Incentivizing Opportunistic Data Collection for Time-Sensitive IoT Applications

Pranvera Kortoçi
University of Helsinki
pranvera.kortoci@helsinki.fi

Abbas Mehrabi
Northumbria University
abbas.mehrabidavoodabadi@northumbria.ac.uk

Carlee Joe-Wong
Carnegie Mellon University
cjoewong@andrew.cmu.edu

Mario Di Francesco
Aalto University
mario.di.francesco@aalto.fi

***Abstract*—Urban environments are the most prevalent application scenario for the Internet of Things (IoT). In this context, effective data collection and forwarding to a cloud (or edge) server are particularly important. This work leverages opportunistic data collection based on the mobile crowd sourcing (MCS) paradigm for time-sensitive IoT applications. Specifically, it introduces an incentive mechanism for the crowd to collect data that are valuable to data consumers in terms of regions of interest and time constraints. The proposed approach successfully incorporates the willingness of the crowd to participate in the data collection as part of the related incentives. It also ensures collection of valuable data via selective user incentivization. Accordingly, a weighted social welfare maximization problem is defined for users to decide which sensors to visit subject to deadline constraints. Following the NP-hardness of the problem, an online heuristic algorithm is proposed for sensors to dynamically incentivize mobile users with a low message and time complexity. The proposed solution is shown to be effective for time-sensitive quality data collection through extensive simulations on realistic mobility traces. It significantly increases the overall social welfare as well as the amount of collected data compared to other approaches.**

***Index Terms*—Incentives, opportunistic data collection, data utility, IoT, mobile crowd sourcing**

I. Introduction

Internet of Things (IoT) sensors are at the core of different applications in smart cities, logistics, and the industrial Internet [1]. For any of them, reliable delivery of sensed data (e.g., to cloud or edge servers) is extremely important. To address such an issue, the concept of *opportunistic IoT* applies the paradigm of delay-tolerant networking to urban scenarios [2]. Accordingly, sensory data sampled by IoT devices are collected by mobile gateways, generally people carrying smartphones, as in *mobile crowd sourcing* (MCS) [3].

The ubiquitous presence of mobile personal devices makes the opportunistic IoT particularly attractive, as either an *alternative* or a *supplement* to a traditional wireless sensing infrastructure. In particular, opportunistic data collection eases the integration of heterogeneous IoT sensors, which may otherwise require different transmission protocols such as Long Range (LoRa) through separate and costly networks [4, 5]. It also extends to isolated networks, e.g., designed as such to reduce energy consumption or resulting from failures [6, 7].

An IoT system is effective only when the sensory data are valuable enough to support applications with different requirements. However, mobile devices are carried by users, whose availability and willingness to participate in data collection are inherently unpredictable. Moreover, sensory data collection causes users both energy and monetary costs, in addition to the burden to modify their *planned* path. Therefore, ensuring user participation in data collection is crucial [8], and thus suitable incentive mechanisms such as monetary compensation, virtual cash, or redeemable credit must be put in place [9].

Incentive mechanisms in MCS generally aim not only at satisfying the economic properties of truthfulness, individual rationality, efficiency, and non-negative social welfare [10–15], but also include optimal user selection and task allocation that guarantees a target *service quality* [11, 16, 17]. Moreover, tasks are often location-dependent [18]; thus, mobile users (as data collectors) that are nearby or plan to travel to a location of interest are often prioritized [9, 14, 16, 19]. Indeed, this article specifically accounts for the *inherent dynamicity* in user mobility and the corresponding *burden* for users to move.

Designing effective incentive mechanisms for users to change their mobility patterns poses several new research challenges. First, the user's cost of collecting data now includes not only the *energy required to collect data* from sensors and forward it to a cloud server, but also the *cost incurred by modifying the user route*. Evaluating each of these costs requires us to go beyond typical MCS incentive studies (Section VI) and extend our attention to evaluating the user's willingness to reach a certain region as a function of the burden put on them. In fact, this task is non-trivial and may be quite user-specific. Furthermore, we *selectively incentivize users to collect data that is valuable to IoT applications*: collecting data of low diversity (i.e., from co-located regions) might simply waste user effort. Incorporating this notion of data value – equivalently, *utility* – into the offered incentives is challenging, as it makes them dependent on multiple users' actions.

We solve these challenges by introducing an incentive-based MCS system in which *individual sensors charge mobile users a dynamic virtual price to collect their data* (Section II). Users decide which region to visit after accepting a sensing task and the associated compensation from the MCS platform, willingly participating to obtain a payoff. In doing so, they consider the service quality requirements of the task and their own costs of collecting data from different sensors; then, they pay the sensors out of the compensation given by the MCS platform. Moreover, users evaluate their expected costs based on the inconvenience in changing their routes to visit regions of interest. Each sensor encodes the utility of its own data and the frequency of user visits into a single quantity, which serves

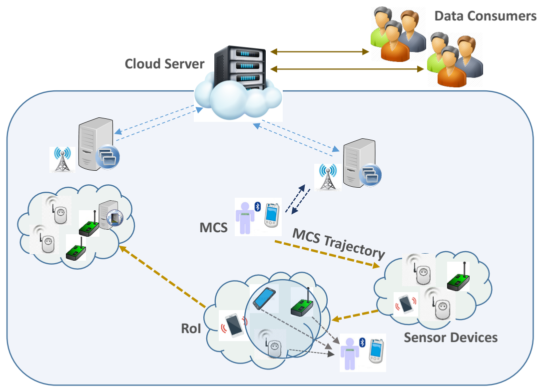


Fig. 1: A mobile crowdsourcing system as considered in this work.

as an incentive for the user to collect data from that sensor. This design thus allows the sensors to incentivize users away from them when their data has recently been collected.

We show that users' decisions regarding which sensors to visit under a given set of sensor prices can be formulated as a weighted social welfare maximization problem. Such an approach allows individual users to evaluate their own costs of altering their routes and collecting data from specific regions by a given time (Section III). Such a problem is computationally hard, and is also complicated by the need to estimate key operating parameters dynamically and efficiently. In this respect, we introduce a Utility-based Opportunistic Data Collection Algorithm (UO-DCA) that addresses weighted social welfare maximization with delay constraints as a distributed process (Section IV). Thus, the MCS platform need not estimate users' personal mobility costs. This design is also flexible to changes in the MCS system; for instance, if the data utility drops, the sensor can simply change its price without having to wait for the cloud server or mobile users to discover such. Extensive simulations based on realistic mobility traces (Section V) show that our proposed solution is effective, providing adequate incentives for users to alter their path and collect data from sensors within a specified deadline.

II. BACKGROUND

A. Reference Architecture

The reference IoT scenario we consider (Fig. 1) includes different components: a set of sensors deployed in a large geographical area; a set of mobile users (MUs), i.e., people carrying personal devices such as smartphones; a platform that organizes the crowdsourcing campaign, residing in the cloud; edge servers deployed near the MUs; and data consumers.

