Collect Spatiotemporally Correlated Data in IoT Networks with an Energy-constrained UAV

Wenzheng Xu, Member, IEEE, Heng Shao, Quinli Shen, Jian Peng, Wen Huang, Weifa Liang, Senior Member, IEEE, Tang Liu, Xin-Wei Yao, Tao Lin, and Sajal K. Das, Fellow, IEEE

Abstract—UAVs (Unmanned Aerial Vehicles) are promising tools for efficient data collections of sensors in IoT networks. Existing studies exploited both spatial and temporal data correlations to reduce the amount of collected redundant data, in which sensors are first partitioned into different clusters, a master sensor in each cluster then collects raw data from other sensors and compresses the received data. An energy-constrained UAV finally collects the maximum amount of compressed data from different master sensors. We however notice that the compressed data from only a portion of clusters are collected by the UAV in the existing studies, while the data from other clusters are not collected at all. In this paper, we study a problem of finding a data collection trajectory for an energy-constrained UAV, so that the accumulative utility of collected data is maximized, where the accumulative utility measures the quality of spatiotemporally correlated data collected from different clusters. We propose a novel $\frac{1}{2} + \epsilon$-approximation algorithm for the problem, where $\epsilon$ is a given constant with $\epsilon > 0$. Experimental results with real datasets show that the accumulative utility by the proposed algorithm is at least 23% larger than those by the existing studies, and the number of clusters collected by the proposed algorithm is from 45% to 105% larger than those by the existing studies.

Index Terms—Mobile data collections; UAVs; spatial data correlations; approximation algorithms.

I. INTRODUCTION

The last decade has witnessed the unprecedented explosion of various Internet of Things (IoT) applications, such as smart city monitoring, disaster monitoring, intelligent transportation monitoring, battlefield monitoring, environment monitoring, etc [3], [16], [26]. Since the battery energy of sensors in IoT networks is limited, it is promising to enable sensors to harvest energy from their surrounding environment, e.g., solar energy or wind energy [4], [19], [21].

In this paper, we study the efficient data collections of sensors in an IoT network, where sensors are sparsely placed, e.g., in an area for several square kilometers. Due to the limited transmission range between sensors (e.g., tens of meters), a large number of relay sensors need to be deployed to ensure the network connectivity, which incurs a high deployment cost. In addition, the amounts of harvested energy by sensors usually are limited and thus may not be enough for a large amount of data transmissions [7], [12]. On the other hand, UAVs are promising tools for efficient data collections of sensors, due to their high flexibility, low cost and ease of deployment [10], [13], [14], [15], [27].

Since a UAV is energy-constrained and there may be many sensors in an IoT network, existing studies exploited both spatial and temporal data correlations to reduce the amount of collected data by the UAV [10], [14], [15], [27], where the spatial data correlations mean that sensing data from nearby sensors are highly correlated [24], and the temporal data correlations indicate that sensing data from the same sensor in a short period are also highly correlated [6], [7].

The existing studies [10], [14], [15], [27] exploited a coarse-grained data collection model, in which sensors in an IoT network usually are first partitioned into different clusters, a master sensor in each cluster then collects raw data from other sensors and compresses the received data. An energy-constrained UAV finally collects the maximum amount of compressed data from different master sensors. For example, Fig. 1(a) shows an IoT network consisting of 12 sensors and there are 15 MB to-be-collected raw data in each of the 12 sensors. The sensors are partitioned into three clusters $C_1$, $C_2$, and $C_3$, and there is a master sensor in each cluster, see masters $v_1$, $v_2$, and $v_3$. In Fig. 1(a), the edge weight between the master sensor $v_i$ of cluster $C_i$ (1 $\leq i \leq 3$) and a slave sensor $v_j$ is the spatial data correlation between $v_i$ and $v_j$. For example, the spatial data correlation between sensors $v_1$ and $v_4$ is 0.5, which means that the probability of the difference between the sensing data from $v_1$ and $v_4$ being smaller than a small threshold at any time is 0.5 [7], [14], [15], [27]. Slave sensors in each cluster send their raw data to their master sensor, and the master sensor compresses raw data, e.g., by applying the compressive sensing theory and/or the sparsity-optimization method K-SVD [10]. For example, Fig. 1(b) shows that slave sensors $v_4$, $v_5$, and $v_9$ send their raw data to their master sensor $v_1$ in cluster $C_1$. The master sensor $v_1$ then compresses the 60 MB raw data from the four sensors in the cluster into, e.g., 40 MB data, as there are 15 MB raw data from each of the four sensors in $C_1$. Notice that all the compressed 40 MB data should be collected if the UAV collects data from cluster $C_1$. Otherwise (only a portion of the compressed data is collected), the raw data...
from the four sensors in cluster $C_1$ cannot be recovered from the partially collected compressed data [7], [14], [15], [27]. Since a UAV is energy-constrained and it may not be able to collect compressed data from all the clusters. Fig. 1(b) shows the data collection trajectory found by the existing studies, where only the compressed data from masters sensors $v_1$ and $v_2$ are collected, and there are 40 MB compressed data in each of the three master sensors $v_1$, $v_2$, $v_3$. The rest of the paper is organized as follows. We review related studies in Section II. We introduce preliminaries and the problem definition in Section III. We propose a novel approximation algorithm for the problem in Section IV. We evaluate the performance of the proposed algorithm in Section V, and conclude the paper in Section VI.

II. RELATED WORK

The study on UAV data collections in IoT network has drawn many attentions. Most studies considered neither the temporal correlations nor spatial correlations of sensing data [8], [13], [23], [29], [30]. For example, Li et al. [13] studied a UAV-assisted data collection problem, in which a UAV is
able to collect sensing data from multiple IoT devices within its communication range simultaneously, e.g., MIMO. They proposed an algorithm to maximize the amount of collected data, under the constraint on the UAV energy capacity. Hu et al. [8] found a UAV-assisted data collection tour to minimize the Age of Information (AoI) of collected data, where the UAV can not only charge sensors but also collect their data. Chun et al. [23] devised an algorithm to find the flying tour for a UAV to monitor multiple restricted regions, such that the consumed time in the tour is minimized. Zhan et al. [29] designed a reinforcement learning algorithm for data collection in a UAV-assisted multi-cell cellular network, which balances the operation time of a UAV and the AoI of collected data. Zhang et al. [30] found a flying trajectory for a UAV, so that the UAV energy consumption for collecting data from marine buoys sensors is minimized. Notice that these studies ignored both the spatial and temporal data correlations, and UAVs may collect redundant data from different sensors.

Some studies exploited spatial data correlation to reduce the amount of collected redundant data [14], [15], [27]. Liu et al. [15] adopted a matrix completion theory for UAV data collections, which first chooses some representative sensors, then dispatches a UAV to collect their data, and finally recovers some uncollected data from the collected data. Yu et al. [27] proposed a spatial data collection strategy, where the network is first partitioned into clusters, some slave sensors are chosen to send their data to their master sensor in each cluster, the master sensor forwards the received data to a UAV, and the UAV finally recovers some uncollected data by a denoising autoencoder, based on neural networks. Liu et al. [14] designed a data collection scheme for a mobile sink, where the mobile sink randomly visits a portion of static sensors, collects their data, and recovers sensing data based on the compressive sensing theory. It can be seen that, since these studies ignored temporal data correlations and the collected data by UAVs are still highly redundant, they are not suitable for data collections in large-scale networks.

