Trajectory Design of Laser-Powered Multi-Drone Enabled Data Collection System for Smart Cities

Yao Du*, Kezhi Wang†, Kun Yang‡§ and Guopeng Zhang¶
*School of Information & Communication Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, China
†Department of Computer & Information Sciences, Northumbria University, Newcastle upon Tyne, UK
‡School of Computer Science & Electronic Engineering, University of Essex, Colchester, UK
§School of Communication Engineering, Xidian University, Xi'an, Shanxi, China
¶School of Computer Science & Technology, China University of Mining and Technology, Xuzhou, Jiangsu, China
Emails: *yaodu@std.uestc.edu.cn, †kezhi.wang@northumbria.ac.uk, ‡§kunyang@essex.ac.uk, ¶gpzhang@cumt.edu.cn

***Abstract*—This paper considers a multi-drone enabled data collection system for smart cities, where there are two kinds of drones, i.e., Low Altitude Platforms (LAPs) and a High Altitude Platform (HAP). In the proposed system, the LAPs perform data collection tasks for smart cities and the solar-powered HAP provides energy to the LAPs using wireless laser beams. We aim to minimize the total laser charging energy of the HAP, by jointly optimizing the LAPs' trajectory and the laser charging duration for each LAP, subject to the energy capacity constraints of the LAPs. This problem is formulated as a mixed-integer and non-convex Drones Traveling Problem (DTP), which is a combinatorial optimization problem and NP-hard. We propose an efficient and novel search algorithm named Drones Traveling Algorithm (DTA) to obtain a near-optimal solution. Simulation results show that DTA can deal with the large-scale DTP (i.e., more than 400 data collection points) efficiently. Moreover, the DTA only uses 5 iterations to obtain the near-optimal solution whereas the normal Genetic Algorithm needs nearly 10000 iterations and still fails to obtain an acceptable solution.**

***Index Terms*—Smart Cities, Internet of Things, Trajectory Optimization, Low Altitude Platforms, Multiple Traveling Salesmen Problem.**

I. Introduction

Low Altitude Platform (LAP) (or known as unmanned aerial vehicle, UAV) has attracted considerable attention [1] due to its high flexibility, low cost and the benefit of line-of-sight (LoS) air-to-ground communication links [2]. In order to collect the desired data, LAPs can fly above the massive Internet of Things Devices (IoTDs) in the Desired Regions (DRs). For example, the DRs in smart cities may include remote factories, farms and crowed buildings, etc.

The critical challenge to design a LAP-enabled data collection system is that the LAP is energy-constrained [3]. Thus, the LAP's flight time is too limited to perform the data collection mission for a whole city. The sun is a promising energy source to prolong the LAP's flight duration. However, the solar panel of a LAP is usually relatively small and the solar radiation intensity is not enough at low altitude. Consequently, the LAP cannot gather enough solar power. Fortunately, the High Altitude Platform (HAP) and the laser charging technique can provide the solution [4].

In this paper, the sun is the energy source of the HAP and thus the HAP can be aloft in the air continually. Different from the solar-powered LAP, the HAP can collect and store more solar energy due to the strong solar radiation and large solar panels. In addition, the key advantage of the solar-powered HAPs is their ability to adjust their positions according to the locations of LAPs. In the 3D aerial network, the HAP is capable of providing a robust wireless backhaul connectivity for LAPs [5], [6].

Moreover, the macro LAPs are relatively small and thus they can be powered by the laser. The HAP is able to make use of the laser charging technique [4] and then it can shot laser beams to power LAPs. In addition, the LAPs can bear the intense laser beams due to the specially designed laser charging panels. However, the human body may be seriously damaged because of being exposed to the laser beams [7]. Therefore, the laser charging spot should be fixed and isolated and the LAPs should return to it in order to be recharged after one flying cycle. Note that the HAP doesn't move in the plane and there is no laser scattering due to its flexibility [8].

Against the above background, in this paper, we aim to minimize the data collection system energy consumption by optimizing the LAPs trajectory and the laser charging duration from the HAP, subject to the energy constraint of the LAPs. We formulate this problem as a typical one deposit multiple traveling salesmen problem with the time window [9] and name this combinatorial optimization problem as the Drones Traveling Problem (DTP) for brevity. In addition, DTP is NP-hard and we can not solve it using convex optimization and the normal block coordinate descent method [3]. Moreover, the latest search algorithms for DTP, such as the Genetic Algorithm and the Particle Swarm Optimization [9], are not time-efficient for drones and most of them could not deal with the large-scale DTP (i.e., more than 400 DRs). We propose an efficient and novel search algorithm named Drones Traveling Algorithm (DTA) for large-scale multi-drone trajectory design.