Data consumers are people or companies that are interested in sensory data characterizing certain physical phenomena. Static sensors are deployed in the sensing field; they periodically collect data from the environment through short- or medium-range communication technologies (i.e., BLE) and store them locally. Data consumers submit requests to access sensory data from a certain region of interest (RoI) to the crowdsourcing platform, e.g., a cloud server; one or more sensors could be located in a RoI. In turn, the cloud server allocates the data collection tasks – which have an associated *deadline* – to the MUs with the help of the edge servers. In fact, it is unlikely that all sensors can reach an edge server with short-range wireless technologies like BLE, thus requiring the help of MUs to collect sensed data. Even longer-range communication technologies like LoRA may not be able to support the high throughput needed to handle all sensors' transmitting data to the same edge server [20].

The cloud server receives a request from data consumers and broadcasts it to the edge servers which, in turn, assign the data collection tasks and monetary incentives[1] to the MUs. The MUs evaluate the costs of the data collection tasks, and if they agree, they visit the sensors and collect their data.

B. System Model

The sensing area consists of K regions of interest (RoIs) denoted by set $\mathcal{R} = \{r_1, r_2, ..., r_K\}$. Multiple sensor devices sense the phenomena within each region; those in region r_k are denoted by $S_k \subset S$, where S is the set of all sensors in the sensing area. The cloud server recruits N MUs (either pedestrians or people in vehicles), denoted by set $\mathcal{U} = \{u_1, u_2, \ldots, u_N\}$, such that each u_n moves within the RoIs to collect data from sensors and transmit them to the nearest edge server. A data collection round has $|T|$ time slots, each lasting for Δt time; u_n starts its path traversal at time slot A_n, then visits the set of RoIs $\mathcal{R}_n^{(t)}$ until time slot t in a certain sequence. Data are time-sensitive, thus the cloud server imposes a deadline D_n for MU u_n to complete its task. Visits of MUs to regions and sensor selection are modeled with binary variables: $x_{nr}^{(t)}$ and $y_{nsr}^{(t)}$ are set to one if region r is visited by MU u_n at time slot t, and sensor s in region r sends data to MU u_n, respectively; otherwise to zero.

Sensor $s \in S_r$ in region r transmits data to MU u_n starting at time slot a_s and ending at d_s. We suppose that a_s coincides with the MU entering the region, after which data collection immediately starts [22]. The ending time slot $a_s \leq d_s \leq |T|$, and thus the amount of data sensed, is chosen by the user as described in Section III. Furthermore, sensor s advertises the *utility* $0 < \varphi_{st} \leq 1$ and the associated reward R_{st} for the data sensed at time slot t during transmission. The utility expresses how data are valuable or beneficial for a given IoT application. We assume that utility is proportional to the data volume. For instance, utility could be related to the accuracy of collected data such as the sampling frequency of a certain signal or the size of an image. Higher utility is more desirable not only for data-intensive applications; in fact, a higher sampling frequency of sensors translates into more samples per unit of time, thus, better characterizes the phenomenon of interest.

Ensuring a certain utility requires some *effort* by sensors. This effort is quantified in terms of a power consumption (per time slot) which depends on both sensing and communication. Specifically, a sensor spends p_s power for sampling; and $P_{st} =$

[1]We assume monetary remuneration for simplicity, as incentives in mobile crowdsourcing are widely studied [21] and out of the scope of this work.

$c_s \cdot \varphi_{st}$ power for transmitting the data to a MU, but only if user u_n collects data from sensor $s \in S_r$ at time slot t (i.e., $y_{nsr}^{(t)} = 1$). Note that the transmission power depends on the instantaneous utility of sensed data. The rationale behind this choice is that the sensor effort increases with the importance of the data; clearly, the amount of recently collected data directly impacts the number of transmitted messages. This model also captures other policies that can improve data utility through higher communication reliability, for instance, by increasing either message redundancy (such as with erasure codes) or the transmission power to reduce channel errors. Similarly, each MU u_n consumes: p_n power for traveling between RoIs; $P_{nst} = c_n \cdot \varphi_{st}$ power for collecting and transmitting data from sensor s in region r (i.e., $x_{nr}^{(t)} = 1$ and $y_{nsr}^{(t)} = 1$).

C. Reward and Profit Model

One of the key components of our system design is the price that sensors charge the MUs for collecting their data, which are paid out of the prices that the cloud server pays to the MU. Similar to prior research [23], our system provides the users with a-priori knowledge on their expected profit, allowing them to take informed decisions on whether or not to carry out a task. We propose a simple method for the sensors to set these prices so as to achieve our goals of incentivizing the collection of useful data (i.e., highly-valuable data that has not been recently collected). While setting such prices itself is an interesting optimization problem, we instead propose a simple method that requires limited computing at the sensors. We then use these prices to analytically derive the profits for sensors and MUs. In the next section, we characterize such a profit to formulate and solve the MUs' problem of deciding which sensors to visit given the incentives offered by the sensors.

We suppose that the cloud server pays the MU the fixed amount of I_s reward for the collected data from sensor s per time slot, which it determines based on consumers' data requests. Each sensor in turn determines the reward for its samples based on the utility and the (historical) frequency of visits by MUs to the RoI of the sensor. Each sensor's goal is to obtain high profits by motivating MUs to visit the corresponding RoIs. Accordingly, the (time-dependent) pricing model for sensor $s \in S_r$ in region r at time t is:

$$R_{st} = R_{s(t-1)} + \Big(\alpha \frac{V_{r(t-1)} - \bar{V_r}^{(t-1)}}{\bar{V_r}^{(t-1)}} t + \beta \frac{\varphi_{st} - \bar{\varphi_s}^{(t)}}{\bar{\varphi_s}^{(t)}}\Big) \cdot \Delta R \tag{1}$$

where ΔR is a constant incremental reward specified by the deployer; $\bar{V_r}^{(t-1)} = \frac{\sum_{n=1}^{N} \sum_{t'=1}^{t-1} x_{nr}^{(t')}}{t-1}$ is the average of the MUs visits to region r up to time slot $t-1$; and $\bar{\varphi_s}^{(t)} = \sum_{t'=1}^{t} \varphi_{st'}/t$ is the average utility of data at sensor s up to time slot t. Sensors with fewer MU visits (e.g., those in remote areas) would then have lower prices, potentially attracting more MUs in the future. Sensors with higher utility would increase their prices to compensate for the higher transmission costs. The weights $\alpha, \beta \in [0, 1]$, with $\alpha + \beta = 1$, reflect the impact of the frequency in MUs' visits and the utility (respectively) on the price; they are determined by the sensors through an internal policy. We empirically evaluate the effects of α and β on the collected data in Section V.

The **profit of each sensor** s from crowdsourced data collection is its overall revenue less the associated costs:

$$\text{Profit}_\text{S}(s) = \text{Revenue}_\text{S}(s) - \text{Cost}_\text{S}(s) \tag{2}$$

The sensor revenue is the sum of the rewards received from all MUs for the data transmitted during its allocated time duration: $\text{Revenue}_\text{S}(s) = \sum_{t=a_s}^{d_s} R_{st}$. The associated cost is the energy consumption for data sampling and transmission to the MU during the allocated time duration: $\text{Cost}_\text{S}(s) = \sum_{t=a_s}^{d_s} a(p_s + P_{st}) \cdot \Delta t$, where the coefficient a scales the energy consumption to be comparable to the monetary sensor reward.

