This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCDS.2017.2727678, IEEE Transactions on Cognitive and Developmental Systems

Biologically Inspired Self-Organizing Map Applied to Task Assignment and Path Planning of an AUV System

Daqi Zhu, Xiang Cao, Bing Sun and Chaomin Luo, *Member, IEEE*

***Abstract*—An integrated biologically inspired self-organizing map (BISOM) algorithm is proposed for task assignment and path planning of an autonomous underwater vehicle (AUV) system in three-dimensional underwater environments with obstacle avoidance. The algorithm embeds the biologically inspired neural network (BINN) into the self-organizing map (SOM) neural networks. The task assignment and path planning aim to arrange a team of AUVs to visit all appointed target locations, while assuring obstacle avoidance without speed jump. The SOM neuron network is developed to assign a team of AUVs to achieve multiple target locations in underwater environments. Then, in order to avoid obstacles and speed jump for each AUV that visits the corresponding target location, the BINN is utilized to update weights of the winner of SOM, and achieve AUVs path planning and effective navigation. The effectiveness of the proposed hybrid model is validated by simulation studies.**

***Index Terms*—Autonomous Underwater Vehicles (AUVs), task assignment and path planning, biologically inspired self-organizing map (BISOM), obstacle avoidance, speed jump**

I. Introduction

AUTONOMOUS underwater vehicle (AUV) has now become a hotspot area in the recent years especially the multi-AUV system due to the high parallelism, robustness and collaboration of high efficiency [1-6]. The task assignment and path planning of AUVs is the fundamental issue in the multiple autonomous underwater vehicles research field. Therefore, how a team of AUVs is assigned to achieve multiple target locations and avoid obstacles autonomously in underwater environments is still an open challenging issue in AUVs.

There are some achievements reported about studies of task assignment and path planning in the recent decades. Deng, *et al.* [7] presented a multi-AUV task assignment and path planning method in combination with LAAF (location-aided task allocation framework) and GMOOP (grid-based multiobjective optimal programming) algorithms under the constraints of underwater acoustic communication. Monn, *et al.* [8] proposed a hierarchical framework for UAVs (unmanned aerial vehicles) task assignment and path planning under the dynamic environments. With regard to the path planning portion, it utilizes shortest-path principle combined with A* method and also augments with a potential field-based algorithm for obstacle avoidance. For the task assignment portion, it uses a negotiation strategy in search of a solution by negotiating for lower costs.

Since there exists a similarity between the multi-task assignment and self-organizing map (SOM) neural networks, a SOM based approach is naturally proposed by Zhu, *et al.* [9-11] to resolve the task assignment for the multi-robot system. The multi-task assignment for a multi-robot system is acquired by the competitive principle of SOM, in which each robot path planning is implemented by adjusting the weight update rules of SOM. Instead of applying SOM algorithm in the multi-task assignment, Sato, *et al*. [12] suggested a switching controller system for wheeled mobile robots in rough terrain with a self-organizing map method. A sensor-based path planning method for the mobile robot is proposed by employing the self-organizing map [13]. However, these approaches based SOM aforementioned require an ideal two-dimensional workspace with NO obstacle which is quite unrealistic in the real world applications. Jung *et al.* [14] showed a hierarchical path planning approach. They use a control algorithm to compute optimal paths in several steps. Zhong *et al.* [15] developed a task assignment algorithm, which can make multiple aerial vehicles attack targets with dynamic values. This method combines multisubgroup ant colony optimization algorithm with a planning algorithm for multidestination routes that obtains a reasonable result in a short time. However, this method is inapplicable to dynamic environments. Also, it should be mentioned that most of the research work about task assignment problem is focused on a team mobile robots and UAV systems.

Inspired by the applications of the SOM algorithm in multi-agent system, Zhu, *et al.* [16-17] attempts to solve multi-task assignment issue for AUV systems. It integrates velocity synthesis and SOM method of three-dimensional (3D) underwater workspace without obstacle avoidance. Although it can solve the multi-AUV control problem in the presence of

Manuscript received May 20, 2016. This work was supported in part by the National Natural Science Foundation of China (51279098, 51575336), Creative Activity Plan for Science and Technology Commission of Shanghai (14JC1402800,16550720200).

Daqi Zhu is with the Laboratory of Underwater Vehicles and Intelligent Systems, Shanghai Maritime University, Shanghai 201306, China (phone: 021-38282874; e-mail: zdq367@aliyun.com).

Xiang Cao and Bing Sun are with the Laboratory of Underwater Vehicles and Intelligent Systems, Shanghai Maritime University, Shanghai 201306, China (e-mail: cxeffort@126.com; hmsunbing@163.com)

Chaomin Luo is with the Department of Electrical and Computer Engineering, University of Detroit Mercy, USA (e-mail: luoch@udmercy.edu).

ocean current, obstacles have not been taken into consideration yet in their project. Additionally, each AUV speed based on the weight function of SOM is also not confined which may exceed AUV's control constraint. In the practical case, it is quite impossible for AUV to achieve such a speed.

In this paper, we focus on the obstacles avoidance and the speed jumps of AUV navigation during multi-AUV task assignment and path planning, and propose an integrated biologically inspired self-organizing map (BISOM) algorithm by embedding the biologically inspired neural networks (BINN) into the SOM neural networks. The overall system primarily involves two sections: task assignment and path planning. First, SOM carries out a task assignment for AUV team members to ensure that each one of them has the optimal target to visit. Then, BINN updates the weights function of SOM according to environmental information, and plans a collision-free trajectory for each AUV from the initial position to its corresponding target location. Because of the BINN features used in path planning of AUV [18-20], for example, the automatic obstacle avoidance with no speed jumps of single AUV navigation, the proposed algorithm is robust in underwater environments, which allows for random motion of target location and obstacles.

The rest of this paper is organized as follows. In Section 2, the problem statement of multi-AUV system is introduced. In Section 3, the proposed method with integration of multi-AUV task assignment and path planning is described. The simulation results under different situations are reported in Section 4. Finally, the conclusion remarks are depicted in Section 5.

II. Problem Statement of Multi-AUV System

The essential issue of the multi-AUV system is to decompose all the tasks into several subtasks, in order that all the AUVs move towards the designated destinations correctly. A schematic diagram of 3D underwater workspace with obstacles, AUVs and targets is illustrated in Fig.1. In the workspace, AUVs, target locations and obstacles are uniform randomly distributed. In this paper it assumes environmental information of 3D underwater workspace is to be *known*, the exact locations of the obstacles have to be known a *priori*. In order to achieve multi-AUV cooperative control, the AUV members are initially assigned to a variety of initial locations.

In order to minimize the battery energy cost, each AUV is assigned to the nearest target location ideally. When the task assignment is fulfilled, the path planning must be achieved and every AUV must be directed to the target location assigned. Since there exist obstacles in the 3D underwater workspace, the path of each AUV must be collision-free. Meanwhile, the distance and speed of each AUV must meet the kinematics requirements. After all target locations in the underwater workspace have been visited by their corresponding AUVs, finally the task assignment and path planning are conducted by the whole multi-AUV team.

