Received November 28, 2018, accepted December 6, 2018, date of publication December 11, 2018, date of current version January 7, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2886172

Wide-Area Vehicle-Drone Cooperative Sensing: Opportunities and Approaches

KAI PENG[1], WEIDONG LIU[1], QIANGUO SUN[1], XIAOQIANG MA[1], (Member, IEEE), MENGLAN HU[1], DESHENG WANG[1], (Member, IEEE), AND JIANGCHUAN LIU[2], (Fellow, IEEE)

[1]School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China
[2]College of Natural Resources and Environment, South China Agricultural University, Guangzhou 510642, China

Corresponding author: Menglan Hu (humenglan@hust.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 91538203, Grant 61502193, Grant 61871436, Grant 61872415, Grant 61671216, Grant 61702204, Grant 61872416, and Grant 61471408.

ABSTRACT The last decade has witnessed unmanned aerial vehicles (UAVs) emerging as a powerful platform for various industrial and civilian applications. However, refrained by limited battery capacities, the hovering time of UAVs is still limited, impeding them from achieving remote tasks, such as wide area inspection. To deal with such long-range applications, a common sense solution is to employ vehicles to transport, launch, and recycle UAVs. Efficient routing and scheduling for UAVs and vehicles can greatly reduce time consumption and financial expenses incurred in long-range inspection. Nevertheless, prior works in vehicle-assisted UAV inspection considered only one UAV, and was incapable of concurrently serving multiple targets distributed in an area. Leveraging multiple UAVs to serve multiple targets in parallel can significantly enhance efficiency and expand service areas. Therefore, in this paper, we propose a novel hybrid genetic algorithm (HGA), which supports the cooperation of one vehicle and multiple drones for wide area inspection applications. HGA allows multiple UAVs to launch and recycle in different locations, minimizing time wastage for both the vehicle and UAVs. Performance evaluation is presented to demonstrate the effectiveness and efficiency of our algorithm when compared with existing solutions.

INDEX TERMS Unmanned aerial vehicle, inspection, routing, scheduling.

I. INTRODUCTION

The enabling Internet-of-Things technology has encouraged many innovative sensing platforms and applications [1], [2]. One emerging yet powerful IoT sensing platform is the Unmanned Aerial Vehicle (UAV). UAV, which develops in the direction of unmanned attendance and intelligence, is light in weight, is small in size, is low cost, and is capable of operating autonomously. With these qualities, UAV has become one of the inevitable trends of the modern sensing applications, including sensing [3]–[5], forest fire spotting [6], pollution monitoring [7], navigation [8], [9], communication relays [10], [11], vehicle networks [12]–[14], cargo delivery [15]–[17], and so on.

These applications customarily require UAVs to visit many different locations in a wide area to collect sensing data. However, due to constrained battery capacities, the maximum service range of the UAVs is strictly limited. To cater for various remote sensing tasks in a wide area, a vehicle is often employed to serve as a carrier and energy supplier to enlarge the range of service provided [18], which is shown in Fig. 1. Although such cooperative paradigm is widely adopted in industry, it brings new challenges on the related path planning and scheduling issues in academia. The main problem to be addressed is how to design optimal routes for the UAVs and the vehicle to achieve the minimum total finish time.

The vehicle-drone cooperation problem has drawn significant attention in literature over recent years. Nevertheless, most of the existing work mainly focused on vehicle-drone cooperative parcel delivery, wherein both vehicles and UAVs are responsible of delivering parcels to customers [15]–[17]. These works cannot be directly applied to the vehicle-drone cooperative sensing problem, where the vehicle only serves as a carrier of UAVs while the UAVs ought to serve the customers. Only a few studies have focused on vehicle-drone cooperative sensing problem [4], [5], [19]–[21]. Most of the

FIGURE 1. Examples of UAV sensing applications.

above work consider only one UAV whereas we incorporate multiple UAVs in wide area inspection to promote the parallelism and efficiency. In addition, different from the above work, this paper proposes to adopt a hybrid genetic algorithm framework [22] to solve the problem. Fig. 2 depicts a scenario where multiple UAVs collaborate with a carrier to serve the customers.

In this case, this paper investigates a novel and challenging problem, referred to as joint routing and scheduling problem for wide-area vehicle-drone cooperative inspection. The routing model is similar to Two-Echelon Vehicle Routing Problems (2E-VRP) [23], which customarily minimized route lengths. However, 2E-VRP neglected scheduling issues in our problem. With both routing and scheduling issues involved, our problem is much more complicated than 2E-VRP. First, we have to determine proper locations where the vehicle stops and launches the UAVs. Such locations are referred to as "parking spots" hereinafter. The selection of parking spots can be formalized as a Facility Location Problem (FLP) [24]. Second, we should carefully plan paths for both the vehicle and multiple drones. Planning routes for UAVs and the carrier can be abstracted as a Vehicle Routing Problem (VRP) [25]. Third, we need to balance the scheduling of tours to multiple drones. Assigning routes for UAVs, in fact, is a Bin Packing Problem (BPP) [26], [27]. Accordingly, the problem investigated in this paper is a combinatorial optimization problem, where each sub-problem is NP-Hard.

By and large, we aim to achieve an overall optimization on parking spot selection, path planning, and tour assignment such that the total finish time is minimized. Nevertheless, these aspects are tightly-coupled as the performance of each aspect directly depends on the others. Therefore, an efficient joint design on parking spot selection, routing, and scheduling design is highly desirable. Notice that 2E-VRP is a generalization of VRP while VRP is a generalization of Traveling Salesman Problem (TSP) [28], [29]. In other words, the problem studied in this paper is much more difficult than 2E-VRP, which is much harder than VRP and TSP. Therefore, the problem investigated is quite challenging.

To this end, we propose a novel Hybrid Genetic Algorithm (HGA), which consists of a series of algorithms. The proposed algorithm is well-designed such that it avoids falling into the local optimal solutions. In addition, it perfectly balances the complexity of the algorithm and the performance. Our main contributions are multi-fold. First, we propose a Minimum Visit Cost Crossover (MVCC) algorithm that selects gene fragments based on the visiting cost of each parking spot. Second, we include a procedure called Population Management in HGA, which ensures a better distribution in population to avoid premature convergence. Third, we design a three-hierarchical search algorithm to improve the quality of offsprings and raise the possibility of retrieving a better solution. Experiment results have validated the remarkable performance of the HGA.

The rest of this paper is organized as follows: Section II discusses related work; Section III states problem formulation and notations; Section IV presents the overview of the proposed algorithm, and Section V discusses each step of the algorithm in detail; Section VI presents a performance evaluation, and section VII concludes this paper.

