CHASE: Charging and Scheduling Scheme for Stochastic Event Capture in Wireless Rechargeable Sensor Networks

Haipeng Dai, *Member, IEEE,* Qiufang Ma, Xiaobing Wu, Guihai Chen, *Member, IEEE,* David K. Y. Yau, *Senior Member, IEEE,* Shaojie Tang, *Member, IEEE,* Xiang-Yang Li, *Fellow, IEEE,* and Chen Tian, *Member, IEEE*

Abstract—In this paper, we consider the scenario in which a mobile charger (MC) periodically travels within a sensor network to recharge the sensors wirelessly. We design joint charging and scheduling schemes to maximize the Quality of Monitoring (QoM) for stochastic events, which arrive and depart according to known probability distributions of time. Information is considered captured if it is sensed by at least one sensor. We focus on two closely related research issues, *i.e.*, how to choose the sensors for charging and decide the charging time for each of them, and how to schedule the sensors' activation schedules according to their received energy. We formulate our problem as the maximum QoM CHArging and SchEduling problem (CHASE).
We first ignore the MC's travel time and study the resulting relaxed version of the problem, which we call CHASE-R. We show that both CHASE and CHASE-R are NP-hard. For CHASE-R, we prove that it can be formulated as a submodular function maximization problem, which allows two algorithms to achieve $1/6$- and $1/(4+\epsilon)$-approximation ratios. Then, for CHASE, we propose approximation algorithms to solve it by extending the CHASE-R results. We conduct simulations to validate our algorithm design.

Index Terms—mobile charging, scheduling, wireless rechargeable sensor network, stochastic event capture, submodular optimization, approximation algorithm.

1 INTRODUCTION

TRADITIONAL wireless sensor networks (WSNs) are constrained by limited battery energy that powers the sensors. Their limited network lifetime is considered a major deployment barrier. Besides, in many applications sensors are located in hazardous or inaccessible areas such as volcanoes [1], inside concrete walls [2], or at the bottom of bridges [3], which makes battery-swapping schemes [4]–[6] unsafe, infeasible, labor-intensive, or costly. To extend the network lifetime, many approaches have been proposed to harvest environmental energy such as solar [7], vibration [8], and wind [9]. However, a limitation of existing energy-harvesting techniques is that it is highly dependent on the ambient environment, which makes the harvesting rate highly unpredictable. The problem can be overcome by recent breakthroughs in wireless power charging technologies [10], which allow energy to be transferred from one storage device to another wirelessly with reasonable efficiency. For example, magnetic resonance coupling is shown to transfer 60 watts [10] at an efficiency of 90% to about 30% when the distance varies from $0.75\,m$ to $2.25\,m$. Since wireless recharging can guarantee a required level of power supply, independent of the ambient environment, and it is contactless, it has found many applications including smart grids [11], body sensor networks [12], and civil structure monitoring [3].

Because power chargers are expensive, it is generally not cost-effective to deploy a large number of them statically for energy provisioning. Instead, existing practical approaches focus on using a mobile charger (MC) to move around the sensors and charge them in turn during a travel schedule, for tasks such as routing [13]–[15] and gathering data [16]. None of these prior efforts have solved the problem of stochastic event capture, however, in key aspects such as scheduling sensors' duty cycles to maximize their ability to capture interesting events of a probabilistic nature. But the problem is fundamental in wireless sensor network design, and it has received attention for both cases of traditional WSNs [17]–[19] and wireless ambient-energy harvesting sensor networks [20], [21]. In this paper, we optimize event capture in a network of sensors wirelessly recharged by an MC. We assume that stochastic events arrive and depart according to known time distributions. An event is said to be captured if it is sensed by at least one sensor.

Note that there are existing practical system platforms that can enhance the performance of event monitoring by wireless recharging. For example, the Wireless Identification and Sensing Platform (WISP) has been applied in individual activity recognition, large-scale urban sensing [22], [23], and structural health monitoring (SHM) [3]. In the SHM application, the civil structure

- *H. Dai, Q. Ma, G. Chen, C. Tian are with the State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China E-mail: haipengdai@nju.edu.cn, mg1633053@smail.nju.edu.cn, gchen@cs.sjtu.edu.cn, tianchen@nju.edu.cn*
- *X. Wu is with Wireless Research Centre, University of Canterbury, New Zealand. E-mail: barry.wu@canterbury.ac.nz*
- *D.K.Y. Yau is with the Information Technology Systems and Design pillar, Singapore University of Technology and Design, Singapore. E-mail: david_yau@sutd.edu.sg*
- *S. Tang is with Naveen Jindal School of Management, University of Texas at Dallas, 800 W. Campbell Road, Richardson, Texas, USA. E-mail: shaojie.tang@utdallas.edu*
- *X. Li is with School of Computer Science and Technology, University of Science and Technology of China, Hefei 230026, China. E-mail: xiangyangli@ustc.edu.cn*

Corresponding author: G. Chen (E-mail: gchen@cs.sjtu.edu.cn). *Manuscript received April 19, 2005; revised August 26, 2015.*

is instrumented with sensor nodes capable of being powered solely by energy transmitted wirelessly to them by a mobile helicopter. Jiang *et al.* [24] are the first to exploit wireless power charging by MCs for efficient stochastic event capture. Their objective is to jointly determine the MCs' movement schedule and the sensors' activation schedule to maximize the Quality of Monitoring (QoM) [18], [20], defined as the average information gained per event by the network. They make several simplifying assumptions: each sensor can only monitor one Point of Interest (PoI), the charging time for each sensor is identical, the event staying time follows an Exponential distribution, and all the sensors follow a simple (q, p) periodic schedule (*i.e.*, the sensors monitor the PoIs for q time every p time). We relax these assumptions in this paper.

In this paper, we consider the scenario in which an MC periodically travels within a sensor network and recharges the sensors wirelessly to enable them to perform tasks of stochastic event capture. We assume that the MC repeats its recharge schedule every period of time τ, and that the schedule (counting both the charging time and travel time of the MC) must complete within time τ_w ($\tau_w < \tau$). For example, the MC is carried by an UAV whose daily shift is from 9am to 11am only (here, $\tau_w = 2\ hour$ and $\tau = 24\ hour$; typically, τ can be a few weeks or even longer), or the MC must be withdrawn for maintenance for some amount of time between recharge schedules.

We address two closely related issues in the wireless recharging and event monitoring. The first is how to choose the sensors for recharging and further decide the charging time for each of them, constrained by the MC's working time τ_w. The second is how to best schedule the sensors' activations based on their received energy, considering that nearby sensors may cover overlapping PoIs. Our goal is to jointly design a charging scheme for the MC and the sensors' activation schedules to maximize the QoM of the stochastic event capture. We define our problem formally as the maximum QoM CHArging and SchEduling problem (CHASE). The coupling between the MC's travel time and the sensor charging time makes the problem highly challenging. Hence, we first ignore the travel time and study the resulting relaxed version of CHASE, which we call CHASE-R. Then, based on the CHASE-R results, we develop solutions for the general CHASE problem.

The main contributions of this paper are as follows:

- We analyze the QoM of stochastic event capture, which admits the possibility that the same PoI is monitored by multiple sensors. We formulate the CHASE and CHASE-R problems, and show that both are NP-hard.
- We reformulate CHASE-R as a monotone submodular function maximization problem under a special sufficient condition. This reformulation of CHASE-R allows two algorithms to achieve $1/6$-approximation and $1/(4+\epsilon)$-approximation, respectively.
- Based on the CHASE-R results, we propose two approximation algorithms for CHASE, when the MC's travel time is considered.
- Importantly, all of the proposed CHASE-R and CHASE solutions are sufficiently general to accommodate diverse activation schedules, event utility functions, and probability distributions of the event staying times.
- We conduct extensive simulations to verify our analytical findings. Simulation results show that our schemes outperform the state of the art.

The remainder of the paper is organized as follows. Section 2 reviews related work. In Section 3, we present preliminaries and a formal definition of the CHASE problem, as well as its relaxed version CHASE-R. In Section 4, we analyze the complexity of the problems and reformulate a special case of the relaxed problem as a monotone submodular function optimization problem. Then, we present approximation algorithms for the relaxed problem and the original problem. Section 5 presents extensive simulations to verify our theoretical results. Section 6 concludes.

2 Related Work

In this section, we mainly review related work on mobile charging, which generally can be classified into three categories.

First, prior work has investigated the energy efficiency of mobile chargers (MCs) [25]–[31]. For example, Wang *et al.* [25] proposed to coordinate multiple MCs to minimize their aggregate travel distance while guaranteeing continuous operation of each sensor, such that the overall energy efficiency is optimized. In a later work [26], their goal is to maximize the difference between the energy harvested by all the sensors and the travel energy expended by all the MCs. Zhang *et al.* [27] presented an optimal scheme for multiple energy-constrained MCs to charge a linear WSN, where the ratio of energy received by all the sensors to the travel energy expended by all the MCs is maximized. Dai *et al.* [30], [31] studied the problem of using minimum number of MCs to keep their jointly charged WSN running forever. Wu *et al.* [32] took sensor placement into consideration while determining charging plans of multiple MCs.

Second, the service delay of MCs has been considered [33]–[36]. Fu *et al.* [33] minimized the overall charging delay of a single MC by planning its charging route and strategy. He *et al.* [34], [35] considered the charging problem when the charging requests of sensors arrive in a dynamic fashion. Their work has been extended to scenarios where the sensors are also mobile [36].

Third, research has addressed network performance issues under mobile charging, from perspectives such as data routing, data collection, sensing coverage, and event monitoring [13]–[16], [24], [37]–[41]. For data routing, Tong *et al.* [13] examined the impact of mobile charging on data routing and WSN deployment. Their work has been expanded [15] to address several realistic issues (*e.g.*, the communication environment is dynamic and unreliable, the charging capacity of an MC is limited, and the sensors are heterogeneous) by jointly considering data routing for the sensors and the charging scheme for an MC. Use of mobile charging to improve data collection in WSNs has also received significant attention. Shi *et al.* [14], [37] applied a single MC to improve the data collection, while reducing the working time within a charging time period. In [16], [38]–[40], MCs are used not only as energy providers but also as data collectors. Zhou *et al.* [41] solved the problem of scheduling an MC to charge sensors to maintain k-coverage in the network at low cost for the MC. Jiang *et al.* [24] are first to investigate the impact of mobile charging on efficient stochastic event capture. However, they make several simplifying assumptions, which limit the practicality of their results. We relax these assumptions in our conference version of this paper [42]. Wu *et al.* [43] proposed a scheme for scheduling multiple MCs to serve collaborative tasks of sensors.

Besides, there also exist a large body of related work on scheduling issues in wireless chargers networks where chargers are static, such as [44]–[57]. We omit detailed description for them as they are quite different from ours.

TABLE 1: Notations

Symbol	Meaning
o_i	Target i
v_i	Sensor i
O_i	Subset of PoIs covered by sensor v_i
V_i	Subset of sensors covering PoI o_i
$\mathcal{L}$	Length of sensor schedule
τ_{ts}	Time duration of a single time slot
S_i	Activation schedule of sensor v_i
$\widehat{S}_i$	Equivalent monitoring schedule for PoI o_i
w_i	Weight of PoI o_i
τ_w	Maximum working time in one charging period
τ	Time period of charging process
τ_i	Charging time allocated to sensor v_i in one charging period
P_c	Working power of MC
p_i	Working power of sensor v_i
E_i	Battery capacity of sensor v_i
η_i	MC's charging efficiency for sensor v_i
c_i	Charging time factor for sensor v_i
l_i	Active time slot budget of sensor v_i
ν_{MC}	MC's speed

3 Problem Statement

3.1 Network Model

We assume that there are m sensors $V = \{v_1, v_2, \ldots, v_m\}$ distributed over a two-dimensional region, which cover n Point of Interests (PoIs) denoted by $O = \{o_1, o_2, \ldots, o_n\}$. Let O_i represent the set of PoIs covered by sensor v_i. In a dense sensor network, typically close-by sensors may cover common PoIs. Hence, we assume that a target, say o_i, is covered by a subset of the sensors V_i. A summary of the notations in this paper is given in Table 1.