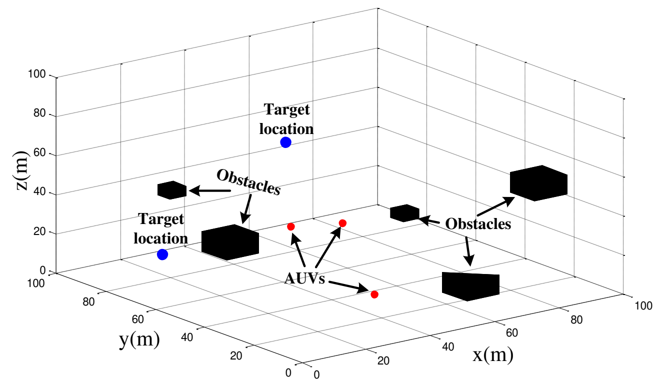


Fig.1. 3D underwater workspace

III. Proposed Algorithm

A. Traditional SOM task assignment algorithm

Due to the similar features between the SOM neural network and multi-AUV system, SOM network can be used to resolve the task assignment issue for the multi-AUV system. The neural network model that contains two layers is shown in Fig.2. There are two layers in SOM neural network: input layer and output layer. The first layer (input layer) is made up of three neurons (x_i, y_i, z_i), each of which represents the center of search target area in the 3D underwater environments. The second layer (output layer) represents AUV's locations. Each neuron of the second layer connects to the neurons in first layer. The connection weights between neurons in input layer and output layer neuron is provided by a 3D weight vector initialized by the AUV's coordinates. Target locations are input to the input layer one by one, and competition iterations are acquired. The SOM decomposes the entire issue into sub problems including the rule of winner selection, the definition of neighborhood function and the rule of weights update. First, the SOM-based method is used to select winners for the target positions of AUVs. Then, the neighborhood function chooses the winner neighbors of AUVs, and simultaneously, determines the moving speed for winners and neighbors. Finally, AUVs reach target locations along the collision-free paths by updating 3D weight vector of winner AUVs. When all target locations have been visited, iterations are complete.

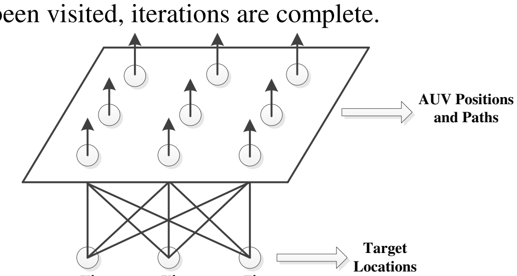


Fig.2. Structure of SOM neural network

(1) Winner selection rules

The winner selection rule is obtained by the following expression [16]:

$$[P_j] \Leftarrow \min\{D_{kjl},\ k=1,2,...K;\ j=1,2,...,J;\ l=1,2,...,L\} \quad (1)$$

where $[P_j]$ denote that the j-th neuron from the k-th group is the selected winner to the l-th input node. D_{kjl} is a relevant Euclid distance value between the correlative two neurons. The winner selection depends on how to deal with the calculation of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCDS.2017.2727678, IEEE Transactions on Cognitive and Developmental Systems

D_{kjl} over iterations. Initially, the Euclid distance of between two neurons is defined as:

$$|T_l - R_{jk}| = \sqrt{(x_l - w_{jkx})^2 + (y_l - w_{jky})^2 + (z_l - w_{jkz})^2} \tag{2}$$

$T_l = (x_l, y_l, z_l)$ represents the coordinate of l-th input neural node, which is also known as the target location. $R_{jk} = (w_{jkx}, w_{jky}, w_{jkz})$ is the coordinate of output neurons which represents one specific AUV location.

Because of the insufficient energy, the workload for each AUV should be taken into consideration to guarantee that all the AUVs can make the best use of the energy and simultaneously avoid the vehicles to break down on the way to targets due to out of energy. A parameter to control the workload of each AUV is defined as:

$$W = \frac{P_j - \bar{W}}{S + \bar{W}} \tag{3}$$

where S is the safe length that a single AUV can traverse without considering running out of energy. Safe length is in the sense that an AUV has sufficient energy on board to drive it from the current location to the current location. In addition, P_j is the actual moving length of the AUV. $\bar{W}$ is the average moving length of a team of AUVs in a certain task.

The weight distance D_{kjl} is given by

$$D_{kjl} = \begin{cases} \left|T_l - R_{jk}\right|, & 0 \le P_j < S \\ \left|T_l - R_{jk}\right|(1+W), & S \le P_j < S_{\max} \\ \infty, & P_j \ge S_{\max} \end{cases} \tag{4}$$

where $S_{\max}$ represents the maximal distance a single AUV can travel.

Three different cases are presented to describe the calculation progress of neural weight distance D_{kjl}. In the first case, where the moving distances are shorter than the safe distances, the winner is straightforward to be found, and the workload balance can ensure the completion of the overall task with the least total and individual consumption. In the second case, where the AUV's moving length is longer than the safe length, the workload equitable measures will be executed to deal with the energy problem. If the AUV's traveling distance is longer than the maximal length which a single AUV can afford in a mission. There is a risk for the AUV to run out of energy. In this condition, D_{kjl} will be set to be ∞, indicating that this AUV will never be recognized as the winner.

(2) Neighborhood updating rules

When the winner selection is finished, it is extended to design of the neighborhood function and weights computation of winner and neighbors. The neighborhood is considered as a sphere with radius λ where the center is the winner node. The neighbors of the winner are calculated as follows:

$$f(d_m, g) = \begin{cases} e^{-d_m^2/g^2(t)}, & d_m < \lambda \\ 0, & others \end{cases} \tag{5}$$

where $d_m = \left|N_m - N_j\right|$ is the distance between the m-th output neuron and the winner. λ is set to a constant value representing the neighborhood range. The nonlinear function $g^2(t)$ is calculated by $g^2(t) = (1-\partial)^t g_0$. ∂ is the update rate determining the calculation time and $\partial < 1$ is constant. The computation time is diminishing as the parameter ∂ increases.

The winners and their corresponding neighbors of the AUVs will move along effective paths. However, in traditional SOM method [14], the update rule for the winner AUV path without considering the obstacle and speed jump is defined as

$$R_{jk}(t+1) = \begin{cases} T_l, & D_{kjl} \le D_{\min} \\ R_{jk}(t) + \partial \times f(d_m, g) \times (T_l - R_{jk}(t)), & D_{kjl} > D_{\min} \end{cases} \tag{6}$$

$D_{\min}$ is the termination condition of iterations. If the condition $D_{kjl} \le D_{\min}$ is satisfied, the weight is replaced by the corresponding target location coordination. ∂ is the update rate with $\partial < 1$, and $f(d,g)$ is the neighborhood function. $T_l = (x_l, y_l, z_l)$ is the position of l-th input neural node in the 3D coordinate system, which also donates the target location. $R_{jk} = (w_{jkx}, w_{jky}, w_{jkz})$ is the SOM coordinate of output neurons which represents the one certain AUV position.

Obviously, in the traditional SOM-based task assignment and path planning approach [13], there are two shortages: 1) the kinematics constraint for AUV is not considered; 2) the obstacles are not considered, neither. By analyzing the performance of the update rule (4), AUV speed jump will occur when the error $T_l - R_{jk}$ between AUV current coordination R_{jk} and corresponding target coordination T_l change suddenly. Especially in the initiate stage of AUV navigation after the winner is selected, $T_l - R_{jk}$ may be very large that the control signal would exceed AUV's maximum speed. It is practically impossible for AUV to gain such a large speed. In addition, the traditional SOM-based task assignment and path planning approach cannot achieve obstacle avoidance. The underwater environment with each AUV moving ability is supposed in an ideal condition.

