# Path planning algorithm for unmanned surface vehicle formations in a practical maritime environment

Yuanchang Liu, Richard Bucknall*

*Department of Mechanical Engineering, University College London, Torrington Place, London WC1E 7JE, UK*

**Abstract**

Unmanned surface vehicles (USVs) have been deployed over the past decade. Current USV platforms are generally of small size with low payload capacity and short endurance times. To improve effectiveness there is a trend to deploy multiple USVs as a formation fleet. This paper presents a novel computer based algorithm that solves the problem of USV formation path planning. The algorithm is based upon the fast marching (FM) method and has been specifically designed for operation in dynamic environments using the novel Constrained FM method. The Constrained FM method is able to model the dynamic behaviour of moving ships with efficient computation time. The algorithm has been evaluated using a range of tests applied to a simulated area and has been proved to work effectively in a complex navigation environment.

*Keywords:* `USV formation, Path planning, Fast marching method`

## 1. Introduction

In recent years, with the benefits of reducing human casualties as well as increasing mission efficiencies, there have been increasing deployments of USVs in both military and civilian applications. However, current available USV platforms

*Corresponding author. Tel: +44 (0)20 7679 3777;
*Email address:* `r.bucknall@ucl.ac.uk` (Richard Bucknall)

have low payload capacity and short endurance times. In order to overcome these shortcomings; the current and future trend of USV operations is to deploy multiple vehicles as a formation fleet to allow cooperative operations. The benefits of using USVs formation operations include wide mission area, improved system robustness and increased fault-tolerant resilience.

Fig. 1 describes a hierarchical structure of a USV formation system. The structure consists of three layers, i.e. Task management layer, Path planning layer and Task execution layer. The Task Management Layer allocates the mission to individual USVs based on a general mission requirement. A mission can be generally defined as a set of way-points including mission start point and end point. According to the mission requirements, the second layer, i.e. the Path Planning Layer, plans feasible trajectories for a USV formation. It should be noted that cooperative behaviour for formation path planning is vital. Each vehicle should establish good communication to ensure formation behaviour. In addition, path re-planning needs to be considered if the formation is travelling in a dynamic environment. Generated paths will then be passed down to the Task Execution Layer, to calculate specific control for each vehicle. In order to improve the robustness of system as well as to minimise system error, real-time velocity and position information is fed back to the Path Planning Layer to modify the path. Also, planned trajectory information is sent back to the Task Management Layer in order to facilitate mission rearrangement. The whole structure is acting as a closed loop system to ensure safety of a USV formation.

As observed from the USV hierarchical structure, the Path Planning Layer plays an important role as it connects both the Task Management Layer and the Task Execution Layer and navigates the formation. Path planning is a complicated task and can be viewed as a multi-optimisation problem. The planned trajectory should be optimised in terms of several aspects such as total distance, navigation time and energy consumption. Also, collision avoidance is important for trajectory. The formation should not collide with any static obstacles (islands, buoys) and other moving vessels. To the best of the authors' knowledge, although several work such as Borrelli et al. (2004), Barfoot and Clark (2004) and Cao et al. (2003) studied formation path planning for unmanned aerial vehicle (UAV), unmanned ground vehicle (UGV) and mobile robots, there is currently no work specifically focused on developing a robust formation path planning algorithm for USVs. This is possibly due to the reasons of high uncertainty and complexity of obstacles in an ocean environment.

Therefore, this paper aims to propose a practical path planning algorithm for USV formation in real navigation environments. It is the first work specifically solving the USV formation problem with algorithm practicability as the main feature of this research. A number of previous works have developed path planning algorithms for USVs; however, nearly all of them (Tam and Bucknall (2013), Naeem et al. (2012), Thakur et al. (2012)), with the notable exception of Kim et al. (2014), simulated algorithms in simple self-constructed environments rather than real ocean environments. The algorithm designed in this paper is able to extract information from a real navigation map to construct a synthetic grid map, where both static and dynamic obstacles are well represented. By using such a map, a collision free path may be generated which can be directly used as a guidance trajectory for practical navigation.

The rest of the paper is organised as follows. Section 2 reviews related work in terms of formation path planning. Sections 3 and 4 describe fundamentals of the method used in this paper as well as the algorithm which models static and dynamic obstacles. Section 5 introduces the USV formation path planning algorithm. Proposed algorithm and methods are verified by simulations in section 6. Section 7 concludes the paper and discusses the future work.

## 2. Literature review

Due to limited resources studying USV formation path planning, and also in order to give a more thorough review of the current research situation; literature from not only USV, but also UAV, UGV and unmanned underwater vehicle

(UUV) have been reviewed in this section. For simplicity, we have named all kinds of autonomous vehicle as ’unmanned vehicle’ in following section.

### *2.1. Formation control structure*

For unmanned vehicle formations, maintenance of the formation shape is of great importance. To maintain the shape, several control structures including leader-follower, virtual structure and behaviour based approaches have been proposed by a number of researchers. In leader-follower approach (Liu et al. (2007),Cui et al. (2010),Morbidi et al. (2011),Peng et al. (2013)), one vehicle is assigned as leader vehicle, which has access to overall navigation information and tracks the predefined path. All the other vehicles in the formation are followers aiming to maintain the desired geometric configuration. In terms of virtual structure approach (Ren (2008),Ghommam et al. (2010),Cong et al. (2011),Mehrjerdi et al. (2011)), the formation is treated as a rigid body and maintained by making each vehicle in the formation follow a reference point in the rigid body. Both of these approaches adopt a centralised control topology, where all the important control decisions are made within at the centre of the system. In comparison, behaviour based approach allows the utilisation of decentralised control. It breaks down the formation tasks into several sub tasks according to different behaviours. In the work of Balch and Arkin (1998), formation maintenance is integrated with other missions such as goal keeping and collision avoidance and the control of each vehicle is the result of a weighted function of these missions.

### *2.2. Multiple vehicles formation path planning*

The nature of unmanned vehicles formation path planning is an optimisation process of multiple objectives, which is more complicated than single vehicle path planning. Fig. 2 compares optimisation objectives of these two kinds of path planning problems. It is noted that besides single vehicle path planning optimisation criteria, more attention is paid to address formation behaviours in formation path planning. The planned trajectories of the formation should, to the most extent, maintain the predefined shape. Also, a certain degree of flexibility such as shape variation or change is preferred to accommodate the navigation environment, which is beneficial to the formation's safety.

To achieve formation path planning, a number of different approaches have been proposed, which could be categorised based on two disciplines:

- Deterministic approach
- Heuristic approach

Deterministic approach is achieved by following a set of defined steps to search for the solution whereas heuristic approach only searches inside a subspace of the search space without following rigorous procedures (Tam et al. (2009)) .

Heuristic approach is designed to provide solutions when classic search methods fail to find exact solutions. Its speciality is in dealing with multi-optimisation problems with fast computational speed. Therefore, a number of heuristic search based algorithms such as genetic algorithm (Zheng et al. (2004), Yang et al. (2006), Kala (2012), Qu et al. (2013)), particle swarm optimisation (Duan et al. (2008), Bai et al. (2009)) and ant colony asexual reproduction optimisation (Asl et al. (2014)) have been used for formation path planning. The algorithms normally use decentralised control topology, where each vehicle of the formation has its own path planning process and cooperates with others through a co-evolution process. However, heuristic path planning algorithm is not able to rigorously maintain the formation shape. Even though trajectories can be coordinated by introducing certain fitness functions, the uncertainty and randomness of a heuristic search makes the path hard to follow a predefined shape and heuristic path planning suffers problems of incompleteness and inaccuracy of search results.

In contrast, deterministic path planning approach has the features of search completeness and consistency. Among them, artificial potential field (APF) is becoming a key method due to its easy implementation and good collision avoidance capability. The theory behind it is to construct two different potential fields, i.e. attractive and repulsive fields around target point and obstacles respectively. An attractive field is constructed across the space with magnitude proportional to the distance to the target point; whereas, a repulsive field is built within a certain area called -"influence area"- around obstacles and the magnitude is inversely proportional to the distance to the obstacle. Based on the potential field, the vehicle can then be guided by following total field gradient. Detailed explanation of this can be referred to Khatib (1986) and Ge and Cui (2002).

In terms of implementation of APF in formation path planning, besides potential fields around target point and obstacles, new fields need to be constructed to keep formation distances as well as avoid collision between vehicles within the formation. Wang et al. (2008) first constructed such potential fields by referring to the concepts of electric field. Each vehicle was treated as point in the electric field with varying electrical polarity. If the distance between vehicles was larger than the expected value, opposite charges were used to attract them to move towards each other; otherwise, like polarities were used to prevent them from colliding when two vehicles were moving within close proximity.

Paul et al. (2008) also applied APF method to solve the problem of UAV formation path planning. Attractive fields between leader-follower as well as follower-follower were built to keep formation shape, and repulsive fields were used to prevent internal collision as well as collision with obstacles. To increase control accuracy as well as to better address the formation shape maintenance problem, attractive potential field was a function of the error value between desired distance and actual leader-follower or follower-follower distance such that any deflection from the desired position can be quickly modified and corrected.

Yang et al. (2011) published work on motion planning for UUV formation in an environment with obstacles based on APF. The algorithm concentrated on overall mission requirements instead of development of individual vehicle's control law and treated UUV formation as a multibody system with each vehicle modelled as a point mass with full actuation. Potential fields for formation path planning were constructed for particular mission requirement, ocean environment and formation geometry.

It should be noted that APF is prone to a local minima problem, which makes the algorithm fail to 'jump out' of local minimum point and reach the target point. Although methods proposed in Sheng et al. (2010) and Xue et al. (2011) solved it by introducing virtual target point the impact was a sacrifice in computation time consequently potential field with single global minimum point is preferred. Garrido et al. (2011) used the fast marching (FM) method to construct potential field with the target point as single minimum point for robot formation path planning. As a method for solving the viscosity solution of the eikonal function, the FM can successfully simulate the propagation of electromagnetic waves. The potential field in which electromagnetic wave transmits has good properties such as absence of local minima. Besides, the gradient of such a potential field is smoother than conventional one, which is more suitable for a vehicle to track. Gomez et al. (2013) further improved the FM method to fast marching square (FMS) method and increased the safety of planned trajectories.

In this paper, the authors improve upon the work of Gomez et al. (2013) and developing its application specifically for USV formation with emphasis on path planning in a dynamic environment. A new constrained FM method is proposed to model the dynamic behaviour of moving ships for collision avoidance. In addition, path replanning capability is incorporated to improve the completeness of the algorithm.

## 3. Eikonal equation and fast marching method

### *3.1. Fast marching method*

The fast marching method was first proposed by J.Sethian in 1996 to track the evolution of interfaces by numerically solving the viscosity solution of eikonal equation :

$$|\nabla(T(\boldsymbol{x}))|W(\boldsymbol{x}) = 1 \tag{1}$$

where $\boldsymbol{x}$ represents the point in metric space, i.e. $\boldsymbol{x} = (\mathrm{x}, \mathrm{y})$ in 2D space and $\boldsymbol{x} = (\mathrm{x}, \mathrm{y}, \mathrm{z})$ in 3D space. T($\boldsymbol{x}$) is time matrix representing the arrival time of interface front at point $\boldsymbol{x}$, and W($\boldsymbol{x}$) is speed matrix and describes local propagating speed at point $\boldsymbol{x}$. By using an upwind finite difference approximation scheme, the solving process of FM is similar to Dijkstra's method but in a continuous way.

When applying FM method to the path planning problem, a more intuitive way to interpret it is from the potential field perspective. In Fig. 3, two round obstacles are located near the centre of the map; while the start and end points are at northwest and southeast corners respectively. The map is represented by binary grid map, where each grid in collision free space has value 1 and grids in obstacle areas have value 0.

FM is then applied on such a grid to simulate an interface propagation process. The interface is used to help build up a potential field, whose potential value on each grid point is the local interface arrival time. The interface begins to proceed from the start point on the grid map by taking local grid values to determine propagation speed. The evolution process of interface is shown in Fig. 4, where the brighter the colour is, the longer the arrival time. When the interface reaches the target point, the potential field (Fig. 5a) is created. The meaning of the colour in the figure is the same as Fig. 4's. In the field, the potential value at each point represents local arrival time of the interface, which subsequently indicates local distance to the start point if a constant speed matrix is used. Since the interface begins propagating from the start point, the potential of the start point is therefore the lowest and is equal to zero. Potential values at other points increase as the interface advances and reach highest value at the end point. Because the interface is not allowed to transmit inside an obstacle area, obstacles' potentials are infinite. Compared with the potential field generated by APF, the potential field of FM has features of global minimum, which avoids local minima problems and increases the completeness of the algorithm. Based on the potential field obtained, the gradient descent method is then applied to find the shortest collision free path by following the gradient of the potential field. Such algorithm is shown in Algorithm 1. The algorithm first determines the highest potential value (*max*), and uses function *RescaleField* to rescale the potential field within the range 0 to *max*. It then computes the gradient of the rescaled potential field and finds an optimal path connecting the end point and the point with the lowest potential. The start point will be eventually added into the path if the lowest point is not the start point. Path generated by using the Algorithm 1 is shown as red line in Fig. 5b. It should be noted that the shortest path is defined in geodesic terms, which means that path has shortest Euclidean distance if the environment has constant W($\boldsymbol{x}$) and is a weighted Riemannian manifold with varying W($\boldsymbol{x}$) (Garrido et al. (2011)).