Only a very few studies focused on both spatial and temporal data correlations, and studied efficient data collections with a UAV [10]. For example, Li et al. [10] proposed a spatially and temporally correlated data aggregation model to reduce data redundancy in UAV-assisted WSNs. Specifically, they first partition sensors into clusters, and slave sensors in each cluster send their data to its master sensor in the cluster. Since the data received from slave sensors and from itself are not only spatially-correlated but also temporally-correlated, the master sensor then compresses data by applying the compressive sensing theory and a sparsity-optimization method K-SVD. The UAV finally collects compressed data from masters. Although these studies exploited both spatial and temporal data correlations, they only used a coarse-grained data collection model, in which all compressed data in each cluster must be collected to recover the original raw data, if the UAV collects data from the cluster, in this paper we adopt a light-weight fine-grained data collection model in which compressed data can be partially collected, thereby increasing the diversity of collected data.

There are also some studies that considered both spatial and temporal data correlations, but UAVs or mobile sinks are not employed to collect data. Instead, the data of each sensor is sent to a base station directly or via the relay of other sensors. For example, Guo et al. [7] devised an fine-grained clustering method based on spatial data correlations, and designed a routing protocol to maximize the utility of collected spatiotemporally-correlated data. Fattoum et al. [6] proposed an adaptive sampling approach, which minimizes sampling rates of sensor nodes while ensuring a high quality of collected data. Xie et al. [24] chose some important sampling points to collect data and estimated the data of not-sampled points from the collected points, based on spatial data correlations. However, it is unknown how to extend the proposed algorithms for UAV data collections.

### III. Preliminaries

In this section, we introduce the network model, spatial data correlation model, channel model, UAV data collection framework, UAV energy consumption model, and define the problem precisely.

#### A. Network model

We consider an IoT network deployed in a critical area, which is used for, e.g., smart city monitoring or environmental monitoring. Assume that there are $n$ sensors in the network with $n \geq 1$. Let $V$ be the set of the $n$ sensors, i.e., $V = \{v_1, v_2, \ldots, v_n\}$. Denote by $(x_i, y_i, 0)$ the coordinate of a sensor $v_i$ with $1 \leq i \leq n$. Assume that the locations of the $n$ sensors are known in the phase of network deployment. Each sensor $v_i$ in $V$ is powered by a rechargeable battery with a capacity of $B_i$, and it can harvest energy from its surrounding environment, e.g., solar energy or wind energy.

In this paper, for the sake of simplicity, we study the employment of a single UAV to collect data from the sparsely-located sensors, which is applicable to a small-scale network, e.g., one square kilometer. In case that the area of the network may be very large (e.g., tens of square kilometers), we need to employ multiple UAVs. The proposed algorithm in this paper can be easily extended to the case with multiple UAVs, e.g., finding the data collection tours of multiple UAVs one by one in a greedy manner, by applying the strategy in work [25].

We consider the data collection of the IoT network, e.g., for one month. For every fixed period $T$, e.g., one day, the UAV is dispatched to collect sensor data. The period $T$ is divided into equal-length time slots, where the duration of each time slot is short, e.g., 5 min. Each sensor $v_i$ generates data with a rate $r_i(t)$ at time slot $t$ with $1 \leq t \leq T$.

Denote by $D_i^R$ the amount of residual data in sensor $v_i$ at the beginning of period $T$. The amount of to-be-collected data in sensor $v_i$ at the end of period $T$ is $\min\{D_i^R + \sum_{t=1}^{T} r_i(t), D_i^S\}$, where $r_i(t)$ is the data rate of sensor $v_i$ at time slot $t$ and $D_i^S$ is the storage capacity of sensor $v_i$. Denote by $\hat{r}_i$ the predicted average data rate of sensor $v_i$ in period $T$, for example, its past average data rate before the period $T$. The predicted amount of to-be-collected data in sensor $v_i$ at the end of period $T$ then is $D_i^{SC} = \min\{D_i^R + \hat{r}_i T, D_i^S\}$. 

Authorized licensed use limited to: CITY UNIV OF HONG KONG. Downloaded on March 24,2024 at 01:10:51 UTC from IEEE Xplore. Restrictions apply.
Each sensor can harvest energy from its surrounding environment, such as solar energy or wind energy. Although environmental conditions, e.g., weather condition, affect the harvesting rate, it is predictable by adopting a machine learning-based prediction algorithm, e.g., the NARNET model in [20]. Denoted by \( e_H(v_i, t) \) the amount of harvested energy by sensor \( v_i \) at time slot \( t \). Also, denote by \( e_R(v_i, t) \) the amount of residual energy of sensor \( v_i \) at the beginning of time slot \( t \). It can be seen that the amount \( P_C(v_i, t) \) of consumed energy by sensor \( v_i \) at time slot \( t \) should be no greater than the sum of the amount \( e_R(v_i, t) \) of residual energy at the beginning of time slot \( t \) and its harvested energy \( e_H(v_i, t) \) at time slot \( t \), i.e., \( P_C(v_i, t) \leq e_R(v_i, t) + e_H(v_i, t) \). The amount of residual energy of sensor \( v_i \) at the end of period \( T \) then is \( e_{Budget}^B = \min\{e_R(v_i, 0) + \sum_{t=1}^{T} e_H(v_i, t) - \alpha v_i T, B_i \} \), where \( e_R(v_i, 0) \) is the amount of residual energy of sensor \( v_i \) at the beginning of period \( T \), \( \alpha v_i T \) is its amount of consumed energy for data sensing in period \( T \), and \( B_i \) is its battery capacity.

On the other hand, each sensor consumes its energy on data sensing, data transmission, and data reception. Following the exiting work in [11], we model the energy consumptions \( P_S(v_i, t) \), \( P_{Tx}(v_i, t) \), \( P_{Rx}(v_i, t) \) of sensor \( v_i \) on data sensing, data transmission and data reception at time slot \( t \) as follows. \( P_S(v_i, t) = \alpha \cdot r_i(t) \), \( P_{Tx}(v_i, t) = (\beta_1 + \beta_2 P_i^j d^2(v_i, u_i)) \cdot f_t \), and \( P_{Rx}(v_i, t) = \gamma \cdot \sum_{j \in N(v_i)} f_{jit} \), where \( \alpha, \beta_1, \beta_2 \) and \( \gamma \) are constants [11], \( r_i(t) \) is the data rate of sensor \( v_i \), \( d(v_i, u_i) \) is the Euclidean distance between sensor \( v_i \) and the UAV at time slot \( t \), and \( f_{jit} \) is the data transmission rate from sensor \( v_i \) to the UAV at time slot \( t \). \( f_{jit} \) is the reception rate of sensor \( v_i \) from its neighbour \( v_j \) at time slot \( t \) and \( N(v_i) \) is the set of neighbors of sensor \( v_i \).

### B. Spatial data correlation model

It is recognized that nearby sensors may generate similar sensing data. Following the study in [7], we can measure the spatial data correlation between sensors \( v_i \) and \( v_j \) as

\[
\text{c}(v_i, v_j) = \frac{1}{N_{ij}},
\]

where \( N_{ij} \) is the number of time slots so that the difference between the data generated by sensors \( v_i \) and \( v_j \) is no more than a small threshold in the past \( N \) time slots, e.g., \( N = 100 \). Notice that the value of \( N_{ij} \) can be calculated with historical data [7].

To collect nonredundant information from the \( n \) sensors in the network, we partition the \( n \) sensors into several clusters, and each cluster consists of a master sensor and some slave sensors. Then, the data collected from slave sensors should be nonredundant with the data collected from the master sensor. By doing so, less redundant data will be collected from each cluster and the energy-constrained UAV is able to collect nonredundant data from more clusters.

