

# Developing an Energy Effective Autonomous USV for Undertaking Missions at the Highlands of Peru

1<sup>st</sup> Rui Song

*Department of Mechanical Engineering  
University College London  
London, UK  
r.song.11@ucl.ac.uk*

4<sup>th</sup> Francisco Cuellar

*Department of Engineering,  
Mechatronics Section  
Pontificia Universidad Catolica del Peru  
San Miguel - Lima, Peru  
cuellar.ff@pucp.pe*

2<sup>nd</sup> Yuanchang Liu

*Department of Mechanical Engineering  
University College London  
London, UK  
yuanchang.liu.10@ucl.ac.uk*

5<sup>th</sup> Richard Bucknall

*Department of Mechanical Engineering  
University College London  
London, UK  
r.bucknall@ucl.ac.uk*

3<sup>rd</sup> Jose Balbuena

*Department of Engineering,  
Mechatronics Section  
Pontificia Universidad Catolica del Peru  
San Miguel - Lima, Peru  
jose.balbuena@pucp.pe*

**Abstract**—This paper describes a novel multi-task allocation method for the autonomous navigation to improve the efficiency for executing mission considering an Unmanned Surface Vehicle (USV) developed by the Pontificia Universidad Catolica del Peru (PUCP). The new method is developed based upon the self-organizing map (SOM) algorithm, with the consideration of the priorities of the sample stations that USV need to visit, as well as the lattice distances from the sample stations to the start point. Using this new method, an optimized order of visiting sequence can be calculated according to the battery energy limit of the USV. The new multi-task allocation method has been verified in simulation environments with results proving the effectiveness and capabilities of the system.

**Keywords**—System identification, Autonomous navigation, Task allocation, Energy saving, Highland Peruvian lakes

## I. INTRODUCTION

It is well known that Unmanned Surface Vehicles (USVs) offer benefits by reducing human resource costs and operating as ‘force multipliers’ with enlarged exploration range, especially in research applications such as hydrographic survey and water quality monitoring. With the increasing interests in the development of USVs to concur complex tasks in constrained environments, it has become important to manufacture USVs so they are able to carry out the tasks in an energy-efficient way.

Regardless of the autonomy level of USVs, most of the existing platforms are powered by batteries [1, 2]. In [3], a remote controlled USV has been developed by the Pontificia Universidad Catolica del Peru (PUCP), and used as a scientific platform to undertake the water quality measurement task at Peruvian highland lakes. The main feature of this USV is that it can be easily assembled and disassembled, which facilitates ease of transporting the vehicle to different locations to undertake missions. However, this transportability limits the size of the USV as well as its battery capacity. Therefore, it is important to improve the efficiency for executing missions to reduce the power consumption and consequently save energy.

Previous research applied path planning algorithms [4-6] to generate an optimized path with the shortest distance for the

USV to follow thereby reducing the energy cost. Most of these algorithms address the problem with only one mission (end) point. However, for most data collection applications, like water monitoring and pollution tracking, where more than one mission measuring points are needed, a multi-task allocation method is required to assist with generating the most efficient task execution strategy. This kind of applications can be regarded as the classic Traveling Salesman Problem (TSP) that aims to find a shortest travel sequence connecting all the cities (mission points). Note that in the TSP, each mission point will be visited only once and return to the start point when the travel is terminated. To solve the TSP, a two-layered heuristic algorithm, the self-organizing map (SOM) based neural network has been applied and modified by some researchers with advantages of relative simplicity, high computational speed and promising performance.

In [7], a dynamic ring structure has been proposed to add or delete the neurons on the ring adaptively according to the specific training situation. [8] further improved the SOM by generating repulsive force field around obstacle areas. In such a case, the neurons are expelled away from obstacles during updating process to ensure the safety. Moreover, a new adaptive strategy was designed by [9], called the prize-collecting traveling salesman problem, with additional penalties given to certain mission points, which locations will be skipped during the travel.

However, all these algorithms do not consider the energy constraint during adaption. The USV may still run out of power during the journey and needs to transit back to the start point with some mission points neglected. In addition, in [8], it has been noticed that although the obstacle avoidance can be achieved by integrating a repulsive force field in the SOM, the influence area of the repulsive force field should be constrained in a certain level, otherwise, the convergence of the algorithm will be limited. To solve these problems, in this paper, a new penalty is given to each mission point as selecting criteria so that mission points can be adaptively chosen to meet the energy requirement. Moreover, a propagation scale limit [10] is

introduced to constrain the dimension of the repulsive force field.

This paper is organized as follows. Section II describes autonomous navigation system of the USV developed by PUCP. Section III explains the proposed new multi-task allocation method in detail. Results are shown in Section IV with discussions. This paper is concluded in Section V.