**Algorithm 1** *Path_Gradient_Descent* Algorithm

**Input:** potential field (T), start point ($p_{start}$), end point($p_{end}$), stepSize

1: $max \leftarrow T.max$
2: $T \leftarrow RescaleField(T, 0, max)$
3: $grad \leftarrow ComputeGrad(T)$
4: $path \leftarrow PathCalculator(grad, p_{end}, stepSize)$
5: **if** $path.endpoint! = p_{start}$ **then**
6: $\quad path.Add(path, p_{start})$
7: **end if**
8: **return** *path*

## 4. Planning space representation

In path planning problems, safety always holds priority no matter what application. To generate a safe trajectory, it is necessary to properly represent the environment in which the path planning algorithm is implemented. It is especially important for USV navigation environments, which include a great deal of maritime uncertainties. Sufficient safe distance should always be maintained between USV and obstacles (both static and dynamic). In this section, the FM based map representation method for both static environment and moving obstacles is described.

### *4.1. Static obstacles representation*

One of the problems associated with path planning by directly using the FM method is the generated path is too close to obstacles. Such a drawback is especially impractical for USVs, because near distance areas around obstacles (mainly islands and coastlines) are usually shallow water, which is not suitable for marine vehicles to navigate. Hence, it is important to keep the planned path a certain distance away from obstacles.

To tackle this problem, FMS method proposed by Gomez et al. (2013) for indoor mobile robots is used in this paper. The basic concept behind FMS is to apply the conventional FM algorithm twice but with different purposes:

- ***step*1** : FM is applied on original binary environment map ($\boldsymbol{M_o}$) to create safety map ($\boldsymbol{M_s}$). Instead of calculating a single interface's propagation by using a USV's mission start point; in this process, multiple interfaces are emitted from all points that represent obstacles (points with value 0 in the binary map) and continue to advance until it reaches the map boundary. Generated map ($\boldsymbol{M_s}$) is shown in Fig. 6b, where each point is assigned a value, ranging from 0 to 1, representing the shortest local arrival time. Since constant propagating speed is used, the local shortest arrival time also determines the shortest distance to obstacles. The further the distance to an obstacle is, the higher the value will be. Such values can be viewed as indices to indicate the safety of local points. Low values represent current locations may be too close to obstacles and consequently may not be safe to proceed; hence USVs should be encouraged to keep travelling in the areas with high index value.

- ***step2*** : FM is used again over the safety map ($\boldsymbol{M_s}$) to generate the potential field. USV's mission start point is now the algorithm's start point. Since $\boldsymbol{M_s}$ is used as a speed matrix in this step, which gives non-constant speed over the space, the interface now tends to remain in places with high propagating speed. The generated potential field should follow the trace of the interface, which is shown in Fig. 7b. Note the field's shape is different to that of Fig. 6b, which was generated by using a constant propagating speed matrix. Potential of nearby obstacles is always higher than at other places', which act as a protecting layer to prevent the path passing too close to obstacles. This can be proved by result paths shown as red lines in Fig. 7a and Fig. 7b.

### *4.2. Dynamic obstacles representation*

To prevent collision with dynamic obstacles or moving ships, most studies in path planning research have adopted the concept of a 'safety area' ('ship domain' in marine vessels collision avoidance) to model the area from which all other vehicles are prohibited. The shape of such area is usually circular and the centre of the area is located on the obstacle's instantaneous position. However, in USV path planning, circular shape area is not always practical, especially when a ship is travelling at high speed, which holds more risks at fore areas than aft and sides. It is more realistic to assign the shape of safety area of a ship according to its velocity.

In this paper, a new method called 'Constrained FM method' has been developed to model the ship domain of a dynamic vessel. In contrast to conventional FM, the Constrained FM method propagates the interface within a certain space rather than over the whole configuration space. Since the points explored by the algorithm have been dramatically reduced, the computation time of the Constrained FM is relatively low. Such a feature increases the capability of the algorithm to deal with dynamic collision avoidance, which requires fast computation speed to handle the position change of a moving obstacle. Fig. 8

compares these two algorithms by propagating interfaces from four start points. Configuration space is constructed as a 400*400 pixels area. It can be observed from Fig. 8b that four propagations have been restrained in four small circular areas. In terms of computation time, conventional FM spends 0.101 s to explore the space whereas it only takes 0.053 s for the Constrained FM, a near 50 % improvement.

To model a dynamic vessel, the Constrained FM method is implemented twice in the algorithm, the flow chart of which is show in Fig 9. It first reads in velocity ($V_i$) of the $i^{th}$ ship, where i is the index of the vessel. Based on $V_i$, the algorithm starts to build the ship domain by adopting the shape proposed in Tam and Bucknall (2010). Ship domain alters its shape according to specific velocity; a more circular shape is constructed if vessel is travelling with low speed and half-elliptical shape is used for a high speed vessel. The dimension of the ship domain is computed by following two equations to calculate aft and fore sections respectively. For aft section, it is defined as:

$$SA_{Aft} = \begin{cases} r_{aft} & \text{if } r_{aft} \geq r_{min}, \\ r_{min} & \text{otherwise.} \end{cases} \tag{2}$$

where $r_{min}$ is the minimum distance must been retained between two vessels. And $r_{aft}$ is computed by:

$$r_{aft} = \begin{cases} velocity \times time & \text{if } velocity \times time < DisLimit, \\ 2 \times DisLimit - (velocity \times time) & \text{otherwise.} \end{cases} \tag{3}$$

where *time* is the scaling factor and defined as 1.0 min in this paper which is appropriate to establish the area a vessel could potentially cover in such time period. However, it should be noted that such a parameter could be customised according to specific needs in a practical navigation situation. $DisLimit$ is a predefined scalar variable to limit the maximum allowable area on the side and stern sections.

For fore section, the equation is defined as:

$$SA_{fore} = \begin{cases} velocity \times time & \text{if } velocity \times time < DisLimit, \\ r_{min} & \text{otherwise.} \end{cases} \tag{4}$$

After the determination of dimension of ship domain ($C_{SD}$), the Constrained FM method will be used to propagate the interface within $C_{SD}$ with the source point located at the instantaneous position of the vessel to be modelled (See Fig. 10a and Fig. 10b). Since other ships are ruled out of entering into a ship domain, which makes the domain act like an obstacle; potential values obtained by running FM method in ship domain are therefore reset to be zero as $T(C_{SD}) = 0$.

Then, a new area called 'collision avoidance area' ($CA$) is constructed so that any path violating the ship domain will be re-calculated to produce an updated trajectory. $CA$'s dimension is controlled by scalar variable $CAScalar$ as:

$$S_{CA} = S_{SD} \times CAScalar \tag{5}$$

where $S_{CA}$ and $S_{SD}$ are the area dimension for collision avoidance area and ship domain area respectively. Equation 5 shows that CA has the same shape as ship domain but enlarged. Constrained FM method is applied again within $CA$ by using all points in the ship domain as start points (See Fig. 10c and Fig. 10d). Generated $CA$ will be further scaled to make potential values inside range from 0 to 1 so that it has uniform representation as the static potential map generated by FMS method.

Fig. 10e illustrates ship domains generated under different speeds. Low speed ships are given a circular shape ship domain so that equal collision risks are distributed around ship. When the ship is travelling at high speed, fore section holds more risks than other sides. Therefore, more emphasis is placed on this area and the area is increased in proportion to speed.

Another kind of collision avoidance of dynamic obstacles, especially for formation path planning, is to prevent internal USVs in the formation from colliding. When two USVs are moving too close to each other from any direction, a repulsive force is needed to maintain safety. Therefore, constrained FM method is still used here but with a circular shape to model formation USVs.

## 5. USV formation path planning

The flow chart for USV path planning algorithm is show in Fig. 11. The algorithm adopts leader-follower formation control structure along with on-line path planning scheme to largely maintain formation shape. Leader USV's target point is mission end point and fixed; whereas, followers' target points are re-planned during each time step according to formation shape requirement. Based on these target points, FM method is iteratively applied for each USV to search for collision free path in real time.

Specific algorithm procedure is discussed here. During each time cycle $\boldsymbol{t}$, leader USV's path is searched first.The algorithm generates a static environment map by using FMS method introduced previously. Since the static environment does not change during the path planning period, generated map is stored as $M_{static}$. Then, based on instantaneous positions and velocities of moving obstacles as well as other USVs in formation, dynamic obstacles representation algorithm is used to model the behaviours of vessels. Synthetic map combining static and dynamic obstacles is finally compounded such that FM method can be used to calculate path for leader vehicle.

Once the leader's path is determined, the algorithm starts to iterate to compute paths for followers. Similar procedures are followed; however, since follower's target points are re-planned during each time step, it is possible that the target point is located within the obstacle (see Fig. 12a) such that the algorithm fails to find the path. Hence, a sub target re-planning algorithm is used to 'remove' the target point to a new feasible place with minimum impact on overall performance. It is computed based on distance reduction scheme as well as dynamic characteristics of the USV and summarised as Algorithm 2:

**Algorithm 2** *Sub_Target_Re-planning* Algorithm

**Input:** sub target point ($p_{sub}$), USV's current point ($p_{usv}$), distance reduction scalar ($RdScalar$)

```
1: while p_sub = obstacle do
2:     p_sub ← (p_sub + p_usv) × RdScalar
3: end while
4: return p_sub
```

In the Algorithm 2, the parameter *RdScalar* varies based on the dynamics of USV, i.e. if the USV has high manoeuvrability, it is able to reduce the distance travelled by a large amount thereby setting *RdScalar* with a small value such as 0.1. Sub target re-planning procedure is shown in Fig. 12b. Based on sub target points, the algorithm computes the trajectory for follower vehicles until all of them have been updated, which is the end of time cycle $\boldsymbol{t}$. Then it will continue the path planning process until leader vehicle arrives at the final target point.

## 6. Simulations

To validate the algorithm, simulations have been carried out using two different tests in the dynamic environment with one moving obstacle and dynamic environment with multiple moving obstacles. We use practical simulation areas to further test the algorithm's capability dealing with real navigation requirement. The algorithm has been coded in Matlab and simulations are run on the computer with a Pentium i7 3.4 Ghz processor and 4Gb of RAM.

In the simulations, we assume that identical USVs are used in formation. Speed of leader USV is set as constant such that it is easier for other USVs to follow. Followers, however, can vary their speeds according to their positions in formation. For example, follower USV needs to remain at the same velocity as leader's when it is moving at desired formation position. If current position of follower deviates from the desired position, it is required for follower to speed up to catch up or slow down to wait for the leader.

### *6.1. Simulation in dynamic environment with one moving obstacle*

In the first test, simulation area is selected near Portsmouth harbour (Fig. 13a), which is a large natural water area and one of the busiest harbours in the UK. The dimension of the area is 2500 m×2500 m, which is transferred to a 500 pixels×500 pixels grid map (Fig. 13b). The start and end points for USV formation are marked as red and purple markers in Fig. 13a. To test the capability of the algorithm dealing with dynamic obstacle, a moving vessel with a constant speed of 6 knot and a constant course of 284°is added into the simulation area.

Simulation results recording the movement sequences of the formation are represented in Fig. 14. Each representative sequence is depicted in both a binary map and the corresponding potential map. In binary maps, the leader USV is drawn in red, and follower1 and follower2 USVs are in magenta and blue. The track of the target ship (TS) is represented as red circles. The binary map is generated based on leader USV's view with its instantaneous position drawn as black square marker.

Since the harbour has a narrow channel, the line formation shape is selected as the desired formation shape with a formation distance of 15 pixels (75 m). However, to validate the algorithm's capability of formation generation, a triangle formation shape is assigned as the initial shape shown in Fig. 14a. In Fig. 14b, safety potential map of the simulation area along with TS is shown. It is clear that both static obstacle area (in dark blue) and safe area (in red) have been identified. In addition, the TS has also been well represented with a circular ship domain and collision avoidance area. After time step 5, the formation forms the line shape and keeps such shape entering into the channel area (Fig. 14c - Fig. 14f). Fig. 14g - Fig. 14l illustrate how the formation avoids the TS. When the formation approaches close to the TS, port side turning is adopted by the leader, and two followers will follow this behaviour. In the corresponding safety potential maps (Fig. 14h, Fig. 14j and Fig. 14l), it can be observed that each USV can stay well outside the ship domain and inside the collision avoidance area of TS to generate a collision avoiding trajectory. After the collision risk is avoided, the formation moves towards target point and reaches it at time step 113.

Evaluations of the algorithm performance and USV formation behaviour are given in Fig. 15. Fig. 15a shows the overall trajectories for the formation, and all of them remain a safe distance away from static obstacles, which proves that the algorithm is able to generate acceptable safe paths in a complex environment. Furthermore, in Fig. 15b, distances between TS and each USV are recorded. It is noted that the closest distances for leader and two followers are approximately 21 pixels, 17 pixels and 25 pixels, which demonstrates that formation can effectively avoid moving obstacle. In terms of formation behaviour, distance errors between actual positions and desired positions for follower1 and follower2 are shown in Fig. 15c. It may be concluded that during initial time steps, large errors occur since two followers are not located at their desired positions. However, both of them can fast navigate to their formation positions by following generated trajectories, and once the formation is formed the formation shape can be well maintained as the error values remain relatively small.

### *6.2. Simulation in dynamic environment with multiple moving obstacles*

A more complex simulation is done in a dynamic environment with multiple moving vessels. Ocean area near Plymouth harbour shown in Fig. 16a is selected as the testing area. In Fig. 16b, planning space has been transformed into a square area with 500×500 pixels dimension representing 2.5×2.5 km area. Now, three virtual target ships are added into the environment travelling at 20 knot (TS1), 6 knot (TS2) and 12 knot (TS3) respectively.

