Energy-Efficient Multi-UAV Data Collection for IoT Networks with Time Deadlines

Oussama Ghdiri, Wael Jaafar, Safwan Alfattani, Jihene Ben Abderrazak, and Halim Yanikomeroglu

Abstract—In this paper, we focus on energy-efficient UAV-based IoT data collection in sensor networks in which the sensed data have different time deadlines. In the investigated setting, the sensors are clustered and managed by cluster heads (CHs), and multiple UAVs are used to collect data from the CHs. The formulated problem is solved through a two-step approach. In the first step, an efficient method is proposed to determine the minimal number of CHs and their best locations. Subsequently, the minimal number of UAVs and their trajectories are obtained by solving the associated capacitated vehicle routing problem. Results show the efficiency of our proposed CHs placement method compared to baseline approaches, where bringing the CHs closer to the dockstation allows significant energy savings. Moreover, among different UAV trajectory planning algorithms, Tabu search achieves the best energy consumption. Finally, the impact of the battery capacity and time deadline are investigated in terms of consumed energy, number of visited CHs, and number of deployed UAVs.

I. INTRODUCTION

The vision of future wireless technologies with numerous smart applications and services has emerged with the Internet of Things (IoT), connecting millions of devices and people. IoT sensor networks are a key enabler for novel applications, including smart cities, connected cars, smart grids, and intelligent transportation. Indeed, the Third Generation Partnership Project (3GPP) has issued recently several standardization documents to support IoT in 5G systems (e.g., TR 23.700-20 and TR 23.700-24). Typically, IoT sensors are low-cost and limited-energy devices with short transmission ranges. In large-scale IoT systems, the deployment of super-sensors or cluster heads (CHs) to aggregate data from neighboring sensors is a practical solution to reduce communication overhead and save sensors' energy. However, even CHs may have a limited communication range, and therefore, a data collector needs to be deployed to gather sensed data and send it to the control center. In that matter, UAVs can be deployed as data collectors [1]. Indeed, UAVs have been presented as an efficient tool to solve several wireless communication challenges [2]–[4]. Accordingly, recent 3GPP standards were issued to regulate different aspects of UAV support in 5G (TR 22.829 and TS 22.125). Due to the agility, strong Line-of-Sight (LoS), and fast deployment of UAVs, the latter are a relevant option for data collection in IoT systems [1]. Nevertheless, UAVs have a limited onboard energy and are sensitive to blockages. Hence, accurate communication channel and power consumption models should be considered when designing data collection missions. On the other hand, large-scale IoT networks encompass sensors with different priorities, criticality levels, and time-sensitive usefulness. Consequently, energy-efficient data collection needs an accurate UAV deployment and trajectory planning to meet time-sensitive constraints, e.g., time deadlines for data collection.

Several works proposed UAV-based sensors' data collection. In our previous work [1], we proposed collecting data from a clustered wireless sensor network using one or several UAVs, aiming to minimize the mission costs in terms of number of clusters and traveled distance by UAVs, while guaranteeing data is collected from all clusters. In this setting, sensor nodes (SNs) were clustered using a K-means-based algorithm, and UAV trajectories designed with different approaches, including genetic algorithm and nearest-neighbor. In [5], the problem of minimizing data collecting time from IoT time-constrained sensors is investigated, where the trajectory of a single collecting UAV and radio resource allocation are optimized. Moreover, the authors of [6] proposed joint optimization of wakeup schedule of SNs and a single UAV's trajectory to minimize the total consumed energy by SNs, while guaranteeing complete data collection from all of them. Finally, authors in [7] presented a data collection framework, where multiple UAVs are deployed to collect data from clustered SNs, aiming to minimize the number of deployed UAVs and their flight times and distances, with respect to a mission deadline. The optimal solution is obtained, which outperforms a benchmark greedy approach. Nevertheless, these works have several limitations. Authors in [1], [5] did not consider energy restrictions of UAVs, while [6] focused only on single UAV trajectory planning. Finally, [7] did not account for energy limitations and air-to-ground accurate channel modeling.

Motivated by the aforementioned points, we propose here a practical data collection approach for large-scale IoT systems, where multiple UAVs collect sensed data with different time deadlines. The proposed solution follows a two-step approach. First, the minimum number of CHs and their locations are determined to guarantee that data from all SNs is aggregated at the CHs. Then, the number of collecting UAVs and their trajectories are optimized to minimize the total consumed energy. Finally, the impact of key parameters, e.g., UAV's battery capacity and data time deadlines, is investigated.