## II. AUTONOMOUS NAVIGATION SYSTEM

A new autonomous navigation system has been developed in this paper as shown in Fig. 1. It includes an additional power monitoring unit in order to record the real-time battery consumption and the requested power for actuation. Using this battery information along with the data received from on-board navigational sensors, a multi-task as well as a path planning algorithm are integrated with the main controller to guide the USV to accomplish missions effectively.

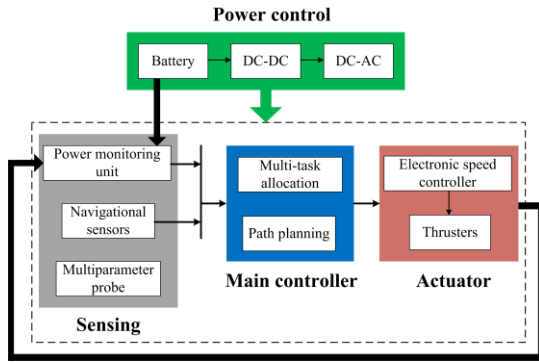


Fig. 1. System diagram of the new autonomous navigation system for the PUCP's USV.

Before the integration, the USV's dynamic model is obtained using the MATLAB built in System Identification (SI) toolbox, and analysing the data obtained from several pre-done field tests (Circle and Zig-Zag movements). The field tests of the USV were performed at lake Languilayo in Cuzco. Fig. 2 is a picture when the PUCP USV is in operation. For data acquisition, several inputs including a pseudorandom binary sequence are applied to the thrusters and the heading response is recorded. The input to the SI is the differential mode thruster velocity  $n_d$ , which causes the vehicle to manoeuvre as required ( $n_c$  is maintained to conserve the operating regime). The acquired data is processed and down-sampled to 1 Hz since this frequency is deemed to be adequate for controller design noting that the dynamic model for the USV changes with the environment.



Fig. 2. Field testing at Lake Languilayo, Languilayo Peru at 3955 MASL.

## III. MULTI-TASK ALLOCATION METHOD

### A. Problem Statement

In practical missions, such as the water quality monitoring mission considered in this paper, due to the spatial distribution of the mission points, the measurements retrieved from one point could also be included in the information provided by other nearby points [9] making it not obligatory for the USV to transit through all points to complete the mission. Therefore, in such a case, differing from the conventional multi-task allocation problem, where the complete visiting of all mission points is compulsory, the problem to address the energy-efficient autonomous navigation should consider: 1) selecting the appropriate mission points according to mission requirements; 2) finding a mission execution sequence to complete the selected mission points; 3) visiting the mission points while avoiding obstacles *en route*; 4) re-planning the execution sequence by considering the energy constraints. To simultaneously address these problems together they can be formulated as a modified TSP problem as:

There are  $n$  possible mission points  $\mathbf{G} = [g_1, \dots, g_n]$ , where  $g_i \in \mathbb{R}^2$ . Each  $g_i$  has an assigned priority  $P(g_i) \geq 0$ , and a distance penalty  $\zeta(g_i) \geq 0$ . The distance between two mission points  $g_i$  and  $g_j$  is  $c(g_i, g_j) \geq 0$ . As shown in Fig. 3, the problem is therefore to find a travel tour  $T$  and visiting a subset of the mission points  $G_T \subseteq \mathbf{G}$ .  $T$  contains a sequence of the selected visited mission points, namely,  $T = [g_{s_1}, \dots, g_{s_k}]$ . Note that,  $g_{s_i} \in G_T$  and  $1 \leq s_i \leq n$ . The total cost of completing the tour ( $C(T)$ ) is then calculated as:

$$C(T) = \sum_{(g_{s_i}, g_{s_{i+1}}) \in T} c(g_{s_i}, g_{s_{i+1}}) + \sum_{g \in G \setminus G_T} P(g) + \sum_{g \in G \setminus G_T} \zeta(g). \quad (1)$$

In this paper,  $C(T)$  is governed by the battery limit  $\beta$ , namely,  $C(T) \leq \beta * F$ , where  $F$  is full battery allowance. Note that,  $T$  herein is a closed tour, which means that  $g_{s_1} = g_{s_k}$ .

For simplicity,  $g_{s_1}$  is set as the USV start point. The travel cost between two mission points is computed as the Euclidean distance  $c(g_{s_i}, g_{s_{i+1}}) = |(g_{s_i}, g_{s_{i+1}})|$  in a planar environment.

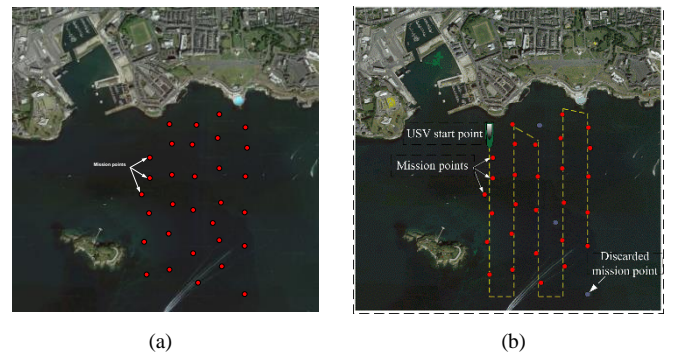


Fig. 3. (a) Pre-deployed water sample stations in a maritime environment. (b) Multiple tasks visited by the USV, with some stations, marked in blue being discarded.

