

University of Groningen

## Clustering-based algorithms for multi-vehicle task assignment in a time-invariant drift field

Bai, Xiaoshan; Yan, Weisheng; Cao, Ming

*Published in:*  
IEEE Robotics and Automation Letters

*DOI:*  
[10.1109/LRA.2017.2722541](https://doi.org/10.1109/LRA.2017.2722541)

**IMPORTANT NOTE:** You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

*Document Version*  
Final author's version (accepted by publisher, after peer review)

*Publication date:*  
2017

[Link to publication in University of Groningen/UMCG research database](#)

*Citation for published version (APA):*

Bai, X., Yan, W., & Cao, M. (2017). Clustering-based algorithms for multi-vehicle task assignment in a time-invariant drift field. IEEE Robotics and Automation Letters, 2(4), 2166-2173. DOI: 10.1109/LRA.2017.2722541

**Copyright**

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

**Take-down policy**

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

# Clustering-based algorithms for multi-vehicle task assignment in a time-invariant drift field\*

Xiaoshan Bai<sup>1,2</sup>, Weisheng Yan<sup>1</sup>, and Ming Cao<sup>2</sup>

**Abstract**—This paper studies the multi-vehicle task assignment problem where several dispersed vehicles need to visit a set of target locations in a time-invariant drift field while trying to minimize the total travel time. Using optimal control theory, we first design a path planning algorithm to minimize the time for each vehicle to travel between two given locations in the drift field. The path planning algorithm provides the cost matrix for the target assignment, and generates routes once the target locations are assigned to a vehicle. Then, we propose several clustering strategies to assign the targets, and we use two metrics to determine the visiting sequence of the targets clustered to each vehicle. Mainly used to specify the minimum time for a vehicle to travel between any two target locations, the cost matrix is obtained using the path planning algorithm, and is in general asymmetric due to time-invariant currents of the drift field. We show that one of the clustering strategies can obtain a min-cost arborescence of the asymmetric target-vehicle graph where the weight of a directed edge between two vertices is the minimum travel time from one vertex to the other respecting the orientation. Using tools from graph theory, a lower bound on the optimal solution is found, which can be used to measure the proximity of a solution from the optimal. Furthermore, by integrating the target clustering strategies with the target visiting metrics, we obtain several task assignment algorithms. Among them, two algorithms guarantee that all the target locations will be visited within a computable maximal travel time, which is at most twice of the optimal when the cost matrix is symmetric. Finally, numerical simulations show that the algorithms can quickly lead to a solution that is close to the optimal.

## I. INTRODUCTION

Multi-vehicle systems have been increasingly exploited to effectively and efficiently accomplish difficult and complex missions [1]. The associated multi-vehicle task assignment problem is to assign a fleet of vehicles to visit a set of target locations, while trying to minimize the vehicles' total travel distance [2] or time [3]. The task assignment problem for a team of vehicles to visit a set of target locations is a variant of the vehicle routing problem (VRP) where a fleet of vehicles need to deliver products from one or several depots to a group of dispersed customers [4], [5]. The VRP is NP-hard, which implies that extremely long computation time may be needed to obtain the optimal solution as the numbers of vehicles and

customers grow. So existing research works usually test their algorithms on the VRP benchmarks and compare the results with those existing solutions of known performances [6]. The VRP has been dealt with using centralized computation methods, including exact [7], [8], and heuristic algorithms [6], [9], [10]; genetic algorithms (GAs) are the typical representative of the latter [11]. The multi-vehicle task assignment problem also has been shown to be NP-hard [12], and heuristic algorithms are usually used to obtain a sub-optimal solution [13], [14]. Taking into account task priority and vehicle loading capacity, Shima et al. [13] designed a GA for multiple unmanned aerial vehicles. If the utility of a task is non-increasing as other tasks are added to the bundle list before the task of concern, the auction algorithms proposed in [14] guaranteed that their solution's objective value is within twice of the optimal. Furthermore, when the matrix specifying the cost for a vehicle to travel between each pair of locations is symmetric, the Prim Algorithm for the multi-robot task assignment ensured that the robots' total travel cost is at most twice of the optimal [15]. However, most of the discussed task assignment algorithms have been developed under the restrictive assumption that there is no external disturbance when a vehicle travels between two given locations.

When a vehicle's motion is affected by external disturbance such as winds or currents, the multi-vehicle task assignment problem consists of two sub-problems, namely how to assign subtasks as sequences of target locations to individual vehicles and how to navigate a vehicle from its initial location to a target location optimally. There are some research works considering both the target assignment and path planning for the employed vehicles [16]–[18]. By simply requiring moving in straight lines between prescribed locations, Han and Chung [16] employed an autonomous underwater vehicle (AUV) to optimally visit several target points considering the ocean currents and obstacles. Furthermore, to enable multiple AUVs to visit several target points in the time-varying (in a discrete time scale) 3-D underwater environment, Zhu et al. [17] employed the velocity synthesis approach to enable each AUV to reach its targets along the shortest path and used the self-organizing map neural network to realize the multi-AUV target point assignment. Grid-modeling based graph methods were designed by Eichhorn [18] for vehicle path planning in a time-varying environment. Delmerico et al. [19] used active aerial exploration for robot path planning through an unknown terrain for search and rescue missions. The path planning methods minimizing the travel distance between

\*This work is supported in part by the China Scholarship Council (grant No. 201506290081), the National Natural Science Foundation of China (grant No. 61633002) and the European Research Council (ERC-StG-307207).

<sup>1</sup> Xiaoshan Bai and Weisheng Yan are with School of Marine Science and Technology, Northwestern Polytechnical University, 127 West Youyi Road, Xi'an, 710072, China. [wsyan@nwpu.edu.cn](mailto:wsyan@nwpu.edu.cn)

<sup>2</sup> Xiaoshan Bai and Ming Cao are with Faculty of Science and Engineering, University of Groningen, Groningen 9747 AG, The Netherlands. [{xiaoshan.bai,m.cao}@rug.nl](mailto:{xiaoshan.bai,m.cao}@rug.nl)

two given locations in [16], [17] do not necessarily lead to the minimal travel time between the locations. More importantly, since the metric matrix representing the minimal travel time between the target locations is in general asymmetric, the existing algorithms, e.g. the Prim algorithm [15], may fail to guarantee their performances.

