Minimizing the Number of Deployed UAVs for Delay-bounded Data Collection of IoT Devices

Junqi Zhang[a]†, Zheng Li[a]†, Wenzheng Xu[a*], Jian Peng[a], Weifa Liang[b], Zichuan Xu[c], Xiaojiang Ren[d], Xiaohua Jia[e]
[a]College of Computer Science, Sichuan University, Chengdu, P. R. China
[b]Research School of Computer Science, The Australian National University, Canberra, Australia
[c]School of Software, Dalian University of Technology, Dalian, P. R. China
[d]School of Telecommunications Engineering, Xidian University, Xi'an, P. R. China
[e]Department of Computer Science, City University of Hong Kong, Hong Kong, P. R. China
Emails: junqizhangscu@163.com, lizheng@scu.edu.cn, wenzheng.xu3@gmail.com, jianpeng@scu.edu.cn, wliang@cs.anu.edu.au, z.xu@dlut.edu.cn, richard.rxj@outlook.com, csjia@cityu.edu.hk
Type your text †The first two authors Junqi Zhang and Zheng Li contributed equally to this work, i.e., co-first authors
*Corresponding author: Wenzheng Xu

IEEE INFOCOM 2021 - IEEE Conference on Computer Communications | 978-1-6654-0325-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/INFOCOM42981.2021.9488887

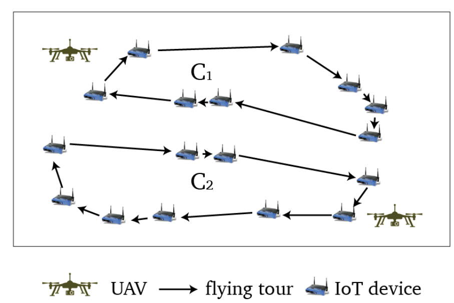
***Abstract*—In this paper, we study the deployment of Unmanned Aerial Vehicles (UAVs) to collect data from IoT devices, by finding the data collection tour of each UAV. To ensure the 'freshness' of the collected data, a strict requirement is that the total time spent in the tour of each UAV, which consists of UAV flying time and data collection time, must be no greater than a given maximum data collection delay B, e.g., 20 minutes. In this paper, we consider a problem of using the minimum number of UAVs and finding their data collection tours, subject to the constraint that the total time spent in each tour is no greater than B. We study two variants of the problem, one is that a UAV needs to fly to the location of each IoT device to collect its data; the other variant is that a UAV is able to collect the data of the IoT device as long as their Euclidean distance is no greater than a given wireless transmission range. For the first variant of the problem, we propose a novel 4-approximation algorithm, which improves the best approximation ratio $4\frac{4}{7}$ so far. For the second variant, we design the first constant factor approximation algorithm. In addition, we evaluate the performance of the proposed algorithms via extensive experiments, and experimental results show that the average numbers of UAVs deployed by the proposed algorithms are from 11% to 19% less than those by existing algorithms.**

***Index Terms*—Mobile data collection; multiple UAV scheduling; minimum cycle cover with neighborhoods; approximation algorithms.**

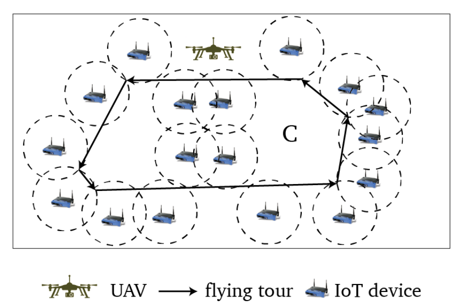
I. Introduction

Due to the flexibility and cost-efficiency of Unmanned Aerial Vehicles (UAVs), they are widely used in many applications, including package delivery, target tracking, emergency aid, charging wireless sensor networks [12], [18], [20], [24], [26], [27], [28], [30], [35], [37], and so on. On the other hand, millions of Internet of Thing (IoT) devices, such as various sensors and smart monitoring devices, have been deployed in many IoT networks in the past a few years for various applications.

In this paper, we study the data collection of IoT devices in a large-scale IoT network, e.g., tens of square kilometers, where IoT devices are only sparsely located at some strategic locations to monitor important Points of Interest (PoIs) in the network. Due to the large scale of the network and the limited energy of IoT devices, sometimes it is unrealistic to allow them to transmit or relay sensing data to a base station via multihop relays.



(a) Data collection without neighborhoods by two UAVs and their data collection tours



(b) Data collection with neighborhoods by only one UAV and its data collection tour

Fig. 1: Data collection by UAVs in two scenarios.

We consider the deployment of multiple light-weight UAVs to collect data from IoT devices, where a UAV can fly to a location nearby an IoT device to collect its data, thereby saving the energy consumption of the IoT device. Fig. 1(a) shows that two UAVs are deployed and they collect data of IoT devices along their data collection flying tours, respectively.

To ensure the 'freshness' of the collected data, a strict requirement is that the total time spent in the tour of each UAV, which consists of UAV flying time and data collection time, should be no greater than a given maximum data collection delay B, e.g., 20 minutes [4]. Otherwise, the collected data is somewhat 'stale'. For example, consider networks in which IoT devices are deployed to monitor bushfires in a forest [16] or PM 2.5 pollution in a city [29], it is important to collect sensing data as quickly as possible.

In this paper, we study a novel *minimum UAV deployment problem*, which is to determine the minimum number of UAVs to-be-deployed and find their data collection tours, such that the data of each IoT device are collected by one of the UAVs, subject to that the total time spent by each UAV in its tour is no greater than the maximum data collection delay B.

We consider two variants of the minimum UAV deployment problem. In the first variant of the problem, which is termed as *the minimum UAV deployment problem without neighborhoods*, a UAV needs to fly to the location of each IoT device to collect its data, see Fig. 1(a). In this case, the wireless transmission range of the IoT device is much shorter than the scale of the network, or the device cannot transmit data in a wireless way. One application example is that each IoT device is an RFID tag. To collect the data of the tag, a UAV must be equipped with an RFID reader and the reader can read the data of the tag only when their Euclidean distance is very short, e.g., a few meters [36]. Another application is the deployment of UAVs to monitor traffic jams in a smart city, where UAVs are dispatched to monitor PoIs (e.g., congested cross roads), and no IoT devices are deployed at all.

In the other variant of the problem, which is referred to as *the minimum UAV deployment problem with neighborhoods*, an IoT device is able to transmit its data to a UAV by wireless transmission and the data can be received by the UAV when the Euclidean distance between the device and the UAV is no greater than a given communication range, see Fig. 1(b). Unlike the wireless communication between two devices on the ground that wireless signals degrade very quickly due to various shadowing and scattering, the radio signals from a ground device to a UAV in the air, or vice visa, degrade much slower, due to less obstacles between them [1], [5]. Therefore, the wireless communication range between a ground IoT device and a UAV usually is much longer than the range between two ground IoT devices, e.g., 500 m vs. 50 m [1], [2], [5]. It then can be seen that by exploiting the long communication range between an IoT device and a UAV, the number of deployed UAVs may be reduced. For example, Fig. 1(a) shows that two UAVs are deployed when a UAV must fly to the location of each IoT device to collect its data, while Fig. 1(b) demonstrates that only one UAV is deployed when taking the communication range into consideration.

The novelties of this paper are two folds. (1) We propose a 4-approximation algorithm for the minimum UAV deployment problem *without* neighborhoods, which improves the best approximation ratio $4\frac{4}{7}$ so far [32]. Then, less numbers of UAVs will be deployed. (2) We design the very first constant factor approximation algorithm for the minimum UAV deployment problem *with* neighborhoods. Experimental results later show that much less number of UAVs will be deployed when taking the wireless communication range (i.e., neighborhoods) into consideration.

The main contributions of this paper are summarized as follows. (i) We study a novel minimum UAV deployment problem, which is determine the minimum number of UAVs to-be-deployed and find their data collection tours, subject to a stringent constraint that the total time spent by each UAV in its tour must be no greater than a given maximum data collection delay. (ii) For the first variant of the problem – the minimum UAV deployment problem *without neighborhoods* where a UAV needs to fly to the location of each IoT device to collect its data, we propose a 4-approximation algorithm, which improves the best approximation ratio $4\frac{4}{7}$ so far. (iii) For the second variant of the problem – the minimum UAV deployment problem *with neighborhoods* where a UAV can collect the data of each IoT device as long as their Euclidean distance is no greater than a given communication range, we design the first constant factor approximation algorithm for it. (iv) We evaluate the performance of the proposed algorithms via extensive experiments. Experimental results show that the numbers of UAVs deployed by the proposed algorithms are around from 11% to 19% less than those by existing algorithms.