To prolong the lifetime of sensors, a mobile charger (MC) periodically starts from a base station (BS) and visits each of a selected subset of the static sensors $V_s \subseteq V$ exactly once, in order to charge the sensors wirelessly. At the end of the charging schedule, the MC returns to the BS. The total working time of the charging schedule, including the travel overhead and the charging time for all the selected sensors, must not exceed τ_w. Furthermore, we assume that the charging schedule is repeated every fixed period of time τ. Hence, the off-duty time for the MC at the BS is at least $\tau - \tau_w$. Note that under such a charging scheme, sensors not chosen for charging will die eventually. This is acceptable since our primary concern is to maximize the overall QoM under the constraint of limited charging time, rather than fairness among charging all the sensors.

We denote the path of the MC by $P = (\pi_0, \pi_1, \ldots, \pi_{|V_s|}, \pi_{|V_s|+1})$ where $\pi_0 = \pi_{|V_s|+1} =$ BS and $\{\pi_i\}_{i=1}^{|V_s|} = V_s$. Denote by t_{ij} the time required for the MC to move between sensor v_i and v_j. Suppose the movement of the MC between sensors is dictated by a motion planning scheme predetermined by the BS and/or the MC. The motion planning is assumed to respect physical constraints, such as following accessible pathways, avoiding obstacles, obeying mechanical limits on speeds and turns, etc, but its details are out of scope of this paper. We only require the motion planning to be predetermined and keep stable, such that t_{ij} is known a priori and it is fixed. To use as much time for charging as possible, the MC should travel on a shortest path P, given by $\arg\min_P \sum_{i=0}^{|V_s|+1} t_{\pi_i \pi_{i+1}}$, that completes a circuit of the sensors. For simplicity, we assume that such a path always exists. Finding the path can then be formulated as a Traveling Salesman Problem (TSP), which is NP-hard. We assume that some good approximation algorithm is used, and the approximate solution is given by $\tau_{TSP}(V_s)$. Moreover, we assume that the MC spends τ_i time for recharging the battery of v_i. Then, we have:

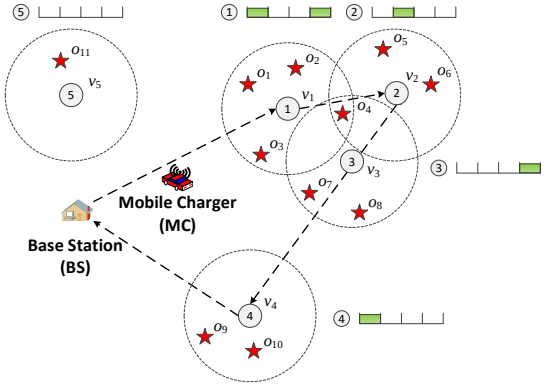


Fig. 1: Network model

$$\tau_{TSP}(V_s) + \sum_{v_i \in V_s} \tau_i \leq \tau_w. \quad (1)$$

The above inequality gives the MC's *working time constraint*. In this paper, we assume a discrete time model for a sensor's schedule, where the duration of a time slot is fixed and given. Specifically, every sensor follows a periodic schedule of identical length $\mathcal{L}$ (in time slots). In each time slot, a sensor can schedule itself to be active or inactive. Hence, we can express the *activation schedule* of sensor v_i by a vector $S_i = (a_{i1}, a_{i2}, \ldots, a_{i\mathcal{L}})$, where $a_{ij} = 1$ indicates that the sensor is active in slot j and $a_{ij} = 0$ indicates the opposite. We assume that the duration of a time slot, say τ_{ts}, is long enough such that the energy cost of turning the sensor on/off can be ignored.

The BS is responsible for determining the overall charging scheme, including the MC's travel path, the charging time allocated for each sensor, and the activation schedules of the sensors. It disseminates the activation schedules to the respective sensors either by a pre-established multi-hop communication mechanism, such as the Collection Tree Protocol (CTP) [58], or through the MC when the MC comes near each sensor for charging. We assume that the energy cost of disseminating the schedules can be ignored. This is because a schedule will not change except for exceptional events such as node breakdowns.

Fig. 2 shows an example of our network model. In this example, 5 sensors altogether cover 11 PoIs. The MC chooses a subset of the sensors $V_s = \{v_1, v_2, v_3, v_4\}$ to charge, and its travel path is $P = \{BS, v_1, v_2, v_3, v_4, BS\}$. The activation schedule of sensor v_1 in V_s is $S_1 = (a_{11}, a_{12}, a_{13}, a_{14}) = (1, 0, 0, 1)$, and that for v_2, v_3, and v_4 are $S_2 = (0, 1, 0, 0)$, $S_3 = (0, 0, 0, 1)$, and $S_4 = (1, 0, 0, 0)$, respectively.

3.2 Energy Consumption Model

Let P_c denote the working power of the MC, and p_i the working power of sensor v_i. Let η_i denote the MC's charging efficiency for sensor v_i, *i.e.*, the ratio of the amount of energy received by v_i to the amount of energy consumed by the MC. The charging efficiency can vary from sensor to sensor, and it depends on factors such as the distance between the corresponding sensor and the MC and the effectiveness of the sensor's antenna. We assume that the leakage power of each sensor is negligible, and each sensor will have used up its energy by the time of its next recharge (which

can be controlled by properly allocating the recharging time of the MC). Therefore, in one charging period of duration τ, the required working energy for sensor v_i under its activation schedule is $p_i \cdot \frac{||S_i||}{\mathcal{L}}\tau$, and it should be equal to the aggregated charged energy for v_i, *i.e.*, $P_c \cdot \tau_i \cdot \eta_i$. Thus, we have $\frac{p_i}{\eta_i P_c \mathcal{L}}\tau \cdot ||S_i||_1 = \tau_i$. For convenience, we define the *charging time factor* $c_i = \frac{p_i}{\eta_i P_c \mathcal{L}}\tau$, which is constant; it can be interpreted as the charging time required for the MC to provide sufficient energy for v_i to be active for one time slot. Hence, we have:

$$c_i \cdot ||S_i||_1 = \tau_i. \tag{2}$$

Further, denote by E_i the battery capacity of sensor v_i, and l_i the maximum number of active time slots sensor v_i can sustain using its limited battery capacity, which we call the *active time slot budget*. It is clear that $l_i = \frac{E_i}{p_i \tau}\mathcal{L}$. Because the total activated time slots in the activation schedule should not exceed the active time slot budget, we have:

$$||S_i||_1 \leq l_i, \tag{3}$$

which we call the *active time slot constraint*. If $l_i \geq \mathcal{L}$ for any sensor v_i, we can ignore the active time slot constraint. This situation occurs when the battery capacity is much larger compared to the working power of the sensor (such as ultra-capacitors [59]) or when the charging process is applied frequently. In general, however, the active time slot constraint should be considered; *e.g.*, when the batteries are of low cost and limited capacity.

3.3 Event Model and QoM Computation

In this section, we first present assumptions on the event dynamics and the properties of the sensors. Then, we propose a general paradigm to compute a PoI's QoM when it is monitored by one or more sensors.

3.3.1 Event Model

For event dynamics, we assume that events at a PoI occur one after another, and events at the same PoI or different PoIs are spatially and temporally independent [17] [18] [24]. After its occurrence, an event stays for some random time before it disappears. We denote by X the event staying time. Similarly, the time duration before the next event occurs, which we call the event inter-arrival time, is random and denoted by Y. Hence the sequence of event arrivals and departures forms a stochastic process. By renewable theory [60], the expected number of event arrivals in a time interval dt equals $\mu_i dt$, where $\mu_i = 1/E(Y)$ and $E(\cdot)$ denotes expectation. As for the event staying time X, we denote the probability density function of X by $f(x)$.

We use a binary sensing model for the sensors [61]. An event is said to be captured if it is sensed by at least one sensor. Assume that the j-th occurring event at PoI i is denoted as e_j^i, which is within range of a sensor for a total (but not necessarily contiguous) amount of time $t_j^i (t_j^i \geq 0)$. We assume that the sensor will, as a result, gain an amount of information $U_j^i(t_j^i)$ about e_j^i, where $U_j^i(x)$ is the utility function of e_j^i. There are different types of event utility functions [17]. Existing QoM analysis typically considers simple cases of the function only (*e.g.*, the Step utility function) [18] [20]. Our analysis in this paper is more general, and covers other types of functions as well. Without loss of generality, we assume $U_j^i(x) = U(x)$ for all the events at all the PoIs; *i.e.*, the utility function is identical for all the events. Furthermore, we assume that the events are *identifiable* [17], *i.e.*, if more than one sensors detect the same event simultaneously, they will know it is the same event and learn exactly the same information.

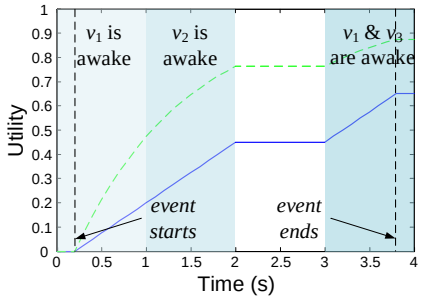


Fig. 2: An instance of utility computation with multiple sensors

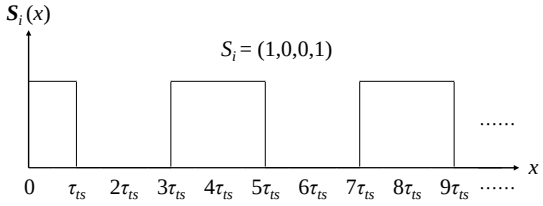


Fig. 3: An example of periodic extension function

To help understand the utility computation for a single event monitored by multiple sensors, we use a simple example for illustration. As can be seen in Fig. 2, PoI o_4 is covered by sensor v_1, v_2 and v_3, whose schedules are $S_1 = (1,0,0,1)$, $S_2 = (0,1,0,0)$ and $S_3 = (0,0,0,1)$, respectively. Let the time duration of a single slot be $\tau_{ts} = 1\,s$. Suppose an event arrives at PoI o_4 at time $t = 0.2\,s$ and leaves at $t = 3.8\,s$, and its utility function is defined as $U(x) = 0.25x$. Then, the aggregate captured utility of this event by the three sensors increases monotonically with time as the blue solid line shows in Fig. 2. Specifically, the utility of the event increases linearly during time period $[0.2, 1]$ as the event is monitored by sensor v_1, and keeps increasing during $[1, 2]$ when v_2 is active in that time duration. After that, the utility remains unchanged in $[2, 3]$ since no sensors are activated. During time period $[3, 3.8]$, the utility resumes its increase at the same speed as before, but not double the speed, although both v_1 and v_3 are active. This is due to the identifiability of events. We also plot the curve when the utility function is given by $U(x) = 1 - e^{-0.8x}$, as illustrated by the green dashed line in Fig. 2. It can be observed that the curve shows a similar trend.

3.3.2 QoM Computation

To start, we express the active status of sensors over time as a function of their individual schedules, by the *periodic extension function* given below. Note that each sensor is assumed to start its schedule at time 0.

Definition 3.1. ***(Periodic Extension Function)*** *Given a schedule S_i of sensor v_i, the periodic extension function $\boldsymbol{S}_i(x)$ ($\boldsymbol{S}_i : [0, +\infty] \mapsto \{0, 1\}$) of S_i is defined as:*

$$\boldsymbol{S}_i(x) = \begin{cases} 1, & (x \in [(k\mathcal{L} + j - 1)\tau_{ts}, (k\mathcal{L} + j)\tau_{ts}], k \in \mathcal{N}, S_i(j) = 1) \\ 0, & otherwise \end{cases}. \tag{4}$$

Note that τ_{ts} denotes the time duration of a single time slot.

We use a simple example for illustration. As shown in Fig. 3, the solid line denotes the value of the periodic extension function

of the schedule $S_i = (1, 0, 0, 1)$. The function takes value 1 when $x \in [4k\tau_{ts}, (4k+1)\tau_{ts}]$ or $x \in [(4k+3)\tau_{ts}, (4k+4)\tau_{ts}]$ for $k \in \mathcal{N}$, and 0 otherwise.

Then, we can derive the mathematical expression of the QoM of a PoI covered by a single sensor in the following lemma.