B. The biologically inspired self-organizing map (BISOM)

To avoid obstacles and update reasonably weight of SOM, BINN is applied to regulate the weights of SOM, and an integrated BISOM method is proposed in this paper. In order to better understand the BISOM algorithm, a biologically inspired neural network [20-22] is introduced for AUV collision-free path planning under obstacles condition.

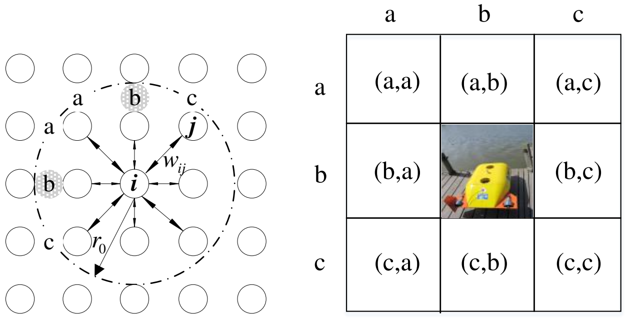
(1) The biologically inspired neural network model

The biologically inspired neural network model is used to the 3D safe obstacle avoidance for AUVs. Firstly, AUV 3D grid coordinate map of working environments is represent by a 3D biologically inspired neural network. Secondly, the AUV path is generated on the basis of the BINN output value.

To facilitate the discussion, a two-dimensional (2D) workspace of the AUV and 2D biologically inspired neural

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCDS.2017.2727678, IEEE Transactions on Cognitive and Developmental Systems

network are presented. A schematic diagram of the BINN is shown in Fig.3 (a) and the receptive field of the ***i***-th neuron is represented by a circle with a radius of r_0 ($r_0 = 2$). The neuron in Fig.3 (a) corresponds to the AUV position in Fig.3 (b).



(a) Receptive field (b) 8 correspondence positions of the grid map

Fig.3. Schematic diagram of BINN

The BINN can be regarded as a neural network architecture which is a dynamic response to the varying environments. The i-th neuron position in the BINN can be set as a vector V_i which represents the underwater position. The dynamics neural activity output of the i-th neuron in the neural network is provided as the following shunting equation [20-22].

$$\frac{dV_i}{dt} = -AV_i + (B - V_i)([I_i]^+ + \sum_{j=1}^{k} w_{ij}[V_j]^+) - (D + V_i)[I_i]^- \quad (7)$$

where k is the number of neighboring neurons. The external input I_i is defined as

$$I_i = \begin{cases} E, & \textit{if it is a target location} \\ -E, & \textit{if it is an obstacle location} \\ 0, & \textit{others} \end{cases} \quad (8)$$

where A, B, D and E are positive constants, -A reflects the passive decay rate of neuron i's activity, B and D are upper and lower limits of V_i respectively, i.e. $V_i \in [-D, B]$. The terms $[I_i]^+ + \sum_{j=1}^{k} w_{ij}[V_j]^+$ and $[I_i]^-$ are the excitatory and inhibitory inputs, respectively. The connection weight w_{ij} between the i-th and j-th neurons can be defined as $w_{ij} = f(|q_i - q_j|)$, where $|q_i - q_j|$ represents the Euclidean distance between adjacent neuron vectors q_i and q_j in the BINN, and $f(|q_i - q_j|)$ is defined as

$$w_{ij} = f(|q_i - q_j|) = \begin{cases} \mu / |q_i - q_j|, & 0 \le |q_i - q_j| \le r_0 \\ 0, & |q_i - q_j| > r_0 \end{cases} \quad (9)$$

where μ and r_0 are position constants. Therefore, each neuron has only local lateral connections in a small region $[0, r_0]$. It is obvious that the weight w_{ij} is symmetric, $w_{ij} = w_{ji}$.

For the AUV's current location in the BINN denoted as P_i, the AUV's next moving location P_n can be obtained by

$$P_n \Leftarrow V_{P_n} = \max\{V_i, i = 1, 2, ..., k\} \quad (10)$$

where k is the number of neighboring neurons of the current neuron P_i which is also all the possible next motion directions of the current location P_i. Because target positions of the external input are positive in the neurons, while the obstacle positions of the external input are negative, the target positions attract the AUV globally, whereas the obstacle areas just push the AUV away to avoid collisions locally, for more detail it may be referred to [18].

(2) The biologically inspired self-organizing map

After AUV's next motion direction location P_n is obtained by (7) and (10), to avoid the sharp speed jumps of AUV in SOM path planning, weight of SOM is updated reasonably:

$$R_{jk}(t+1) = \begin{cases} T_l, & D_{kjl} \le D_{\min} \\ R_{jk}(t) + \partial \times f(d_m, g) \times (P_n - R_{jk}(t)), & D_{kjl} > D_{\min} \end{cases} \quad (11)$$

$P_n = (x_n, y_n, z_n)$ is AUV's position in the next moving direction, $R_{jk} = (w_{jkx}, w_{jky}, w_{jkz})$ is the SOM neuron output which represents the current AUV position. Compared with the update rule for the winner AUV in the equation (6), as the distance between neighboring grid is short in the grid map, and AUVs always move toward neighboring grid of their current location based on the equation (11), $P_n - R_{jk}$ must be appropriate data and meet AUV's speed control constraint (unit speed one). Therefore, AUVs do not occur the sharp speed jumps. In addition, due to the BINN inhibitory property to obstacles, the path planning method based on BISOM can avoid either static or dynamic obstacles. Thus, the proposed BISOM multi-AUV task assignment and path planning algorithm is applicable in the underwater environments with random motion of target location and obstacles. It is described in the flowchart as shown in Fig.4.

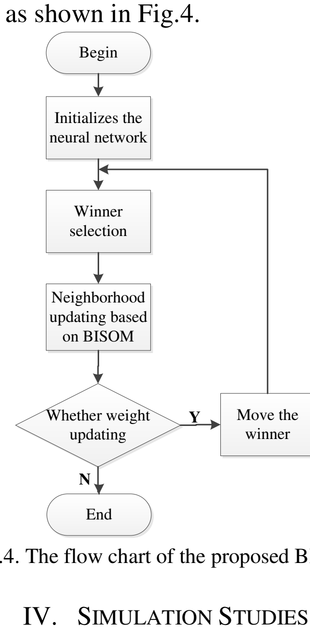


Fig.4. The flow chart of the proposed BISOM

IV. Simulation Studies

To validate the efficiency of the proposed BISOM algorithm, three different scenarios including static and dynamic target locations and obstacle have been studied. AUVs, obstacles and target locations are randomly deployed in the 3D underwater workspace. In all simulation studies, the underwater workspace is designed as 100×100×100. The objective of this paper is to find a strategy for multi-AUV task assignment and path planning, hence each AUV may be regarded as a point without size, and the speed control constraint for each AUV is set as one unit each step. The turning radius for each AUV is negligible in comparison with 3D underwater workspace, in which each AUV is assumed to move omni-directionally. The implementation of the proposed algorithm is based on the precondition that the change speed of target location and obstacle are lower than the AUV speed.