The rest of the paper is organized as follows. Section II introduces network model and data collection models, and defines the problems. Sections III and IV propose approximation algorithms for the minimum UAV deployment problem without neighborhoods and with neighborhoods, respectively. Section V evaluates the performance of the proposed algorithms. Section VI reviews related work, and Section VII concludes this paper.

II. Preliminaries

A. Network model

We consider an IoT application scenario where many IoT devices are deployed in an area to monitor important Points of Interest (PoIs). For example, IoT devices are used to monitor PM 2.5 pollution in a city [29] or bushfires in a forest [16]. Assume that there are n devices $v_1, v_2, \ldots, v_n$ deployed at some strategic locations in the monitoring area, where n is a positive integer. Let V be the set of the n devices, i.e., $V = \{v_1, v_2, \ldots, v_n\}$. Denote by $(x_i, y_i, 0)$ the coordinate of a device v_i with $1 \leq i \leq n$.

Notice that the monitor area may be very large. For example, in an IoT network for monitoring forest fires, the forest area may be tens of square kilometers [16]. Then, the devices are sparsely located in the monitoring area. To form a connected IoT network, a traditional solution is to deploy relay devices. However, since the transmission range between two devices on the ground usually is only tens of meters, a large number of relaying devices need to be deployed to form a connected network, thereby incurring high deployment cost.

In contrast, in this paper we consider the deployment of UAVs to collect data from IoT devices, because UAVs are flexible and no relaying devices need to be deployed, thus saving the deployment cost.

B. Data collection models

We consider two data collection scenarios. The first scenario is referred to as *the data collection without neighborhoods*, that is, a UAV needs to fly to the location of each IoT device to collect its data. One application example of this model is that each device is an RFID tag. To collect the data of the tag, a UAV must be equipped with an RFID reader and the reader can read the data of the tag only when their Euclidean distance is very short, e.g., a few meters. The other scenario is referred to as *the data collection with neighborhoods*, that is, each IoT device can send its data to a UAV in a wireless way, and the UAV can collect the data from the device as long as their Euclidean distance is no greater than a given wireless transmission range. In the following, we introduce the data collection models in the two scenarios, respectively.

Assume that K UAVs are deployed to collect data from the n IoT devices, where the value of K is unknown and will be determined later. The n devices in V need to be partitioned into K disjoint subsets $V_1, V_2, \dots, V_K$, where UAV k collects the data from the devices in V_k, and $1 \le k \le K$. Let $V_k = \{v_1, v_2, \dots, v_{n_k}\}$, where $n_k = |V_k|$. Denote by Δ_i the amount of to-be-collected data of a device v_i in V. Assume that each UAV can collect data from v_i at a data rate b. Then, it takes $\rho(v_i) = \frac{\Delta_i}{b}$ time to collect all data of v_i. Finally, denote by η the flying speed of each UAV.

1) Data collection model in the scenario without neighborhoods: We first introduce the data collection model in the scenario without neighborhoods, where a UAV must fly to the location of each device to collect its data. Assume that UAV k collects the data from devices in V_k in the order of $v_1, v_2, \dots, v_{n_k}$, where $1 \le k \le K$. The data collection flying tour C_k of UAV k is defined as follows. A UAV k first collects the data of device v_1 at the location of v_1, it then flies to collect the data of device v_2, and so on. After collecting the data of the last device v_{n_k}, the UAV finally returns to the location of v_1. That is, the data collection trajectory of UAV k is a *closed tour* $C_k = v_1 \to v_2 \to \dots \to v_{n_k} \to v_1$ with $1 \le k \le K$, where n_k is number of devices in V_k. After collecting the data from each IoT device v_i, UAV k can forward the data to a remote station via 4G/5G communications. Fig. 1(a) shows that 17 IoT devices are deployed in a monitoring area and two UAVs are dispatched to collect the data of the devices.

The total time spent by each UAV k in its flying tour C_k consists of its flying time between IoT devices and the data collection time of devices in V_k. Denote by $w(v_i, v_{i+1})$ the UAV flying time between devices v_i and v_{i+1}, i.e., $w(v_i, v_{i+1}) = \frac{d(v_i, v_{i+1})}{\eta}$, where $d(v_i, v_{i+1})$ is the Euclidean distance between devices v_i and v_{i+1}, and η is the flying speed of UAV k. The flying time in tour C_k then is $\sum_{i=1}^{n_k} w(v_i, v_{i+1})$, where $v_{n_k+1} = v_1$.

The total time $w_1(C_k)$ spent by UAV k in its tour C_k under the scenario without neighborhoods then is

$$w_1(C_k) = \sum_{i=1}^{n_k} w(v_i, v_{i+1}) + \sum_{i=1}^{n_k} \rho(v_i), \qquad (1)$$

where $\rho(v_i)$ is the data collection time of device v_i.

It is important to collect data from devices in V_k as quickly as possible. Otherwise, the collected data will be 'stale'. Denote by B the maximum data collection delay [4]. Then, the total spent time $w_1(C_k)$ of each UAV k in its flying tour C_k must be no greater than B, i.e., $w_1(C_k) \le B$ with $1 \le k \le K$.

2) Data collection model in the scenario with neighborhoods: We then introduce the data collection model with neighborhoods, where a UAV can collect the data of each IoT device when their Euclidean distance is no greater than a given communication range R, e.g., $R = 600$ m [1], [5]

Assume that all UAVs fly at the same altitude h such that the coverage range of each UAV is maximized, where $h < R$ and the optimal altitude h can be obtained from the work in [1], e.g., $h = 300$ m. Denote by $D(v_i)$ the set of locations that a UAV can collect data from a device v_i at altitude h, i.e., $D(v_i) = \{(x, y, h) \mid (x - x_i)^2 + (y - y_i)^2 + (h - 0)^2 \le R^2\}$, where $(x_i, y_i, 0)$ is the coordinate of device v_i. It can be seen that $D(v_i)$ is a disk centered at point (x_i, y_i, h) with a radius $R_0 = \sqrt{R^2 - h^2}$ at altitude h.

Recall that we assumed that each UAV k collects the data of the devices in set V_k in the order of $v_1, v_2, \dots, v_{n_k}$, where $V_k = \{v_1, v_2, \dots, v_{n_k}\}$, $n_k = |V_k|$, and $1 \le k \le K$. The data collection flying tour C_k of UAV k in the scenario with neighborhoods is defined as follows. UAV k first collects data of device v_1 at a location p_1 in $D(v_1)$, it then flies to a location p_2 in $D(v_2)$ and collect the data of device v_2, and so on. After collecting data of the last device v_{n_k} at a location p_{n_k} in $D(v_{n_k})$, the UAV finally returns to the starting location p_1. The flying tour C_k of UAV k can be represented as $C_k = p_1 \to p_2 \to \dots \to p_{n_k} \to p_1$, where p_i is a location in the neighborhood $D(v_i)$ of device v_i with $1 \le i \le n_k$, and $1 \le k \le K$. Fig. 1(b) shows the flying tour of a UAV, where dotted circles represent the neighborhoods of devices.

Similar to Eq. (1), the total time $w_2(C_k)$ spent by UAV k in its flying tour C_k can be calculated as

$$w_2(C_k) = \sum_{i=1}^{n_k} w(p_i, p_{i+1}) + \sum_{i=1}^{n_k} \rho(v_i), \qquad (2)$$

where $w(p_i, p_{i+1})$ is the UAV flying time between locations p_i and p_{i+1}, p_i and p_{i+1} are located in the neighborhoods $D(v_i)$ and $D(v_{i+1})$, respectively, and $\rho(v_i)$ is the data collection time of device v_i.

Notice that the total spent time $w_2(C_k)$ of each UAV k in its flying tour C_k must be no greater than the maximum data collection delay B, i.e., $w_2(C_k) \le B$ with $1 \le k \le K$.