The rest of this paper is organized as follows. Section II introduces the system model and the optimization problem. In Section III, we introduce DTA to solve the proposed DTP. Section IV provides the simulation results. Finally, we conclude the paper in Section V.

II. System Model and Problem Formulation

We consider a new laser-powered multi-drone enabled IoT data collection system, as illustrated in Fig.1. In the proposed system, there are two kinds of drones, the LAPs and a solar-powered HAP served as the energy charging station for all the LAPs. The multiple LAPs are employed as the mobile data collectors to collect information from the IoTDs on the ground.

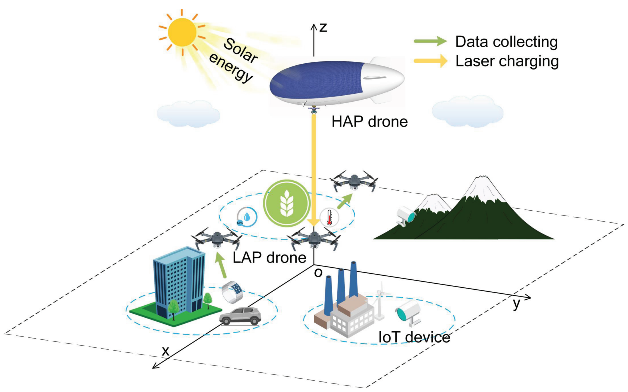

Fig. 1. The proposed laser-powered multi-drone enabled data collection system for smart cities

In the proposed system, a frequency division duplexing (FDD) mode is considered for the links between the HAP and LAPs [5], the time division duplexing (TDD) mode is applied in each DR covered by the LAP [10].

Without loss of generality, a three-dimensional (3D) Euclidean coordinate is adopted. We define O as the laser charging spot which is the geometric center of the target area. In our proposed system, there are K DRs. We assume the geometric center of each DR is fixed and known [3]. The HAP designs the LAPs trajectory and controls M LAPs, each j-th LAP hovers above S_j DR centers one by one to collect IoTDs' information.

The location of the HAP and the j-th LAP are denoted by (0, 0, H) and (x_j, y_j, h) respectively, $j \in \mathcal{M} = \{1, 2, ... ,M\}$. We assume each LAP completes the data collection operations of each k-th DR in T^k seconds, $k \in \mathcal{K} = \{1, 2, ... ,K\}$, during which the IoTDs upload their data to the LAP.

A. Trajectory Model

In our proposed system, each LAP flies straightly from one DR to another. Each LAP hovers above the geographic coordinate center of each DR to collect the IoTDs data. We consider the HAP hovers at (0, 0, H) and shots laser beams to the LAPs one by one at a fixed position $q_j[0]$ = (0, 0, h). Each LAP is powered by a large capacitance, which can be recharged by laser power beams at the laser charging spot to maintain their continuous flight. In addition, according to our proposed system model, each j-th LAP will return to the same safe laser charging spot (i.e., $q_j[0]$) after one flying cycle. Therefore, one has

$$q_j[S_j] = q_j[0], \forall j \in \mathcal{M} \tag{1}$$

We assume each j-th LAP serves S_j DRs. Furthermore, we define $q_j[t]$, $\forall t \in \mathcal{T}_j = \{1, 2, ... ,S_j\}$, as the coordinate of the t-th DR geographic center the j-th LAP hovers above. We define $d_j[t]$ as the distance between the geographic coordinate center of DRs, thus one can have

$$d_j[t] = \|q_j[t] - q_j[t-1]\|, \forall j \in \mathcal{M}, \forall t \in \mathcal{T}_j \tag{2}$$

Note that if the j-th LAP chooses the k-th DR as its t-th hovering DR, $a_j^k[t] = 1$, otherwise, $a_j^k[t] = 0$. In order to guarantee that each DR is covered and served only once, we have

$$\sum_{j=1}^{M}\sum_{t=1}^{S_j} a_j^k[t] = 1, \forall j \in \mathcal{M}, \forall k \in \mathcal{K}, \forall t \in \mathcal{T}_j \tag{3}$$

The hovering time of each j-th LAP is given as

$$T_j^H = \sum_{k=1}^{K}\sum_{t=1}^{S_j} a_j^k[t]T^k, \forall j \in \mathcal{M}, \forall k \in \mathcal{K}, \forall t \in \mathcal{T}_j \tag{4}$$

The total meters of each j-th LAP flight tour length is

$$L_j = \sum_{t=1}^{S_j} d_j[t], \forall t \in \mathcal{T}_j, \forall j \in \mathcal{M} \tag{5}$$