II. RELATED WORK

A significant amount of existing research on UAV-based service systems have been reported in the literature. Motlagh *et al.* [10] surveyed low-altitude UAV-based service systems in a wide area, with an emphasis on the communication between a fleet of UAVs. According to [3] and [6], it comes to the conclusion that UAVs have gained remarkable performance in various monitoring and sensing applications and have earned considerable amount of attention in recent years.

The routing problem for vehicle-drone cooperative sensing problem is quite similar to the 2E-VRP [23] problem. 2E-VRP aims to obtain a set of primary and secondary routes in a two-echelon distribution system, in order to satisfy the demands of all customers and reduce the system cost. A few recent studies investigate the 2E-VRP problem. Luo *et al.* [20] proposed two heuristics to solve the problem: the first heuristic builds an entire tour for all customers and splits it by routes; the second constructs the ground carrier tour and assigns UAV flights to the tour. Savuran and Karakaya [19] proposed a route optimization method for a carrier-launched UAV system based on the Genetic Algorithm [30]. Manyam *et al.* [21] proposed a mixed integer linear programming formulation and presented a branch-and-cut method to solve the routing problem. Notice that in the above work only one UAV is taken into account in all the existing work, which suffers from inefficiency as the number of targets increases.

Only few papers [4], [5] consider utilizing multiple UAVs to support wide-area inspection. Hu *et al.* [4] proposed to utilize a vehicle carrying multiple UAVs to perform sensing tasks over a target area. Another work [5] also studied a vehicle-assisted UAV inspection problem. In [5], multiple UAVs are allowed to be launched and recycled from the

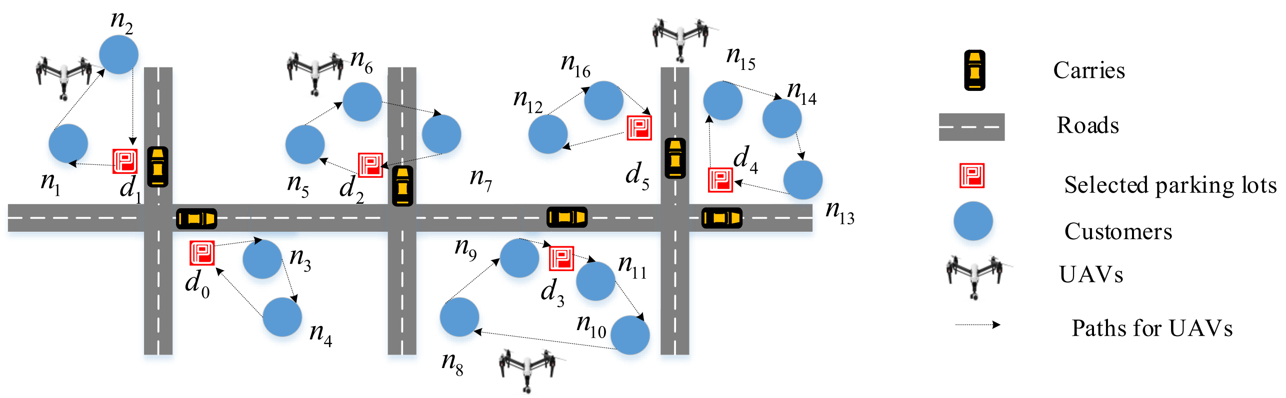


FIGURE 2. Illustration of the scenario where UAVs serve the customers with a carrier.

vehicle in different locations, minimizing time wastage for both the vehicle and UAVs. In this paper, we also employ multiple UAVs such that the customers shall be served simultaneously, thereby significantly enhancing inspection efficiency. In addition, different from all of the above work, this paper proposes to adopt a novel hybrid genetic algorithm framework to solve the problem.

In recent years, UAV-based parcel delivery has attracted considerable research attention. Murray and Chu [15] studied a so-called Flying Sidekick Traveling Salesman Problem (FSTSP) for drone-assisted parcel delivery problem and presented a series of simple heuristics. Ferrandez *et al.* [16] presented an algorithm that first finds the optimal launch locations using the K-means clustering algorithm [31] and then decides the carrier route based on the Genetic Algorithm. Campbell *et al.* [32] formed and optimized the models for drone delivery in collaboration with a truck. Another work [17] derived a number of worst-case results on VRP with drones, i.e., the maximum savings that can be obtained from using drones. It may be noticed that all the studies are concentrated on the routing for vehicle-drone cooperative parcel delivery problem, where both vehicles and drones visit customers to deliver parcels. Such work can't be directly applied for to the vehicle-drone cooperative sensing problem investigated in this paper, where drones visit customers for data collection while the vehicle only serves as a mobile platform for drone launching/recycling.

The design of the algorithm proposed in this paper is partly motivated by [22], which proposed a hybrid genetic algorithm framework for solving VRP. Notice that VRP is a sub-problem of our problem. The intuition behind [22] is that good solutions may be derived by crossover from parent solutions. Olivera and Viera [33] addressed a Vehicle Routing Problem with Multiple Trips (VRPMT), which is also a sub-problem of our problem. Cheikh *et al.* [34] proposed a mutable neighborhood search algorithm for VRPMT, through which encouraging solutions are obtained. They considered a scenario where one carrier is allowed to perform multiple trips while the duration of routes assigned to the same carrier is finite. However, the algorithm proposed in our paper not only applies HGA as its framework, but also adopts a set of local search operations for parking spot selection, route planning, and route assignment.

III. PROBLEM FORMULATION

In this paper, the joint routing and scheduling problem for vehicle-drone cooperative inspection is investigated. Due to the limited battery capacity, a single UAV can only visit customers within a small range during one flight. Accordingly, a carrier vehicle is employed to transport, launch, and recycle the UAVs for the sake of expanding the service area. It is assumed that there lies a road network in the target area. Several parking spots on the road are selected and sequentially visited by a carrier with multiple UAVs for wide-area inspection. At each selected parking spot, UAVs are launched to visit the nearby customers according to the planned routes. Each UAV may perform multiple routes at one parking spot since its battery can be replaced on the carrier. When the UAVs finish their tasks, the carrier shall leave for the next selected parking spot. The process is repeated till all the customers have been served in the target area.

The problem can be decomposed: First, the selection of parking spots can be formalized as a Facility Location Problem (FLP). Also, planning routes for UAVs and the carrier can be abstracted as a VRP. Third, assigning routes for UAVs, in fact, is a Bin Packing Problem (BPP). Accordingly, the problem investigated in this paper is a combinatorial optimization problem, where each sub-problem is NP-Hard. We formulate the sub-problems as follows.

For simplicity, the following assumptions are made:

- The coordinates of each customer are known to us.
- The road network lying in the target region is modeled as a connected graph in our paper.
- Both the carrier and the UAVs travel at constant speeds, which are denoted as $s_{carrier}$ and s_{uav}, respectively.
- The time taken by the UAV to serve a customer is negligible.