C. Problem definitions

In this paper, we study a novel *minimum UAV deployment problem*, which is to minimize the number of deployed UAVs to collect data from all devices, subject to the constraint that the total time spent by each UAV in its tour is no greater than the maximum data collection delay B. In the following, we consider two variants of the problem under the two data collection models introduced in the previous subsection.

We first formulate a variant of the problem under the data collection model without neighborhoods. Given an IoT network $G = (V, E; w : E \mapsto \mathbb{R}^{\geq 0}, \rho : V \mapsto \mathbb{R}^{\geq 0})$ and a maximum data collection delay B, a UAV needs to fly to the location of each IoT device to collect its data. *The minimum UAV deployment problem without neighborhoods* in G is to determine the minimum number K of UAVs to be deployed, and find their flying tours $C_1, C_2, \ldots, C_K$ to collaboratively collect data from all devices in V, subject to the constraint that the total time $w_1(C_k)$ spent in each tour C_k is no greater than the maximum data collection delay B.

The other variant of the problem under the data collection model with neighborhoods can be formulated similarly. Specifically, given an IoT network $G = (V, E; w : E \mapsto \mathbb{R}^{\geq 0}, \rho : V \mapsto \mathbb{R}^{\geq 0})$, a maximum data collection delay B, and a disk $D(v_i)$ of each device v_i which centers at the location of v_i with a radius R_0 at altitude h, the data of device v_i can be collected by a UAV when the UAV hovers at any location in $D(v_i)$. *The minimum UAV deployment problem with neighborhoods* in G is to determine the minimum number K of deployed UAVs and find their flying tours $C_1, C_2, \ldots, C_K$ to collect the data from all devices, subject to the constraint that the total time $w_2(C_k)$ of each UAV k spent in its tour C_k is no greater than the maximum data collection delay B.

III. Approximation algorithm for the minimum UAV deployment problem without neighborhoods

In this section we deal with the minimum UAV deployment problem without neighborhoods, by proposing a 4-approximation algorithm for it.

A. Algorithm framework

Given an IoT network $G = (V, E; w : E \mapsto \mathbb{R}^{\geq 0}, \rho : V \mapsto \mathbb{R}^{\geq 0})$, an auxiliary complete graph $G' = (V, E; w' : E \mapsto \mathbb{R}^{\geq 0})$ is constructed from G, and the weight of each edge (v_i, v_j) in G' is $w'(v_i, v_j) = w(v_i, v_j) + \frac{\rho(v_i)+\rho(v_j)}{2}$, where $w(v_i, v_j)$ is the flying time between devices v_i and v_j, $\rho(v_i)$ and $\rho(v_j)$ are the durations for collecting data from v_i and v_j, respectively. Following a similar analysis in the work [37], the optimal solutions to the minimum UAV deployment problem in G and G' are equal.

Let $\delta_i = \frac{B}{i}$ with $2 \leq i \leq n$, where δ_i is referred to as an *edge weight threshold*.

The basic idea of the proposed algorithm is that, the algorithm finds a set $\mathcal{C}_i$ of tours that visit nodes in G' based on a given edge weight threshold δ_i, subject to the constraint that the cost of each found tour in $\mathcal{C}_i$ is no greater than B. The final solution $\mathcal{C}$ to the problem then is the set with the minimum number of tours, i.e., $|\mathcal{C}| = \min_{2 \leq i \leq n}\{|\mathcal{C}_i|\}$. The approximation algorithm for the minimum UAV deployment problem without neighborhoods is presented in `Algorithm 1`.

Algorithm 1 Algorithm for the minimum UAV deployment problem without neighborhoods (**approAlgNoNei**)

Input: an IoT network $G = (V, E; w : E \mapsto \mathbb{R}^{\geq 0}, \rho : V \mapsto \mathbb{R}^{\geq 0})$, and a maximum data collection delay B
Output: a set $\mathcal{C}$ of tours to visit all devices in V, such that the total spent time of each tour in $\mathcal{C}$ by a UAV is no greater than B.

Construct a graph $G' = (V, E; w' : E \mapsto \mathbb{R}^{\geq 0})$ from G;
Construct a trivial solution $\mathcal{C} = \{C_1, C_2, \ldots, C_n\}$, where each tour C_i consists of only a single node v_i in V, and $n = |V|$;
for $i \leftarrow 2$ to n **do**
  Let $\delta_i \leftarrow \frac{B}{2}$; /* set an edge weight threshold */
  Invoke `Algorithm 2` to find a set $\mathcal{C}_i$ of tours to visit all nodes in G' based on the edge weight threshold δ_i, subject to that the cost of each tour in $\mathcal{C}_i$ is no greater than B;
  if $|\mathcal{C}_i| < |\mathcal{C}|$ **then**
    Let $\mathcal{C} \leftarrow \mathcal{C}_i$; /* find a better solution */
  end if
end for

B. Algorithm

We here show how to find a set $\mathcal{C}_i$ of tours that visit all devices in G', based on a given edge weight threshold δ_i.

Recall that graph $G' = (V, E; w' : E \mapsto \mathbb{R}^{\geq 0})$ is constructed from G, where $w'(v_i, v_j) = w(v_i, v_j) + \frac{\rho(v_i)+\rho(v_j)}{2}$ for each edge (v_i, v_j) in E.

We first remove the edges with their weights strictly greater than δ_i from G'. Assume that there are q connected components $CC_1, CC_2, \ldots, CC_q$ in the residual graph after the edge removals, where q is a positive integer.

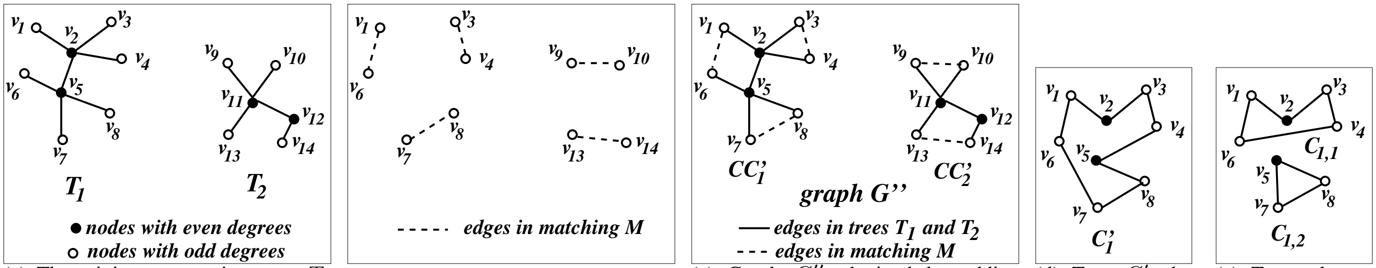
We then find a minimum spanning tree (MST) T_j in each connected component CC_j with $1 \leq j \leq q$, see Fig. 2(a). For each tree T_j, denote by V_j^o the set of odd degree nodes in T_j. Let V^o be the set of odd degree nodes in the q trees $T_1, T_2, \ldots, T_q$, i.e., $V^o = \bigcup_{j=1}^{q} V_j^o$.

We further construct a complete graph $G^o = (V^o, E^o; w^o : E^o \mapsto \mathbb{R}^{\geq 0})$, where there is an edge (u, v) in E^o for any two nodes u and v in V^o and the weight $w^o(u, v)$ of edge (u, v) in G^o is equal to its weight $w'(u, v)$ in G', i.e., $w^o(u, v) = w'(u, v)$. We find a minimum weighted perfect matching M in graph G^o, see Fig. 2(b).

We obtain a graph G'' by adding the edges in matching M to the q trees $T_1, T_2, \ldots, T_q$, i.e., $G'' = M \bigcup (\bigcup_{j=1}^{q} T_j)$, see Fig. 2(c). Assume that there are q' connected components $CC'_1, CC'_2, \ldots, CC'_{q'}$ in G''. Notice that the two endpoints of some edge in M may lie in two different trees. Therefore, the number q' of connected components in G'' is no greater than q, i.e., $q' \leq q$.

For each connected component CC'_j in G'' with $1 \leq j \leq q'$, it can be seen that the degree of each node in CC'_j is even. Then, there is a Eulerian circuit C_j^e in CC'_j [31]. We can obtain a closed tour C'_j that visits each node in connected component CC'_j only once, by shortcutting duplicated nodes in C_j^e, see Fig. 2(d).