The **profit an MU** u_n achieves from crowdsourced data collection is obtained as:

$$\text{Profit}_\text{MU}(n) = \text{Revenue}_\text{MU}(n) - \text{Cost}_\text{MU}(n) \tag{3}$$

The revenue of MU n is the reward it receives from the cloud server for collecting data from all visited sensors and transmitting it to the cloud. Precisely:

$$\text{Revenue}_\text{MU}(n) = \sum_{t=1}^{|T|} \sum_{r=1}^{K} x_{nr}^{(t)} (\sum_{s=1}^{S_r} y_{nsr}^{(t)} \cdot (d_s - a_s) \cdot I_s) \tag{4}$$

MUs prefer to visit sensors whose data cost less and that are close to their intended path. Similarly, MUs must also visit sensors that the consumer values. Since the price of sensory data increases with the number of visits and utility, MUs are incentivized to visit sensors sampling data with high utility up to the point where the price increases too much. At this point, MUs start prioritizing other sensors, resulting in a trade-off between data utility and corresponding price over time.

The costs associated with MU n's data collection include: the energy consumption or inconvenience of path traversal (p_n); the rewards (R_{st}) to the sensors s for collecting their data at time t; and the energy consumption for transmitting the data to a nearby edge server (P_{nst}):

$$\begin{aligned}\text{Cost}_\text{MU}(n) = &\Bigg(\sum_{\forall t, t', t<t'} x_{nr}^{(t)} \cdot x_{nr'}^{(t')} \cdot \prod_{\forall\ t<t''<t'} (1 - x_{nr''}^{(t'')})\Bigg) \\ &(t' - t - \max_{s: y_{nsr}^{(t)}=1} \{d_s - a_s\} - \sum_{\forall s \in S_r} y_{nsr}^{(t)} \cdot (d_s - a_s)) \cdot b(p_n \cdot \Delta t) \\ &+ \sum_{t=1}^{|T|} \sum_{r=1}^{K} x_{nr}^{(t)} \Big(\sum_{s=1}^{S_r} y_{nsr}^{(t)} \big(\sum_{t=a_s}^{d_s} (R_{st} + cP_{nst}) \cdot \Delta t \big) \Big)\end{aligned} \tag{5}$$

The weights b and c allow us to scale the energy cost to be a monetary value that is comparable to the reward of the MU. Note that the time of data collection and transmission in region r has to be excluded from the time for traveling between region r visited at time t and the next region r' visited at time t'. To do this, the first term in the right hand side of Eq. (5) states that for every two subsequent time slots t and t' for which the MU visits ROIs r and r' (the first summation) without stopping at any other ROI r'' between them (the product), the time for data collection from the sensors (the $\max$ term) and the data transmission to the edge (the subsequent summation) should be subtracted from the time spent in these ROIs $(t-t')$. The second term indicates the overall reward paid by the MU to the sensors in each region along with the overall energy consumption for transmitting the data to a nearby edge server.

III. Mobile Crowdsourcing with Delay Constraints

The following formulates an optimization problem for incentive-based mobile crowdsourcing subject to delay constraints (MCSD). To fairly share the obtainable profits between both MUs and sensors, we further define an adjustable weighting parameter $0 \leq \gamma \leq 1$ in the objective function. The parameter γ allows us to tune the selfishness / generosity of the MUs. As we show below, the MUs act solely in their own best interest for $\gamma = 1$, while more generous MUs – e.g., those that collect data altruistically – might use a lower value of γ. The edge server could as well set $\gamma = 0.5$ to maximize the sensor and MU profit equally. We encapsulate these different scenarios by maximizing the weighted social welfare of the system, so as to encourage both sensors and MUs to participate in data sampling and data collection tasks (respectively) by increasing their profits based on the weighting parameter γ. In particular, the MCSD problem is defined as:

$$\max_{x,y,d} \mathrm{U} = \gamma \sum_{n=1}^{N} \mathrm{Profit_{MU}}(n) + (1-\gamma) \sum_{t=1}^{|T|} \sum_{n=1}^{N} \sum_{r=1}^{K} x_{nr}^{(t)} \Big(\sum_{s=1}^{S_r} y_{snr}^{(t)} \cdot \mathrm{Profit_S}(s) \Big) \tag{6}$$

Subject to:

$$\Big(\sum_{\forall t<t'} \big(x_{nr}^{(t)} \cdot x_{nr'}^{(t')} \cdot \big(\prod_{\forall t<t''<t'} (1 - x_{nr''}^{t''}) \big) (t'-t) \big) \Big) \Delta t \leq D_n, \quad \forall 1 \leq r, r', r'' \leq K, 1 \leq n \leq N \tag{7}$$

$$\sum_{t=1}^{|T|} x_{nr}^{(t)} \leq 1, \quad \forall 1 \leq n \leq N, 1 \leq r \leq K \tag{8}$$

$$\mathrm{Profit_S}(s) \geq 0, \quad \forall s \in \mathcal{S} \tag{9}$$

$$\mathrm{Profit_{MU}}(n) \geq 0, \quad \forall n \in \mathcal{U} \tag{10}$$

$$a_s \leq d_s \leq |T|, \quad d_s \in \mathbb{N} \tag{11}$$

$$x_{nr}^{(t)}, y_{snr}^{(t)} \in \{0,1\} \quad \forall 1 \leq n \leq N, 1 \leq r \leq K, s \in S \tag{12}$$

The objective function in Eq. (6) expresses the weighted social welfare of the data collection system, as the summation of the achievable profit for all MUs and sensors in a data collection round. Eq. (7) ensures that the total time taken by an MU – including the visits to RoIs, collecting data from the sensors and transmitting the data to a nearby edge server – does not exceed a certain deadline decided by the cloud server. Eq. (8) states that each mobile user should visit every RoI at most once in a data collection round for better coverage of the whole sensing area. Eqs. (9)–(10) specify that the achievable profits for sensors and MUs [given in Eqs. (2)–(3)] must be non-negative, and thus are incentivized to participate. Moreover, the task deadline should be between the task starting time and the end of the data collection round [Eq. (11)]. Finally, Eq. (12) restricts the variables to (binary) integer values. Accordingly, the MCSD problem is an integer programming problem; the problem is also non-linear due to the relation in Eq. (7).