We apply the method in [7] to partition the \( n \) sensors into, e.g., \( Q \) clusters \( C_1, C_2, \ldots, C_Q \), and find a master sensor \( v_i \) for each cluster, where the value of \( Q \) is determined by the method. Specifically, let \( v_i \) be the master sensor of cluster \( C_i \). Denote by \( D_{max}^\text{opt} \) the maximum amount of potentially collected data from sensor \( v_i \) with its residual energy and harvested energy in the period \( T \) [7], where

\[
D_{max}^\text{opt} = \min\{R_i^\text{max} \cdot e_{Budget}^B, D_i^C \},
\]

\( R_i^\text{max} \) is the maximum data transmission rate from sensor \( v_i \) to the UAV, \( e_{Budget}^B \) is its energy budget, \( P_i^\text{t} \) is its transmission power, and \( D_i^C \) is its amount of to-be-collected data.

The amount of suppressed redundant data transmission by a slave sensor \( v_j \) in cluster \( C_i \) is \( D_{ij}^\text{max} \cdot c(i, v_j) \), and the total amount of suppressed redundant data transmission in the network is \( \sum_{j=1}^{Q} \sum_{v_j \in C_i} D_{ij}^\text{max} \cdot c(i, v_j) \). The redundant data suppression maximization problem is to partition the \( n \) sensors into \( Q \) clusters, e.g., \( C_1, C_2, \ldots, C_Q \), and find a master sensor \( v_i \) in each cluster \( C_i \), such that the amount of suppressed redundant data transmission is maximized, where the value of \( Q \) is to be determined. Note that there is a \((0.5 - \varepsilon)\)-approximation algorithm in [7] for the redundant data suppression maximization problem, where \( \varepsilon \) is a given constant with \( 0 < \varepsilon < 0.5 \).

### C. Channel model

Consider a ground sensor \( v_i \) and the UAV that hovers at a location \( p_j \) in the air. Following the study in [18], the average signal-to-noise ratio \( SNR_{ij} \) is \( SNR_{ij} = P_{i,j}^{\text{LoS}} S N R_{ij}^{\text{LoS}} + P_{i,j}^{\text{NLoS}} S N R_{ij}^{\text{NLoS}} \), where \( P_{i,j}^{\text{LoS}} \) is the probability of Line-of-Sight (LoS) link between the sensor and the UAV, and its value can be calculated by the method in [1], \( P_{i,j}^{\text{NLoS}} = 1 - P_{i,j}^{\text{LoS}} \) is the probability of Non-Line-of-Sight (NLoS) link, \( S N R_{ij}^{\text{LoS}} \) and \( S N R_{ij}^{\text{NLoS}} \) are the signal-to-noise ratios of LoS and NLoS links, respectively. Specifically, \( S N R_{ij}^{\text{LoS}} = P_i^t \left( \frac{\epsilon}{\pi f c d(v_i, p_j)} \right)^2 \frac{1}{\eta_{\text{LoS}}} \) and \( S N R_{ij}^{\text{NLoS}} = P_i^t g_i^t \left( \frac{\epsilon}{\pi f C d(v_i, p_j)} \right)^2 \frac{1}{\eta_{\text{NLoS}}} \), where \( P_i^t \) is the transmission power of the sensor \( v_i \), \( g_i^t \) is its antenna gain, \( P_N \) is the power of Gaussian white noise, \( c \) is the speed of light, \( f_c \) is the frequency of the radio carrier (e.g., \( f_c = 2.4 \text{ GHz} \)), \( d(v_i, p_j) \) is the Euclidean distance between sensor \( v_i \) and location \( p_j \), \( \eta_{\text{LoS}} \) and \( \eta_{\text{NLoS}} \) are the average shadow fadings for the LoS and NLoS links, respectively, e.g., \( \eta_{\text{LoS}} = 1 \text{ dB} \) and \( \eta_{\text{NLoS}} = 20 \text{ dB} \) in an urban environment [1]. Then, the data collection rate \( R_{ij} \) from sensor \( v_i \) to the UAV at location \( p_j \) is \( R_{ij} = W \cdot \log_2 (1 + S N R_{ij}) \), where \( W \) is the bandwidth allocated to the sensor \( v_i \) by the UAV, e.g., 20 MHz.

### D. UAV data collection framework

At the end of every fixed period \( T \), e.g., one day, a UAV is dispatched to quickly collect data from sensors in the network. The UAV is located at a depot \( r \) initially. The UAV departs from the depot, flies along a planned trajectory for the data collection task, and returns to the depot to replenish its energy after completing the task. The UAV hovers at an optimal height \( h \) to collect sensor data, e.g., \( h = 300 \text{ m} \), as the data transmission rate from a sensor to the UAV at a higher or a lower height is smaller than the optimal height \( h \) [1].

Due to the limited energy capacity of the UAV, it may not be able to collect data from all sensors. Instead, it collects only nonredundant information from the sensors. Assume that the UAV collects data from \( Q' \) of \( Q \) clusters with \( Q' \leq Q \), and \( Q' \)
is to be determined later. For the sake of convenience, we assume that the UAV collects data from clusters $C_1, C_2, \ldots, C_Q$ one by one. For example, Fig. 1(c) in Section I shows that the network is divided into $Q = 3$ clusters, and UAV collects data from clusters $C_1, C_2,$ and $C_3$ one by one, where $Q' = Q = 3$.

Denote by $P_r = r \rightarrow p_1 \rightarrow p_2 \rightarrow \cdots \rightarrow p_{Q'} \rightarrow r$ the UAV data collection trajectory, where $p_i$ is the hovering location above the master sensor $v_i$ in cluster $C_i$ at height $h$. In other words, assume that the coordinate of the master sensor $v_i$ is $(x_i, y_i, 0)$, the UAV hovering location $p_i$ for collecting data from sensors in cluster $C_i$ is then $(x_i, y_i, h)$. The rationale behind the assumption that the UAV hovers above the master sensor $v_i$, for data collections in cluster $C_i$ is that, the data from the master sensor is most representative and more data should be collected from the master sensor than another slave sensor in the cluster, and the data transmission rate from the master sensor to the UAV is maximized when the UAV hovers above the sensor, e.g., an LoS link [1].

For any cluster $C_i$, the UAV may collect data from both the master sensor $v_i$ and slave sensors in the cluster. When the UAV hovers above cluster $C_i$, the master sensor $v_i$ first sends its data to the UAV, and every slave sensor $v_j$ in $C_i$ overhears the transmitted data by $v_i$ at the same time. Slave sensor $v_j$ thus knows its similar data with the transmitted data by the master sensor $v_i$, and then sends its nonredundant data to the UAV. Denote by $t_i$ and $t_{ij}$ the data collection durations allocated by the UAV to the master sensor $v_i$ and a slave sensor $v_j$ in cluster $C_i$, respectively. Then, the amounts of collected data from the master sensor $v_i$ and the slave sensor $v_j$ in cluster $C_i$, respectively, are data transmission rates from the master sensor $v_i$ and the slave sensor $v_j$ to the UAV, respectively.

Due to some spatial data similarity $c(v_i, v_j)$ between the master sensor $v_i$ and the slave sensor $v_j$, the amount $D_{ij}$ of obtained data by the slave sensor $v_j$ is the sum of the amount of collected data from $v_j$ itself and the similar data collected from $v_i$, i.e., $D_{ij} = D_j + D_i \cdot c(v_i, v_j)$.