We finally split tour C'_j into, say n_j, subtours $C_{j,1}, C_{j,2}, \ldots, C_{j,n_j}$ such that the cost of each subtour is no greater than B and the number n_j of split subtours is no more than $\lceil \frac{w'(C'_j)}{B/2} \rceil$, i.e., $n_j \leq \lceil \frac{w'(C'_j)}{B/2} \rceil$ [14], see Fig. 2(e).



(a) The minimum spanning trees T_1 and T_2 of connected components CC_1 and CC_2, respectively

(b) Matching M

(c) Graph G'' obtained by adding edges in M to trees T_1 and T_2

(d) Tour C_1' derived from the Eulerian circuit of component CC_1' in G''

(e) Two subtours $C_{1,1}$ and $C_{1,2}$ split from tour C_1'.

Fig. 2: An illustration of the execution of `Algorithm` 2.

The detailed algorithm for finding a set $\mathcal{C}_i$ of tours that visit all nodes in G' based on a given edge weight δ_i is presented in `Algorithm 2`.

Algorithm 2 Algorithm for finding a set of tours that visit all nodes in G' based on a given edge weight threshold δ_i

Input: A graph $G' = (V, E; w' : E \mapsto \mathbb{R}^{\geq 0})$, a maximum cost B of each UAV tour, and an edge weight threshold δ_i

Output: A set $\mathcal{C}_i$ of tours so that the cost of each tour in $\mathcal{C}_i$ is no greater than B

Remove the edges with weights greater than δ_i from G'. Assume that there are q connected components $CC_1, CC_2, \ldots, CC_q$ in the residual graph after the edge removals;
Find an MST T_j in each component CC_j with $1 \leq j \leq q$;
Let V^o be the set of odd degree nodes in the q MSTs;
Construct a complete graph $G^o = (V^o, E^o; w^o : E^o \mapsto \mathbb{R}^{\geq 0})$;
Find a minimum weighted perfect matching M in G^o;
Obtain a graph G'' by adding the edges in M to the q trees $T_1, T_2, \ldots, T_q$, i.e., $G'' = M \bigcup (\bigcup_{j=1}^{q} T_j)$. Assume that there are $q'(\leq q)$ connected components $CC_1', CC_2', \ldots, CC_{q'}'$ in G'';
Let $\mathcal{C}_i \leftarrow \emptyset$; /* the set of obtained tours */
for $j \leftarrow 1$ to q' **do**
    Find a Eulerian circuit C_j^e in connected component CC_j' and obtain a tour C_j' visiting nodes in CC_j' by shortcutting duplicated nodes in C_j^e;
    Split tour C_j' into n_j subtours $C_{j,1}, C_{j,2}, \ldots, C_{j,n_j}$ so that the cost of each subtour is no greater than B and $n_j \leq \lceil \frac{w'(C_j')}{B/2} \rceil$;
    Let $\mathcal{C}_i \leftarrow \mathcal{C}_i \cup \{C_{j,1}, C_{j,2}, \ldots, C_{j,n_j}\}$;
end for

C. Algorithm analysis

Lemma 1: Given a complete graph $G' = (V, E)$ and an edge weight function $w' : E \mapsto \mathbb{R}^{\geq 0}$, assume that the edge weights in G' satisfy the triangle inequality. For any closed tour C in G' with $w'(C) \leq B$, there are no more than $i-1$ edges in C with their edge weights strictly greater than $\frac{B}{i}$, where i is a given integer with $i \geq 1$.

Proof: We distinguish into two cases: (1) there are no more than $i-1$ edges in C; and (2) there are no less than i edges in C. Case (1) where C contains no more than $i-1$ edges, the lemma immediately follows. Consider Case (2) where C contains no less than i edges, suppose that there are at least i edges in C with their edge weights greater than $\frac{B}{i}$. Then, the weighted sum $w'(C)$ of edges in C is larger than $i \cdot \frac{B}{i} = B$, which contradicts the assumption $w'(C) \leq B$. The lemma then follows. ∎

Theorem 1: Given an IoT network $G = (V, E; w : E \mapsto \mathbb{R}^{\geq 0}, \rho : V \mapsto \mathbb{R}^{\geq 0})$ and a maximum data collection delay B, there is a 4-approximation algorithm, `Algorithm 1`, for the minimum UAV deployment problem without neighborhoods, which takes time $O(n^4)$, where $n = |V|$.

Proof: Following a similar analysis in the work [37], the values of the optimal solutions to the problem in G and G' are equal. We here only show that `Algorithm 1` delivers a 4-approximate solution to the problem in G', which also is a 4-approximate solution to the problem in G.

Assume that an optimal solution to the problem in G' consists of K^* tours $C_1^*, C_2^*, \ldots, C_{K^*}^*$. We now estimate an upper bound on the number $|\mathcal{C}_i|$ of delivered tours by `Algorithm 2` with an edge weight threshold $\delta_i = \frac{B}{i}$, where $2 \leq i \leq n$. Following `Algorithm 2`, the number $|\mathcal{C}_i|$ of delivered tours is

$$
\begin{aligned}
|\mathcal{C}_i| &\leq \sum_{j=1}^{q'} \lceil \frac{w'(C_j')}{B/2} \rceil \leq \frac{\sum_{j=1}^{q'} w'(C_j')}{B/2} + q, \text{ as } q' \leq q \\
&= \frac{w'(G'')}{B/2} + q, \text{ as } w'(C_j') \leq w'(C_j^e),\ w'(G'') = \sum_{j=1}^{q'} w'(C_j^e) \\
&= \frac{\sum_{j=1}^{q} w'(T_j) + w'(M)}{B/2} + q, \text{ as } G'' = M \bigcup (\bigcup_{j=1}^{q} T_j). \quad (3)
\end{aligned}
$$

In the following, we estimate the weights $\sum_{j=1}^{q} w'(T_j)$ and $w'(M)$, respectively.

Following lemma 1, each of the optimal K^* tours $C_1^*, C_2^*, \ldots, C_{K^*}^*$ contains no more than $i-1$ edges with weights greater than $\frac{B}{i}$. We partition the K^* tours into i groups $\mathcal{C}_0^*, \mathcal{C}_1^*, \ldots, \mathcal{C}_{i-1}^*$, where a tour C_l^* is contained in a group $\mathcal{C}_j^*$ if C_l^* contains exactly j edges with weights greater than $\frac{B}{i}$, where $1 \leq l \leq K^*$. Let $k_j = |\mathcal{C}_j^*|$ with $0 \leq j \leq i-1$. Then, $\sum_{j=0}^{i-1} k_j = K^*$.

We estimate the weights $\sum_{j=1}^{q} w'(T_j)$ and $w'(M)$ as follows, where the proofs of Ineq. (4) and Ineq. (5) are omitted, due to space limitation.

$$
\sum_{j=1}^{q} w'(T_j) \leq \frac{i+1}{i} k_0 B + \sum_{j=1}^{i-1} k_j B - \frac{qB}{i}, \quad (4)
$$

$$
w'(M) \leq \frac{i+2}{2i} k_0 B + \sum_{j=1}^{\lfloor i/2 \rfloor} \frac{i+2j}{2i} k_j B + \sum_{j=\lfloor i/2 \rfloor + 1}^{i-1} k_j B - \frac{qB}{i}. \quad (5)
$$

By combining Inequalities (4) and (5), we have

$$\sum_{j=1}^{q} w'(T_j) + w'(M) \ \le \ (1.5 + 2/i)k_0 B + \sum_{j=1}^{i-1} 2k_j B - \frac{2qB}{i}. \qquad (6)$$

By combining Ineq. (3) and Ineq. (6), we upper bound the number of delivered tours in $\mathcal{C}_i$ as

$$|\mathcal{C}_i| \ \le \ (3 + 4/i)k_0 + 4\sum_{j=1}^{i-1} k_j - \frac{4q}{i} + q. \qquad (7)$$

Consider the case where $i = 4$, we have $|\mathcal{C}_4| \le 4\sum_{j=0}^{3} k_j = 4K^*$, as $K^* = \sum_{j=0}^{3} k_j$.

