Near Optimal Charging and Scheduling Scheme for Stochastic Event Capture with Rechargeable Sensors

Haipeng Dai*, Lintong Jiang*, Xiaobing Wu*, David K. Y. Yau†, Guihai Chen*‡, and ShaoJie Tang§
*State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, Jiangsu 210046, CHINA
†Department of Computer Science, Purdue University, West Lafayette, IN, USA
‡Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, CHINA
§Department of Computer and Information Science, Temple University, Philadelphia, USA
Emails: {dhpphd2003, jltong216, tangshaojie}@gmail.com, {wuxb, gchen}@nju.edu.cn, yau@cs.purdue.edu

***Abstract*—Though much existing work exploits wireless power charging to enhance sensor network performance such as routing and data aggregation, few efforts focus on issues of stochastic event capture. In this paper, we consider the scenario in which a mobile charger (MC) periodically travels within a sensor network to recharge the sensors wirelessly, to maximize the Quality of Monitoring (QoM) for stochastic events. Towards this goal, two closely related research issues need to be addressed. One is how to choose the sensors for charging and decide the charging time for each of them; the other is how to best schedule the sensors' activation schedules according to their received energy. In this paper, we jointly design the charging scheme and sensor schedules to maximize the QoM. We formulate our problem formally as the maximum QoM charging and scheduling problem (MQCSP).**

Obtaining an exact solution of MQCSP is challenging. Thus we first ignore the MC's travel time and study the resulting relaxed version of MQCSP, R-MQCSP. We show both MQCSP and R-MQCSP are NP-hard. For R-MQCSP, however, under a special condition, we prove that it can be formulated as a submodular function maximization problem. This formulation allows a 1/6-approximation algorithm for the general case, and a unified algorithm with a series of approximation factors (up to 1-1/e) for a special case. Then, for MQCSP, we propose approximation algorithms by extending our R-MQCSP results. Finally, we conduct extensive trace-driven simulations to validate our algorithm design. The empirical results corroborate our theoretical analysis.

I. 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. To extend the network lifetime, many approaches have been proposed to harvest environmental energy such as solar [1], vibration [2], and wind [3]. 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 [4], which allow energy to be transferred from one storage device to another wirelessly with reasonable efficiency. Since wireless recharging may guarantee the power supply, independent of the ambient environment, it has found many applications including Smart Grids [5], body sensor networks [6] and civil structures monitoring [7].

Since power chargers are expensive, it is not cost-effective to deploy a large number of these chargers statically for the energy provisioning [8]. Instead, existing practical approaches focus on using one single or multiple mobile chargers (MCs) [9] to move around the sensors and charge them in turn during the travel schedule, for applications such as routing [10] [11] [12], and data gathering [13]. None of these efforts solve the problem of stochastic event capture. But the problem is a fundamental problem in wireless sensor network design, which concerns the scheduling of sensors' duty cycles to maximize their ability to capture interesting events of a probabilistic nature. The problem has received attention for both traditional WSNs [14] [15] [16] and wireless ambient-energy harvesting sensor networks [17] [18].

Moreover, there have emerged a number of applications employing wireless power charging technology to enhance event monitoring performance, such as Wireless Identification and Sensing Platform (WISP) applied in fields ranging from individual activity recognition to large-scale urban sensing [19] [20], and structural health monitoring (SHM) applications [7] where a civil structure is instrumented with sensor nodes capable of being powered solely on energy transmitted to the sensor node wirelessly by a mobile helicopter. Jiang *et al.* [21] are the first to exploit wireless power charging by MCs for efficient stochastic event capture, motivated by these applications. Their objective is to jointly determine the MC's movement and sensor activation schedules to maximize the Quality of Monitoring (QoM), defined as the average information gained per event by the network. They make simplifying assumptions that each sensor can only monitor one Point of Interest (PoI), the charging time for each sensor is identical, the event staying time follows exponential distribution, and all sensors follow a simple periodic schedule (q, p), i.e., sensors monitor PoIs for q time of every p time. We relax these assumptions in this paper.

In this paper, we consider the scenario in which an MC periodically travels within the sensor network and recharges a selected group of sensors wirelessly to enable them for the task 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 a human operator and the operator's 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 in most cases), or the MC must be withdrawn for maintenance for some amount of time between recharge schedules.

In this paper, we address two closely related issues in the wireless recharging and event monitoring. The first is how to choose the sensors for recharge 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 close-together 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 (MQCSP). The coupling between the MC's travel time and the sensor charging time makes our problem highly challenging. Hence, we first ignore the travel time and study the resulting relaxed version of MQCSP, which we call R-MQCSP. Then, based on our results for R-MQCSP, we develop solutions for the general MQCSP problem.

The main contributions of this paper are as follows:

- We analyze the QoM of stochastic event capture, which considers the possibility that the same PoI may be monitored by multiple sensors. We formulate the MQCSP and R-MQCSP problems, and show that both of them are NP-hard.
- We reformulate R-MQCSP as a monotone submodular function maximization problem under a special sufficient condition. This reformulation of R-MQCSP allows an algorithm which achieves $1/6$-approximation for the QoM maximization. Most importantly, all of our solutions based on this reformulation, including following ones, are generic to accommodate general activation schedules, event utility functions, and probability distributions of the event staying times.
- We also discuss a special case where the active time slots constraint, which refers to the constraint on the number of active time slots in a sensor's schedule caused by small battery capacity or long period of charging process, can be lifted. It allows a better and unified algorithm achieving a series of approximation factors (up to $1-1/e$) under different parameter settings.
- On the basis of our R-MQCSP results, we propose an approximation algorithm for MQCSP, which considers the MC's travel time. Besides, we also propose an approximation algorithm for the special case without the active time slots constraint.
- We conduct trace-driven extensive simulations to verify our analytical findings. Simulation results show that our schemes outperform the existing one.