## B. SOM Algorithm

The SOM [11] is a two-layered artificial neural network as shown in Fig. 4, and it implements the unsupervised learning strategy to train the neurons in the output layer according to the input information. Neurons in the output layer are normally connected using certain topology such as rectangular and hexagonal grid, and due to the nature of the TSP problem, where a closed visiting loop is requested, a ring form topology is therefore applied in this paper. During the training process, neurons are competitively adapted according to the mission points contained in the input layer. As mentioned previously, such a ring form make the USV capable of starting and finishing the whole mission from the same point, which facilitates the launch and recovery of the USV.

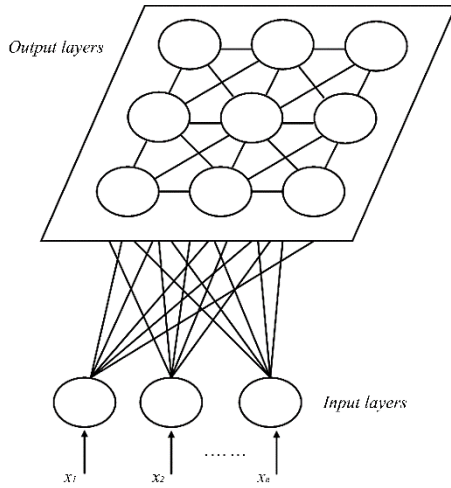


Fig. 4. The SOM scheme, where  $x_i$  is the task point (source: [12]).

When applying the SOM for multi-task allocation, each mission point  $g_i$  in the input layer is associated with a Cartesian coordinate  $(g_x, g_y)$  representing its location. The weights of the neurons on the output space are in the same dimension and indicates the locations of neurons. Throughout the training process, neurons compete to become the winner.

The unsupervised learning strategy consists of two different steps:

1) *Winner selection*: A mission point ( $g_i$ ) is selected from the input layer. The Euclidean distance between this mission point and each neuron in the output space is calculated by:

$$d_{gv} = |g_i - v_j| = \sqrt{(g_{xi} - v_{xj})^2 + (g_{yi} - v_{yj})^2}, \quad (2)$$

where  $v_j$  is the location of the  $j$ th neurons. Note that the start point is set as the first selected mission point. The winner neuron is then selected as the one having the minimum value of  $d_{gv}$ , namely:

$$v_{win} = \operatorname{argmin}_v(d_{gv}). \quad (3)$$

2) *Neighbourhood updating*: After the winner neuron  $v_{win}$  has been determined, the neighbourhood updating step will move the  $v_{win}$  as well as its neighbour neurons towards new

positions with different scales. In the conventional SOM algorithm, such a updating step is calculated without considering obstacle avoidance making it unsuitable to be directly implemented onto USV applications. Therefore, in this paper, the improved SOM algorithm proposed in [8] has been adopted, and to successfully address the collision avoidance problem, the new updating step is calculated as:

$$v_j = \begin{cases} v_j + \mu * f(d, \sigma) * \left( \frac{g_i - v_j}{\|g_i - v_j\|} + F_{rep} \right), & \text{if } d_{obs} \leq d_{min} \\ v_j + \mu * f(d, \sigma) * \frac{g_i - v_j}{\|g_i - v_j\|}, & \text{otherwise} \end{cases}, \quad (4)$$

where  $\mu$  is the learning rate that determines the computational time.  $f(d, \sigma)$  is the neighbouring function that identifying the size of the neighbourhood of the winner neuron. According to [8],  $f(d, \sigma)$  herein is defined as:

$$f(d, \sigma) = e^{-(d^2/\sigma^2)}, \quad (5)$$

where  $d$  is the lattice distance on the ring topology, and  $\sigma$  is the neighbouring function variance (or the gain parameter).

Note that as the SOM is an iterative adaption process, during each iteration,  $\sigma$  should be monotonically decreasing as time increases. This is due to the reason that during the initial stages, neurons stay far away from the mission points. A relative large  $\sigma$  value is preferred to move the neurons fast towards selected mission point. While at the latter epoch, neurons will be close to mission points and formed a relative stable situation. The competition between neurons become less, and a small value of  $\sigma$  should be used. In such a case,  $\sigma$  is defined as:

$$\sigma = (1 - 0.0005t)\sigma. \quad (6)$$

The novelty of the method proposed in [8] is the repulsive force field  $F_{rep}$  included in (4). Such a force field is added in the neurons updating process to assist with the collision avoidance. It should be noted that in the maritime environment, obstacle areas can be varied in different times, i.e. rising tide makes obstacle area shrink; whereas, decaying tide enlarges the obstacle dimension. Therefore, the influence area of the force field in [8] can be adaptively changed, and such an adaption method will be explained in the next subsections together with other important modifications to the SOM approach introduced in this paper.