Recall that the number of tours in $\mathcal{C}$ delivered by `Algorithm 1` is $|\mathcal{C}| = \min_{i=2}^{n}\{|\mathcal{C}_i|\} \le |\mathcal{C}_4| \le 4K^*$. This indicates that $\mathcal{C}$ is a 4-approximation solution.

The analysis of the time complexity is omitted, due to space limitation. The theorem then follows. ■

IV. Approximation algorithm for the minimum UAV deployment problem with neighborhoods

In this section we consider the minimum UAV deployment problem with neighborhoods, where a UAV can collect the data of each IoT device when their Euclidean distance is no greater than a given communication range. We propose a novel approximation algorithm for the problem.

A. Algorithm framework

Given an IoT network $G = (V, E; w : E \mapsto \mathbb{R}^{\ge 0}, \rho : V \mapsto \mathbb{R}^{\ge 0})$, neighborhoods of devices in V, and a maximum data collection delay B, the algorithm framework for the problem with neighborhoods is similar to the one without neighborhoods in the previous section. That is, it finds a set $\mathcal{C}_i$ of tours that visit devices in G based on a given edge weight threshold $\delta_i = \frac{B}{i}$, subject to the constraint that the cost of each tour in $\mathcal{C}_i$ is no greater than B, where $2 \le i \le n$. The final solution $\mathcal{C}$ to the problem then is the set with the minimum number of tours, i.e., $|\mathcal{C}| = \min_{2\le i\le n}\{|\mathcal{C}_i|\}$. The algorithm for the minimum UAV deployment problem with neighborhoods is presented in `Algorithm 3`.

Algorithm 3 Algorithm for the minimum UAV deployment problem with neighborhoods (**approAlgNei**)

Input: a network G, the neighborhood $D(v_i)$ of each device v_i in V, and a maximum data collection delay B

Output: a set $\mathcal{C}$ of tours to visit the neighborhoods of nodes in V, so that the total spent time of each tour in $\mathcal{C}$ is no more than B.

1: Construct a trivial solution $\mathcal{C} = \{C_1, C_2, \ldots, C_n\}$, where each tour C_i consists of only a single node v_i in V, and $n = |V|$;
2: **for** $i \leftarrow 2$ to n **do**
3:     Let $\delta_i \leftarrow \frac{B}{i}$; /* set an edge weight threshold */
4:     Invoke `Algorithm 4` to find a set $\mathcal{C}_i$ of tours to visit neighborhoods of nodes in G based on δ_i, subject to that the cost of each tour in $\mathcal{C}_i$ is no greater than B;
5:     **if** $|\mathcal{C}_i| < |\mathcal{C}|$ **then**
6:         Let $\mathcal{C} \leftarrow \mathcal{C}_i$; /* find a better solution */
7:     **end if**
8: **end for**
9: **return** $\mathcal{C}$.

B. Algorithm

We now show how to find a set $\mathcal{C}_i$ of tours that visit neighborhoods of all devices in G, which is *different* from the one in Section III-B for the problem without neighborhoods, since now the data of each IoT device can be collected by a UAV as long as their Euclidean distance is no more than their communication range.

For any two nodes v_j and v_l in V, recall that their neighborhoods are $D(v_j)$ and $D(v_l)$, respectively. Denote by $c(D_j, D_l)$ *the minimum flying time* between the neighborhoods $D(v_j)$ and $D(v_l)$, which is precisely defined as follows. If the two neighborhoods $D(v_j)$ and $D(v_l)$ overlap with each other, i.e., the Euclidean distance $d(v_j, v_l)$ between nodes v_j and v_l is no greater than $2R_0$, we define $c(D_j, D_l) = 0$, where R_0 is the radius of each neighborhood. On the other hand, if $D(v_j)$ and $D(v_l)$ are disjoint with each other, we define $c(D_j, D_l) = \frac{d(v_j,v_l)-2R_0}{\eta}$, where η is the flying speed of a UAV. Then,

$$c(D_j, D_l) = \begin{cases} 0, & \text{if } d(v_j, v_l) \le 2R_0 \\ \dfrac{d(v_j, v_l) - 2R_0}{\eta}, & \text{if } d(v_j, v_l) > 2R_0 \end{cases} \qquad (8)$$

We partition the IoT devices in V into several disjoint subsets as follows. We first construct an auxiliary graph $G' = (V, E; w' : E \mapsto \mathbb{R}^{\ge 0})$ from G, where the weight $w'(v_j, v_l)$ of each edge (v_j, v_l) in E is the minimum flying time $c(D_j, D_l)$ between neighborhoods $D(v_j)$ and $D(v_l)$, i.e., $w'(v_j, v_l) = c(D_j, D_l)$. We then obtain a graph $G'' = (V, E'')$ from G', by removing the edges with their weights strictly greater than the given edge weight threshold $\delta_i = \frac{B}{i}$ from G'. Assume that there are q connected components $CC_1, CC_2, \ldots, CC_q$ in G'', where q is a positive integer. We partition the devices in V into q disjoint subsets $V_1, V_2, \ldots, V_q$, where V_j is the set of nodes in connected component CC_j with $1 \le j \le q$. Let $V_j = \{v_1, v_2, ..., v_{n_j}\}$, where n_j is the number of nodes in V_j.

Having the q disjoint subsets $V_1, V_2, \ldots, V_q$, we can find an approximate shortest tour C_j to visit all neighborhoods of nodes in each subset V_j, by invoking an existing algorithm for the *Traveling Salesman Problem with Neighborhoods (TSPN)* [9], where $1 \le j \le q$. Assume that $C_j = p_1 \to p_2 \to \cdots \to p_{n_j} \to p_1$, where p_l is a hovering location in the neighborhood $D(v_l)$ of a node v_l with $1 \le l \le n_j$.

Recall that $w_2(C_j)$ represents the total time of a UAV spent in tour C_j, where $w_2(C_j) = \sum_{l=1}^{n_j} w(p_l, p_{l+1}) + \sum_{l=1}^{n_j} \rho(v_l)$, $w(p_l, p_{l+1})$ is the flying time between p_l and p_{l+1}, $\rho(v_l)$ is the duration of data collection from device v_l.

We redefine the weight $w_2'(p_l, p_{l+1})$ of each edge (p_l, p_{l+1}) in C_j as $w_2'(p_l, p_{l+1}) = w(p_l, p_{l+1}) + \frac{\rho(v_l)+\rho(v_{l+1})}{2}$. It can be seen that $w_2'(C_j) = \sum_{l=1}^{n_j} w_2'(p_l, p_{l+1}) = \sum_{l=1}^{n_j}(w(p_l, p_{l+1}) + \frac{\rho(v_l)+\rho(v_{l+1})}{2}) = \sum_{l=1}^{n_j} w(p_l, p_{l+1}) + \sum_{l=1}^{n_j} \rho(v_l) = w_2(C_j)$

We finally obtain, say b_j, subtours $C_{j,1}, C_{j,2}, \ldots, C_{j,b_j}$ from C_j by the tour splitting procedure in [14], such that the cost of each subtour $C_{j,l}$ is no greater than B and the number b_j of obtained subtours is no more than $\lceil \frac{w_2'(C_j)}{B/2} \rceil$, i.e., $b_j \le \lceil \frac{w_2'(C_j)}{B/2} \rceil$.

The set $\mathcal{C}_i$ of obtained tours based on a given edge weight δ_i then is $\mathcal{C}_i = \bigcup_{j=1}^{q}(\bigcup_{l=1}^{b_j} C_{j,l})$. The detailed algorithm for finding a set $\mathcal{C}_i$ of tours that visit neighborhoods of nodes in G based on a given edge weight δ_i is presented in `Algorithm` 4.