The formation now starts with line shape and the desired formation shape is triangular with formation distance as 15 pixels (75m). Movement sequences of the formation are represented in Fig. 17, which includes both the original binary maps as well as the potential maps. In the potential maps, it is shown that the algorithm can well define the ship domain and collision avoidance areas of three target ships based on their velocities. TS1 has the highest velocity thereby forming an half-elliptical shape. In contrast, the other two ships are relatively slow, so more circular shapes are assigned. Between them, because TS3 has larger speed than TS2, generated area of TS3 has a longer radius than TS2's. In addition, to prevent internal collision, internal USV is viewed as a circle with radius representing safe distance in potential map.

To assess the algorithm, first of all, trajectories generated by the algorithm are shown in Fig. 18a. It is clear that each path maintains a good position to the others and does not collide with any static obstacles. Fig. 18b shows the distances between target ships and each USV for whole simulation time period. Smallest distance occurs at time step 61 with the value of 11 pixels (55 m) between TS2 and follower1, which means that the formation does not collide with any target ships. In terms of formation behaviour, Fig. 18c records the distance error values. Except the initial formation generation stages, the values remain close to zero for most of simulation time, which means that the formation shape is well maintained.

## 7. Conclusions and future work

This paper introduced and discussed a path planning algorithm for the USV formation navigation. Fahimi (2007) and Antonelli et al. (2006) have previously investigated the problem of USV formation, the emphasis of these works is on robust control (Level 3 in Fig. 1) instead of path planning (Level 2 in Fig. 1).

The algorithm we introduced in this paper is the first work specifically dealing with the USV formation path planning problem. The algorithm developed is based on the FM method, which has features of fast computation speed and low computation complexity. To particularly address the dynamic problem in path planning, a Constrained FM method has been proposed and developed to construct two areas, i.e. ship domain area and collision avoidance area, to ensure the planned trajectory to not violate any forbidden area. In addition, the output from the algorithm shows that collision free paths can be generated for formations for complex, practical and for both static and dynamic environments. More importantly, since all of the simulations are taken in real navigation areas, it is worth mentioning that the algorithm is practical and can potentially be developed to advance navigation in manned ships.

For future work, the algorithms proposed will be improved in several ways. First, the practicability of planned paths can be further increased. COLREGS, which is the international martime collision avoidance regulation, is largely obeyed by most navigators when taking collision avoidance manoeuvres and should also be integrated into current algorithms. Second, the trajectory could be optimised in terms of aspects such as energy consumption, and environment influences such as current and wind. Thirdly, a mission planning module can be included into the algorithm. The module is a self-decision making system, which can accordingly assign different missions based on specific requests. This will enormously improve the autonomy of USVs, which is the ultimate goal of this research.

## 8. Acknowledgements

This work is supported by the ACCeSS group.The Atlantic Centre for the innovative design and Control of Small Ships (ACCeSS) is an ONR-NNRNE programme with Grant no. N0014-10-1-0652, the group consists of universities and industry partners conducting small ships related researches. The first author would like to thank the China Scholarship Council (CSC) for supporting his studies at the University College London, UK. The authors are also indebted to Konrad Yearwood for his valuable critique of this paper and Javier V. Gomez for his valuable discussions and suggestions for the algorithm.

## References

Antonelli, G., Arrichiello, F., Chiaverini, S., 2006. Experiments of formation control with collisions avoidance using the null-space-based behavioral control, in: Control and Automation, 2006. MED'06. 14th Mediterranean Conference on, IEEE. pp. 1–6.

Asl, A.N., Menhaj, M.B., Sajedin, A., 2014. Control of leader follower formation and path planning of mobile robots using asexual reproduction optimization (aro). Applied Soft Computing. 14, Part C, 563 –76.

Bai, C., Duan, H., Li, C., Zhang, Y., 2009. Dynamic multi-uavs formation reconfiguration based on hybrid diversity-pso and time optimal control, in: Intelligent Vehicles Symposium, IEEE., pp. 775–9.

Balch, T., Arkin, R.C., 1998. Behavior-based formation control for multirobot teams. IEEE Transactions on Robotics and Automation. 14, 926–39.

Barfoot, T., Clark, C., 2004. Motion planning for formations of mobile robots. Robotics and Autonomous Systems. 46, 65 – 78.

Borrelli, F., Keviczky, T., Balas, G., 2004. Collision-free uav formation flight using decentralized optimization and invariant sets, in: 43rd IEEE Conference on Decision and Control, CDC., pp. 1099–1104 Vol.1.

Cao, Z., Xie, L., Zhang, B., Wang, S., Tan, M., 2003. Formation constrained multi-robot system in unknown environments, in: Proceedings. ICRA '03. IEEE International Conference on Robotics and Automation., pp. 735–740 vol.1.

Cong, B.L., Liu, X.D., Chen, Z., 2011. Distributed attitude synchronization of formation flying via consensus-based virtual structure. Acta Astronautica. 68, 1973 –86.

Cui, R., Ge, S.S., How, B.V.E., Choo, Y.S., 2010. Leader follower formation control of underactuated autonomous underwater vehicles. Ocean Engineering. 37, 1491 –502.

Duan, H., Ma, G., Luo, D., 2008. Optimal formation reconfiguration control of multiple ucavs using improved particle swarm optimization. Journal of Bionic Engineering. 5, 340 –7.

Fahimi, F., 2007. Sliding-mode formation control for underactuated surface vessels. Robotics, IEEE Transactions on 23, 617–22.

Garrido, S., Moreno, L., Lima, P.U., 2011. Robot formation motion planning using fast marching. Robotics and Autonomous Systems. 59, 675 –83.

Ge, S., Cui, Y., 2002. Dynamic motion planning for mobile robots using potential field method. Autonomous Robots. 13, 207–22.

Ghommam, J., Mehrjerdi, H., Saad, M., Mnif, F., 2010. Formation path following control of unicycle-type mobile robots. Robotics and Autonomous Systems. 58, 727 –36.

Gomez, J.V., Lumbier, A., Garrido, S., Moreno, L., 2013. Planning robot formations with fast marching square including uncertainty conditions. Robotics and Autonomous Systems. 61, 137 –52.

Kala, R., 2012. Multi-robot path planning using co-evolutionary genetic programming. Expert Systems with Applications. 39, 3817 –31.

Khatib, O., 1986. Real-time obstacle avoidance for manipulators and mobile robots. The international journal of robotics research. 5, 90–8.

Kim, H., Kim, D., Shin, J.U., Kim, H., Myung, H., 2014. Angular rate-constrained path planning algorithm for unmanned surface vehicles. Ocean Engineering. 84, 37 – 44.

Liu, S.C., Tan, D.L., Liu, G.J., 2007. Robust leader-follower formation control of mobile robots based on a second order kinematics model. Acta Automatica Sinica. 33, 947 –55.

Mehrjerdi, H., Ghommam, J., Saad, M., 2011. Nonlinear coordination control for a group of mobile robots using a virtual structure. Mechatronics. 21, 1147 –55.

Morbidi, F., Bullo, F., Prattichizzo, D., 2011. Visibility maintenance via controlled invariance for leader follower vehicle formations. Automatica. 47, 1060 –7.

Naeem, W., Irwin, G.W., Yang, A., 2012. Colregs-based collision avoidance strategies for unmanned surface vehicles. Mechatronics. 22, 669 –78.

Paul, T., Krogstad, T.R., Gravdahl, J.T., 2008. Modelling of uav formation flight using 3d potential field. Simulation Modelling Practice and Theory. 16, 1453 –62.

Peng, Z., Wen, G., Rahmani, A., Yu, Y., 2013. Leader follower formation control of nonholonomic mobile robots based on a bioinspired neurodynamic based approach. Robotics and Autonomous Systems. 61, 988 –96.

Qu, H., Xing, K., Alexander, T., 2013. An improved genetic algorithm with co-evolutionary strategy for global path planning of multiple mobile robots. Neurocomputing. 120, 509 –17.

Ren, W., 2008. Decentralization of virtual structures in formation control of multiple vehicle systems via consensus strategies. European Journal of Control. 14, 93 – 103.

Sheng, J., He, G., Guo, W., Li, J., 2010. An improved artificial potential field algorithm for virtual human path planning, in: Entertainment for Education. Digital Techniques and Systems. Springer, pp. 592–601.

Tam, C., Bucknall, R., 2010. Collision risk assessment for ships. Journal of Marine Science and Technology. 15, 257–70.

Tam, C., Bucknall, R., 2013. Cooperative path planning algorithm for marine surface vessels. Ocean Engineering. 57, 25 – 33.

Tam, C., Bucknall, R., Greig, A., 2009. Review of collision avoidance and path planning methods for ships in close range encounters. Journal of Navigation. 62, 455–76.

Thakur, A., Svec, P., Gupta, S.K., 2012. Gpu based generation of state transition models using simulations for unmanned surface vehicle trajectory planning. Robotics and Autonomous Systems. 60, 1457 –71.

Wang, J., Wu, X., Xu, Z., 2008. Potential-based obstacle avoidance in formation control. Journal of Control Theory and Applications. 6, 311–6.

Xue, Y., Clelland, D., Lee, B., Han, D., 2011. Automatic simulation of ship navigation. Ocean Engineering. 38, 2290 –305.

Yang, D., Jinyin Chenand Matsumoto, N., Yamane, Y., 2006. Multi-robot path planning based on cooperative co-evolution and adaptive cga, in: IAT '06. IEEE/WIC/ACM International Conference on Intelligent Agent Technology., pp. 547–50.

Yang, Y., Wang, S., Wu, Z., Wang, Y., 2011. Motion planning for multi-hug formation in an environment with obstacles. Ocean Engineering. 38, 2262 –9.

Zheng, C., Ding, M., Zhou, C., Li, L., 2004. Coevolving and cooperating path planner for multiple unmanned air vehicles. Engineering Applications of Artificial Intelligence. 17, 887 –96.

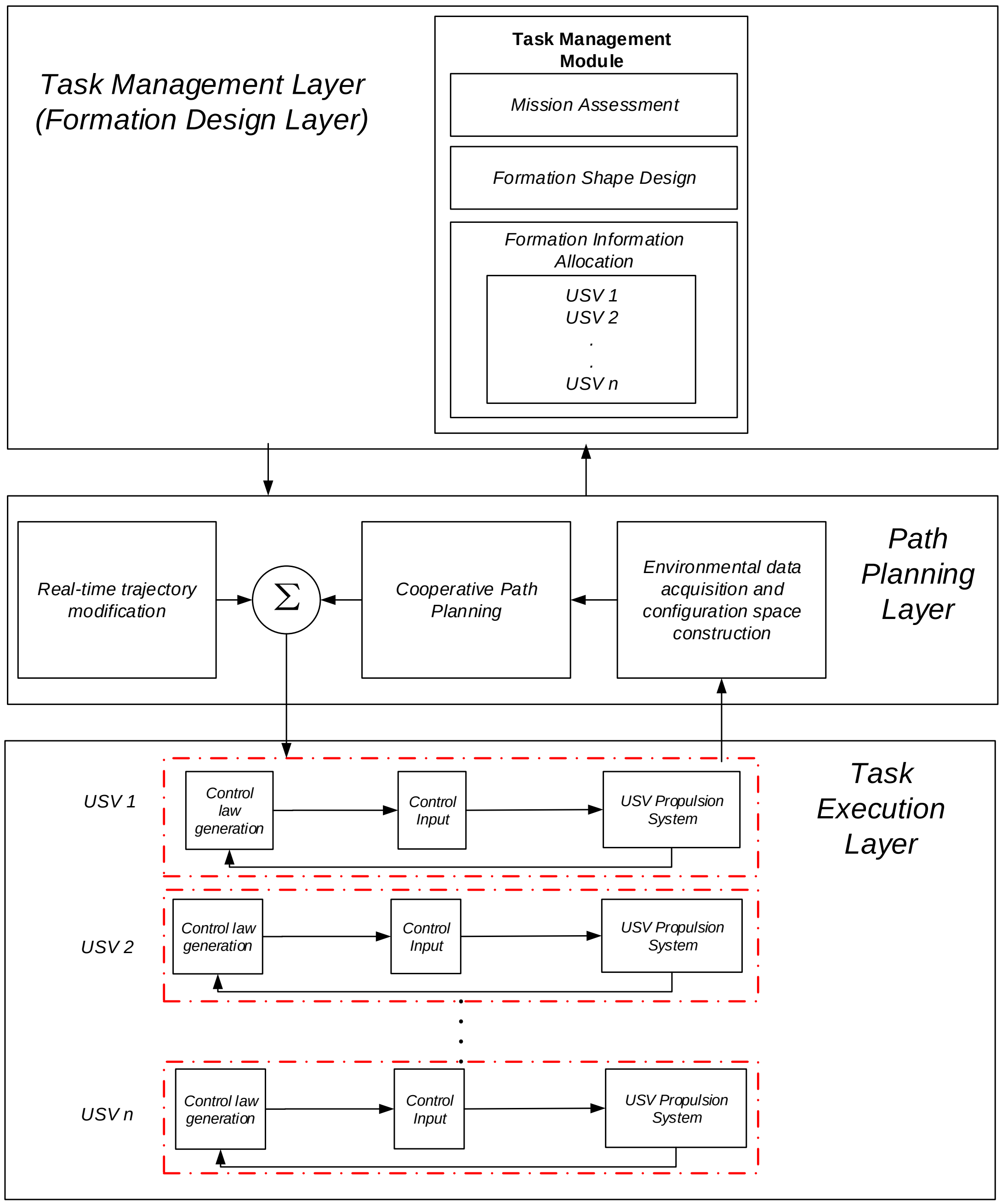



Figure 1: Hierarchy of multiple USVs system.