It can be seen that, in the proposed data collection model, each sensor is required to have very limited computing ability. On one hand, the sensing data generated by each sensor usually is stored in its memory in order of their sensing time. Then, the transmitted data by the master sensor of each cluster also is in order of their sensing time. On the other hand, when a slave sensor overhears the transmitted data from its master sensor, it can remove its redundant data with the overhead data, by simply performing numerical comparisons. Notice that many existing studies required that a sensor has some processing ability to remove redundant data, e.g., [10], [31], and the cost of such a sensor usually is small, e.g., no more than 10 dollars. For example, consider a sensor consists of a DHT22 sensing unit, an ESP8266 WiFi SoC, and an STM32F103C8T6 micro CPU, where the DHT22 sensing unit is able to sense temperature and humidity data, the ESP8266 WiFi SoC can transmit and overhear data, and the STM32F103C8T6 micro CPU can read from and write to the memory, compare and remove similar data by performing only about 10 lines of codes. Of course, it is desirable to have less requirements on sensors to remove redundant data, and this is one of our future work.

E. UAV energy consumption model

The UAV consumes its energy on flying and data collections. Recall that the UAV data collection trajectory is $P_r = r \rightarrow p_1 \rightarrow p_2 \rightarrow \cdots \rightarrow p_{Q'} \rightarrow r$, where $p_i$ is the hovering location above the master sensor $v_i$ in cluster $C_i$ at height $h$. For each cluster $C_i$, its total data collection time is $t(C_i) = t_i + \sum_{v_j \in C_i \setminus \{v_i\}} t_{ij}$, where $t_i$ and $t_{ij}$ are data collection durations of master sensor $v_i$ and a slave sensor $v_j$ in cluster $C_i$, respectively. The amount of energy consumption for data collection in cluster $C_i$ then is $t(C_i) \cdot \xi$, where $\xi$ is the UAV energy consumption rate for hovering and data collection.

On the other hand, during the flying from locations $p_i$ to $p_{i+1}$ in the data collection trajectory $P_r$ with $0 \leq i \leq Q'$, the UAV first accelerates from zero to a speed $\nu_{\max}$ (e.g., 10 m/s) with an acceleration speed, then flies with the speed $\nu_{\max}$ towards location $p_{i+1}$, and finally decelerates from $\nu_{\max}$ to zero when the UAV arrives at $p_{i+1}$. Following existing studies in [22], [28], the UAV flying energy consumption $e_{flying}^i$ from locations $p_i$ to $p_{i+1}$ then is $e_{flying}^i = e_{\text{acc}}^i + e_{\text{dec}}^i + e_{\text{cap}}^i$, where $\eta$ is the energy consumption rate at speed $\nu_{\max}$, $d_{i,i+1}$ is the Euclidean distance from locations $p_i$ to $p_{i+1}$, $e_{\text{acc}}^i$ and $e_{\text{dec}}^i$ are the flying distances for acceleration and deceleration, respectively, $e_{\text{cap}}^i$ and $e_{\text{dec}}^i$ are its energy consumptions for acceleration and deceleration, respectively. The energy consumption of the UAV in its data collection trajectory $P_r$ then is $w(P_r) = \sum_{i=1}^{Q'} \left( t(C_i) \cdot \xi + \sum_{j=0}^{Q'} e_{flying}^{i,j} \right)$. Notice that the UAV energy consumption $w(P_r)$ should be no more than its battery capacity $e_{\text{cap}}^P$. 

F. Problem definition

Since sensing data is not only spatially-correlated but also temporally-correlated, we introduce a utility function $f(\cdot)$ to measure the quality of spatiotemporally-correlated data [7], where the utility function $f(\cdot)$ has a diminishing return property. Assume that function $f(\cdot)$ is increasing, twice-differentiable, and strictly concave, e.g., $f(x) = \log_2(1 + x)$. For a cluster $C_i$, the accumulative utility of collected data from its master sensor $v_i$ and slave sensors is $f(D_i) + \sum_{v_j \in C_i \setminus \{v_i\}} f(D_{ij}) = f(D_i) + \sum_{v_j \in C_i \setminus \{v_i\}} f(D_{ij}) + c(v_i, v_j) \cdot f(D_i)$. The accumulative utility of collected data from the $Q$ clusters of the network then is

$$\sum_{i=1}^{Q}(f(D_i) + \sum_{v_j \in C_i \setminus \{v_i\}} f(D_{ij})).$$

In this paper, we consider a data collection utility maximization problem as follows. Given the amounts of to-be-collected data $D_i^\text{budget}$ and residual energy $e_{\text{cap}}^P$ of each sensor $v_i$, with $1 \leq i \leq n$, assume that the $n$ sensors have been partitioned into $Q$ clusters $C_1, C_2, \ldots, C_Q$ (see Section III-B). The problem is to find a data collection flying trajectory $P_f$ for a UAV, and calculate the data collection durations $t_i$ and $t_{ij}$ for the master sensor $v_i$ and each slave sensor $v_j$ in each cluster $C_i$, respectively, so that the accumulative utility $\sum_{i=1}^{Q}(f(D_i) + \sum_{v_j \in C_i \setminus \{v_i\}} f(D_{ij}))$ of collected data from
sensors in the $Q$ clusters is maximized, where the energy consumption of the UAV in the trajectory $P_t$ is no greater than its energy capacity $e^{CAP}$.

Notice that if the data spatial correlation $c(v_i, v_j)$ between any two sensors is zero, each cluster $C_1$ consists of only the master sensor $v_i$ in the solution delivered by the algorithm [7]. The number $Q$ of clusters then is equal to the number $n$ of sensors. The accumulative utility of collected data from sensors in the network is

$$
\sum_{i=1}^{n} f(D_i),
$$

where $D_i$ is the amount of collected data from sensor $v_i$. By utilizing the utility function $f(.)$, we are able to collect data from more sensors, since the data from the same sensor are temporally-correlated. For example, consider two data collection strategies. The first strategy is less than that in the second strategy, i.e.,

$$
f(10) + f(0) = \log_2(1+10) + \log_2(1+0) = 3.46 < f(5) + f(5) = 2\log_2(1+5) = 5.17.
$$

G. A related problem

We introduce a related problem – the orienteering problem. Given an undirected and weighted graph $G(V \cup \{r\}, E)$ and a length budget $L_{max}$, there is a path $v(u, v)$ associated with each edge $(u, v)$ in $E$, where $r$ is the root and edge weights in $G$ satisfy the triangle inequality. The problem is to find an $r$-rooted closed tour with its length no greater than $L_{max}$, so that the number of nodes visited in the tour is maximized. Notice that there is a $\frac{1}{2}$-approximation algorithm for the problem [2], which will serve as a subroutine of the proposed algorithm in this paper.

IV. APPROXIMATION ALGORITHM

In this section, we propose a novel $\frac{1}{2\log\epsilon}$-approximation algorithm for the data collection utility maximization problem, where $\epsilon$ is a given constant with $\epsilon > 0$.

A. Algorithm basic idea

The basic idea behind the proposed algorithm is to transform the data collection utility maximization problem to the orienteering problem. There are however two major differences between the two problems. The first difference is that we do not know the data collection utility of each cluster in the former problem since the data collection duration of each sensor in the cluster is unknown, while the utilities of visiting different nodes in the latter problem are given and uniform, i.e., equal to one. The second difference is that both edge weights (i.e., the UAV flying energy consumption) and node weights (i.e., data collection energy consumption) are taken into consideration in the UAV tour of the former problem, whereas only edges weights are counted in the found tour of the latter problem.

We devise a series of smart graph transformations to address the differences. Specifically, in the first graph transformation, we transform each cluster into several virtual nodes so that each virtual node is associated with a node weight (i.e., UAV data collection energy consumption) and a profit (i.e., the utility of visiting the virtual node). In the second graph transformation, we transform the graph into an only edge-weighted graph. In the last transformation, we transform each virtual node into several nodes with uniform profits, through a novel scaling technique.