In our previous work [20], the multi-AUV routing problem was studied in temporally piece-wise constant ocean currents aiming at minimizing the total travel time. In addition, time-optimal coverage control of multiple vehicles in a drift field was studied in [21] where the time-optimal paths were generated over a sequence of discrete time instants. In this paper, we investigate the task assignment problem for which several dispersed vehicles need to visit a set of target locations in a time-invariant drift field while trying to minimize the total travel time. To solve the problem, we first design a path planning method to deal with the vehicle path planning in currents. Then, we propose several clustering strategies to assign the target locations to the vehicles, and we use two metrics to put the target locations assigned to each vehicle in an ordered sequence. Our main contributions are as follows. Firstly, based on the accessible area analysis and optimal control theory, the proposed planning algorithm can generate the time-optimal path for a vehicle to travel between two prescribed locations in a drift field, which provides the travel cost matrix to be used later for the task assignment. Secondly, a lower bound on the optimality of the solution to the task assignment problem with the asymmetric travel cost matrix is achieved using one of the proposed clustering strategies. As the task assignment problem is NP-hard [12], [15], the lower bound can be used to approximately measure the quality of a solution. Lastly, we have studied how the asymmetric travel cost matrix caused by the drift field influences the performances of different clustering algorithms. Two novel algorithms, in the form of integrating the clustering strategies with the target-inserting metrics, guarantee that the total travel time to visit all the target locations is within a reasonable computable upper bound, which, when the cost matrix is symmetric, is twice of the optimal.

The rest of this paper is organized as follows. In Section II, the formulation of the task assignment problem is given. Section III presents the path planning algorithm which generates the optimal navigation control law, and in Section IV several target clustering strategies and two target inserting metrics are discussed. We present the simulation results in Section V and conclude the paper in Section VI.

## II. PROBLEM FORMULATION

To formulate the problem rigorously, we first introduce the definition of the arborescence of a digraph in graph theory.

*Definition 1:* (arborescence [22]) An *arborescence* is a digraph with a single root in which, there is exactly one directed path from the root to any other vertex.

Based on Definition 1, we extend the concept of arborescence with a single root to a general one with several roots.

*Definition 2:* (Generalized arborescence) A *generalized arborescence* is a digraph with several roots in which, there is exactly one directed path from one and only one of all the roots to any non-root vertex.

Now we are ready to define the research problem.

### A. Problem description

Consider a fleet of  $m$  homogeneous vehicles initially randomly distributed in a planar time-invariant drift field. They need to visit  $n$  target locations while trying to minimize the total travel time. The vehicles are not required to return to their initial locations after visiting the targets (namely we are considering a variation of the open vehicle routing problem [23]), and their net speed is affected by the speed of the currents in the drift field.

### B. Formulation as an optimization problem

We use the vector  $\vec{v}_c = [v_{cx}, v_{cy}]^T$  to describe the drift velocity of the time-invariant field with respect to some coordinate system fixed to the ground. Note that  $\vec{v}_c$  changes with locations. We assume that the vehicles are driven by constant thrust, and consequently their velocity  $\vec{v}$  is with constant speed  $v$  relative to the field [13], [24]. Since the dimension of the drift field is significantly larger than the vehicles' size, we assume that the vehicles are free of turning ratio constraints. The kinematics of each vehicle are

$$\dot{x} = v \cos \psi + v_{cx}, \quad \dot{y} = v \sin \psi + v_{cy}, \quad (1)$$

where  $[x, y]^T$  is the vehicle's position and  $\psi$  is the vehicle's navigation angle.

We label the target locations by  $1, \dots, n$ , and let  $\mathcal{T} = \{1, \dots, n\}$  be the set of these indices. Let  $\mathcal{R}$  denote the set of indices of all the vehicles' initial locations, namely  $\mathcal{R} = \{n+1, \dots, n+m\}$ ,  $m \leq n$ . For each pair of distinct  $i \in \mathcal{T} \cup \mathcal{R}$  and  $j \in \mathcal{T}$ , let  $t(i, j)$  denote the minimal time for a vehicle to travel from  $i$  to  $j$  using a properly designed navigation control. Let  $\sigma_{ij}$  be the path-planning mapping that maps the indices  $i \in \mathcal{T} \cup \mathcal{R}$  and  $j \in \mathcal{T}$  of the starting and ending locations of a vehicle to a binary value, which equals one if and only if it is planned that there exists a vehicle travels from location  $i$  to  $j$ . So  $\sigma_{ii} = 0$  for all  $i \in \mathcal{T} \cup \mathcal{R}$ . Then, the problem is to minimize the total travel time for visiting all the target locations

$$f = \sum_{i \in \mathcal{R} \cup \mathcal{T}, j \in \mathcal{T}} t(i, j) \sigma_{ij}, \quad (2)$$

subject to

$$\sum_{i \in \mathcal{R} \cup \mathcal{T}} \sigma_{ij} = 1, \quad \forall j \in \mathcal{T}; \quad (3)$$

$$\sum_{j \in \mathcal{T}} \sigma_{ij} \leq 1, \quad \forall i \in \mathcal{T} \cup \mathcal{R}; \quad (4)$$

$$\sum_{i, j \in \mathcal{S}} \sigma_{ij} \leq |\mathcal{S}| - 1, \quad \forall \mathcal{S} \subseteq \mathcal{T}, |\mathcal{S}| \geq 2. \quad (5)$$

Constraint (3) ensures that each target location is visited once and only once; (4) ensures that each target and vehicle initial

location is departed at most once; and (5) guarantees that there is no subtour among the target locations.

*Remark 1: If ignoring the effect of the field currents on the speed of the vehicles, the task assignment problem just presented reduces to the uncapacitated multi-depot open vehicle routing problem with the symmetric travel cost matrix [5]. We refer the interested reader to [5] for detailed discussions on the relationship between a standard vehicle routing problem and the multi-depot open vehicle routing problem.*

After formulating the task assignment problem as a constrained minimization problem, we present in the following section a component of the path planning that is critical for solving the overall optimization problem.

### III. PATH PLANNING ALGORITHM GIVEN THE STARTING AND TARGET LOCATIONS

To plan the optimal path that minimizes the travel time in a given field with currents, we first look at the accessible region of a vehicle starting from an arbitrary location. Then using optimal control theory, we construct the navigation rule that guides a vehicle to travel between two given locations following the path using the minimum time.

#### A. Accessible region analysis

As before,  $\vec{v}_c$  is used to denote the velocity of the currents, which changes with location; its amplitude is  $v_c$ . As in (1), the vehicle's velocity relative to the field is  $\vec{v}$  with the amplitude  $v$ . Similarly, we use  $\vec{v}_n$  to denote the vehicle's net velocity with amplitude  $v_n$ . Obviously, the vehicle can reach all locations of the field given enough time if  $v > v_c$ . For this reason, we make this standing assumption for the rest of the paper.

