Journal of Intelligent & Robotic Systems (2019) 94:237–249
https://doi.org/10.1007/s10846-018-0787-7

Complete Coverage Path Planning of Autonomous Underwater Vehicle Based on GBNN Algorithm

Daqi Zhu[1] · Chen Tian[1] · Bing Sun[1] · Chaomin Luo[2]

Received: 22 July 2016 / Accepted: 31 January 2018 / Published online: 10 February 2018

Abstract

For the shortcomings of biologically inspired neural network algorithm in the path planning of robots, such as high computational complexity, long path planning time etc Glasius Bio-inspired Neural Network (GBNN) algorithm is proposed to improve the algorithm, and applied to the complete coverage path planning of autonomous underwater vehicle (AUV). Firstly, the grid map is constructed by discretizing the two-dimensional underwater environment. Secondly, the corresponding dynamic neural network is built on the grid map. Finally, complete coverage path of AUV is planned based on the GBNN strategy and the path of AUV at the edge of obstacles is optimized by some typical path templates. The simulation results show that the AUV can completely cover the entire workspace and immediately escape from deadlocks without any waiting. Meanwhile, the efficiency of complete coverage path planning is high with short path planning time and low overlapping coverage rate by using the algorithm proposed in this paper.

Keywords Autonomous underwater vehicle (AUV) · Complete coverage path planning · GBNN algorithm · Neural network

1 Introduction

Compared with remotely operated vehicle (ROV), autonomous underwater vehicle's intelligence is higher, with broader range of underwater activities, and can go deep into the underwater space where the general submersible cannot reach to perform the specified tasks in the complex ocean environment [1]. At present, the complete coverage detecting underwater areas, searching and rescuing in the deep sea are the main tasks of the autonomous underwater vehicle (AUV), and the AUV must have the ability of complete coverage path planning when performs these tasks, therefore, the complete coverage path planning algorithm of the AUV is a hotspot and essential issue in the AUV researches. Generally, complete coverage path planning of the AUV can be described as follows: the AUV which starts from the initial point, sails along the shortest possible path or uses the shortest possible time to go through the entire underwater workspace except obstacles, which requires the AUV to avoid all the obstacles safely, and the overlapping paths should be as short as possible [2–4]. To date, the complete coverage path planning algorithm has been extensively studied, e.g., random-covering algorithm [5, 6], cellular decomposition algorithm [7–9] and neural network algorithm [10–15], etc.

Gage [5] proposed a random-covering algorithm, where the robot moves straight along a direction based on random movement strategy. At every fixed time or encountering an obstacle, the robot will backward rotate a random angle, and then continue to move along a straight line. This method does not need too many sensors and is quite simple, but it is inefficient and has high overlapping path rate. Otherwise, due to the reason that the strategy of path planning is too simple, the robot usually cannot escape from deadlocks in the complex terrain. Thus, this method is difficult to guarantee that the entire workspace can be covered [6].

✉ Daqi Zhu
zdq367@aliyun.com

Chen Tian
1099780850@qq.com

Bing Sun
hmsunbing@163.com

Chaomin Luo
luoch@udmercy.edu

[1] Laboratory of Underwater Vehicles and Intelligent Systems, Shanghai Maritime University, Haigang Avenue 1550, Pudong new district, Shanghai, 201306, China

[2] Department of Electrical and Computer Engineering, University of Detroit Mercy, Detroit, MI 48221, USA

Latombe [7] proposed the trapezoidal cellular decomposition algorithm, which can solve the low coverage rate of the random-covering algorithm. In this method, the workspace is decomposed into several non-overlapping trapezoidal cells according to the obstacle locations. The robot visits each cell based on back-and-forth motions, then moves to another cell by adjacency graphs until the entire workspace is covered. The method can guarantee the robot covers the entire region, but there are redundant movements between cells, so the high overlapping rate and low efficiency of path planning are caused. For this deficiency, Choset [8] proposed the boustrophedon cellular decomposition algorithm, which reduces the number of cells, thus the redundant paths between cells are decreased and the overlapping rate is lowered. However, the algorithms of [7] and [8] all require the prior knowledge of the workspace. For this, Acar [9] gave a sensor-based coverage method by improving the boustrophedon cellular decomposition algorithm, which ensures that the coverage tasks don't need any prior knowledge.

With the rapid development of artificial intelligence, it has been widely used in the path planning of the robot [10], e.g., artificial neural network algorithm [11, 12], ant colony algorithm [16, 17] and genetic algorithm [18, 19], etc. Among them, the artificial neural network algorithm gets more attention due to its high intelligence. However, most of the artificial neural network algorithms have shortcomings, such as long learning time and learning delay, thus, the real-time performance of path planning is difficult to be guaranteed. Therefore, a non-learning, adaptive biologically inspired neural network algorithm is proposed by Yang, and applied to the complete coverage path planning of the mobile robot [13, 14]. In this method, the two-dimensional grid map cells correspond to the neurons of neural network one-to-one, and the robot path is autonomously planned by the dynamic activity landscape of the neural network and the previous robot location. The method determines the external inputs of neurons through the state of two-dimensional grid map and directly calculates the neural activities without any learning procedures, and the robot can automatically avoid obstacles and escape deadlocks. On this basis, Yan [15] combines biologically inspired neural network algorithm with the D_S information fusion algorithm, not only builds a dynamic underwater environment map, but also solves the AUV complete coverage path planning in unknown environment.

Although the complete coverage path planning based on biologically inspired neural network algorithm has the advantages of non-learning, self-adaption, there are shortcomings such as the high algorithm complexity and the complicated calculation. When the robot is stuck in deadlock, it has to wait at its current location for a long time until the neural activity value of the deadlock is smaller than its neighboring locations, so that it can escape from the deadlock [13]. Thus, the efficiency of path planning is lower.

Aiming at above issues, inspired by two-layer feed forward neural network algorithm in [20], Glasius Bio-inspired Neural Network algorithm (GBNN) is proposed by improving the method of calculating neural activity in this paper, and applied to complete coverage path panning of the AUV. The proposed algorithm uses the difference equation to calculate the neural activity. Compared with the differential equation in [13], the calculation is simpler, and the real-time performance is better and also without any learning procedures. The AUV can quickly plan a collision-free complete coverage path and the path planning is efficient.

This paper is organized as follows. In Section 2, the complete coverage path planning method based on GBNN algorithm is presented. In Section 3, the simulation experiments for various situations are given. Finally and the conclusion is given in Section 4.

2 Complete Coverage path Planning of the AUV Based on GBNN Algorithm

In this section, the method of complete coverage path planning of the AUV based on GBNN algorithm is proposed. Firstly, the two-dimensional grid map is constructed by discretizing the underwater workspace of the AUV, and then a neural network corresponding to the grid map is built. Secondly, according to the state of the grid map, the neural network activity is updated constantly by GBNN algorithm. Finally, complete coverage path of the AUV is planned through the path planning strategy. In addition, when the AUV sails near some obstacles, its avoiding obstacles path is optimized by some typical path optimization templates.