We assume each LAP flies in a straight line from one DR to the next DR with the same constant speed v m/s. Therefore, the flight time of each j-th LAP is given as

$$T_j^F = \frac{L_j}{v}, \forall j \in \mathcal{M} \tag{6}$$

B. The Drones Energy Consumption Model

Assuming the LAP consumes energy for its hovering and flight, we define P^H as the power consumption when the LAP is hovering, whereas P^F as the LAP power consumption when it is flying straightly. Then each j-th LAP energy consumption is

$$E_j = P^H T_j^H + P^F T_j^F, \forall j \in \mathcal{M} \tag{7}$$

However, each j-th LAP could not fly perpetually because of its laser capacitance capacity limitation C_j, therefore

$$E_j \leq C_j, \forall j \in \mathcal{M} \tag{8}$$

The laser energy each j-th LAP received from the HAP should be enough for its flight

$$\eta_j P_j^l \tau_j \geq E_j, \forall j \in \mathcal{M} \tag{9}$$

The τ_j is the laser charging time for each j-th LAP. We define P_j^r as the external-cavity laser power received at the j-th LAP laser powering beam receiver, P_j^l as the laser beam power shot to the j-th LAP and α_j as the laser attenuation coefficient. Then $\eta_j = \frac{P_j^r}{P_j^l} = e^{-\alpha_j(H-h)}$ is the laser transmission efficiency between the j-th LAP and the HAP [4].

C. Problem Formulation

In this subsection, we propose the DTP, where we aim to minimize the energy consumption of the proposed data collection system. Let $\boldsymbol{A} = \{a_j^k[t], \forall j \in \mathcal{M}, \forall k \in \mathcal{K}, \forall t \in \mathcal{T}_j\}$, $\boldsymbol{S} = \{S_j, \forall j \in \mathcal{M}\}$, $\boldsymbol{Q} = \{q_j[t], \forall j \in \mathcal{M}, \forall t \in \mathcal{T}_j\}$ and $\boldsymbol{\tau} = \{\tau_j, \forall j \in \mathcal{M}\}$.

In the optimization problem below, we aim to jointly optimize the DRs selection (i.e., $\boldsymbol{A}$), the multi-LAP routes (i.e., $\boldsymbol{Q}$), the hovering times of each LAP (i.e., $\boldsymbol{S}$) and the laser charging durations (i.e., $\boldsymbol{\tau}$). The DTP is formulated as

$$\mathcal{P}1: \underset{\boldsymbol{A}, \boldsymbol{S}, \boldsymbol{Q}, \boldsymbol{\tau}}{\text{minimize}} \sum_{j=1}^{M} \eta_j P_j^l \tau_j \tag{10a}$$

$$\text{s.t. } E_j \leq C_j, \forall j \in \mathcal{M} \tag{10b}$$

$$\eta_j P_j^l \tau_j \geq E_j, \forall j \in \mathcal{M} \tag{10c}$$

$$q_j[S_j] = q_j[0], \forall j \in \mathcal{M} \tag{10d}$$

$$\sum_{j=1}^{M} \sum_{t=1}^{S_j} a_j^k[t] = 1, \forall j \in \mathcal{M}, \forall k \in \mathcal{K}, \forall t \in \mathcal{T}_j \tag{10e}$$

$$d_j[t] = \|q_j[t] - q_j[t-1]\|, \forall t \in \mathcal{T}_j, \forall j \in \mathcal{M} \tag{10f}$$

$$\mathcal{T}_j = \{1, 2, ..., S_j\}, \forall j \in \mathcal{M} \tag{10g}$$

The LAPs use the laser energy to fly and hover, the charging duration τ_j^* is optimal if and only if the laser energy is not wasted, which means the equality holds in (10c). Therefore, we can relax the DTP (i.e., $\mathcal{P}1$) by holding the equality of (10c) and obtain

$$\tau_j^* = \frac{\sum_{k=1}^{K} \sum_{t=1}^{S_j} P^H a_j^k[t] T^k + \sum_{t=1}^{S_j} P^F d_j[t] v^{-1}}{\eta_j P_j^l} \tag{11}$$

Let the optimal charging time substitute into the original $\mathcal{P}1$, the objective function becomes the total energy consumption of the M LAPs. One can have

$$\underset{\boldsymbol{A}, \boldsymbol{S}, \boldsymbol{Q}}{\text{minimize}} \sum_{j=1}^{M} \sum_{k=1}^{K} \sum_{t=1}^{S_j} P^H a_j^k[t] T^k + \sum_{j=1}^{M} \sum_{t=1}^{S_j} P^F d_j[t] v^{-1}$$
$$\text{s.t. (10b), (10d), (10e), (10f), (10g)} \tag{12}$$

One can see that problem (12) is still NP-hard [9] and it is known to be difficult to find its optimal solution. Note that different from the other optimization problems [3], [8], [11], the S_j is also required to be optimized. Moreover, constraint (10g) makes it a complex combinatorial optimization problem and we can not solve it using convex optimization and the block coordinate descent method.