**Assumption 1:** It holds for all locations of the field and all time that  $v > v_c$ .

Consequently, with this assumption, each vehicle can travel from any given location to any given other target location and the travel time depends on the path planned and the associated navigation rule, which will be discussed in the following subsection.

#### B. Optimal navigation law

We now show how to navigate a vehicle between any two given locations with the minimum traveling time.

*Lemma 1:* Under Assumption 1, for a vehicle with kinematics (1), to travel with the minimum time between any given starting and ending locations of the time-invariant drift field with the current velocity  $\vec{v}_c$ , the rate of change of the vehicle's optimal navigation angle  $\psi^*$  must satisfy

$$\begin{aligned} \dot{\psi}^* &= -\frac{\partial v_{cx}}{\partial y} \cos^2 \psi^* + \left( \frac{\partial v_{cx}}{\partial x} - \frac{\partial v_{cy}}{\partial y} \right) \sin \psi^* \cos \psi^* \\ &\quad + \frac{\partial v_{cy}}{\partial x} \sin^2 \psi^*. \end{aligned} \quad (6)$$

*Proof:* Let  $t_0$  and  $t_f$  be the starting and finishing times respectively. Then to minimize the travel time, is to minimize the objective function

$$J = \int_{t_0}^{t_f} dt = t_f - t_0.$$

Define the corresponding Hamiltonian to be

$$\begin{aligned} H(t, [x, y]^T, \lambda, \psi) &= 1 + \lambda^T [\dot{x}, \dot{y}]^T \\ &= 1 + \lambda_1 (v \cos \psi + v_{cx}) \\ &\quad + \lambda_2 (v \sin \psi + v_{cy}), \end{aligned} \quad (7)$$

where  $\lambda = [\lambda_1, \lambda_2]^T$  is the two-dimensional Lagrangian multiplier. From Pontryagin's minimum principle of variational analysis in optimal control theory [25, P188], it must be true that the optimal Lagrangian multiplier  $\lambda^*$  and the optimal navigation angle  $\psi^*$  satisfy

$$\dot{\lambda}^* = -\frac{\partial H}{\partial [x, y]^T} \quad (8)$$

$$0 = \frac{\partial H}{\partial \psi} \quad (9)$$

Since (9) holds for all  $t \geq t_0$ , the time derivative of its right-hand side must also be zero. So we have

$$\dot{\lambda}_1^* \sin \psi^* + \lambda_1^* \dot{\psi}^* \cos \psi^* = \dot{\lambda}_2^* \cos \psi^* - \lambda_2^* \dot{\psi}^* \sin \psi^*.$$

Combining with what can be obtained from (8)

$$\begin{aligned} \dot{\lambda}_1^* &= -\lambda_1^* \frac{\partial v_{cx}}{\partial x} - \lambda_2^* \frac{\partial v_{cy}}{\partial x} \\ \dot{\lambda}_2^* &= -\lambda_1^* \frac{\partial v_{cx}}{\partial y} - \lambda_2^* \frac{\partial v_{cy}}{\partial y}, \end{aligned}$$

the optimal navigation control  $\psi^*$  must satisfy (6) when  $t_f - t_0$  is minimized. ■

Because of Assumption 1, we know that a solution  $\psi$ , and thus the optimal solution  $\psi^*$ , always exist. Theorem 1 gives a necessary condition on  $\dot{\psi}^*$ ; what remains to be determined is the initial orientation  $\psi^*(0)$ . After knowing  $\psi^*(0)$  and  $\dot{\psi}^*$ , the optimal navigation angle  $\psi^*(t)$ ,  $t > 0$ , can be determined through the integration of  $\dot{\psi}^*$  over  $t$ . However, to determine  $\psi^*(0)$  with the initial location  $[x_0, y_0]^T$  and the finishing location  $[x_f, y_f]^T$ , one needs to solve the two-point boundary problem:

$$\begin{cases} x_f = x_0 + \int_{t_0}^{t_f} [v \cos(\psi^*(0) + \int_{t_0}^t \dot{\psi}^* d\tau) + v_{cx}] dt, \\ y_f = y_0 + \int_{t_0}^{t_f} [v \sin(\psi^*(0) + \int_{t_0}^t \dot{\psi}^* d\tau) + v_{cy}] dt. \end{cases} \quad (10)$$

The solution  $\psi^*(0)$  to (10) in general can be found numerically using the shooting method [26]. As becomes clear later in an example, the structure of the current velocity  $[v_{cx}, v_{cy}]^T$  in the field can be utilized to simplify the computation.

Let  $i, j$  and  $k$  be three arbitrary different locations. We first give some property on the optimal travel time matrix of the task assignment problem.

*Lemma 2:* The minimum travel times for a vehicle to travel between two locations can be asymmetric; the minimum travel times between any three locations  $i, j$  and  $k$  satisfy the inequality  $t(i, k) \leq t(i, j) + t(j, k)$ .

*Proof:* We first prove that the minimum travel times between two locations is in general asymmetric. To simplify the proof, we consider the case when the currents  $\vec{v}_c$  are spatially invariant. Thus, the optimal navigation angle for a

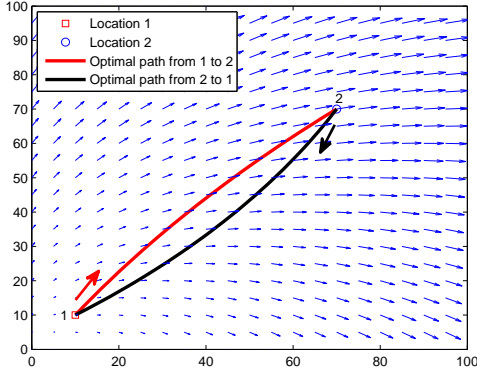


Fig. 1: Optimal path planning for a vehicle with  $v = 1$  to travel between two locations in the drift field  $\vec{v}_c = 10^{-2}[0.3x+0.2y, -0.2x+0.3y]^T$  where the minimum travel times are  $t(1, 2) = 73.0058\text{s}$  and  $t(2, 1) = 103.3586\text{s}$ .

vehicle to travel from  $i$  to  $j$  is constant according to Theorem 1. In other words, the direction of the net velocity for the vehicle to travel from  $i$  to  $j$  is directly towards  $j$  and the magnitude of the net velocity is constant, and vice versa. Consequently, the optimal travel times  $t(i, j)$  and  $t(j, i)$  are asymmetric as long as the magnitudes of the two net speeds are different since the travel distances are both the Euclidean distance between  $i$  and  $j$ . The magnitudes of the net speeds for a vehicle to travel from  $i$  to  $j$  and from  $j$  to  $i$  are the same only when  $\vec{v}_c$  is perpendicular to the vector pointing from  $i$  to  $j$ . Thus, the first half of the statement is proved. (Using the path planning method, we give an example in Fig. 1 to show the property of the asymmetric minimum travel times between two locations.)