The proposed algorithm is a hybrid method, in which the major parameters of the proposed method can be decided by real world applications like the velocity of AUVs and the space of the environments. The most important parameters that should be set are from the neural networks and there has some work accomplished about the parameters of the BISOM in previous work [18], [19], [21], [22], [23]. The results in previous research show that the proposed algorithm is robust with model parameters variations. Therefore, the parameters can be selected in a relatively broad range. All the chosen parameters in the simulated experiments have been listed in Table 1.

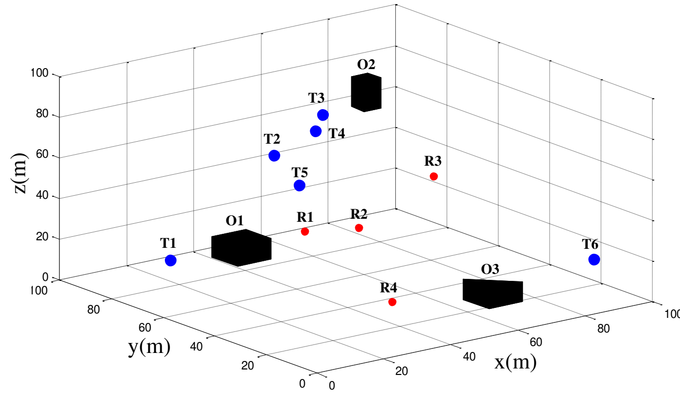
Table 1 Control parameters

Parameters	Value	Description
A	2	the passive decay rate of the BISOM
B	1	the upper bounds of the BISOM
D	1	the lower bounds of the BISOM
E	1	a positive constant
μ	0.3	a positive constant

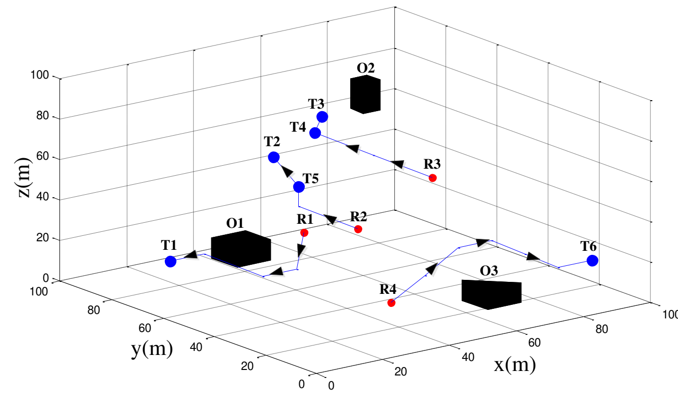
A. *Static environments*

In the first case, there are static target locations in 3D underwater workspace with static obstacles. The initial state is set in Fig.5 (a), where red and blue points represent AUVs' initial positions and target locations, respectively. Four AUVs (R) and six targets (T) are randomly distributed in the underwater workspace. It is noted that in this case the target numbers are greater than the AUV number which implies that part of the AUVs needs to reach more than one target position.

Due to the fact that the traditional SOM algorithm determines the task target by the calculation and comparison of distance between each AUV and the target locations. AUVs choose to reach the closest target position all the time regardless of obstacles and sharp speed jumps. However, for each AUV, it may have some difficulty reaching the assigned target positions with straight line and fastest speed since AUVs may collide obstacles and exceed speed control limitation.



(a) The initial state



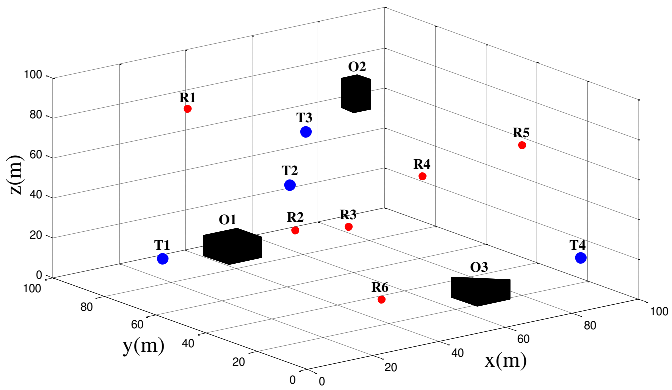
(b) Planned paths

Fig.5. Task assignment and path planning when target locations outnumber AUVs

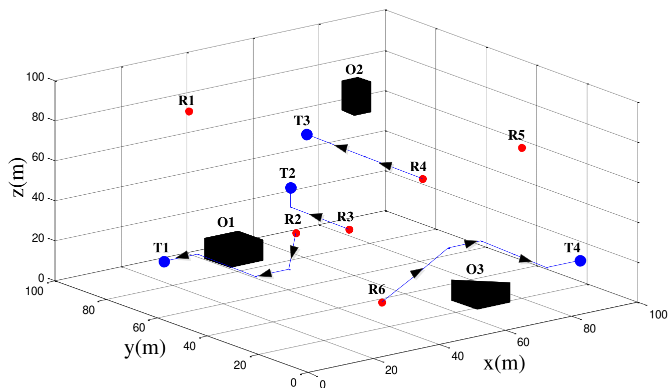
According to the proposed BISOM algorithm, for a constructed biologically inspired neural network in accordance with 3D underwater environments, target locations are set as the maximum neuron activity value and obstacle location's activity value is the minimum. Hence, AUVs can always move towards their target locations without any collisions. In addition, AUVs always move toward neighboring coordinates of their current location based on the BISOM method. $P_n - R_{jk}$ must be appropriate and effective, therefore AUVs never take place the sharp speed jumps.

Fig.5 (b) shows navigation paths of AUVs in the event that the target locations outnumber the AUVs. By SOM task assignment algorithm, $R1$ is arranged for $T1$, $R2$ is arranged for $T2$ and $T5$, $R3$ is arranged for $T3$ and $T4$, and $R4$ is arranged for $T6$. There exist obstacles between $R1$ and $T1$, between $R4$ and $T6$. It is easily to find that the proposed BISOM algorithm generates collision-free paths to reach target locations in Fig.5 (b). However, based on the traditional SOM task assignment and path planning algorithm, $R1$ fails to reach its target location $T1$ and $R4$ does not neither reach its target location $T6$, as they would move along the straight line, and run into obstacles.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCDS.2017.2727678, IEEE Transactions on Cognitive and Developmental Systems



(a) The initial state



(b) Planned paths

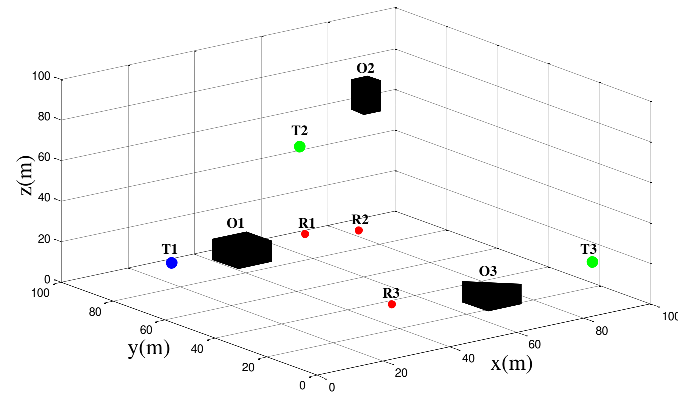
Fig.6. Task assignment and path planning when AUVs outnumber target locations

In addition to the former condition, the proposed BISOM method can also be applied to the case when AUV numbers are more than ones of target locations, meaning that part of the AUVs has no task and keeps still. To confirm the efficiency of the algorithm under different conditions, another simulation result is described in Fig.6. Six AUVs are randomly distributed in the 3D underwater environments, with only four targets. From Fig.6 (b), it is clear to see that the nearest AUVs visit all the corresponding target locations and if it is far away from the target, the AUVs will always keep stationary.