O. Ghdiri and J. Ben Abderrazak are with ESPRIT School of Engineering Tunis, Tunisia, W. Jaafar and H. Yanikomeroglu are with Carleton University, Canada, and S. Alfattani is with the University of Ottawa, Canada, and King Abdulaziz University, Saudi Arabia.

arXiv:2009.06838v1 [eess.SP] 15 Sep 2020

II. SYSTEM MODEL

A. Network and Channel Models

We consider an IoT sensor network consisting of a set $\mathcal{M} = \{1, \ldots, M\}$ of SNs, randomly located in a large area and constantly collecting time-sensitive data. We assume that SNs are heterogeneous, i.e., sensed data by SNs are of different kind, having different importance in time. Indeed, some sensed data are more critical and need to be collected faster than other less-critical data. Due to the limited energy and communication capability of typical IoT sensors, we assume that a set of $\mathcal{K} = \{1, \ldots, K\}$ cluster heads (CHs) can be deployed to collect, aggregate, and transmit sensed data to the data collector. In our system, collection of data is operated by multiple UAVs, which fly and hover above/close to CHs for a sufficient time to collect the aggregated data.

Each sensor transmits its sensed data to its associated CH using power P_{SN}, while a CH communicates with the collecting UAV using power $P_{\text{CH}} > P_{\text{SN}}$. Assuming that IoT sensors communicate with their associated CH using the time-division multiple access (TDMA) protocol, the received signal-to-noise ratio (SNR) at CH c for the signal transmitted by SN s, denoted γ_{sc}, can be written as

$$\gamma_{sc} = \frac{P_{\text{SN}} d_{sc}^{-\alpha}}{\sigma^2}, \forall s \in \mathcal{M}, \ \forall c \in \mathcal{K} \quad (1)$$

where $d_{sc} = \|\mathbf{q}_s - \mathbf{q}_c\|$ is the distance between IoT sensor s and CH c, $\mathbf{q}_i = [x_i, y_i, z_i]$ is the 3D location of node i ($i = s$ or c), α is the path loss exponent, and σ^2 is the noise power. This communication link is considered successful if $\gamma_{sc} \geq \gamma_{\text{th}}$, where γ_{th} is a selected SNR threshold. Hence, a maximum communication range for each SN s associated with CH c is defined as [1]

$$d_{sc} \leq d_{\text{th}} = \left(\frac{P_{\text{SN}}}{\sigma^2 \gamma_{\text{th}}}\right)^{1/\alpha}, \forall s \in \mathcal{A}_c, \ c \in \mathcal{K} \quad (2)$$

where $\mathcal{A}_c$ is the set of SNs associated with CH c.

Given (2), CHs are deployed to collect data from SNs and send it to UAVs. Let $\mathcal{U} = \{1, \ldots, U\}$ be the set of available UAVs for CHs data collection. For the sake of simplicity, we assume that a UAV flies at a fixed altitude H above ground, such that H respects authority regulations and safety considerations. Also, UAVs are equipped with sensors that allow obstacle avoidance at a certain safety distance, denoted d_{safe}, when they fly at maximum speed v_{max}.

When UAV u hovers to receive data from CH c, the air-to-ground channel between them can be expressed using the probabilistic path loss model given by [8]

$$\Lambda_{cu} = Pr_{cu}^{\text{LoS}} l_{cu}^{\text{LoS}} + Pr_{cu}^{\text{NLoS}} l_{cu}^{\text{NLoS}}, \ \forall c \in \mathcal{K}_u, \ \forall u \in \mathcal{U}, \quad (3)$$

where Λ_{cu} is the average path-loss between CH c and UAV u, $\mathcal{K}_u \subset \mathcal{K}$ is the subset of ordered CHs to visit by UAV u, Pr_{cu}^{LoS} is the LoS probability, $Pr_{cu}^{\text{NLoS}} = 1 - Pr_{cu}^{\text{LoS}}$ is the NLoS probability, and l_{cu}^{LoS} and l_{cu}^{NLoS} are LoS and NLoS path-loss, respectively. These parameters are written as [8]

$$Pr_{cu}^{\text{LoS}} = 1 / \left(1 + a e^{b \ (\theta_{cu} - a)}\right), \ \forall c \in \mathcal{K}_u, \ \forall u \in \mathcal{U}, \quad (4)$$

and

$$l_{cu}^m = 20 \log(4\pi f_c / V) + 20 \log(d_{iu}) + \beta_m, \ m = \text{LoS or NLoS} \quad (5)$$

where a and b are constant values determined from the environment (rural, urban, etc.), $\theta_{cu} = \frac{180}{\pi} \times \arcsin\left(\frac{H}{d_{cu}}\right)$ is the elevation angle between UAV u and CH c, f_c is the carrier frequency, V is the speed of light, and β_m is the excessive path-loss coefficient. Using Friis formula, the received power at UAV u from CH c is expressed by

$$P_{cu} = P_{\text{CH}} - \Lambda_{cu}, \ \forall c \in \mathcal{K}_u, \ \forall u \in \mathcal{U}, \quad (6)$$

where P_{CH} is the unit power (in Watt), i.e. equal to 0 dBm.