Lemma 3.1. *The QoM of a PoI, say o_i, covered by a single sensor v_j ($v_j \in V_i, |V_i| = 1$) under schedule S_j, whose periodic extension function is $\mathbf{S}_j(x)$, is given by:*

$$Q(i|S_j) = \frac{1}{\mathcal{L}\tau_{ts}} \int_0^{\mathcal{L}\tau_{ts}} \int_t^{+\infty} U(\int_t^y \mathbf{S}_j(x)dx) f(y-t)dydt. \quad (5)$$

Proof: Since the schedule of each sensor is fixed and periodic, we only need to consider the utility of events starting at time t ($t \in [0, \mathcal{L}\tau_{ts}]$). Specifically, for an event that starts at time t ($t \in [0, \mathcal{L}\tau_{ts}]$) and ends at time y ($y \in [t, +\infty)$), its utility is $U(\int_t^y \mathbf{S}_j(x)dx)$. As the event staying time follows the probability density function $f(x)$, the expected achieved utility for an event is $\int_t^{+\infty} U(\int_t^y \mathbf{S}_j(x)dx) f(y-t)dy$. By considering all possible events that occur in $[0, \mathcal{L}\tau_{ts}]$, we get Eq. (5). □

Suppose $S_i = (a_{i1}, \ldots, a_{i\mathcal{L}})$ and $S_j = (a_{j1}, \ldots, a_{j\mathcal{L}})$ are two different vectors. We define the "OR" operation of the vectors as $S_i \vee S_j = (a_{i1} \vee a_{j1}, \ldots, a_{i\mathcal{L}} \vee a_{j\mathcal{L}})$. The following lemma shows the QoM expression for a PoI covered by multiple sensors.

Lemma 3.2. *The QoM of PoI o_i covered by a set of sensors $V_i = \{v_{1'}, v_{2'}, \ldots, v_{m'}\}$, each of which having schedule S_j ($j = 1', 2', \ldots, m'$), is given by:*

$$Q(i) = Q(i|S_{1'}, S_{2'}, \ldots, S_{m'}) = Q(i|\bigvee_{v_j \in V_i} S_j). \quad (6)$$

In other words, the QoM achieved by the multiple sensors can be equivalently viewed as that by one single sensor with schedule $\bigvee_{v_j \in V_i} S_j$.

Proof: Referring back to the definition of QoM, we only have to show that the overall utility available for any particular event e_j^i gained by the collection of sensors V_i, namely $U(t_j^i)$, is exactly equal to the utility $U_a(t_j^i)$ gained by a virtual sensor v_a with schedule $\bigvee_{v_j \in V_i} S_j$. This can be derived by the identifiable and identical assumptions about the events. We omit the details to save space. □

For simplicity of exposition, we call $\widehat{S}_i = \bigvee_{v_j \in V_i} S_j$ the *equivalent monitoring schedule* for PoI o_i. We stress that our analysis can compute the QoM of a PoI in the presence of both single and multiple monitoring sensors. It can also accommodate general activation schedules, event utility functions, and probability distributions of the event staying times $f(x)$.

3.4 Problem Formulation

To sum up, we formally formulate our problem CHASE as **P1** shown below.

$$\textbf{(P1)} \quad \max_{S_i} \quad \sum_{i=1}^{n} w_i Q(i)$$
$$\text{s.t.} \quad (1), (2), (3).$$

Note that w_i is a normalized weight associated with the PoI o_i, which can be interpreted as the frequency of event occurrences of o_i or the importance of o_i. The decision variables are the activation schedules S_is of all the sensors. Note that the charging time for each sensor τ_i can be determined by S_i using Eq. (2), and the subset of sensors V_s selected for charging exactly contains the sensors v_is with non-zero activation schedules S_is. The quantities τ_w, c_i, p_i, τ, η_i, P_c, $\mathcal{L}$, E_i, l_i, and w_i are given constants.

Note that different heuristic algorithms, such as evolutionary techniques [62], [63], simulated annealing [64], and particle swarm optimization [65], can be used to solve CHASE, and they may show good performance in particular practical scenarios. Nevertheless, they do not guarantee good performance theoretically. In contrast, our proposed techniques provide provable approximation ratios, which improve upon the heuristics by bounding the loss of performance.

3.5 Roadmap of Our Solution

As evidenced by the above formulation, the full CHASE problem is complex. It involves the selection of the candidate set of sensors V_s for charging, the coupling between the MC's travel time and the allocation of charging time among the sensors, the active time slot constraint, and careful computation of the QoM. Among these factors, it is particularly hard to account for the travel time accurately. Hence, we start by considering a relaxed version of CHASE, which we call CHASE-R, that ignores the travel time; *i.e.*, we assume $\tau_{TSP}(V) = 0$. Besides amenable to analysis, importantly CHASE-R is also meaningful in practice, as τ_w can be much bigger than $\tau_{TSP}(V)$ due to long required charging time necessitated by typically limited charging efficiencies of MCs. For example, the charging time for the voltage to reach $1.8\ V$ for a WISP tag equipped with a $100\ uF$ capacitor can be as large as 155 seconds, when the RFID reader is 10.0 meters away [66]. After solving CHASE-R, we will accordingly develop solutions for the general CHASE problem by putting the MC's travel time back into consideration.

4 Theoretical Analysis

In this section, we show that the CHASE-R and CHASE problems stated above are NP-hard. Then, we reformulate the problems and present approximation algorithms for each of them, respectively.

4.1 Hardness of Problems

We now show that both CHASE-R and CHASE are NP-hard, and that they cannot be approximated within a factor better than $(1 - 1/e)$. To do that, we state the following well-known NP-hard problem and a related lemma.

Definition 4.1. ***(Maximum Coverage Problem)*** *[67] Given a collection of subsets $S = \{S_1, S_2, \ldots, S_m\}$ of the universal set $U = \{e_1, e_2, \ldots, e_n\}$ and a positive integer k, find a subset $S' \subseteq S$ such that $|S'| \leq k$ and the number of covered elements $|\cup_{S_i \in S'} S_i|$ is maximized.*

Lemma 4.1. *[68] For any $\epsilon > 0$, the Maximum Coverage Problem (MCP) cannot be approximated within a factor $(1 - 1/e + \epsilon)$ unless $P = NP$.*

We have the following theorem about the complexity of our problems.

Theorem 4.1. *Both CHASE-R and CHASE are NP-hard. For any $\epsilon > 0$, there are no $(1 - 1/e + \epsilon)$ approximation solutions to them unless $P = NP$.*

Proof: We can reduce the CHASE-R problem to the MCP problem by setting $\mathcal{L} = 1$, $w_i = 1/n$, and $c_i = c$, where c is a constant, and setting $l_i \geq \mathcal{L}$ for any sensor v_i such that the active time slot constraint can be removed. Hence, CHASE-R is at least as hard as MCP. For CHASE, we set $\tau_w/c = k + 1/2$ (where

k is an integer) and $\tau_{TSP}(V) < 1/2\ c$. We can then prove that CHASE is also at least as hard as MCP. By Lemma 4.1, the result follows. □

Remark: CHASE involves finding a shortest path to visit all the sensors in V_s. This component TSP problem is already NP-hard. The method in Theorem 4.1 is useful in that it applies to the easier CHASE-R problem that omits the travel time as well.

4.2 Reformulation of CHASE-R

Because CHASE-R is NP-hard, we seek approximation algorithms to solve it efficiently. In the following, we reformulate CHASE-R as a monotone submodular function maximization problem subject to constraints including a partition matroid constraint. Before detailing the reformulation, we present some necessary definitions.

Definition 4.2. *[69] Let S be a finite ground set. A real-valued set function $f : 2^S \mapsto \mathbb{R}$ is* normalized*,* monotonic *and* submodular *if it satisfies the following three conditions: (i) $f(\emptyset) = 0$; (ii) $f(A \cup \{e\}) - f(A) \geq 0$ for any $A \subseteq S$ and $e \in S \backslash A$; and (iii) $f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$ for any $A \subseteq B \subseteq S$ and $e \in S \backslash B$.*

For simplicity, we use $f_A(e) = f(A + e) - f(A)$ to denote the marginal value of element e with respect to A. Note that here, we use $A + e$ instead of $A \cup \{e\}$.

Definition 4.3. *[69] Given that $S = \bigcup_{i=1}^{k} S'_i$ is the disjoint union of k sets and $l_1, \ldots, l_k$ are positive integers, a* partition matroid *$\mathcal{M} = (S, \mathcal{I})$ is a matroid in which $\mathcal{I} = \{X \in S : |X \cap S'_i| \leq l_i$ for $i = 1, 2, \ldots, k\}$.*

Denote by $\mathbf{a}_{ij}$ the activating time slot a_{ij} of sensor v_i. We define the ground set S as:

$$S = \{\mathbf{a}_{11}, \mathbf{a}_{12}, \ldots, \mathbf{a}_{1\mathcal{L}}, \ldots, \mathbf{a}_{m1}, \mathbf{a}_{m2}, \ldots, \mathbf{a}_{m\mathcal{L}}\}. \tag{7}$$

We equivalently define the sensor schedule S_i as a subset of S, namely $S_i = \{\mathbf{a}_{i1'}, \mathbf{a}_{i2'}, \ldots, \mathbf{a}_{i\mathcal{L}'}\}$ if and only if $a_{ij'} = 1\,(j' = 1', 2', \ldots, \mathcal{L}')$. Furthermore, S can be partitioned into m disjoint sets, $S'_1, S'_2, \ldots, S'_m$, where $S'_i = \{\mathbf{a}_{i1}, \mathbf{a}_{i2}, \ldots, \mathbf{a}_{i\mathcal{L}}\}$. We say that S'_i is the *candidate activation schedule* of sensor v_i, since any feasible schedule S_i is a subset of S'_i. It is clear that any scheduling policy X consisting of all the sensor schedules, namely $X = \{S_1, S_2, \ldots, S_m\}$, is subject to $|X \cap S'_i| = |S_i| \leq l_i$. Thus, we write the independent sets as:

$$\mathcal{I} = \{X \subseteq S : |X \cap S'_i| \leq l_i \text{ for } i = 1, 2, \ldots, m\}. \tag{8}$$

Note that it is easy to prove that $\mathcal{M} = \{S, \mathcal{I}\}$ is a matroid.

Moreover, define $\mathbf{c}_{ij} = c_i$ as the charging time factor for time slot a_{ij}. The working time constraint can be rewritten as $\sum_{\mathbf{a}_{ij} \in X} \mathbf{c}_{ij} \leq \tau_w$, which is exactly a *knapsack constraint*. Hence, we have the following lemma.

Lemma 4.2. *The working time constraint in CHASE-R can be written as a knapsack constraint on the ground set S, while the active time slot constraint can be written as a partition matroid constraint.*

Consequently, we can rewrite the optimization problem CHASE-R as **RP1** shown below.

$$\begin{aligned} \textbf{(RP1)} \quad \max_{X} \quad & f(X) = \sum_{i=1}^{n} w_i Q(i| \bigvee_{v_j \in V_i} S_j) \\ \textit{s.t.} \quad & X \in \mathcal{I}, \\ & S_i = X \cap S'_i \quad \forall i = 1, 2, \ldots, m, \\ & \sum_{\mathbf{a}_{ij} \in X} \mathbf{c}_{ij} \leq \tau_w. \end{aligned}$$

Note that the decision variable in **RP1** is the scheduling policy X, which consists of elements $\mathbf{a}_{ij}$s denoting the activation time slot a_{ij} of sensor v_i. By comparing **RP1** with **P1**, we can see that the decision variables change from the activation schedules S_i in **P1** to the scheduling policy X in **RP1**, and the two are essentially equivalent.

Next, we show that the optimization function $f(X)$ exhibits a desirable property as stated in the following lemma.

Lemma 4.3. *If the utility function $U(x)$ is* concave*, then the objective function $f(X)$ in **RP1** is a monotone submodular function.*

Proof: To prove the monotonicity and submodularity of the objective function $f(X)$, we have to verify if the three conditions in Def. 4.2 hold for $f(X)$.

First, it is easy to see that $f(\emptyset) = 0$, which means that the first condition holds for $f(X)$.