2.1 Building Grid Map and Neural Network

In this paper, the grid map is built by discretizing the underwater environment, where the workplace of the AUV is decomposed into cells with the same size and shape. And each cell has two states: obstacle or free, as shown in Fig. 1a,

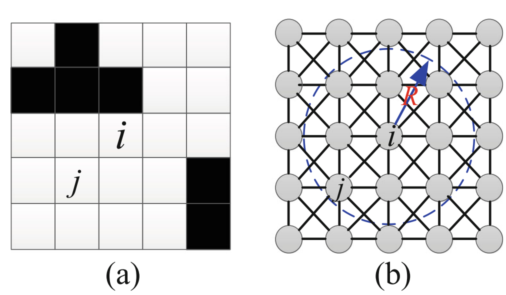


Fig. 1 The model of two-dimensional underwater environment. **a** The two-dimensional grid map. **b** The two-dimensional neural network

the black cells are obstacles, the white cells are free. Then, a two-dimensional neural network is built on the grid map, and each cell of the grid map corresponds to each neuron of the neural network, e.g., the ith cell of Fig. 1a corresponds to ith neuron of Fig. 1b. In Fig. 1b, the receptive field of the ith neuron is represented by a circle with a radius of R, and each neuron has lateral connections only to neighboring neurons within its receptive filed.

2.2 GBNN Algorithm

GBNN algorithm is a discrete-time Hopfield-type neural network, whose dynamic model is described as

$$x_i(t+1) = g\left(\sum_{j=1}^{M} W_{ij}[x_j(t)]^+ + I_i\right) \tag{1}$$

where the transfer function is chosen as

$$g(x) = \begin{cases} -1, x < 0 \\ \beta x, 0 \le x < 1, \beta > 0 \\ 1, x \ge 1 \end{cases} \tag{2}$$

where $x_i(t+1)$ is the neural activity of the ith neuron at $t+1$; $x_j(t)$ is the neural activity of the jth neuron at t, and the jth neuron is laterally connected with ith neuron. Otherwise, $[x_j(t)]^+ = max[x_j(t), 0]$, it represents that only the positive neural activities can influence other neurons and propagate globally, but the negative neural activities can't propagate outward and only have local effect; $M$is the number of neural connections of the ith neuron to its neighboring neurons within the receptive filed R; W_{ij} is the connection weight between the ith neuron and jth neuron and is defined as

$$W_{ij} = \begin{cases} e^{-\alpha|i-j|^2}, \ 0 < |i-j| \le R \\ 0, |i-j| > R \end{cases} \tag{3}$$

where $|i-j|$ is the Euclidean distance between ith neuron and jth neuron, α and R are all positive constants, and R is the radius of the receptive field shown in Fig. 1b, in general, $R = 2$ for the two-dimensional neural network, which means each neuron has only lateral connections to its eight neighboring neurons within receptive field. In addition, the connection weight is a scalar and is symmetric, i.e., $W_{ij} = W_{ji}$. I_i is the external input of the ith neuron, which is given as

$$I_i = \begin{cases} +E, & \text{if it is an uncovered area} \\ -E, & \text{if it is an obstacle area} \\ 0, & \text{if it is an covered area} \end{cases} \tag{4}$$

where E is the external excitatory input and $-E$ is the external inhibitory input. To guarantee that the neural activities of uncovered areas are maximum while the neural activities of obstacles areas are minimum, the external input E must be greater than the sum of excitatory inputs from the lateral neural connections, which means $E >> \sum_{j=1}^{M} W_{ij}[x_j(t)]^+$, in general, $E >> 1$, $\sum_{j=1}^{M} W_{ij}[x_j(t)]^+ \in [0, 1)$.

The neural activity is bounded in the finite interval $[-1, 1]$ by Eq. 2, which guarantees that the neural network is stable. Combining with Eqs. 1 and 2, if the ith neuron is uncovered, its external inputs include the external excitatory input: $I_i = +E$ and the sum of excitatory inputs from the lateral neural connections: $\sum_{j=1}^{M} W_{ij}[x_j(t)]^+$, and $\sum_{j=1}^{M} W_{ij}[x_j(t)]^+ + E >> 1$, therefore, the neural activity of the uncovered neuron is 1; If the ith neuron is obstacle, its external input is made of the external inhibitory input: $I_i = -E$ and the sum of excitatory inputs from the lateral neural connections: $\sum_{j=1}^{M} W_{ij}[x_j(t)]^+$, and $\sum_{j=1}^{M} W_{ij}[x_j(t)]^+ - E < 0$, which guarantees that the neural activity of the obstacle area is -1; If the ith neuron is covered, its external input: $I_i = 0$and the excitatory inputs only from the sum of excitatory inputs from the lateral neural connections, due to the $\sum_{j=1}^{M} W_{ij}[x_j(t)]^+ \in [0, 1)$, the neural activity of covered area is $\beta \sum_{j=1}^{M} W_{ij}[x_j(t)]^+ \in [0, 1)$, generally, $0 < \beta \le 1$. In conclusion, the neural activity dynamically changes according to the varying environment. The positive neural activity can propagate to the entire workspace through lateral connections among neurons, thus the uncovered areas can globally attract the AUV to visit. However, the negative neural activity can't propagate outward and only stays locally, so that the obstacle areas have only local effect to push the AUV away to avoid collisions. For the covered areas, the external excitation inputs of the corresponding neurons all become to be zeros, and they cannot attract the AUV to visit. Their excitatory inputs are only from the sum of incentive value transmitted from the side connection neurons. With the uncovered areas are gradually covered, the covered areas' excitatory inputs also become smaller and smaller, therefore, their neural activities will constantly decay to zero.

2.3 Strategy of complete coverage path selecting

Due to the limitation of power, when the AUV performs coverage tasks, it should sail the shortest path with the least revisited areas and make turns of sailing direction as fewer as possible to avoid consuming excessive energy. Thus, the previous AUV sailing direction must be taken into account in the complete coverage path planning. And, for a given current AUV location P_c, the next AUV location P_n is defined as [13]:

$$P_n \Leftarrow x_{p_n} = max\{x_k + cy_k, k = 1, 2, ..., m\} \tag{5}$$

where c is a positive constant; x_k is neural activity of the kth neuron, which is calculated by Eq. 1; m is the number of neighboring neurons of the P_cth neuron; $y_k$is a monotonically increasing function of the difference between the current and next AUV sailing directions and given as

$$y_k = 1 - \frac{\triangle\varphi_k}{\pi} \tag{6}$$

$$\begin{aligned} \triangle\varphi_k &= |\varphi_k - \varphi_c| = |a\tan 2(y_{p_k} - y_{p_c}, x_{p_k} - x_{p_c}) \\ &\quad -a\tan 2(y_{p_c} - y_{p_p}, x_{p_c} - x_{p_p})| \end{aligned} \tag{7}$$

where $\triangle\varphi_k$ is the turning angle between the current sailing direction and the next sailing direction; (x_{p_p}, y_{p_p}), (x_{p_c}, y_{p_c}) and (x_{p_k}, y_{p_k}) represent the previous location P_p coordinate, the current location P_c coordinate and the next possible location P_k coordinate, respectively.

As shown in Fig. 2, when the AUV sails straight, $\triangle\varphi_k = 0$, while the AUV sails backward, $\triangle\varphi_k = \pi$, thus, $\triangle\varphi_k \in [0, \pi]$, $y_k \in [0, 1]$. When the AUV reaches the next location from the current location, the next location becomes a new current location. Then the AUV will continue to select the new next location by Eq. 5 until the whole objective workspace is covered. If the selected next location is the same as the current location, the AUV will still stay the current location. In a word, the AUV can plan a complete coverage path according to the dynamic neural activity and the current position through Eq. 5.