B. UAV Data Collection Model

We assume that the mission time, i.e., the maximum time for UAVs to collect data from CHs and return to their dockstation, is T_F and divided into N time slots (TSs). Thus, TSs are equal and are of length $\delta = \frac{T_F}{N}$. We define the 3D location of UAV u in time slot t as $\mathbf{q}_u^t = [x_u^t, y_u^t, H]$. Since the UAV's speed is limited by v_{max}, its traveled distance in one TS is constrained by

$$\|\mathbf{q}_u^{t+1} - \mathbf{q}_u^t\| = \sqrt{(x_u^{t+1} - x_u^t)^2 + (y_u^{t+1} - y_u^t)^2} \leq v_{\text{max}}\delta. \quad (7)$$

In our system, each CH c uploads its data to its associated UAV u at rate R_{cu} in bits/second (bps), determined using the Shannon equation as follows:

$$R_{cu} = W \log_2(1 + \gamma_{cu}), \ \forall c \in \mathcal{K}_u, \ \forall u \in \mathcal{U}, \quad (8)$$

where W is the total bandwidth of the channel and $\gamma_{cu} = \frac{\bar{P}_{cu}}{\sigma^2}$ is the SNR, where $\bar{P}_{cu} = 10^{\frac{P_{cu}}{10}}$ is the linear value of P_{cu}. Accordingly, the required number of TSs to upload one packet to the UAV can be expressed by

$$T_{1,cu} = \left\lceil \frac{S_p}{R_{cu}\delta} \right\rceil, \ \forall c \in \mathcal{K}_u, \ \forall u \in \mathcal{U}, \quad (9)$$

where S_p is the size of the packet in bits and $\lceil \cdot \rceil$ is the ceiling function. Given Q_c data packets to be transmitted by CH c, then the required data collection time (in TSs) by UAV c is

$$T_{cu} = Q_c T_{1,cu} \ \forall c \in \mathcal{K}_u, \ \forall u \in \mathcal{U}. \quad (10)$$

Due to the different priorities of collected data at each CH, we define by $T_{d,c}$ the time deadline of data stored in CH c, i.e., if not totally collected within this deadline, the data present in the CH becomes outdated and is dropped[1]. Hence, data collection has to respect the following constraint, assuming that all packets in one CH are collected by only one UAV

$$\sum_{m=1}^{\phi_u(c)} [T_{mu} + T_{f,u}(m-1, m)] \leq T_{d,c}, \ \forall c \in \mathcal{K}_u, \quad (11)$$

where $\phi_u(c)$ designates the rank (i.e. index) of CH c in $\mathcal{K}_u$, T_{mu} is the data collection time of CH ranked m in $\mathcal{K}_u$, and $T_{f,u}(m-1, m)$ is the flight time (in TSs) from the hovering location to collect data from CH ranked $(m-1)$ to that associated to the CH ranked m in $\mathcal{K}_u$, with $m = 0$

---

[1] Typically, this deadline is determined by the most critical sensed data.

designating the dockstation. The flight time of UAV u between two locations $\mathbf{q}_{u,i}$ and $\mathbf{q}_{u,i'}$, where i and i' are two CHs ranks in the UAV path, can be calculated as follows:

$$T_{f,u}(i,i') = \frac{||\mathbf{q}_{u,i'} - \mathbf{q}_{u,i}||}{v_u \, \delta}, \tag{12}$$

with $v_u \leq v_{\max}$ is the flying speed of the UAV (in m/sec).

C. UAV Energy Model

The UAV is energy-constrained due to limited on-board battery. The battery lifetime depends on several factors, e.g., UAV's energy source, type, weight, speed, etc. Typically, the UAV's energy consumption consists of the propulsion energy and the communication energy. Without loss of generality, communication energy is several orders of magnitude smaller than propulsion energy. Hence, it is neglected in the considered energy model. For the propulsion energy, we adopt the propulsion-power model for rotary-wing UAVs of [9]

$$
\begin{aligned}
P_{\text{prop,u}}(v_u) = &\zeta_I \left(\sqrt{1 + \frac{v_u^4}{4v_0^4}} - \frac{v_u^2}{2v_0^2} \right)^{1/2} \\
&+ \zeta_B \left(1 + \frac{3v_u^2}{U_{\text{tip}}^2} \right) + \frac{1}{2} d_0 \psi r A v_u^3,
\end{aligned} \tag{13}
$$

where ζ_B and ζ_I are the blade profile power and induced power, respectively. U_{tip} is the tip speed of the rotor blade, v_0 is the mean rotor induced velocity, ψ is the air density, d_0 is the fuselage drag ratio, r is the rotor solidity, and A denotes the rotor disc area. In order to obtain the power consumption for hovering, we use the same equation above, but with null flying speed $v_u = 0$, thus, we get

$$P_{\text{hov,u}} = P_{\text{prop,u}}(v_u = 0) = \zeta_B + \zeta_I. \tag{14}$$

Using (13)–(14), the consumed energy of UAV u during a time period T can be given by

$$E_u(v,T) = \begin{cases} P_{\text{prop,u}}(v) \times T, & \text{if } v > 0 \\ P_{\text{hov,u}} \times T, & \text{if } v = 0. \end{cases} \tag{15}$$