B. First graph transformation

In the first graph transformation, we transform each cluster $C_i$ into several virtual nodes so that each virtual node is associated with a node weight (i.e., UAV data collection energy consumption) and a profit (i.e., the utility of visiting the virtual node). For each sensor $v_j$ in a cluster $C_i$, its maximum data collection time is $t_j^{max} = \min\{\frac{D_j^C}{P_j}, \frac{e_j^{Budget}}{P_j}\}$, where $D_j^C$ is its amount of to-be-collected data, $R_j$ is its data transmission rate, $e_j^{Budget}$ is its amount of residual energy, and $P_j$ is its transmission power.

Denote by $\delta$ the data collection time unit, e.g., $\delta = 5$ seconds. This implies that if the UAV collects data from a sensor $v_j$ in cluster $C_i$, its data collection time is one of the values in set $\{\delta, 2\delta, \ldots, k_2\delta\}$, where $k_2$ is the maximum number of time units for collecting all data from sensor $v_j$. That is,

$$
k_j = \left\lfloor \frac{t_j^{max}}{\delta} \right\rfloor.
$$

Then, the number of time units for collecting data from sensors in cluster $C_i$ is no greater than $K_i = k_i + \sum_{v_j \in C_i \backslash \{v_i\}} k_j$, where $k_i$ and $k_j$ are the maximum numbers of time units for collecting all data from the master sensor $v_i$ and a slave sensor $v_j$ in cluster $C_i$, respectively. For example, assume that there are three sensors $v_1, v_7, v_8$ in cluster $C_1$, and $v_1$ is the master sensor in $C_1$, see Fig. 2(a). The amount of to-be-collected raw data in each of the three sensors is 10 MB. Assume that the data transmission rate from each sensor to the UAV is 8 Mbps (i.e., one MB per second). Then, the maximum number of time slots for collecting all data from each sensor in $C_1$ is $\frac{10MB}{8Mbps \times 5\text{seconds}} = 2$. Then, the number of time units for collecting data from the three sensors in cluster $C_1$ is no more than $K_1 = 2 \times 3 = 6$.

For each cluster $C_i$, we transform it into $K_i$ virtual nodes $v_{i,1}, v_{i,2}, \ldots, v_{i,K_i}$. The node weight $h(v_{i,k})$ of each virtual node is:

$$
h(v_{i,k}) = \frac{K_i}{\sum_{v_j \in C_i \backslash \{v_i\}} \frac{D_j^C}{P_j}} \frac{D_j^C}{P_j} + \frac{e_j^{Budget}}{P_j},
$$

where $D_j^C$ is the amount of collected data from sensor $v_j$.

**Fig. 2:** An illustration of the first graph transformation that each virtual node is associated with a node weight (i.e., UAV data collection energy consumption) and a profit (i.e., the utility of visiting the virtual node).
node \(v_{i,l}\) is the UAV data collection energy consumption for one time slot, i.e., \(h(v_{i,l}) = \xi \cdot \delta\), where \(\xi\) is the UAV energy consumption rate for hovering and data collection, and \(\delta\) is the duration of the time unit.

On the other hand, a profit \(p_{i,t}\) is associated with each virtual node \(v_{i,t}\) with \(1 \leq t \leq K_i\), which is the increased accumulative utility if the UAV collects data from a sensor at the \(t\)th time slot. To see from which sensor the UAV should collect data at the \(t\)th time slot, the increased accumulative utility of each sensor in \(C_i\) is first calculated. Then, the data from the sensor with the maximum increased accumulative utility is collected at the \(t\)th time slot. Specifically, recall that \(D_i\) and \(D_j\) are the amounts of collected data from the master sensor \(v_i\) and a slave sensor \(v_j\) in \(C_i\), respectively. Initially, \(D_i = 0\) and \(D_j = 0\). Following Eq. (1), the increased accumulative utility of the master sensor \(v_i\) in \(C_i\) at the \(t\)th time slot is \(f(D_i) = f(D_i) + \sum_{v_j \in C_i \setminus \{v_i\}} (f(D_j) + c(v_i, v_j) \cdot (D_i - R_i \delta))\), where \(R_i \delta\) is the amount of data that sensor \(v_i\) can send to the UAV at the \(t\)th time slot. Similarly, the increased accumulative utility of a slave sensor \(v_j\) in \(C_i\) at the \(t\)th time slot is \(f(D_j) + c(v_i, v_j) \cdot (D_j - R_i \delta)\). For example, Fig. 2(b) shows that six virtual nodes are transformed from cluster \(C_i\). To see from which sensor the UAV should collect data at the first time slot, the increased accumulative utility of each sensor in \(C_i\) is calculated as follows. Assume that the utility function is \(f(x) = \log_2(1 + x)\). In addition, the spatial data correlation \(c(v_1, v_2)\) between \(v_1\) and \(v_2\) is 0.3, while the correlation \(c(v_1, v_3)\) between \(v_1\) and \(v_3\) is 0.4. The increased accumulative utility of the master sensor \(v_i\) is \(f(0 + 5 MB) = f(0) + \sum_{v_j \in C_i \setminus \{v_i\}} (f(0 + c(v_1, v_j) \cdot (0 + 5 MB)) = \log_2(1 + 5 MB) = 5.49\). The increased accumulative utility of slave sensor \(v_j\) is \(f(0 + 5 MB + c(v_1, v_j) \cdot 0 - f(0 + c(v_1, v_j) \cdot 0) = 2.58\), while the increased accumulative utility of slave sensor \(v_2\) also is 2.58. Therefore, at the first time slot, the data from the master sensor \(v_1\) should be collected by the UAV and the increased accumulative utility is 5.49. Following Eq. (1), the increased accumulative utility of each virtual node \(v_{i,t}\) is calculated. Similarly, the profits of rest five virtual nodes \(v_{1,2}, v_{1,3}, \ldots, v_{1,6}\) can be calculated.

There is an important property for the profits of the virtual nodes, that is, the profits of the \(K_i \) virtual nodes \(v_{1,1}, v_{1,2}, \ldots, v_{1,K_i}\) constructed from each cluster \(C_i\) are in a decreasing order, see Fig. 2(b).

A graph \(G_1(V_1 \cup \{r\}, E_1)\) is constructed from the original IoT network, where \(V_1\) is the set of virtual nodes constructed from the \(Q\) clusters, i.e., \(V_1 = \bigcup_{i=1}^{Q} \{v_{1,1}, v_{1,2}, \ldots, v_{1,K_i}\}\). There is an edge in \(E_1\) between any two nodes in \(V_1\). The node weight \(h(v_{i,t})\) of each virtual node \(v_{i,t}\) is the UAV data collection energy consumption for one time slot. The edge weight \(w_{i,t}(v_{i,t}, v_{i,t})\) between two virtual nodes \(v_{i,t_1}\) and \(v_{i,t_2}\) constructed from two clusters \(C_i\) and \(C_j\) is the UAV flying energy consumption between the two hovering locations in the two clusters. In addition, the profit \(p_{i,t}\) of visiting virtual node \(v_{i,t}\) is the increased accumulative utility if the UAV collects data at the \(t\)th time slot. It can be seen that the data collection utility maximization in the original network can be transformed to a problem \(P_1\) of finding an \(r\)-rooted tour in \(G_1\), such that the profit sum of nodes in the tour is maximized, while ensuring that the sum of node weights and edge weights in the tour is no more than the UAV energy capacity.