III. Trajectory Design

In this section, we consider the multi-drone trajectory design. We first analyze the DTP in order to obtain the optimal direct graph $\boldsymbol{s}^*$ and then we propose the DTA.

A. Analysis of The Drones Traveling Problem

By observing the structure of problem (12), we find that the solutions $\boldsymbol{A}$, $\boldsymbol{S}$ and $\boldsymbol{Q}$ actually define a direct graph $\boldsymbol{s}$ consisting of N points and M cycles with the same starting point [12] (i.e., the laser charging spot $q_j[0]$), therefore, solving problem (12) is equivalent to obtaining the optimal direct graph $\boldsymbol{s}^*$ to minimize the object function of problem (12). Furthermore, in order to obtain the optimal direct graph $\boldsymbol{s}^*$, we first give the analysis of the DTP as follows.

Similar to the standard definition of the multiple traveling salesmen problem [9], we describe each LAP as a salesman, the DRs are treated as cities and the hovering duration T^k is described as the time window in each k-th city. Given any candidate direct graph $\boldsymbol{s}$, we can obtain the solutions $\boldsymbol{A}$, $\boldsymbol{S}$ and $\boldsymbol{Q}$ of problem (12), we define the summation of the LAPs' energy consumption (i.e., the object function of problem (12)) as the utility function denoted by $f(\boldsymbol{s})$.

Based on the above observation, we discover that the poor solutions of DTP usually have many crossover points, we describe each crossover point as a knot in the graph. To be specific, there are two kinds of knots in the graph, i.e., the knots in the cycle that crosses over itself and the knots between any two cycles. The well-known *2-opt algorithm* is an efficient algorithm to eliminate the first kind self-knot by reordering the route of a cycle [13].

Motivated by *2-opt algorithm* [13], we introduce Theorem 1 to eliminate the second kind of knots between any two cycles.

Theorem 1. *Define the No-knot graph as the graph with no knots. Given a candidate direct graph $\boldsymbol{s}$ with one or more knots between any two cycles, the total LAPs' tour length of the No-knot graph is always less than the tour length of $\boldsymbol{s}$. The No-knot graph can be obtained by reordering the drones route $\boldsymbol{Q}$ using the Flow Direction Method.*

Proof. The sum of the two sides of the triangle is greater than the third side. Refer to Example 1 to find the introduction of the *Flow Direction Method*. □

Example 1. *The Flow Direction Method in the DTP: By exploring the structure of the knots in the candidate direct graphs, we summarize all kinds of knots between any two cycles into 'Reverse Flow Knot' and 'Co-current Flow Knot', which are defined and illustrated in Fig. 2.*

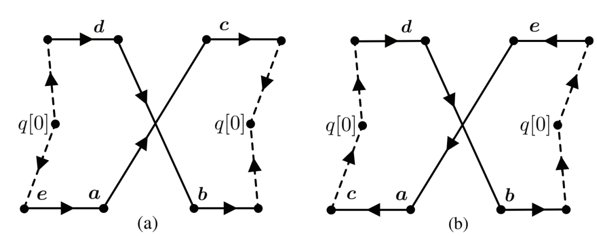

Fig. 2. (a) Co-current flow knot; (b) Reverse flow knot

In Fig. 2, each $q[0]$ is the same charging spot. The other trajectory points are included in the dotted line. We reorder the two LAPs' route as follows. For the co-current flow knot, *the routes are reordered and given as* $q[0] \rightarrow \cdots \rightarrow \boldsymbol{d} \rightarrow \boldsymbol{c} \rightarrow \cdots \rightarrow q[0]$ *and* $q[0] \rightarrow \cdots \rightarrow \boldsymbol{e} \rightarrow \boldsymbol{a} \rightarrow \boldsymbol{b} \rightarrow \cdots \rightarrow q[0]$. *For the reverse flow knot, the new routes are* $q[0] \rightarrow \cdots \rightarrow \boldsymbol{e} \rightarrow \boldsymbol{d} \rightarrow \cdots \rightarrow q[0]$ *and* $q[0] \rightarrow \cdots \rightarrow \boldsymbol{c} \rightarrow \boldsymbol{a} \rightarrow \boldsymbol{b} \rightarrow \cdots \rightarrow q[0]$.

B. The Drones Traveling Algorithm (DTA)

Before we introduce the DTA (i.e., Algorithm 1), we first give a brief introduction to our search mechanism.