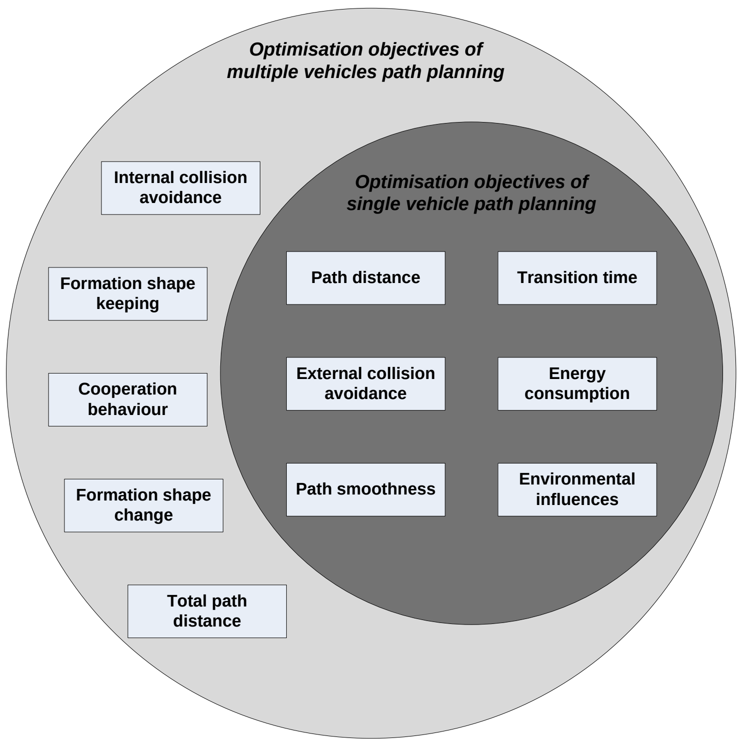



Figure 2: Comparison of formation and single vehicle path planning.

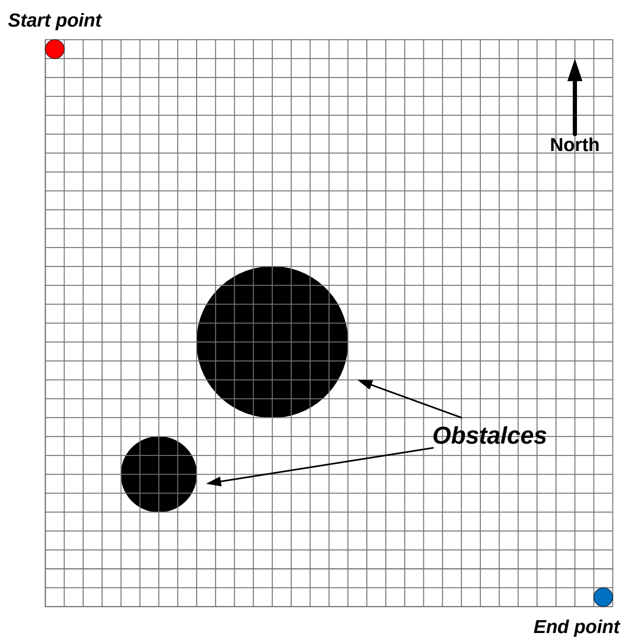



Figure 3: Grid map.

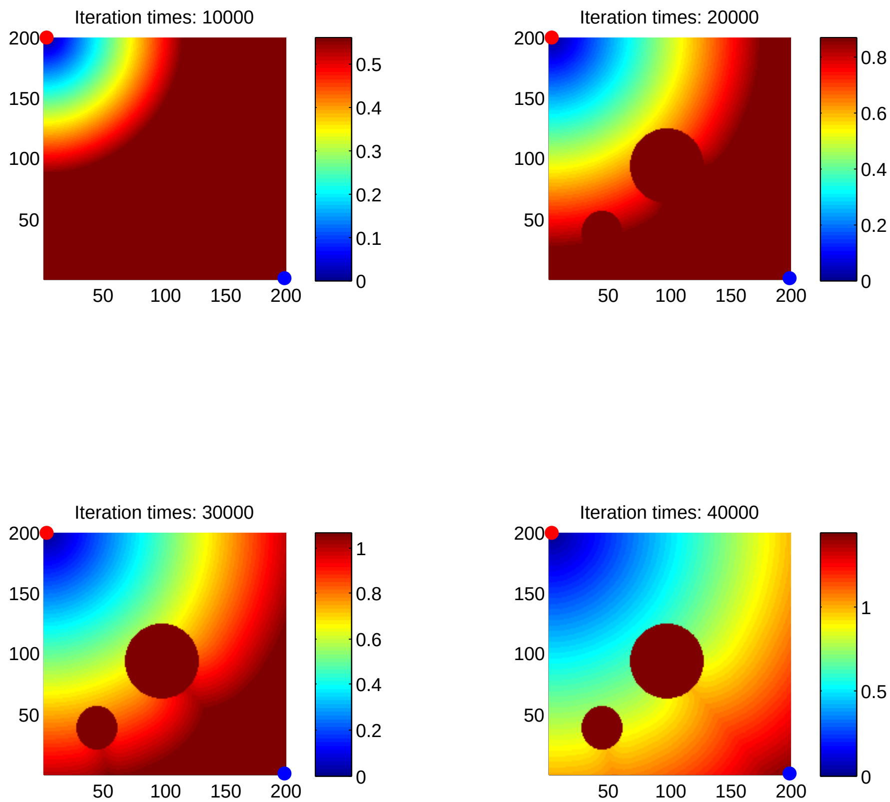
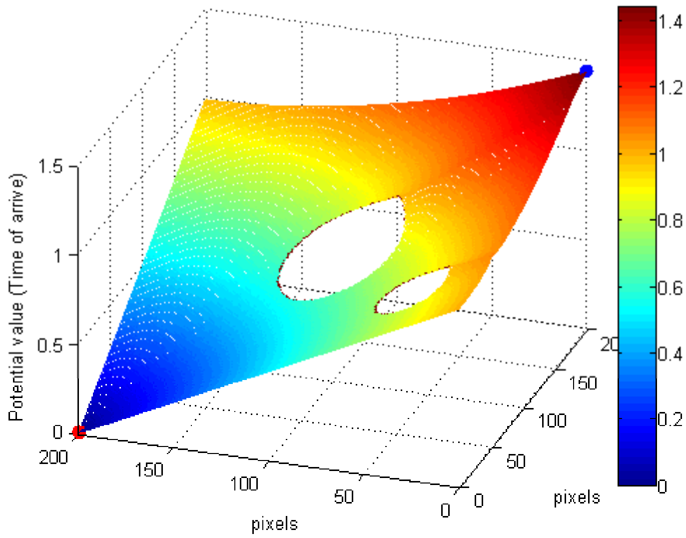
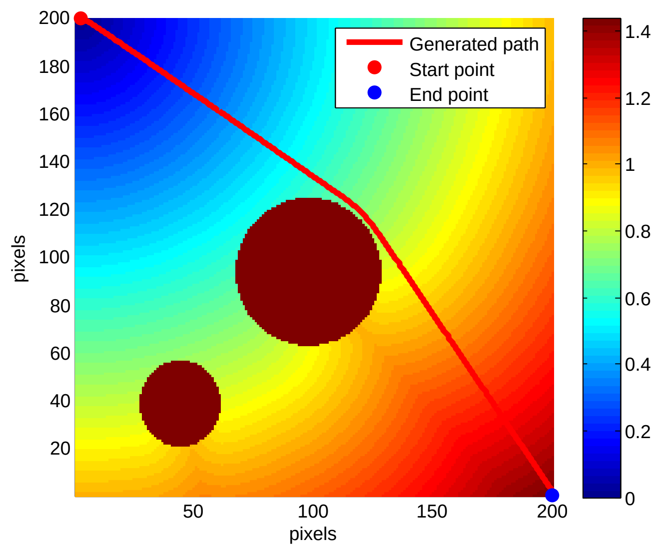



Figure 4: Simulating interface propagation process by using FM method. Interface starts to emit from (0, 200) and ends at (200, 0). Processes are recorded at iteration times 10000, 20000, 30000, 40000 respectively. Colour in the figure represents the local interface arrival time. The brighter the colour is, the longer the arrival time will be.

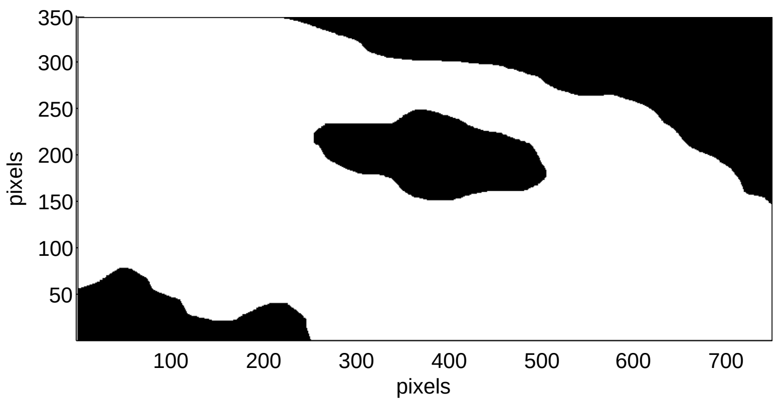



(a)

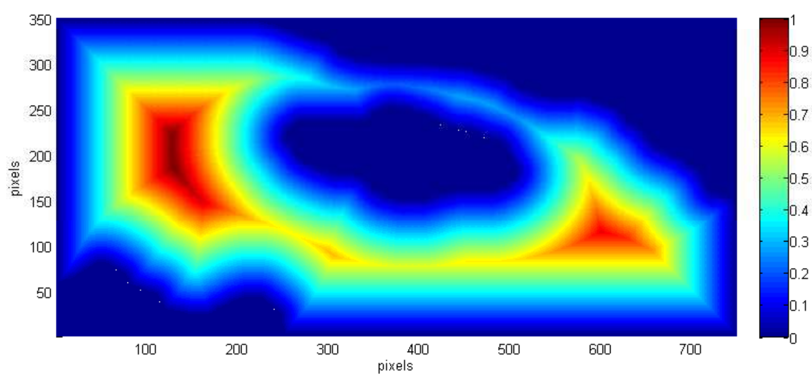



(b)

Figure 5: (a)Potential field generated by running FM method. Local potential value represents local interface arrival time. (b) Path generated by following gradient of potential field.

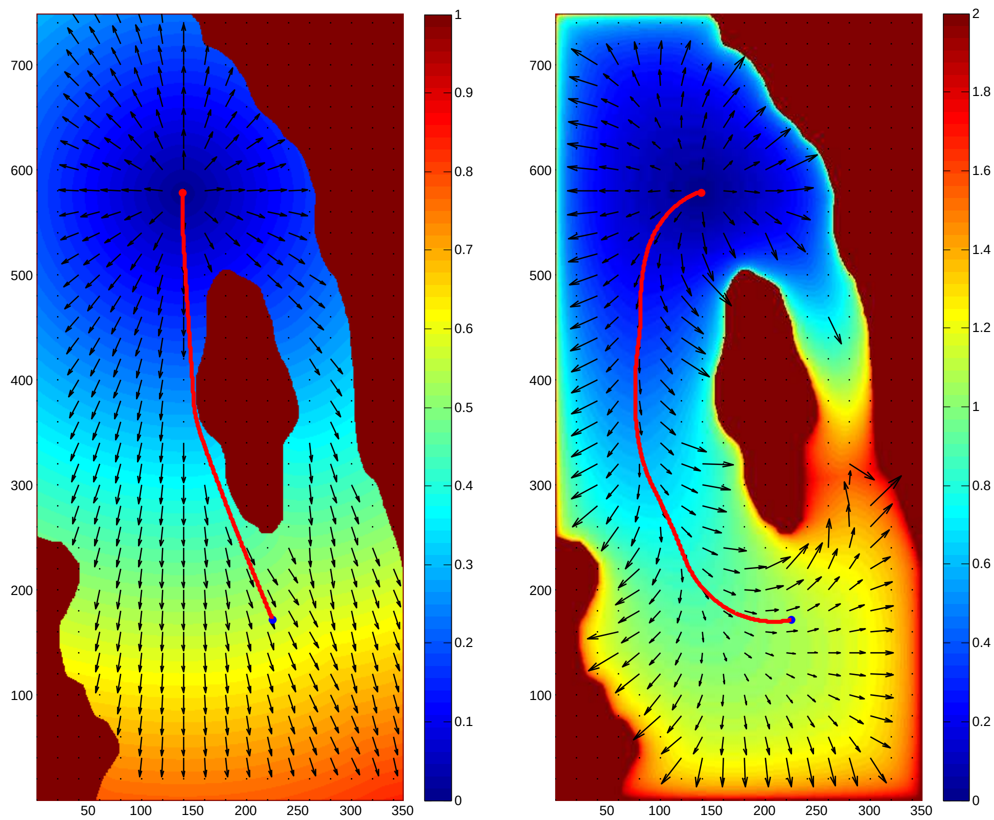



(a)




(b)

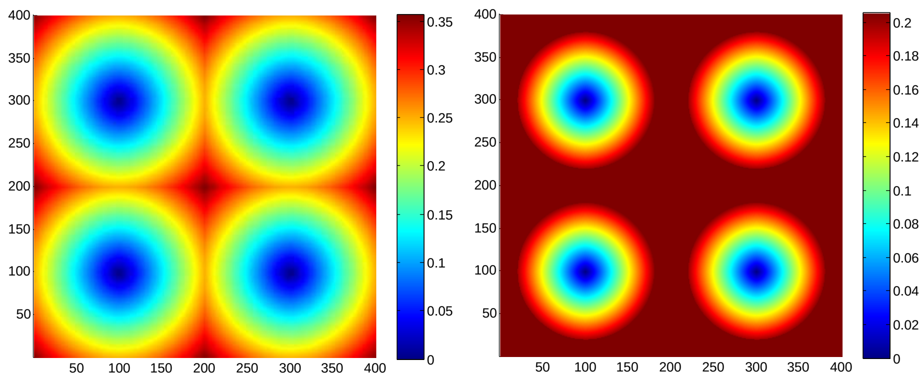
Figure 6: (a) Original environment map ($\boldsymbol{M_o}$) in binary format. (b) Safety map ($\boldsymbol{M_s}$) generated by FM method.