Under the optimal navigation law (6), a vehicle takes the minimum travel time  $t(i, k)$  to travel from  $i$  to  $k$ . It is obvious that only if  $j$  is located on the optimal path from  $i$  to  $k$ , one has  $t(i, k) = t(i, j) + t(j, k)$ . On the other hand,  $t(i, k)$  would not be the minimum travel time should  $t(i, k) > t(i, j) + t(j, k)$ , since the navigation law shown in Theorem 1 is time optimal. The proof is complete. ■

#### IV. TASK ASSIGNMENT ALGORITHMS

##### A. Target clustering strategies

In this subsection, three strategies are presented to cluster the target locations to the vehicles based on the optimal travel time matrix  $t = t(i, j)_{i \in \mathcal{R}, j \in \mathcal{T}}$  obtained from the path planning algorithm.

1) *Voronoi clustering*: Inspired by the coverage control study where each vehicle can reach any point of its partitioned area with the shortest travel time among all the vehicles [21], we first propose the Voronoi clustering strategy assigning each target  $k$  to the vehicle  $j^*$  such that

$$j^* = \operatorname{argmin}_{j \in \mathcal{R}} t(p_j, k), \quad (11)$$

where  $t(p_j, k)$  is the minimum travel time for vehicle  $j$  to visit target  $k$  from its initial location  $p_j$ . In the case that a

target location is on one of the boundaries of the Voronoi areas, it is randomly clustered to one of the vehicles whose Voronoi areas share the boundary.

2) *Extended Voronoi clustering*: In the task assignment problem, the vehicles need to visit *all* the target locations which is different from the coverage control problem [21] where a vehicle in essence only visits one target (although its location is unknown beforehand). In other words, Voronoi clustering might lead to assignment unfairness to the target locations. Thus, we extend the Voronoi clustering strategy by assigning each target according to the locations of those targets already assigned and the locations of all the vehicles.

Let  $\mathcal{T}_j$  contain the indices of those targets that have already been assigned to vehicle  $j$  and the target set  $\mathcal{T}^u$  contain the indices of those unclustered targets, which is initialized as  $\mathcal{T}$ . Then, the first target  $k^*$  in  $\mathcal{T}^u$  to be clustered and its assigned vehicle  $j^*$  are determined by

$$(j^*, k^*) = \operatorname{argmin}_{i \in \mathcal{T}_j, j \in \mathcal{R}, k \in \mathcal{T}^u} t(i, k), \quad (12)$$

where the targets already assigned to the vehicles affect the clustering of the remaining targets. After clustering target  $k^*$ ,  $\mathcal{T}^u$  is updated to

$$\mathcal{T}^u = \mathcal{T}^u \setminus \{k^*\}, \quad (13)$$

while the targets assigned to vehicle  $j^*$  are updated to

$$\mathcal{T}_{j^*} = \mathcal{T}_{j^*} \cup \{k^*\}. \quad (14)$$

3) *Marginal-cost-based clustering*: In this subsection, a marginal-cost-based clustering strategy is designed which determines the visiting sequence of a target during its clustering process.

Let  $o_j$  be the route containing the ordered targets already assigned to  $j$ . Then, the first target  $k^*$  in  $\mathcal{T}^u$  to be clustered, its assigned vehicle  $j^*$  and the inserting position  $q^*$  are

$$(k^*, j^*, q^*) = \operatorname{argmin}_{k \in \mathcal{T}^u, j \in \mathcal{R}, q \leq |o_j|+1} t(o_j \oplus_q k) - t(o_j), \quad (15)$$

where the operation  $o_j \oplus_q k$  inserts target  $k$  at the  $q$ th position of  $o_j$ . Target  $k$  is inserted to the end of  $o_j$  if  $q = |o_j| + 1$ , and  $t(o_j)$  denotes the total travel time for vehicle  $j$  to visit all the targets in  $o_j$ .

##### B. Target-visiting metrics

For targets clustered by the strategies in IV-A.1 and IV-A.2, their visiting sequence is not determined. Putting the target locations assigned to each vehicle into a sequence to minimize the vehicle's travel time is in fact the traveling salesman problem (TSP) [27]. In this subsection, we design two target-visiting metrics: the nearest inserting principle and smallest marginal cost principle.

**Algorithm 1** The Extended Voronoi clustering for achieving a min-cost generalized arborescence (MCGA) of a directed graph.

**Input:** Locations of targets in  $\mathcal{T}$  and vehicles in  $\mathcal{R}$ , the travel time matrix  $t$  for digraph  $\mathcal{G}$ .

**Output:** An MCGA of  $\mathcal{G}$ .

```

1: Initialize  $MCGA \leftarrow \mathcal{R}$ .
2: while  $\mathcal{T} \neq \emptyset$  do
3:    $(j^*, p^*) \leftarrow \operatorname{argmin}_{(j,p) \in MCGA \times \mathcal{T}} t(j, p)$ .
4:   Add  $p^*$  in  $MCGA$  and connect it with  $j^*$  using an edge
   with weight  $t(j^*, p^*)$ .
5:    $\mathcal{T} \leftarrow \mathcal{T} \setminus \{p^*\}$ .
6: end while

```

*1) Nearest inserting principle:* The first metric is the nearest inserting principle where vehicle  $j$  always inserts an unordered target location in  $\mathcal{T}_j$  with the smallest travel time into the end of its route  $o_j$ . Let the target set  $\mathcal{T}_j^u$  contain the targets in  $\mathcal{T}_j$  that have not been inserted into  $o_j$ . Then, the first target in  $\mathcal{T}_j^u$  to be inserted for vehicle  $j$  is

$$k^* = \operatorname{argmin}_{i=o_j(|o_j|), k \in \mathcal{T}_j^u} t(i, k), \quad (16)$$

Then,  $\mathcal{T}_j^u$  and  $o_j$  are updated as

$$\mathcal{T}_j^u = \mathcal{T}_j^u \setminus \{k^*\}, \quad o_j = o_j \oplus_{|o_j|+1} k^*. \quad (17)$$

The inserting procedure continues until all the targets in  $\mathcal{T}_j$  are inserted into vehicle  $j$ 's target list  $o_j$ .