- There are sufficient UAV batteries on the carrier and the time taken by replacing a UAV's battery is negligible.
- Constrained by limited battery capacities, the hovering distance of each UAV is finite, which is denoted as d hereinafter.

Let $G = (V, E)$ be a graph where V is the vertex set and E is the edge set. V is divided into two subsets: N and D. Let $N = \{n_1, n_2, \ldots, n_{N_c}\}$ be the set of customers that need to be served. Let $D = \{d_0, d_1, d_2, \ldots, d_{D_c}\}$ represent the set of candidate parking spots where d_0 is distinguished as the base where the carrier leaves at the beginning. A cost matrix $A = (a_{v_1 v_2})$ is defined on E, representing the distance between the vertex v_1 and vertex v_2. A carrier carrying m identical UAVs, denoted as $U = \{u_1, u_2, \ldots, u_m\}$, leaves from the base to serve the customers.

First, some parking spots need to be selected from the set of candidate parking spots (i.e., D) to form the carrier's route. In real applications, we usually take samples along the road network which lies in the region to serve as the candidate parking spots. For each candidate parking spot, let $x(d_j)$ denote whether the candidate parking spot d_j is selected ($x(d_j) = 1$) or not ($x(d_j) = 0$). Many aspects need to be taken into account jointly when selecting proper parking spots, such as the distribution of customers around each candidate parking spot, the distance to the customers, etc. Apparently, such problem can be categorized as a FLP.

Second, for each selected parking spot, the routes for each UAV need to be determined. Each UAV route contains a parking spot and at least one customer, which is denoted as $r_i = \{d_x, n_y, \ldots\}$ where $d_x \in D$ and $n_y \in N$. Correspondingly, the total length of a UAV route r_i is defined as follows:

$$l(r_i) = a_{r_i^1 r_i^{|r_i|}} \sum_{x=1}^{|r_i|-1} a_{r_i^x r_i^{x+1}} \tag{1}$$

where r_i^x denotes the $x-th$ element of r_i and $|r_i|$ represents the size of r_i. To build the routes for each UAV at a selected parking spot d_j, three problems need to solved. The first issue is to select the customers that are to be served at d_j. Such operation is referred to as assigning customers to d_j hereinafter. The second issue is to build the feasible UAV routes that cover all the customers assigned to d_j. For each UAV route r_i, let $x(r_i, d_j)$ represent whether the route r_i is supposed to be finished at the parking spot d_j ($x(r_i, d_j) = 1$) or not ($x(r_i, d_j) = 0$). Last but not least, reasonably schedule the UAVs to finish the routes. Clearly, it can be formalized as a Vehicle Routing Problem with Multiple Trips (VRPMT), which is a variant of classical VRP. In this paper, we use $y(r_i, u_k)$ to denote whether the route r_i is assigned to the UAV u_k ($y(r_i, u_k) = 1$) or not ($y(r_i, u_k) = 0$). Let $R = \{r_1, r_2, \ldots, r_{n_r}\}$ be the set of routes in a solution. The total distance that UAV u_j travels at the parking spot d_j is defined as follows:

$$l(u_k, d_j) = \sum_{r_i \in R} x(r_i, d_j) y(r_i, u_k) l(r_i) \tag{2}$$

Third, the sequence to visit each selected parking spot need to be determined. Although we have obtained the selected parking spots and the UAV routes at each parking spot, the sequence for the carrier to visit them may have great influence on the total time consumption. Notice that the starting point of the carrier's route shall always be the base d_0. Indubitably, it is a classical Traveling Salesman Problem (TSP). Let $r_0 = \{d_0, d_j, \ldots\}$ denote the route for the carrier. Correspondingly, the total length of the carrier's route is defined as follows:

$$l_{carrier} = a_{r_0^1 r_0^{|r_0|}} + \sum_{x=1}^{|r_0|-1} a_{r_0^x r_0^{x+1}} \tag{3}$$

where r_0^x denotes the $x-th$ element of r_0 and $|r_0|$ represents the size of the r_0.

The goal pursued in our paper is to minimize the total time consumption, denoted as $C(s)$, while satisfying several requirements:

$$\min\ C(s) = \frac{l_{carrier}}{s_{carrier}} + \frac{\sum_{d_j \in D} x(d_j) \max(l(u_k, d_j))}{s_{uav}} \tag{4}$$

$$\text{s.t. } m \geq 1 \tag{5}$$

$$0 \leq l(r_i) \leq d \quad \forall r_i \in R \tag{6}$$

$$\sum_{r_i \in R} |r_i| - 1 = N_c \tag{7}$$

Constraint 6 states that at least one UAV is employed to collaborate with the carrier. Constraint 7 guarantees that the length of a route cannot exceed the limit of hovering distance of a UAV. Constraint (7) ensures that all customers are served. Finally, we list the notation and terminology in Table 1.

TABLE 1. Notation and terminology.

Notation	Definition
N	a set of customers to be served
D	a set of candidate parking spots
U	a set of UAVs employed
n_i	a customer
d_j	a parking spot
u_k	a UAV
d	the maximum single flight distance of a UAV
r_i	a UAV route
$s_{carrier}$	the speed of the carrier
s_{uav}	the speed of each UAV
M	the population
R_{d_j}	a set containing all the UAV routes at the parking spot d_j
$l(r_i)$	the length of the UAV route r_i
$C(s)$	the cost of the solution s

IV. THE OVERVIEW OF HGA

In this paper, a novel algorithm named improved Hybrid Genetic Algorithm (HGA) is proposed. As indicated by its name, the design of HGA is partially motivated by the Genetic

Algorithm (GA). It is based on the idea that a better solution can be obtained by combining the "valuable" parts of other solutions.

A. THE STRUCTURE OF HGA

The population M containing multiple feasible solutions is maintained by HGA. In general, HGA works by iteratively selecting two parent solutions from M and retrieving a better child solution by a crossover operation. At every iteration, HGA executes four steps to generate a feasible solution. First, it selects two parent solutions from M using Binary Tournament algorithm. It then constructs an offspring solution utilizing a so-called MVCC algorithm, which shall be specified hereinafter. Next, it optimizes the offspring solution with the help of Local Search algorithm. Finally, update the population M. We present its pseudo codes in Algorithm 1 while the flowchart of HGA is shown in Fig. 3.

B. SOLUTION REPRESENTATION

To better present our proposed algorithm, the representation of a feasible solution is introduced in this section. A feasible solution S contains two parts: $r_0 = \{d_0, d_1, \ldots\}$ and $R = \{R_{d_0}, R_{d_1}, \ldots\}$ where r_0 stands for the route for the carrier and R stands for the set of UAV routes, corresponding to each selected parking spot in r_0. Let $R_{d_i} = \{R_{d_i}^{u_1}, \ldots, R_{d_i}^{u_m}\}$ represent the set of UAV routes at the parking spot d_i where $R_{d_i}^{u_k}$ denotes the set of UAV routes for the UAV u_k at d_i. Each UAV route is denoted as $r_i = \{d_x, n_y, \ldots\}$ where $d_x \in D$ and $n_y \in N$. Notice that at most one parking spot and at least one customer are included in each UAV route.