Second, we check whether the monotonicity property holds for $f(X)$. Suppose we have a set $A \subseteq S$ and an element $e_1 \in S \backslash A$ and $e_1 = \mathbf{a}_{ij}$. We can then regard $f(A + e_1)$ as the resulting overall QoM obtained by activating the time slot a_{ij} of sensor v_i based on the original scheduling policy as far as A is concerned. As a result, the equivalent monitoring schedule of PoI o_k, which is covered by v_i ($o_k \in O_i$), may be changed accordingly. Specifically, suppose the original and changed equivalent monitoring schedules of o_k are $\widehat{S}_k^{<A>}$ and $\widehat{S}_k^{<A+e_1>}$, respectively. Suppose the time slot a_{kj} is activated for $\widehat{S}_k^{<A+e_1>}$. In addition, we use the expression $\mathbf{a}_{ij} \in \widehat{S}_k^{<A>}$ to indicate that the j_{th} time slot of $\widehat{S}_k^{<A>}$ is active, namely $a_{kj} = 1$ and $\mathbf{a}_{ij} \notin \widehat{S}_k^{<A>}$ the opposite. In reality, both $\mathbf{a}_{ij} \in \widehat{S}_k^{<A>}$ and $\mathbf{a}_{ij} \notin \widehat{S}_k^{<A>}$ are possible. However, due to limited space, we consider only the case $\mathbf{a}_{ij} \notin \widehat{S}_k^{<A>}$, since it is more complicated.

We have the following important observation:

$$\widehat{\mathbf{S}}_k^{<A+e_1>}(x) - \widehat{\mathbf{S}}_k^{<A>}(x) = \begin{cases} 1, & x \in [(k\mathcal{L} + j - 1)\tau_{ts}, (k\mathcal{L} + j)\tau_{ts}] \\ & (k \in \mathcal{N}) \\ 0, & otherwise \end{cases} \tag{9}$$

which immediately leads to:

$$\int_t^y \widehat{\mathbf{S}}_k^{<A+e_1>}(x)dx - \int_t^y \widehat{\mathbf{S}}_k^{<A>}(x)dx \geq 0 \tag{10}$$

for any $y \geq t \geq 0$.

Next, we note that any utility function $U(x)$ must increase monotonically from zero to one as a function of the total observation time, *i.e.*, $U(x) \geq 0$ and $U(y) - U(x) \geq 0$ for any $y \geq x \geq 0$. Hence, combining Eq. (5) and Eq. (10), we have:

$$\begin{aligned} & Q(k|\widehat{S}_k^{<A+e_1>}) - Q(k|\widehat{S}_k^{<A>}) \\ =& \frac{1}{\mathcal{L}\tau_{ts}} \int_0^{\mathcal{L}\tau_{ts}} \int_t^{\infty} [U(\int_t^y \widehat{\mathbf{S}}_k^{<A+e_1>}(x)dx) - U(\int_t^y \widehat{\mathbf{S}}_k^{<A>}(x)dx)] \\ & \cdot f(y - t)dydt \\ \geq& 0. \end{aligned}$$

Therefore,

$$f_A(e_1) = \sum_{o_k \in O_i} w_k[Q(k|\widehat{S}_k^{<A+e_1>}) - Q(k|\widehat{S}_k^{<A>})] \geq 0,$$

which means that the monotonicity property holds for $f(X)$.

Third, we check the last condition for $f(X)$. Similar to the monotonicity property analysis, suppose we have set $A \subseteq B \subseteq S$ and element $e_1 \in S \backslash B$ ($e_1 = \mathbf{a}_{ij}$). The original equivalent monitoring schedules for o_k in A and B are $\widehat{S}_k^{<A>}$ and $\widehat{S}_k^{<B>}$, and they respectively change to $\widehat{S}_k^{<A+e_1>}$ and $\widehat{S}_k^{<B+e_1>}$ after adding the element e_1. To save space, we consider only the case of $\mathbf{a}_{ij} \notin \widehat{S}_k^{<A>}$ and $\mathbf{a}_{ij} \notin \widehat{S}_k^{<B>}$ in this paper.

Before the detailed proof, we show an important property for the utility function $U(x)$. For any $y \geq x \geq 0$ and $\delta \geq 0$, we have:

$$U(x+\delta) \geq \frac{y-x}{y+\delta-x}U(x) + (1 - \frac{y-x}{y+\delta-x})U(y+\delta), \quad (11)$$

and

$$U(y) \geq \frac{\delta}{y+\delta-x}U(x) + (1 - \frac{\delta}{y+\delta-x})U(y+\delta). \quad (12)$$

Note that $x \leq x+\delta \leq y+\delta$ and $x \leq y \leq y+\delta$, and $U(x)$ is concave. Adding up the left sides and the right sides of Eqs. (11) and (12), and simplifying the inequality, we have:

$$U(x+\delta) - U(x) \geq U(y+\delta) - U(y) \quad (13)$$

for $y \geq x \geq 0$ and $\delta \geq 0$.

Since $A \subseteq B$, we have:

$$\widehat{\mathbf{S}}_k^{<B>}(x) - \widehat{\mathbf{S}}_k^{<A>}(x) \geq 0 \quad (14)$$

for $x \in [0, +\infty]$.

Moreover, it is easy to see that:

$$\widehat{\mathbf{S}}_k^{<A+e_1>}(x) - \widehat{\mathbf{S}}_k^{<A>}(x) = \widehat{\mathbf{S}}_k^{<B+e_1>}(x) - \widehat{\mathbf{S}}_k^{<B>}(x) \quad (15)$$

as $\mathbf{a}_{ij} \notin \widehat{S}_k^{<A>}$ and $\mathbf{a}_{ij} \notin \widehat{S}_k^{<B>}$ for $x \in [0, +\infty]$.

Therefore, from Eqs. (13), (14), and (15), it is obvious that:

$$\begin{aligned}
&[Q(k|\widehat{S}_k^{<A+e_1>}) - Q(k|\widehat{S}_k^{<A>})] - [Q(k|\widehat{S}_k^{<B+e_1>}) - Q(k|\widehat{S}_k^{<B>})] \\
=&\frac{1}{\mathcal{L}\tau_{ts}} \int_0^{\mathcal{L}\tau_{ts}} \int_t^{\infty} \{[U(\int_t^y \widehat{\mathbf{S}}_k^{<A+e_1>}(x)dx) - U(\int_t^y \widehat{\mathbf{S}}_k^{<A>}(x)dx)] \\
&- [U(\int_t^y \widehat{\mathbf{S}}_k^{<B+e_1>}(x)dx) - U(\int_t^y \widehat{\mathbf{S}}_k^{<B>}(x)dx)]\} f(y-t)dydt \\
\geq& 0.
\end{aligned}$$

and:

$$\begin{aligned}
&f_A(e_1) - f_B(e_1) \\
=& \sum_{o_k \in O_i} w_k\{[Q(k|\widehat{S}_k^{<A+e_1>}) - Q(k|\widehat{S}_k^{<A>})] - [Q(k|\widehat{S}_k^{<B+e_1>}) \\
&- Q(k|\widehat{S}_k^{<B>})]\} \\
\geq& 0.
\end{aligned}$$

We thus conclude that the third condition holds for $f(X)$ as well, and the result follows. □

In general, utility functions can be concave (*e.g.*, the Step, Exponential, and Linear functions in [17]) or not (*e.g.*, the S-shaped and Delayed Step functions [17]). For our purposes, *we consider only utility functions $U(x)$ that are concave hereafter*. The utility functions of many real-world applications appear to be concave [17] [18] [19] [20] [21]. Hence, our contributions are not diminished significantly.

Algorithm 1 Basic algorithm for CHASE-R with active time slot constraint

Input: The objective function $f(\cdot)$, the ground set S, the partition matroid $\mathcal{M}$, the knapsack constraint, the candidate activation schedules $S'_1, \ldots, S'_m$.
Output: Solution X and the sensor schedules $S_1, \ldots, S_m$.
1: Reduce knapsack constraint by applying Lemma 4.4 with $0 < \epsilon < 0.5$; let $\{\mathcal{P}_t\}_{t=1}^T$ denote the resulting partition matroids;
2: **for** each $t \in [T]$ **do**
3: Run the greedy algorithm from [70] under 2 partition matroids $\mathcal{M}$ and $\mathcal{P}_t$ to obtain solution X_t;
4: **end for**
5: $t^* \leftarrow \arg\max_{t=1}^T f(X_t)$;
6: Starting with the trivial partition of X_{t^*} into single elements, greedily merge parts as long as each part satisfies the knapsack constraint, until no further merge is possible. Consequently, X_{t^*} can be partitioned into k parts $\{X_{t^*}^j\}_{j=1}^k$;
7: $X \leftarrow \arg\max_{j=1}^k f(X_{t^*}^j)$, $S_i \leftarrow X \cap S'_i$ for $i = 1, \ldots, m$;

4.3 Approximation Algorithms for CHASE-R

Having proved that the objective function of our problem is submodular, we now aim to find approximation algorithms with and without the active time slot constraint for CHASE-R. We will show that the presence of the active time slot constraint makes the problem significantly more complex.

4.3.1 Approximation Algorithm with Active Time Slot Constraint

For this case, we tailor the approach proposed by Gupta *et al.* [71] to our settings, and obtain an improved approximation. Their work targets p-system and q-knapsack in max-min optimization, where a p-system is similar to, but more general than, the intersection of p matroids. At a high level, their approach extends ideas from Chekuri and Khanna [72] that reduce knapsack constraints to partition matroids by an enumeration method. We list the main result of this reduction as follows.

Lemma 4.4. *Given any knapsack constraint $\sum_{i=1}^n w_i \cdot x_i \leq B$ and fixed $0 < \epsilon < 1$, there is a polynomial-time computable collection $\mathcal{P}_1, \ldots, \mathcal{P}_T$ of $T = n^{O(1/\epsilon^2)}$ partition matroids such that:*
1. For every $X \in \bigcup_{t=1}^T \mathcal{P}_t$, we have $\sum_{i \in X} w_i \leq (1+\epsilon) \cdot B$.
2. $\{X \subseteq [n] | \sum_{i \in X} w_i \leq B\} \subseteq \bigcup_{t=1}^T \mathcal{P}_t$.

Note that we use notations similar to [71] for consistency. We assume that Ω is the intersection of the partition matroid and knapsack constraints. By scaling weights in the knapsack constraint, we assume without loss of generality that the knapsack has capacity exactly one. Let $\mathcal{C}$ denote the weights in the knapsack constraint. Assume that the optimal QoM of CHASE-R is OPT and its corresponding solution is X^{OPT}.

We propose the algorithm specified in Algorithm 1, which is devised based on the algorithm proposed in [71].

Theorem 4.2. *Algorithm 1 for CHASE-R with active time slot constraint can achieve $1/6$-approximation, and its time complexity is $O((m\mathcal{L})^2 nT)$.*

Proof: We omit the details of the proof to save space. □

We improve the approximation factor from $\frac{1}{(p+2)(3q+1)} = 1/12$, obtained by [71] for p-system and q-knapsack constraints, to $1/6$. This is because we give a tighter bound for the number of partitioned parts k at Step 6 in Algorithm 1 than that in [71]. Besides, although the algorithm proposed in [73] for 1 matroid

Algorithm 2 Enhanced algorithm for CHASE-R with active time slot constraint

Input: The objective function $f(\cdot)$, the ground set S, the partition matroid $\mathcal{M} = (S, \mathcal{I})$, the knapsack constraint, the candidate activation schedules $S'_1, \ldots, S'_m$, and the error threshold ϵ.
Output: Solution X and the sensor schedules $S_1, \ldots, S_m$.

$u^* \leftarrow \max_{e \in S} f(e)$;
for $\rho \in \{\frac{u^*}{2}, (1+\epsilon)\frac{u^*}{2}, (1+\epsilon)^2\frac{u^*}{2}, \ldots, m\mathcal{L}u^*\}$ **do**
  $\zeta \leftarrow u^*_\rho \leftarrow \max\{f(e) : \frac{f(e)}{\mathbf{c}_e} \geq \rho\}$; ▷ $\mathbf{c}_e$ is the corresponding coefficient of element e;
  $S' = \emptyset$;
  while $\zeta \geq \frac{\epsilon}{m\mathcal{L}} u^*_\rho$ and $\sum_{e \in S'} \mathbf{c}_e \leq 1$ **do**
    for each $e \in S$ **do**
      if $S' + e \in \mathcal{I}$, $f(S' + e) - f(S') \geq \zeta$, and $\frac{f(S'+e)-f(S')}{\mathbf{c}_e} \geq \rho$ **then**
        $S' \leftarrow S' + e$;
        if $\sum_{e \in S'} \mathbf{c}_e > 1$ **then**
          $S'_\rho \leftarrow S'$, $X_\rho \leftarrow S' \backslash \{e\}$, $X'_\rho \leftarrow \{e\}$;
          continue with the next value of ρ;
        end if
      end if
    end for
    $\zeta \leftarrow \frac{1}{1+\epsilon}\zeta$;
  end while
  $X_\rho \leftarrow S'_\rho \leftarrow S'$, $X'_\rho \leftarrow \emptyset$;
end for
$X \leftarrow \arg\max_\rho X_\rho$, $S_i \leftarrow X \cap S'_i$ for $i = 1, \ldots, m$;

and k knapsack constraints can achieve an $(1 - 1/e - \varepsilon)$-approximation, it requires that all the sets of at most 10^{12} items to be enumerated to form a feasible solution at the first stage, which limits its practicality.

