristic information on the distance between cities and prior knowledge on pheromone stored but also conducts experiments with specific research interest.

It is assumed that there are m ants on the starting point. For each step of the path planning, the ant K , according to the rule (3.1), decides its next move.

$$P_{a+1} = \begin{cases} \arg \max_{(a+1, \eta, \tau) \in \text{allowed}_a} [\tau_{a+1} \bullet \eta_{a.a+1}] & q \leq q_0 \\ J & q > q_0 \end{cases} \quad (3.1)$$

In the rule 4, j denotes a random variable resulted from rule (3.2). To make the searching results more comprehensive, q and q_0 are set as a constant number and a random number respectively at the range of 0 and 1.

For the ant on the arbitrary point $P_a(i_a, j_a, k_a)$ of a plane $\prod_a (a=1, 2, 3 \dots n)$ the probability of $P_{a+1}(i_{a+1}, j_{a+1}, k_{a+1})$ on the plane is \prod_{a+1} calculated as rule (3.2) shows:

$$P_{a.a+1} = \begin{cases} \frac{\tau_{a+1} (\eta_{a.a+1})^\beta}{\tau_{a+1} (\eta_{a.a+1})^\beta} & \text{Feasible point} \\ 0 & \text{otherwise} \end{cases} \quad (3.2)$$

where τ_{a+1} denotes the pheromone concentration of $P_{a+1}(i_{a+1}, j_{a+1}, k_{a+1})$ on the plane of

$\eta_{a,a+1}$, $\eta_{a,a+1}$ is a heuristic function that serves as an important part of path planning algorithm and is used to represent future information. A large number of experiment results show that the heuristic function plays an essential role in helping ant colony algorithm obtain the global optimal path within a reasonable time. Based on what is mentioned above, the distance between any two adjacent feasible points meets the standards of this paper, that is, to pursue the most secure and the shortest path. The heuristic function is expressed as rule (3.3):

$$\eta_{a,a+1} = \frac{1}{d(p_a, p_{a+1})} \quad (3.3)$$

β is a heuristic factor, reflecting how important the heuristic information is to the moving ants in selecting its path. For the mobile robots path planning, there are so many points ants can pass by that it's almost unlikely for each and every point to get pheromone. Thus, the optimal results of the path may lead to the local solution. The simulation experiment shows that a good answer is hard to get after setting β as a constant number. The value of β in this paper is calculated by the rule (3.4). (t is critical moment)

$$\beta = \begin{cases} \frac{(4t-2q)}{t}, & 0 \leq q < t \\ 2, & t \leq q \end{cases} \quad (3.4)$$

3.3 Updating Rules of Pheromone Concentration

As the pheromone concentration of nodes on the best tour will be updated after the completion of each iteration, the difference of pheromone concentration between the best ant tour and the worst one will become bigger, which encourages ants to search for the optimal paths in the vicinity of the best tour and leads to the higher efficiency of the path searching algorithm.

After the completion of iteration, the pheromone concentration of the frequently visited paths needs to be updated in case that the heuristic information would be diminished by the decay information. The updating rules in this paper lay the emphasis on both the global pheromone and the local pheromone. The formula (3.5) shows how the global pheromone update works:

$$\tau_{ijk} = (1-\rho)\tau_{ijk} + \rho\Delta\tau_{ijk} \quad (3.5)$$

$$\Delta\tau_{ijk} = \sum_{t=1}^m \Delta\tau_{ijk}^t \quad (3.6)$$

$$\Delta\tau_{ijk} = \begin{cases} 1/L_t & \text{When the } T \text{th ant passes through} \\ & \text{the point during the iteration} \\ 0 & \text{otherwise} \end{cases} \quad (3.7)$$