Algorithm 4 *Algorithm for finding a set of tours that visit all neighborhoods of nodes in G based on a given edge weight threshold* δ_i

---

Input: A graph $G = (V, E; w : E \mapsto \mathbb{R}^{\geq 0}, \rho : V \mapsto \mathbb{R}^{\geq 0})$, a maximum delay B, and an edge weight threshold δ_i

Output: A set $\mathcal{C}_i$ of tours so that the cost of each tour in $\mathcal{C}_i$ is no greater than B

Construct an auxiliary graph $G' = (V, E; w' : E \mapsto \mathbb{R}^{\geq 0})$ from G, where the weight $w'(v_j, v_l)$ of each edge (v_j, v_l) in E is the minimum flying time $c(D_j, D_l)$ between neighborhoods $D(v_j)$ and $D(v_l)$;
Obtain a graph $G'' = (V, E'')$ from G' by removing the edges with weights greater than δ_i from G'. Assume that there are q connected components $CC_1, CC_2, \ldots, CC_q$ in G''. Denote by V_j the set of nodes in CC_j with $1 \leq j \leq q$.
Let $\mathcal{C}_i \leftarrow \emptyset$; /* the set of obtained tours */
for $j \leftarrow 1$ to q **do**
  Find an approximate tour C_j to visit all neighborhoods of nodes in V_j, by invoking the algorithm in [9] for the TSPN problem;
  Split tour C_j into, say b_j, subtours $C_{j,1}, C_{j,2}, \ldots, C_{j,b_j}$ so that the cost of each subtour is no greater than B and $b_j \leq \lceil \frac{w_2'(C_j)}{B/2} \rceil$;
  Let $\mathcal{C}_i \leftarrow \mathcal{C}_i \cup \{C_{j,1}, C_{j,2}, \ldots, C_{j,b_j}\}$;
end for
return $\mathcal{C}_i$.

C. Algorithm analysis

Theorem 2: Given an IoT network $G = (V, E; w : E \mapsto \mathbb{R}^{\geq 0}, \rho : V \mapsto \mathbb{R}^{\geq 0})$, a maximum data collection delay B, and the radius R_0 of each neighborhood, there is an approximation algorithm, `Algorithm` 3, for the minimum UAV deployment problem with neighborhoods, such that the number of delivered tours is no greater than $(27+108\lambda) \cdot K^* - (67.2\lambda + 12.5)$, where K^* is the minimum number of tours, $\lambda = \frac{r}{B}$, and r is the UAV flying time for a distance of R_0.

Proof: The proof is similar to the one in Theorem 1 of Section III-C, omitted, due to space limitation.

Remark: Notice that the value of λ usually is very small. For example, a DJI Phantom 4 Pro UAV can fly at a speed of $\eta = 10\ m/s$ [22]. Assume that the radius of each neighbourhood is $R_0 = 500$ m [1] and the maximum data collection delay B is 30 minutes. Then, the value of λ is

$$\lambda = \frac{r}{B} = \frac{\frac{R_0}{\eta}}{B} = \frac{\frac{500\ m}{10\ m/s}}{30\ min} = \frac{1}{36}. \quad (9)$$

In this case, the number of tours in $\mathcal{C}$ delivered by `Algorithm 3` can be upper bounded as

$$\begin{aligned} |\mathcal{C}| &\leq (27 + 108\lambda)K^* - 12.5 - 67.2\lambda \\ &\leq 30K^* - 14.36, \text{ where } \lambda = \frac{1}{36}. \end{aligned} \quad (10)$$

Then, we have that $|\mathcal{C}| \leq 30K^* - 15$, as $|\mathcal{C}|$ is an integer. ■

V. Performance Evaluation

A. Simulation Environment

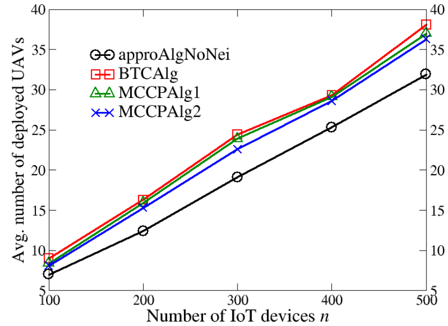
We consider an IoT network area in a 5 km×5 km×1 km three-dimensional Euclidean space. There are from 100 to 500 IoT devices randomly deployed on the ground of the area. The amount of to-be-collected data Δ_i of each IoT device v_i is randomly chosen from an interval from 5 MB to 10 MB [33]. The data transmission rate b is 1 Mb/s [23]. The flying speed η of each UAV is 10 m/s [22]. The maximum data collection delay B ranges from 20 minutes to one hour. On the other hand, when the data of each IoT device can be collected by a UAV with wireless transmissions, we assume that all UAVs hover at an altitude $h = 300$ m and the transmission range R ranges from 400 m to 600 m [1]. Then, the radius of each neighborhood is $R_0 = \sqrt{R^2 - h^2}$.

To evaluate the performance of the proposed algorithm `approAlgNoNei` for the minimum UAV deployment problem *without* neighborhoods, we consider three existing benchmarks. (i) Algorithm `BTCAlg` [13] proposed a 5-approximation algorithm for the problem. (ii) Algorithm `MCCPAlg1` [32] delivered an improved $4\frac{2}{3}$-approximate solution to the problem. (iii) Algorithm `MCCPAlg2` [32] found a $4\frac{4}{7}$-approximate solution, by better refining the algorithm `MCCPAlg1`. Notice that $4\frac{4}{7} < 4\frac{2}{3}$.

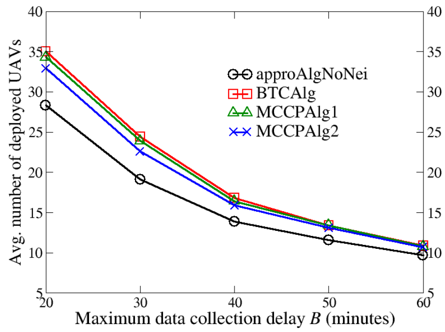
On the other hand, to evaluate the performance of the proposed algorithm `approAlgNei` for the minimum UAV deployment problem *with neighborhoods*, in addition to the three algorithms `BTCAlg`, `MCCPAlg1`, and `MCCPAlg2` that ignore neighborhoods, we compare with another benchmark algorithm `minMaxNei` [7] that takes the neighborhoods into consideration, which is to find a given number K of closed tours to visit the neighborhoods of n nodes such that the longest length among the K found tours is minimized, where at least one location in each neighborhood must be visited by a tour. We find the minimum number K of UAVs deployed by invoking algorithm `minMaxNei` multiple times with different numbers of UAVs, such that the longest tour length is no greater than B.

B. Algorithm performance for the minimum UAV deployment problem without neighborhoods

We first evaluate the performance of different algorithms by increasing the number of IoT devices n from 100 to 500, while fixing the maximum delay B at 30 minutes and the UAV flying speed η at 10 m/s. Fig. 3(a) shows that the average number of UAVs deployed by each algorithm increases with the growth of the number of IoT devices n, as more UAVs need to be deployed in a larger network. Fig. 3(a) also demonstrates that the average number of UAVs by the proposed algorithm `approAlgNoNei` is about from 11% to 19% less than those by the three benchmark algorithms `BTCAlg`, `MCCPAlg1` and `MCCPAlg2`. For example, the average numbers of UAVs

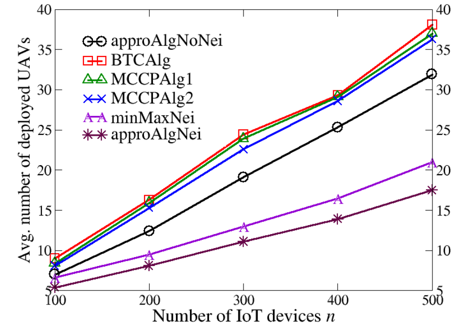


(a) Vary the IoT network size n from 100 to 500, when $B = 30$ minutes and $\eta = 10$ m/s

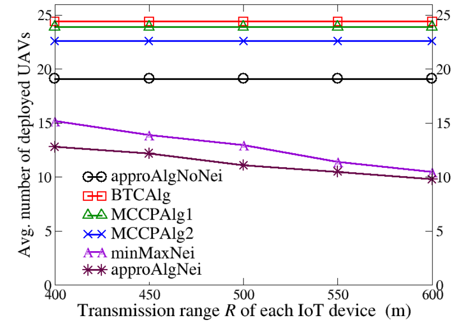


(b) Vary the maximum data collection delay B from 20 minutes to one hour, when $n = 300$ and $\eta = 10$ m/s

Fig. 3: Performance of different algorithms for the minimum UAV deployment problem *without neighborhoods*



(a) Vary the IoT network size n from 100 to 500, when $B = 30$ minutes and $\eta = 10$ m/s



(b) Vary the transmission range R of each IoT device from 400 m to 600 m, when $n = 300$, $B = 30$ minutes and $\eta = 10$ m/s

Fig. 4: Performance of different algorithms for the minimum UAV deployment problem *with neighborhoods*

by algorithms `approAlgNoNei`, `BTCAlg`, `MCCPAlg1` and `MCCPAlg2` are about 19.1, 24.4, 23.9, and 22.6, respectively, when there are $n = 300$ IoT devices in the network.