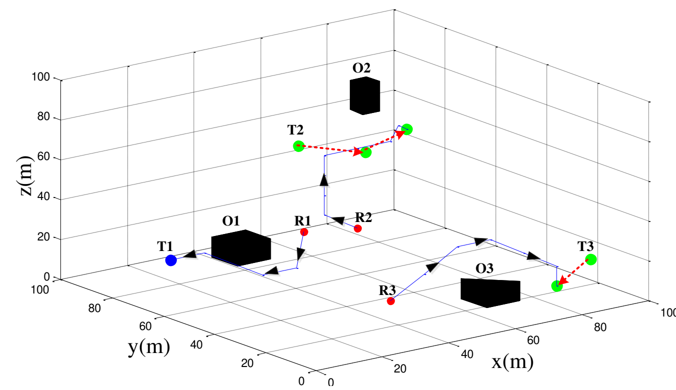
B. Dynamic target locations

For dynamic target locations, the simulated experiment designs three AUVs visiting for equal numbers of target locations. Among them, $T1$ is still static during the whole work process while $T2$ and $T3$ move around randomly. The speeds of $T2$ and $T3$ are lower than that of the AUV. Their initial locations are illustrated in Fig. 7(a).

Since there exist some similarities compared with the static target cases, $R1$, $R2$, and $R3$ are initially assigned to $T1$, $T2$ and $T3$, respectively, according to the principle of the shortest linear distance. However, after ten iterations, $T2$ and $T3$ leave their original position. $T2$ first moves to (30,20,90), then to (50,30,90), and $T3$ moves to (80,10,10). For the environments variation, the proposed BISOM method makes the search task rescheduled and re-plan the paths. By comparison, $R2$ is still assigned for $T2$, and $R3$ for $T3$. Correspondingly, $R2$ and $R3$ change their moving direction moving the motivation towards the original target locations to their new spots. The adjusted trajectory is illustrated in Fig. 7(b).



(a) The initial state



(b) Planned paths

Fig.7. Task assignment and path planning for dynamic target locations

Besides, comparison in terms of AUV speed for $R1$ between SOM and BISOM algorithm is shown in Table 2 and Fig.8 (The red solid lines: the BISOM algorithm; the blue solid lines: the SOM algorithm). In Table 2 and Fig.8, it is clear that the speeds of SOM navigation are far beyond AUV's control constraint (The green dashed lines: unit speed one), but the proposed BISOM algorithm fully meets AUV's control constraint requirements. By analyzing the reason, the next motion direction of the AUV is neighboring coordinate of its current location based on the proposed BISOM algorithm. The mobile speed of AUV must meet its control constraint requirements, and AUVs have no sharp speed jumps occurred. However, the traditional SOM task assignment and path planning algorithm considers no the constraint of AUV speed, thus, AUV speed jump will occur when the error $T_l - R_{jk}$ between AUV current coordination R_{jk} and corresponding target coordination T_l changes suddenly. Especially, at the beginning of AUV navigation, after the winner is selected, $T_l - R_{jk}$ may be very large, thus it must break AUV's control constraint. It is practically impossible for AUV to achieve such a large speed.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCDS.2017.2727678, IEEE Transactions on Cognitive and Developmental Systems

Table.2 Comparison in terms of speed for *R1* between SOM and BISOM algorithms

Iterations / Algorithm	1	2	3	4	5	6	7	8	9	10
SOM	3.852	3.437	3.021	2.671	2.364	2.138	1.937	1.802	1.701	1.603
BISOM	0.838	0.785	0.739	0.698	0.663	0.634	0.619	0.598	0.577	0.558

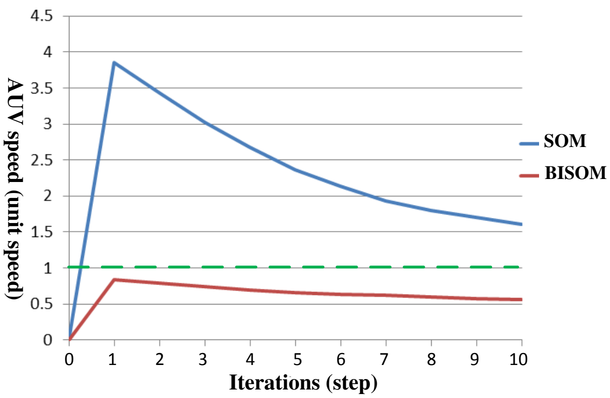
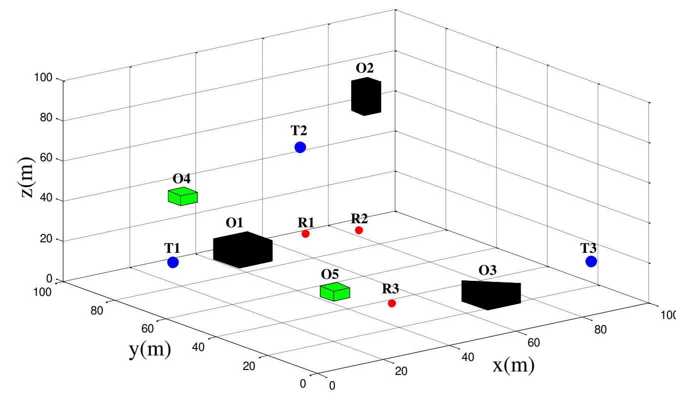


Fig.8. Comparison in terms of speed for *R1* between SOM and BISOM algorithms

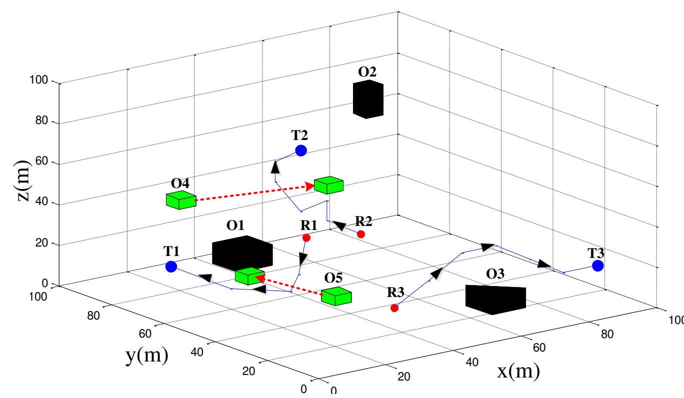
C. Dynamic obstacles

The proposed BISOM algorithm can avoid not only static obstacles, but dynamic ones also. Initial locations for AUVs, target locations and obstacles are depicted in Fig. 9 (a), where $O1$ and $O2$ are two moving obstacles. The speeds of $O1$ and $O2$ are lower than that of the AUV. By taking advantage of SOM, $R1$, $R2$ and $R3$ are initially assigned, respectively, to $T1$, $T2$, and $T3$. At the beginning of the stage, only $O1$, a static obstacle, remains between $R1$ and $T1$. $R1$ moves around obstacle $O1$ *to* reach its target location $T1$. However, in this simulation, the obstacle $O5$ moves to the planned path of $R1$. If $R1$ continuously traverses along original trajectory, a collision with $O5$ would occur. However such a problem can be solved by the BINN, as the BINN is able to adjust $R1$'s next motion direction and SOM's weight function to be updated according to real-time changes in environmental information. It is possible for AUVs to change their moving directions to bypass the mobile obstacles, but impossible to process this situation with the traditional SOM algorithm.