In this formula, ρ denotes the pheromone evaluation and $1-\rho$ represents the pheromone decay parameter. To avoid the continual accumulation of pheromone, we set $0 < \rho < 1$. In addition, $\Delta\tau_{ijk}$ is the increasing amount of pheromone of the point (i, j, k) in an iteration. The local pheromone update in this paper employs the updating rules of path points. Every time an ant selects a path point, the rules will be applied to update the pheromone concentration immediately, as shown in the formula (3.8):

$$\Delta\tau_{ijk} = (1 - \mu)\tau_{ijk} + \mu\tau_0 \quad (3.9)$$

The ant colony system algorithm serves as a kind of multi agent and bionic optimization algorithm, in which the cooperation and coordination among the multiple agents significantly enhance the convergence performance and solution efficiency. In the process of searching the best tour, it is likely for two ants with the opposite searching directions meet with each other and bring about their searching information respectively. Then, by combining the information ants take along, a novel feasible path can be generated, as shown in formula (3.9):

$$L_{NEW} = L(k1) + L(k2) \quad (3.10)$$

when the ants $k1$ and $k2$ met with each other

If the new feasible path is more effective than the then optimal path, the then optimal path should be replaced and its pheromone needs to be updated so as to improve the convergence efficiency. The formula (3.10) explains how the updating rules of local pheromone work when ants meet.

$$\tau = (1 - \mu)\tau + \mu\tau \quad (3.11)$$

$$\Delta\tau = 1 / L_{NEW} \quad (3.12)$$

where μ is set between 0 and 1; τ_0 is a relatively small constant and represents the initial value of pheromone of each feasible point.

3.4 Algorithm Procedures

(1) To initialize each parameter, input the basic data, and set $N_s = 0$, where N_s symbolizes the step counter for the optimal path. Set $N_r = 0$, where N_r represents the number of paths gained from ant colony since after finishing the global update until the present time. Initialize the $tabu_a$, and set $allowed_a = 0$ and $direct_a = 0$.

(2) To set the number of ants on the starting point O is m, and add O into the $tabu$.

(3) For a random ant k, it starts from the node $P_a(i_a, j_a, k_a)$ as the center to the next node $P_{a+1}(i_{a+1}, j_{a+1}, k_{a+1})$ in accordance with rule (4) and rule (5).

(4) When each ant finished a move, the pheromone should be updated as per rule 11. As mentioned in step 3, the ants keep going until one of them reaches the goal S.

(5) To calculate the optimal path L' of the iteration and get the optimal paths we have found by the formula $L = \min\{L', L\}$. The update should be completed by the updating rule (8) of global pheromone.

(6) To tell whether or not ants met with each other. If they met, the formula 12 of the local pheromone is applied and go back to step 5.

4. Three-dimensional Environment Modeling

Ant Colony algorithm for searching is adopted in an area covering 21 by 21 kilometers all along to avoid the entire obstacle path. To address the problem, the deepest point within the height range of this region is set up as 0, so that the height of other points is defined accordingly. The start point and end point are denoted by (1, 10,800) and (21, 4, 1000) respectively [6.7]. At last, we apply the software MATLAB to stimulate the space as shown in **Figure 5**.

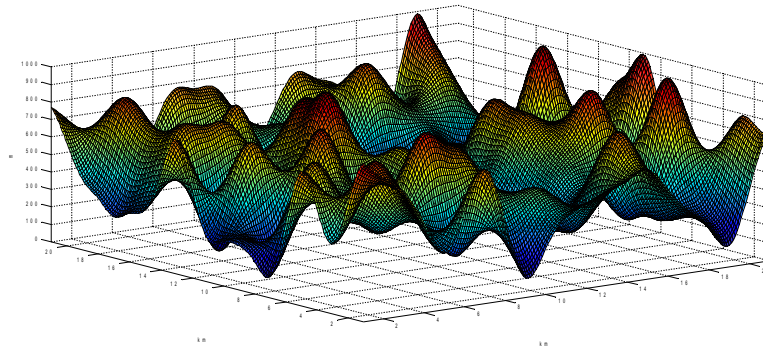


Figure 5 :The process of 3D environment modeling

We introduce the MATLAB simulation to 3D path planning of ant colony algorithm. The planning space extends for 21 by 21 kilometers, according to the third part, which is abstracted into a 21 by 21 by 21 kilometers planning space. Within the three dimension space, each node in x axis and y axis is at the distance of one kilometer, and the distance of each node in the z axis is 200 m. The three dimension coordinates of the starting point and goal point are [1, 10, 4] and [21, 4, 5], respectively. The basic settings of the algorithm: the number of underwater robots is 20, algorithm iteration is 400 times. The results of simulation experiments are as follows: Figure6 shows the trends of best individual fitness. Figure 7 demonstrate the simulation effect of basic ant colony algorithm. Figure 8 demonstrate the simulation effect of the improved ant colony algorithm.