V. THE DESIGN OF HGA

A. POPULATION INITIALIZATION

HGA first initializes the population M with a *population_initialization* procedure, as shown in Algorithm 2. The procedure constructs several feasible solutions in a greedy manner and adds them into the population M. To generate a solution, the procedure executes three steps. First, it selects qualified parking spots using *parking_spot_selection* procedure. Then, it constructs feasible UAV routes for each selected parking spot via Sweep algorithm. Finally, it schedules the UAVs to perform the tasks via *UAV_scheduling* procedure.

Algorithm 1 Overview of HGA

Initialize the population M
for $i = 0 \rightarrow IT_{max}$ **do**
  (a) Binary Tournament algorithm is applied to select two parent P_1 and P_2 from the population M
  (b) MVCC algorithm is used to get an offspring C
  (c) Educate offspring C using the Local Search algorithm to obtain the optimized solution S
  (d) Update the population M with Population Management procedure
end for
Return the solution S with the minimum $C(s)$ in M

Algorithm 2 *population_initialization*

$M \leftarrow \emptyset$
for $i = 0 \rightarrow IT_{size}$ **do**
  $r_0 \leftarrow$ call *parking_spot_selection* procedure
  $R \leftarrow$ construct UAV routes for each selected parking spot via Sweep algorithm
  $S \leftarrow$ call *UAV_scheduling* procedure
  add S to the population M
end for
Return M

1) PARKING SPOT SELECTION

The *parking_spot_selection* procedure is shown in Algorithm 3, which selects the parking spots greedily. For each customer n_i, if the distance from it to a parking spot d_j is less than d, we say that d_j is covered by n_i and vice versa. The procedure first selects the customer with the minimum μ_{n_i} value, which is defined as follows:

$$\mu_{n_i} = \frac{|CD_{n_i}|}{|D|} \tag{8}$$

where CD_{n_i} denotes the set of candidate parking spots covered by n_i. Then it probabilistically selects the parking spot

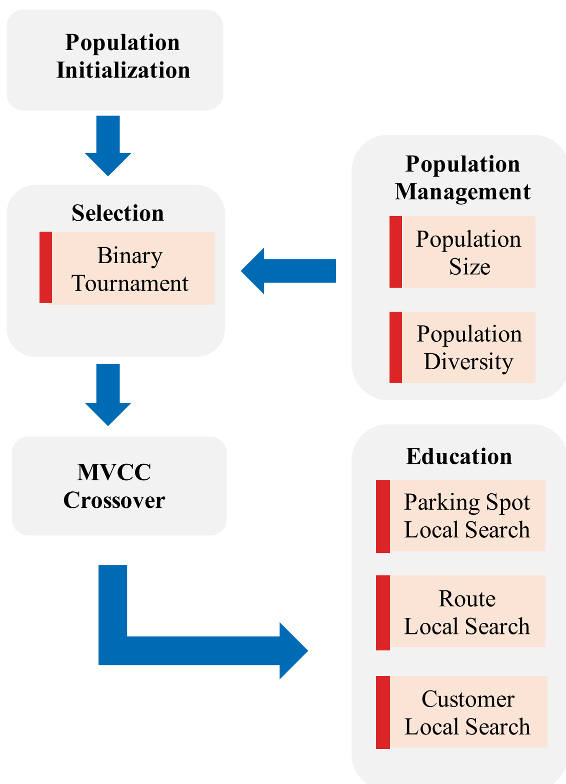


FIGURE 3. Flowchart of HGA.

Algorithm 3 *parking_spot_selection*

$r_0 \leftarrow \{d_0\}$
$N_{temp} \leftarrow N$
while $N_{temp} \neq \varnothing$ **do**
  select n_i from N_{temp} with the minimum μ_{n_i}
  select d_j from CD_{n_i} based on $p(d_j)$
  if $d_j \notin r_0$ **then**
    $r_0 \leftarrow r_0 \cup \{d_j\}$
  end if
  $N_{temp} \leftarrow N_{temp} \setminus \{n_i\}$
end while
run 2-opt algorithm on r_0
Return r_0

from CD_{n_i} with the probability function given as follows:

$$p(d_j) = \frac{|CN_{d_j}|}{N} \tag{9}$$

where CN_{d_j} represents the set of customers covered by d_j. At last, it determines the sequence of visiting the selected parking spots via 2-opt algorithm, which is a simple yet efficient algorithm for solving TSP.

2) UAV ROUTES CONSTRUCTION

For each customer, we assign it to its nearest selected parking spots in r_0. Then Sweep algorithm is utilized to construct feasible UAV routes.

3) UAV SCHEDULING

UAV_scheduling procedure is used to reasonably assign the routes to the UAVs such that the time consumption is minimized, as shown in Algorithm 4. For each selected parking spot d_j, the procedure iterates to assign the routes to the UAVs evenly. At every iteration, we select r_i with the greatest length among all unassigned routes in R_{d_j}. The route r_i is assigned to the UAV u_j with the least amount of tasks.

Algorithm 4 *UAV_scheduling*

for $d_j \in r_0$ **do**
  while $R_{d_j} \neq \varnothing$ **do**
    select r_i with the maximum $l(r_i)$ from R_{d_j};
    select u_k with the minimum $\sum_{r_i \in R^{u_k}_{d_j}} l(r_i)$ from U;
    set $y(r_i, u_k)$ to be 1;
    $R^{u_k}_{d_j} = R^{u_k}_{d_j} \cup \{r_i\}$
    $R_{d_j} = R_{d_j} \setminus \{r_i\}$
  end while
end for

B. PARENT SELECTION AND CROSSOVER

Since the population has been initialized, HGA then iteratively selects two solutions P_1 and P_2 as the parents to produce a child solution. In our paper, Binary Tournament algorithm is adopted to select two solutions from the population, since it is rather simple yet efficient.

After the parents are selected, a Minimum Visit Cost Crossover (MVCC) algorithm is proposed to construct a child solution with the qualified gene fragments from the parents, as shown in Algorithm 5. MVCC tends to give high priority to those parking spots with the minimum cost, which are defined as follows:

$$\theta_{d_j} = \alpha \frac{\sum_{r_i \in R_{d_j}} l(r_i)}{N_{d_j}} + \beta \frac{1}{N_{d_j}} \tag{10}$$

where N_{d_j} represents the number of customers assigned to the parking spot d_j. The first part of θ_{d_j} represents the average distance between the d_j and its assigned customers while the second part is the reciprocal of N_{d_j}. Accordingly, higher probability of being selected is assigned to the parking spot which owns more nearby customers.