Moreover, we employ *pruning techniques* when implementing this algorithm to speed up the computation, since the number $T = (m\mathcal{L})^{O(1/\epsilon^2)}$ of produced partition matroids is still large. We omit the details to save space.

4.3.2 Enhanced Approximation Algorithm with Active Time Slot Constraint

In this section, we propose an enhanced algorithm to address CHASE-R with active time slot constraint. The enhanced algorithm is based on the algorithm proposed in [74] for maximizing a submodular function subject to a p-system and q-knapsack constraints. Compared with Algorithm 1, this algorithm achieves not only better performance guarantee, but it is also faster. Algorithm 2 specifies the enhanced algorithm.

By the classical results in [74], we have the following theorem.

Theorem 4.3. *Algorithm 2 for CHASE-R with active time slot constraint can achieve $1/(4+\epsilon)$-approximation where ϵ is an arbitrarily small positive value, and its time complexity is $O(n \cdot \frac{m\mathcal{L}}{\epsilon^2} \log^2 \frac{m\mathcal{L}}{\epsilon})$.*

Proof: We omit details of the proof to save space. □

4.3.3 Approximation Algorithm without Active Time lot Constraint

If $l_i \geq \mathcal{L}$ for any sensor v_i, then the active time slot constraint can be safely relaxed. This situation occurs when the battery capacity is large relative to the working power of the sensor (*e.g.*, ultra-capacitors [59]) or we apply the charging process frequently. In this case, we can resort to a unified greedy algorithm, namely Algorithm 3, to find an optimized QoM. Note that in this algorithm, $\mathbf{c}'_i$ refers to the corresponding charging time factor for time slot $\mathbf{a}'_i$.

Algorithm 3 Unified algorithm for CHASE-R without active time slot constraint

Input: The objective function $f(\cdot)$, the ground set S, the knapsack constraint, the candidate activation schedules $S'_1, \ldots, S'_m$.
Output: Solution X and the sensor schedules $S_1, \ldots, S_m$.

$X \leftarrow \emptyset$, $X_1 \leftarrow \emptyset$, $X_2 \leftarrow \emptyset$, $S_i \leftarrow \emptyset$ for $i = 1, \ldots, m$;
If $k = 0$, then $k' \leftarrow 1$; else $k' \leftarrow k - 1$;
$X_1 \leftarrow \arg\max f(D), \forall D \in S, |D| \leq k', \sum_{\mathbf{a}'_i \in D} \mathbf{c}'_i \leq \tau_w$;
for all $D \in S$ ($|D| = k$ and $\sum_{\mathbf{a}'_i \in D} \mathbf{c}'_i \leq \tau_w$) **do**
  $I \leftarrow S$;
  while $I \backslash D \neq \emptyset$ **do**
    $\mathbf{a}'_t \leftarrow \arg\max_{\mathbf{a}'_i \in I \backslash D} \frac{f_D(\mathbf{a}'_i)}{\mathbf{c}'_i}$;
    if $f_D(\mathbf{a}'_t) \leq 0$ **then**
      break;
    end if
    if $\sum_{\mathbf{a}'_i \in D} \mathbf{c}'_i + \mathbf{c}'_t \leq \tau_w$ **then**
      $D \leftarrow D \cup \{\mathbf{a}'_t\}$;
    else
      $I \leftarrow I \backslash \{\mathbf{a}'_t\}$;
    end if
  end while
  if $f(X_2) \leq f(D)$ **then**
    $X_2 \leftarrow D$;
  end if
end for
$X \leftarrow \arg\max\{f(X_1), f(X_2)\}$, $S_i \leftarrow X \cap S'_i$ for $i = 1, \ldots, m$;

This algorithm includes two parts. The first part enumerates all possible subsets of S with cardinality less than or equal to k', so as to find the best feasible solution for the highest QoM. The second part starts from every feasible subset D with cardinality k, and searches greedily in S to find a best possible solution. Finally, the algorithm outputs the best observable solution based on the results of the above two parts.

We have the following theorem based on the results obtained by [75].

Theorem 4.4. *Algorithm 3 for CHASE-R without active time slot constraint achieves approximation factors of $\frac{1-1/e}{2} \approx 0.3161$, $\frac{1-1/e}{2-1/e} \approx 0.3873$, $\frac{1-1/e}{3/2-1/e} \approx 0.5584$, $1 - 1/e \approx 0.6321$ for $k = 0, 1, 2, 3$, respectively. Its time complexity is $O((m\mathcal{L})^{k+2}n)$.*

Proof: We omit details of the proof to save space. □

By Theorem 4.1, we claim that Algorithm 3 for $k = 3$ is in fact the best possible for any polynomial-time approach unless $P = NP$.

4.4 Approximation Algorithms for CHASE

Based on the proposed constant approximation algorithms for CHASE-R, we now consider the original problem CHASE and propose approximation algorithms for it.

As shown in Algorithm 4, the solution calls Algorithm 1 at the first step to obtain a feasible solution X_R for CHASE-R. Subsequently, we sort the elements in X_R in descending order by their cost efficiency in event monitoring, defined as the ratio of the overall QoM enhancement yielded by a given active time slot to the charging time required for an MC to enable that time slot to be active. We iteratively remove an element with the least cost efficiency in X (X is initialized as X_R) until $\tau_{TSP}(\bigcup_{|X \cap S'_i| > 0} v_i) \leq \tau_w - \sum_{\mathbf{a}'_i \in X} \mathbf{c}'_i$. Note that we employ the *nearest neighbor algorithm* to solve the TSP problem. Finally, we obtain a feasible solution X for CHASE.

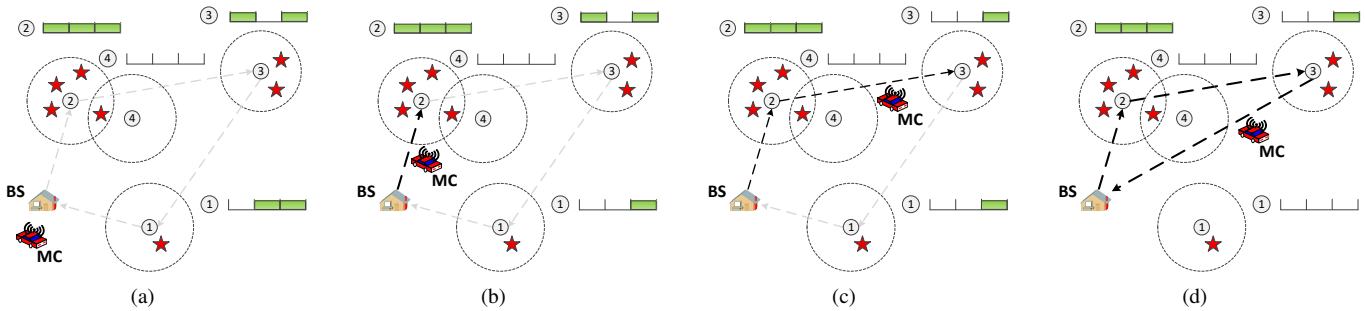


Fig. 4: An instance of Algorithm 4 for CHASE

Algorithm 4 Algorithm for CHASE

Input: The sensors set $V = \{v_1, \ldots, v_m\}$, the PoIs set $O = \{o_1, \ldots, o_n\}$, the objective function $f(\cdot)$, the ground set S, the partition matroid $\mathcal{M}$, the knapsack constraint, the candidate activation schedules $S'_1, \ldots, S'_m$.

Output: The sensor schedules $S_1, \ldots, S_m$.

1: Call Algorithm 1 to obtain the solution X_R for CHASE-R;
2: Sort $X_R \leftarrow \{\mathbf{a}'_1, \ldots, \mathbf{a}'_K\}$ such that $\mathbf{a}'_t = \arg\max_{\mathbf{a}'_i \in X_R \setminus X_R^{t-1}} \frac{f_{X_R^{t-1}}(\mathbf{a}'_i)}{\mathbf{c}'_i}$ $(X_R^{t-1} = \{\mathbf{a}'_1, \ldots, \mathbf{a}'_{t-1}\})$;
3: $X \leftarrow X_R$, $t \leftarrow K$;
4: **while** $\tau_{TSP}(\bigcup_{|X \cap S'_i|>0} v_i) > \tau_w - \sum_{\mathbf{a}'_i \in X} \mathbf{c}'_i$ **do**
5: $X \leftarrow X \backslash \mathbf{a}'_t$;
6: $t \leftarrow t - 1$;
7: **end while**
8: $S_i \leftarrow X \cap S'_i$ for $i = 1, \ldots, m$;

Fig. 4 illustrates an instance of Algorithm 4. Suppose after employing Algorithm 1, the schedules for Sensor 1, 2, 3, and 4 are $(0, 1, 1)$, $(1, 1, 1)$, $(1, 0, 1)$, and $(0, 0, 0)$, respectively, as demonstrated in Fig. 4(a). Meanwhile, the travel path passing by Sensors 1, 2, and 3 that have non-empty active time slots is shown by the grey dashed lines, since the available time left for the MC after charging Sensors 1, 2, and 3 cannot allow the MC to continue traveling along any edge of the path. After sorting the elements in X_R at Step 2, Algorithm 4 finds that the second active time slot of Sensor 1 has the minimum cost efficiency. It thus removes that active time slot, and the time saved then allows the MC to travel from the BS to Sensor 2, as illustrated by the dark dashed line in Fig. 4(b). Likewise, Fig. 4(c) shows the result after removing the first active time slot of Sensor 3 which has the second smallest cost efficiency, and using the saved time for the MC's further travel. Finally, in Fig. 4(d), the third time slot of Sensor 1 is de-activated; consequently, the MC is able to charge all the sensors and finish the charging path within the time constraint, which means that $\tau_{TSP}(\bigcup_{|X \cap S'_i|>0} v_i) \leq \tau_w - \sum_{\mathbf{a}'_i \in X} \mathbf{c}'_i$ is satisfied. Note that the MC no longer needs to pass by Sensor 1 because it has no active time slot. By doing so, we obtain a feasible solution for the original CHASE problem.

Theorem 4.5. *Algorithm 4 for CHASE based on Algorithm 1 achieves* $\frac{1}{6}(1 - \frac{\tau_{TSP}(V)+max\{c_i\}_{i=1}^m}{\tau_w})$*-approximation. The time complexity of this algorithm is* $O((m\mathcal{L})^2 nT)$.

Proof: We omit details of the proof to save space. □

Where there is no confusion, we call Algorithm 4, which works based on Algorithm 1, the CHASE algorithm. We can also base our algorithm (for the CHASE problem) on the enhanced algorithm for CHASE-R with active time slot constraint. This algorithm differs from Algorithm 4 in that it calls Algorithm 2 rather than Algorithm 1 at Step 1. We call this alternative algorithm E-CHASE. We have the following theorem.

Theorem 4.6. *Algorithm 4, which works based on Algorithm 2, achieves* $\frac{1}{4+\epsilon}(1 - \frac{\tau_{TSP}(V)+max\{c_i\}_{i=1}^m}{\tau_w})$*-approximation. Its time complexity is* $O(n \cdot \frac{m\mathcal{L}}{\epsilon^2} \log^2 \frac{m\mathcal{L}}{\epsilon} + m^3\mathcal{L})$.

Proof: We omit the details of the proof to save space. □

Lastly, if the active time slot constraint is not needed, we can modify Algorithm 4 by replacing Algorithm 1 (called at Step 1) with Algorithm 3. The following theorem gives a performance guarantee of this revised algorithm.

Theorem 4.7. *The revised algorithm for CHASE without active time slot constraint achieves approximation factors of* $\frac{1-1/e}{2}c$, $\frac{1-1/e}{2-1/e}c$, $\frac{1-1/e}{3/2-1/e}c$, $(1 - 1/e)c$ *for* $k = 0, 1, 2, 3$, *respectively, where* $c = 1 - \frac{\tau_{TSP}(V)+max\{c_i\}_{i=1}^m}{\tau_w}$. *Its time complexity is* $O((m\mathcal{L})^{k+2}n + m^3\mathcal{L})$ *for* $k = 0, 1, 2, 3$.