*2) Smallest marginal cost principle:* The other one is the smallest marginal cost principle, which determines the first target  $k^*$  in  $\mathcal{T}_j^u$  to be inserted and its visiting sequence  $q^*$  for each vehicle  $j$  by

$$(k^*, q^*) = \operatorname{argmin}_{q \leq |o_j|+1, k \in \mathcal{T}_j^u} t(o_j \oplus_q k) - t(o_j). \quad (18)$$

Then,  $\mathcal{T}_j^u$  and  $o_j$  are updated as

$$\mathcal{T}_j^u = \mathcal{T}_j^u \setminus \{k^*\}, \quad o_j = o_j \oplus_{q^*} k^*. \quad (19)$$

### C. Correctness of the proposed strategies

Let  $\mathcal{G}$  be a digraph whose vertices contain all the vehicles' initial positions and the target locations. The weight for a directed edge is the minimum time for a vehicle to travel from the starting vertex to the ending vertex if at least one vertex represents a target location, and is otherwise zero. Compared with the Prim algorithm used to find a minimum spanning tree for a undirected graph [28], we use the clustering strategy proposed in IV-A.2 to obtain a min-cost generalized arborescence (MCGA) for the digraph  $\mathcal{G}$ . The procedure to achieve an MCGA is shown in Algorithm 1. Let  $f_a$  be the sum of all the edge weights of an MCGA of  $\mathcal{G}$ , and  $f_o$  be the optimal objective value in (2). Then, we first investigate some property of the optimal solution to the problem with an asymmetric travel cost matrix.

*Lemma 3:* It holds that  $f_a \leq f_o$ .

*Proof:* We first prove the statement when  $m$ , the number of all the vehicles, is one. In this case, only one vehicle needs to visit all the target locations, which is a

variant of the TSP [29]. An optimal route to visit all the targets is in fact an arborescence of  $\mathcal{G}$  according to Definition 1. As  $f_a$  is the cost of the min-cost arborescence,  $f_a \leq f_o$ .

When  $m > 1$ , from the definition of the generalized arborescence in Definition 2, the optimal solution of the problem is also a generalized arborescence of  $\mathcal{G}$ , in which both the outdegree and indegree of each vertex is at most one. As  $f_a$  is the sum of all the edge weights of the min-cost generalized arborescence, the proof is complete. ■

The min-cost generalized arborescence contains exactly  $m - 1$  zero cost edges where  $m$  is the number of all the vehicles. Removing the zero cost edges to get an arborescence for each vehicle and duplicating each directed edge of the arborescence but with the opposite direction, we can construct a Eulerian graph for each vehicle (this is inspired by the multi-vehicle algorithm [30]). Let  $f_{da}$  be the sum of all the edge weights of the arborescences after duplicating their directed edges.

*Lemma 4:* The optimal total travel time  $f_o$  is upper bounded by  $f_o \leq f_{da}$ , where  $f_{da} = 2f_a$  if the travel cost matrix is symmetric.

*Proof:* For the first statement, similar to the multi-vehicle algorithm operating on undirected graphs [30], we can obtain a TSP tour for each vehicle based on the corresponding Eulerian graph. As the directed edges satisfy the inequality in Lemma 2, the total travel time of each vehicle is at most the sum of all the edge weights of the duplicated arborescence for each vehicle. Thus, the total travel time of all the vehicles is not greater than the sum of all the edge weights of the duplicated generalized arborescence. As the total travel time of each feasible solution is an upper bound for the optimal solution, the first statement is proved.

When the travel cost matrix is symmetric, the minimum travel times between any two vertices in  $\mathcal{G}$  are the same. Thus,  $f_{da} = 2f_a$  as  $f_{da}$  is the sum of all the edge weights of the duplicated generalized arborescence. ■

Using the extended Voronoi clustering strategy in Algorithm 1, we achieve a min-cost arborescence for each vehicle. Then, we can utilize the target-inserting metrics proposed in IV-B.2 to put the targets on each arborescence into sequence. Integrating the extended Voronoi clustering strategy with the smallest marginal cost principle, we obtain a task assignment algorithm, called *EVM* for simplification. Let  $f_{EVM}$  be the total travel time of the solution resulting from EVM.

*Theorem 1:* The task assignment algorithm EVM guarantees that  $f_{EVM}/f_o \leq f_{da}/f_a$ .

*Proof:* The proof is conducted by induction. The solution resulting from EVM has the same target assignment compared with that of the duplicated min-cost generalized arborescence as they use the same target clustering strategy (12). Let  $f_{da}^j$  be the sum of all the edge weights of the duplicated arborescence for vehicle  $j$ . Then,  $f_{da} = \sum_{j \in \mathcal{R}} f_{da}^j$ . The first target  $k^*$  to be inserted in  $o_j$  is determined by (18) for EVM. It is straightforward to see that the first target inserted in  $o_j$  is the same as the first target inserted in the min-cost arborescence for vehicle  $j$  according to line 3 of

Algorithm 1. Thus,  $f_{EVM}^{j1} \leq f_{da}^{j1}$  as  $f_{EVM}^{j1} = f_{da}^{j1}$ , where the superscripts 1 and  $j$  are associated with the total travel time for vehicle  $j$  to visit the first target inserted in  $o_j$ .

Now suppose the first  $|\mathcal{T}_j| - 1$  targets inserted in  $o_j$  and those inserted in the arborescence for vehicle  $j$  are the same and  $f_{EVM}^{j|\mathcal{T}_j|-1} \leq f_{da}^{j|\mathcal{T}_j|-1}$ , where  $\mathcal{T}_j$  contains all the targets in the end assigned to vehicle  $j$ . As the inequality specified in Lemma 2 holds for the optimal travel times between the vertices in  $\mathcal{G}$  and according to (18), for EVM the marginal travel time incurred by inserting the last target  $k$  into  $o_j$  is

$$\begin{aligned} \delta f_{EVM}^j &= \min_{q \leq |o_j|+1} t(o_j \oplus_q k) - t(o_j) \\ &= \min \left\{ \min_{q \leq |o_j|-1} (t(o_j^q, k) + t(k, o_j^{q+1}) \right. \\ &\quad \left. - t(o_j^q, o_j^{q+1})), t(p_j, k) + t(k, o_j^1) - t(p_j, o_j^1), \right. \\ &\quad \left. t(o_j^{|o_j|}, k) \right\} \\ &\leq \min_{q \leq |o_j|-1} \{t(o_j^q, k) + t(k, o_j^q), \\ &\quad t(k, o_j^{q+1}) + t(o_j^{q+1}, k)\}, \end{aligned} \quad (20)$$

where  $p_j$  is the initial location of vehicle  $j$  and  $o_j^q$  is the  $q$ th target on  $o_j$ . On the other hand, considering the travel time cost on duplicating the edge of the min-cost arborescence, the minimum travel time incurred by inserting the last target  $k$  into the arborescence for vehicle  $j$  is

$$\delta f_{da}^j = t(k, o_j^*) + \min_{q \leq |o_j|} t(o_j^q, k). \quad (21)$$

where  $q^* = \operatorname{argmin}_{q \leq |o_j|} t(o_j^q, k)$ . It then follows that  $\delta f_{EVM}^j \leq \delta f_{da}^j$ .