Theorem 1 (NP-hardness)**.** *The MCSD problem is NP-hard.*

Proof: For simplicity, we consider the special case in which the system contains only one MU, one sensor node in every RoI, and one edge server. The NP-hardness of this special case trivially implies the same for the general problem with multiple MUs, more than one sensor per RoI, and edge servers covering only a subset of RoIs. Consider an instance of the 0/1 capacitated knapsack (CK) problem with capacity C and N items of profit p_i and weight w_i, $i \in N$. The objective is the selection of a subset $N' \subseteq N$ of items that maximizes $\sum_{i \in N'} p_i$ subject to the capacity of the knapsack $\sum_{i \in N'} w_i \leq C$. A reduction of such a CK instance is provided next. The set of N items in CK are mapped to the set of K RoIs in the sensing area and the capacity is mapped to the deadline D_n of MU n. The weight w_i of item i in the CK problem is mapped to (i) the total time t_i to visit RoI $i \in K$, (ii) time $t_s^{(i)}$ to collect data from sensor node at that RoI , and (iii) one time slot $t_e^{(i)}$ to transmit data to the edge server, namely, $t_i + t_s^{(i)} + t_e^{(i)}$. Furthermore, the profit of item i in the CK problem, p_i, is mapped to the achievable weighted social welfare $\gamma \cdot \mathrm{Profit_{MU}}(n) + (1-\gamma) \cdot \mathrm{Profit_S}(s_i)$ when MU n visits RoI i to collect data from sensor s_i therein. The MU can visit RoIs covered by the edge server in any order. Now, the problem of visiting a subset $K' \subseteq K$ of RoIs by MU n for collecting data from sensors with the objective of maximizing $\sum_{i \in K'} (\gamma \cdot \mathrm{Profit_{MU}}(n) + (1-\gamma) \cdot \mathrm{Profit_S}(s_i))$ subject to the deadline, i.e., $\sum_{i \in K'} (t_i + t_s^{(i)} + t_e^{(i)}) \leq D_n$ is equivalent to the above-mentioned instance of the CK problem. Since the 0/1 CK problem is NP-hard [24], so is the special case considered here, as it is the (more general) MCSD problem. ∎

Theorem 2 (Feasibility)**.** *The MCSD problem is feasible.*

Proof: A solution to the MCSD problem is feasible if it is a sequence of RoI visits by an MU n that satisfies the constraints in Eqs. (7)–(10). Assume that the traversal of MU n at time slot t' is within the deadline, i.e., $t' \leq D_n$. Furthermore, assume that t_{min} is the minimum time needed by MU n to visit the nearest unvisited RoI containing at least one new sensor node. Two cases follow: (1) $t' + t_{min} + 2\Delta t \leq D_n$: the MU can still visit the nearest RoI and collect data for at least one time slot from at least one sensor by satisfying Eqs. (9)–(10). The total time is still within the deadline [Eq. (7)] and the selected RoI has not yet been visited [Eq. (8)]; (2) $t' + t_{min} + 2\Delta t > D_n$: the time to visit the nearest RoI, collect data from at least one sensor and transmitting it to an edge server exceeds the deadline of MU. In this case, the MU can stop and the solution obtained until time t' is feasible. ∎

Note that the existence of a feasible solution for the MCSD problem could not have otherwise been deduced from the 0/1 CK reduction due to the more restricted settings in Theorem 1.

IV. Utility-based Opportunistic Data Collection

The MCSD problem cannot be solved efficiently since it is NP-hard. Exhaustive search methods are infeasible, due to their extremely high computational complexity and the unavailability of contextual information in advance (e.g., instantaneous utility / rewards of sensed data and the power consumption of nodes). This section presents a scalable and low-complexity heuristic for the MCSD problem – Utility-based Opportunistic Data Collection (UO-DCA).

A. UO-DCA: An Online Greedy Algorithm

The most challenging task in designing an efficient algorithm for the MCSD problem is collecting the instantaneous sensory data from the devices in different RoIs at a large scale. All interactions could be directly controlled by the cloud, at the expense of high latency and an increase of traffic in the backhaul. In contrast, UO-DCA leverages the mobile edge computing (MEC) paradigm [25] to move the communication / processing tasks to the edge of the network, nearby mobile users. Accordingly, edge servers manage the tasks for the RoIs under their coverage (Fig. 1).

Local coordinators in each RoI inform edge servers about the average utility of sensory data and energy consumption of the devices in that area. Each MU contacts the nearest edge server at the beginning of its traversal to receive the contextual information of the RoIs, including their physical distances. These can be obtained, for instance, by using GPS. The MU then aggregates this contextual information to decide on the most suitable RoI to visit. Visiting that RoI results in the locally-maximum weighted social welfare while satisfying the deadline associated with the overall data collection time – including traversal to that RoI, receiving data from sensors and transmitting data to a nearby edge server. Since MUs have no prior information about the time duration during which sensors in each RoI transmit their data, we propose our online heuristic that approximates the locally achievable weighted social welfare from each unvisited RoI.

Estimating the time required for future data collection. Assume that at a given time slot t, mobile user u_n with arrival time A_n needs to decide which region to visit next, given the set of unvisited RoIs nearby. The MU computes the achievable profit for the sensors upon receiving the necessary information from edge server $e \in E$ derived as follows:

$$e = \text{argmin}\{d^{(t)}_{ne'}, \forall e' \in E\} \tag{13}$$

where $d^{(t)}_{ne'}$ is the physical distance between mobile user u_n and edge server $e' \in E$ at time slot t. The information received from a nearby edge server includes the average utility $\bar{\varphi}_i$, power consumption $\bar{P}_i$ and the physical distance d_i to each unvisited region $r_i \in \{\mathcal{R} - \mathcal{R}^{(t)}_n\} \cap \mathcal{R}_e$. Note that $\mathcal{R}_e$ indicates the set of RoIs which edge server e covers.

The MU needs to precompute the achievable weighted social welfare from every unvisited RoI i in real-time to visit the next (best) RoI. However, the number of time slots at which sensors in each RoI i transmit data to the MU $(d_s - a_s)$ is not known in advance. Thus, the MU must approximately estimate this information beforehand. In UO-DCA, an MU uses the following approximation to estimate the time that it should spend to collect data from the selected sensors in such a region ($t^{(i)}_s$) and transmit it to a nearby edge server ($t^{(i)}_e$):

$$t^{(i)}_s + t^{(i)}_e \approx \frac{\bar{\varphi}_i/(d_i \cdot \bar{P}_i)}{\sum_j \bar{\varphi}/(d_j \cdot \bar{P}_j)}(D_n - t - t_i),\ \forall j \in \mathcal{R} - \mathcal{R}^{(t)}_n \tag{14}$$

Here t_i, the time to visit RoI i, is known in advance and announced by the edge server to the MUs. From Eq. (14), the higher the utility φ in region i compared to other regions $j \in \mathcal{R}_e$ and the smaller the transmission power (i.e., energy consumption) of the sensors in the region, the longer the time considered by the MU for data collection and transmission there. We further note that Eq. (14) does not require the MU to know individual sensors' data utility or power information.