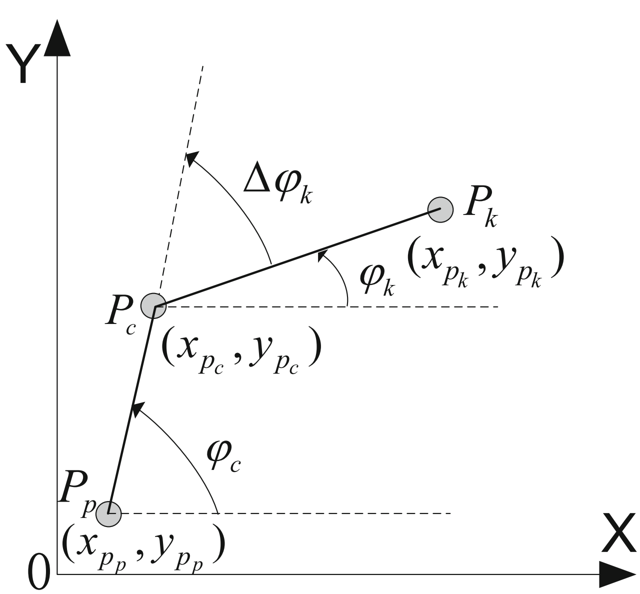


Fig. 2 The sailing direction of AUV

2.4 The Optimization of the Complete Coverage Path at the Edge of Obstacles

It is generally known that the underwater environment is very complex and there are many obstacles with arbitrary shapes. When the AUV encounters obstacles, its paths of avoiding obstacles planned by Eq. 5 may be cluttered and overlapped. Aiming this problem, on the basis of the complete coverage path optimization templates of cleaning robot in [21], and thinking that the AUV's sailing trend is from top to bottom, left to right, five kinds of the path templates are given in Fig. 3 to optimize the paths of the AUV at the edge of obstacles [15]. Therefore, during the AUV sailing, if the AUV finds obstacles around it, it will judge if the current position is matching some kind of path optimization template. If not matching, the AUV will still plan the path with obstacles avoidance by Eq. 5; if matching, the AUV will plan the avoiding obstacles path through the corresponding template of Fig. 3 to reduce the overlapping paths as far as possible.

As shown in Fig. 3, P_p, P_c is the previous AUV location, current location, respectively. Choosing P_c as the center, its eight neighboring cells are numbered clockwise from the bottom-left corner. In Fig. 3a, when the AUV sails upward, if the obstacles are above the AUV and the right cell 6 is uncovered, the cell 6 will be chosen as the next location, after the AUV reaches cell 6, if the cell 7 is uncovered, then it will be covered; In Fig. 3b, when the AUV sails downward, if the obstacles are below the AUV and right cell 6 is uncovered, the cell 6 will be covered firstly, then, if the cell 5 is uncovered, it will be chosen as the next AUV location; In Fig. 3c, when the AUV sails upward, if the obstacles are in the bottom-left and left cell 2 is uncovered, the cell 2 will be covered firstly; In Fig. 3d, when the AUV sails downward, if the obstacles are in the top-left and left cell 2 is uncovered, the cell 2 will be covered firstly; In Fig. 3e, when AUV sails from right to left, if the obstacles are in the bottom-right and the below cell 8 is uncovered, the cell 8 will be covered firstly by the AUV.

3 Simulation Studies

The simulation environment is 120× 120 m^2 two-dimensional underwater workspace, which is represented by a discretized grid map. The size of each cell is 4 m × 4 m, and the size of the grid map is 30 × 30 cells, thus the neural network has 30×30 topologically organized neurons, where all the neural activities are initialized to zero. In addition, the AUV is simplified as a particle and its shape and size

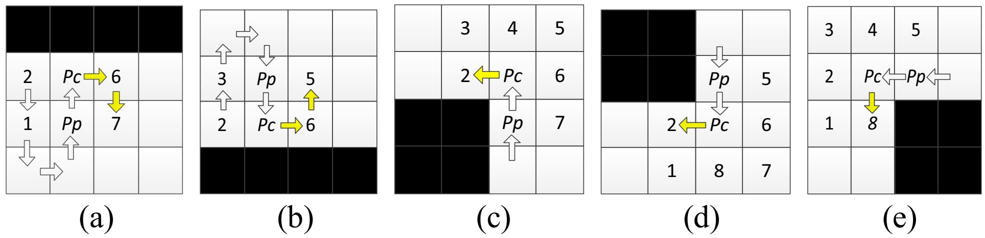


Fig. 3 The five kinds of path optimization template at the edge of obstacles. **a** The obstacles are above the AUV. **b** The obstacles are below the AUV. **c** The obstacles are in the bottom-left of the AUV. **d** The obstacles are in the top-left of the AUV. **e** The obstacles are in the bottom-right of the AUV

are not considered. And, the GBNN algorithm parameters are set as $\beta = 0.7, \alpha = 2$; $E = 100$ for the external input; $R = 2$ for the radius of the receptive field; and $c = 0.5$ for the strategy of path planning. The simulation results of complete coverage path of the AUV in the static and dynamic environment are implemented by MATLAB.

3.1 Complete Coverage Path Planning of the AUV in the Static Underwater Environment

In this section, the complete coverage path planning of the AUV in the static environment is presented. In the static environment, the obstacles are considered as still without any moving. The initial simulation environment is shown in Fig. 4. In Fig. 4a, there are multiple irregular obstacles in the grid map, the AUV begins to perform the complete coverage task from S (1,30). Figure 4b is the neural activity landscape of the neural network corresponding to this grid map. With the large external inhibitory inputs ($-E$), the neural activities of obstacles keep as -1 and only have local effect, thus the obstacle areas just locally push the AUV away to avoid collision. With the large external excitatory inputs ($+E$), the neural activities of uncovered areas are 1. For the proposed algorithm, an external excitation signal is introduced for all corresponding neurons that need to be covered, and the external excitation inputs will always exist as long as the area is not covered by AUV. Therefore, the neuronal activity value can be always in peak condition and propagate outward, thus the uncovered areas always attract the AUV to visit. For the covered areas, their external inputs are zero and the excitatory inputs just from the lateral neural connection, thus the neural activities begin to decay to zero gradually. This mechanism can guarantee the complete coverage path planning in the underwater environment.

During the AUV sailing, the neural activity landscape of neural network updates dynamically with the varying grid map, the AUV plans the complete coverage path by the strategy of path planning in Eq. 5 and the optimization

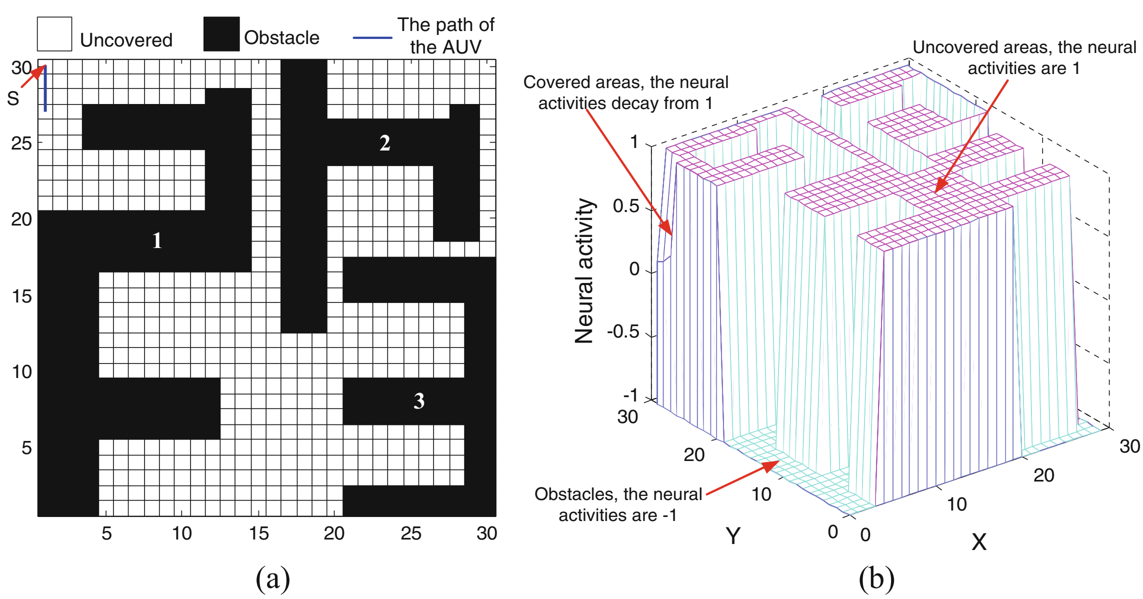


Fig. 4 The initial static underwater workspace of the AUV. **a** The grid map of the AUV underwater workspace. **b** The neural activity landscape of neural network corresponding to the grid map