The remainder of the paper is organized as follows. In Section II, we give preliminaries and a formal definition of the MQCSP problem as well as its relaxed version of R-MQCSP. In Section III, 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 IV presents extensive simulations to verify our theoretical results. Section V concludes.

TABLE I
NOTATIONS USED

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
$\mathcal{L}$	The length of sensor schedule
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	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 to sensor v_i
c_i	Charging time factor for sensor v_i
l_i	Active time slot budget of sensor v_i

II. PROBLEM STATEMENT

A. 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 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 some common PoIs. Hence, in general, target o_i is covered by a subset of the sensors V_i.

To prolong the lifetime of the sensors, an 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, some sensors failed to be chosen for charging are doomed to die out. It is reasonable since our primary concern is to maximize the overall QoM under the constraint of limited charging time, rather than the fairness of charging time allocation. We will consider issues related to fairness in further work.

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. To use time for charging as much 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 such a path can be formulated as a Traveling Salesman Problem (TSP). Since TSP 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:

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

The above inequality gives the MC's *working time constraint*. In this paper, we assume a discrete time model for a sensor's activation 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 schedule of sensor v_i by a vector $S_i = (a_{i1}, a_{i2}, \ldots, a_{i\mathcal{L}})$, where component $a_{ij} = 1$ 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 is long enough such that the energy cost for turning the sensor on/off can be ignored.

Denote by P_c the working power of the MC, η_i the charging efficiency for sensor v_i, and p_i the working power of sensor v_i. We define the *charging time factor* c_i as the charging time required for the MC to give sufficient energy for v_i to be active for one time slot, i.e., $c_i = \frac{p_i}{\eta_i P_c \mathcal{L}} \tau$. 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 (this can be guaranteed by properly allocating the recharging time of the MC). 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 because of its limited battery capacity, which we call active time slot budget. We have $l_i = \frac{E_i}{p_i \tau} \mathcal{L}$. Furthermore, we have:

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

which we call the *active time slots constraint*. If $l_i \geq \mathcal{L}$ for any sensor v_i, then we can ignore the active time slots constraint. This situation occurs when the battery capacity is much larger compared to the working power of the sensor (such as ultra-capacitors [22]) or we apply the charging process frequently. Nevertheless, we argue that it is still necessary to consider the general case where active time slots constraint takes effect, since in most applications sensors are equipped with cheap batteries of small capacity, and we may not afford the labor costs of frequent charging processes.

A summary of the notations in this paper is given in Table I.

B. Event Monitoring Model

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

For the event dynamics, we assume that the events at a PoI occur one after another, and the events at the same PoI or different PoIs are spatially and temporally independent [14] [15] [21] [23]. 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, the expected number of event arrivals in a time interval dt equals $\mu_i dt$, where $\mu_i = 1/E(Y)$. As for the event staying time X, we assume that the **pdf** of X is $f(x)$.

We use a binary sensing model for the sensors [24]. 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. For simplicity, we assume that $U_j^i(x) = U(x)$ for all the events at all the PoIs. We assume that the utility function has the following property.

***Observation** 2.1:* The utility function $U(x)$ increases 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 $y \geq x \geq 0$.

Another important assumption is that the events are *identifiable* [14]. (Please see [14] for a justification of the assumption.) That is, when more than one sensor detects the same simultaneously, they will know that it is the same event. Furthermore, if more than one sensors observe the same event simultaneously, they learn exactly the same information.

Prior QoM analysis either considers that a PoI is covered by only one sensor [14], or considers only special cases of the event utility function and event dynamics [15] [17] (for example, only the step utility function is considered in [15]). We generalize the prior analysis to cover other types of the events also.

***Definition** 2.1:* (**Periodic Extension Function**) Given a schedule S_i of sensor v_i, the periodic extension function $\mathbf{S}_i(x)$ ($\mathbf{S}_i : [0, +\infty] \mapsto \{0, 1\}$) of S_i is defined as:

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

We first present the following lemma, which is similar to Theorem 7 in [14].

***Lemma** 2.1:* The QoM of a PoI, say o_i, covered by a single sensor v_j ($v_j \in V_i, |V_i| = 1$) with schedule S_j, whose periodic extension function is $\mathbf{S}_j(x)$, is given by:

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

Proof: The above formula follows from the fact that the overall utility available for any particular event, which starts at time t ($t \in [0, \mathcal{L}]$) and ends at time y ($y \in [t, +\infty)$), depends on the total length of the intersecting region $\int_t^y \mathbf{S}_j(x)dx$. ■

Suppose $S_i = (a_{i1}, \dots, a_{i\mathcal{L}})$ and $S_j = (a_{j1}, \dots, a_{j\mathcal{L}})$ are two different vectors, we define "OR" operation of vectors as $S_i \vee S_j = (a_{i1} \vee a_{j1}, \dots, a_{i\mathcal{L}} \vee a_{j\mathcal{L}})$.