## C. Modifications of the SOM algorithm

1) *Constraint  $F_{rep}$* : The adaptive repulsive force field is constructed by running the fast marching (FM) method with the introduction of a constraint factor  $\alpha$  as:

$$\begin{aligned} D_{rep}(x, y) &\leftarrow \text{FM}(\vec{p}_{obs}, \alpha) \\ F_{rep} &= \nabla(D_{rep}) \end{aligned}, \quad (7)$$

where  $\vec{p}_{obs}$  is the obstacles' locations relative to the coordinate of map.  $\text{FM}(\cdot, \alpha)$  represents the procedure of running the FM method [13] constrained by factor  $\alpha$ . In this process, after running the FM method, potential values of all points  $D(x, y)$  in the domain of interest will be compared with  $\alpha$ , and any values higher than  $\alpha$  will be reset to  $\alpha$ , which has been mainly introduced to control the influence area of obstacles. For example, as shown in Fig. 5, when  $\alpha$  decreases, the influence

area of obstacle also decreases. Also, the higher the potential, the safer the position is.

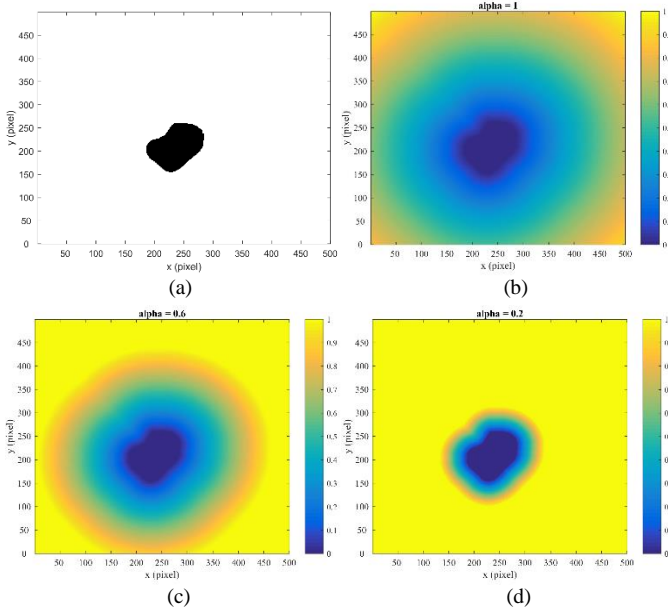


Fig. 5. Changes of potential field with different  $\alpha$  values on a map of  $500 \times 500$  grids. (a) Grid map with one obstacle. (b), (c) and (d) are the potential fields when  $\alpha = 1, 0.6$  and  $0.2$ , respectively. The colour from blue to yellow represents the influence strength of the obstacle, where blue indicates the highest influence and yellow means there is no influence.

2) *Measurement priority*  $P(g_i)$ : Due to the limited energy onboard, mission points should be prioritised such that points containing important measurements data should be assigned with high priority and visited by the USV in the first place. In this paper, the updating process proposed in [14] has been employed to allocate the missions according to different priorities. Considering the predefined priorities of measurements at different mission point, the winner neuron selected in the aforementioned step is first chosen as a winner candidate. The candidate  $v_s$  for the current mission point  $g_i$  will become the winner only if its Euclidean distance to  $g_i$  is shorter than  $P(g_i)$ , namely  $|(v_s, g_i)| \leq P(g_i)$ . Otherwise,  $v_s$  will be discarded.

3) *Distance penalty*  $\zeta(g_i)$ : To govern the total travel cost compliant with the battery limit, mission points to be visited need further refinement while the USV is navigating. With the continuous energy consumption, initially allocated mission points could be possibly discarded as the USV may run out of battery to reach them. Such a mission points selection process can be summarised as: during current epoch  $t$ , if the governing criteria  $C_t(T) \leq \beta_t * F$  is not satisfied, then the  $g_i$  with lower priority and higher  $\zeta$  will be discarded first. The mission point will be removed from the input layer until the governing criteria is satisfied.

The pseudo-code of the modified SOM algorithm is described in Algorithm 1.