(a)

(b)

Figure 7: (a) Potential field and corresponding path generated by FM method. (b) Potential field and corresponding path generated by FMS method.




(a) (b)

Figure 8: Comparison between conventional FM and constrained FM method. (a)Interface propagation from four start points by using conventional FM method. (b) Interface propagation from four start points by using constrained FM method. Constrained area is built as circle with radius of 20 pixels.

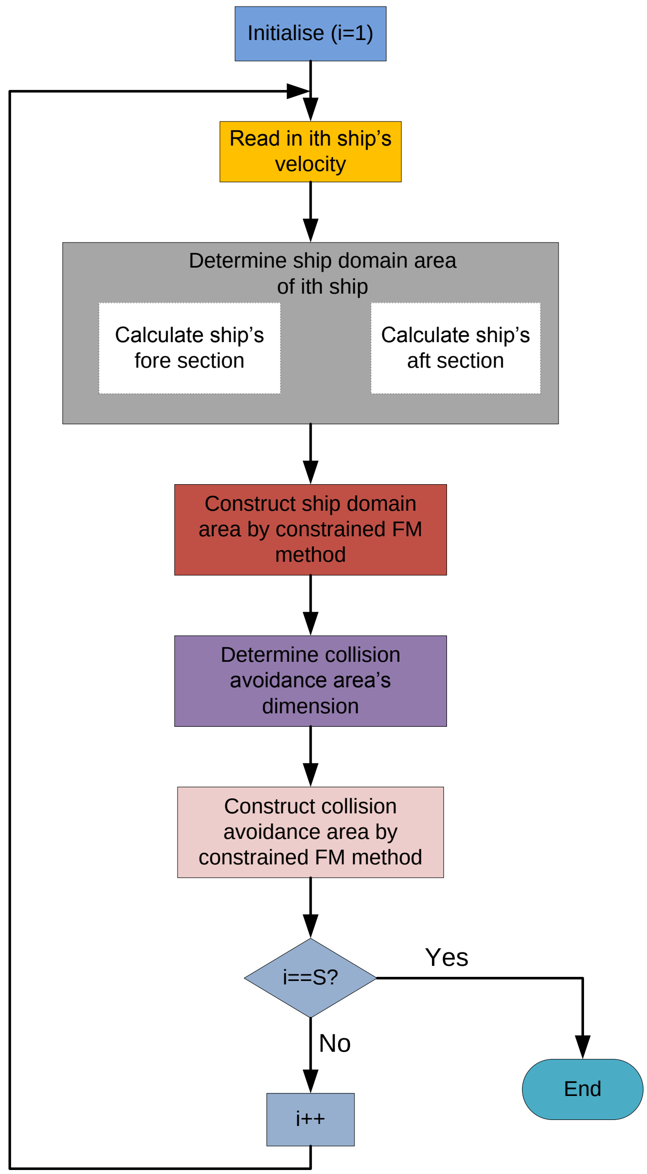



Figure 9: Algorithm flow chart of moving ships modelling by using constrained FM method

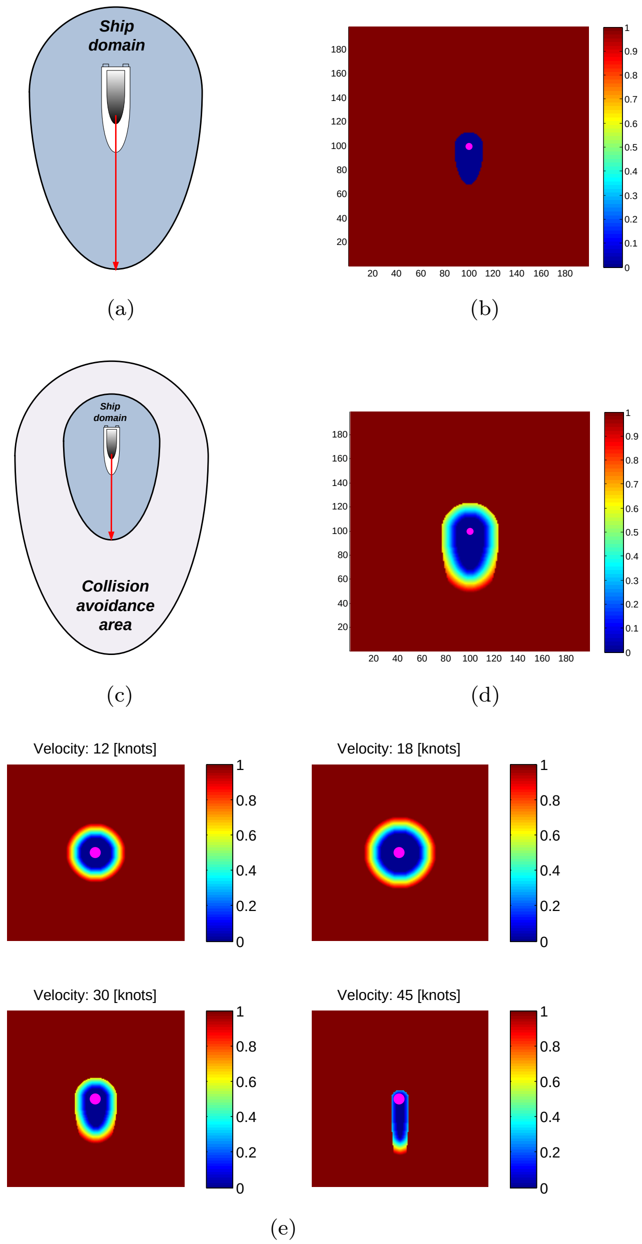



Figure 10: (a) Ship domain area. (b) Ship domain constructed using constrained FM method by using ship's position as start point. (c) Ship domain and collision avoidance area. (d) Collision avoidance area constructed using constrained FM method by using points in ship domain as start points. (e) Different ship domains under different speeds.

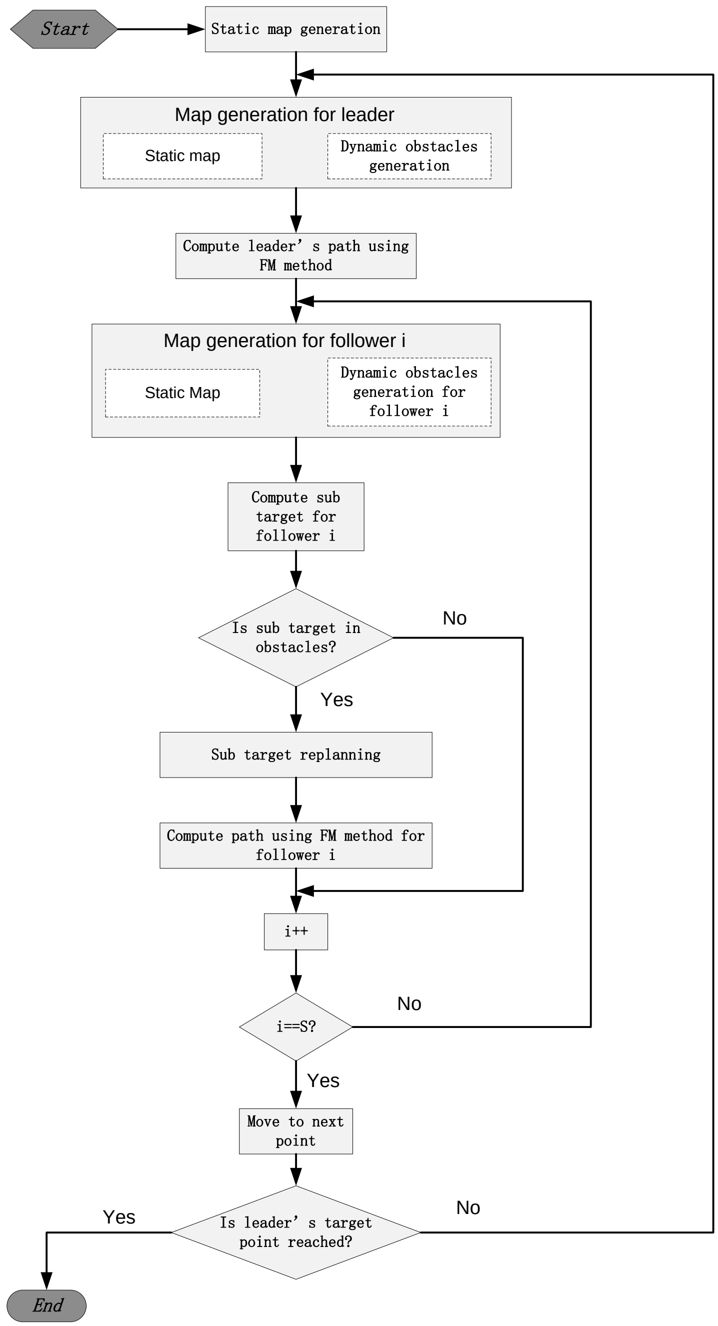



Figure 11: Algorithm flow chart of USV formation path planning

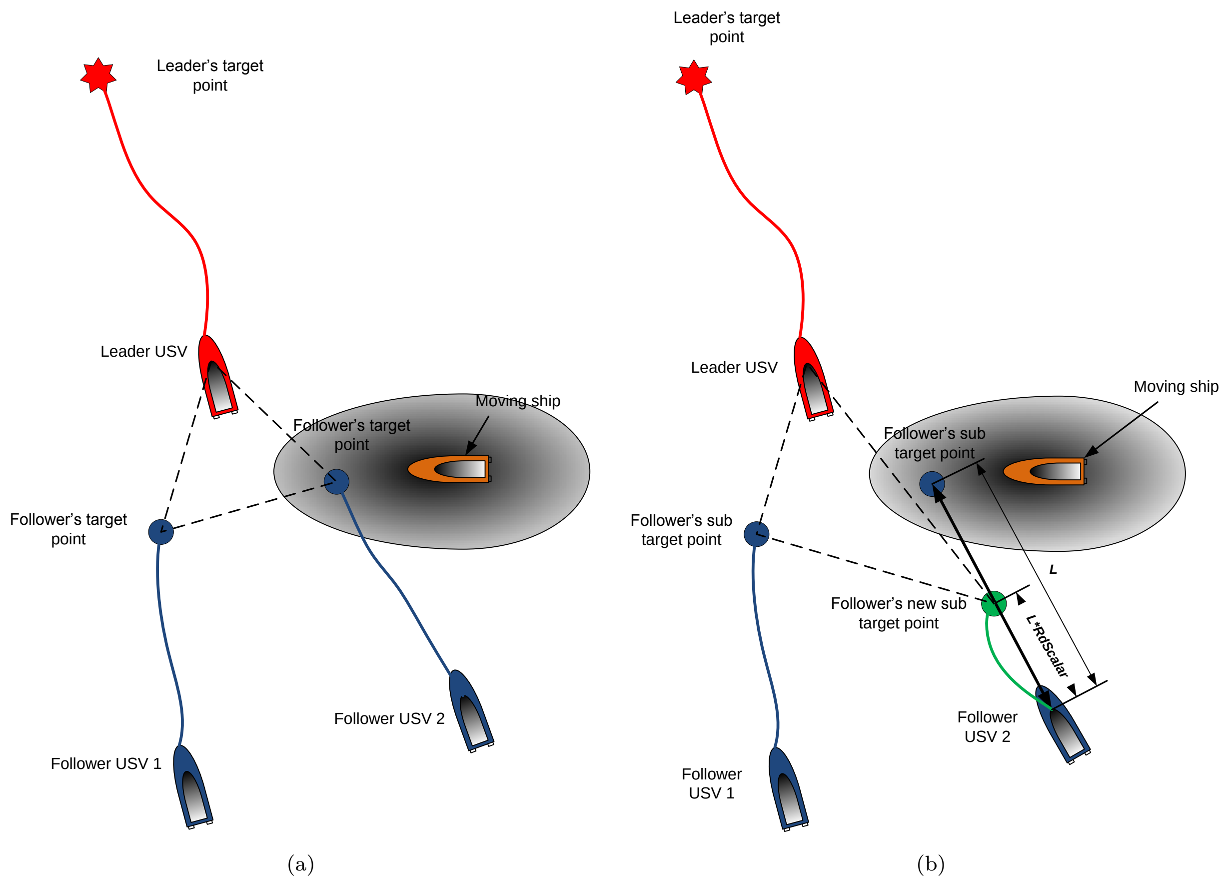
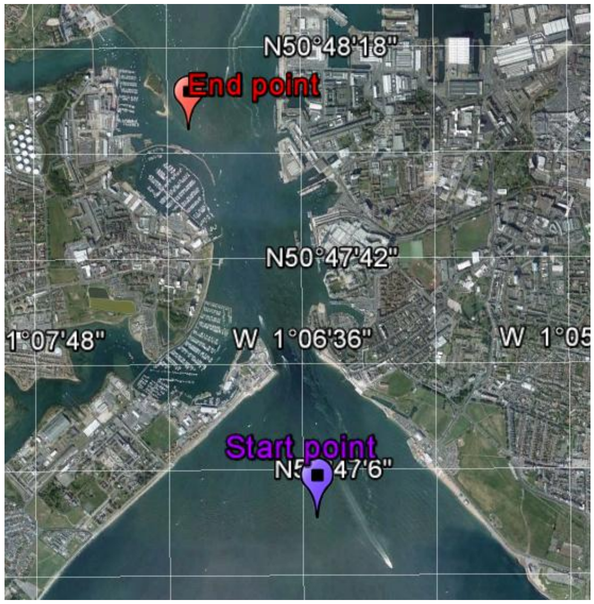
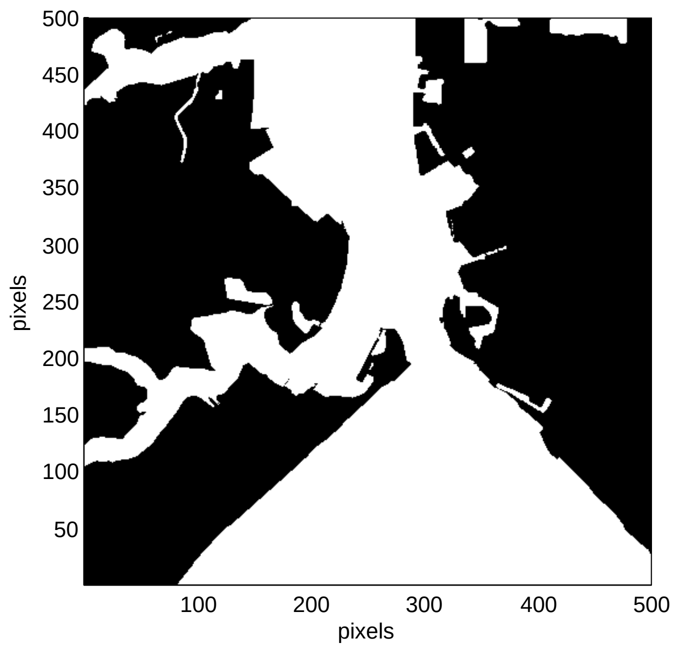