***Lemma** 2.2:* The QoM of PoI o_i covered by multiple sensors $V_i = \{v_{1'}, v_{2'}, \dots, v_{m'}\}$, each of which has schedule S_j ($j = 1', 2', \dots, m'$), is given by:

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

Hence, 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: This follows directly from the identifiable assumption. 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 $f(x)$.

C. Problem Formulation

Summarizing all the objective and constraints in previous chapters, we formulate our problem MQCSP as follows.

$$\begin{aligned} \textbf{Max} \quad & \sum_{i=1}^{n} w_i Q(i) \\ \textbf{s.t.} \quad & (1), (2), (3). \end{aligned}$$

Note that w_i is a normalized weight associated with the PoI o_i, which can be interpreted as the frequency of event occurrence of o_i or the importance of o_i. The decision variables are the subset of sensors V_s selected for charging, the MC travel time $\tau_{TSP}(V_s)$, the charging time for each sensor τ_i, and the activation schedule S_i of each sensor. 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.

D. Roadmap of Our Solution

As we can see from the above formulation, the selection of candidate charging sensors set V_s, the coupling relationship between travel time and charging time allocations for candidate sensors, the active time slots constraint, along with the sophisticated computation of QoM, make the problem highly challenging. Among these factors, we emphasize that the travel time plays an important role in problem hardness. For this reason, to start analyze this problem under a simple setting, we first consider a relaxed version of MQCSP, which is called R-MQCSP and ignores the travel time, i.e., $\tau_{TSP}(V) = 0$. Apart from its benefits to theoretical analysis, we emphasize that the consideration of R-MQCSP is also meaningful in practice, as we typically have τ_w being much bigger than $\tau_{TSP}(V)$ due to the time-consuming charging processes resulted from the low charging efficiencies of MCs. For example, according to [25], the charging time for voltage to reach $1.8\ V$ to power a WISP equipped with a $100\ uF$ capacitor can be as large as 155 seconds, when the RFID reader is 10.0 meters away.

Based on our results for R-MQCSP, we develop solutions for MQCSP by reconsidering the MC's travel time.

III. Theoretical Analysis

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

A. Problem Hardness

We now show that both R-MQCSP and MQCSP are NP-hard, and that they cannot be approximated within a factor better than $(1 - 1/e)$. First of all, we present the following well-known NP-hard problem and a relevant lemma.

***Definition** 3.1:* (**Maximum Coverage Problem**) [26] Given a collection of subsets $S = \{S_1, S_2, \dots, S_m\}$ of the universal set $U = \{e_1, e_2, \dots, e_n\}$, 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** 3.1:* [27] For any $\epsilon > 0$, the Maximum Coverage Problem (MCP) cannot be approximated within a factor $(1 - 1/e + \epsilon)$ unless $P = NP$.

The following theorem shows the complexities of our problems.

***Theorem** 3.1:* Both R-MQCSP and MQCSP are NP-hard. For any $\epsilon > 0$, there is no $(1 - 1/e + \epsilon)$ approximation solution to R-MQCSP or MQCSP unless $P = NP$.

Proof: We can restrict the R-MQCSP problem to the MCP problem by setting $\mathcal{L} = 1$, $w_i = 1/n$ and $c_i = c$ where c is a constant, and assuming $l_i \geq \mathcal{L}$ for any sensor v_i such that the active time slots constraint can be removed. Thus we conclude R-MQCSP is at least as hard as MCP. We omit the details to save space. As for MQCSP, we set $\tau_w/c = k + 1/2$ (k is an integer) and $\tau_{TSP}(V) < 1/2\ c$. Using similar techniques of R-MQCSP analysis, we can also prove that MQCSP is at least as hard as MCP. ■

B. Reformulation of R-MQCSP

Because both R-MQCSP and MQCSP are NP-hard, we can only seek an approximation algorithm to solve R-MQCSP. In the following, we reformulate R-MQCSP as a monotone submodular function maximization problem subject to some constraints. Before going to the detail of the reformulation, we first present some necessary definitions.

***Definition** 3.2:* [28] Let S be a finite ground set. A real-valued set function $f : 2^S \mapsto \mathbb{R}$ is *normalized*, *monotonic* and *submodular* if and only if it satisfies the following 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** 3.3:* [28] Given $S = \bigcup_{i=1}^{k} S_i'$ is the disjoint union of k sets, $l_1, l_2, \ldots, l_k$ are positive integers, a *partition matroid* $\mathcal{M} = (S, \mathcal{I})$ is a matroid where $\mathcal{I} = \{X \in S : |X \cap S_i'| \le l_i \text{ for } i = 1, 2, \ldots, k\}$.

We will show that R-MQCSP fits perfectly well in the realm of a monotone submodular function maximization problem subject to constraints including a partition matroid constraint. We start with a definition of ground set S. Denote by $\mathbf{a}_{ij}$ the activating time slot a_{ij} of sensor v_i, then S is given by:

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

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 $a_{ij'} = 1\,(j' = 1', 2', \ldots, \mathcal{L}')$. Further, S can be partitioned into m disjoint sets, $S_1', S_2', \ldots, S_m'$, which is given by $S_i' = \{\mathbf{a}_{i1}, \mathbf{a}_{i2}, \ldots, \mathbf{a}_{i\mathcal{L}}\}$. We say S_i' is the *candidate activation schedule* of sensor v_i, as any feasible schedule S_i is a subset of S_i'. It is obvious that any scheduling policy X consisting of all sensor schedule S_i, namely $X = \{S_1, S_2, \ldots, S_m\}$, is subject to $|X \cap S_i'| = |S_i| \le l_i$. Then we write the independent sets as:

$$\mathcal{I} = \{X \subseteq S : |X \cap S_i'| \le l_i \text{ for } i = 1, 2, \ldots, m\}. \quad (8)$$

On the other hand, it can be easily proved that $\mathcal{M} = \{S, \mathcal{I}\}$ is a matroid.