Estimating sensor profit. In our system, a MU considers collecting data from a fraction of sensors at each RoI for fairness purposes, as well as to obtain heterogeneous yet valuable data from all RoIs within the deadline. More specifically, let m_i be the number of sensors in region r_i announced by the edge server. UO-DCA approximates the number of sensors selected for data transmission in region i (i.e., for which $y^{(t)}_{nsr} = 1$) as $\lceil (m_i / \sum_j m_j) \cdot m_i \rceil$ for $\forall j \in \mathcal{R} - \mathcal{R}^{(t)}_n$, i.e., the fraction of sensors selected is proportional to the fraction of sensors located in region i compared to other regions. Candidate sensors in a region transmit all their data to an MU; upon receiving such data, the MU then transmits them sequentially (i.e., in consecutive time slots) to the nearest edge server. Thus, the time period during which the MU collects data from selected sensors in region r_i follows from Eq. (14):

$$t^{(i)}_s = \frac{\frac{\bar{\varphi}_i/(d_i \cdot \bar{P}_i)}{\sum_j \bar{\varphi}_j/(d_j \cdot \bar{P}_j)}(D_n - t - t_i)}{\lceil (m_i / \sum_j m_j) \cdot m_i \rceil + 1},\ \ \forall j \in \mathcal{R} - \mathcal{R}^{(t)}_n \tag{15}$$

Note that the approximation above is obtained under the assumption that the selected sensors in region r_i send their data to the MU with equal time duration. With no change in data utility or sensors power consumption, the MU spends less time for data collection as the number of sensors in the RoI increases to collect valuable data fairly from unvisited RoIs in its traversal. The revenue of the sensors in region r_i is then:

$$\text{Revenue}_\text{S}(i) \approx (\lceil (m_i / \sum_j m_j) \cdot m_i \rceil) \cdot t^{(i)}_s \cdot \bar{R}_i \tag{16}$$

where $t^{(i)}_s$ is given in Eq. (15) and $\bar{R}_i$ is the corresponding average reward that should be paid to the sensors in region r_i for collecting their data. Finally, the cost associated with sensors in region r_i is approximated as follows:

$$\text{Cost}_\text{S}(i) \approx (\lceil (m_i / \textstyle\sum_j m_j) \cdot m_i \rceil) \cdot a \cdot (t^{(i)}_s \cdot (p_s + \bar{P}_i)) \tag{17}$$

Estimating MU profit. Similarly, the revenue that an MU obtains by visiting region r_i is given by:

$$\text{Revenue}_\text{MU}(i) \approx (\lceil (m_i / \sum_j m_j) \cdot m_i \rceil) \cdot t^{(i)}_s \cdot \bar{I}_i \tag{18}$$

where $\bar{I}_i$ is the average reward that the cloud server pays to the MU for sending the collected sensory data in region r_i. The corresponding cost for an MU is approximated as follows:

$$\text{Cost}_{\text{MU}}(i) \approx b \cdot (t_i \cdot p_n) + (\lceil (m_i / \sum_j m_j) \cdot m_i \rceil) \cdot t_s^{(i)} \cdot (\bar{R}_i + c\bar{P}_{ni}) \tag{19}$$

where $\bar{P}_{ni}$ is the average power to transmit the data collected by the MU from sensors in region i to a nearby edge server.

Choosing the RoI to visit. Given the above, the region that an MU selects (at time t) to visit next is obtained as follows:

$$c = \underset{\forall j \in \mathcal{R} - \mathcal{R}_n^{(t)}}{\text{argmax}} \left\{ (u_j) \wedge t_s^{(j)} > 0 \right\} \tag{20}$$

where u_j, the local achievable weighted social welfare (Eq. (6)) when visiting region r_j, comes from Eqs. (16–19):

$$u_j = \gamma(\text{Revenue}_{\text{S}}(j) - \text{Cost}_{\text{S}}(j)) + (1-\gamma)(\text{Revenue}_{\text{MU}}(j) - \text{Cost}_{\text{MU}}(j)) \tag{21}$$

Once the MU moves to the selected region r_c, the sensors in this RoI send their data for the time duration of $t_s^{(c)}$ within $\lceil t_s^c / \Delta t \rceil$ time slots. At each time slot, the set of sensors which are selected for data transmission to MU is determined as follows. First, the MU sorts all sensors in the region in decreasing order of φ/P values at the current time slot and stores them in set S. Then, the first $\lceil (m_i / \sum_j m_j) \cdot m_i \rceil$ sensors from set S which have the highest φ/P values and satisfy the conditions in Eqs. (9)–(10) are selected for data transmission to the MU at the current time slot. Upon collecting and transmitting the data of sensors in region r_c to the edge server, the MU updates its remaining time and executes the same above-mentioned heuristic to find the next RoI to visit in its traversal. The MU, at every time slot, evaluates whether the remaining time is sufficient to visit at least one more RoI. If not, the MU does not accept the subsequent data collection task.

UO-DCA is described in Algorithm 1. Although the algorithm refers to a certain mobile user at a given time slot, the same procedure is executed by all MUs at different time slots.

Complexity Analysis. The following provides an analysis of UO-DCA, starting from its time and message complexity.

Theorem 3. *UO-DCA has a time complexity of* $O(K(\frac{K+1}{2} + M\log(M)))$ *and a message complexity of* $O(K(E+M))$*, where* K *is the number of RoIs,* E *is the number of edge servers, and* $M = \max(|S_k|, 1 \leq k \leq K)$ *is the maximum number of sensors per RoI.*

Proof: The complexity analysis is done for one MU; however, it applies to all the MUs in the system. In the worst case, an MU visits all RoIs within the specified deadline for a data collection task. For each unvisited RoI, the MU first computes the achievable profits and the associated costs in $O(1)$ time (line 3), for a total time complexity of $O(i)$ with i unvisited RoIs at a certain time slot. After the visit, the MU sorts the sensors in decreasing order of their φ/P values

Algorithm 1: UO-DCA for mobile user $u_n \in \mathcal{U}$ at time slot $1 \leq t \leq |T|$

```
1  Determine edge server e according to Eq. (13);
2  foreach region r_i ∈ {R − R_n^(t)} ∩ R_e do
3      Compute Profit_S(i), Cost_S(i), Profit_MU(i) and Cost_MU(i)
         [Eqs. (16)–(19)];
4  Find the target region r_c according to Eq. (20);
5  Move to region r_c within t_c time duration;
6  foreach time slot t + t_c ≤ t' ≤ t + t_c + ⌈t_s^c/Δt⌉ do
7      Sort sensors in region r_c in decreasing order of their φ/P
         values at time slot t' into set S;
8      Select first ⌈(m_c/Σ_j m_j) · m_c⌉, ∀j ∈ R − R_n^(t) sensors
         from set S such that Profit_S(s') > 0 and Profit_MU(n) > 0
         for each selected sensor s';
9      Collect data for Δt time from the selected sensors;
10 Transmit collected data from selected sensors in region r_c
     sequentially to a nearby edge server;
11 Remove region r_c from the set of unvisited RoIs;
12 Set current time slot to t_cur = t + (φ̄/(d_c·P̄_c)) / (Σ_j φ̄_j/(d_j·P̄_j)) · (D_n − t);
13 Update nearby edge server e according to Eq. (13);
14 if t_cur + min{t_r, ∀r ∈ {R − R_n^(t_cur)} ∩ R_e} > D_n then
15     Terminate data collection by mobile user u_n;
16 else Repeat for mobile user u_n at time slot t_cur ;
```

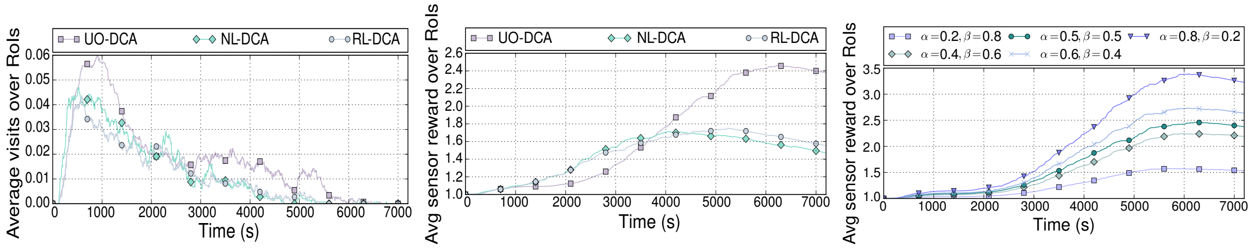
(line 7) with a worst case time complexity of $O(M\log(M))$, where $M = \max(|S_k|, 1 \leq k \leq K)$ is the maximum number of sensors in the RoI. The MU can visit all RoIs within the deadline, therefore, the worst case time complexity of UO-DCA is $O(K(\frac{K+1}{2} + M\log(M)))$ for K RoIs.