- We define the local optimum graph as the graph with no self-knot in each LAP cycle;
- We develop an efficient method to *Jump from the local optimum graph to a better graph* where the better graph is defined as the graph with less $f(\boldsymbol{s})$. The *Jump operations* includes three simple operations: the *Exchanging*, *Shifting* and the *Knot Removing*;
- The near-optimal or even the optimal graph $\boldsymbol{s}^*$ can be obtained by the iterations between the *Local Optimum Search Method* (LOSM) and the *Jump Method* (introduced next) until the utility function $f(\boldsymbol{s})$ does not decrease.

1) LOSM: We define LOSM as that the *2-opt algorithm* [13] has been applied to each j-th LAP cycle in graph $\boldsymbol{s}$.

2) Jump Method: The Jump method is reordering the route $\boldsymbol{Q}$ of the direct graph $\boldsymbol{s}$. We define $\boldsymbol{a}$ as the current DR, which is picked from the set of DRs (i.e., $\mathcal{K}$). We also define that DR $\boldsymbol{b}$ is from $\mathcal{N}(\boldsymbol{a})$, which is the neighborhood [13] of $\boldsymbol{a}$. We define $\mathcal{N}(\boldsymbol{a})$ consists of 6 DRs, which is suitable to cover enough neighbor points and reduce the algorithm complexity [14]. In addition, $\boldsymbol{c}$ is the next DR connected to $\boldsymbol{a}$.

Then the '*Exchanging*' operation is defined as that exchanging $\boldsymbol{b}$ and $\boldsymbol{c}$ in the route $\boldsymbol{Q}$; the '*Shifting*' operation is defined as that taking $\boldsymbol{b}$ out and inserting $\boldsymbol{b}$ between $\boldsymbol{a}$ and $\boldsymbol{c}$; the '*Knot Removing*' operation is defined as that using the *Flow Direction Method* to eliminate all the knots between any two cycles. Note that the Jump Method is terminated until all the DRs and their neighbor DRs have been searched.

Example 2. *Given a candidate direct graph* $\boldsymbol{s}$ *in which* $\boldsymbol{a}$ *and* $\boldsymbol{b}$ *are in the same cycle, the results of the Exchanging and Shifting can be illustrated as Fig. 3.*

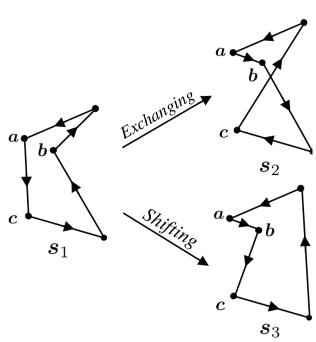

Fig. 3. The 'Exchanging' and 'Shifting' operations in the same cycle

One can see that the tour length of $\boldsymbol{s}_2 > \boldsymbol{s}_1$ and the tour length of $\boldsymbol{s}_3 < \boldsymbol{s}_1$, which means only the shifting operation works, and $\boldsymbol{s}_3$ becomes the new graph.

Example 3. *Given a candidate direct graph* $\boldsymbol{s}$ *in which the* $\boldsymbol{b}$ *and* $\boldsymbol{c}$ *are in two different cycles, the results of the Exchanging and Shifting can be illustrated as Fig. 4.*

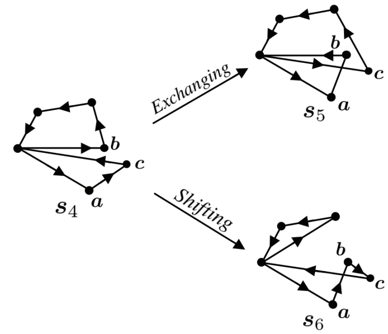

Fig. 4. The 'Exchanging' and 'Shifting' operations in two different cycles

One can see that the tour length of $\boldsymbol{s}_5 > \boldsymbol{s}_4$ and the tour length of $\boldsymbol{s}_6 < \boldsymbol{s}_4$, which means $\boldsymbol{s}_6$ becomes the new graph. We illustrate the Examples in Fig. 5 for easily understanding.

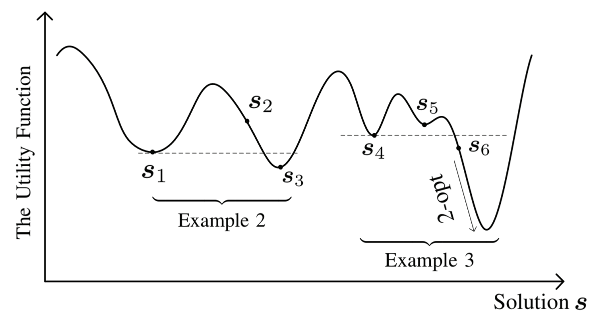

Fig. 5. An approximate illustration of the examples

Algorithm 1: The Drones Traveling Algorithm (DTA) for $\mathcal{P}1$

Input: $\boldsymbol{A}^0$, $\boldsymbol{Q}^0$, $\boldsymbol{S}^0$: Random Solutions
Output: $\boldsymbol{A}^*$, $\boldsymbol{Q}^*$, $\boldsymbol{S}^*, \boldsymbol{\tau}^*$: The Near-optimal Solutions