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

***Lemma** 3.2:* The working time constraint in R-MQCSP can be written as a knapsack constraint on the ground set S, while the active time slots constraint can be written as a partition matroid constraint.

As a consequence, we can rewrite the optimization problem R-MQCSP as follows:

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

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

***Lemma** 3.3:* If the utility function $U(x)$ is *concave*, then the objective function $f(X)$ in the optimization problem R-MQCSP is a monotone submodular function.

Proof: Its complete proof is available in [29], [30]. ■

It is easy to see that the step utility function, exponential utility function and linear utility function are concave, while the S-shaped utility function and delayed step utility function are not [14]. To make the problem amenable, *we assume that the utility function $U(x)$ is concave hereafter*. Nevertheless, since in most of cases utility function is concave [14]–[18], such treatment will not reduce our contribution significantly.

Algorithm 1 Algorithm for R-MQCSP with active time slots 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 3.4 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 [31] 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 $i = 1, \ldots, m$;

C. Approximation Algorithms for R-MQCSP

Having proved that the objective function of our problem is submodular, now we attempt to find approximation algorithms with and without active time slots constraint for R-MQCSP. We will show that the presence of active time slots constraint makes a huge difference in problem complexity.

1) Approximation Algorithm with Active Timeslots Constraint: For this case, we tailor the approach proposed by Gupta *et al.* [32] to our scenarios, 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 [33] reducing knapsack constraints to partition matroids by an enumeration method. We list the main result of this reduction as follows.

***Lemma** 3.4:* Given any knapsack constraint $\sum_{i=1}^{n} w_i \cdot x_i \le B$ 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 $\sum_{i \in X} w_i \le (1+\epsilon) \cdot B$.
2. $\{X \subseteq [n] | \sum_{i \in X} w_i \le B\} \subseteq \bigcup_{t=1}^{T} \mathcal{P}_t$.

We propose an algorithm as shown in Algorithm 1, which is devised based on the algorithm proposed in [32]. Note that we use notation similar to [32] for consistency.

***Theorem** 3.2:* Algorithm 1 for R-MQCSP with active time slots constraint can achieve $1/6$-approximation, and its time complexity is $O((m\mathcal{L})^2 nT)$.

Proof: Its complete proof is available in [29], [30]. ■

We emphasize that we improves the approximation factor from $\frac{1}{(p+2)(3q+1)} = 1/12$, obtained by [32] 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 2 Unified Algorithm for R-MQCSP without active time slots constraint

```
Input: The objective function f(·), the ground set S, the knapsack
    constraint, the candidate activation schedules S'_1, ..., S'_m.
Output: Solution X and the sensor schedules S_1, ..., S_m.
X = ∅, X_1 = ∅, X_2 = ∅, S_i = ∅ for i = 1, ..., m;
If k = 0, then k' = 1; else k' = k − 1;
X_1 = arg max f(D), ∀D ∈ S, |D| ≤ k', Σ_{a'_i ∈ D} c'_i ≤ τ_w;
for all D ∈ S (|D| = k and Σ_{a'_i ∈ D} c'_i ≤ τ_w) do
    I = S;
    while I\D ≠ ∅ do
        a'_t = arg max_{a'_i ∈ I\D} f_D(a'_i) / c'_i;
        if f_D(a'_t) ≤ 0 then
            break;
        end if
        if Σ_{a'_i ∈ D} c'_i + c'_t ≤ τ_w then
            D ← D ∪ {a'_t};
        else
            I = I\{a'_t};
        end if
    end while
    if f(X_2) ≤ f(D) then
        X_2 = D;
    end if
end for
X ← arg max{f(X_1), f(X_2)}, S_i = X ∩ S'_i for i = 1, ..., m;
```

Algorithm 1, than that in [32]. Besides, although the algorithm proposed in [34] for 1 matroid and k knapsack constraints can achieve an $(1-1/e-\varepsilon)$-approximation, it requires that all sets of at most 10^{12} items should be enumerated to form a feasible solution at the first stage, and thus makes itself purely theoretical.

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.

2) Approximation Algorithm without Active Timeslots Constraint: If $l_i \geq \mathcal{L}$ for any sensor v_i, then the active time slots constraint can be safely relaxed. This situation occurs when the battery capacity is much larger compared to the working power of the sensor (such as ultra-capacitors [22]) or we apply the charging process frequently. Thus we can resort to a unified greedy algorithm, namely Algorithm 2, to find an optimized QoM. Note that in this algorithm $\mathbf{c}'_i$ refers to the corresponding charging time factor for time slot $\mathbf{a}'_i$.

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 achieves the highest QoM. The second part starts from every feasible subset D with cardinality k, and search 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 propose the following theorem based on the results obtained by [35].