Algorithm 5 MVCC

$r^C_0 \leftarrow \varnothing$
$R^C \leftarrow \varnothing$
$N_{served} \leftarrow \varnothing$
$\theta_{median} \leftarrow Median\{\theta^{P2}_{d_0}, \theta^{P2}_{d_1}, \ldots\}$
for $d^{P1}_i \in r^{P1}_0$ **do**
  if $\theta^{P1}_{d_i} < \theta_{median}$ **then**
    $r^C_0 \leftarrow r^C_0\ cup\{d^{P1}_i\}$
    $R^C \leftarrow R^C \cup R_{d_i}$
    add the customers of R_{d_i} into N_{served}
  end if
end for
for $d^{P2}_i \in r^{P2}_0$ **do**
  if $R^{P2}_{d_i}$ contains unserved customers **then**
    $r^C_0 \leftarrow r^C_0 \cup \{d^{P2}_i\}$
    Delete the customers in N_{served} from $R^{P2}_{d_i}$
    $R^C \leftarrow R^C \cup R^{P2}_{d_i}$
    add the customers of $R^{P2}_{d_i}$ into N_{served}
  end if
end for
$C \leftarrow \{r^C_0, R^C\}$
Return C

Since MVCC involves more than one solution, a superscript is added on each notation so as to distinguish them, as shown in Algorithm 5. For instance, r^{P1}_0 represents the route for the carrier in the parent solution $P1$. MVCC first calculates the median of the parking spot cost of the parent solution $P2$, which is denoted as θ_{median}. Then it searches the parent solution $P1$ to discover the parking spot whose cost is less than θ_{median}, such that the obtained child solution is improved. At last, it scans through the parent solution $P2$ to include those unserved customers in the child solution. Notice that N_{served} containing those customers who have been served is maintained by MVCC to avoid duplicate customers in child solution. An example of MVCC is shown in Fig. 4.

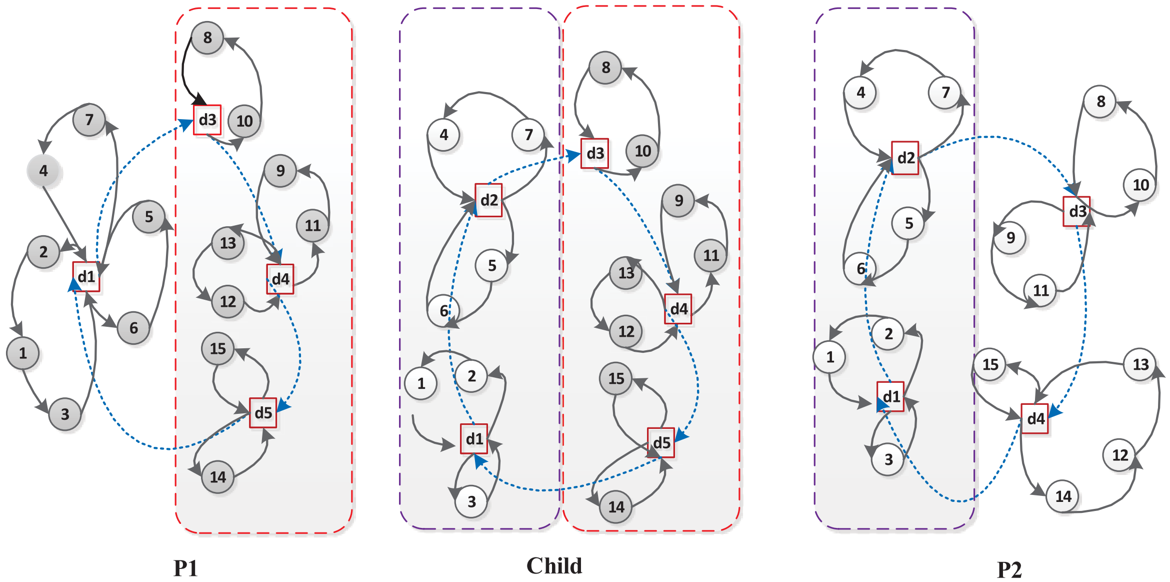


FIGURE 4. Illustration of MVCC.

C. EDUCATION

The dominant gene fragments of the parent solutions P_1 and P_2 can be passed on to the offspring C via MVCC. For further optimization, the Local Search algorithm is adopted to improve the offspring solution. It contains three procedures: *parking_spot_local_search* procedure, *route_local_search* procedure and *customer_local_search* procedure, which aim to optimize the selection of parking spots, merge the short routes and optimize the access order of customers within each UAV route, respectively.

1) PARKING SPOT LOCAL SEARCH

The *parking_spot_local_search* procedure adopts two kinds of operations on each selected parking spot: **remove** and **replace**, so as to optimize the solution. As shown in Algorithm 6, the procedure first removes the selected parking spot with few assigned routes and reallocated the routes to its neighboring parking spots. Given a parking spot, its neighboring parking spots $Neigh(d_j)$ is defined as follows:

$$Neigh(d_j) = \{d_x | a_{d_x d_j} < d\} \tag{11}$$

The procedure then replaces each selected parking spot with its neighboring parking spots, trying to find a solution with less cost. Notice that M_{num} is a tunable parameter that need to be determined after conducting a large number of experiments.

2) ROUTE LOCAL SEARCH

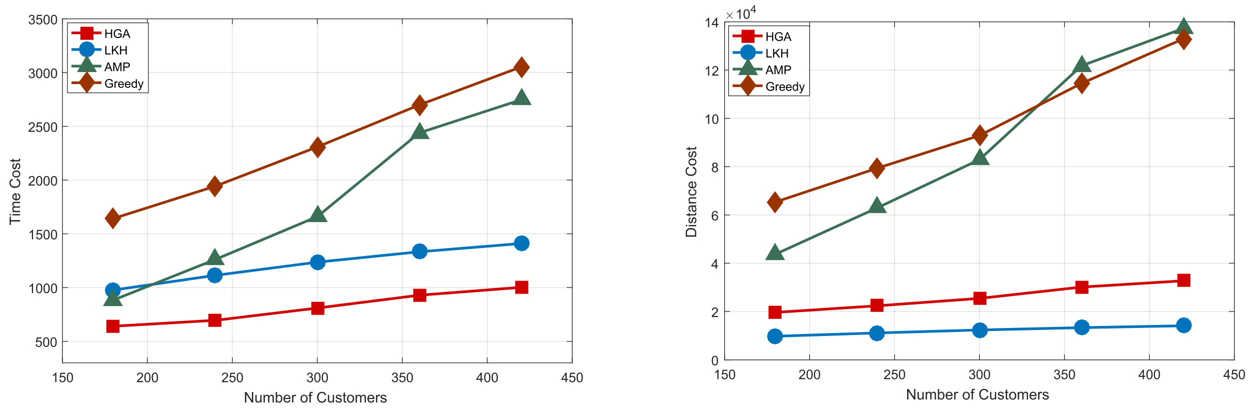
The main purpose of *route_local_search* procedure is to merge the routes which are too short. As shown in Algorithm 7, the procedure first picks out all the short routes, namely the routes whose length is less than *DIS*. *DIS* is also a tunable parameter. Then it tries to merge the short route with