Hence, the related battery status at TS t, denoted $S_u(t)$, can be expressed by

$$S_u(t) = S_u(t-1) - E_u(v,\delta), \ \forall t > 1, \tag{16}$$

where $S_u(t-1)$ is the battery's status at the end of TS $(t-1)$, and $S_u(0)$ is the initial battery capacity. The latter is expressed as $S_u(0) = S_{\text{ini}} + S_{\min}$, where S_{ini} is the battery capacity dedicated for the mission, while $S_{\min}$ is a safety capacity, reserved for emergency pull back to the dockstation. Hence, $S_u(t) \in [S_{\min}, S_u(0)]$.

III. PROBLEM FORMULATION

In this section, we formulate our optimization problem aiming to minimize the total consumed energy for data collection, by optimizing the deployment of CHs, the number of required UAVs, and their trajectories. Let $K_u = |\mathcal{K}_u|$ be the cardinality

of the set $\mathcal{K}_u$, $\forall u \in \mathcal{U}$. Then, using (15), the consumed energy by UAV u during the mission is given by

$$E_u(\mathcal{K}_u) = \sum_{m=1}^{K_u+1} \left[E_u(0, T_{mu} \, \delta) + E_u(v_u, T_{f,u}(m-1,m) \, \delta) \right], \tag{17}$$

where $T_{f,u}(K_u, K_u+1) = \|\mathbf{q}_{u,K_u} - \mathbf{q}^0\|/v_u\delta$, and $\mathbf{q}^0$ is the initial location of all UAVs, i.e., the dockstation. Therefore, the optimization problem can be formulated as follows:

$$\min_{\substack{\mathcal{K},\mathcal{L},\{\mathcal{A}_c\}_{c\in\mathcal{K}} \\ \mathcal{U},\{\mathcal{K}_u,\mathcal{L}_u\}_{u\in\mathcal{U}}}} \sum_{u=1}^{U} E_u(\mathcal{K}_u) \tag{P1}$$

$$\text{s.t.} \quad d_{sc} \leq d_{\text{th}}, \ \forall s \in \mathcal{A}_c, \forall c \in \mathcal{K}, \tag{P1.a}$$

$$S_u(t) \geq S_{\min}, \ \forall u \in \mathcal{U}, \forall 1 \leq t \leq T_u, \ \forall u \in \mathcal{U} \tag{P1.b}$$

$$\sum_{m=0}^{\phi_u(c)} (T_{mu} + T_{f,u}(m-1,m)) \leq T_{d,c}, \tag{P1.c}$$

$$\|\mathbf{q}_u^t - \mathbf{q}_{u'}^t\| \geq d_{\text{safe}}, \forall t \geq 1, \forall (u,u') \in \mathcal{U}^2, \ u \neq u' \tag{P1.d}$$

$$\|\mathbf{q}_u^{t+1} - \mathbf{q}_u^t\| \leq v_{max}\delta, \ \forall 1 \leq t \leq T_u, \ \forall u \in \mathcal{U}, \tag{P1.e}$$

where $T_u \leq N$ is the effective mission completion time for UAV u (in TSs), $\mathcal{L} = \{\mathbf{q}_c\}_{c\in\mathcal{K}}$ is the set of selected locations for CHs, and $\mathcal{L}_u = \{\mathbf{q}_{u,i}\}_{i\in\mathcal{K}_u}$ is the set of ordered UAV hovering locations to collect data from its associated CHs.

The objective function is the total consumed energy by all UAVs, given their trajectories $\mathcal{L}_u$ associated to their sets of CHs $\mathcal{K}_u$. (P1.a) ensures the successful communication between SNs in $\mathcal{A}_c$ and their associated CH c. (P1.b) guarantees that enough energy is available in the battery to complete the mission at any TS t, while (P1.c) satisfies the time deadline condition when a UAV collects data from CHs. Also, (P1.d) ensures that no collisions between UAVs occur, and finally, (P1.e) limits the flying distance, for a given $v_{\max}$.

Problem (P1) is NP-hard. Indeed, in the special case of already deployed CHs, the problem becomes finding the best routes for UAVs, while respecting the energy and time deadlines conditions. The latter can be seen as the capacitated vehicle routing problem with time windows (CVRPTW) [10]. The CVRPTW can be described as selecting the routes for a number of vehicles, aiming to serve a group of customers within time windows. Each vehicle has a limited capacity, which is used to depart from a depot point, serve a number of customers along its route, then return to the same depot point. The objective of the CVRPTW is to minimize the total transport costs. Logically, the vehicles, customers, and transport costs, can be assimilated by the UAVs, CHs, and energy consumption, respectively. Since the CVRPTW is known to be NP-hard [10], then by restriction, (P1) is NP-hard.