Combining (20), (21) and  $f_{EVM}^{j|\mathcal{T}_j|-1} \leq f_{da}^{j|\mathcal{T}_j|-1}$ , we get

$$\begin{aligned} f_{EVM}^{j|\mathcal{T}_j|} &= f_{EVM}^{j|\mathcal{T}_j|-1} + \delta f_{EVM}^j \\ &\leq f_{da}^{j|\mathcal{T}_j|-1} + \delta f_{da}^j. \end{aligned} \quad (22)$$

As  $f_{da}^j = f_{da}^{j|\mathcal{T}_j|-1} + \delta f_{da}^j$ , it holds that  $f_{EVM}^{j|\mathcal{T}_j|} \leq f_{da}^{j|\mathcal{T}_j|}$  for each vehicle  $j$ . Thus,  $\sum_{j \in \mathcal{R}} f_{EVM}^j \leq \sum_{j \in \mathcal{R}} f_{da}^j$ , which proves  $f_{EVM} \leq f_{da}$ . Combining with  $f_a \leq f_o$  in view of Lemma 3 and  $f_{EVM} \leq f_{da}$ , we have  $f_{EVM}/f_o \leq f_{da}/f_a$ . ■

Theorem 1 gives an upper bound of the worst case performance of EVM compared with an optimal solution with the asymmetric travel cost matrix, which extends the upper bound result in [15] for the problem with the symmetric travel cost matrix. Furthermore, based on Lemma 4, the upper bound  $f_{da}/f_a$  is 2 if the travel cost matrix is symmetric which is the same as in [15]. We now investigate the property of the marginal-cost-based clustering strategy (MC) in IV-A.3 which directly puts the target locations assigned to each vehicle into the sequence during their assignment. Let  $f_{MC}$  be the total travel time of a solution resulting from MC.

**Theorem 2:** The task assignment algorithm MC guarantees that the total travel time  $f_{MC} \leq f_{EVM}$ .

*Proof:* The proof is carried out by induction. The algorithm MC assigns the target locations according to (15), while EVM is based on (12). One can check that the first targets chosen by the two algorithms are the same. Thus,

$f_{MC}^1 \leq f_{EVM}^1$ , where the superscript 1 means the total travel time after assigning the first target.

Now suppose the first  $|\mathcal{T}| - 1$  targets assigned by the two algorithms are the same and  $f_{MC}^{|\mathcal{T}|-1} \leq f_{EVM}^{|\mathcal{T}|-1}$ . From the inequality in Lemma 2 and according to (15), for MC the marginal travel time incurred by inserting the last target  $k$  is

$$\begin{aligned} \delta f_{MC} &= \min_{j \in \mathcal{R}, q \leq |o_j|+1} t(o_j \oplus_q k) - t(o_j) \\ &\leq \min_{q \leq |o_{j^*}|+1} t(o_{j^*} \oplus_q k) - t(o_{j^*}), \end{aligned} \quad (23)$$

where  $j^* = \operatorname{argmin}_{i \in o_j, j \in \mathcal{R}} t(i, k)$  is determined by (12) and  $o_j$  contains those targets already assigned to vehicle  $j$ . On the other hand, EVM assigns the last target  $k$  as

$$\delta f_{EVM} = \min_{q \leq |o_{j^*}|+1} t(o_{j^*} \oplus_q k) - t(o_{j^*}). \quad (24)$$

where  $j^* = \operatorname{argmin}_{i \in o_j, j \in \mathcal{R}} t(i, k)$ . Thus,  $\delta f_{MC} \leq \delta f_{EVM}$ .

Combining (23), (24) and  $f_{MC}^{|\mathcal{T}|-1} \leq f_{EVM}^{|\mathcal{T}|-1}$ , we get

$$\begin{aligned} f_{MC} &= f_{MC}^{|\mathcal{T}|-1} + \delta f_{MC} \\ &\leq f_{EVM}^{|\mathcal{T}|-1} + \delta f_{EVM}. \end{aligned} \quad (25)$$

As  $f_{EVM} = f_{EVM}^{|\mathcal{T}|-1} + \delta f_{EVM}$ , it holds that  $f_{MC} \leq f_{EVM}$ . ■

The proposed clustering-based algorithms can be applied to the task assignment problem for time-varying drift fields by assigning the target locations to the vehicles based on a time-varying travel cost matrix which needs to get updated as the drift field changes. This obviously will affect the visiting sequence of target locations, and as a result, the performance of the solution constructed by the clustering-based algorithms at any given time can only be guaranteed for a limited time thereafter.

Now we have presented all the theoretical results of this paper, in the following section, we carry out simulation studies.

## V. SIMULATIONS

One can obtain four task assignment algorithms after integrating the target clustering strategies with the target-inserting metrics: integrating the Voronoi clustering strategy with the nearest principle (VN); integrating the Voronoi clustering strategy with the smallest marginal cost principle (VM); integrating the extended Voronoi clustering strategy with the nearest principle (EVN); and integrating the extended Voronoi clustering strategy with the smallest marginal cost principle (EVM). As the marginal-cost-based clustering strategy (MC) directly determines the targets' visiting sequence during their assignment, it is already a task assignment algorithm. Integrating with the proposed path planning method, the existing task assignment algorithms can be used to solve the task assignment problem. The proposed clustering-based algorithms are compared with a GA which is a popular heuristic algorithm for VRP [11]. The GA encodes each target as a numbered gene and inserts  $m - 1$  marker genes into the target genes. Then, each chromosome represents a candidate solution to the task assignment problem. The GA

TABLE I: The average solution quality  $q$  of the algorithms (A) on 400 scenarios for the task assignment problem under different instances (I) where  $n50m10$  means 10 vehicles need to visit 50 target locations.

$I \backslash A$	VN	VM	EVN	EVM	MC	GA
$n50m10$	1.8641	1.5099	1.6811	1.3222	1.1581	1.4626
$n100m10$	2.0078	1.5877	1.7956	1.3725	1.2077	1.7987
$n110m10$	2.0090	1.5770	1.7955	1.3730	1.2159	1.8968
$n120m10$	2.0180	1.5888	1.8059	1.3792	1.2264	1.9993
$n120m12$	2.0333	1.6067	1.7750	1.3662	1.2076	2.0869
$n120m14$	2.0481	1.6188	1.7499	1.3575	1.1918	2.1669
$n120m16$	2.0570	1.6318	1.7293	1.3468	1.1774	2.2309
$n120m18$	2.0607	1.6399	1.7127	1.3338	1.1660	2.2963
$n120m20$	2.0592	1.6418	1.7003	1.3276	1.1562	2.3777

employs the widely used tournament selection because of its efficiency and simplicity, which preserves gene diversity while guaranteeing all individuals might be selected [31].