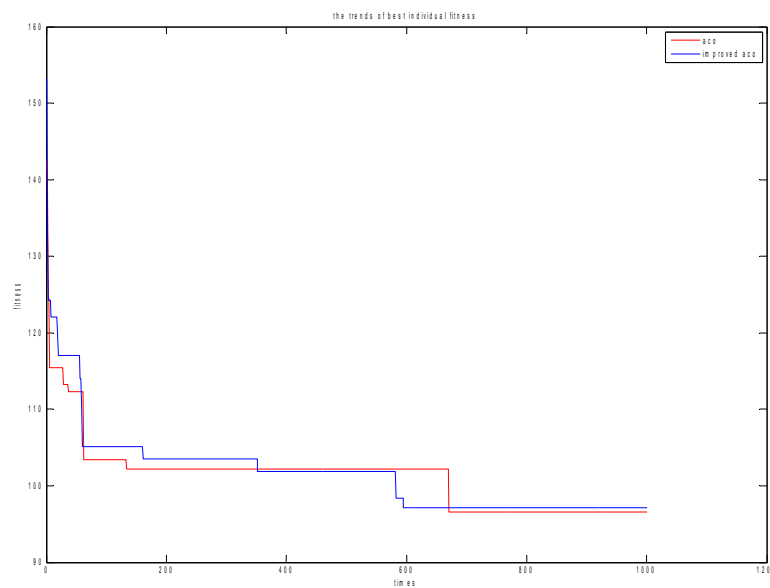


Figure 6: The trends of best individual fitness

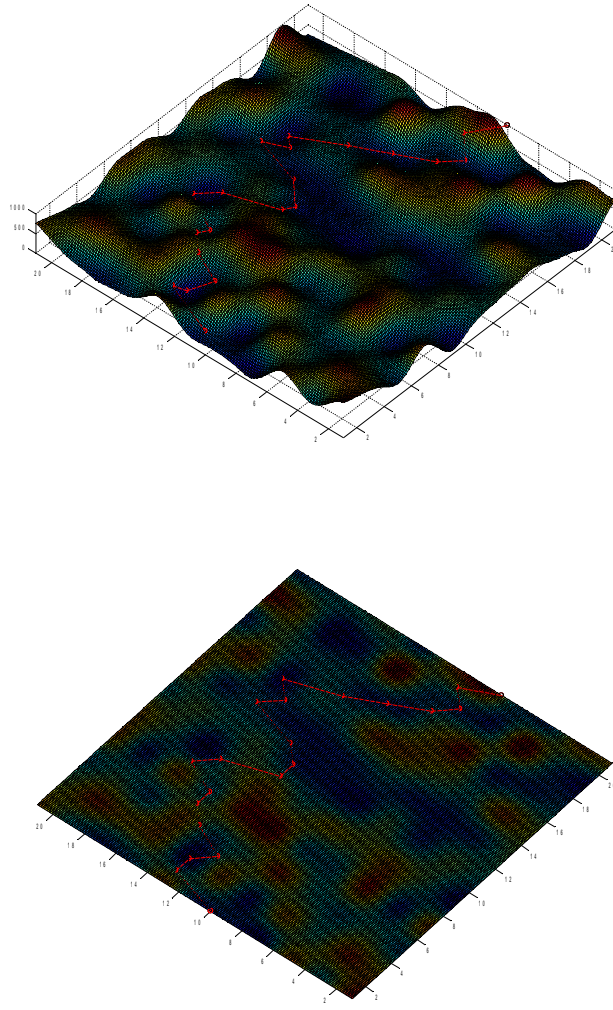
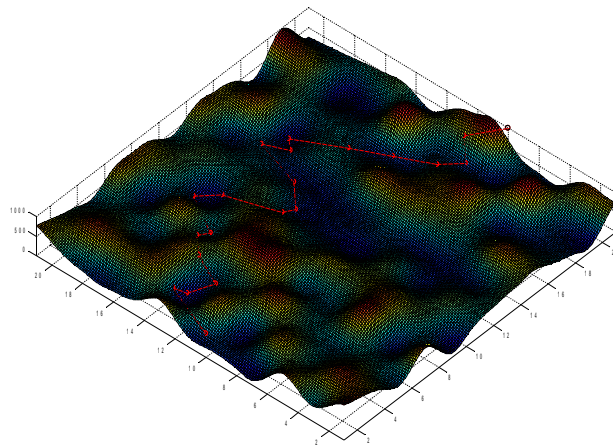