Figure 12: (a) Sub targets planned for two followers with follower2's located in moving ship's ship domain area. (b) Sub target re-planned for follower2, new target point (plotted in green colour) is outside ship domain.




(a)




(b)

Figure 13: (a) Simulation area (Portsmouth harbour). (b) Binary map of simulation area.

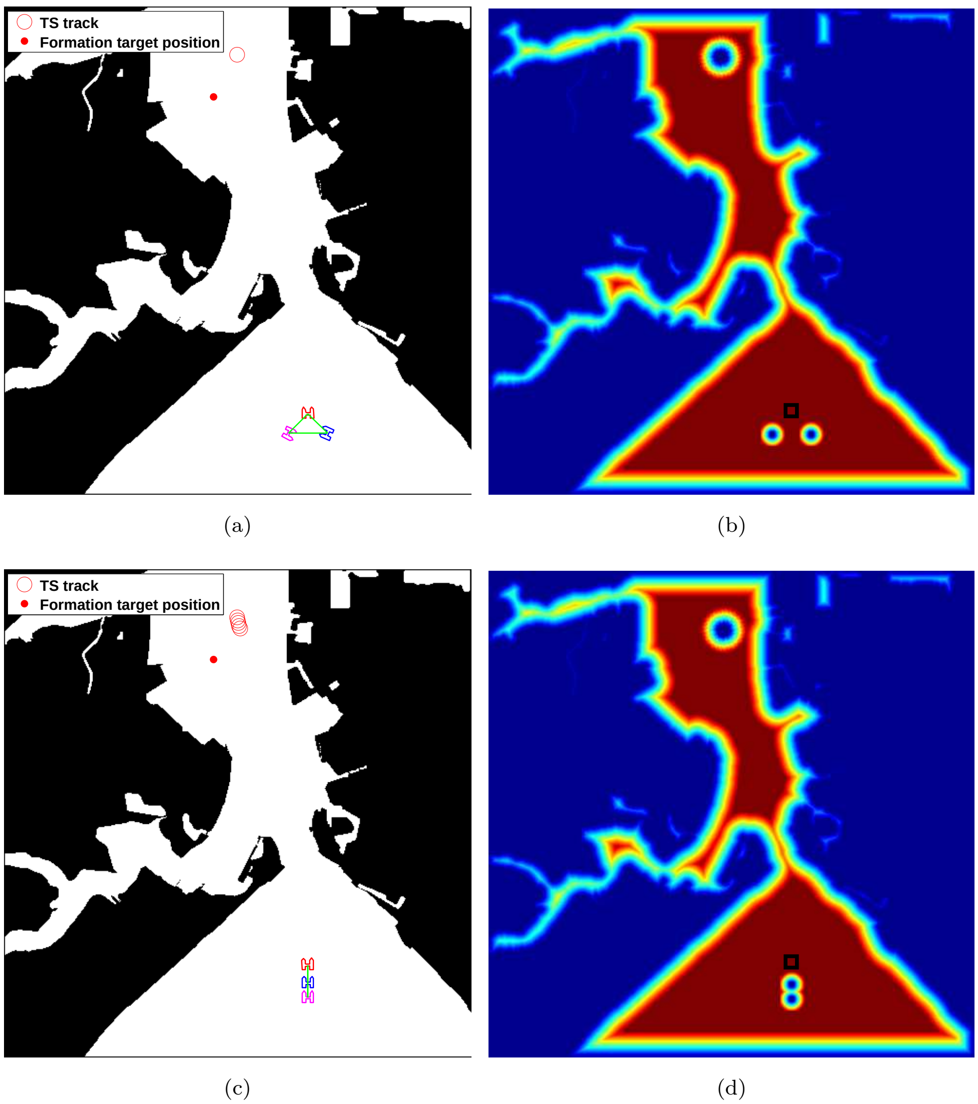



Figure 14: Formation movement sequences and corresponding potential maps. (a)-(b) Time step = 1. (c)-(d) Time step = 5.

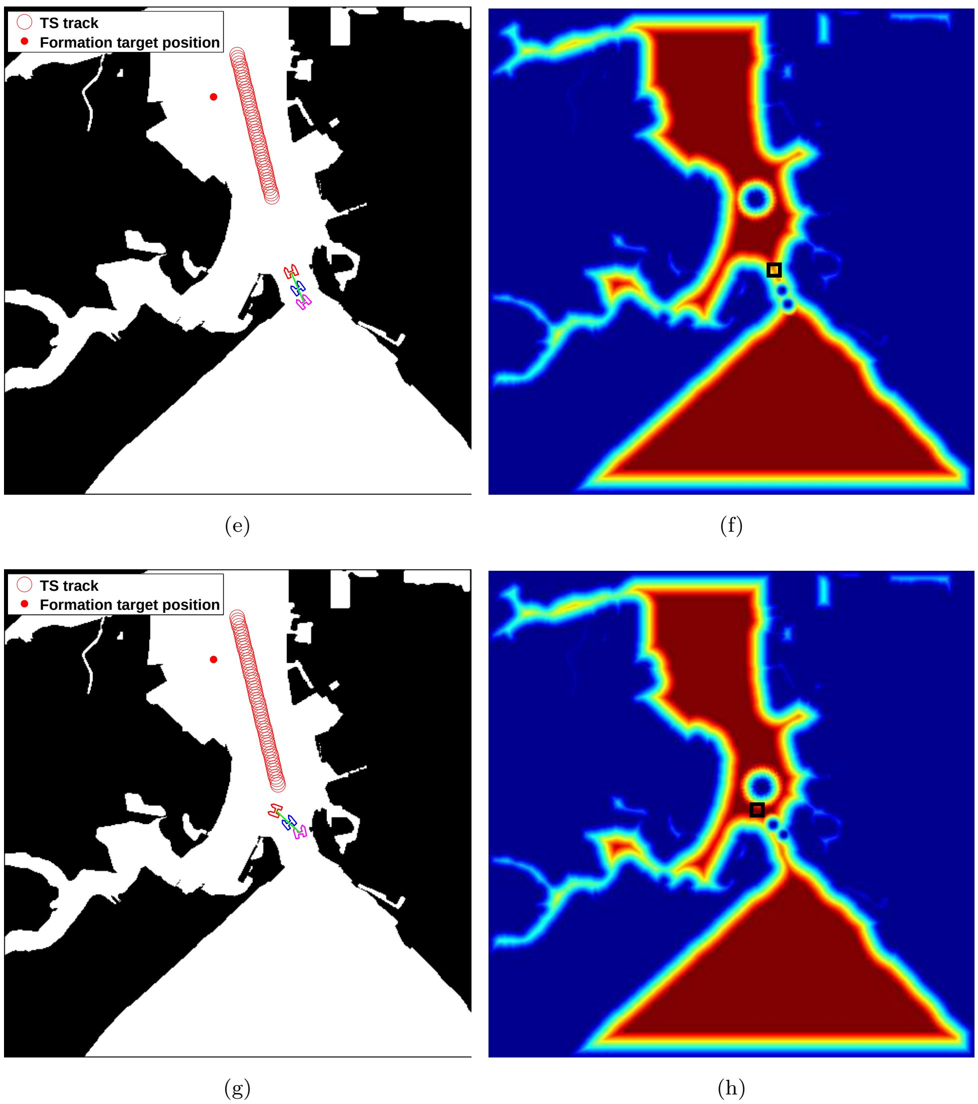
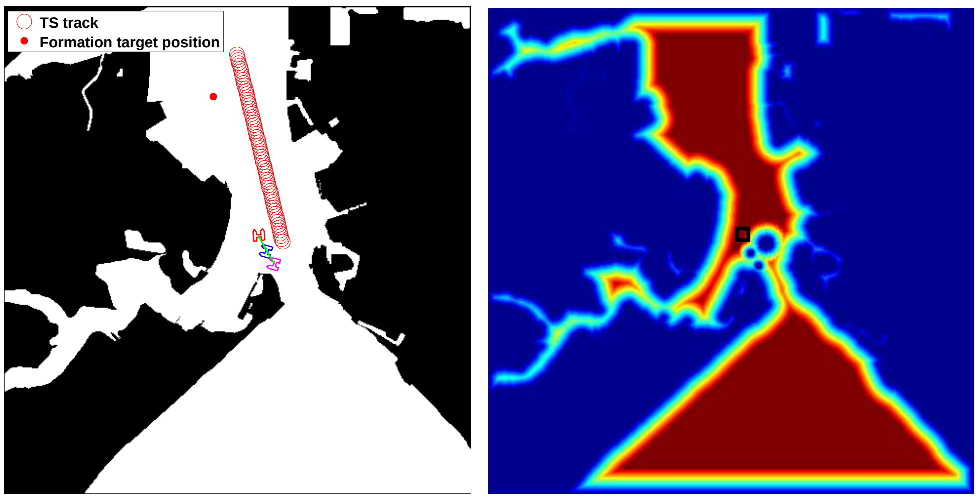
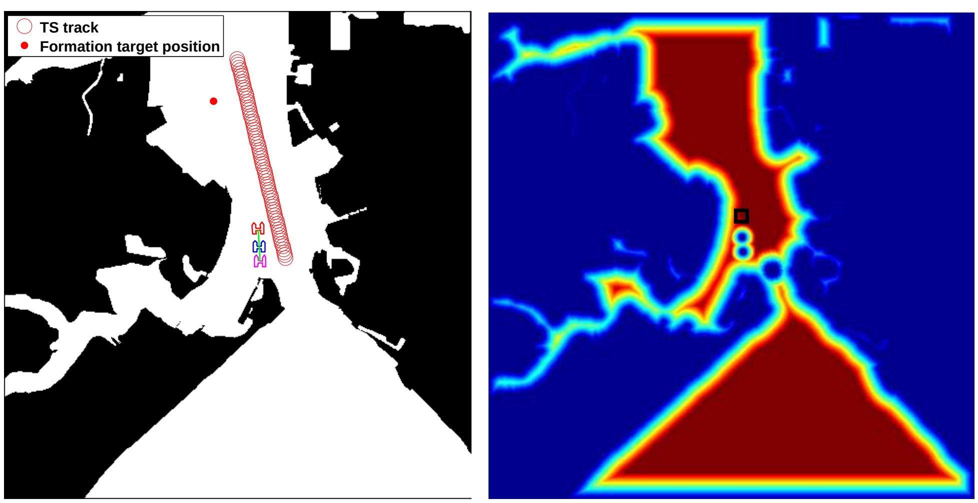



Figure 14: Formation movement sequences and corresponding potential maps. (e)-(f) Time step = 51. (g)-(h) Time step = 60.