We then study the performance of different algorithms by varying the maximum data collection delay B from 20 minutes to one hour, when $n = 300$ and $\eta = 10$ m/s. Fig. 3(b) plots the performance of algorithms `approAlgNoNei`, `BTCAlg`, `MCCPAlg1` and `MCCPAlg2`, from which it can be seen that the average number of UAVs deployed by each algorithm decreases quickly with the increase of the maximum data collection delay B. Fig. 3(b) also shows that the average number of UAVs by algorithm `approAlgNoNei` is around from 9.5% to 15.5% less than those by algorithms `BTCAlg`, `MCCPAlg1` and `MCCPAlg2`.

C. Algorithm performance for the minimum UAV deployment problem with neighborhoods

We now evaluate the performance of different algorithms for the minimum UAV deployment problem with neighborhoods. Fig. 4(a) plots the performance of different algorithms by varying the number of IoT devices n from 100 to 500, when $B = 30$ minutes, $R = 500$ m and $\eta = 10$ m/s. It can be seen that the numbers of UAVs by both algorithms `approAlgNei` and `minMaxNei` are much smaller than those by the four algorithms `approAlgNoNei`, `BTCAlg`, `MCCPAlg1` and `MCCPAlg2`, as the former two algorithms take the transmission range of each IoT device into consideration, i.e., a UAV can collect the data of the IoT device when their Euclidean distance is no greater than the transmission range, whereas the latter four algorithms do not consider the transmission range and a UAV needs to fly to the location of the IoT device to collect its data. In addition, Fig. 4(a) shows that the average number of UAVs deployed by the proposed algorithm `approAlgNei` is around from 14.5% to 18% less than that by algorithm `minMaxNei`. For example, the average numbers of UAVs by algorithms `approAlgNei` and `minMaxNei` are about 5.5 and 6.6, respectively, when $n = 100$ IoT devices.

Fig. 4(b) shows the performance of different algorithms by varying the transmission range R of each IoT device from 400 m to 600 m, when $B = 30$ minutes, $n = 300$ and $\eta = 10$ m/s. It can be seen that the numbers of UAVs by both algorithms `approAlgNei` and `minMaxNei` decreases with the growth of the transmission range R, as the flying distance of each UAV is shorter with a larger transmission range R and the saved UAV flying time can be used for collecting data from more IoT devices, thereby deploying less UAVs. In contrast, Fig. 4(b) demonstrates that the numbers of UAVs by the other four algorithms `approAlgNoNei` `BTCAlg`, `MCCPAlg1` and `MCCPAlg2` do not change with the increase of the transmission range, since the four algorithms do not consider the transmission range and each UAV flies to the location of each IoT device to collect its data. Finally, Fig. 4(b) plots that the average number of UAVs by the proposed algorithm `approAlgNei` is around from 6.5% to 15.5% less than that by algorithm `minMaxNei`.

VI. Related work

Some existing studies considered a scenario of dispatching a mobile sink to collect data from sensors in an IoT network. For example, Xu *et al.* [25] studied the problem of dispatching a mobile sink to collect data from sensors such that the lifetime of the network is maximized. Ren *et al.* [23] investigated the problem of using a mobile sink to collect data in a renewable sensor network deployed on a roadside, such that the amount of data collected from all sensors is maximized. On the other hand, there are also some studies focusing on deploying multiple mobile sinks to collect data from sensors. For example, Konstantopoulos *et al.* [15] studied the problem of dispatching multiple mobile sinks to collect sensor data to maximize the data throughput, while ensuring network connectivity and balancing energy consumption among sensors. They proposed algorithms to cluster sensors in a network, find cluster heads, and determine the rendezvous nodes, followed by dispatching mobile sinks to collect data from rendezvous nodes. However, due to various obstacles in the ground such as rocks, rivers and buildings, mobile sinks can not move freely and they may not be able to reach to the locations of some sensors.

There are some recent studies on the deployment of UAVs to gather data in an IoT network. For example, Zhan *et al.* [33] studied the problem of dispatching a UAV to collect data from sensors so as to minimize the maximum energy consumption among all sensors, while ensuring that sensor data are reliably collected. Ebrahimi *et al.* [11] considered the problem of clustering densely-located sensors, constructing a data collection tree for each cluster, and finding a flying trajectory for a UAV to gather data from cluster heads, so that the UAV flying distance is minimized. Zhan *et al.* [34] studied the problem of dispatching a UAV which starts from and ends at a given location so that the number of sensors with their data collected by the UAV within a given duration is maximized. Li *et al.* [17] considered the problem of deploying an energy constrained UAV to collect data in an IoT network in different scenarios, one is that the hovering coverage of different sensors do not overlap with each other, the other is that there are coverage overlapping. They also considered a partially data collection maximization problem. Notice that unlike existing studies, we focused on dispatching the minimum number of UAVs to collect data from an IoT network, subject to that the spent time of each UAV tour is no greater than a given maximum delay upper bound.

We note that there are some studies on the minimum cycle cover problem without neighborhoods, which is to find the minimum number of cycles to cover all nodes in a graph, such that the length of each cycle is no more than a given bound B. For example, Arkin *et al.* [3] first proposed a 6-approximation algorithm for the problem. Khani *et al.* [13] then designed a 5-approximation algorithm. Yu *et al.* [32] recently designed two approximation algorithms for the problem, one has an approximation ratio $4\frac{2}{3}$ while the other has an approximation ratio $4\frac{4}{7}$. In contrast, the proposed algorithm in this paper can deliver a 4-approximate solution. On the other hand, for the single-rooted minimum cycle cover problem with each cycle must contain a root node, Nagarajan and Ravi [21] proposed an $O(\log B)$-approximation algorithm, where B is the length constraint of each tour. Dai *et al.* [6] studied the problem of deploying the minimum number of charging vehicles to fully charge a set of energy-critical sensors, by utilizing the approximation algorithm in [21]. However, the cost of each obtained tour by the algorithms in [6] and [21] may exceed the length bound. In contrast, Zhang *et al.* [37] and Liang *et al.* [19] proposed approximation algorithms for the single-rooted minimum cycle cover problem, such that the length of each obtained tour is no greater than the length bound B.

There are some studies on the Traveling Salesman Problem with Neighborhoods (TSPN) in a 2D Euclidean space, which is to find a single shortest tour such that at least one location in each disk is visited. When the radii of different disks are equal, Adrian *et al.* [8] first proposed a 7.62-approximation algorithm and recently improved the ratio to 6.75 [9]. On the other hand, when the radii of different disks are different, Adrian *et al.* [10] devised a constant factor approximation algorithm. In addition, Deng *et al.* [7] recently studied the problem of finding K closed tours to visit all disks in a 2D space, such that the length among the K found tours is minimized, for which they proposed approximation algorithms.

VII. Conclusion

In this paper, we first formulated the minimum UAV deployment problem, which is to determine the minimum number of to-be-deployed UAVs and find their data collection tours to collect data from IoT devices in a network, subject to that the total time spent in each tour in greater than a maximum data collection delay. We then proposed a 4-approximation algorithm for the problem without neighborhoods, which improved the best approximation ratio $4\frac{4}{7}$ so far. We also designed the first constant factor approximation algorithm for the problem with neighborhoods. We finally evaluated the performance of the proposed algorithms via simulation experiments, and experimental results showed that the number of deployed UAVs by the proposed algorithms are about 11% to 19% less than those by existing algorithms.

Acknowledgement

The work by Wenzheng Xu was supported by the National Natural Science Foundation of China (NSFC) with grant number 61602330, Sichuan Science and Technology Program (Grant No. 2018GZDZX0010 and 2017GZDZX0003). The work by Jian Peng was supported by the National Key Research and Development Program of China (Grant No. 2017YFB0202403) and Sichuan Key Research and Development Program (Grant No. 2018GZ0098). The work by Weifa Liang was supported by Australian Research Council under its Discovery Project Scheme with Grant No. DP200101985.