Figure 7: ant colony algorithm



POS (CENet2015) 001

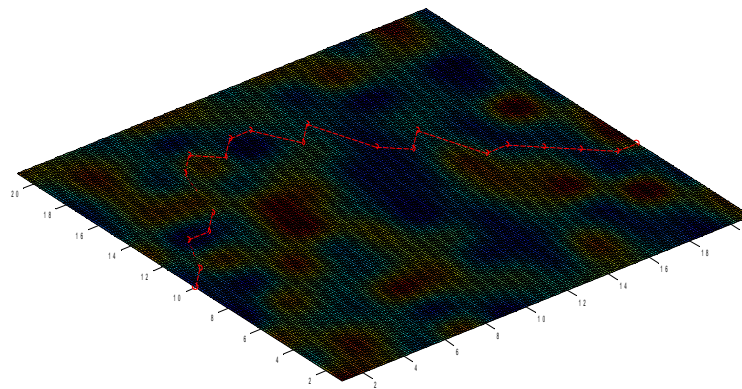


Figure 8: improved ant colony algorithm

The results of computer simulation experiment show that the algorithm proposed in this paper is more effective than the basic ant colony algorithm. The improved version of ant colony algorithm keeps appropriate distance away from the obstacles and meets the demand of the navigation route, while further enhancing the convergence speed compared to the original ACS algorithm.

5. Conclusion

This paper employs the improved ant colony algorithm to conduct research on the underwater robot path planning in the three dimensional space, and presents the environment modeling methods and optimal searching path algorithm in details. The results of computer simulation experiment verify the suitability and effectiveness of the proposed algorithm in helping the path planning in the three-dimensional space. However, the ant colony algorithm, as a sort of stereotype random searching algorithm, takes more time than expected during its operation. Therefore, more research efforts are still needed to explore how to enhance the operation speed and global search capability.

References

- [1] Y.L. Hao, J.J. Zhang. *AUV Path Planning in 3D Seabed Environment Using Genetic Algorithm*[J], Engineering Science , Vol.5 No.11,56-57,Nov. 2003 (In Chinese)
- [2] Q.J. Duan. *Research on Real Time Path Planning Method of Autonomous Underwater Vehicle*[D], Harbin Engineering University ,2007 (In Chinese)
- [3] K.P. Caroll, S.R. McClaran, E.L. Nelson, D.M. Barnett,D.K. Friesen, and G.N. Williams. *AUV Path Planning: An A* Approach to Path Planning width Consideration of Variable Vehicle Speeds and Multiple, Overlapping, Time-Dependent Exclusion Zones* [C].Proc. Symp. on AUV Underwater Vehicle Technology, Washington DC: IEEE, 1992: 79-84.
- [4] H. HU, X.S. Cai. *Path Planning of Robots in 3D Space Based on an Improved Ant Colony Algorithm*[D],College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, Jinhua, 2012 (In Chinese)
- [5] X.B. Hu, X.Y. Huang. *Path planning in 3-D space for robot based on ant colony algorithm*[J], Journal of Chongqing University, 27(8): 132-135,2004. (In Chinese)

- [6] L.Q. Liu, Y.T. Dai. *3D Space Path Planning of Complex Environmental Underwater Vehicle*[C], Proceedings of the 2009 International Joint Conference on Computational Sciences and Optimization, CSO 2009, v 2, p 204-209, 2009.
- [7] P .Meng,T.F. Ning. *Path Planning Based on Hybrid Adaptive Ant Colony Algorithm for AUV*[C], Proceedings - 11th International Symposium on Distributed Computing and Applications to Business, Engineering and Science, DCABES 2012, p 157-160, 2012,

POS (CENet2015) 001