Algorithm 6 *parking_spot_local_search*

for $d_j \in r_0$ **do**
  if $|R_{d_j}| \le M_{num}$ **then**
    reassign the routes in R_{d_j} to $Neigh(d_j)$
  end if
end for
for $d_j \in r_0$ **do**
  for $d_i \in Neigh(d_j)$ **do**
    Replace d_i with d_j
    if $C(s_{d_i}) > C(s_{d_j})$ **then**
      $r_0 \leftarrow r_0 \setminus \{d_i\}$
      $r_0 \leftarrow r_0 \cup \{d_j\}$
    end if
  end for
end for

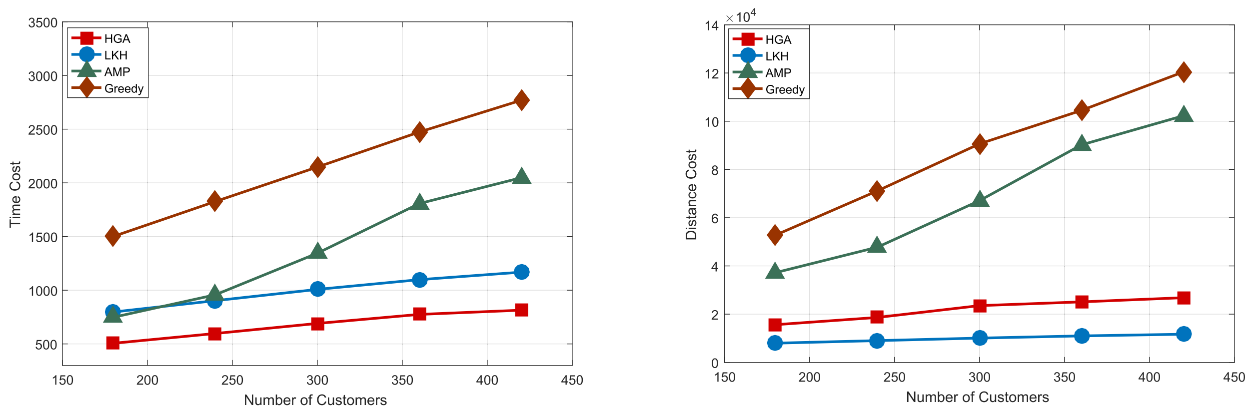
Algorithm 7 *route_local_search*

$R_{short} \leftarrow \emptyset$
for $r_i \in R$ **do**
  if $l(r_i) \le DIS$ **then**
    $R_{short} = R_{short} \cup \{r_i\}$
  end if
end for
for $r_i \in R_{short}$ **do**
  for $r_j \in Neigh(r_i)$ **do**
    $r_{new} \leftarrow$ merge route r_i and r_j
    if $l(r_{new}) < d$ **then**
      $R_{short} = R_{short} \setminus \{r_i\}$
    end if
  end for
end for



(a) (b)

FIGURE 5. Results with varying number of customers (changing density). (a) Time cost. (b) Distance cost.



(a) Time cost (b) Distance cost

FIGURE 6. Results with varying number of customers (fixed density). (a) Time cost. (b) Distance cost.

its neighboring routes. Given a route, its neighboring routes $Neigh(r_i)$ is defined as follows:

$$Neigh(r_i) = \{r_x \,|\, \max(a_{n_i n_x}) < d, n_i \in r_i, n_x \in r_x\} \quad (12)$$

It may occur that there are no suitable routes to merge with in $Neigh(r_i)$. If this is the case, the procedure simply ignores such route.

3) CUSTOMER LOCAL SEARCH

The *customer_local_search* procedure is used to optimize the access order of customers within each UAV route. Two kinds of operations are conducted upon each UAV route r_i: **Exchange** and **Replace**. The **Exchange** operation exchanges the order between two customers of r_i. The **Replace** operation exchanges one customer of r_i with its neighboring routes. If either one of the operation improves the current solution, such operation is confirmed.

D. POPULATION MANAGEMENT

Every time we intend to add a solution into the population M, Population Management procedure is invoked to manage the population in two aspects: population diversity and population size.

1) POPULATION DIVERSITY MANAGEMENT

To avoid useless computation, any two solutions in the population M should be distinct. Thus, a strict condition for the population M is proposed:

$$\forall P_1, P_2 \in M, P_1 \neq P_2, \mid C(P_1) - C(P_2) \mid > \Delta \quad (13)$$

where P_1, P_2 are two solutions of the population M and $C(P_1)$, $C(P_2)$ are the cost of P_1 and P_2, respectively. Therefore, every time a solution s_i is added into the population M, we compare s_i with each solution s_j in M by calculating $C(s_i) - C(s_j)$. If it is greater than Δ, it means s_i and s_j are well spaced. Otherwise, it is necessary to compare them via each selected parking spot and its assigned routes.

2) POPULATION SIZE MANAGEMENT

To speed up the calculation, the size of M is set as M_{size}. If the size of M is greater than M_{size} after inserting the new solutions, we remove $|M| - M_{size}$ solutions with the greatest $C(s)$ values from the memory to avoid overflow.

VI. COMPUTATIONAL EXPERIMENTS

A significant amount of experiments have been conducted to assess the performance of HGA. To evaluate the efficiency of HGA, we compare it with three baseline algorithms: a greedy algorithm (Greedy), AMP (a relaxed version of AMP [33]), and the Lin-Kernighan Heuristic (LKH) [35]. LKH is an effective heuristic algorithm for solving classical TSP. We adapt LKH to our problem by assuming that the carrier serves all the customers, without the employment of UAVs. Greedy, a heuristic for solving VRP, first plans the route for UAVs and then assigns the routes to its nearest parking spot. AMP is an efficient algorithm for solving VRPMT. It is assumed that the UAVs are allowed to travel unlimited distance when applying AMP to our problem.

Two performance metrics are of particular interest to us. The first is the time cost of a solution. It is defined by the total amount of time taken by the carrier to leave from the base, launch UAVs to serve all the customers, and return to the base. Indubitably, less value of the time consumption means higher efficiency. The second metric is the distance cost of a solution, including the distance that the carrier travels and that of the UAVs travel. Likewise, less distance cost indicates a less costly system.