REFERENCES

[1] A. Al-Hourani, S. Kandeepan and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 569–572, Dec. 2014.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–114, Aug. 2002.

[3] E. M. Arkin, R. Hassin and A. Levin, "Approximations for minimum and min-max vehicle routing problems," *J. Algorithms*, vol. 59, pp. 1–18, 2006.

[4] A. M. Bedewy, Y. Sun, and N. B. Shroff, "The age of information in multihop networks," *IEEE/ACM Trans. Netw.*, vol. 27, no. 3, pp. 1248–1257, 2019.

[5] S. Chandrasekharan, K. M. Gomez, A. Al-Hourani, K. Sithamparanathan, T. Rasheed, L. Goratti, L. Reynaud, D. Grace, I. Bucaille, T. Wirth, and S. Allsopp, "Designing and implementing future aerial communication networks," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 26–34, 2016.

[6] H. Dai, X. Wu, G. Chen, L. Xu and S. Lin, "Minimizing the number of mobile chargers for large-scale wireless rechargeable sensor networks," *Comput. Commun.*, vol. 46, no. 15, pp. 54–65, June 2014.

[7] L. Deng, W. Xu, W. Liang, J. Peng, Y. Zhou, L. Duan, and S. K. Das, "Approximation algorithms for min-max cycle cover problems with neighborhoods," *IEEE/ACM Trans. Netw. (ToN)*, 2020, vol. 28, no. 4, pp. 1845–1858, Aug. 2020.

[8] A. Dumitrescu and J. S. B. Mitchell, "Approximation algorithms for TSP with neighborhoods in plane," *J. Algorithms*, vol. 48, no. 1, pp. 135–159, 2003.

[9] A. Dumitrescu and C. D. Tóth, "The traveling salesman problem for lines, balls, and planes," *ACM Trans. Algorithm (TALG)*, vol. 12, no. 3, article no. 43, 2016.

[10] A. Dumitrescu and C. D. Tóth, "Constant-factor approximation for TSP with disks," *A Journey Through Discrete Math.*, Springer, Cham, pp. 375–390, 2017.

[11] D. Ebrahimi, S. Sharafeddine, P. Ho and C. Assi, "UAV-aided projection-based compressive data gathering in wireless sensor networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1893–1905, Apr. 2019.

[12] Q. Guo, J. Peng, W. Xu, W. Liang, X. Jia, Z. Xu, Y. Yang, and M. Wang, "Minimizing the longest tour time among a fleet of UAVs for disaster area surveillance," *IEEE Trans. Mobile Comput. (TMC)*, DOI: 10.1109/TMC.2020.3038156, to appear.

[13] M. Khani and M. R. Salavatipour, "Approximation algorithms for min-max tree cover and bounded tree cover problems," *Algorithmica*, vol. 69, pp. 443–460, 2014.

[14] S. Khuller, M. Purohit, and K. K. Sarpatwar, "Analyzing the optimal neighborhood: algorithms for partial and budgeted connected dominating set problems," *SIAM J. Discrete Math*, vol. 34, no. 1, pp. 251–270, 2020.

[15] C. Konstantopoulos, G. Pantziou, D. Gavalas, A. Mpitziopoulos and B. Mamalis, "A rendezvous-based approach enabling energy-efficient sensory data collection with mobile sinks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 5, pp. 809–817, May 2014.

[16] M. T. Lazarescu, "Design of a WSN platform for long-term environmental monitoring for IoT applications," *IEEE J. Emerging Select. Topics Circuits Syst.*, vol. 3, no. 1, pp. 45–54, Mar. 2013.

[17] Y. Li, W. Liang, W. Xu, and X. Jia, "Data collection of IoT devices using an energy-constrained UAV," in *Proc. 34th IEEE Int. Parallel Distributed Process. Symp. (IPDPS)*, 2020, pp. 644–653.

[18] Y. Liang, W. Liang, J. Peng, X. Jia, Y. Zhou and L. Duan, "Nonredundant information collection in rescue applications via an energy-constrained UAV," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2945–2958, Apr. 2019.

[19] W. Liang, W. Xu, X. Ren, X. Jia and X. Lin, "Maintaining large-scale rechargeable sensor networks perpetually via multiple mobile charging vehicles," *ACM Trans. Sensor Netw.*, vol. 12, no. 2, pp. 1–26, May 2016.

[20] Y. Lin, T. Wang and S. Wang, "UAV-assisted emergency communications: an extended multi-armed bandit perspective," *IEEE Commun. Lett.*, vol. 23, no. 5, pp. 938–941, May 2019.

[21] V. Nagarajan and R. Ravi, "Approximation algorithms for distance constrained vehicle routing problems," *Netw.*, vol. 59, no. 2, pp. 209–214, 2012.

[22] *Phantom 4 specification.* (2019). [Online]. Available: https://www.dji.com/phantom-4-adv/info

[23] X. Ren, W. Liang and W. Xu, "Data collection maximization in renewable sensor networks via time-slot scheduling," *IEEE Trans. Comput.*, vol. 64, no. 7, pp. 1870–1883, July 2015.

[24] K. T. San, E. Y. Lee and Y. S. Chang, "The delivery assignment solution for swarms of UAVs dealing with multi-dimensional chromosome representation of genetic algorithm," in *Proc. IEEE 7th Annu. Ubiquitous Comput. Electron. Mobile Commun. Conf.*, 2016, pp. 1–7.

[25] X. Xu, J. Luo and Q. Zhang, "Delay tolerant event collection in sensor networks with mobile sink," in *Proc. 29th IEEE Int. Conf. Comput. Commun. (INFOCOM)*, 2010, pp. 1–9.

[26] W. Xu, Z. Xu, J. Peng, W. Liang, T. Liu, X. Jia, and S. K. Das "Approximation algorithms for the team orienteering problem," in *Proc. 39th IEEE Int. Conf. Comput. Commun. (INFOCOM)*, 2020, pp. 1389–1398.

[27] W. Xu , W. Liang, Z. Xu, J. Peng, D. Peng, T. Liu, X. Jia, and S. K. Das, "Approximation algorithms for the generalized team orienteering problem and its applications," *IEEE/ACM Trans. Netw. (ToN)*, DOI: 10.1109/TNET.2020.3027434, to appear.

[28] W. Xu, W. Liang, X. Jia, H. Kan, Y. Xu, and X. Zhang, "Minimizing the maximum charging delay of multiple mobile chargers under the multi-node energy charging scheme," *IEEE Trans. Mobile Comput. (TMC)*, DOI: 10.1109/TMC.2020.2973979, to appear.

[29] 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.

[30] C. You and R. Zhang, "3D trajectory optimization in Rician fading for UAV-enabled data harvesting," *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 3192–3207, June 2019.

[31] W. Yu and Z. Liu, "Improved approximation algorithms for some min-max and minimum cycle cover problems," *Theoretical Comput. Sci.*, pp. 45–58, 2016.

[32] W. Yu, Z. Liu and X. Bao, "New approximation algorithms for the minimum cycle cover problem," *Theoretical Comput. Sci.*, vol. 793, pp. 44–58, 2019.

[33] C. Zhan, Y. Zeng and R. Zhang, "Energy-efficient data collection in UAV enabled wireless sensor network," *IEEE Wireless Commun. Lett.*, vol. 7, no. 3, pp. 328–331, June 2018.

[34] C. Zhan, Y. Zeng and R. Zhang, "Trajectory design for distributed estimation in UAV-enabled wireless sensor network," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 10155–10159, Oct. 2018.

[35] W. Zhang, K. Song, X. Rong and Y. Li, "Coarse-to-fine UAV target tracking with deep reinforcement learning," *IEEE Trans. Automat. Science Eng.*, vol. 16, no. 4, pp. 1522–1530, June 2019.

[36] J. Zhang, X. Wang X, Z. Yu, Y. Lyu, S. Mao and X. Wang, "Robust RFID based 6-DoF localization for unmanned aerial vehicles," *IEEE Access*, vol. 7, pp. 77348–77361, 2019.

[37] Q. Zhang, W. Xu, W. Liang , J. Peng, T. Liu and T. Wang, "An improved algorithm for dispatching the minimum number of electric charging vehicles for wireless sensor networks," *Wireless Netw.*, vol. 25, no. 3, pp. 2689–2705, Dec. 2019.