Theorem 3.3: Algorithm 2 for R-MQCSP without active time slots constraint achieves approximation factor $\frac{1-1/e}{2}\approx 0.3161$, $\frac{1-1/e}{2-1/e}\approx 0.3873$, $\frac{1-1/e}{3/2-1/e}\approx 0.5584$, $1-1/e\approx 0.6321$ for $k=0,1,2,3$, respectively. Its time complexity is $O((m\mathcal{L})^{k+2}n)$.

Proof: Its complete proof is available in [29], [30]. ■

According to Theorem 3.1, we claim that Algorithm 2 for $k=3$ is in fact the best possible for any polynomial-time approach unless $P=NP$.

Algorithm 3 Algorithm for MQCSP

```
Input: The sensors set V = {v_1, ..., v_m}, the PoIs set O =
    {o_1, ..., o_n}, the objective function f(·), the ground set S,
    the partition matroid M, the knapsack constraint, the candidate
    activation schedules S'_1, ..., S'_m.
Output: The sensor schedules S_1, ..., S_m.
1: Call Algorithm 1 to obtain the solution X_R for R-MQCSP;
2: Sort X_R = {a'_1, ..., a'_K} such that a'_t =
   arg max_{a'_i ∈ X_R \ X_R^{t-1}} f_{X_R^{t-1}}(a'_i) / c'_i  (X_R^{t-1} = {a'_1, ..., a'_{t-1}});
3: X ← X_R, t = K;
4: while τ_TSP(⋃_{|X ∩ S'_i| > 0} v_i) > τ_w − Σ_{a'_i ∈ X} c'_i do
5:     X ← X\a'_t;
6:     t = t − 1;
7: end while
8: S_i ← X ∩ S'_i for i = 1, ..., m;
```

D. Approximation Algorithms for MQCSP

On the basis of the proposed constant approximation algorithms for R-MQCSP, now we consider the original problem MQCSP and propose approximation algorithms.

As shown in Algorithm 3, we call Algorithm 1 at the first step to obtain a feasible solution X_R for R-MQCSP. Subsequently, we sort elements in X_R in descending order by their cost efficiency. We iteratively remove the element with 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 MQCSP.

Theorem 3.4: Algorithm 3 for MQCSP can achieve $\frac{1}{6}(1-\frac{\tau_{TSP}(V)+max\{c_i\}_{i=1}^m}{\tau_w})$-approximation. The time complexity of this algorithm is $O((m\mathcal{L})^2 nT)$.

Proof: Its complete proof is available in [29], [30]. ■

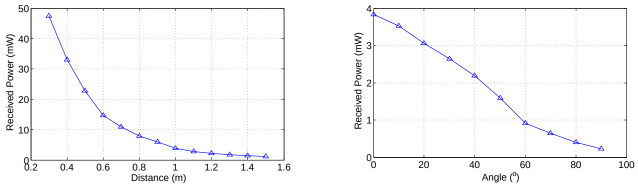
Further, if the active time slots constraint can be removed in this case, we can modify Algorithm 3 by replacing Algorithm 1 called at step 1 with Algorithm 2. The following theorem gives the performance guarantee of this revised algorithm.

Theorem 3.5: The revised algorithm for MQCSP without active time slots constraint can achieves approximation factor $\frac{1-1/e}{2}c$, $\frac{1-1/e}{2-1/e}c$, $\frac{1-1/e}{3/2-1/e}c$, $(1-1/e)c$ 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 the details of proof to save space. ■

IV. Performance Evaluation

In this section, we first describe the traces collected by our testbed. Then we present simulation results to verify our theoretical findings.



(a) Received Power Vs. Distance (b) Received Power Vs. Angle

Fig. 1. Received Power with Varying Distance or Angle

A. Evaluation Setup

1) Trace Collection: For trace collection, we utilized the energy harvesting development kit P2110-EVAL-01 for wireless sensors produced by Powercast [36]. The kit is mainly composed of an RF transmitter, RF energy harvesting receiver boards and antennas, and wireless sensor boards. In this kit, the TX91501 transmitter is the source of energy. The wireless sensor boards are powered by the P2110 Powerharvester Receiver converting RF energy into DC power.

As illustrated in Fig. 1, we can see that the received power of the wireless sensor board decreases dramatically with an increasing distance between RF transmitter and wireless sensor board when the sensor board is put right in the front of the TX91501 transmitter. The situation is similar when the angle from the line through the transmitter and sensor board to the reference direction of the transmitter increases from $0°$ to $90°$.

2) Parameter Settings: Unless otherwise stated, we use the following parameter settings. According to the collected traces, we set the range of received power of sensor to $[15\,mW, 45\,mW]$, which can be interpreted into a charging efficiency η_i between $[0.5\,\%, 1.5\,\%]$. We randomly distribute 20 sensors and 50 PoIs in a $120\,m \times 120\,m$ region throughout the simulations, with any PoI being 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 MC is set to $3\,W$. The battery capacity of sensor is randomly selected from a range $[100\,J, 1000\,J]$. Furthermore, we set the sensing radius of sensor to $20\,m$, and the sensor schedule length $\mathcal{L} = 4$. We assume the considered event type has step utility function and its event staying time follows $f(x) = \lambda e^{-\lambda x}$ where $\lambda = 1$. We set $\tau = 2\,week$, $\tau_w = 8.2\,hour$, and the MC's speed $\nu_{MC} = 0.05\,m/s$. Besides, the default value of duration of time slot is set to be $1\,s$.