$r \leftarrow 1$; // the iterations of DTA
Generate a direct graph $\boldsymbol{s}^r$ using $\boldsymbol{A}^0$, $\boldsymbol{Q}^0$, $\boldsymbol{S}^0$;
$f(\boldsymbol{s}^0) \leftarrow \infty$; // initialize the utility function
while $f(\boldsymbol{s}^r) < f(\boldsymbol{s}^{r-1})$ **do**
    Use the *2-opt algorithm* to obtain the local optiaml direct graph $\boldsymbol{s}^r$;
    Use Algorithm 2 to *Jump* from graph $\boldsymbol{s}^r$ to a new direct graph $\boldsymbol{s}^{r+1}$;
    $r \leftarrow r + 1$;
end
Use direct graph $\boldsymbol{s}^r$ to obtain $\boldsymbol{A}^*$, $\boldsymbol{Q}^*$, $\boldsymbol{S}^*$;
Use Equation (11) to obtain the optimal $\boldsymbol{\tau}^*$;
return: $\boldsymbol{A}^*$, $\boldsymbol{Q}^*, \boldsymbol{S}^*, \boldsymbol{\tau}^*$.

Based on the above introduction, we provide the detailed Jump Method of DTA as Algorithm 2.

Algorithm 2: The Jump Method of DTA

Input: $\boldsymbol{s}$: The Local Optimum Direct Graph
Output: $\widetilde{\boldsymbol{s}}$: A New Direct Graph

$\widetilde{\boldsymbol{s}} \leftarrow \boldsymbol{s}$; // initialize graph $\widetilde{\boldsymbol{s}}$
for $k \leftarrow$ *1 to* K **do**
  $\boldsymbol{a} \leftarrow$ the k-th DR; // the current DR
  $\boldsymbol{c} \leftarrow$ the next DR connected to $\boldsymbol{a}$;
  for $i \leftarrow$ *1 to the size of* $\mathcal{N}(\boldsymbol{a})$ **do**
    Select an unselected DR $\boldsymbol{b}_i$ from $\mathcal{N}(\boldsymbol{a})$;
    Exchange $\boldsymbol{b}_i$ and $\boldsymbol{c}$ in the DRs route $\boldsymbol{Q}$ and generate a new graph $\boldsymbol{g}$; // the Exchanging operation, refer to Example 2 and Example 3
    if *the graph* $\boldsymbol{g}$ *is feasible and the Utility Function* $f(\boldsymbol{g}) < f(\widetilde{\boldsymbol{s}})$ **then**
      $\widetilde{\boldsymbol{s}} \leftarrow \boldsymbol{g}$;
      break;
    end
    Insert $\boldsymbol{b}_i$ between $\boldsymbol{a}$ and $\boldsymbol{c}$ in $\boldsymbol{Q}$ and generate a new graph $\boldsymbol{g}$; // the Shifting operation, refer to Example 2 and Example 3
    if *the graph* $\boldsymbol{g}$ *is feasible and the Utility Function* $f(\boldsymbol{g}) < f(\widetilde{\boldsymbol{s}})$ **then**
      $\widetilde{\boldsymbol{s}} \leftarrow \boldsymbol{g}$;
      break;
    end
    Reorder route $\boldsymbol{Q}$ by the '*Flow Direction Method*' and generate a new graph $\boldsymbol{g}$; //the Knot Removing operation, refer to Example 1
    if *the graph* $\boldsymbol{g}$ *is feasible and the Utility Function* $f(\boldsymbol{g}) < f(\widetilde{\boldsymbol{s}})$ **then**
      $\widetilde{\boldsymbol{s}} \leftarrow \boldsymbol{g}$;
      break;
    end
  end
end
return: $\widetilde{\boldsymbol{s}}$.

IV. Simulation Results

In this section, simulation results are presented to show the effectiveness of our proposed DTA. We suppose that there are 3 LAPs flying over K = 500 DRs, which are distributed within a geographic area of size 5 km × 5 km. In addition, the HAP is suspended at a fixed altitude H = 5 km and the LAP flies at a fixed altitude h = 100 m. Therefore, the laser transmission efficiency η_j is given by 0.6 [4]. The laser beam power is set as 1 kW [7]. The LAP hovering power consumption is set as P^H = 50 W and the LAP flight consumption is set as P^F = 100 W. The LAP hovering time T^k is set as 5 seconds. The LAP flight speed is set as 10 m/s. We run all the simulation on the computer with the 3.20 GHz CPU and 8 GB RAM. The simulation software is Matlab 2017a running on Windows 10.