In our configuration, N_c customers are randomly generated within a l units $\times$ l units region, with 1 unit representing 1 kilometer in reality. A carrier with 5 identical UAVs are employed to serve the customers in the target region. Each UAV is capable of traveling 1 unit per flight. The speed of the carrier and each UAV are set as $5m/s$ and $3m/s$, respectively. First, we fix l to be 100 and only vary N_c in the range $[150, 450]$. With N_c increasing, the density of customers in the target region increases as well. The corresponding results are represented graphically in Fig. 5. Second, we not only vary N_c in the range $[150, 450]$, but also range l from 100 to 500 such that the density of the customers is fixed. The results are presented in Fig. 6.

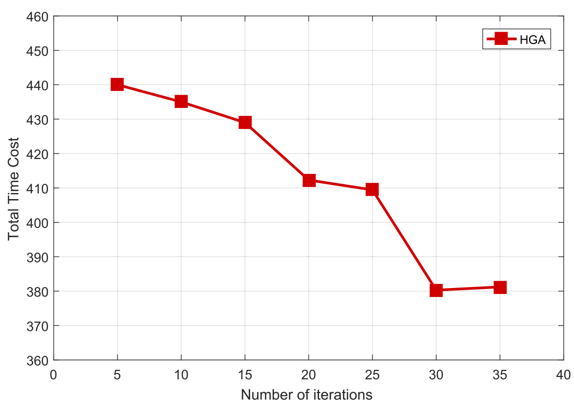


FIGURE 7. Time cost with varying number of iterations.

Details shown in Figs. 5 - 6 present the efficiency of our proposed algorithm HGA. It can be observed from Figs. 5(a) - 6(a) that HGA outperforms any other algorithms significantly, in terms of the time consumption. For both case, HGA requires least amount of time and reasonable amount of traveling distance. When applied to our problem, LKH performs impressively well. It gains best performance with respect to the distance cost.

Lastly, we study the impact of the number of iterations on the performance of HGA. We fix N_c to be 300 and l to be 200 and vary the number of iterations. The corresponding result in Fig. 7 shows that as the number of iterations grows, the results converge and we finally obtain superior results.

VII. CONCLUSIONS

This paper has investigated a combinatorial optimization problem and proposed a novel improved Hybrid Genetic Algorithm (HGA) to solve it. HGA is able to achieve an excellent trade-off between the complexity of the algorithm and the performance. In addition, we have proposed a Minimum Visit Cost Crossover (MVCC) algorithm to generate an offspring solution with less cost. Population Management procedure is utilized to manage the population, such that a better solution is more likely to be constructed. Furthermore, we have designed a three-hierarchical search algorithm to improve the solution. At last, the results of a significant amount of experiments we conducted have validated the efficiency and effectiveness of HGA.

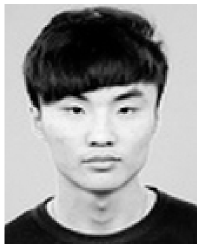
REFERENCES

[1] M. Chen, Y. Miao, Y. Hao, and K. Hwang, "Narrow band Internet of Things," *IEEE Access*, vol. 5, pp. 20557–20577, 2017.
[2] H. Zhou, X. Wang, X. Chen, X. Li, and S. Xu, "Data offloading techniques through vehicular ad hoc networks: A survey," *IEEE Access*, vol. 6, pp. 65250–65259, 2018.
[3] I. Colomina and P. Molina, "Unmanned aerial systems for photogrammetry and remote sensing: A review," *ISPRS J. Photogram. Remote Sens.*, vol. 92, pp. 79–97, Jun. 2014.
[4] M. Hu *et al.*, "Joint routing and scheduling for vehicle-assisted multi-drone surveillance," *IEEE Internet Things J.*, to be published, doi: 10.1109/JIOT.2018.2878602.
[5] M. Hu *et al.*, "On the joint design of routing and scheduling for vehicle-assisted multi-UAV inspection," *Future Gener. Comput. Syst.*, vol. 94, pp. 214–223, May 2018.
[6] C. Yuan, Y. Zhang, and Z. Liu, "A survey on technologies for automatic forest fire monitoring, detection, and fighting using unmanned aerial vehicles and remote sensing techniques," *Can. J. Forest Res.*, vol. 45, no. 7, pp. 783–792, 2015.
[7] Y. Yang, Z. Zheng, K. Bian, L. Song, and Z. Han, "Real-time profiling of fine-grained air quality index distribution using UAV sensing," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 186–198, Feb. 2018.
[8] Y. Zhang, M. Chen, N. Guizani, D. Wu, and V. C. Leung, "SOVCAN: Safety-oriented vehicular controller area network," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 94–99, 2017.
[9] M. Hu, Z. Chen, K. Peng, X. Ma, P. Zhou, and J. Liu, "Periodic charging for wireless sensor networks with multiple portable chargers," *IEEE Access*, to be published, doi: 10.1109/ACCESS.2018.2885949.
[10] N. H. Motlagh, T. Taleb, and O. Arouk, "Low-altitude unmanned aerial vehicles-based Internet of Things services: Comprehensive survey and future perspectives," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 899–922, Dec. 2016.