IV. SOLUTION APPROACH

In (P1), we notice that the optimization of parameters $\mathcal{U}$ and $\{\mathcal{K}_u,\mathcal{L}_u\}_{u\in\mathcal{U}}$ directly depends on the selected parameters $\mathcal{K}, \{\mathcal{A}_c\}_{c\in\mathcal{K}}$ and $\mathcal{L}$. Moreover, since their associated constraints are independent, problem (P1) can be divided into two cascaded problems as follows:

1) A first problem of sensors clustering and CHs placement, where only $\mathcal{K}$, $\mathcal{L}$, and $\{\mathcal{A}_c\}_{c \in \mathcal{K}}$ are optimized, aiming to minimize the number of deployed CHs is formulated.[2]

2) Then, given $\mathcal{K}$, $\mathcal{L}$ and $\{\mathcal{A}_c\}_{c \in \mathcal{K}}$, a second problem of UAV trajectory planning is formulated in order to minimize the total consumed energy, through the optimization of $\mathcal{U}$ and $\{\mathcal{K}_u, \mathcal{L}_u\}_{u \in \mathcal{U}}$.

Hence, we present next the two described problems and propose efficient approaches to solve them.

A. IoT Sensors Clustering

Typically, IoT sensors are deployed in large areas in order to sense, process, and communicate relevant data for some applications. Due to their limited battery, IoT sensors need to carefully use their energy, while prolonging the life of collected data as much as possible. To do so, IoT sensors can be grouped into disjoint and non-overlapping clusters to reduce the amount of used energy. SNs scan the surrounding environment and transmit sensed data to CHs, which aggregate and transmit obtained information to the UAVs [11].

Several state-of-the-art clustering techniques exist, e.g., K-means [12], density-based spatial clustering, and hierarchical cluster analysis (HCA) [13]. Nevertheless, due to the low-complexity of K-means, we focus here on its customization, aiming to group SNs and deploy the minimal number of CHs. Thus, the associated clustering problem can be written as

$$\min_{\substack{\mathcal{K}, \mathcal{L}, \\ \{\mathcal{A}_c\}_{c \in \mathcal{K}}}} \quad \sum_{c=1}^{K} \sum_{s=1}^{M} a_{sc}\, d_{sc}^2 \qquad \text{(P2)}$$

$$\text{s.t.} \quad d_{sc} \leq d_{\text{th}}, \forall s \in \mathcal{A}_c, \forall c \in \mathcal{K} \qquad \text{(P2.a)}$$

$$|\mathcal{A}_c| \leq F, \quad \forall c \in \mathcal{K}, \qquad \text{(P2.b)}$$

where $a_{sc} \in \{0, 1\}$ is a binary variable indicating whether SN s is associated with CH c or not, $\forall s \in \mathcal{A}_c$. Constraint (P2.b) guarantees fairness in associating SNs with CHs, where F is the maximum number of SNs associated with one CH. Finally, $|\mathcal{A}_c|$ is the cardinality of the set $\mathcal{A}_c$.

Conventionally, K-means clustering requires a predefined number of CHs. Hence, in [1], the authors proposed a K-means algorithm to solve (P2) without any constraint, then reexecuted it iteratively until (P2.a) is met. Such method is time consuming and inaccurate, as the clustering performance may vary with the algorithm initialization. Therefore, we propose here to improve the method of [1] by integrating both constraints (P2.a)–(P2.b) into the clustering process. The proposed approach is presented in Algorithm 1, and described as follows. First, locations of K CHs are randomly initialized. Then, each SN is assigned to the closest CH with respect to (P2.a)–(P2.b). Next, the locations of CHs are updated by calculating the resulting mean location of associated SNs for each CH. This procedure is repeated until convergence, i.e., the calculated locations remain unchanged. Since we

---

[2]This objective has a direct impact on the consumed energy of data collection since a lower number of CHs would reduce the mission time and energy consumption of UAVs.

aim to deploy the minimal number of CHs, we execute the aforementioned steps for an increasing number of CHs until a solution to (P2) is obtained, i.e., K is determined.

Although the calculated CHs locations are an adequate solution for (P2), this may not be the case for the main problem (P1). In this context, we propose to improve the locations of CHs by making them closer to the dockstation. Indeed, CH c has a mobility margin if its furthest associated SN, denoted s_0, respects (P2.a) loosely, i.e., $d_{s_0 c} < d_{\text{th}}$. Hence, making the CHs closer to the dockstation would increase the probability to respect data collection deadlines, shorten the mission time, and improve the energy consumption of UAVs. This procedure is in lines 32-34 of Algorithm 1.