C. Second graph transformation

In the second graph transformation, we transform the both node-weighted and edge-weighted graph \(G_1(V_1 \cup \{r\}, E_1)\) into an only edge-weighted graph \(G_2(V_2 \cup \{r\}, E_2)\), where \(V_2 = V_1\) and \(E_2 = E_1\). The profit \(p_{i,t}\) of each virtual node \(v_{i,t}\) in \(G_2\) is equal to its profit in \(G_1\).

The edge weight \(w_2(v_{i_1,t_1}, v_{i_2,t_2})\) between any two virtual nodes \(v_{i_1,t_1}\) and \(v_{i_2,t_2}\) is

\[
w_2(v_{i_1,t_1}, v_{i_2,t_2}) = w_1(v_{i_1,t_1}, v_{i_2,t_2}) + \frac{h(v_{i_1,t_1}) + h(v_{i_2,t_2})}{2},
\]

where \(w_1(v_{i_1,t_1}, v_{i_2,t_2})\) is the UAV flying energy consumption between the hovering locations of two clusters \(C_i\) and \(C_j\), \(w_1(v_{i_1,t_1}, v_{i_2,t_2}) = 0\) if \(i = j\) and \(w_1(v_{i_1,t_1}, v_{i_2,t_2})\) are the node weights of virtual nodes \(v_{i_1,t_1}\) and \(v_{i_2,t_2}\), respectively. For example, Fig. 3(a) shows a graph \(G_1\) that consists of root \(r\) and only three virtual nodes. Fig. 3(b) shows the edge weight \(w_2(v_{i_1,t_1}, v_{i_2,t_2})\) between nodes \(v_{i_1,t_1}\) and \(v_{i_2,t_2}\) in graph \(G_2\) is \(w_2(v_{i_1,t_1}, v_{i_2,t_2}) = 10 + \frac{5 \times 5}{2} = 15\).

![Fig. 3: An illustration of the second graph transformation](image)

We later will show that the problem \(P_1\) in graph \(G_1\) is equivalent to a problem \(P_2\) of finding an \(r\)-rooted tour in \(G_2\) such that the profit sum of nodes in \(G_2\) is maximized, under the constraint that the weighted sum of edges in the tour is no greater than the UAV energy capacity.

D. Third graph transformation

Notice that the problem \(P_2\) in \(G_2\) is still different from the orienteering problem, where the profits of different nodes in \(G_2\) are different, while the orienteering problem requires that the profits of different nodes are uniform. Then, we still cannot apply the \(1\)-approximation algorithm for the orienteering problem [2] to solve problem \(P_2\) in \(G_2\).

In the third graph transformation, we transform \(G_2\) to another auxiliary graph \(G_3\) such that the profits of different nodes are uniform in \(G_3\), by losing a performance of only \(\theta\), where \(\theta = \frac{\epsilon}{\xi}\), and \(\epsilon\) is a given constant with \(\epsilon > 0\).

Given graph \(G_2(V_2 \cup \{r\}, E_2)\), let \(n_2\) be the number of virtual nodes in \(G_2\), i.e., \(n_2 = |V_2|\). Denote by \(p_{\text{max}}\) the maximum profit of nodes in \(V_2\), i.e., \(p_{\text{max}} = \max_{v_i \in V_2} p_i\).
nodes in $G_3$ as possible, while ensuring that the weighted sum of edges in the tour is no greater than the UAV energy capacity. For example, Fig. 4(d) shows that the found tour by the algorithm is $r \rightarrow v_{1,1,1} \rightarrow v_{1,1,2} \rightarrow v_{1,2,1} \rightarrow r$. The found tour indicates that the UAV will collect data only from cluster $C_1$ for two time units, since virtual nodes $v_{1,1}$ and $v_{1,2}$ are constructed from cluster $C_1$, while the UAV does not collect any data from cluster $C_2$.

We refer to the approximation algorithm as Algorithm 1.

Algorithm 1 Algorithm ApproAlg for the data collection utility maximization problem

**Input:** The amounts of to-be-collected data $D_i^C$ and residual energy $e_i^{\text{Budget}}$ of each sensor $v_i$ with $1 \leq i \leq n$, $n$ clusters $C_1, C_2, \ldots, C_Q$ of the $n$ sensors, a data collection time unit $\delta$, and a performance loss value $\epsilon$ with $\epsilon > 0$.

**Output:** A data collection tour of a UAV, and the data collection duration of each sensor in each cluster.

1. First graph transformation $/*$
2. For each cluster $C_i$, transform it into $K_i$ virtual nodes $v_{i,1}, v_{i,2}, \ldots, v_{i,K_i}$, where the node weight $h(v_{i,l})$ of each virtual node $v_{i,l}$ is the UAV data collection energy consumption for one time slot, and its profit $p_{i,l}$ is the increased accumulative utility when the UAV collects data from a sensor at the $l$th time slot. Construct graph $G_1(V_1 \cup \{r\}, E_1)$, where $V_1$ is the set of the virtual nodes constructed from the $Q$ clusters; $r$
3. Second graph transformation $/*$
4. Construct a graph $G_2(V_2 \cup \{r\}, E_2)$ from $G_1$, where $V_2 = V_1 = E_2 = E_1$, the edge weight $w_2(v_{i,l}, v_{j,l'})$ between any two virtual nodes $v_{i,l}$ and $v_{j,l'}$ is $w_2(v_{i,l}, v_{j,l'}) = w_2(v_{i,l}, v_{j,l'}) + h(v_{i,l})h(v_{j,l'})$.
5. Third graph transformation $/*$
6. Let $\theta \leftarrow \frac{\delta}{2}$, $n_2 \leftarrow |V_2|$.
7. Let $\lambda \leftarrow \frac{p_{\text{max}}}{\theta}$, where $p_{\text{max}}$ is the maximum profit of nodes in $G_2$.
8. Construct a graph $G_3(V_3 \cup \{r\}, E_3)$ from $G_2$, where the edge weight $w_3(a,b)$ between any two nodes $a$ and $b$ in $G_2$ is its weight $w_2(a,b)$ in $G_2$, and the profit of a node $a \in V_3$ in graph $G_3$ is $p_3(a) = \left\lfloor \frac{p_{a,l}}{\lambda} \right\rfloor$.
9. Construct a graph $G_4(V_4 \cup \{r\}, E_4)$ from $G_3$, where $p_4$ nodes $a_1, a_2, \ldots, a_{p_4}$ are added to $V_3$ for each node $a_1 \in V_3$ in graph $G_3$.
11. Construct a tour $P^3_r$ in graph $G_2$ from $P^3_r$.
12. Construct a tour $P^3_r$ in $G_1$ and calculate the data collection duration of each sensor in every cluster from $P^3_r$.

E. Algorithm analysis

**Theorem 1:** Given the amounts of to-be-collected data $D_i^C$ and residual energy $e_i^{\text{Budget}}$ of each sensor $v_i$ with $1 \leq i \leq n$, an energy-constrained UAV located at depot $r$, and a data collection time unit $\delta$, assume that the $n$ sensors are partitioned into $Q$ clusters $C_1, C_2, \ldots, C_Q$. There is a $1/3$-approximation algorithm, i.e., Algorithm 1, for the data collection utility maximization problem, where $\epsilon$ is a given constant with $\epsilon > 0$.

**Proof:** The proof is contained in the supplementary file.
A cluster is collected by the UA V if the sensing data of a sensor in the cluster is collected by the UA V. The coverage ratio of clusters is defined as the ratio of the number of clusters collected by the UA V to the total number of clusters. A cluster is randomly chosen from the 54 sensors. The battery energy capacity of each sensor is 10.8 kJ [7]. Each sensor is equipped with a solar panel and its size is 37 mm \times 37 mm. We use real 89 energy harvesting profiles from the National Renewable Energy Laboratory [17] (from March 1 to March 31 in 2023). The energy harvesting profile of a sensor is randomly chosen from the 89 energy profiles. We consider the employment of different approaches from the one in [27]. (iv) Algorithm SpatiotemporalAlg [10] incorporates both spatial and temporal data correlations.