Monte Carlo simulations are carried out to test the proposed algorithms, where all the experiments have been performed on an Intel Core *i5*–4590 CPU 3.30 GHz with 8 GB RAM, with algorithms compiled by Matlab under Windows 7. The solution quality of each algorithm is quantified by

$$q = \frac{f}{f_a}, \quad (26)$$

where  $f$  is the objective value in (2) and  $f_a$  is the sum of all the edge weights of an MCGA of the target-vehicle digraph  $\mathcal{G}$ . Since  $f_a \leq f_o$ , from Lemma 3 where  $f_o$  is the total travel time of an optimal solution, a value of the ratio  $q$  closer to 1 means a better performance of the solution.

The algorithms are tested on the task assignment problem for multiple vehicles with  $v = 1$  in a square drift field with edge length  $10^3\text{m}$  and  $\vec{v}_c = 10^{-3}[0.3x + 0.2y, -0.2x + 0.3y]^T$ . The number of chromosomes in the GA is empirically set as 120, and the crossover rate and mutation rate for the GA are 0.9 and 0.1. The GA terminates at the maximal iteration number 350. Several instances  $n50m10$ ,  $n100m10$ ,  $n110m10$ ,  $n120m10$ ,  $n120m12$ ,  $n120m14$ ,  $n120m16$ ,  $n120m18$  and  $n120m20$  are generated where  $n50m10$  means 10 vehicles need to visit 50 target locations. For each instance, 400 scenarios of the initial positions of the targets and vehicles are randomly generated. The average  $q$  of the algorithms on each instance is shown in Table I, and the corresponding average computation time for each algorithm is listed in Table II. Firstly, Table I shows VM performs better than VN, and EVM performs better than EVN. The four algorithms first cluster the target locations to the vehicles, and then employ the nearest principle or the smallest marginal cost principle to put the target locations assigned to each vehicle into sequence. In other words, the target locations have the same assignment for VM and VN, and for EVM and EVN. Thus, the better performances of VM over VN and EVM over EVN imply that the smallest marginal cost principle is more effective than the nearest principle to put the target locations into sequence. The reason lies partly in the fact that for

TABLE II: The corresponding average computation time ( $s$ ) for the algorithms (A) to get the solution to the task assignment problem under different instances (I).

$I \backslash A$	VN	VM	EVN	EVM	MC	GA
$n50m10$	0.0279	0.0281	0.0418	0.0423	0.0326	86.7761
$n100m10$	0.0500	0.0574	0.0984	0.1410	0.0590	140.6570
$n110m10$	0.0519	0.1130	0.0988	0.2904	0.0722	150.0892
$n120m10$	0.2381	0.2424	0.3665	0.4288	0.2493	163.2563
$n120m12$	0.2415	0.2440	0.4363	0.4434	0.2593	172.2250
$n120m14$	0.2480	0.2483	0.4453	0.4472	0.2616	176.7572
$n120m16$	0.2510	0.2541	0.4537	0.4566	0.2821	183.0387
$n120m18$	0.2563	0.2571	0.4655	0.4685	0.2913	197.1980
$n120m20$	0.2585	0.2597	0.4733	0.4744	0.3048	226.9434

each vehicle, the smallest marginal cost principle leads to the ordering of the target locations after computing the incurred travel cost at all possible positions on the vehicle's route; in contrast, the nearest principle is myopic in the sense that it inserts the target location with the minimal incurred cost at the end of the vehicle's route.

Secondly, Table I shows EVM performs better than VM, and EVN performs better than VN, which reflects the advantage of the extended Voronoi clustering strategy over the Voronoi clustering strategy. The reason is that the former strategy clusters a target using both the vehicles' initial locations and the clustered targets' locations, while the later only uses the vehicles' initial locations. Finally, MC performs better than EVM as shown in Table I, which verifies Theorem 2. The reason is that MC considers the overall incurred travel cost for all the vehicles when clustering a target, which makes use of the complete geographic information on the vehicles' locations and the clustered targets' locations. In addition, the  $q$  of VM, EVN, EVM and MC shown in Table I is stably below twice of the optimal for each instance, which displays the efficient and robust performances of the algorithms. However, the GA's performance deteriorates as the size of the problem increases as shown in the last column of Table I. Finally, in Table II the mean computation time of the proposed algorithms does not increase too much compared with the GA, which shows the proposed algorithms can scale with the problem size. The reason is that the clustering-based algorithms are heuristic in the sense that they assign the targets based on the geographic locations of the vehicles as well as the locations of those targets already clustered. The small computation time implies in particular the efficiency of the proposed algorithms when the solution's quality is improved by location-based clustering.

To further evaluate the solution quality  $q$  for the 400 scenarios of each instance, the Wilcoxon signed-rank test is carried out in a two-tail test with the 5% significance level for each pair of the algorithms. It is clear that the  $q$  of 400 scenarios on each instance differ significantly between the proposed algorithms ( $q$  from left to right corresponds to  $MC \rightarrow EVM \rightarrow VM \rightarrow EVN \rightarrow VN$ ). This implies the algorithms have an increasingly better performance as  $VN - EVN - VM - EVM - MC$ . The Wilcoxon signed-rank test shows that the performance of the GA deteriorates



as the problem size increases, which is better than VM and worse than EVM for  $n50m10$ , better than VN, worse than VM and the same as EVN for  $n100m10$ , better than VN and worse than EVN for  $n110m10$ , worse than EVN and the same as VN for  $n120m10$ , and worse than VN for the remaining instances. The results of the Wilcoxon signed-rank test are consistent with what is shown in Table I.

## VI. CONCLUSION

In this paper, we have investigated the task assignment problem in which multiple dispersed vehicles need to efficiently visit a set of target locations in a time-invariant drift field. A path planning method has been first designed which enables the vehicles to move between two prescribed locations in a drift field with the minimal time. The travel cost matrix resulting from the path planning method provides the route information for the target location assignment. In addition, several target clustering strategies and two target inserting principles have been proposed. The target clustering strategies assign the target locations to the vehicles based on the travel cost matrix while the target inserting principles put the target locations assigned to each vehicle into sequence. Integrating the clustering strategies and the target inserting principles, we have obtained several algorithms which can efficiently solve the target visiting problem. The paper's practical contributions are threefold: First, the deduced optimal navigation law provides a guiding principle for navigating vehicles in the presence of winds or currents. Second, the definition of the generalized arborescence helps to clarify for practitioners the space of feasible solutions to the task assignment problem. Third, the four clustering-based strategies offer a set of tools for different scenarios in logistic applications. The proposed algorithms will be extended in a drift field where some obstacles exist. We are also planning to test the algorithms using robot fish.