#### Algorithm 1 Modified Self-organising Map Algorithm

**Input:** set of mission points  $G$ , parameters  $(\mu, d, \sigma, \alpha, d_{min}, error_{min})$

- 1: Initialise  $N$  neurons of SOM with a ring topology as  $v = (v_1, \dots, v_N)$ , iter  $\leftarrow 1$
- 2: **while**  $error \neq error_{min}$  **do**
- 3:     Select  $g_i$  from  $G$ ,  $i$  start from 1 to  $n$
- 4:     Find  $v_{win}$ : **if** iter == 1 **then**
- 5:         choose  $v_{win}$  according to (2) and (3)
- 6:         **else** choose  $v_{win}$  according to (2) and (3) such that  

$$|(v_s, g_i)| \leq P(g_i)$$
- 7:         **end if**
- 8:     **for** each neighbourhood neurons  $v_j$  of the winning neuron  $v_i$
- 9:         **do** neighbourhood updating according to (4)
- 10:     **end for**
- 11:     iter  $\leftarrow$  iter + 1
- 12: **end while**
- 13: **if**  $C_t(T) > \beta_t * F$  **then**
- 14:     resort  $\zeta(g)$  in descending order
- 15:     discard  $g_i$  from  $G$  where  $\zeta(g_i) = \zeta(1)$
- 16: **end if**
- 17: **return** to step 2.

**Output:**  $v$

#### IV. SIMULATION RESULTS

In order to test the proposed modified SOM multi-task allocation method, three simulations are carried out in an environment containing 18 islands regarded as the obstacles. Such a simulation environment map with the size of  $500 \times 500$  m is shown in Fig. 6. A total number of 30 mission points marked in blue have been predefined and randomly located in the map. To run the simulations, this environment map is further rasterised into a binary grid map with the size of  $500 \times 500$  pixels, where 1 pixel = 1 m.

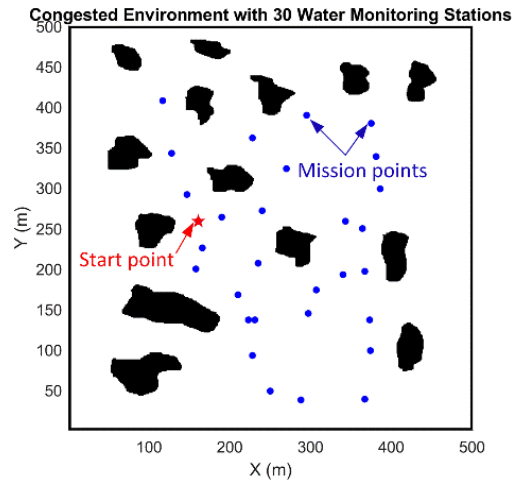


Fig. 6. Simulation environment map with the size of  $500 \times 500$  m. The obstacles are represented in black. Blue dots are the mission points (water monitoring stations). The USV start point is marked in shaded red star.

The three tests are:

1) *Test the ability of saving travel distance cost:* In this test, the modified SOM method is run in a situation, where all the mission points are required to be visited with the total travel distance cost marked as  $d_{total}(SOM)$ .  $d_{total}(SOM)$  is compared with the total travel distance cost of another 100 randomly generated sequence of mission points. The total distance comparison result is shown in Fig. 7. It can be seen that using the modified SOM method, the total distance cost can be largely reduced.

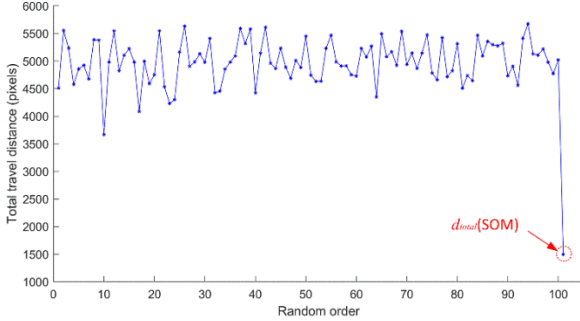


Fig. 7. Total travel distance cost comparison result.

2) *Test of mission points being assigned with different priorities constrained by battery limit:* four different levels of priorities are defined and listed in TABLE I with level 4 representing the highest priority and level 1 representing the lowest. Priority for each mission point is randomly assigned and visualised shown in Fig. 8(a). The simulation is first run when the battery limit  $\beta = 0.9$ . Fig. 8 shows the neurons updating process at different epoch, when  $iter = 1, 4, 66, 121, 129, 130, 132, 134, 135$  and  $140$ . As shown in Fig. 8(a), the algorithm starts with 30 neurons, which are the same number as the mission points and their locations are close to the USV start point. However, the number of the neurons is adaptively changed during the updating process with more neurons being added in the initial stages to provide an optimised result. For example, at iteration 4 shown in Fig. 8(b), the number of neurons is increased; while in Fig. 8(c), the number of neurons reduced as the updating process is approaching a stable state. Details of the neurons adding and deleting procedures are explained in [8]. Fig. 8(d) and (e) show that the number of neurons stays at 30 with the majority of the neurons staying close to their allocated mission points' positions with the exception of two mission points: the upper left one with priority level of 1, and the upper middle one with priority level of 3. However, in Fig. 8(f) the priority level of the mission point pointed by an arrow becomes red, which means that at iteration 130, the total travel distance cost  $C(T_{130})$  is beyond the battery limit, if the USV continues to follow the current neurons sequence ( $v_{130}$ ). Therefore, according to the mission selection criterion in Section III, the mission point with the lowest priority level (1 at iteration 130) and the longest lattice distance from the mission start point (5 at iteration 130) is first discarded. As the updating process progresses, it can be seen from Fig. 8(g), (h) and (i), more