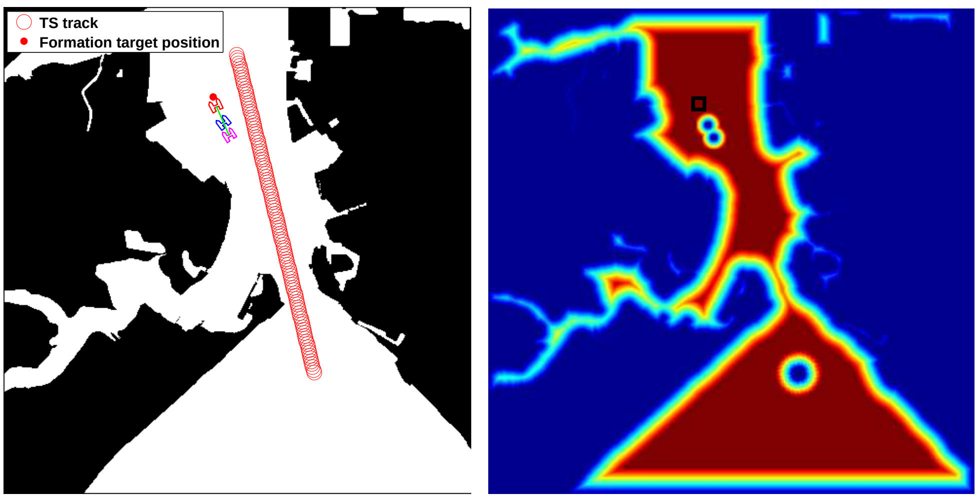



(i) (j)



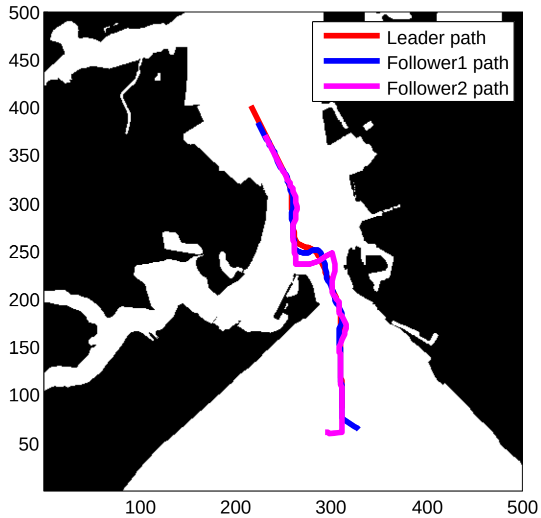

(k) (l)

Figure 14: Formation movement sequences and corresponding potential maps. (i)-(j) Time step = 67. (k)-(l) Time step = 75.

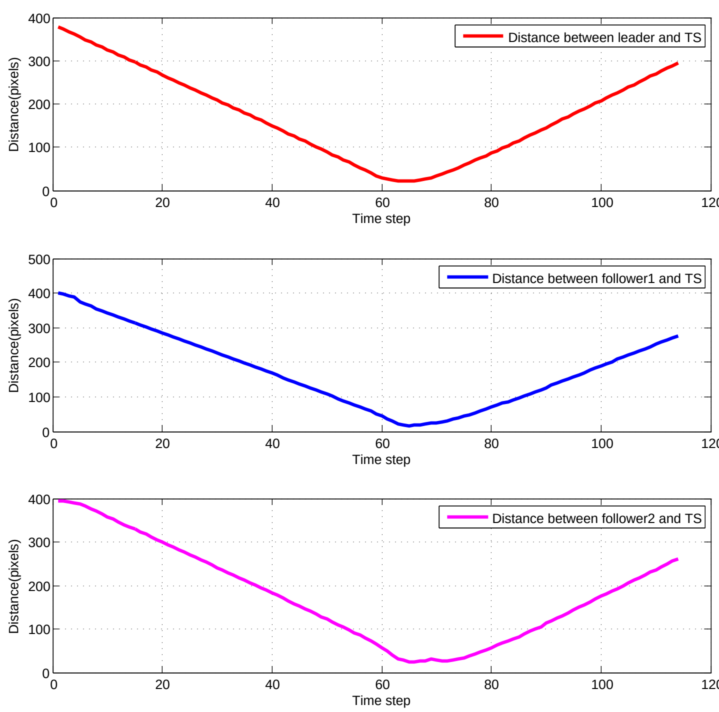



(m) (n)

Figure 14: Formation movement sequences and corresponding potential maps. (m)-(n) Time step = 113.

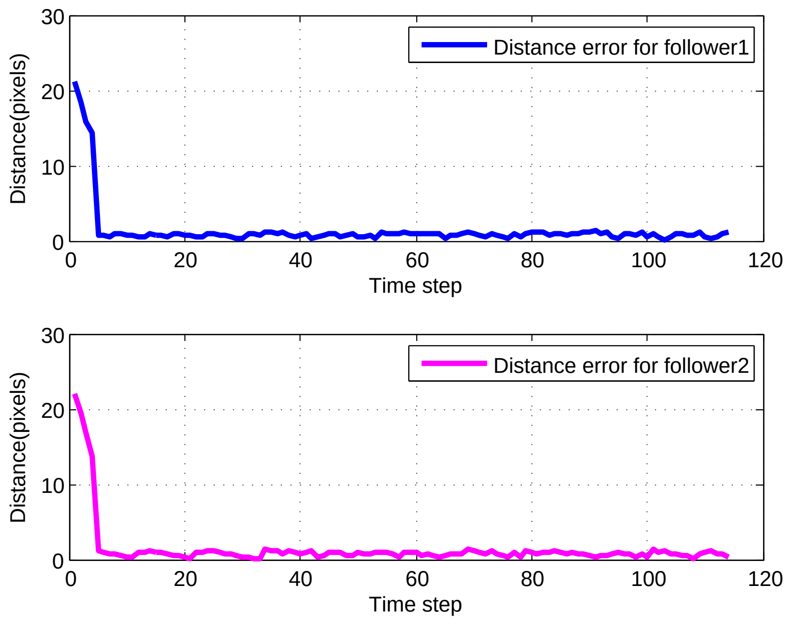



(a)



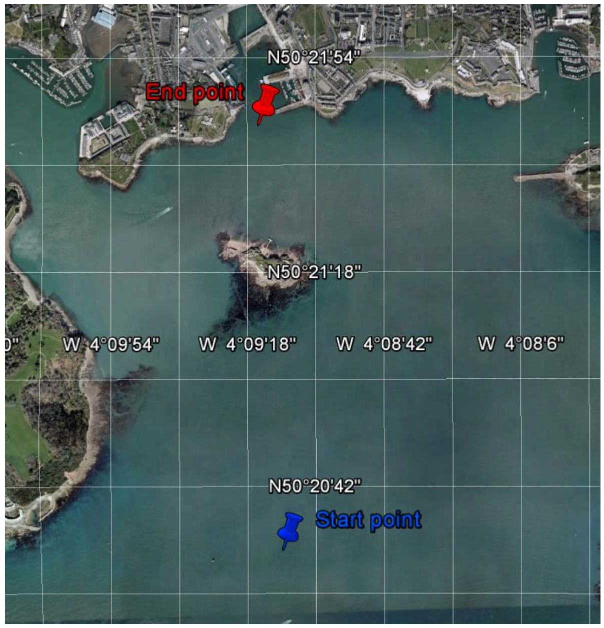

(b)

Figure 15: Evaluation results. (a) Trajectories for formation. (b) Distance between TS and each USV in formation.

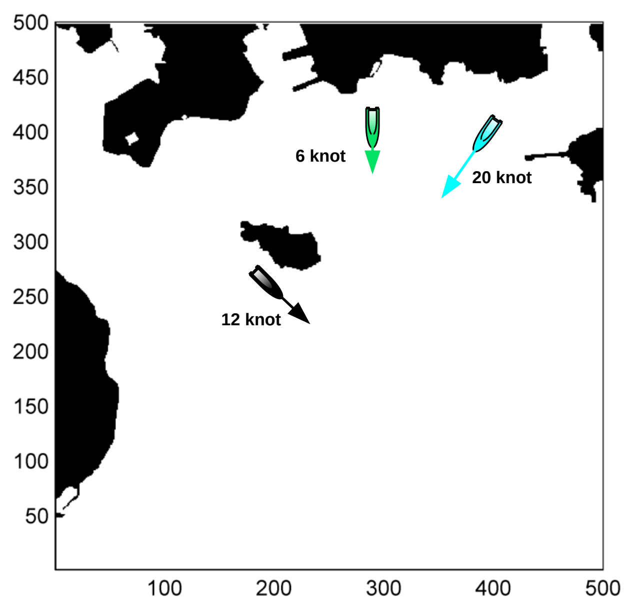



(c)

Figure 15: Evaluation results. (c) Distance errors for follower1 and follower2.




(a)




(b)

Figure 16: (a) Simulation area (Plymouth harbour). (b) Binary map of simulation area.

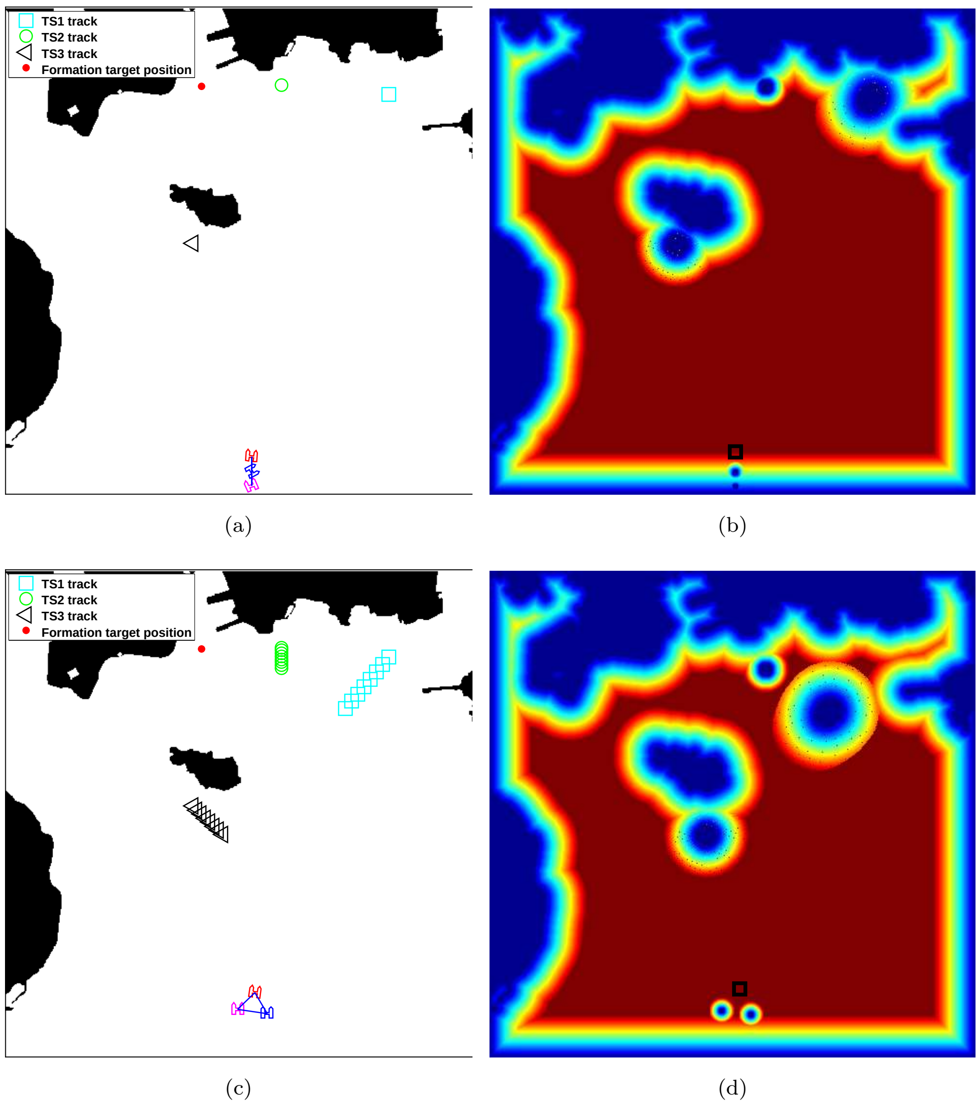



Figure 17: Formation movement sequences and corresponding potential maps. (a)-(b) Time step = 1. (c)-(d) Time step = 8.