templates of avoiding obstacles path in Fig. 3. As shown in Fig. 5a, when the AUV arrives at location (11,21), it stuck in the deadlock, where all the neighboring locations are either obstacles or covered areas. For the neuron (11,21), its external input all becomes to be zero, and the excitatory inputs are just from the side neural connections. And then the neural activity decays. Meanwhile, with the neural activity propagation among neurons, the positive neural activities from uncovered areas have propagated to the neuron (11,21) and the surrounding neighboring neurons. And for the neurons, the more they close to the uncovered areas, the more they can get excitatory inputs, thus the neural activities of neighboring locations (except obstacles) are all larger than the activity at (11,21). Therefore, the AUV can immediately plan a conventional point-to-point path like in [22] to escape from the deadlock without any waiting or staying. In order to further prove that the AUV can immediately get out of deadlocks in GBNN algorithm, take deadlock (11,21) for example, the neural activities of neighboring locations when the AUV stuck in the deadlock (11,21) is shown in Table 1.

In Table 1, when the AUV stuck in deadlock (11,21), the neural activity at (11,21) decays from 1 to 8.7620e-13. Meanwhile, with the positive neural activities from the uncovered areas, it is observed that all the neural activity values of neighboring locations at point (11,21) (except the obstacle area with the value -1) have been larger than deadlock neural activity value 8.7620e-13. And the neighboring locations are closer to the uncovered areas, their neural activities are larger. Thus, the AUV can immediately escape from the deadlock and pass some covered areas to the nearest uncovered location (4,28) according to the largest neural activity value marked with red underline in Table 1, then continues to perform the coverage task. The path of escaping from deadlock (11,21) is shown in Fig. 5b. Later, when the AUV arrives at location (16,30), (26,23), (30,30) and (28,14) are all stuck in deadlock, in the same way, it also can immediately escape from deadlocks without any staying and quickly sail to the nearest uncovered area. The specific results are shown in Fig. 5c – f.

As shown in Fig. 6, the AUV accomplishes the coverage task when arrives at the point F(28,6). By Eq. 5, the AUV plans the complete coverage path through the dynamic activity landscape of the neural network and the previous AUV position, so as to make fewer turnings. In addition, at the edge of some obstacles, the paths of the AUV with obstacle avoidance are optimized to reduce overlapped and cluttered paths by the path optimization templates of Fig. 3. For example, when the AUV sails from location (1,21) to (1,20), there are obstacles below it and the right location is uncovered, then the AUV turns right to (2,21) by the path optimization template in Fig. 3b, afterwards, it turns up to (2,22). Similarly, when the AUV arrives at location (4,24) from (4,23), through the template in Fig. 3a, the AUV firstly turns right to (5,24) and then turns down to (5,23). When the AUV sails from location (15,17) to (15,16), it turns left to cover (14,16) by the template of Fig. 3d. When the AUV arrives at (30,28) from (30,27), it then turns left to location (29,28) by the path template in Fig. 3c. When the AUV sails right location (28,28) to left (27,28), then it turns down to (27,27) by the template of Fig. 3e. From the above analysis, through the complete coverage path planning method based on GBNN algorithm, in the static environment, the AUV not only can automatically plan a reasonable path with obstacles avoidance until the entire workspace is completely covered, but also can immediately escape from deadlocks, and the efficiency of path planning is high.

3.2 Complete Coverage Path Planning of the AUV in the Dynamic Underwater Environment

In this paper, the dynamic underwater environment is defined as an environment with some dynamic obstacles. In this section, the simulation results of complete coverage path planning of the AUV in the dynamic environment with suddenly appearing obstacles and suddenly moving obstacles are given.

As shown in Fig. 7a, there are multiple obstacles with different shapes and sizes. Among them, obstacle 5 is a dynamic obstacle which occupies the assigned areas in the initial time, and the AUV cannot visit. Figure 7b is the corresponding neural activity landscape. The neural activities of the uncovered areas are 1, the neural activities of the obstacles are -1, and the neural activities at the covered areas decay from 1 to 0 constantly. The AUV starts to completely cover the workspace from S (1,1), as the same as the path planning in the static environment, the AUV plans complete coverage path through Eq. 5 and path optimization templates. However, when the AUV arrives at location (14,8), a rectangular obstacle 8 suddenly appears in front of the AUV. As shown in Fig. 7d, the neural activities of the areas with the suddenly placed obstacle 8 immediately become -1 from 1 with the large external inhibitory. And, then the AUV can rapidly plan the reasonable path of avoiding obstacles to nearest uncovered areas according to the change of the neural activities (see Fig. 7c).

In Fig. 7e, when the AUV arrives at F (30,1), it finishes the coverage tasks, but at the same time, dynamic obstacle 5 suddenly moves to the bottom-left of the current position. The corresponding areas of the grid map update as new uncovered areas and new obstacle areas, respectively. Thus, the neural activities at the previous obstacle 5 location become 1 and the neural activities of the current position become -1 at once, which are shown in Fig. 7f. Since

Fig. 5 The paths of the AUV escaping from the each deadlock. **a** The AUV stuck in deadlock (11,21). **b** The AUV escapes from deadlock (11,21). **c** The AUV escapes from deadlock (16,30). **d** The AUV escapes from deadlock (26,23). **e** The AUV escapes from deadlock (30,30).**f** The AUV escapes from deadlock (28,14)