Algorithm 1 IoT sensors clustering algorithm

1: **Input:** $\mathcal{M}$, $\{\mathbf{q}_s\}_{s \in \mathcal{M}}$, and $\max_{\text{it}}$ %$\max_{\text{it}}$ is the max. number of CHs
2: **Output:** Optimal $\mathcal{K}$, $\mathcal{L}$, and $\{\mathcal{A}_c\}_{c \in \mathcal{K}}$
3: Set $k = (|\mathcal{M}| \div F)$
4: **for** k to $\max_{\text{it}}$ **do**
5:     Set $\mathcal{L}$ randomly %Initial locations of CHs
6:     Set $\mathcal{L}_{old}$ %Set of zeros
7:     Set $\mathcal{A}_c = \emptyset$, $\forall c = 1, \ldots, k$ %c indicates the rank of
8:     CH in $\mathcal{L}$
9:     Set $error = dist(\mathcal{L}, \mathcal{L}_{old})$ %Convergence parameter
10:     **while** $error \neq 0$ **do**
11:         **for** $s \in \mathcal{M}$ **do**
12:             Calculate d_{sc}, $\forall c = 1, \ldots, k$
13:             Find $c_0 = \arg(\min(d_{sc}))$
14:             **if** $d_{sc_0} \leq d_{\text{th}}$ & $|\mathcal{A}_{c_0} \cup \{s\}| \leq F$ **then**
15:                 $\mathcal{A}_{c_0} = \mathcal{A}_{c_0} \cup \{s\}$ %Associate SN s with
16:                 CH c_0
17:             **else**
18:                 $\mathcal{A}_{k+1} = \mathcal{A}_{k+1} \cup \{s\}$ %Put c in set of no
19:                 association
20:             **end if**
21:         **end for**
22:         Set $\mathcal{L}_{old} = \mathcal{L}$
23:         Calculate $\mathcal{L}$ as the mean location of the associated
24:         SNs in $\mathcal{A}_c$, $\forall c = 1, \ldots, k$
25:         Calculate $error = dist(\mathcal{L}, \mathcal{L}_{old})$
26:     **end while**
27:     **if** $\mathcal{A}_{k+1} = \emptyset$ **then**
28:         $\mathcal{K} = \{1, \ldots, k\}$
29:         Break
30:     **end if**
31: **end for**
32: **while** $\max(d_{sc}) < d_{\text{th}}$, $\forall s \in \mathcal{A}_c$ **do**
33:     Get $\mathcal{L}$ closer to the dockstation
34: **end while**

B. Multi-UAV Trajectory Planning

Given $\mathcal{K}$, $\mathcal{L}$, and $\mathcal{A}_c$, $\forall c \in \mathcal{K}$, the trajectory planning problem can be formulated as

$$\min_{\mathcal{U}, \{\mathcal{L}_u, \mathcal{K}_u\}_{u \in \mathcal{U}}} \quad \sum_{u=1}^{U} E_u(\mathcal{K}_u) \qquad \text{(P3)}$$

$$\text{s.t.} \quad \text{(P1.b)} - \text{(P1.e)}.$$

As discussed previously, this problem is NP-hard, and can be assimilated to a CVRPTW problem. In order to solve it, we model our system as a graph, described as follows. Let $G = (\mathcal{D}, \mathcal{E})$ be a complete graph, where $\mathcal{D} = \{0, 1, \ldots, K\}$ is a set of vertices (nodes) representing the dockstation (node

0) and K CHs, and $\mathcal{E}$ is the set of directed edges connecting the nodes. A directed edge from node i to node j, denoted e_{ij}, represents the UAV's flying operation from i to j and the hovering operation at j. A cost associated to the edge e_{ij}, called $\chi_u(e_{ij}) = E_u(v_u, \mu_1(e_{ij})) + E_u(0, \mu_2(e_{ij}))$, is expressed as the sum of the UAV's flying and hovering energy, where $\mu_1(e_{ij}) = T_{f,u}(i,j)\delta$ and $\mu_2(e_{ij}) = T_{ju}\delta$ respectively, $\forall u \in \mathcal{U}$.[3] Moreover, the time deadlines at CHs are represented in the graph by a time window $\omega_c = [o_c, v_c]$, where o_c and v_c are the minimum and maximum instants for data collection, for each node $c \in \mathcal{D}$. This time window defines when data collection at node c can begin and end. Finally, trajectory of UAV u in $\mathcal{G}$ can be defined by $\mathcal{K}_u$, where each element of $\mathcal{K}_u$ is the ordered node to visit. Accordingly, (P3) can be reformulated as problem (P4) detailed below:

$$\min_{\substack{\mathcal{U}, \{\mathcal{K}_u, \\ \mathcal{L}_u\}_{u \in \mathcal{U}}}} \quad \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{D}} \sum_{j \in \mathcal{D}} \chi_u(e_{ij})\, b_{u,ij} \tag{P4}$$