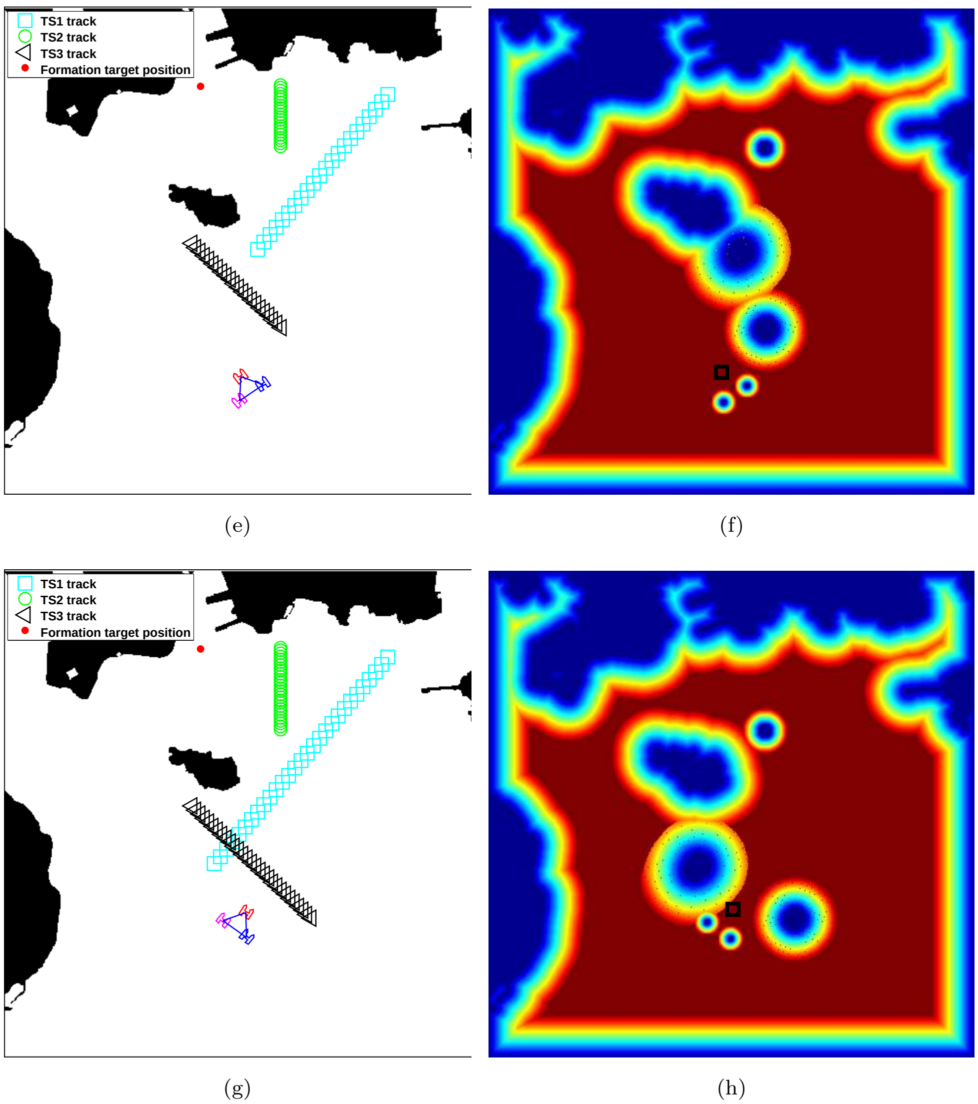
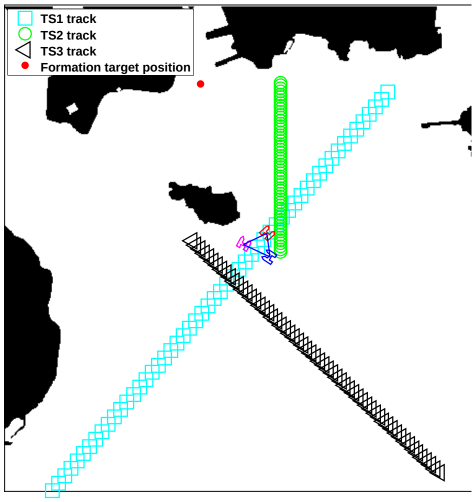
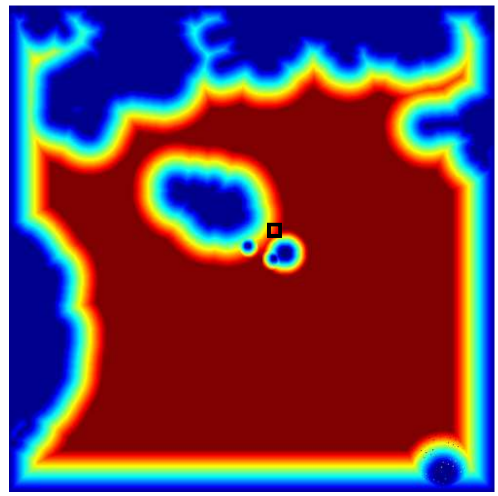
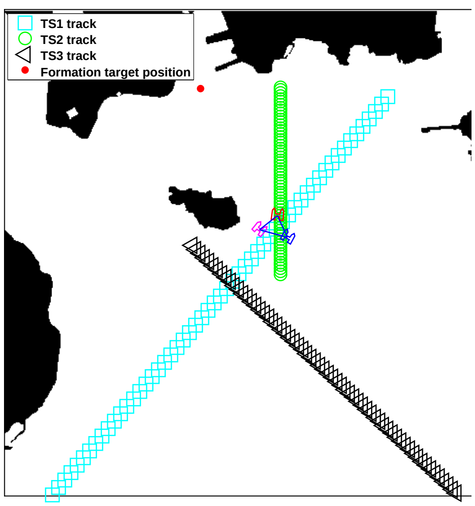


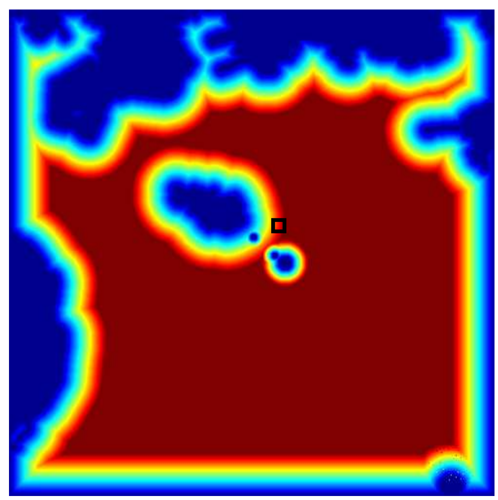

Figure 17: Formation movement sequences and corresponding potential maps. (e)-(f) Time step = 22. (g)-(h) Time step = 29.



(i) (j)



(k) (l)

Figure 17: Formation movement sequences and corresponding potential maps. (i)-(j) Time step = 59. (k)-(l) Time step = 65.

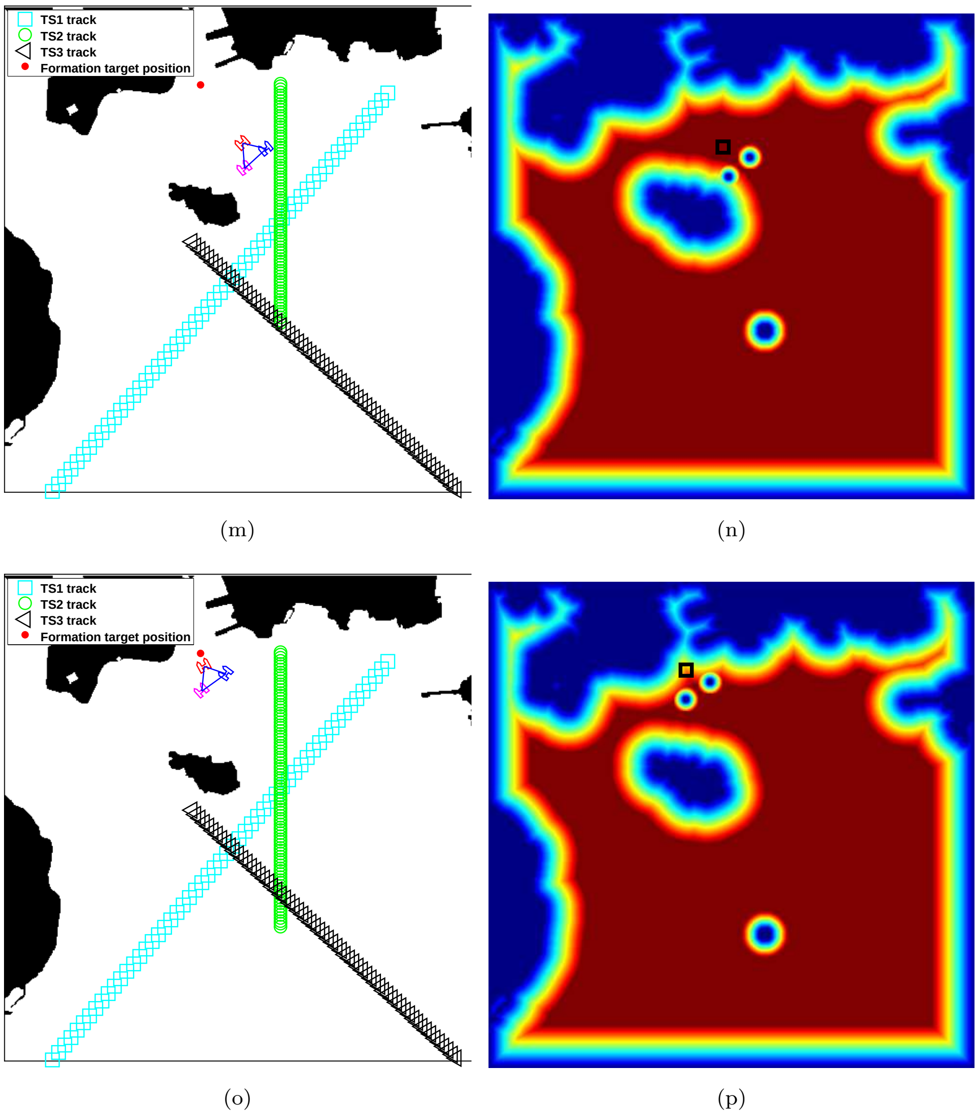
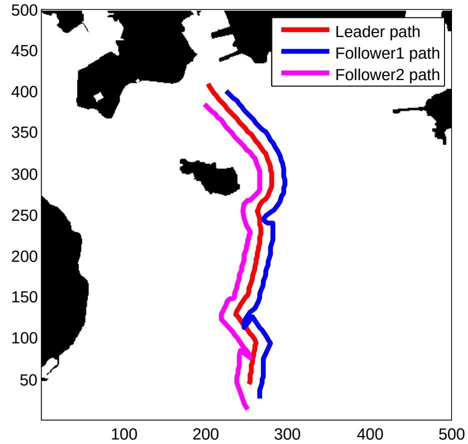
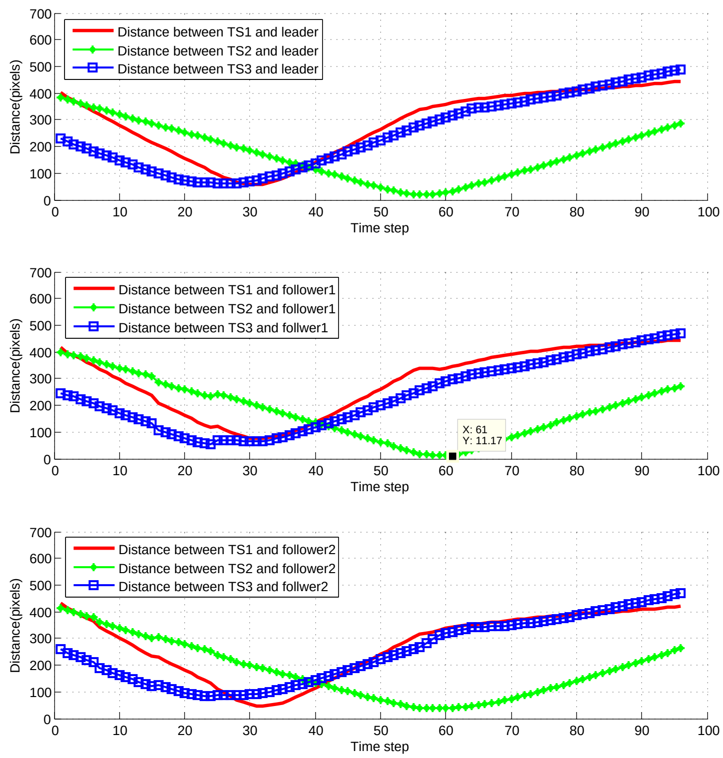



Figure 17: Formation movement sequences and corresponding potential maps. (m)-(n) Time step = 83. (o)-(p) Time step = 97.

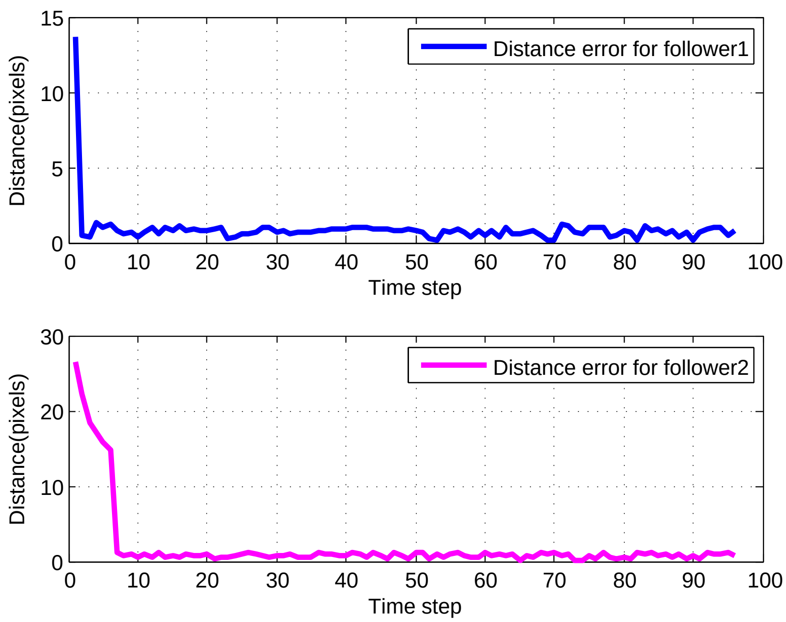



(a)




(b)

Figure 18: Evaluation results. (a) Trajectories for formation. (b) Distance between target ships and each USV in formation.




(c)

Figure 18: Evaluation results. (c) Distance errors for follower1 and follower2.