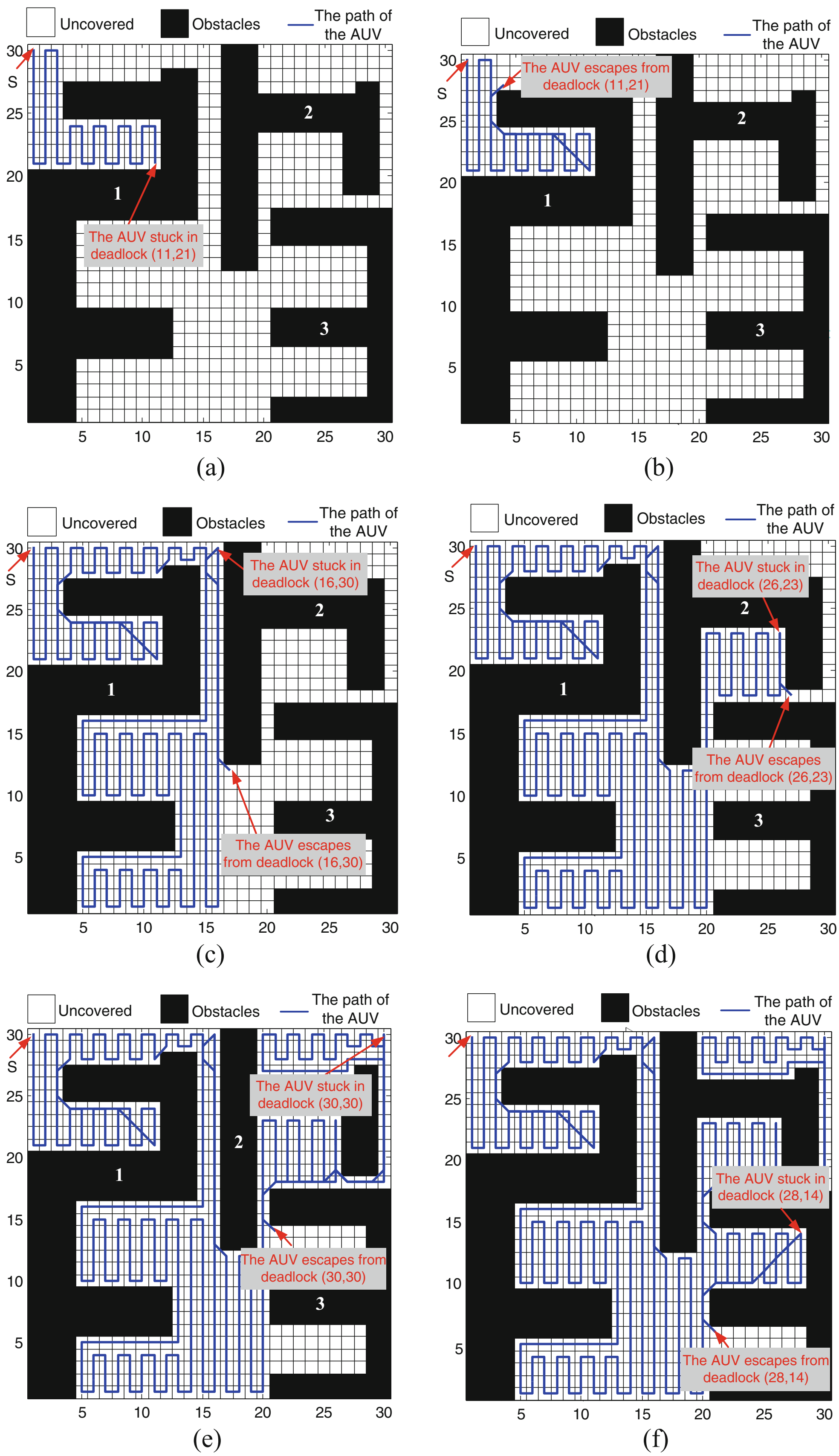


Table 1 The neural activities of neighboring locations when the AUV stuck in the deadlock (11,21) in GBNN algorithm

(x , y) Neural Activity y / x	20	21	22	23	24	25	26	27	28
1	−1	3.9109e-08	2.7221e-07	1.7957e-06	1.1179e-05	6.3014e-05	2.9762e-04	0.0010	0.0023
2	−1	7.0322e-08	5.1936e-07	3.7667e-06	2.7906e-05	1.8904e-04	0.0012	0.0059	0.0169
3	−1	7.2772e-08	5.4842e-07	4.1857e-06	3.3544e-05	**2.8933e-04**	**0.0027**	**0.0258**	0.1258
4	−1	4.2281e-08	2.7238e-07	1.5799e-06	**7.1648e-06**	−1	−1	−1	**1**
5	−1	1.5880e-08	8.1678e-08	3.2887e-07	**7.3841e-07**	−1	−1	−1	1
6	−1	4.3824e-09	1.7900e-08	5.1812e-08	**8.0032e-08**	−1	−1	−1	1
7	−1	9.8071e-10	3.2825e-09	7.4524e-09	**9.0664e-09**	−1	−1	−1	1
8	−1	1.9097e-10	5.4540e-10	1.0331e-09	**1.0644e-09**	−1	−1	−1	1
9	−1	3.3881e-11	8.5478e-11	**1.4108e-10**	1.2941e-10	−1	−1	−1	1
10	−1	5.6356e-12	**1.2911e-11**	1.9161e-11	1.6107e-11	−1	−1	−1	1
11	−1	**8.7620e-13**	1.8661e-12	2.5548e-12	2.0136e-12	−1	−1	−1	1
12	−1	−1	−1	−1	−1	−1	−1	−1	−1

the positive neural activities can propagate to the entire workspace and the uncovered areas can globally attract the AUV, then the AUV sets out again from (30,1) to visit the new uncovered areas. As shown in Fig. 7g and h, when the AUV reaches (16,24), the whole workspace is completely covered without any omission. Therefore, through GBNN algorithm, the AUV not only can plan a reasonable path of avoiding obstacles for suddenly appearing obstacles, but also can completely cover the new uncovered areas caused by suddenly moving obstacle.

3.3 Comparison with the biologically inspired neural network algorithm

To further test the priority of GBNN algorithm proposed in this paper, the complete coverage path planning of the AUV based on biologically inspired neural network algorithm (with the abbreviation BNN algorithm in the following Tables for convenience) in [13] is given in Fig. 8 (the algorithm parameters are set as $A = 20, \mu = 0.7, B = D = 1, R = 2, E = 100$). Then, the performance of the two algorithms is compared from the coverage rate, the overlapping coverage rate, the time of the AUV planning path, etc.

As shown in Fig. 8, the AUV also sets out from S (1,30), and plans a collision-free complete coverage path by Eq. 5 and the optimization templates of Fig. 3 until it reaches the final location F (28,6). And, the AUV will stuck in deadlocks when it arrives at location (11,21), (16,30), (26,23), (30,30) and (28,14). Compared with Fig. 6, the paths of escaping from deadlocks are more cluttered. In addition, in the biologically inspired neural network