B. Performance Evaluation for Algorithms of R-MQCSP

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

1) Evaluation of Different Event Types: In this scenario, we focus on the event types whose event staying time follows $f(x) = \lambda e^{-\lambda x}$ $(\lambda = 0.25, 0.5, 1, 2)$ as in [14], [21], while its utility function $U(x)$ either follows step utility function or exponential utility function $f(x) = Ae^{-Ax}$, where $A = 5$ as in [14]. Note that we use Algorithm 2 with $k = 3$. It can be seen in Figs. 2 and 3 that the overall QoM always increases with an increasing τ_w. Nevertheless, the marginal gain of QoM diminishes as τ_w increases. This is mainly due to the facts that event capture utility function is concave and redundant coverage of PoIs will occur when sensors have larger active time slot budgets with 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, and therefore its probability of being detected and capture utility will be enhanced.

Besides, by comparing Fig. 2 and 3, we conclude that the achieved overall QoMs for events with step utility function always exceeds that for events with exponential utility function. This can be explained by the difference in efficiency of event capture. For events with step utility function, full information about an event is obtained instantaneously on detection. On the contrary, it needs a lot of time to obtain most information of an event with exponential utility function, and infinite time for full information, which is relatively inefficient.

2) Evaluation of Different Control Parameter k: We proceed to evaluate the impact of control parameter k on the overall QoM in Algorithm 2, and plot all the results in Fig. 4. Not surprisingly, it can be seen that the larger k we choose, the higher overall QoM we obtain. Nevertheless, the distinction between the overall QoMs of different k is not obvious. This observation also implies that we can choose a small k to reduce the time complexity without incurring large performance degradation.

In addition, it can be observed that the overall QoM exceeds $1 - 1/e \approx 0.63$ in Fig. 4, which indeed corroborates Theorem 3.3 as the optimal overall QoM is absolutely no more than 1.

3) Varying Period of Charging Process: To see how the period of 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 in a relatively smaller range $[20\,mW, 35\,mW]$ to ease the computation. Fig. 5 exhibits 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 result in a reduced QoM. Moreover, the fact that the overall QoM is larger than $1/6 \approx 0.17$ actually supports Theorem 3.2.

Again, we can see that the achieved overall QoM improves with a larger τ_w.

C. Performance Evaluation for Algorithms of MQCSP

We proceed to verify the performance of algorithms of MQCSP, which consider the travel time. We inherit the parameters defined in IV-B3.

1) Impact of MC's Speed on Working Time Allocation and QoM: Fig. 6 shows that if the speed of MC ν_{MC} increases, the travel time will be reduced, leading to a larger charging time. Note that both travel time and aggregate charging time are normalized with respect to maximum working time τ_w. It can be seem that the portion of the aggregate travel time is below $10\,\%$ when ν_{MC} grows to $0.1\,m/s$, which is still quite small. Another important finding about Fig. 6 is that the sum of travel time and aggregate charging time is not necessarily