Similarly, with no any obstacle, initially, between $R2$ and $T2$, $R2$ could have moved in the shortest straight line to reach $T2$. Nevertheless when $O4$ stays, BISOM will re-plan a collision-free path to bypass it. Fig.9 (b) shows the trajectories re-planned during the search process. For the traditional SOM algorithm, $R2$ will run into obstacle $O4$ at 150 steps, but $R2$ can bypass obstacle $O4$ and reach its target $T2$ based on BISOM.



(a) The initial state



(b) Planned paths

Fig.9. Task assignment and path planning with dynamic obstacles

Additionally, Comparison in terms of speed for $R2$ between SOM and BISOM algorithms is shown in Table 3 and Fig.8 (The red solid lines: the BISOM algorithm; the blue solid lines: the SOM algorithm). In Table 3 and Fig.10, navigation speed of $R2$ for SOM is beyond AUV's control constraint requirements (The green dashed lines: unit speed one). Whereas the proposed BISOM algorithm can meet AUV's control constraint requirements, AUVs have no any sharp speed jumps. This is because the next motion direction of the AUV is adjacent coordinate of its current location based on the proposed BISOM algorithm, and mobile speed of AUV must meet its control constraint requirements.

Table.3 Speed comparison for *R2* between SOM and BISOM algorithms

Iterations / Algorithm	1	30	60	90	120	150	180	210	240	270
SOM	4.536	3.358	2.857	2.281	1.896	-	-	-	-	-
BISOM	0.956	0.753	0.567	0.394	0.272	0.169	0.104	0.063	0.029	0

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCDS.2017.2727678, IEEE Transactions on Cognitive and Developmental Systems

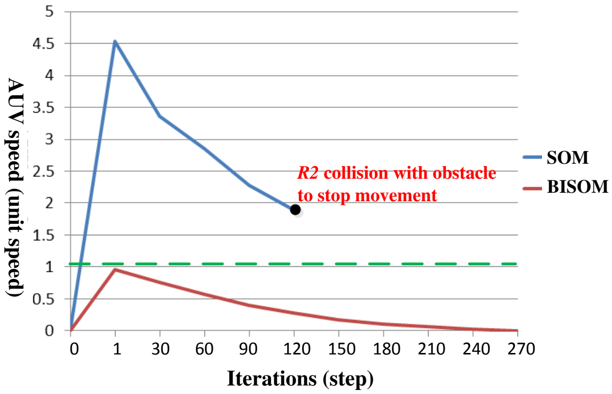
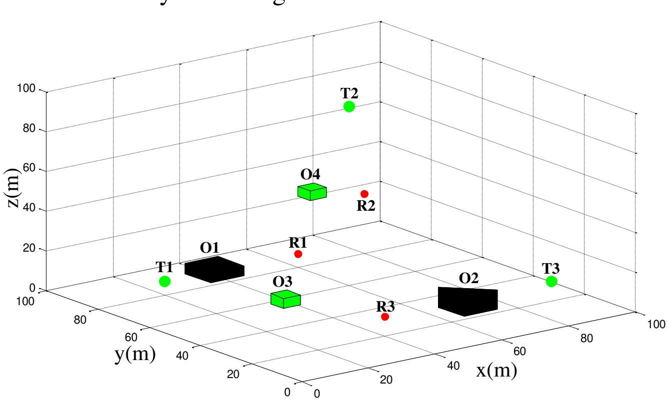


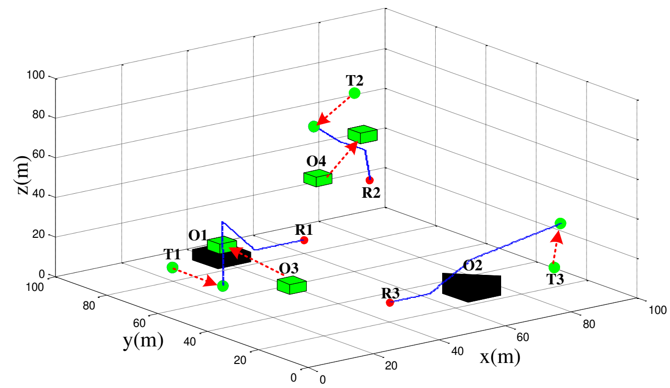
Fig.10. Speed comparison for *R2* between SOM and BISOM algorithms

D. Dynamic obstacles and dynamic targets

In order to further validate the performance of the proposed algorithm, the simulated experiment designs three AUVs that visit for equal numbers of dynamic target locations in the 3D underwater environments with dynamic obstacles. Initial locations for AUVs, target locations and obstacles are depicted in Fig. 11 (a). Among them, $T1$, $T2$ and $T3$ move around randomly, $O3$ and $O4$ are two moving obstacles. The speeds of targets and obstacles are lower than that of the AUV. By taking advantage of SOM, $R1$, $R2$ and $R3$ are initially assigned, respectively, to $T1$, $T2$, and $T3$. At the beginning of the stage, targets and obstacles remain static, in light of the principle of the shortest linear distance. However, after a period of time, targets (to $T1$, $T2$, and $T3$) and obstacles ($O3$ and $O4$) leave their original position. For the environments variation, the proposed BISOM method reschedules the search mission and replan the paths. Correspondingly, $R1$, $R2$ and $R3$ change their moving direction from the motivation towards the original target locations to their new spots, and bypass the mobile obstacles. The adjusted trajectory is illustrated in Fig.11 (b). From this simulation, it verifies BISOM has the ability to complete task assignment and path planning with dynamic obstacles and dynamic targets.



(a) The initial state

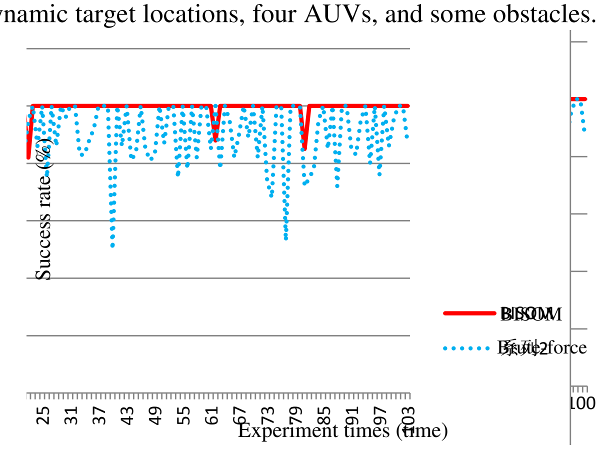


(b) The motion paths of AUVs, targets and obstacles

Fig.11. Task assignment and path planning with dynamic obstacles and dynamic targets