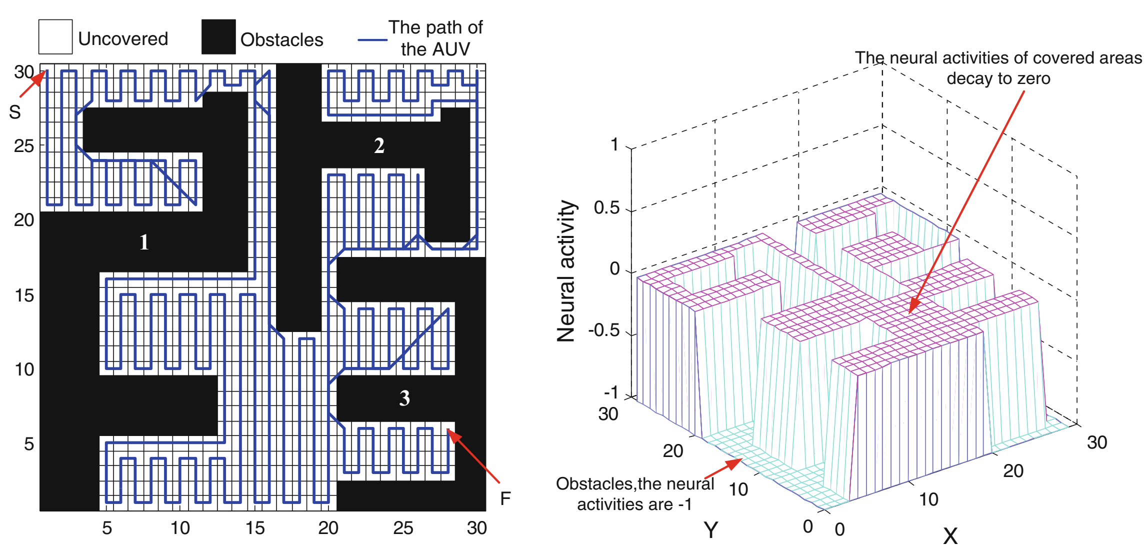
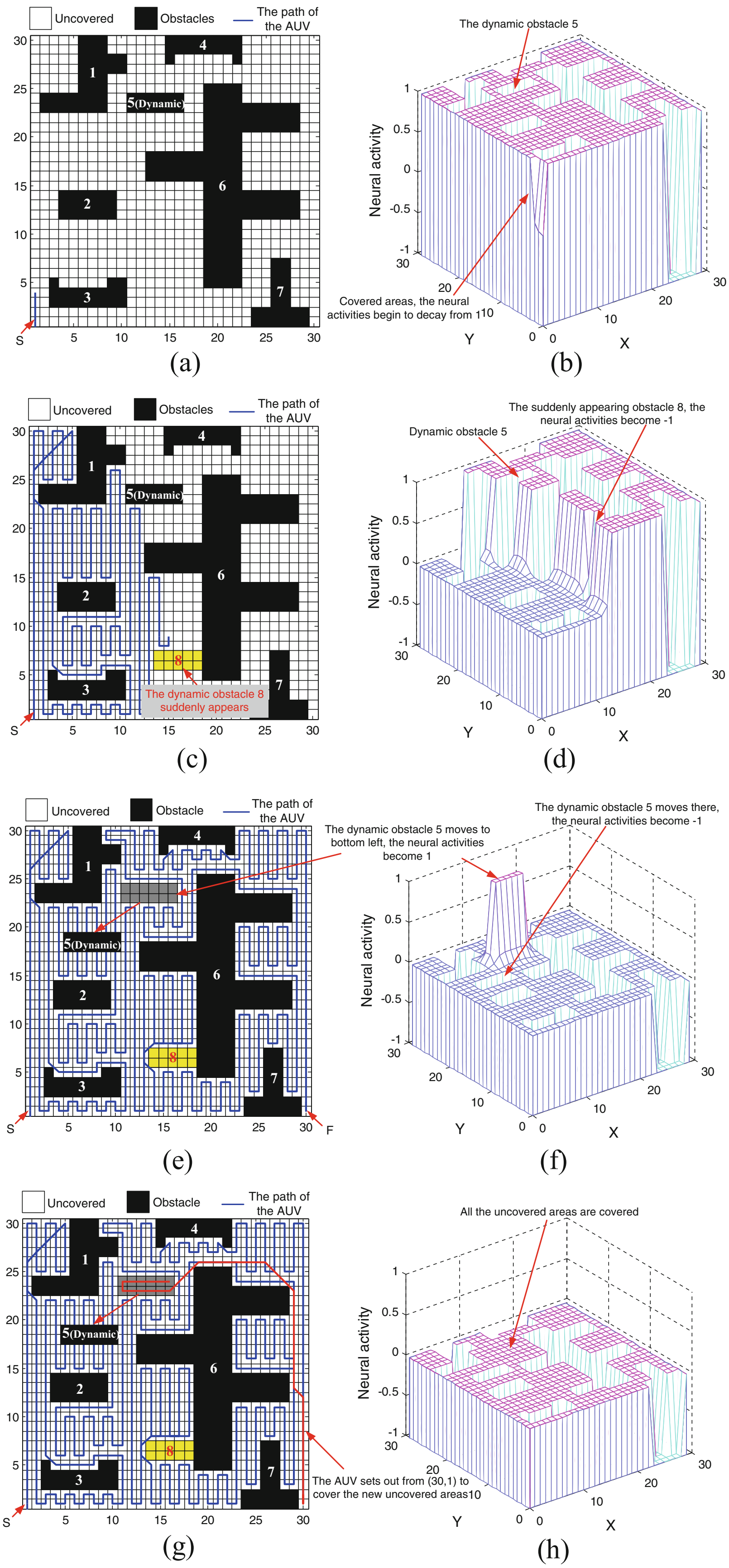


Fig. 6 The complete coverage path of the AUV in the static underwater workspace based on GBNN algorithm

Fig. 7 The complete coverage paths of the AUV in the dynamic environment.**a** The initial underwater simulation environment. **b** The initial neural activity landscape of the neural network. **c** The rectangular obstacle 8 suddenly appears in front of the AUV. **d** The neural activity landscape when a rectangular obstacle 8 suddenly appears. **e** The dynamic obstacle 5 suddenly moves to the bottom-left of the current position. **f** The neural activity landscape when obstacle 5 suddenly moves to the bottom-left. **g** The AUV completes the supplementary coverage of the new uncovered areas and the whole workspace is covered. **h** The neural activity landscape of the neural network when the whole workspace is covered



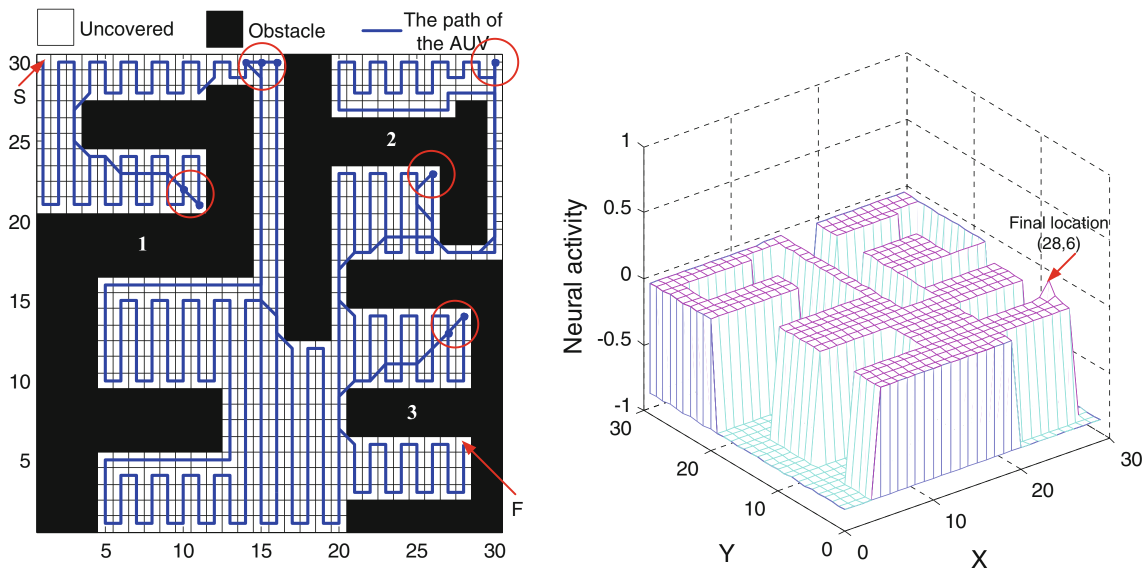


Fig. 8 The complete coverage path of the AUV based on biologically inspired neural network algorithm

algorithm, when the AUV stuck in the deadlock, the excitatory inputs of the deadlock are also just from the lateral neural connections, although the positive neural activities from uncovered areas are propagating to the entire workspace. Since the neural activities have a slow passive descending process under the passive decay term, thus the neural activity of the deadlock is larger than all the activities of neighboring locations in a longer time. According to Eq. 5, the AUV will always choose the current deadlock location as the next position, which means the AUV will stay in the deadlock multiple step times. And the AUV will begin to escape from the deadlock by sailing the point-to-point path until the neural activities at the deadlock are smaller than neighboring locations [13]. Therefore, the AUV needs a long time to escape from deadlocks. Similarly, take deadlock (11,21) for example. The neural activity values of neighboring locations when the AUV stuck in the deadlock (11,21) with biologically inspired neural network algorithm are shown in Table 2. When the AUV stuck in deadlock (11,21), the neural activity at (11,21) is 0.1145 which is larger than the neighboring locations. Thus the AUV has to stay in the deadlock (11,21) multiple step times where the activity value at (11,21) will decay at each step time, and escapes from (11,21) until its activity value is smaller than the neighboring locations. The number of overlapping steps of the AUV staying in deadlocks is given in Table 3.