Proof: We omit details of the proof to save space. □

5 Performance Evaluation

In this section, we present simulation results that verify our analysis and illustrate the performance of our algorithms.

5.1 Evaluation Setup

Unless otherwise stated, we use the following parameter settings. We set the range of received power of a sensor to $[15\,mW, 45\,mW]$, which can be interpreted as a charging efficiency η_i between $[0.5\,\%, 1.5\,\%]$. In our experiments, we randomly distribute 20 sensors and 50 PoIs in a $120\,m \times 120\,m$ region, where any PoI is covered by at least one sensor. The working power p_i of sensor v_i is randomly selected from $[50\,\mu W, 100\,\mu W]$, while that of the MC is set to $3\,W$. The battery capacity of a sensor is randomly selected from the range $[100\,J, 1000\,J]$. Furthermore, we set the sensing radius of a sensor to $20\,m$ and the sensor schedule length $\mathcal{L} = 4$. We assume that the considered event type has a Step utility function and its event staying times follow $f(x) = \lambda e^{-\lambda x}$ where $\lambda = 1$. We set $\tau = 2\,week$, $\tau_w = 8.2\,hour$, and the MC's speed $\nu_{MC} = 0.05\,m/s$. Lastly, the default duration of a time slot is set to be $1\,s$.

5.2 Baseline Setup

Because there are no existing algorithms for joint mobile charging and scheduling of sensors for stochastic event capture in a

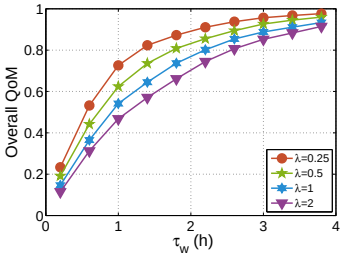


Fig. 5: QoM vs. τ_w for Step utility function

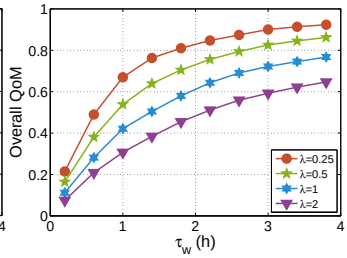


Fig. 6: QoM vs. τ_w for Exponential utility function

wireless rechargeable sensor network, we develop two algorithms for comparison, *i.e.*, JPW algorithm and RANDOM algorithm. The first algorithm is obtained by specializing the Joint Periodic Wake-up (JPW) algorithm in [24]. Specifically, an MC in JPW distributes its charging time evenly to each sensor; only when it arrives at the position of a sensor can it charge the sensor (this assumption is realistic in that the effective charging distance of chargers is typically far less than the distances between sensors). The energy cost for turning the sensor on/off is ignored, as before. In addition, we set $\eta_i = 1\,\%$ for any sensor v_i, and the time duration of a duty cycle in JPW is exactly equal to that of the sensor schedule. ν_{MC} is set to $0.1\,m/s$. For the other parameters, we use the same settings as those in Section 5.3.3. The second algorithm RANDOM differs from JPW in that it randomly chooses for charging the same number of sensors as that for CHASE, and then uniformly distributes the charging time among the selected sensors. Note that every point on the plots for RANDOM represents an average result over 100 randomly generated instances.

5.3 Performance Evaluation for CHASE-R Algorithms

In this section, we first investigate the cases without considering the active time slot constraint. In particular, we evaluate the overall QoM under different event types or different values of the control parameter k. Then, we study the relationship between the period of the charging process τ and the overall QoM. This relationship shows the impact of the active time slot constraint.

5.3.1 Impact of Event Types

In this set of experiments, we focus on the event types whose event staying times follow $f(x) = \lambda e^{-\lambda x}$ ($\lambda = 0.25, 0.5, 1, 2$) [17], [24], whereas the utility function $U(x)$ is either the Step utility function or the Exponential utility function $f(x) = Ae^{-Ax}$, where $A = 5$ [17]. Note that we use Algorithm 3 with $k = 3$. It can be seen in Figs. 5 and 6 that the overall QoM always increases as τ_w increases. However, the marginal gain of the QoM decreases as τ_w increases. The reason is that the event capture utility function is concave and redundant coverage of the PoIs becomes more likely when the sensors have larger active time slot budgets under a larger τ_w. Moreover, a smaller λ will lead to a larger overall QoM under the same τ_w. This is because the expected staying time of events grows as λ decreases; therefore, its probability of being detected, as well as the utility of sensing, increases. Besides, by comparing Fig. 5 and Fig. 6, we see that the achieved overall QoM for events with Step utility always exceeds that for events with Exponential utility. This can be explained by differences in the efficiency of event capture. For events under Step utility, full information about an event is obtained instantaneously on detection. In contrast, it can require a lot more time to obtain most information about an event under Exponential utility.

5.3.2 Impact of Control Parameter k

We proceed to evaluate the impact of the control parameter k on the overall QoM in Algorithm 3, and plot the results in Fig. 7. Not surprisingly, it can be seen that the larger k we choose, the higher overall QoM we obtain. However, the differences between the overall QoM under different k are not obvious. This observation suggests that we can choose a small k to reduce the time complexity without incurring much performance degradation. In addition, it can be observed that the overall QoM exceeds $1 - 1/e \approx 0.63$ in Fig. 7, which is consistent with Theorem 4.4 as the optimal overall QoM cannot exceed 1.

5.3.3 Impact of Period of Charging Process

To see how the period of the charging process τ impacts the overall QoM, we set $E_i = 100\,J$ and $p_i = 100\,\mu W$, and let the received power of each sensor randomly fluctuate within a relatively smaller range of $[20\,mW, 35\,mW]$ to ease computation. Fig. 8 shows the trend that the overall QoM decreases with an increasing τ. This is because an increasing τ leads to higher charging time factors c_i and smaller active time slot budgets l_i, both of which finally lead to a reduced QoM. Moreover, that the overall QoM is larger than $1/6 \approx 0.17$ is consistent with Theorem 4.2. Again, we can see that the achieved overall QoM increases with τ_w.

5.4 Performance Evaluation for the CHASE Algorithms

We proceed to verify the performance of the algorithms for CHASE, which consider the MC's travel time. The experiments use the same parameters as in Section 5.3.3.

5.4.1 A Solution to an CHASE Instance

Fig. 9 illustrates the selected sensors for charging, the sensor schedules, the MC's travel path, and the achieved QoM for the PoIs, for an E-CHASE solution to a CHASE problem instance. Note that the BS is located at $(0, 0)$; the sensors are marked as circles and the PoIs as triangles. The filled color of a triangle indicates the achieved QoM for the corresponding PoI, which varies from 0 to 1 as the color bar shows. A blue circle indicates that the corresponding sensor is not chosen at Step 1 in Algorithm 4, while a green circle indicates that the corresponding sensor is greedily removed in the while loop.

5.4.2 Impact of MC's Speed on Working Time Allocation and QoM

Fig. 10 shows that if the speed of the MC ν_{MC} increases, the travel time is reduced, leading to a larger aggregate time for charging. Note that both the travel time and aggregate charging time are normalized with respect to the maximum working time τ_w. It can be seen that the fraction of the aggregate travel time keeps below $10\,\%$ when ν_{MC} grows to $0.1\,m/s$, which is still small. Another interesting finding from Fig. 10 is that the sum of the travel time and aggregate charging time is not necessarily equal to τ_w (the gap is up to $4\,\%$ when $\nu_{MC} = 0.02$). This situation happens because we require the active time slot budget l_i for each sensor to be an integer. The requirement can be relaxed, and we can assign the residual working time to charging sensors. We expect the overall QoM to increase as a result, but the details are left

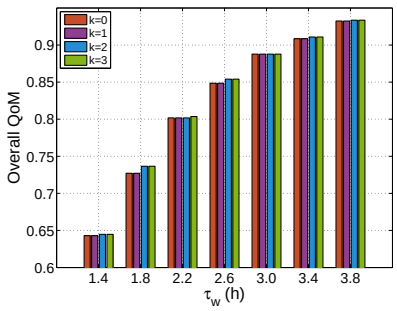


Fig. 7: Impact of control parameter k

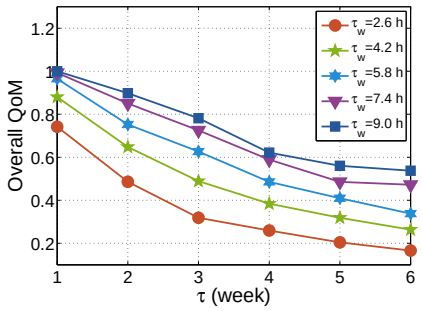


Fig. 8: QoM vs. τ

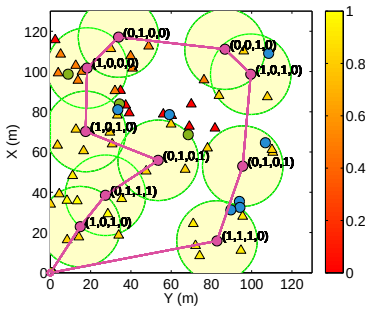


Fig. 9: CCF A solution to a CHASE problem instance

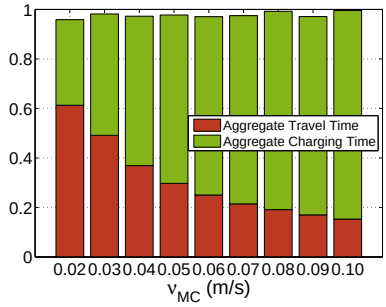


Fig. 10: Impact of MC's speed on working time allocation

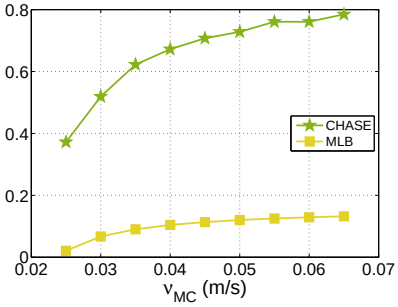


Fig. 11: Impact of MC's speed on overall QoM

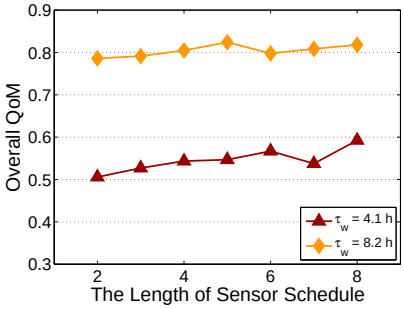


Fig. 12: QoM vs. length of sensor schedule

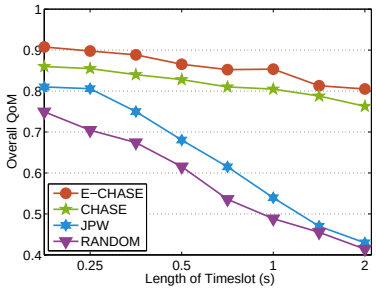
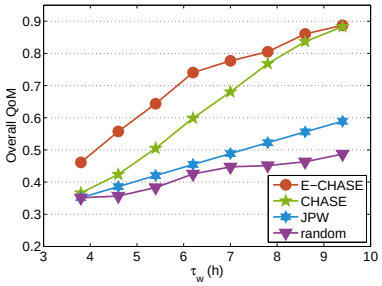
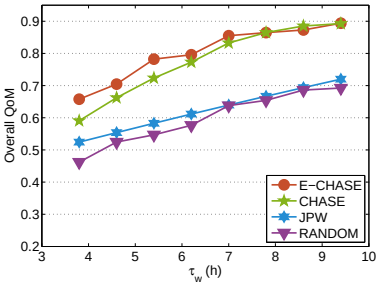


Fig. 13: QoM vs. duration of time slots



(a) Duration of time slot is $1\,s$



(b) Duration of time slot is $0.5\,s$

Fig. 14: QoM vs. τ_w

for future work. As expected, the overall QoM is enhanced with a faster MC, as the red solid line shows in Fig. 11. Moreover, it is always bigger than the "maximum" lower bound, which is indicated as the green dotted line "MLB" in the figure and given by $\frac{1}{6}(1 - \frac{\tau_{TSP}(V)+max\{c_i\}_{i=1}^{m}}{\tau_w})$. This finding corroborates Theorem 4.5 in Section 4.4.