A MU must discover and communicate with the nearest edge server to select the subsequent RoI to visit at each time slot. Such a communication involves sending $O(E)$ messages, where E is the maximum number of edge servers in the sensing area, and receiving $O(1)$ messages in reply. Upon visiting the selected region, the MU receives data from at most M sensors with a worst-case message complexity of $O(M)$. Again, discovering the nearest edge server and transmitting collected data take $O(E)$ and $O(1)$ messages, respectively. The MU visits at most K regions in the worst case; thus, the overall message complexity of UO-DCA is given by: $M_{UO-DCA} \in O(K(E+M))$. ■

V. Evaluation

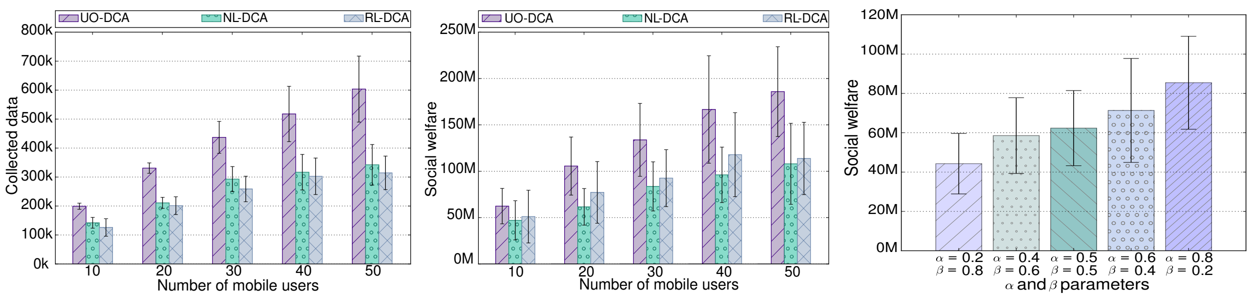
A. Simulation Setup

The considered urban IoT scenario is represented by a varying number of MUs and 200 sensors randomly deployed in a metropolitan area of $4,480$ by $3,500$ meters. All sensors had a transmission range of 100 m. The area is further divided into square regions of size 140 by 140 meters (i.e., inscribed in a circle with a radius of 99 meters) and 24 edge servers cover not-overlapping subsets of such regions. The ONE simulator v.1.6.0 was employed to generate mobility traces based on the roads in the city of Helsinki and pedestrians walking with a speed between 0.5 and 1.5 m/s along streets as well as pedestrian paths. The sampling frequency of a sensor expresses the utility of sensory data. Specifically, data are obtained



(a) (b) (c)

Fig. 2: (a) Average number of visits to RoIs as well as average sensor reward over RoIs (b) for the considered algorithms and (c) as a function of the α and β parameters for UO-DCA as a function of simulation time with 10 MUs in the network.



(a) (b) (c)

Fig. 3: (a) Average collected data as well as social welfare of the system (b) as a function of the number of MUs for the considered algorithms and (c) as a function of the α and β parameters [Eq. (1)] for UO-DCA with 10 MUs in the network.

with a random sampling frequency between $f_{min} = 10$ and $f_{max} = 100$ Hz and their utilities fall in the range $[0, 1]$.

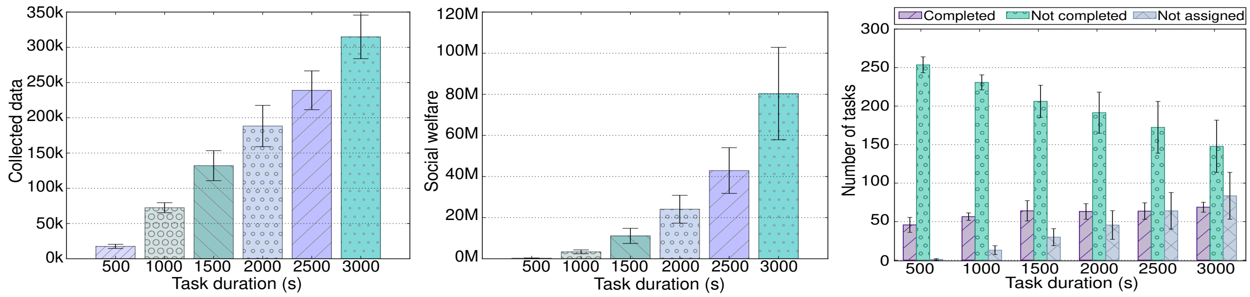
The parameters α and β in Eq. (1) are set to 0.5, unless otherwise stated. Also, we set the weighting parameter in the objective function [namely, Eq. (6)] to $\gamma = 0.5$ to achieve a fair share of profits between MUs and sensors. The system comprises 300 tasks generated with a minimum time duration of 500 seconds (i.e., $D_n - A_n \geq 500$ s), unless otherwise stated. The length of the time slot was set to one second and the experiments lasted for 2 hours of simulated time. The figures report the average values over ten runs along with the related standard deviations as error bars when meaningful.

B. Trace-based Simulations

A custom Python simulator was employed to assess the performance of the proposed mobile crowdsourcing data collection scheme UO-DCA (i.e., Algorithm 1) against two other schemes: *Nearest Location Data Collection Algorithm* (NL-DCA), where an MU selects to visit the region with the shortest distance and collects data in such a region from candidate sensors similar to UO-DCA; and *Random Location Data Collection Algorithm (RL-DCA)*, where a mobile user *randomly* selects to visit a region and collects data in such a region from candidate sensors similarly to UO-DCA.

Comparison of Considered Algorithms. Fig. 2a shows the average number of visits to all RoIs over time for three schemes: UO-DCA, NL-DCA, and RL-DCA. Initially, there are zero visits (as no tasks have been generated yet), the number increases over time, and then it decreases by the end of the simulation. That is because most of the tasks are assigned for completion at the beginning of the simulation time, making the MUs occupied, with less time available to complete other tasks. Moreover, there are fewer tasks whose starting time falls by the end of the simulation time. Fig. 2b shows how the average reward over all sensors varies over time due to MUs visiting the sensors in the respective RoIs. The parameters α and β in Eq. (1) are set to 0.5 and R_s at time $t = 0$ is set to 1. The average sensor reward increases as more MUs visit RoIs, while it decreases more slowly as the visits to RoIs drop. Hence, the reward model of the sensors in Eq. (1) retains past rewards and slowly adapts over time to new ones. Overall, UO-DCA yields a higher number of visits to RoIs and thus a higher reward for the sensors.