As shown in Table 3, in GBNN algorithm, the AUV has no any staying and no any overlapping steps at each deadlock. However, in biologically inspired neural network algorithm, the AUV not only revisits deadlocks multiple times, but also stuck in new deadlocks again during escaping

Table 2 The neural activities of neighboring neurons when the AUV stuck in the deadlock (11,21) in biologically inspired neural network algorithm

(x , y) Neural Activity — y / x	20	21	22	23	24	25	26	27	28
1	−08333	3.9335e-9	4.3298e-8	4.7293e-7	4.9243e-6	4.4246e-5	1.6683e-4	3.9649e-4	6.8233e-4
2	−08333	3.3638e-9	3.5198e-8	3.8816e-7	4.4696e-6	5.2722e-5	6.7783e-4	0.0023	0.0044
3	−0.8333	4.3780e-9	2.8235e-8	2.8145e-7	3.2458e-6	4.8932e-5	8.5872e-4	0.0222	0.0508
4	−08333	1.9941e-8	4.4666e-8	1.7221e-7	1.3589e-6	−08333	−08332	−0.8176	0.8356
5	−08333	1.6552e-7	2.7588e-7	3.1538e-7	2.2404e-7	−08333	−0.8333	−0.8120	0.8369
6	−08333	1.5364e-6	2.4539e-6	2.4180e-6	1.4783e-6	−08333	−0.8333	−0.8120	0.8369
7	−08333	1.5183e-5	2.2892e-5	2.0922e-5	1.1995e-5	−08333	−0.8333	−0.8120	0.8369
8	−08333	1.5596e-4	2.1354e-4	1.6926e-4	8.4359e-5	−0.8333	−0.8333	−0.8120	0.8369
9	−08332	0.0016	0.0019	0.0011	4.1612e-4	−0.8333	−0.8333	−0.8120	0.8369
10	−08321	0.0147	0.0132	0.0042	0.0011	−0.8333	−0.8333	−0.8120	0.8369
11	−08317	**0.1145**	0.0270	0.0061	0.0013	−0.8333	−0.8333	−0.8182	0.8361
12	−08323	−08316	−08319	−08330	−08333	0.8333	−0.8333	−0.8270	−0.8033

Table 3 The number of overlapping steps of the AUV staying in deadlocks

Algorithm	The overlapping steps of the AUV at each deadlock.								
	(11,21)	(10,22)	(16,30)	(15,30)	(14,30)	(26,23)	(30,30)	(28,14)	(27,13)
BNN algorithm	8	5	7	16	2	5	42	8	5
GBNN algorithm	0	0	0	0	0	0	0	0	0

Table 4 Comparison of two algorithms in the complete coverage path planning

Algorithm / Indicators	BNN algorithm	GBNN algorithm
The coverage rate	100%	100%
The total sailing time of the AUV	2894.92	136.19
The total steps of the AUV	665	565
The overlapping steps of the AUV	164	64
The overlapping coverage rate	24.66%	11.33%
The turning numbers escaping from deadlock paths	35	26

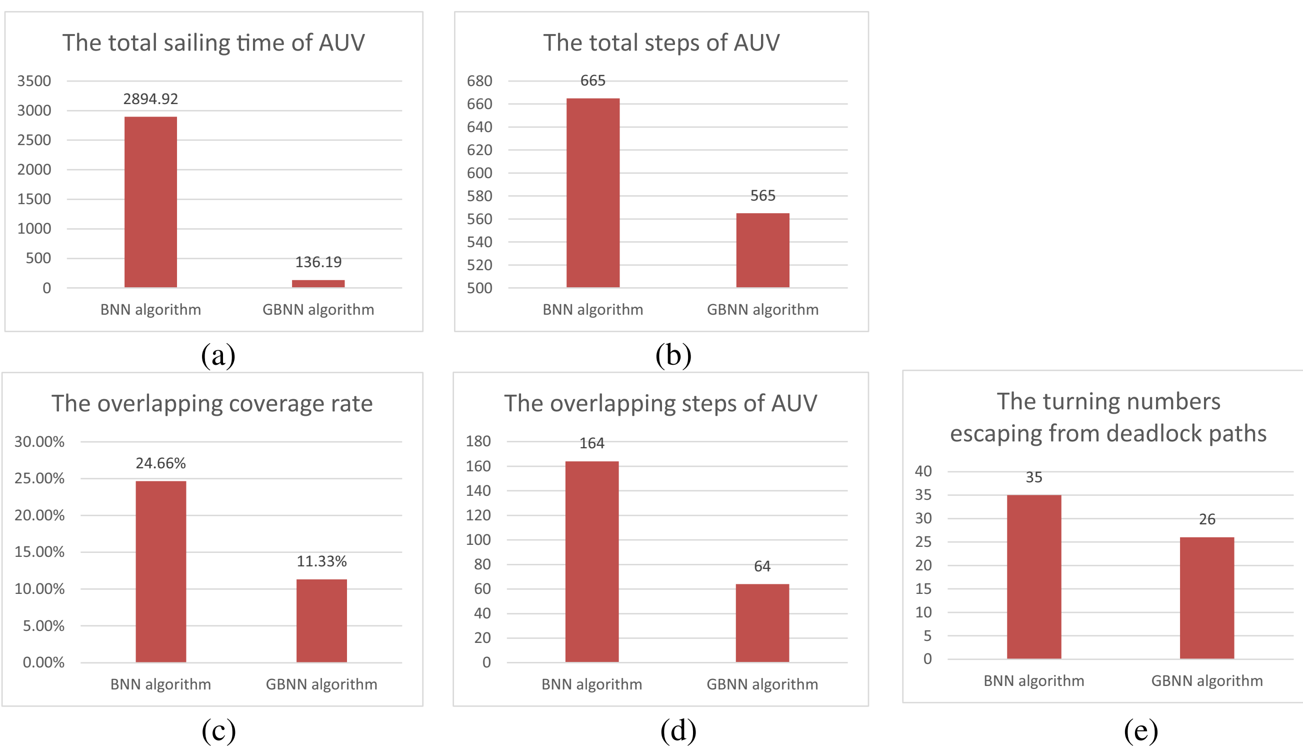


Fig. 9 Comparison of complete coverage path planning with graphical representation

from the previous deadlocks, such as at location (10,22), (15,30), (14,30), etc. And, as can be seen from Fig. 8, the AUV leaves the dot-like overlapping traces at each deadlock, but this phenomenon doesn't exist in the path of GBNN algorithm shown in Fig. 6.

As shown in Table 4, in order to show the efficiency of the proposed method, the biologically inspired neural network algorithm is adopted to do comparison in complete coverage path planning from the following items: (1) The coverage rate, (2) The total sailing steps/time of the AUV, (3) The overlapping coverage rate/steps, (4) The turning numbers.

As can be seen in Table 4, both of the two algorithms can reach 100% coverage rate. However, for biologically inspired neural network algorithm, due to the reason that differential equation is computationally complex and the AUV needs a long time to escape from deadlocks, thus the total sailing time of the AUV ups to 2894.92s. And because the AUV stays in the deadlocks for repeated step times, it needs 665 steps to finish the coverage task, among them, the number of overlapping steps is 164 steps and the overlapping coverage rate ups to 24.66%. However, in GBNN algorithm, the difference equation is computationally simpler and the AUV can immediately escape from deadlocks without any staying, thus the AUV just needs 136.19s and 565 steps to completely cover the same workspace. Therefore, the real-time capability of path planning is better. And except the 64 overlapping steps in the path of escaping from deadlocks, there are no other overlapping steps any more. The overlapping coverage rate is only 11.33% and less than half of biologically inspired neural network algorithm. Otherwise, there are greater differences between the paths of the AUV escaping from deadlocks in the two algorithms. For example, in GBNN algorithm, the number of turning in the paths of escaping from deadlocks is only 26, but the number of turning is 35 in biologically inspired neural network algorithm, which is 25.6% more than GBNN algorithm. In order to make the result more clearly, comparison with graphical representation is given in Fig. 9. Therefore, compared with biologically inspired neural network algorithm, in the proposed method, the AUV can not only cover the entire objective workspace rapidly, but also can immediately escape from deadlocks without any overlapping. Meanwhile, the efficiency of path planning is improved, the path planning time is effectively shortened and the overlapping coverage rate is reduced. For the AUV, it can save more power and perform more tasks.