Next, we validate the performance of DTA as compared to two well-known benchmark schemes, which are designed for the DTP, namely the near-neighbor Greedy search algorithm and the genetic algorithm [9]. The near-neighbor Greedy search algorithm takes turns to choose the nearest but not yet visited DR as the next DR until all the DRs are visited.

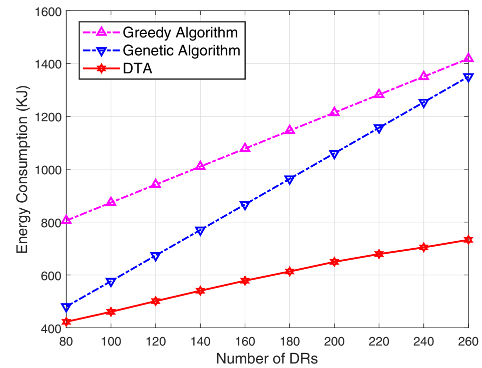

Fig. 6. The energy consumption versus the number of DRs.

One can see from Fig. 6 that with the increasing of the number of DRs, the energy consumption increases, as expected. Furthermore, the system designed by DTA consumes less energy compared with the other benchmarks. Therefore DTA is energy-efficient.

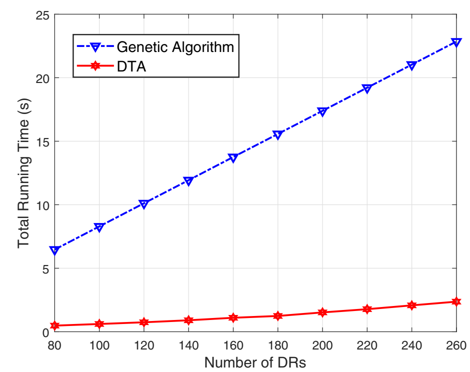

Fig. 7. The running time versus the number of DRs.

Fig. 7 presents the time-effectiveness of the DTA. One can see that the running speed of DTA is about 10 times faster than Genetic Algorithm and the performance of the DTA is much better than the Genetic Algorithm according to Fig. 6. Therefore, the DTA is time-effective and thus it is suitable for the large-scale multi-drone trajectory design.

In Fig. 8, we test the DTA in the large scale scenario (i.e., 500 DRs). In Fig. 8 (a) to (c), compared with the greedy algorithm and the genetic algorithm, the DTA's solution graph has no knots and has the least energy consumption, thus the DTA outperforms the other algorithms. Moreover, the DTA only uses 5 iterations to obtain the acceptable near-optimal solution whereas the Genetic Algorithm uses 10000 iterations and fails to obtain an acceptable solution. In Fig. 8 (d), we show the DTA efficiency. One can see from Fig. 8 (d) that the utility function is significantly reduced during each DTA Jump operation. Therefore, the DTA is very efficient.