[11] M. Chen and Y. Hao, "Task offloading for mobile edge computing in software defined ultra-dense network," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 587–597, Mar. 2018.
[12] C. Huang, Y. Chen, S. Xu, and H. Zhou, "The vehicular social network (VSN)-based sharing of downloaded geo data using the credit-based clustering scheme," *IEEE Access*, vol. 6, pp. 58254–58271, 2018.
[13] M. Hu, J. Luo, Y. Wang, M. Lukasiewycz, and Z. Zeng, "Holistic scheduling of real-time applications in time-triggered in-vehicle networks," *IEEE Trans. Ind. Informat.*, vol. 10, no. 3, pp. 1817–1828, Aug. 2014.
[14] M. Chen, Y. Hao, K. Lin, L. Hu, and Z. Yuan, "Label-less learning for traffic control in an edge network," *IEEE Netw.*, vol. 32, no. 6, pp. 8–14, Nov. 2018.
[15] C. C. Murray and A. G. Chu, "The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery," *Transp. Res. C, Emerg. Technol.*, vol. 54, pp. 86–109, May 2015.
[16] S. M. Ferrandez, T. Harbison, T. Weber, R. Sturges, and R. Rich, "Optimization of a truck-drone in tandem delivery network using k-means and genetic algorithm," *J. Ind. Eng. Manage.*, vol. 9, no. 2, pp. 374–388, 2016.
[17] X. Wang, S. Poikonen, and B. Golden, "The vehicle routing problem with drones: Several worst-case results," *Optim. Lett.*, vol. 11, no. 4, pp. 679–697, 2017.
[18] M. Chen, Y. Tian, G. Fortino, J. Zhang, and I. Humar, "Cognitive Internet of Vehicles," *Comput. Commun.*, vol. 120, pp. 58–70, May 2018.
[19] H. Savuran and M. Karakaya, "Route optimization method for unmanned air vehicle launched from a carrier," *Lect. Notes Softw. Eng.*, vol. 3, no. 4, p. 279, 2015.
[20] Z. Luo, Z. Liu, and J. Shi, "A two-echelon cooperated routing problem for a ground vehicle and its carried unmanned aerial vehicle," *Sensors*, vol. 17, no. 5, p. 1144, 2017.
[21] S. G. Manyam, D. W. Casbeer, and K. Sundar, "Path planning for cooperative routing of air-ground vehicles," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2016, pp. 4630–4635.
[22] T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins, "A unified solution framework for multi-attribute vehicle routing problems," *Eur. J. Oper. Res.*, vol. 234, no. 3, pp. 658–673, 2014.
[23] R. Cuda, G. Guastaroba, and M. G. Speranza, "A survey on two-echelon routing problems," *Comput. Oper. Res.*, vol. 55, pp. 185–199, Mar. 2015.
[24] G. Laporte, "The traveling salesman problem: An overview of exact and approximate algorithms, " *Eur. J. Oper. Res.*, vol. 59, no. 2, pp. 231–247, 1992.
[25] S. N. Kumar and R. Panneerselvam, "A survey on the vehicle routing problem and its variants," *Intell. Inf. Manage.*, vol. 4, no. 3, pp. 66–74, 2012.
[26] S. C. H. Leung, X. Zhou, D. Zhang, and J. Zheng, "Extended guided tabu search and a new packing algorithm for the two-dimensional loading vehicle routing problem," *Comput. Oper. Res.*, vol. 38, no. 1, pp. 205–215, 2011.
[27] M. Chen, Y. Hao, M. Qiu, J. Song, D. Wu, and I. Humar, "Mobility-aware caching and computation offloading in 5G ultradense cellular networks," *Sensors*, vol. 16, no. 7, pp. 974–987, 2016.
[28] K. L. Hoffman, M. Padberg, and G. Rinaldi, "Traveling salesman problem," in *Encyclopedia of Operations Research and Management Science*. Springer, 2013, pp. 1573–1578.
[29] K. Peng, M. Hu, C. Cai, X. Ma, Y. Yang, and J. Liu, "On simultaneous power replenishment for wireless sensor networks with multiple portable chargers," *IEEE Access*, vol. 6, pp. 63120–63130, 2018.
[30] M. Srinivas and L. M. Patnaik, "Genetic algorithms: A survey," *Computer*, vol. 27, pp. 17–26, Jun. 1994.
[31] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *Appl. Statist.*, vol. 28, no. 1, pp. 100–108, 1979.
[32] J. F. Campbell, D. Sweeney, and J. Zhang, "Strategic design for delivery with trucks and drones," Tech. Rep., 2017.
[33] A. Olivera and O. Viera, "Adaptive memory programming for the vehicle routing problem with multiple trips," *Comput. Oper. Res.*, vol. 34, no. 1, pp. 28–47, 2007.
[34] M. Cheikh, M. Ratli, O. Mkaouar, and B. Jarboui, "A variable neighborhood search algorithm for the vehicle routing problem with multiple trips," *Electron. Notes Discrete Math.*, vol. 47, pp. 277–284, Feb. 2015.
[35] K. Helsgaun, "General k-opt submoves for the Lin–Kernighan TSP heuristic," *Math. Program. Comput.*, vol. 1, nos. 2–3, pp. 119–163, 2009.

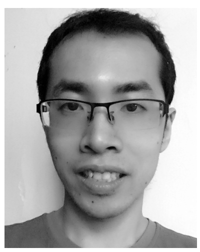
KAI PENG received the B.S., M.S., and Ph.D. degrees from the Huazhong University of Science and Technology in 1999, 2002, and 2006, respectively. His current research interests are in the areas of wireless sensor networking and big data processing.

WEIDONG LIU received the B.E. degree from the Huazhong University of Science and Technology, China, in 2017, where he is currently pursuing the M.S. degree with the School of Electronic Information and Communications.

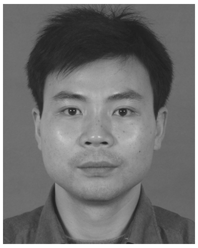


QIANGUO SUN received the B.E. degree from the Huazhong University of Science and Technology, China, in 2016, where he is currently pursuing the M.S. degree with the School of Electronic Information and Communications.

XIAOQIANG MA received the B.Eng. degree from the Huazhong University of Science and Technology, China, in 2010, and the M.Sc. and Ph.D. degrees from Simon Fraser University, Canada, in 2012 and 2015, respectively. His areas of interests include wireless networks, social networks, and cloud computing.



MENGLAN HU received the B.E. degree in electronic and information engineering from the Huazhong University of Science and Technology, China, in 2007, and the Ph.D. degree in electrical and computer engineering from the National University of Singapore, Singapore, in 2012. He is currently an Associate Professor with the School of Electronic Information and Communications, Huazhong University of Science and Technology, China. His research interests include cloud computing, parallel and distributed systems, scheduling and resource management, and wireless networking.



DESHENG WANG received the M.S. degree in electronics and information engineering from the Chinese Academy of Sciences, in 2001, and the Ph.D. degree in electronics and information engineering from the Huazhong University of Science and Technology, in 2004. He is currently an Associate Professor with the Huazhong University of Science and Technology. His current research interests include edge computing, high-density cooperative networking, radio resource allocation, and cross layer-based radio link protocol design. He is a member of the China Broadband Wireless IP Standard Group and FuTURE.

JIANGCHUAN LIU received the B.Eng. degree *(cum laude)* in computer science from Tsinghua University, Beijing, China, in 1999, and the Ph.D. degree in computer science from The Hong Kong University of Science and Technology in 2003. He is currently a Professor with the College of Natural Resources and Environment, South China Agricultural University. He has served on the Editorial Boards of the IEEE Transactions on Big Data, the IEEE Transactions on Multimedia, the IEEE Communications Surveys and Tutorials, the IEEE Access, the IEEE Internet of Things Journal, *Computer Communications*, and *Wireless Communications and Mobile Computing* (Wiley). He was the Steering Committee Chair of the IEEE/ACM IWQoS, from 2015 to 2017, and a TPC Co-Chair of the IEEE IC2E 2017 and IEEE/ACM IWQoS 2014. He serves as an Area Chair for the IEEE INFOCOM, ACM Multimedia, IEEE ICME, and so on.