4 Conclusion

In this paper, GBNN algorithm is proposed and applied to complete coverage path planning of the AUV. The algorithm has no learning procedures and is computationally simple. The AUV plans the complete coverage path through the dynamic neural activity and the previous AUV location, and the paths at the edge of some obstacles are optimized by path optimization templates. The simulation results show that whether in the static or dynamic underwater environment, the AUV can not only plan a reasonable path quickly with obstacles avoidance until the entire workspace is covered, but also immediately escapes from deadlocks. And the path planning is efficient with short path planning time and low overlapping coverage rate. In the future study, the ocean current will be taken into account in the complete coverage path planning.

Acknowledgments This project is supported by the National Key Research and Development Plan (2017YFC0306302), National Natural Science Foundation of China (51575336, 61503239), Creative Activity Plan for Science and Technology Commission of Shanghai (16550720200).

References

1. Joung, T.H., Lee, J.H., Nho, I., Kim, B.J.: A study on the design and manufacturing of a deep-sea unmanned underwater vehicle based on structural reliability analysis. Ships Offshore Struc. **33**(3), 125–132 (2009)
2. Hameed, I.A.: Intelligent coverage path planning for agricultural robots and autonomous machines on Three-Dimensional terrain. J. Intell. Robot Syst. **74**(3), 965–983 (2014)
3. Hsu, P., Lin, C., Yang, M.: On the complete coverage path planning for mobile robots. J. Intell. Robot. Syst (2014). https://doi.org/10.1007/s10846-013-9856-0
4. Galceran, E., Carreras, M.: A survey on coverage path planning for robotics. Robot. Auton. Syst. **61**(12), 1258–1276 (2013)
5. Gage, D.W.: Randomized search strategies with imperfect sensors. In: Proceedings SPIE mobile robots VIII, pp. 270–279 (1993)
6. Ko, I., Kim, B., Park, F.C.: Randomized path planning on vector fields. Int. J. Rob. Res. **33**(13), 1664–1682 (2014)
7. Latombe, J.C.: Exact cell decomposition. robot motion planning, pp. 200–247. Springer, US (1991)
8. Choset, H.: Coverage of known spaces: the boustrophedon cellular decomposition. Auton. Robot. **9**(3), 247–253 (2000)
9. Acar, E.U., Choset, H., Lee, J.Y.: Sensor-based coverage with extended range detectors. IEEE T. Robot. **22**(1), 189–198 (2006)
10. AI Marzouqi, M., Jarvis, R.A.: Robotic covert path planning: a Survey. In: Proceedings IEEE international conference on robotics, automation and mechatronics, pp. 77–82 (2011)
11. Sadeghzadeh, M., Calvert, D., Abdullah, H.A.: Self-Learning visual servoing of robot manipulator using Explanation-Based fuzzy neural networks and Q-Learning. J. Intell. Robot. Syst. **78**(1), 83–104 (2015)
12. Gao, L., Pan, S.L., Shen, J.: A robot path planning scheme based on neural network. J. Theor. Appl. Inf. Technol. **46**(2), 654–658 (2012)
13. Yang, S., Luo, C.: A neural network approach to complete coverage path planning. IEEE T. Syst. Man CY. B. **34**(1), 718–724 (2004)
14. Luo, C.M., Yang, S.X., Mo, H.W., Li, X.D.: Safety aware robot coverage motion planning with Virtual-obstacle-based navigation. In: Proceedings IEEE international conference information and automation, pp. 2110–2115 (2015)

15. Yan, M.Z., Zhu, D.Q., Yang, S.X.: Complete coverage path planning in an unknown underwater environment based on D-S data fusion real-time map building. Int. J. Distrib. Sens N (2012). https://doi.org/10.1155/2012/567959
16. Li, J.H., Yang, J.G., Tian, X.J., Gao, M.: An improved ant colony algorithm for robot path planning. Soft. Comput (2016). https://doi.org/10.1007/s00500-016-2161-7
17. Zhang, C., Wang, X., Du, Y.: Complete coverage path planning based on ant colony algorithm. In: Proceedings IEEE international conference on mechatronics and machine vision in practice, pp. 357–361 (2008)
18. Dogru, S., Marques, L.: Energy efficient coverage path planning for autonomous mobile robots on 3D terrain. In: Proceedings IEEE international conference on autonomous robot systems and competitions, pp. 118–123 (2015)
19. Kapanoglu, M., Alikalfa, M., Ozkan, M.: A pattern-based genetic algorithm for multi-robot coverage path planning minimizing completion time. J. Intell. Manuf. **23**(3), 1035–1045 (2012)
20. Glasius, R., Komoda, A., Gielen, S.: A biologically inspired neural net for trajectory formation and obstacle avoidance. Biol. Cybern. **6**(74), 511–520 (1996)
21. Luo, C.M., Yang, S.X., Stacey, D.A., Jofriet, J.C.: A solution to vicinity problem of obstacles in complete coverage path planning. In: Proceedings IEEE international conference on robotics and automation, pp. 612–617 (2002)
22. Li, H., Yang, S.X., Biletskiy, Y.: Neural network based path planning for a multi-robot system with moving obstacles. In: Proceedings IEEE international conference on automation science and engineering, pp. 163–168 (2008)

Daqi Zhu was born in Anhui, China. He received the B.Sc. degree in physics from the Huazhong University of Science and Technology, Wuhan, China, in 1992, and the Ph.D. degree in electrical engineering from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2002. He is currently a Professor with the Information Engineering College, Shanghai Maritime University, Shanghai, China. His current research interests include neural networks, fault diagnosis, and control of autonomous underwater vehicle.

Chen Tian was born in Xuzhou, China. She is a postgraduate student of Laboratory of Underwater Vehicles and Intelligent Systems. Her research interest is multi-AUV complete coverage.

Bing Sun was born in Haimen, China. He received the B.Sc. degree in communication engineering and M.S. degree in communication and information systems and Ph.D. degree in power electronics and power transmission from Shanghai Maritime University, Shanghai, China, in 2009, 2011 and 2014, respectively. He is currently a lecture and master supervisor in the Colleague of Information Engineering, Shanghai Maritime University. His current research interests include path planning and tracking control of underwater vehicles.

Chaomin Luo (S'01M'08) received the B.Sc. degree in radio engineering from Southeast University, Nanjing, China, in 1994, the M.Sc. degree in engineering systems and computing from the University of Guelph, Guelph, ON, Canada, in 2002, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2008. In 2008, he was an Assistant Professor with National Taipei University, Taipei, China. In 2009, he joined the University of Detroit Mercy, Detroit, Michigan, USA, where he is currently an Associate Professor with Advanced Mobility Laboratory. His current research interests include robotics and automation, intelligent systems, computational intelligence, mechatronics, very large scale integration, and embedded systems. He was the Panelist in the Department of Defense, USA, 2015-2016, 2016-2017 NDSEG Fellowship program, and National Science Foundation, USA, GRFP program, 2016-2017.