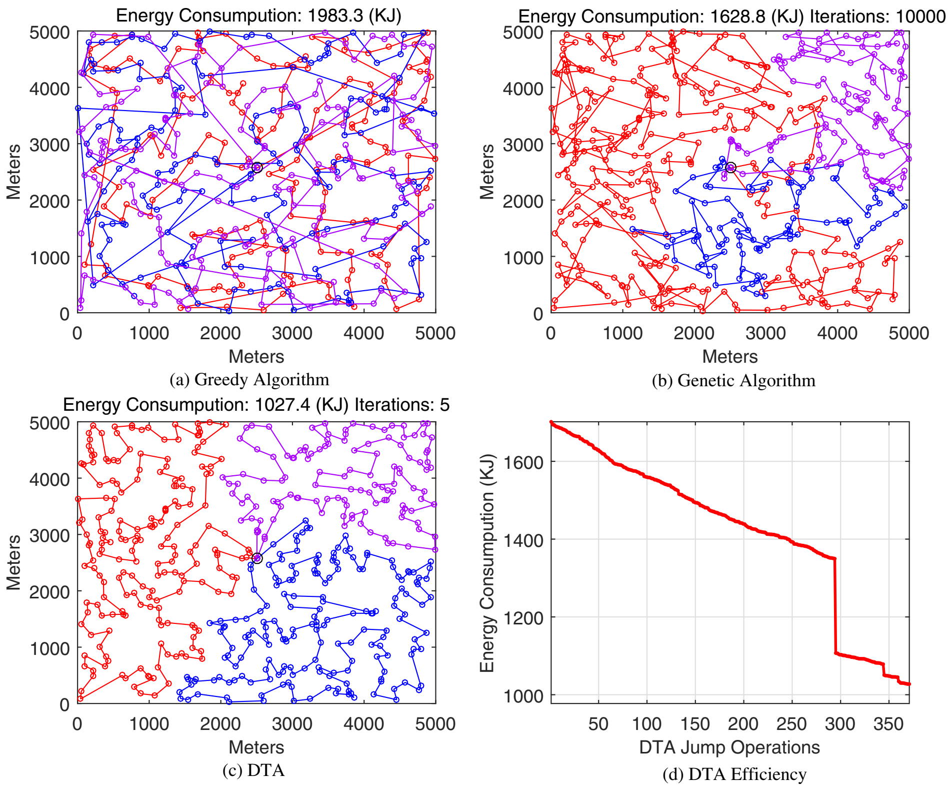

Fig. 8. The DTA energy consumption compared with the greedy algorithm and the genetic algorithm.

V. Conclusion

In this paper, we optimize the LAPs trajectory and the HAP laser power duration for energy saving purpose. The optimization problem is NP-hard. We propose an efficient search algorithm named DTA to obtain a near-optimal solution. Simulation results show that DTA only uses 5 iterations to obtain the near-optimal solution whereas the Genetic Algorithm uses 10000 iterations and still fails to obtain an acceptable solution in the case of the large scale DTP.

Acknowledgment

The work in the paper was partly funded by Natural Science Foundation of China (Grant No. 61620106011, 61572389, 61971421), Zhongshan City Team Project (Grant No. 180809162197874) and UK EPSRC Project NIRVANA (EP/L026031/1).

References

[1] J. Wang, C. Jiang, Z. Han, Y. Ren, R. G. Maunder, and L. Hanzo, "Taking drones to the next level: Cooperative distributed unmanned-aerial-vehicular networks for small and mini drones," *IEEE Vehicular Technology Magazine*, vol. 12, no. 3, pp. 73–82, Sep. 2017.

[2] J. Wang, C. Jiang, Z. Wei, C. Pan, H. Zhang, and Y. Ren, "Joint UAV hovering altitude and power control for space-air-ground IoT networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1741–1753, Apr. 2019.

[3] Y. Du, K. Wang, K. Yang, and G. Zhang, "Energy-efficient resource allocation in UAV based MEC system for IoT devices," in *2018 IEEE GLOBECOM*, Dec. 2018, pp. 1–6.

[4] Q. Zhang, W. Fang, Q. Liu, J. Wu, P. Xia, and L. Yang, "Distributed laser charging: A wireless power transfer approach," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3853–3864, 2018.

[5] M. Mozaffari, A. Taleb Zadeh Kasgari, W. Saad, M. Bennis, and M. Debbah, "Beyond 5G with UAVs: Foundations of a 3D wireless cellular network," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 357–372, Jan. 2019.

[6] K. Yang, S. Ou, K. Guild, and H. Chen, "Convergence of ethernet PON and IEEE 802.16 broadband access networks and its QoS-aware dynamic bandwidth allocation scheme," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 2, pp. 101–116, Feb. 2009.

[7] H. Kaushal and G. Kaddoum, "Applications of lasers for tactical military operations," *IEEE Access*, vol. 5, pp. 20 736–20 753, 2017.

[8] Y. Sun, D. Xu, D. W. K. Ng, L. Dai, and R. Schober, "Optimal 3D-trajectory design and resource allocation for solar-powered UAV communication systems," *IEEE Transactions on Communications*, vol. 67, no. 6, pp. 4281–4298, Jun. 2019.

[9] H. Zhou, M. Song, and W. Pedrycz, "A comparative study of improved GA and PSO in solving multiple traveling salesmen problem," *Applied Soft Computing*, vol. 64, pp. 564–580, 2018.

[10] K. Yang, S. Ou, H. Chen, and J. He, "A multihop peer-communication protocol with fairness guarantee for IEEE 802.16-based vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3358–3370, Nov. 2007.

[11] K. Wang, P. Huang, K. Yang, C. Pan, and J. Wang, "Unified offloading decision making and resource allocation in ME-RAN," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 8159–8172, Aug. 2019.

[12] D. B. West *et al.*, *Introduction to graph theory*. Prentice Hall, Upper Saddle River, New Jersey, 1996, vol. 2.

[13] G. A. Croes, "A method for solving traveling-salesman problems," *Operations research*, vol. 6, no. 6, pp. 791–812, 1958.

[14] Z. Yuan, "Solving multiple traveling salesmen problem with minimal maximum," *Computer Systems and Applications*, vol. 27, pp. 145–149 (in Chinese), 2018.