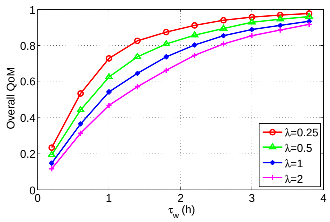


Fig. 2. QoM Vs. τ_w for Step Utility Function

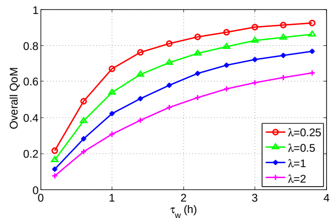


Fig. 3. QoM Vs. τ_w for Exponential Utility Function

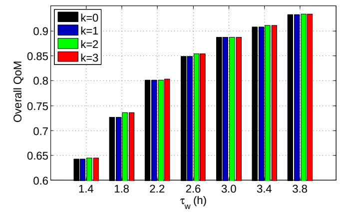


Fig. 4. Evaluation of Impact of Control Parameter k

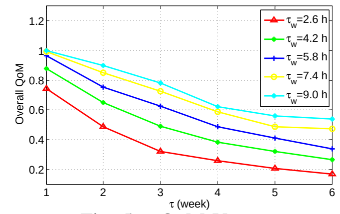


Fig. 5. QoM Vs. τ

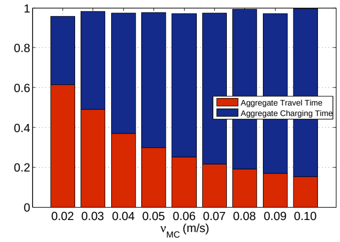


Fig. 6. Impact of MC's Speed on Working Time Allocation

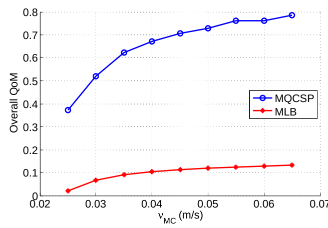


Fig. 7. Impact of MC's Speed on Overall QoM

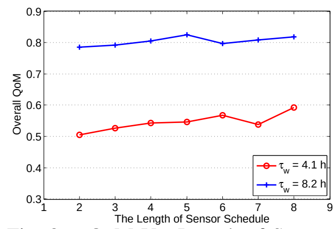


Fig. 8. QoM Vs. Length of Sensor Schedule

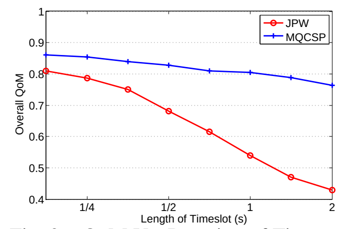


Fig. 9. QoM Vs. Duration of Timeslot

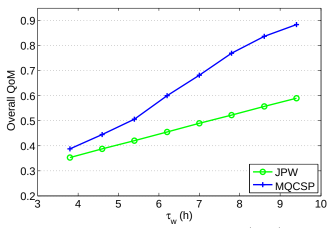


Fig. 10. QoM Vs. $\tau_w(1\,s)$

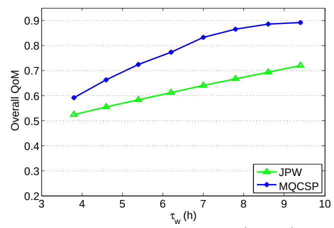


Fig. 11. QoM Vs. $\tau_w(0.5\,s)$

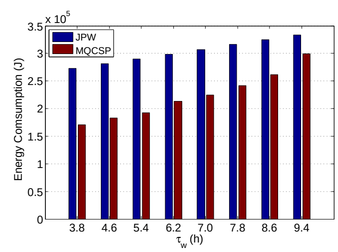


Fig. 12. Comparison of Energy Consumptions

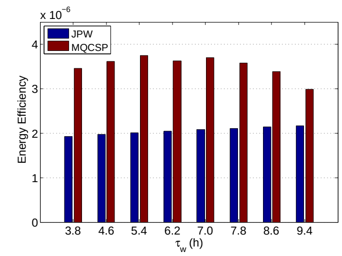


Fig. 13. Comparison of Energy Efficiency

equal to τ_w (the gap is up to $4\,\%$ when $\nu_{MC} = 0.02$). This situation happens as we require the active time slot budget l_i for each sensor must be integer. This requirement can be relaxed, and we can assign remaining working time to charging sensors. Consequently, overall QoM will be further improved. We will discuss it in future work to save space.

Furthermore, the overall QoM is enhanced with a fast speed of MC, as the red solid line shows in Fig. 7. Moreover, it is always bigger than the "maximum" lower bound, which is referred to as the green dotted line "MLB" in the figure, and is given by $\frac{1}{6}(1-\frac{\tau_{TSP}(V)+max\{c_i\}_{i=1}^{m}}{\tau_w})$. This finding corroborates Theorem 3.4.

2) Varying Length of Sensor Schedule: Fig. 8 demonstrates the trend of the overall QoM with the increasing length of sensor schedule $\mathcal{L}$ with $\tau_w = 4.1\,hour$ and $\tau_w = 8.2\,hour$ respectively. It is easy to find that for any $\mathcal{L}$, the overall QoM of its multiples always exceeds that of its own, such as $\mathcal{L} = 2$, $\mathcal{L} = 4$ and $\mathcal{L} = 8$. However, the overall QoM doesn't increase monotonically with $\mathcal{L}$. We will investigate this phenomenon thoroughly in the future.

D. Performance Comparison with Existing Works

In this section we compare our scheme to the scheme, Joint Periodic Wake-up (JPW), proposed by [21]. To make the comparison feasible, we first tailor JPW to our concerned scenarios. That is, a MC dictated by JPW evenly distributes its charging time to each sensor in the field, and only when it arrives the position of a sensor can it charge this sensor (this assumption is more practical since the received power decreases dramatically with increased distance or angle as Fig. 1(a) and Fig. 1(b) indicate). Of course, the energy cost for turning the sensor on/off is still ignored. In addition, we set $\nu_{MC} = 0.1\,m/s$, $\eta_i = 1\,\%$ for any sensor v_i, and the time duration of duty cycle in JPW is exactly equal to that of sensor schedule. The travel power of the MC is set to $50\,W$. For other parameters, we use the same parameters as those in IV-B3.

1) Varying Length of Time Slot: Note that the default value of duration of time slot is $1\,s$ in above sections. Now we vary the duration of time slot, and plot the overall QoMs of both JPW and MQCSP in Fig. 9. As can be seen, our algorithm MQCSP always outperforms JPW, especially with a long duration of time slot. Moreover, the overall QoMs rises with an decreasing duration of time slot for both schemes, since events with the same staying time becomes more likely to be detected (as time interval between non-consecutive active time slots shrinks) and captured in its early phase with high utility. This result is consistent with that of [14] and [21].

2) Varying Maximum Working Time: In Fig. 10, we observe that the overall QoMs rise with an increasing maximum working time τ_w for both JPW and MQCSP, and MQCSP is more competitive than JPW due to its higher QoMs. Besides, the performance improvement of MQCSP over JPW becomes more significant as τ_w increases, and achieves about $50\,\%$ when $\tau_w = 9.4\,hour$. Note that the duration of time slot here is set to $1\,s$. For the case where duration of time slot is reduced to $0.5\,s$, the overall QoMs of both schemes are substantially enhanced, as illustrated in Fig. 11. On average, our algorithm obtains a performance gain of about $24\,\%$ over JPW.