We consider following three performance metrics. (i) Accumulative utility measures the quality of collected data with spatiotemporal correlations and was defined in Section III-F; (ii) Coverage ratio of clusters. A cluster is collected by the UAV if the sensing data of a sensor in the cluster is collected by the UAV. The coverage ratio of clusters is defined as the ratio of the number of clusters collected by the UAV to the number of clusters Q in the network. (iii) Coverage ratio of sensors is the ratio of the number of sensors collected by the UAV to the total number n of sensors.

B. Algorithm performance

We first study the performance of different algorithms by varying the number of sensors from 100 to 500. Fig. 5(a) shows that the accumulative utility by each of the five algorithms increases with the number n of sensors in the network, as the UAV flying energy consumption is smaller in a denser network, and the UAV thus has more energy to collect data from more sensors. Fig. 5(a) also shows that the accumulative utility by the proposed algorithm ApproAlg is from 23% to 38% larger than those by the other four algorithms, as the UAV

V. PERFORMANCE EVALUATION

A. Experimental environment

We consider an IoT network deployed in a 1 km \times 1 km area. The number of sensors in the network is from 100 to 500, which are randomly placed in the area. We adopt real temperature sensing data of 54 sensors for 31 days (from March 1 to March 31 in 2004) from the Intel Berkeley Research Lab [9]. Then, the sensing data of a sensor is randomly chosen from the 54 sensors. The battery energy capacity of each sensor is 10.8 kJ [7]. Each sensor is equipped with a solar panel and its size is 37 mm \times 37 mm. We use real 89 energy harvesting profiles from the National Renewable Energy Laboratory [17] (from March 1 to March 31 in 2023). The energy harvesting profile of a sensor is randomly chosen from the 89 energy profiles. We consider the employment of a DJI Phantom 4 RTK UAV to collect data from sensors [5], its battery capacity is 89.2 Wh and its maximum flying time is 30 min. The UAV hovers at a height h = 300 m to collect data from sensors in each cluster.

We compare the proposed algorithm ApproAlg with following four benchmark algorithms. (i) Algorithm maxThroughput [13] considers neither spatial nor temporal data correlations, which schedules an energy-constrained UAV to maximize the amount of collected data from sensors; (ii) Algorithm spatialAlg1 [27] takes spatial data correlations into consideration; (iii) Algorithm spatialAlg2 [15] also considers spatial data correlations, but adopts a different approach from the one in [27]; (iv) Algorithm SpatiotemporalAlg [10] incorporates both spatial and temporal data correlations.

We consider following three performance metrics. (i) Accumulative utility measures the quality of collected data with spatiotemporal correlations and was defined in Section III-F; (ii) Coverage ratio of clusters. A cluster is collected by the UAV if the sensing data of a sensor in the cluster is collected by the UAV. The coverage ratio of clusters is defined as the ratio of the number of clusters collected by the UAV to the number of clusters Q in the network. (iii) Coverage ratio of sensors is the ratio of the number of sensors collected by the UAV to the total number n of sensors.

B. Algorithm performance

We first study the performance of different algorithms by varying the number of sensors from 100 to 500. Fig. 5(a) shows that the accumulative utility by each of the five algorithms increases with the number n of sensors in the network, as the UAV flying energy consumption is smaller in a denser network, and the UAV thus has more energy to collect data from more sensors. Fig. 5(a) also shows that the accumulative utility by the proposed algorithm ApproAlg is from 23% to 38% larger than those by the other four algorithms, as the UAV
in the solution found by algorithm ApproAlg collects data from more clusters in the fine-grained data collection model.

Fig. 5(b) demonstrates that the coverage ratio of clusters collected by algorithm ApproAlg is from 45% to 105% larger than those by the other four algorithms. For example, Fig. 5(b) plots that the coverage ratio of clusters by algorithms ApproAlg, maxThroughput, spatialAlg1, spatialAlg2, and SpatiotemporalAlg are 54%, 37%, 30%, 27%, and 24%, respectively, when there are 100 sensors in the network. Fig. 5(b) also shows that the coverage ratio of clusters by each of the five algorithms decreases when there are more sensors, as the UAV is energy-constrained and the number of clusters collected by the UAV will not increase too much with the growth of the number of sensors. Fig. 5(c) plots that the coverage ratio of sensors by ApproAlg is from 46% to 90% larger than those by the other four algorithms. In addition, it can be seen from Fig. 5(b) and Fig. 5(c) that the coverage ratio of clusters by algorithm ApproAlg is smaller than its coverage ratio of sensors. For example, its coverage ratio of clusters is 24% while its coverage ratio of sensors is 31%, when there are 500 sensors. The rationale behind is that the UAV in the solution delivered by algorithm ApproAlg is more likely to collect data from clusters with many sensors, and barely collect data from clusters with only a few sensors, when the energy in the UAV is constrained.

We then evaluate the algorithm performance by varying the maximum data rate \( R_{\text{max}} \) from 2 Kbps to 10 Kbps, where the data rate of a sensor is randomly chosen in an interval \([0, R_{\text{max}}]\). Fig. 6(a) shows that the accumulative utility by algorithm ApproAlg is from 12% to 40% larger than those by the other four algorithms. In addition, the accumulative utility by algorithm ApproAlg slightly increases with the growth of the maximum data rate \( R_{\text{max}} \), whereas the accumulative utilities by the other four algorithms decrease with the growth of \( R_{\text{max}} \). The rationale behind the phenomenon is that, although there are more to-be-collected data in each sensor with the growth of \( R_{\text{max}} \), the UAV in the solution by algorithm ApproAlg collects data from more clusters than the other four algorithms, and both the coverage ratios of clusters and sensors do not change too much with the growth of \( R_{\text{max}} \), see Fig. 6(b) and Fig. 6(c).

In contrast, the UAV by algorithm maxThroughput collects more data from each sensor, while the UAV by each of the three algorithms spatialAlg1, spatialAlg2, and SpatiotemporalAlg will collect data from less clusters and less sensors, see Fig. 6(b) and (c).

We finally investigate the algorithm performance by increasing the data collection time unit \( \delta \) from 5 to 60 seconds, when there are 100 sensors. Fig. 7(a) plots that the accumulative utility by each of the four existing algorithms maxThroughput, spatialAlg1, spatialAlg2, and SpatiotemporalAlg does not change with the growth of \( \delta \), as the raw data or compressed data in each cluster in the solutions by the four algorithms will be fully collected, if the UAV collects data from the cluster. In contrast, Fig. 7(a) demonstrates that the accumulative utility by algorithm ApproAlg slightly decreases when the time unit \( \delta \) decreases from 5 to 20 seconds, and the utility becomes even smaller than that by algorithm SpatiotemporalAlg when \( \delta \) is 60 seconds. The rationale behind the phenomenon is as follows. Recall that, in the proposed algorithm ApproAlg, if the UAV collects data from a sensor \( v_j \), its data collection time is one of the values in set \([\delta, 2\delta, \ldots, k_j\delta]\), where \( k_j = \left\lceil \frac{R_{\text{max}}}{\delta} \right\rceil \), and \( t_j^{\text{max}} \) is the maximum time for collecting all data from sensor \( v_j \). Then, if the value of the data collection time unit \( \delta \) is larger, the UAV collects more data in each time unit, and thus has less options on the data collection time. In addition, the difference between the two values \( k_j\delta \) and \( t_j^{\text{max}} \) is larger, since \( 0 \leq k_j\delta - t_j^{\text{max}} < \delta \), where \( k_j\delta = \left\lceil \frac{R_{\text{max}}}{\delta} \right\rceil \). For example, the maximum data collection time \( t_j^{\text{max}} \) of a sensor \( v_j \) is 50 seconds, while the data collection time unit \( \delta \) is 60 seconds. The UAV will spend one data collection time unit for 60 seconds to collect data from sensor \( v_j \), and waste 10(= 60 − 50) seconds on the data collection. This indicates that, if the data collection time unit \( \delta \) is too large, the performance of algorithm ApproAlg may be worse than the existing algorithms that collect all raw data or compressed data from each cluster.