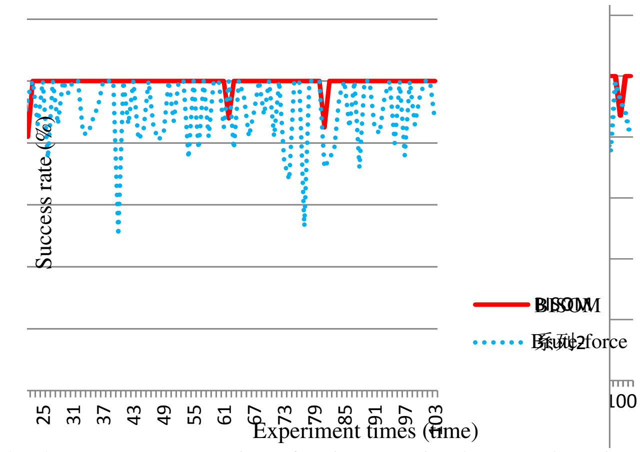
E. Comparison of different algorithms

To verify the success rate of BISOM with the task assignment and path planning, comparison studies with the brute-force algorithm will be conducted in this section. The brute force algorithm, which is also called the "naïve" is the simplest algorithm that can be used in path planning. The idea is that the AUV location and target location are compared to check if they are equal. In the case of a mismatch, the AUV moves to the next position in the direction of the target and comparison is repeated, until a target location is reached. The algorithm works with two variables: an "AUV location" and a "target location". For all possibly valid locations, AUV and target are compared, while AUV location and target location is the same, the path planning is completed [24]. The comparison studies involve eight static target locations, five AUVs, and some obstacles with environments scale of $100\times100\times100$. The target locations, AUVs and obstacles are randomly deployed. Different algorithms are used to arrange the multi-AUVs to visit all appointed targets. In order to ensure the accuracy of the experiments, we conducted the experiments many times. In each experiment, the positions of the obstacles, targets and AUVs are reset. The success rate of performing 100 times of task assignment and path planning using two different algorithms is depicted in Fig.12 (a). Additionally, Fig.12 (b) shows the experimental result of environments involve seven dynamic target locations, four AUVs, and some obstacles.



(a) The success rate comparison of environment involve eight static target locations, five AUVs, and some obstacles

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCDS.2017.2727678, IEEE Transactions on Cognitive and Developmental Systems



(b) The success rate comparison of environments involve seven dynamic target locations, four AUVs, and some obstacles

Fig.12 Success rate comparison between BISOM and brute-force algorithms

It is very clear to see that the proposed BISOM algorithm reaches 100% success rate under a large number of experiments in Fig. 12. It means that the most tasks are successfully executed. However, the brute-force algorithm only reaches 100% success rate for few experiments. In some special cases, the success rate of brute-force algorithm is only 50%. By comparison, it is found that under certain circumstances, the success rate of the proposed BISOM algorithm is below 100% but it is still superior to the brute-force algorithm. By analysis, the brute-force algorithm only provides optimum solution under no obstacle conditions. However, BISOM works properly for obstacle avoidance, therefore it deserves a high success rate in the environments filled with obstacles.

In order to further validate the performance of the proposed algorithm, comparison studies with the potential field-based particle swarm optimization (PPSO) algorithm will be carried out. The potential field is implemented in the fitness function of the particle swarm optimization (PSO), which provides with reasonable evaluations of the underwater environments. The PSO algorithm plans a path by iteratively improving a candidate solution with regard to the fitness function [27]. The potential function value calculation is based on the attractive potential function and repulsive potential function, which are mainly related to the target relative distance and velocity, and the obstacles relative distance. After the calculation of the attractive and repulsive potential, the fitness function can be obtained by

$$f(p)=U_{att}(p,v)+U_{rep}(p,v) \tag{12}$$

where $f(p)$ is the potential function; p is the position of particle; $U_{att}(p, v)$ denote the attractive potential function generated by the target; $U_{rep}(p, v)$ denotes the repulsive potential function generated by the obstacle; v is the velocity of particle. The attractive potential field functions are presented as follows

$$U_{att}(p,v)=\alpha_p\left\|p_{tar}-p\right\|^2+\alpha_v\left\|v_{tar}-v\right\|^2 \tag{13}$$

where p_{tar} denote the positions of the target; v and v_{tar} denote the velocities of the particle and the target, respectively; $\left\|p_{tar}-p\right\|$ is the Euclidean distance between the particle and the target; $\left\|v_{tar}-v\right\|$ is the magnitude of the relative velocity between the target and the particle; α_p and α_v are scalar positive parameters.

The repulsive potential can be defined as follows

$$U_{rep}(p,v)=\begin{cases}\eta\left(\dfrac{1}{\left\|p_{obs}-p\right\|-v^2/2a}-\dfrac{1}{\rho_0}\right), & 0<\left\|p_{obs}-p\right\|-v^2/2a<\rho_0\\ 0, & \left\|p_{obs}-p\right\|-v^2/2a\geq\rho_0\end{cases} \tag{14}$$

where ρ_0 is a positive constant describing the influence range of the obstacle; $\left\|p_{obs}-p\right\|$ is the Euclidean distance between the particle and the obstacle; deceleration of magnitude a is applied to the particle to reduce its velocity; and η is a positive constant. In the simulate experiment, population size is 10; the parameters are set as α_p=0.3, α_v=0.1, ρ_0=0.5 and η= 0.6.

The comparison studies involve six dynamic target locations, three AUVs, and some obstacles with environments scale of $100\times100\times100$. The target locations, AUVs and obstacles are randomly deployed. Two algorithms are applied to the multi-AUVs that are directed to visit all the appointed targets. In these conditions, two algorithms respectively complete 50 times simulation experiment of task assignment and path planning. To make a clear distinction between the two algorithms, Table 4 lists the mean and standard deviation statistics of path length, time of complete the task, energy consumption by both algorithms. It is reasonable to conclude that the integrated algorithm of BISOM performs better than the PPSO in each item of simulation results. Hence, it distinguishes itself with the shorter path length and time, and lower energy consumption [28]. By analysis, the PPSO algorithm doesn't have the function of task allocation for AUVs. However, BISOM can not only perform the path planning and complete task assignment, but also it can better complete the task in the environments filled with obstacles.

Table.4 Performance comparison between BISOM and PPSO

Performance / Algorithm	Path length	Time of complete the task	Energy consumption
BISOM	342.47±25.15	215.97±18.43	281.53±22.16
PPSO	467.19±29.83	277.32±21.74	399.48±27.85

V. Conclusion

In this paper, a novel integrated BISOM algorithm associated with the self-organizing map neural networks and biologically inspired neural networks is proposed to resolve the task assignment and path planning issue of a multi-AUV system. The simulated experiments prove the proposed BISOM algorithm is feasible and effective for 3D underwater workspace with obstacles. It can not only avoid obstacle, but also overcome the sharp speed jump of the SOM algorithm.

Though the proposed method has the superiority performance, there are still some problems that have not been fully considered such as ocean currents, communications, localization, etc. It needs a further study for complex underwater environments and actual AUV systems with the method like deep learning and reinforcement learning [25-26]. In addition, in order to achieve a better task assignment and path planning of the multi-AUV systems, cooperation among each AUV needs to be strengthened in the future as well.

REFERENCES

[1] L. Paull, S. Saeedi, M. Seto and H. Li, "AUV navigation and localization: a review," *IEEE J Oceanic Eng.*, vol.39, no.1, pp.131-149, Jan. 2014.