Next, we compare our algorithm with JPW in terms of energy for the second case. As a MC following JPW needs to visit all sensor nodes under any circumstance, the energy overhead for travel is huge. So does the overall energy consumption. It can be seen in Fig. 12 that the energy consumption for JPW is $11\,\% \sim 58\,\%$ higher than that of MQCSP. On the other hand, with a large τ_w, the MC dictated by MQCSP is able to include more sensor nodes for charging, and thereby incurs substantial energy increment, which is also indicated by Fig. 12. In addition, we attempt to evaluate the energy efficiency of two schemes, which is define as (overall QoM)/(energy consumption). Fig. 13 demonstrates that MQCSP enjoys a gain of $38\,\% \sim 80\,\%$ over JPW.

ACKNOWLEDGMENT

This work is supported by National 973 project of China under Grant No.2012CB316200, National Natural Science Foundation of China under Grant No.61133006, No.60903179 and No.61021062, and Research and Innovation Project for College Graduate Students of Jiangsu Province in 2012 under Grant No.CXZZ12_0056. We would also like to thank Qingyuan Jiang for his valuable work.

V. CONCLUSION

In this paper, we have studied the maximum QoM charging and scheduling problem. This problem targets a general event model with arbitrary utility function, staying time and schedule. To pave the way to this problem, we studied its relaxed version, and approximately solved it by transforming this version into a submodular function maximization problem, under the condition that event utility function is concave. Consequently, the original problem has also been approximately solved by considering the travel time of the MC.

Trace-driven simulation results confirm the correctness of the theoretical analysis. In the future, we will pursue algorithms providing better approximation factors. And we will generalize our solution by taking into account the fairness of charging time allocation.

REFERENCES

[1] 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.

[2] S. Meninger, J. Mur-Miranda, R. Amirtharajah, A. Chandrakasan, and J. Lang, "Vibration-to-electric energy conversion," *VLSI*, vol. 9, no. 1, pp. 64–76, 2001.

[3] C. Park and P. Chou, "Ambimax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes," in *SECON'06*, vol. 1, 2006, pp. 168–177.

[4] 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.

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

[6] 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.

[7] 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.

[8] H. Dai, L. Xu, X. Wu, C. Dong, and G. Chen, "Impact of mobility on energy provisioning in wireless rechargeable sensor networks," in *WCNC*, 2013.

[9] H. Dai, X. Wu, L. Xu, G. Chen, and S. Lin, "Using minimum mobile chargers to keep large-scale wireless rechargeable sensor networks running forever," in *ICCCN*, 2013.

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

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

[12] 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.

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

[14] 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," *TOSN*, vol. 7, no. 2, pp. 1–49, Aug. 2010.

[15] 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," *MobiHoc*, p. 155, 2009.

[16] S. Tang and L. Yang, "Morello: A quality-of-monitoring oriented sensing scheduling protocol in sensor networks," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 2676–2680.

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

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

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

[20] M. Buettner, B. Greenstein, A. Sample, J. Smith, and D. Wetherall, "Revisiting smart dust with rfid sensor networks," in *Proceedings of the 7th ACM Workshop on Hot Topics in Networks*, 2008.

[21] F. Jiang, S. He, P. Cheng, and J. Chen, "On optimal scheduling in wireless rechargeable sensor networks for stochastic event capture," in *MASS*, 2011.

[22] T. Zhu, Y. Gu, T. He, and Z.-L. Zhang, "Achieving long-term operation with a capacitor-driven energy storage and sharing network," *TOSN*, vol. 8, no. 4, p. 32, 2012.

[23] H. Dai, X. Wu, L. Xu, and G. Chen, "Practical scheduling for stochastic event capture in wireless rechargeable sensor networks," in *WCNC*, 2013.

[24] K. Kar, S. Banerjee *et al.*, "Node placement for connected coverage in sensor networks," 2003.

[25] L. Fu, P. Cheng, Y. Gu, J. Chen, and T. He, "Minimizing charging delay in wireless rechargeable sensor networks," in *INFOCOM*, 2013.

[26] D. Hochba, "Approximation algorithms for np-hard problems," *ACM SIGACT News*, vol. 28, no. 2, pp. 40–52, 1997.

[27] U. Feige, "A threshold of ln n for approximating set cover," *Journal of the ACM (JACM)*, vol. 45, no. 4, pp. 634–652, 1998.

[28] A. Schrijver, *Combinatorial optimization: polyhedra and efficiency*. Springer Verlag, 2003.

[29] "http://gps.nju.edu.cn/˜hpdai/dh/mqcsp-tr.pdf."

[30] "http://cs.nju.edu.cn/wuxb/mqcsp-tr.pdf."

[31] M. Fisher, G. Nemhauser, and L. Wolsey, "An analysis of approximations for maximizing submodular set functions II," *Polyhedral combinatorics*, pp. 73–87, 1978.

[32] A. Gupta, V. Nagarajan, and R. Ravi, "Robust and maxmin optimization under matroid and knapsack uncertainty sets," *Arxiv preprint arXiv:1012.4962*, 2010.

[33] C. Chekuri and S. Khanna, "On multidimensional packing problems," *SIAMJ. Comput*, vol. 33, no. 4, pp. 837–851, 2004.

[34] C. Chekuri and J. Vondrák, "Randomized pipage rounding for matroid polytopes and applications," *CoRR, abs/0909.4348*, 2009.

[35] H. Zhang, Y. Jiang, S. Rangarajan, and B. Zhao, "Multicast video delivery with switched beamforming antennas in indoor wireless networks," in *INFOCOM*, 2011.

[36] PowerCast. [Online]. Available: www.powercastco.com