On the other hand, Fig. 7(b) shows that the running time of algorithm ApproAlg becomes smaller when the time unit \( \delta \) is larger, since less number \( k_j \) of virtual nodes are constructed from each sensor in the first graph transformation of algorithm ApproAlg, and the running time of the algorithm is proportional to the number of constructed virtual nodes from
all sensors, where \( k_2 = \lceil \frac{m-n}{\epsilon} \rceil \). In contrast, the running time of each of the other four algorithms does not change with the increase of \( \delta \), as no virtual nodes are constructed in the four algorithms.

VI. CONCLUSION

Unlike existing studies that adopted the coarse-grained data collection model, in which all compressed data in each cluster must be collected to recover the original raw data, if the UAV collects data from the cluster, in this paper we used a light-weight fine-grained data collection model in which compressed data can be partially collected.

Under the fine-grained data collection model, we investigated a problem of finding a data collection trajectory for an energy-constrained UAV, so that the accumulative utility of collected data is maximized. We proposed a novel \( 1 + \epsilon \)-approximation algorithm for the problem, by transforming the problem to the orienteering problem via a series of three smart graph transformations, where \( \epsilon \) is a constant with \( \epsilon > 0 \). In the first graph transformation, we exploited spatial-temporal similarities and transformed each cluster into several virtual nodes so that each virtual node is associated with a node weight (i.e., UAV data collection energy consumption) and a profit (i.e., the utility of visiting the virtual node). In the second graph transformation, we transformed the both node-weighted and edge-weighted graph obtained in the first graph transformation into an only edge-weighted graph. In the last transformation, we transformed each virtual node into several nodes with uniform profits, through a scaling technique, so that the approximation algorithm for the orienteering problem can be applied in the graph obtained in the third graph transformation.

Experimental results with real datasets showed that the accumulative utility by the proposed algorithm is at least 23% larger than those by the existing studies, and the number of clusters collected by the proposed algorithm is from 45% to 105% larger than those by the existing studies.

ACKNOWLEDGEMENT

The work by Wenzheng Xu was supported by the National Science Foundation of China (NSFC) with grant number 62272328 and Sichuan Science and Technology Program with grant number 24NSFJQ0152. The work by Jian Peng was supported by the Cooperative Program of Sichuan University and Yibin (2020CDYB-30), the Cooperative Program of Sichuan University and Zigong (2022CDZG-6), the Key R&D Program of Sichuan Province of China (222YDF3599), and Sichuan Science and Technology Program under Grant 2022ZDZX0011.

REFERENCES


Wenzheng Xu (M’15) received the BSc, ME, and PhD degrees in computer science from Sun Yat-Sen University, Guangzhou, P.R. China, in 2008, 2010, and 2015, respectively. He currently is an Associate Professor at Sichuan University. Also, he was a visitor at both the Australian National University and the Chinese University of Hong Kong. His research interests include Internet of Things, UAV networks, mobile computing, approximation algorithms, combinatorial optimization, online social networks, and graph theory. He has published more than 100 papers on prestigious journals and conferences, such as ToN, TC, TMC, TPDS, TKDE, INFOCOM, ICDCS, etc. He is a member of the IEEE.

Heng Shao received the BE degree in Computer Science and Technology from Sichuan University, P. R. China, in 2022. He now is a master student in College of Computer Science at Sichuan University. His current research interests include IoT networks and social networks.

Qunli Shen received the BE degree in Network Engineering from Zhejiang Normal University, P. R. China, in 2021. He now is a master student in College of Computer Science at Sichuan University. His current research interests include UAV networking and IoT networks.

Jian Peng is a Professor at College of Computer Science, Sichuan University. He received his B.A. degree and PhD degree from the University of Electronic Science and Technology of China (UESTC) in 1992 and 2004, respectively. His recent research interests include Internet of Things, UAV networks, big data, and cloud computing.

Wen Huang received the Ph.D. degree from the School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, China, in 2022. He is working a Postdoctoral Fellow with the College of Computer Science, Sichuan University, Chengdu. His main research interests include cryptography, differential privacy, and IoT networks.

Weifa Liang (M’99–SM’01) received the PhD degree from the Australian National University in 1998, the ME degree from the University of Science and Technology of China in 1989, and the BSc degree from Wuhan University, China in 1984, all in Computer Science. He currently is a Professor in the Department of Computer Science at City University of Hong Kong. Prior to the current position, he was a Professor at the Australian National University. His research interests include design and analysis of energy efficient routing protocols for wireless ad hoc and sensor networks, Internet of Things, edge and cloud computing, Network Function Virtualization and Software-Defined Networking, design and analysis of parallel and distributed algorithms, approximation algorithms, combinatorial optimization, and graph theory. He serves as an Associate Editor for the IEEE Trans. Communications. He is a senior member of the IEEE.

Tang Liu received the BS degree in computer science from University of Electronic and Science of China, in 2003, and the MS and PhD degrees in computer science from Sichuan University, in 2009 and 2015, respectively. He is a professor in the College of Computer Science, Sichuan Normal University. Since 2015 to 2016, he was a visiting scholar with the University of Louisiana at Lafayette. His research interests include wireless charging and wireless sensor networks.

Xinwei Yao received the PhD degree in information engineering from the Zhejiang University of Technology, Hangzhou, China, in 2013. He is currently an associate professor with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. From March 2012 to February 2013, he was a visiting scholar with the Loughborough University, Leicestershire, United Kingdom. From August 2015 to July 2016, he was a visiting professor with the University of Buffalo. The State University of New York, Buffalo, NY. He was the recipient of the distinguished associate professor Award and the Outstanding Doctoral Thesis Award with the Zhejiang University of Technology. He has served on technical program committees of many IEEE/ACM conferences. He is a member of the ACM. His current research interests include the area of Terahertz-band communication networks, electromagnetic nanonetworks, wireless ad hoc and sensor networks, wireless power transfer, and the Internet of Things.

Tao Lin is a full Professor at College of Computer Science, Sichuan University. His recent research interests include intelligent software engineering and optimization.

Sajal K. Das is the Chair of Computer Science Department and Daniel St. Clair Endowed Chair at the Missouri University of Science and Technology in the States. His current research interests include theory and practice of wireless sensor networks, big data, cyber-physical systems, smart healthcare, distributed and cloud computing, security and privacy, biological and social networks, applied graph theory and game theory. Dr. Das directed numerous funded projects in these areas totaling over $15M and published extensively with more than 600 research articles in high quality journals and refereed conference proceedings. Dr. Das serves as the founding Editor-in-Chief of the Pervasive and Mobile Computing journal, and as Associate Editor of IEEE Transactions on Mobile Computing, ACM Transactions on Sensor Networks, etc. He is a co-founder of the IEEE PerCom, IEEE WoWMoM, and ICDCN conferences, and served on numerous conference committees as General Chair, Program Chair, or Program Committee member. Dr. Das is a Fellow of the IEEE.