[2] E. Fiorelli, N. Leonard, P. Bhatta, D. Paley, R. Bachmayer and D. Fratantoni, "Multi-AUV control and adaptive sampling in Monterey Bay," *IEEE J Oceanic Eng.*, vol.31, no. 4, pp. 935-948, Oct. 2006.

[3] I. S. Kulkarni and D. Pompili, "Task allocation for networked autonomous underwater vehicles in critical missions," *IEEE J Sel. Area. Comm.*, vol. 28, no.5, pp.716-727, Jun. 2010.

[4] L. Jaillet and J. M. Porta, "Path planning under kinematic constraints by rapidly exploring manifolds," *IEEE Trans. Robot.*, vol. 29, no.1, pp.105-117, Feb. 2013.

[5] S. Petillo and H. Schmidt, "Exploiting adaptive and collaborative AUV autonomy for detection and characterization of internal waves," *IEEE J Oceanic Eng.*, vol.39, no.1, pp. 150-164, Jan. 2014.

[6] C. Petres, Y. Pailhas, P. Patron, Y. Petillot, J. Evans and D. Lane, "Path planning for autonomous underwater vehicles," *IEEE Trans. Robot.*, vol.23, no.2, pp. 331-341, April. 2007.

[7] Y. Deng, P P J Beaujean, E. An and C. Edward, "Task allocation and path planning for collaborative autonomous underwater vehicles operating through an underwater acoustic network," *J. Robotics*, vol. 2013, pp. 1-15, 2013.

[8] Y. T. Tian, M. Yang, X. Y. Qi and Y. M. Yang, "Multi-robot task allocation for fire-disaster response based on reinforcement learning," *The 8th Int. Conf. Machine Learn. Cybern.*, Baoding, China, pp. 2312-2317, Jul, 2009.

[9] A. Zhu and S. X. Yang, "A neural network approach to dynamic task assignment of multi-robot," *IEEE Trans. Neural Networ.*, vol. 17, no. 5, pp. 1278-1287, Sept. 2006.

[10] A. Zhu and S. X. Yang, "An improved SOM-based approach to dynamic task assignment of multi-robots," *World Congress Intell. Control Automat.*, Jinan, China, pp.2168-2173, Jul, 2010.

[11] X. Cao and D. Q. Zhu, Multi-AUV Task Assignment and Path Planning with Ocean Current Based on Biological Inspired Self-organizing Map and Velocity Synthesis Algorithm, *Intell. Autom. Soft Co.*, 2015, doi: 10.1080/ 10798587.2015.1118277:1-16.

[12] M. Sato, A. Kanda and K. Ishii, "A switching controller system for a wheeled mobile robot," *J. Bionic Eng.*, vol.4, no.4, pp. 281-289, Dec. 2007.

[13] Z. Hendzel, "Collision free path planning and control of wheeled mobile robot using self-organizing map," *B. Pol. Acad. Sci-Tech.*, vol. 53, no. 1, pp.39-47, 2005.

[14] D. Jung, J. Ratti and P. Tsiotras, "Real-time implementation and validation of a new hierarchical path planning scheme of UAVs via hardware-inthe-loop simulation," *J. Intell. Robot. Syst.*, vol. 54, no. 1-3, pp. 163-181, 2009.

[15] L. Zhong, Q. Luo, D. Wen, S. D. Qiao, J. M. Shi and W. M. Zhang, "A task assignment algorithm for multiple aerial vehicles to attack targets with dynamic values," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 236-248, 2013.

[16] D. Q. Zhu, H. Huan and S. X. Yang, "Dynamic task assignment and path planning of multi-AUV system based on an improved self-organizing map and velocity synthesis method in three-dimensional underwater workspace," *IEEE Trans. Cybernetics*, vol. 43, no. 2, pp. 504-514, Apr. 2013.

[17] H. Huang, D. Q. Zhu and F. Ding, "Dynamic task assignment and path planning for multi-AUV system in variable ocean current environment," *J. Intell. Robot. Syst.*, vol.74, no.2, pp. 999-1012, Jun. 2014

[18] S. X. Yang and C. Luo, "A neural network approach to complete coverage path planning," *IEEE Trans. Syst. Man Cy. B.*, vol. 34, no. 1, pp.718-725, Feb. 2004.

[19] H. Li, S. X. Yang, and M. L. Seto, "Neural-network-based path planning for a multirobot system with moving obstacles," *IEEE Trans. Syst. Man Cy. C.*, vol. 39, no. 4, pp. 410-419, Jul. 2009.

[20] M. Z. Yan, D. Q. Zhu and S. X. Yang, "A novel 3-D bio-inspired neural network model for the path planning of an AUV in underwater environments," *Intell. Autom. Soft Co.*, vol.19, no. 4, pp.555-566, 2013.

[21] C. Luo and S. X. Yang, "A bioinspired neural network for real-time concurrent map building and complete coverage robot navigation in unknown environments," *IEEE Trans. Neural Networ.*, vol.19, no.7, pp.1279-1298, Jul. 2008.

[22] J. J. Ni and S. X. Yang, "Bioinspired neural network for real-time cooperative hunting by multirobots in unknown environments," *IEEE Trans. Neural Networ.*, vol.22, no.12, pp.2062-2077, Dec. 2011.

[23] S. X. Yang, "Neural network approaches to real-time motion planning and control of robotic systems," *Ph.D. dissertation, Dept. Elect. Comput. Eng., Univ. of Alberta*, Edmonton, Canada, 1999.

[24] W. Y. Wang and Y. C. Jiang, "Community-aware task allocation for social networked multiagent systems," *IEEE Trans. Cybernetics*, vol.44, no. 9, pp.1529-1543, Sept. 2014.

[25] Y. Liao, S. Kodagoda and Y. Wang, "Place classification with a graph regularized deep neural network," *IEEE Trans. Cognitive and Developmental Syst.*, DOI:10.1109/TCDS.2016.2586183, 2016.

[26] V. Meola, D. Caligiore and V. Sperati, "Interplay of rhythmic and discrete manipulation movements during development: a policy-search reinforcement-learning robot model," *IEEE Trans. Cognitive and Developmental Syst.*, DOI:10.1109/TAMD.2015.2494460, 2015.

[27] X. Cao, D. Q. Zhu and S. X. Yang, "Multi-AUV cooperative target searching under ocean current based on PPSO approach," *Underwater Technology*. 2015, 33(1):31-39.

[28] C. Luo, S. X. Yang, X. Li, M. Q. H. Meng, "Neural-dynamics-driven complete area coverage navigation through cooperation of multiple mobile robots", *IEEE Trans. Ind. Electron.,* vol.64, no. 1, pp. 750-760, Jan, 2017.

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.

Xiang Cao was born in Sichuan, China. He received the B.Sc. degree in electronic and information engineering from Southwest University, Chongqing, China, in 2004, and the M.Sc. degree in communication and information systems from Shanghai Maritime University, Shanghai, China, in 2011, and the Ph.D. degree in power electronics and power transmission from Shanghai Maritime University, Shanghai, China, in 2016. His current research interests include target searching and path planning of underwater vehicles

Bing Sun was born in was born in Haimen, China. He receivedthe 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 in the Colleague of Information Engineering, Shanghai Maritime University.

His current research interests include tracking control and fault-tolerant control of underwater vehicles.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCDS.2017.2727678, IEEE Transactions on Cognitive and Developmental Systems

Chaomin Luo (S'01–M'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.