Fig. 3a shows the average data collected by all sensors as a function of the number of MUs. The average data collected with the three algorithms increases with the number of MUs – intuitively, more MUs collect more data. The proposed UO-DCA outperforms both NL-DCA and RL-DCA. This gap increases with the number of MUs in the system. NL-DCA outperforms RL-DCA as choosing to visit the nearest RoI results in lower (e.g., travel) costs for the MUs; at the same time, it results in more time available to visit a multitude of (other) RoIs, thereby collecting more data. Fig. 3b shows the average social welfare of the system as a function of the number of MUs. Similarly, UO-DCA yields higher social welfare compared to both NL-DCA and RL-DCA. Moreover, the social welfare is slightly higher for RL-DCA compared to NL-DCA. This seemingly counter-intuitive result is because collecting data from the nearest RoIs does not yield maximum profit for the sensors and the MUs.



(a) (b) (c)

Fig. 4: Average (a) data collected over time, (b) social welfare of the system, and (c) number of tasks that are completed, (assigned but) not completed, and not assigned as a function of the task duration for UO-DCA with 20 MUs in the network.

Impact of Weights α and β. Fig. 2c and Fig. 3c show the average sensors' reward and the social welfare of the system as a function of the α and β parameters [Eq. (1)]. Recall that the α parameter weighs the number of visits to RoIs, whereas the β parameter weighs the data utility in the reward that sensors request for their data (see Section V-A). As α increases from 0.2 to 0.8 (while β decreases from 0.8 to 0.2), the sensors request a higher reward for their data as they get more visits by MUs. Similarly, the social welfare of the system in Fig. 3c increases with α, as it is the dominant parameter. Thereby, the reward paid to sensors heavily contributes to the social welfare. Intuitively, one would set a high α parameter to increase the social welfare. However, this incurs two main drawbacks: first, by increasing α, the *locality* of the data (a dimension of utility) decreases (i.e., a low β); second, a high social welfare reflects a high cost for data consumers to buy and access sensory data (see Fig. 1). With α dominating such system dynamics, the different stakeholders of a MCS such as data consumers, MUs, and (owners of) sensors trade-off their costs and profits. That is, setting a high α value benefits the sensors as it increases their reward and the social welfare of the system; however, the utility of data decreases and data consumers pay a higher price for sensory data. By contrast, the average visits to RoIs and the amount of collected data (not shown here) do not vary with the α and β parameters.

Impact of Task Duration. Fig. 4a shows the average collected data as a function of the time duration of the tasks (i.e., $\delta = D_n - A_n$) in a network with 20 MUs for UO-DCA. The collected data increases rapidly with δ. Recall that the size of an RoI is 140 by 140 meters and pedestrians walk with a speed between 0.5 and 1.5 m/s. MUs that are assigned a task would either collect data from the RoI they reside in, or attempt to reach another RoI that has not yet been visited for the given task and that yields positive profit. In fact, given the size of a RoI, the amount of collected data is very low for values of δ that are comparable to the time needed to traverse it. As δ increases, MUs have more time to visit new RoIs and collect data from them, resulting in a higher amount of collected data. Moreover, a higher locality of the phenomena (i.e., higher accuracy of the data) requires smaller RoI sizes. Thereby, tasks with smaller time duration δ can be assigned to MUs, making δ a crucial design parameter of a MCS. Similar to Fig. 4a, the results in Fig. 4b present an identical pattern: the social welfare increases with δ.

Fig. 4c shows the distribution of tasks during one data collection campaign of 2 hours as a function of the time duration δ of the tasks. The campaign consists of 300 tasks. These tasks can be (i) assigned and *completed* successfully by an MU, (ii) assigned to an MU but *not completed*, as it is not feasible to reach an RoI in terms of time, or the feasible RoI yields negative social welfare for the system, or (iii) *not assigned* as all MUs are occupied completing other tasks. The number of completed tasks is low for a small δ, especially for $\delta = 500$ s, as MUs are able to reach very few feasible RoIs. In most cases, tasks are initially assigned to MUs who immediately drop them, as it is not feasible to complete the tasks. The number of tasks (assigned but) not completed decreases with δ. At the same time, the number of tasks that are completed successfully increases, as MUs have more time to visit feasible RoIs. Moreover, MUs carrying out longer tasks lead to an increased number of not assigned tasks.

VI. Related Work

Opportunistic data collection in the IoT has received considerable attention in the last few years [2]. Casadei et al. [26] model opportunistic IoT services using the aggregate computing approach. Kortoci et al. [27] leverage the fog networking paradigm and devise a protocol that offloads data sampled by storage-constrained sensors to mobile gateways. Fadda et al. [28] consider task assignment with the goal to minimize costs while covering all sensors in a certain area. However, none of these solutions explicitly considers incentives for user participation in data collection, as addressed in this work.

Many works in mobile crowd sourcing [11, 12, 29] target outsourcing sensory data collection to the public crowd by focusing on service and data quality. Instead, [13] addresses the truthfulness and integrity of sensory data, while other works [9, 14, 16, 18, 30] focus on time- and location-dependent tasks. Similarly, our work considers the collection of "valuable" time-sensitive data, where any mobile user can complete a task if they are satisfied with the expected profit. However, we leverage the concept of utility to characterize how sensory data are valuable. In addition, [31] considers an incentive-aware time-sensitive data collection scheme whose focus is users' cooperation to relay data among them, to ultimately reach a data requestor. By contrast, our solution requires no explicit user cooperation, but only the willingness of a user to collect data.

Several works design incentive and pricing mechanisms to ensure user participation [15, 32] while reducing their sensing effort [9, 10, 16, 32]. While we similarly account for the cost to complete a task, we focus on the *user-specific* cost incurred by modifying their route and selectively incentivize them to collect valuable data. Moreover, our work provides users with expected cost and revenue values prior to task completion, as opposed to [15, 32]. Similar to [17], we focus on trade-offs between task quality and completion cost, while emphasizing the dynamics of a time-varying price for sensor data that accounts for network dynamics such as utility and frequency of user visits to the sensors.

VII. Conclusion

IoT applications in urban scenarios can benefit from mobile crowd sourcing as long as sensory data are properly collected and transmitted to the cloud for further processing. In such a context, this work devises an incentive-based solution in which individual sensors charge mobile users a dynamic virtual price to collect their data. Upon accepting a sensing task and the corresponding compensation, a user decides which sensors to visit based on the task's service quality requirements and its own costs relative to task completion. By encoding the quality of the sensory data and frequency of user visits into the virtual price, each sensor dynamically incentivizes users to either collect data or not, depending on sensor's data having been recently collected by others. The proposed incentive-based scheme accounts for such dynamics and significantly increases the amount of collected data by up to 70% compared to other baseline approaches, while simultaneously yielding a higher social welfare by up to 60% for all system's stakeholders.