5.4.3 Impact of Length of Sensor Schedule

Fig. 12 shows the trend of the overall QoM when the length of the sensor schedule $\mathcal{L}$ increases under $\tau_w = 4.1\,hours$ and $\tau_w = 8.2\,hours$, respectively. We can see that, although the overall QoM does not increase monotonically with $\mathcal{L}$ exactly, there is a general tendency for it to rise with $\mathcal{L}$.

5.5 Performance Comparison with Existing Work

5.5.1 Impact of Length of Time Slots

The default duration of a time slot is $1\,s$ in the experiments so far. In this subsection, we vary the time slot duration, and plot the overall QoM for both JPW and CHASE in Fig. 13. Note that the proposed E-CHASE and CHASE algorithms consistently outperform JPW and RANDOM, especially when the time slot duration is long. The performance gains of E-CHASE and CHASE over JPW and RANDOM are $40.8\,\%$ and $53.5\,\%$, and $34.0\,\%$ and $46.1\,\%$, respectively. Moreover, the overall QoM increases as the time slot duration decreases for all the four schemes. The reason is that information about an event with the same staying time becomes more likely to be captured in its early phase, as the time interval between successive active time slots shrinks. This result is consistent with the analysis in [17] and [24].

5.5.2 Impact of Maximum Working Time

In Fig. 14(a), we observe that the overall QoM rises with an increasing maximum working time τ_w for all the four schemes, and that both E-CHASE and CHASE outperform JPW or RANDOM with respect to achieved QoM. Moreover, the performance gains of E-CHASE and CHASE over JPW or RANDOM become more significant as τ_w increases. They achieve improvements of about $50.6\,\%$ and $82.1\,\%$, and $49.9\,\%$ and $81.3\,\%$, respectively, when $\tau_w = 9.4\,hours$. The time slot duration here is set to $1\,s$. When

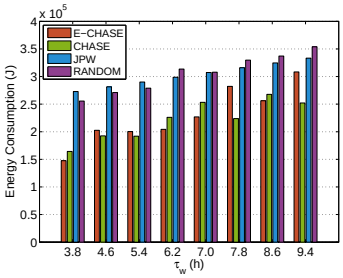


Fig. 15: Comparison of energy consumptions

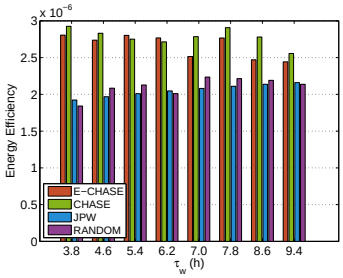


Fig. 16: Comparison of energy efficiency

the duration is reduced to $0.5\ s$, the overall QoM of all the schemes are substantially enhanced, as illustrated in Fig. 14(b). On average, the performance gains by E-CHASE and CHASE over JPW and RANDOM are about $28.8\,\%$ and $35.0\,\%$, $24.3\,\%$ and $30.1\,\%$, respectively.

Next, we compare the proposed algorithms with JPW and RANDOM in terms of energy for the second case. As an MC following JPW needs to always visit all the sensors, the energy overhead for travel can be large. Similarly, although an MC following RANDOM visits the same number of sensors as CHASE, RANDOM chooses these sensors randomly, without trying to manage the expected travel energy, and its energy consumption can still be high. It can be seen from Fig. 15 that on average the energy consumptions for JPW and RANDOM are $37.1\,\%$ and $37.3\,\%$ higher than that of E-CHASE, and $38.9\,\%$ and $39.4\,\%$ higher than that of CHASE. Furthermore, with a large τ_w, the MC under E-CHASE or CHASE is able to include more sensors for charging. Fig. 15 shows that the resulting energy increment can be substantial. Lastly, we evaluate the energy efficiency defined as the ratio of the overall QoM to the total energy consumption. Fig. 16 demonstrates that E-CHASE obtains average gains of $30.0\,\%$ and $27.2\,\%$ over JPW and RANDOM, respectively, whereas the corresponding numbers for CHASE are $35.7\,\%$ and $32.6\,\%$.

6 Conclusion

We have solved the problem of QoM maximization when a sensor network is used to monitor stochastic events, by jointly designing the sensors' mobile wireless charging and activation schedules. The problem has a general event model that admits different utility functions and different probability distributions of the event arrival and staying times. To solve the problem, we first tackled a relaxed version that ignored the MC travel overhead. We developed approximation algorithms for this relaxed problem by transforming it into a submodular function maximization problem, under the condition that the event utility function was concave. Based on solutions to the relaxed problems, we then developed approximation algorithms to solve the original problem when the MC's travel time overhead was also considered. Diverse simulation results verified the theoretical analysis and illustrated the performance of the proposed algorithms relative to two comparison benchmarks. It is interesting for future research to improve the approximation factors of the solutions and account for fairness issues when covering the whole set of PoIs.

Acknowledgment

This work is supported in part by the National Key R&D Program of China under Grant No. 2018YFB1004704 and 2018YFB0803400, in part by the National Natural Science Foundation of China under Grant No. 61502229, 61872178, 61832005, 61672276, 61872173, 61321491, 61520106007, and 61751211, in part by the Natural Science Foundation of Jiangsu Province under Grant No. BK20181251, in part by the Fundamental Research Funds for the Central Universities under Grant 021014380079, in part by China National Funds for Distinguished Young Scientists with No. 61625205, Key Research Program of Frontier Sciences, CAS, No. QYZDY-SSW-JSC002, in part by NSF CNS 1526638, and in part by and S'pore MOE (SUTDT12017004).

References

[1] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh, "Deploying a wireless sensor network on an active volcano," *IEEE internet computing*, 2006.

[2] S. Jiang, "Optimum wireless power transmission for sensors embedded in concrete," Ph.D. dissertation, Florida International University, 2011.

[3] D. Mascareñas, E. Flynn, M. Todd, G. Park, and C. Farrar, "Wireless sensor technologies for monitoring civil structures," *Sound and Vibration*, vol. 42, no. 4, pp. 16–21, 2008.

[4] H. Schioler and T. D. Ngo, "Trophallaxis in robotic swarms - beyond energy autonomy," in *10th International Conference on Control, Automation, Robotics and Vision*, 2008.

[5] T. D. Ngo, H. Raposo, and H. Schioler, "Potentially distributable energy: Towards energy autonomy in large population of mobile robots," in *International Symposium on Computational Intelligence in Robotics and Automation*, 2007.

[6] T. D. Ngo, H. Raposo, and H. Schiøler, "Multiagent robotics: toward energy autonomy," *Artificial Life and Robotics*, vol. 12, no. 1-2, pp. 47–52, 2008.

[7] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava, "Design considerations for solar energy harvesting wireless embedded systems," in *Proceedings of the 4th international symposium on Information processing in sensor networks*. IEEE Press, 2005, p. 64.

[8] S. Meninger, J. Mur-Miranda, R. Amirtharajah, A. Chandrakasan, and J. Lang, "Vibration-to-electric energy conversion," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 9, no. 1, pp. 64–76, 2001.

[9] C. Park and P. Chou, "Ambimax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes," in *IEEE SECON*, 2006.

[10] A. Kurs, A. Karalis, M. Robert, J. D. Joannopoulos, P. Fisher, and M. Soljacic, "Wireless power transfer via strongly coupled magnetic resonances," *SCIENCE*, vol. 317, no. 5834, pp. 83–86, 2007.

[11] M. Erol-Kantarci and H. Mouftah, "Suresense: sustainable wireless rechargeable sensor networks for the smart grid," *IEEE Wireless Communications*, vol. 19, no. 3, pp. 30–36, 2012.

[12] F. Zhang, J. Liu, Z. Mao, and M. Sun, "Mid-range wireless power transfer and its application to body sensor networks," *Open Journal of Applied Sciences*, vol. 2, no. 1, pp. 35–46, 2012.

[13] B. Tong, Z. Li, G. Wang, and W. Zhang, "How wireless power charging technology affects sensor network deployment and routing," in *IEEE ICDCS*, 2010.

[14] Y. Shi, L. Xie, Y. T. Hou, and H. D. Sherali, "On renewable sensor networks with wireless energy transfer," in *IEEE INFOCOM*, 2011.

[15] Z. Li, Y. Peng, W. Zhang, and D. Qiao, "J-RoC: a joint routing and charging scheme to prolong sensor network lifetime," in *IPSN*, 2011.

[16] M. Zhao, J. Li, and Y. Yang, "Joint mobile energy replenishment and data gathering in wireless rechargeable sensor networks," in *ITC*, 2011.

[17] D. K. Y. Yau, N. K. Yip, C. Y. T. Ma, N. S. V. Rao, and M. Shankar, "Quality of monitoring of stochastic events by periodic and proportional-share scheduling of sensor coverage," *ACM Transactions on Sensor Networks*, vol. 7, no. 2, pp. 1–49, Aug. 2010.

[18] S. He, J. Chen, D. K. Yau, H. Shao, and Y. Sun, "Energy-efficient Capture of Stochastic Events by Global- and Local-Periodic Network Coverage," *ACM MobiHoc*, 2009.

[19] S. Tang and L. Yang, "Morello: A quality-of-monitoring oriented sensing scheduling protocol in sensor networks," in *IEEE INFOCOM*, 2012.

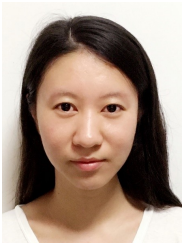
[20] Z. Ren, P. Cheng, J. Chen, D. K. Y. Yau, and Y. Sun, "Dynamic Activation Policies for Event Capture with Rechargeable Sensors," in *IEEE ICDCS*, 2012.

[21] S. Tang, X. Li, X. Shen, J. Zhang, G. Dai, and S. Das, "Cool: On coverage with solar-powered sensors," in *IEEE ICDCS*, 2011.

[22] M. Buettner, R. Prasad, M. Philipose, and D. Wetherall, "Recognizing daily activities with RFID-based sensors," in *ACM Ubicomp*, 2009.