mission points are discarded as displayed in red numbers. Because of the reduction of the mission points, the neurons number is also decreased accordingly with the final result shown in Fig. 8(j).

TABLE I. RISK WEIGHTINGS IN DIFFERENT SCENARIOS

Priority level	Priority range
1, low priority	$0 \leq P(g_i) \leq 5$
2, middle priority	$0 \leq P(g_i) \leq 25$
3, high priority	$0 \leq P(g_i) \leq 250$
4, very high priority	$0 \leq P(g_i) \leq 2500$

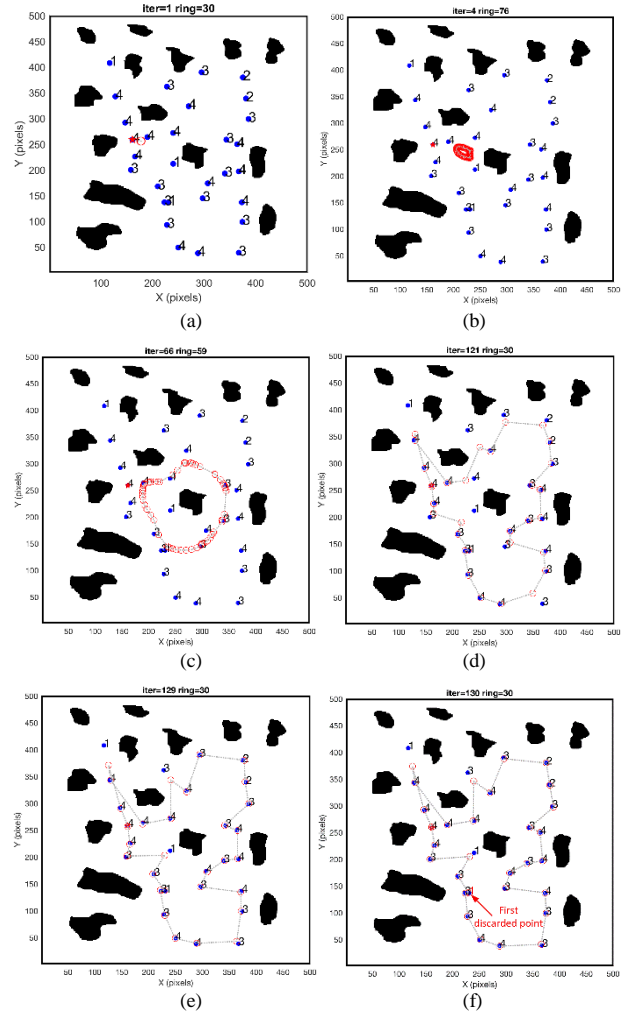


Fig. 8. Neurons updating process when the battery limit  $\beta = 0.9$ . Blue dots are the mission points, and the start point is marked as shaded red star. The priority level is marked beside each mission point in black number. If the mission point is discarded, the priority level will change to red. Neurons are marked as red circle, and 'ring' means the number of neurons

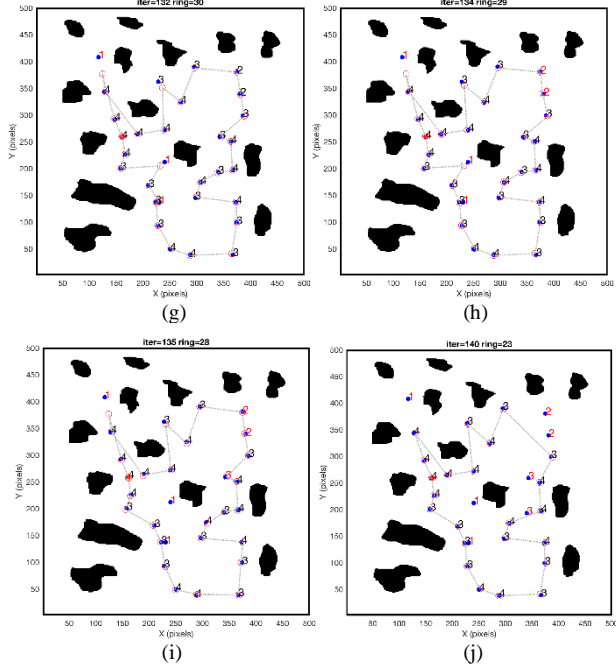


Fig. 8. (Continued)

The modified SOM algorithm is then applied under the same battery limit, namely  $\beta = 0.9$ , but with different priority levels assigned to mission points. Different settings of the priority levels as well as their corresponding results are shown in Fig. 9. It can be seen that under different priority settings, the algorithm can always generate a sequence of  $v$  under the battery constraint, which proves the robustness of the algorithm.