As a future work, we seek to implement the proposed solution on top of an existing mobile crowdsourcing platform.

Acknowledgment

This work was partially supported by: the Academy of Finland under grants 299222, 319710, and 326346; and the US National Science Foundation under grant CNS-1751075.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
[2] N. Fernando, S. W. Loke, I. Avazpour, F.-F. Chen, A. B. Abkenar, and A. Ibrahim, "Opportunistic Fog for IoT: Challenges and Opportunities," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8897–8910, 2019.
[3] X. Kong, X. Liu, B. Jedari, M. Li, L. Wan, and F. Xia, "Mobile Crowdsourcing in Smart Cities: Technologies, Applications, and Future Challenges," *IEEE IoT Journal*, vol. 6, no. 5, pp. 8095–8113, 2019.
[4] G. Premsankar, B. Ghaddar, M. Sabicki, and M. Di Francesco, "Optimal configuration of LoRa networks in smart cities," *IEEE Transactions on Industrial Informatics*, vol. 16, pp. 7243–7254, December 2020.
[5] S. Yang, U. Adeel, Y. Tahir, and J. A. McCann, "Practical opportunistic data collection in wireless sensor networks with mobile sinks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 5, pp. 1420–1433, 2016.
[6] Y.-C. Tseng, F.-J. Wu, and W.-T. Lai, "Opportunistic data collection for disconnected wireless sensor networks by mobile mules," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1150–1164, 2013.
[7] L. Luo, C. Huang, T. Abdelzaher, and J. Stankovic, "Envirostore: A cooperative storage system for disconnected operation in sensor networks," in *IEEE INFOCOM 2007*, pp. 1802–1810, IEEE, 2007.
[8] C. Miao *et al.*, "Balancing quality and budget considerations in mobile crowdsourcing," *Decision Support Systems*, vol. 90, pp. 56–64, 2016.
[9] L. Gao, F. Hou, and J. Huang, "Providing long-term participation incentive in participatory sensing," in *Computer Communications (INFOCOM), 2015 IEEE Conference on*, pp. 2803–2811, IEEE, 2015.
[10] H. Jin *et al.*, "CENTURION: Incentivizing multi-requester mobile crowd sensing," in *IEEE INFOCOM 2017*, pp. 1–9, 2017.
[11] L. Pu *et al.*, "Crowdlet: Optimal worker recruitment for self-organized mobile crowdsourcing," in *IEEE INFOCOM 2016*, pp. 1–9, 2016.
[12] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: Incentive mechanism design for mobile phone sensing," in *MobiCom 2012*, pp. 173–184, 2012.
[13] H. Jin *et al.*, "Theseus: Incentivizing truth discovery in mobile crowd sensing systems," in *ACM MobiHoc 2017*, ACM, 2017.
[14] Z. Feng, Y. Zhu, Q. Zhang, L. M. Ni, and A. V. Vasilakos, "TRAC: Truthful auction for location-aware collaborative sensing in mobile crowdsourcing," in *IEEE INFOCOM 2014*, pp. 1231–1239, 2014.
[15] G. Hui, H. L. Chi, T. Jian, Y. Dejun, H. Pan, and W. Wendong, "Online quality-aware incentive mechanism for mobile crowd sensing with extra bonus," *IEEE Transactions on Mobile Computing*, 2018.
[16] S. He, D.-H. Shin, J. Zhang, and J. Chen, "Toward optimal allocation of location dependent tasks in crowdsensing," in *INFOCOM, 2014 Proceedings IEEE*, pp. 745–753, IEEE, 2014.
[17] W. Gong, B. Zhang, and C. Li, "Task Assignment in Mobile Crowdsensing: Present and Future Directions," *IEEE Network*, vol. 4, no. 32, pp. 1–8, 2018.
[18] Y. Tong, Z. Zhou, Y. Zeng, L. Chen, and C. Shahabi, "Spatial crowdsourcing: a survey," *The VLDB J.*, vol. 29, no. 1, pp. 217–250, 2020.
[19] R. Zhou, Z. Li, and C. Wu, "A Truthful Online Mechanism for Location-Aware Tasks in Mobile Crowd Sensing," *IEEE Transactions on Mobile Computing*, 2017.
[20] M. C. Bor, U. Roedig, T. Voigt, and J. M. Alonso, "Do LoRa Low-Power Wide-Area Networks Scale?," in *Proc. 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, (New York, NY, USA), pp. 59–67, ACM, 2016.
[21] X. Zhang, Z. Yang, W. Sun, Y. Liu, S. Tang, K. Xing, and X. Mao, "Incentives for mobile crowd sensing: A survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 54–67, 2016.
[22] G. Anastasi, M. Conti, and M. Di Francesco, "Reliable and energy-efficient data collection in sparse sensor networks with mobile elements," *Performance Evaluation*, vol. 66, no. 12, pp. 791–810, 2009.
[23] N. Kaufmann, T. Schulze, and D. Veit, "More than fun and money. Worker Motivation in Crowdsourcing-A Study on Mechanical Turk.," in *AMCIS*, vol. 11, pp. 1–11, Detroit, Michigan, USA, 2011.
[24] H. Kellerer, U. Pferschy, and D. Pisinger, "Introduction to NP-Completeness of knapsack problems," in *Knapsack problems*, pp. 483–493, Springer, 2004.
[25] ETSI, "Mobile-Edge Computing Introductory Technical White Paper," Sep. 2014.
[26] R. Casadei *et al.*, "Modelling and simulation of opportunistic IoT services with aggregate computing," *Future Generation Computer Systems*, vol. 91, pp. 252–262, 2019.
[27] P. Kortoçi, L. Zheng, C. Joe-Wong, M. Di Francesco, and M. Chiang, "Fog-based data offloading in urban IoT scenarios," in *Proc. of IEEE INFOCOM 2019*, pp. 784–792, May 2019.
[28] E. Fadda, G. Perboli, and R. Tadei, "Customized multi-period stochastic assignment problem for social engagement and opportunistic IoT," *Computers & Operations Research*, vol. 93, pp. 41 – 50, 2018.
[29] M. Karaliopoulos, I. Koutsopoulos, and M. Titsias, "First learn then earn: Optimizing mobile crowdsensing campaigns through data-driven user profiling," in *Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 271–280, ACM, 2016.
[30] Y. Tong, L. Chen, and C. Shahabi, "Spatial crowdsourcing: Challenges, techniques, and applications," *Proceedings of the VLDB Endowment*, vol. 10, no. 12, pp. 1988–1991, 2017.
[31] Y. Zhan, Y. Xia, Y. Liu, F. Li, and Y. Wang, "Incentive-aware time-sensitive data collection in mobile opportunistic crowdsensing," *IEEE Trans. on Vehicular Technology*, vol. 66, no. 9, pp. 7849–7861, 2017.
[32] Y. Zhan, Y. Xia, J. Zhang, and Y. Wang, "Incentive mechanism design in mobile opportunistic data collection with time sensitivity," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 246–256, 2018.