[23] M. Buettner, B. Greenstein, A. Sample, J. Smith, and D. Wetherall, "Revisiting smart dust with RFID sensor networks," in *ACM HotNets*, 2008.
[24] F. Jiang, S. He, P. Cheng, and J. Chen, "On optimal scheduling in wireless rechargeable sensor networks for stochastic event capture," in *IEEE MASS*, 2011.
[25] C. Wang, J. Li, F. Ye, and Y. Yang, "Multi-vehicle coordination for wireless energy replenishment in sensor networks," in *IEEE IPDPS*, 2013.
[26] ——, "Recharging schedules for wireless sensor networks with vehicle movement costs and capacity constraints," in *IEEE SECON*, 2014.
[27] S. Zhang, J. Wu, and S. Lu, "Collaborative mobile charging for sensor networks," in *IEEE MASS*, 2012.
[28] C. Wang, J. Li, F. Ye, and Y. Yang, "Improve charging capability for wireless rechargeable sensor networks using resonant repeaters," in *IEEE ICDCS*, 2015.
[29] C. Wang, J. Li, Y. Yang, and F. Ye, "A hybrid framework combining solar energy harvesting and wireless charging for wireless sensor networks," in *IEEE INFOCOM*, 2016.
[30] H. Dai *et al.*, "Using minimum mobile chargers to keep large-scale wireless rechargeable sensor networks running forever," in *IEEE ICCCN*, 2013, pp. 1–7.
[31] ——, "Minimizing the number of mobile chargers for large-scale wireless rechargeable sensor networks," *Computer Communications*, vol. 46, pp. 54–65, 2014.
[32] T. Wu *et al.*, "Charging oriented sensor placement and flexible scheduling in rechargeable wsn," in *IEEE INFOCOM*, 2019. Accepted.
[33] L. Fu, P. Cheng, Y. Gu, J. Chen, and T. He, "Minimizing charging delay in wireless rechargeable sensor networks," in *IEEE INFOCOM*, 2013.
[34] L. He, Y. Gu, J. Pan, and T. Zhu, "On-demand charging in wireless sensor networks: Theories and applications," in *IEEE MASS*, 2013.
[35] L. He, L. Fu, L. Zheng, Y. Gu, P. Cheng, J. Chen, and J. Pan, "Esync: An energy synchronized charging protocol for rechargeable wireless sensor networks," in *ACM MobiHoc*, 2014.
[36] L. He, P. Cheng, Y. Gu, J. Pan, T. Zhu, and C. Liu, "Mobile-to-mobile energy replenishment in mission-critical robotic sensor networks," in *IEEE INFOCOM*, 2014.
[37] L. Xie, Y. Shi, Y. T. Hou, W. Lou, H. D. Sherali, and S. F. Midkiff, "On renewable sensor networks with wireless energy transfer: the multi-node case," in *IEEE SECON*, 2012.
[38] S. Guo, C. Wang, and Y. Yang, "Mobile data gathering with wireless energy replenishment in rechargeable sensor networks," in *IEEE INFOCOM*, 2013.
[39] L. Xie, Y. Shi, Y. T. Hou, W. Lou, H. D. Sherali, and S. F. Midkiff, "Bundling mobile base station and wireless energy transfer: Modeling and optimization," in *IEEE INFOCOM*, 2013.
[40] L. Xie, Y. Shi, Y. T. Hou, W. Lou, and H. D. Sherali, "On traveling path and related problems for a mobile station in a rechargeable sensor network," in *ACM MobiHoc*, 2013.
[41] P. Zhou, C. Wang, and Y. Yang, "Leveraging target k-coverage in wireless rechargeable sensor networks," in *IEEE ICDCS*, 2017.
[42] H. Dai *et al.*, "Near optimal charging and scheduling scheme for stochastic event capture with rechargeable sensors," in *IEEE MASS*, 2013, pp. 10–18.
[43] T. Wu *et al.*, "Collaborated tasks-driven mobile charging and scheduling: A near optimal result," in *IEEE INFOCOM*, 2019. Accepted.
[44] H. Dai *et al.*, "Safe charging for wireless power transfer," in *IEEE INFOCOM*, 2014, pp. 1105–1113.
[45] ——, "SCAPE: Safe Charging with Adjustable PowEr," in *IEEE ICDCS*, 2014, pp. 439–448.
[46] ——, "Impact of mobility on energy provisioning in wireless rechargeable sensor networks," in *IEEE WCNC*, 2013, pp. 962–967.
[47] ——, "Quality of energy provisioning for wireless power transfer," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 2, pp. 527–537, 2015.
[48] ——, "Omnidirectional chargability with directional antennas," in *IEEE ICNP*, 2016, pp. 1–10.
[49] Lanlan *et al.*, "Radiation constrained fair wireless charging," in *IEEE SECON*, 2017, pp. 1–9.
[50] H. Dai *et al.*, "Optimizing wireless charger placement for directional charging," in *IEEE INFOCOM*, 2017, pp. 2922–2930.
[51] ——, "Radiation constrained scheduling of wireless charging tasks," in *ACM MobiHoc*, 2017, pp. 17–26.
[52] ——, "Safe charging for wireless power transfer," *IEEE/ACM Transactions on Networking*, vol. 25, no. 6, pp. 3531–3544, 2017.
[53] ——, "Robustly safe charging for wireless power transfer," *IEEE INFOCOM*, pp. 1–9, 2018.
[54] N. Yu *et al.*, "Placement of connected wireless chargers," in *IEEE INFOCOM*, 2018, pp. 1–9.
[55] H. Dai *et al.*, "Radiation constrained scheduling of wireless charging tasks," *IEEE/ACM Transactions on Networking*, vol. 26, no. 1, pp. 314–327, 2018.
[56] ——, "SCAPE: Safe Charging with Adjustable Power," *IEEE/ACM Transactions on Networking*, vol. 26, no. 1, pp. 520–533, 2018.
[57] ——, "Wireless charger placement for directional charging," in *IEEE/ACM Transactions on Networking*, vol. 26, no. 4, 2018, pp. 1865–1878.
[58] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection Tree Protocol," in *ACM SenSys*, 2009, pp. 1–14.
[59] T. Zhu, Y. Gu, T. He, and Z.-L. Zhang, "Achieving long-term operation with a capacitor-driven energy storage and sharing network," *ACM Transactions on Sensor Networks*, vol. 8, no. 4, p. 32, 2012.
[60] S. M. Ross, *Introduction to probability models*. Academic press, 2014.
[61] K. Kar and S. Banerjee, "Node placement for connected coverage in sensor networks," in *WiOpt*, 2003.
[62] H. F. Sheikh, I. Ahmad, and D. Fan, "An evolutionary technique for performance-energy-temperature optimized scheduling of parallel tasks on multi-core processors," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 3, pp. 668–681, 2016.
[63] H. F. Sheikh and I. Ahmad, "Niched evolutionary techniques for performance, energy, and temperature optimized scheduling in multi-core systems," in *IEEE IGSC*, 2015, pp. 1–6.
[64] K. A. Dowsland and J. M. Thompson, "Simulated annealing," in *Handbook of natural computing*. Springer, 2012, pp. 1623–1655.
[65] J. Kennedy, "Particle swarm optimization," in *Encyclopedia of machine learning*. Springer, 2011, pp. 760–766.
[66] L. Fu, P. Cheng, Y. Gu, J. Chen, and T. He, "Minimizing charging delay in wireless rechargeable sensor networks," in *IEEE INFOCOM*, 2013.
[67] D. Hochba, "Approximation algorithms for np-hard problems," *ACM SIGACT News*, vol. 28, no. 2, pp. 40–52, 1997.
[68] U. Feige, "A threshold of ln n for approximating set cover," *Journal of the ACM*, vol. 45, no. 4, pp. 634–652, 1998.
[69] A. Schrijver, *Combinatorial optimization: polyhedra and efficiency*. Springer Verlag, 2003.
[70] M. Fisher, G. Nemhauser, and L. Wolsey, "An analysis of approximations for maximizing submodular set functions II," *Polyhedral combinatorics*, pp. 73–87, 1978.
[71] A. Gupta, V. Nagarajan, and R. Ravi, "Robust and maxmin optimization under matroid and knapsack uncertainty sets," *Arxiv preprint arXiv:1012.4962*, 2010.
[72] C. Chekuri and S. Khanna, "On multidimensional packing problems," *SIAMJ. Comput*, vol. 33, no. 4, pp. 837–851, 2004.
[73] C. Chekuri and J. Vondrák, "Randomized pipage rounding for matroid polytopes and applications," *CoRR, abs/0909.4348*, 2009.
[74] A. Badanidiyuru and J. Vondrák, "Fast algorithms for maximizing submodular functions," in *SIAM SODA*, 2014.
[75] H. Zhang, Y. Jiang, S. Rangarajan, and B. Zhao, "Multicast video delivery with switched beamforming antennas in indoor wireless networks," in *IEEE INFOCOM*, 2011.

Haipeng Dai received the B.S. degree in the Department of Electronic Engineering from Shanghai Jiao Tong University, Shanghai, China, in 2010, and the Ph.D. degree in the Department of Computer Science and Technology in Nanjing University, Nanjing, China, in 2014. He is a research assistant professor in the Department of Computer Science and Technology in Nanjing University. His research papers have been published in many prestigious conferences and journals such as ACM MobiSys, ACM MobiHoc, ACM VLDB, ACM SIGMETRICS, IEEE INFOCOM, IEEE ICDCS, IEEE ICNP, IEEE JSAC, IEEE TON, IEEE TMC, IEEE TPDS, and IEEE TOSN. He is an IEEE and ACM member. He received Best Paper Award from IEEE ICNP'15, Best Paper Award Runner-up from IEEE SECON'18, and Best Paper Award Candidate from IEEE INFOCOM'17.

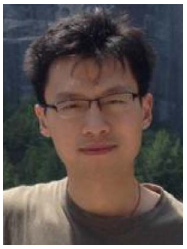


Qiufang Ma received the B.S. degree in the College of Information Engineering from Nanjing University of Finance and Economics, Nanjing, China, in 2016. She is studying towards the masters degree in the Department of Computer Science and Technology in Nanjing University. Her research interests focus on wireless charging.

Xiaobing Wu is with Wireless Research Centre, University of Canterbury, New Zealand. He received his PhD degree in 2009 from Nanjing University, ME degree in 2003 and BS degree in 2000 from Wuhan University, all in Computer Science. His research interests are in the fields of wireless networking and communications, Internet of Things and cyber physical systems. He won Honoured Mention Award in ACM MobiCom 2009 Demos and Exhibitions. He was an Associate Professor at the Department of Computer Science and Technology, Nanjing University.

Guihai Chen received B.S. degree in computer software from Nanjing University in 1984, M.E. degree in computer applications from Southeast University in 1987, and Ph.D. degree in computer science from the University of Hong Kong in 1997. He is a professor and deputy chair of the Department of Computer Science, Nanjing University, China. He had been invited as a visiting professor by many foreign universities including Kyushu Institute of Technology, Japan in 1998, University of Queensland, Australia in 2000, and Wayne State University, USA during Sept. 2001 to Aug. 2003. He has a wide range of research interests with focus on sensor networks, peer-to-peer computing, high-performance computer architecture and combinatorics.

David K.Y. Yau received the B.Sc. from the Chinese University of Hong Kong, and M.S. and Ph.D. from the University of Texas at Austin, all in computer science. He has been Professor at Singapore University of Technology & Design since 2013. Since 2010, he has been Distinguished Scientist at the Advanced Digital Sciences Centre, Singapore. He was Associate Professor of Computer Science at Purdue University (West Lafayette). He received an NSF CAREER award. He won Best Paper award in 2017 ACM/IEEE IPSN and 2010 IEEE MFI. His papers in 2008 IEEE MASS, 2013 IEEE PerCom, 2013 IEEE CPSNA, and 2013 ACM BuildSys were Best Paper finalists. His research interests include cyber-physical system and network security/privacy, wireless sensor networks, smart grid IT, and quality of service. He serves as Associate Editor of IEEE Trans. Network Science and Engineering. He was Associate Editor of IEEE Trans. Smart Grid, Special Section on Smart Grid Cyber-Physical Security (2017), IEEE/ACM Trans. Networking (2004-09), and Springer Networking Science (2012-2013); Vice General Chair (2006), TPC co-Chair (2007), and TPC Area Chair (2011) of IEEE ICNP; TPC co-Chair (2006) and Steering Committee member (2007-09) of IEEE IWQoS; TPC Track co-Chair of 2012 IEEE ICDCS; and Organizing Committee member of 2014 IEEE SECON. He is a senior member of IEEE.



Shaojie Tang is an Assistant Professor in the Department of Information Systems at University of Texas at Dallas. He received his Ph.D degree from Department of Computer Science at Illinois Institute of Technology in 2012. He received B.S. in Radio Engineering from Southeast University, P.R. China in 2006. He is a member of IEEE. His main research interests focus on wireless networks (including sensor networks and cognitive radio networks), social networks, security and privacy, and game theory. He has served on the editorial board of Journal of Distributed Sensor Networks. He also served as TPC member of a number of conferences such as ACM MobiHoc, IEEE ICNP and IEEE SECON.

Xiang-Yang Li received the bachelor degree from Tsinghua University, China, in 1995, and the MS and PhD degrees from University of Illinois at Urbana-Champaign in 2000 and 2001, respectively. He was a professor with the Illinois Institute of Technology. He is currently a professor and an executive dean of the School of Computer Science and Technology with the University of Science and Technology of China. He has authored a monograph entitled Wireless Ad Hoc and Sensor Networks: Theory and Applications and co-edited several books, including Encyclopedia of Algorithms. His research interests include wireless networking, mobile computing, security and privacy, cyber physical systems, and algorithms. He is a recipient of the China NSF Outstanding Overseas Young Researcher (B). He was an IEEE Fellow (2015) and an ACM Distinguished Scientist (2015). He holds the EMC-Endowed Visiting Chair Professorship at Tsinghua University from 2014 to 2016. He has received the six best paper awards, and one best demo award. He is an editor of several journals and has served many international conferences in various capacities.

Chen Tian is an associate professor at State Key Laboratory for Novel Software Technology, Nanjing University, China. He was previously an associate professor at School of Electronics Information and Communications, Huazhong University of Science and Technology, China. Dr. Tian received the BS (2000), MS (2003) and PhD (2008) degrees at Department of Electronics and Information Engineering from Huazhong University of Science and Technology, China. From 2012 to 2013, he was a postdoctoral researcher with the Department of Computer Science, Yale University. His research interests include data center networks, network function virtualization, distributed systems, Internet streaming and urban computing.