$$\text{s.t.} \quad \sum_{u=1}^{U} \sum_{i \in \mathcal{D}} b_{u,ij} \le 1, \quad \forall j \in \mathcal{D} \tag{P4.a}$$

$$\sum_{j \in \mathcal{D}} b_{u,0j} = \sum_{i \in \mathcal{D}} b_{u,i0} = 1, \quad \forall u \in \mathcal{U} \tag{P4.b}$$

$$\sum_{i=1}^{j-1} \mu_1(e_{i(i+1)}) + \mu_2(e_{i(i+1)}) \in [o_c, v_c], \; \forall j \in \mathcal{K}_u \tag{P4.c}$$

$$\sum_{i \in \mathcal{D}} \sum_{j \in \mathcal{D}} \chi_u(e_{ij}) b_{u,ij} \le S_u(0) - S_{\min}, \; \forall u \in \mathcal{U}, \tag{P4.d}$$

where $b_{u,ij}$ is the binary variable indicating the selection of the edge $i - j$ within UAV u's trajectory. Constraint (P4.a) ensures that each CH is visited at most once by exactly one UAV. (P4.b) guarantees that each UAV departs and returns to the dockstation. Condition (P4.c) emphasizes that the data collection time cannot exceed the time deadline. Finally, (P4.d) guarantees that the consumed energy by any UAV respects the battery capacity. Subsequently, problem (P4) can be solved using any of the available CVRPTW heuristic or metaheuristic approaches, such as gradient descent, simulated annealing, Tabu search, etc. [14].

V. NUMERICAL RESULTS

We assume M=2000 IoT sensors randomly and uniformly distributed within a geographical area of 5×5 km^2. Unless specified otherwise, we use the simulation parameters of Table I, as presented in the next page.

Fig. 1 presents the clustering performance of several algorithms, expressed by the average number of clusters as a function of communication range, d_{th} (averaging over 100 scenarios of SNs). We consider three clustering algorithms, namely proposed Algo. 1, K-means-based clustering in [1] (K-means Algo. [1]), and HCA-based clustering (HCA [13]). As d_{th} increases, a smaller number of CHs is needed, as the latter can be reached by a higher number of SNs. Moreover, Algo.

1 achieves the best clustering performance. Indeed, unlike the other algorithms, Algo. 1 is capable of adjusting the clusters to constraints (P2.a)–(P2.b) on-the-fly, i.e., while processing the SNs.

In Fig. 2, we depict a scenario where Algo. 1 is used to group 2000 SNs into 48 clusters, from which data is collected using 11 UAVs. We distinguish two variations, Algo. 1 as is, and "Algo. 1 (Short)", where lines 32–34 are omitted. It is obvious that Algo. 1 would result in a more energy-efficient data collection, since UAVs would fly for less time, assuming that they hover directly above each CH to collect data.
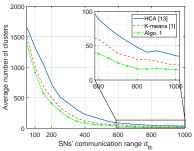
For the clustering design of Fig. 2, Figs. 3–4 compare the consumed energy (in kilojoule), number of visited CHs, and number of deployed UAVs performances, as functions of the battery capacity, for different UAV trajectory planning approaches, namely Tabu search (Tabu), simulated annealing (SA), and guided local search (GLS). Performances of these methods are almost similar, with a preference for Tabu. Indeed, Tabu consumes less or equal energy to SA and GLS. Also, as the battery capacity increases (above 2250 mAh), the performance saturates, guaranteeing that data is collected from all CHs. Finally, the optimal battery capacity of 2500 mAh is provided by Tabu, where minimum energy (=167.9 kJ) and number of UAVs (=12) are achieved.
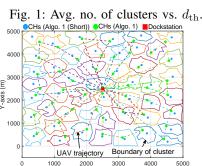
In Figs. 5–6, the same aforementioned performances are depicted as functions of the time deadline, for variations of Algo. 1, including "Algo. 1 (Short)" and "Algo. 1 (Range)", where in the latter, UAVs do not hover directly above CHs, but rather within a range of 150 m from the initial UAV locations defined in Algo. 1. For all approaches, as the deadline increases, the consumed energy reaches a peak then decreases. Below this peak, the deadline is very small such that it prevents collecting data from all CHs as the latter will be outdated, hence, a small number of UAVs is deployed, which consumes low energy due to short trajectories. In contrast, beyond the peak point, the deadline is long enough to deploy few UAVs that visit all CHs. The peak point corresponds to the critical deadline for which the maximum number of UAVs is deployed to collect data from the CHs. "Algo. 1 (Range)" presents the best energy consumption for deadlines above 90 sec, while visiting and deploying similar numbers of CHs and UAVs respectively, as Algo. 1. Indeed, "Algo. 1 (Range)" compensates for the prolonged data collecting time and energy (due to longer distances to CHs) by shorter flying time and energy. For deadlines below 90 sec, Algo. 1 performs either better or similarly to "Algo. 1 (Range)", in terms of energy and number of visited CHs, since hovering exactly above CHs improves the data transmission and thus respects the deadline. Finally, "Algo. 1 (Short)" presents the worst performances as it spends more energy to reach CHs at distant locations, or in contrast, abandon them due to their data becoming outdated.