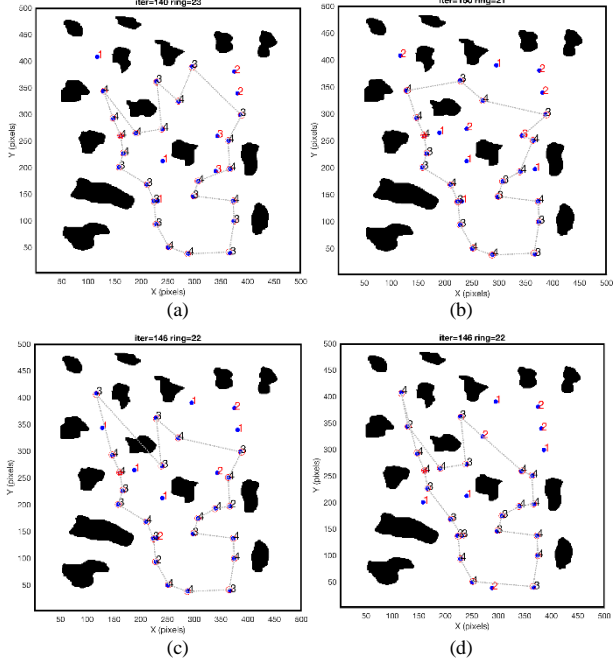


Fig. 9. Modified SOM simulation results under battery limit  $\beta = 0.9$ . (a), (b), (c) and (d) are the same environment, but with different settings of priority levels.

3) *Test of mission points being assigned with same priorities under different battery limits:* To further validate the robustness of the modified SOM algorithm, different battery limits are tested from  $\beta = 1, 0.95, 0.85, 0.75, 0.65$  to  $0.6$ . When  $\beta = 1$ , the equipped battery storage is able to support the full operation of the USV, therefore all mission points can be successfully visited as shown in Fig. 10(a). Also, when  $\beta$  is higher than  $0.6$ , the algorithm can generate the sequence successfully by appropriately selecting the mission points to visit. However, when  $\beta = 0.6$ , the algorithm shows a warning of ‘battery is too low’ as shown in Fig. 10(f) meaning  $\beta = 0.6$  is the minimum battery allowance. Therefore, even though all the mission points with priority level 1-3 have been neglected, the  $C(T_{\beta=0.6})$  is still high and the USV could not complete the mission within the present energy storage.

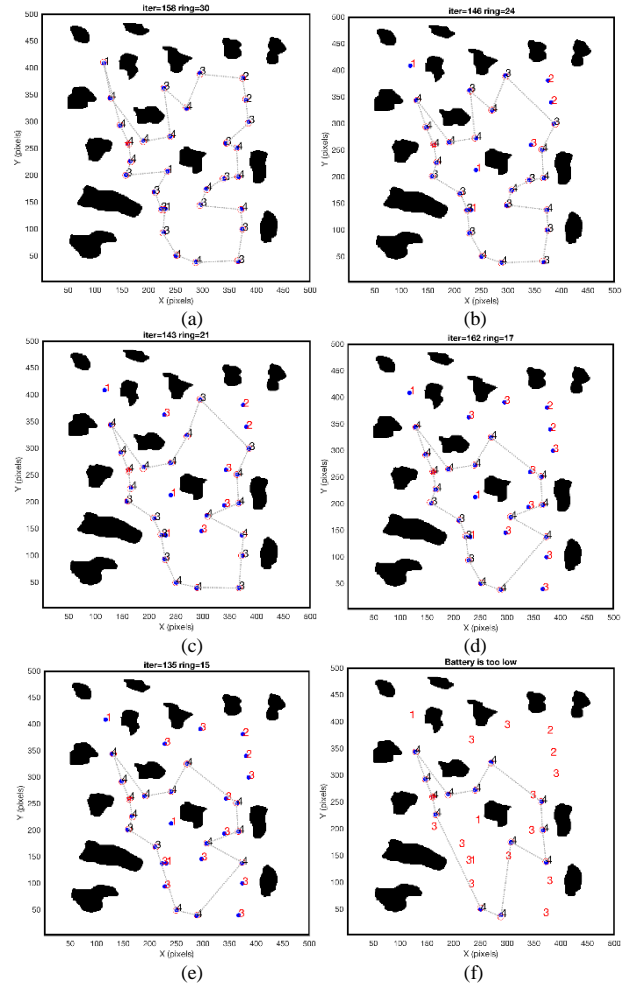


Fig. 10. Modified SOM simulation results with same priority level settings, but different battery limit, when (a)  $\beta = 1$ , (b)  $\beta = 0.95$ , (c)  $\beta = 0.85$ , (d)  $\beta = 0.75$  (e)  $\beta = 0.65$  and (f)  $\beta = 0.6$ .