## REFERENCES

- [1] G. Binetti, D. Naso, and B. Turchiano, "Decentralized task allocation for surveillance systems with critical tasks," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1653–1664, 2013.
- [2] J. Yu, S.-J. Chung, and P. G. Voulgaris, "Target assignment in robotic networks: Distance optimality guarantees and hierarchical strategies," *Automatic Control, IEEE Transactions on*, vol. 60, no. 2, pp. 327–341, 2015.
- [3] S. L. Smith and F. Bullo, "Monotonic target assignment for robotic networks," *Automatic Control, IEEE Transactions on*, vol. 54, no. 9, pp. 2042–2057, 2009.
- [4] P. Toth and D. Vigo, *Vehicle routing: problems, methods, and applications*. Siam, 2014, vol. 18.
- [5] E. Lalla-Ruiz, C. Expósito-Izquierdo, S. Taheripour, and S. Voß, "An improved formulation for the multi-depot open vehicle routing problem," *OR spectrum*, vol. 38, no. 1, pp. 175–187, 2016.
- [6] C. Prins, "A simple and effective evolutionary algorithm for the vehicle routing problem," *Computers & Operations Research*, vol. 31, no. 12, pp. 1985–2002, 2004.
- [7] A. Bettinelli, A. Ceselli, and G. Righini, "A branch-and-cut-and-price algorithm for the multi-depot heterogeneous vehicle routing problem with time windows," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 5, pp. 723–740, 2011.
- [8] M. Mahmoudi and X. Zhou, "Finding optimal solutions for vehicle routing problem with pickup and delivery services with time windows: A dynamic programming approach based on state-space-time network representations," *Transportation Research Part B: Methodological*, vol. 89, pp. 19–42, 2016.
- [9] J. Renaud, G. Laporte, and F. F. Boctor, "A tabu search heuristic for the multi-depot vehicle routing problem," *Computers & Operations Research*, vol. 23, no. 3, pp. 229–235, 1996.
- [10] J. W. Escobar, R. Linfati, P. Toth, and M. G. Baldoquin, "A hybrid granular tabu search algorithm for the multi-depot vehicle routing problem," *Journal of Heuristics*, vol. 20, no. 5, pp. 483–509, 2014.
- [11] G. Laporte, "Fifty years of vehicle routing," *Transportation Science*, vol. 43, no. 4, pp. 408–416, 2009.
- [12] G. A. Korsah, A. Stentz, and M. B. Dias, "A comprehensive taxonomy for multi-robot task allocation," *The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1495–1512, 2013.
- [13] T. Shima, S. J. Rasmussen, A. G. Sparks, and K. M. Passino, "Multiple task assignments for cooperating uninhabited aerial vehicles using genetic algorithms," *Computers & Operations Research*, vol. 33, no. 11, pp. 3252–3269, 2006.
- [14] H.-L. Choi, L. Brunet, and J. P. How, "Consensus-based decentralized auctions for robust task allocation," *Robotics, IEEE Transactions on*, vol. 25, no. 4, pp. 912–926, 2009.
- [15] M. G. Lagoudakis, M. Berhault, S. Koenig, P. Keskinocak, and A. J. Kleywegt, "Simple auctions with performance guarantees for multi-robot task allocation," in *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 1. IEEE, 2004, pp. 698–705.
- [16] J. Han, J. Ok, and W. K. Chung, "An ethology-based hybrid control architecture for an autonomous underwater vehicle for performing multiple tasks," *Oceanic Engineering, IEEE Journal of*, vol. 38, no. 3, pp. 514–521, 2013.
- [17] D. Zhu, H. Huang, 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," *Cybernetics, IEEE Transactions on*, vol. 43, no. 2, pp. 504–514, 2013.
- [18] M. Eichhorn, "Optimal routing strategies for autonomous underwater vehicles in time-varying environment," *Robotics and Autonomous Systems*, vol. 67, pp. 33–43, 2015.
- [19] J. Delmerico, E. Mueggler, J. Nitsch, and D. Scaramuzza, "Active autonomous aerial exploration for ground robot path planning," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 664–671, 2017.
- [20] W. Yan, X. Bai, X. Peng, L. Zuo, and J. Dai, "The routing problem of autonomous underwater vehicles in ocean currents," in *OCEANS 2014-TAIPEI*. IEEE, 2014, pp. 1–6.
- [21] L. Zuo, J. Chen, W. Yan, and Y. Shi, "Time-optimal coverage control for multiple unicycles in a drift field," *Information Sciences*, 2016.
- [22] G. Gordon and E. McMahon, "A greedoid polynomial which distinguishes rooted arborescences," *Proceedings of the American Mathematical Society*, vol. 107, no. 2, pp. 287–298, 1989.
- [23] K. Fleszar, I. H. Osman, and K. S. Hindi, "A variable neighbourhood search algorithm for the open vehicle routing problem," *European Journal of Operational Research*, vol. 195, no. 3, pp. 803–809, 2009.
- [24] M. Soullignac, "Feasible and optimal path planning in strong current fields," *Robotics, IEEE Transactions on*, vol. 27, no. 1, pp. 89–98, 2011.
- [25] D. E. Kirk, *Optimal control theory: an introduction*. Courier Corporation, 2012.
- [26] D. D. Morrison, J. D. Riley, and J. F. Zancanaro, "Multiple shooting method for two-point boundary value problems," *Communications of the ACM*, vol. 5, no. 12, pp. 613–614, 1962.
- [27] D. L. Applegate, R. E. Bixby, V. Chvatal, and W. J. Cook, *The traveling salesman problem: a computational study*. Princeton university press, 2011.
- [28] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity*. Courier Corporation, 1982.
- [29] G. Gutin and A. P. Punnen, *The traveling salesman problem and its variations*. Springer Science & Business Media, 2006, vol. 12.
- [30] S. Rathinam, R. Sengupta, and S. Darbha, "A resource allocation algorithm for multivehicle systems with nonholonomic constraints," *IEEE Transactions on Automation Science and Engineering*, vol. 4, no. 1, pp. 98–104, 2007.
- [31] C. W. Ahn and R. S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," *IEEE transactions on evolutionary computation*, vol. 6, no. 6, pp. 566–579, 2002.