VI. CONCLUSION

In this paper, we investigated the multi-UAV data collection problem in clustered IoT networks, where sensed data have time deadlines. Aiming to optimize the data collection

---

[3]This is valid assuming that all UAVs are of the same type, i.e., having the same mechanical and communication characteristics.

Fig. 1: Avg. no. of clusters vs. d_{th}.

Fig. 3: Energy vs. battery capacity.

Fig. 5: Energy vs. deadline.

Fig. 2: Clustering and UAV trajectories.

Fig. 4: No. of CHs and UAVs vs. battery capacity.

Fig. 6: No. of CHs and UAVs vs. deadline.

TABLE I: Simulation parameters

UAV altitude	H=100 m	Bandwidth	W=10 MHz
UAV speed	v_u=30 m/sec	Speed of light	V=3 ×10^8 m/sec
Distance threshold	d_{th}=600 m	Mission time	T_f=4 min
Carrier frequency	f_c=2 GHz	Environment parameters	a=9.61 b=0.16
Time deadline	$T_{d,c}$=140 sec	Max SNs per cluster	F=120
Noise power	σ^2=−109 dBm	Packet size	S_p=1 Kbyte
TS length	δ=100 msec	Path loss	α=2.7
Battery capacity	$S_u(0)$=3,500 mAh	Number of packets	Q_c=50,000

deployment costs in terms of energy consumption, we propose a two-step solution. In the first step, the number and locations of CHs, which gather data from associated IoT sensors, are optimized using a customized K-means approach. Subsequently, an energy-efficient data collection framework which uses the minimal number of UAVs is presented; in this framework, the UAV trajectories are defined with respect to time deadlines of collected data and energy constraints. Simulation results show the efficiency of our customized K-means clustering compared to baseline approaches. Also, moving CHs closer to the dockstation provided a significant energy gain. On the other hand, it is shown that Tabu search achieves the best UAV trajectory design, compared to other methods. Finally, the impact of the battery capacity and time deadline is studied in terms of energy consumption, number of visited CHs, and number of deployed UAVs.

REFERENCES

[1] S. Alfattani, W. Jaafar, H. Yanikomeroglu, and A. Yongacoglu, "Multi-UAV data collection framework for wireless sensor networks," in *Proc.*

IEEE Glob. Comm. Conf. (GLOBECOM), Dec. 2019, pp. 1–6.

[2] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-D placement of an unmanned aerial vehicle base station (UAV-BS) for energy-efficient maximal coverage," *IEEE Wireless Commun. Lett.*, vol. 64, no. 6, pp. 434–437, Aug. 2016.

[3] N. Cherif, W. Jaafar, H. Yanikomeroglu, and A. Yongacoglu, "On the optimal 3D placement of a UAV base station for maximal coverage of UAV users," in *Proc. IEEE Glob. Comm. Conf. (GLOBECOM)*, Dec. 2020.

[4] W. Jaafar, S. Naser, S. Muhaidat, P. C. Sofotasios, and H. Yanikomeroglu, "Multiple access in aerial networks: From orthogonal and non-orthogonal to rate-splitting," *ArXiv*, 2020. [Online]. Available: https://arxiv.org/abs/2005.13122

[5] M. Samir, S. Sharafeddine, C. M. Assi, T. M. Nguyen, and A. Ghrayeb, "UAV trajectory planning for data collection from time-constrained IoT devices," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 34–46, Jan. 2020.

[6] 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, Jun. 2018.

[7] A. T. Albu-Salih and S. A. H. Seno, "Energy-efficient data gathering framework-based clustering via multiple UAVs in deadline-based WSN applications," *IEEE Access*, vol. 6, pp. 72 275–72 286, Nov. 2018.

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

[9] Y. Zeng, J. Xu, and R. Zhang, "Energy minimization for wireless communication with rotary-wing UAV," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2329–2345, Mar. 2019.

[10] J. K. Lenstra and A. H. G. R. Kan, "Complexity of vehicle routing and scheduling problems," *Networks*, vol. 11, no. 2, pp. 221–227, 1981.

[11] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. Ann. Hawaii Int. Conf. Syst. Sci.*, Jan. 2000.

[12] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient K-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. and Machine Intelli.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[13] F. Nielsen, *Ch. 8: Hierarchical Clustering - Introduction to HPC with MPI for Data Science.* Springer, 2016.

[14] D. Knuth. Capacitated vehicle routing problem, OR-tools. [Online]. Available: https://developers.google.com/optimization/routing/cvrp