## V. CONCLUSIONS

A multi-task allocation method based on the SOM algorithm has been developed for the energy-saving purpose in the application of water quality monitoring and pollution detection

using an USV. In the proposed method, the mission points, where the USV need to visit have been selected based on the predefined measurement priority as well as the distance penalties that are related to the energy constraint of the USV. During the updating process, a constraint factor has been considered to limit the influence area of the obstacles.

The simulations results show that under different priority settings and battery limit, the modified SOM method can robustly generate the sequence of mission points for the USV to follow. Also, when the total cost governing criteria cannot be satisfied, the algorithm is able to provide an alert for the USV to recharge the battery. It should be noted that only a conceptual design has been proposed in this paper to address the problem of energy-efficient autonomous navigation, further improvements are still needed: by considering the alternative USV velocities against consumption; and by taking into account of environment factors as well as combining the path planning and controllers to achieve a fully autonomous guidance, navigation and control (GNC) for USV.

#### ACKNOWLEDGMENT

This work is supported under grant No. 009-2016 from the National Fund for Scientific and Technological Development, FONDECYT from CIENCIACTIVA of CONCYTEC and by the Embassy of the United Kingdom in Peru. The authors also acknowledge the support from the Pontifica Universidad Catolicadel Peru, Department of Engineering, Section of Mechatronics, and University College London, Department of Mechanical Engineering.

#### REFERENCES

[1] W. Naeem, R. Sutton, and J. Chudley, "Modelling and control of an unmanned surface vehicle for environmental monitoring," in *UKACC International Control Conference*, 2006, pp. 1-6.

[2] W. B. Klinger, I. R. Bertaska, K. D. v. Ellenrieder, and M. R. Dhanak, "Control of an Unmanned Surface Vehicle With Uncertain Displacement and Drag," *IEEE Journal of Oceanic Engineering*, vol. 42, no. 2, pp. 458-476, 2017.

[3] J. Balbuena, D. Quiroz, R. Song, R. Bucknall, and F. Cuellar, "Design and implementation of an USV for large bodies of fresh waters at the highlands of Peru," in *OCEANS 2017; Anchorage*, 2017, pp. 1-8.

[4] S. Campbell, M. Abu-Tair, and W. Naeem, "An automatic COLREGs-compliant obstacle avoidance system for an unmanned surface vehicle," *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, vol. 228, no. 2, pp. 108-121, 2014.

[5] J. Leng, J. Liu, H. Xu, and Ieee, "Online Path Planning Based on MILP for Unmanned Surface Vehicles," *2013 Oceans - San Diego*, 2013 2013.

[6] X. Yan, P. Yi, D. Zhu, and L. Fu, "Research Of Unmanned Surface Vessel ( USV) Path- Planning Algorithm Based on ArcGIS," in *Second International Conference on Transportation Information and Safety*, ed. Reston, VA: Reston, VA: American Society of Civil Engineers, 2013, pp. 2125-2134.

[7] B. Angéniol, G. de La Croix Vaubois, and J.-Y. Le Texier, "Self-organizing feature maps and the travelling salesman problem," *Neural Networks*, vol. 1, no. 4, pp. 289-293, 1988/01/01/ 1988.

[8] Y. Liu and R. Bucknall, "Efficient multi-task allocation and path planning for unmanned surface vehicle in support of ocean operations," *Neurocomputing*, vol. 275, pp. 1550-1566, 2018/01/31/ 2018.

[9] J. Faigl and G. A. Hollinger, "Self-Organizing Map for the Prize-Collecting Traveling Salesman Problem," in *Advances in Self-Organizing Maps and Learning Vector Quantization*, Cham, 2014, pp. 281-291: Springer International Publishing.

[10] R. Song, Y. Liu, and R. Bucknall, "A multi-layered fast marching method for unmanned surface vehicle path planning in a time-variant maritime environment," *Ocean Engineering*, vol. 129, pp. 301-317, 2017/01/01/ 2017.

[11] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464-1480, 1990.

[12] S. M. Guthikonda, "Kohonen self-organizing maps," 2005.

[13] S. Garrido, M. Malfaz, and D. Blanco, "Application of the fast marching method for outdoor motion planning in robotics," *Robotics and Autonomous Systems*, vol. 61, no. 2, pp. 106-114, 2013.

[14] J. Faigl and G. A. Hollinger, "Unifying multi-goal path planning for autonomous data collection," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, 2014, pp. 2937-2942: IEEE.