Collaborative Mobile Charging for Sensor Networks

Sheng Zhang[†‡], Jie Wu[‡], and Sanglu Lu[†]
[†]State Key Lab. for Novel Software Technology, Nanjing University, P. R. China
[‡]Department of Computer and Information Sciences, Temple University, USA

***Abstract*—The limited battery capacity of sensor nodes has become the biggest impediment to wireless sensor network (WSN) applications. Two recent breakthroughs in the areas of wireless energy transfer and rechargeable lithium batteries promise the use of mobile vehicles, with high volume batteries, as mobile chargers that transfer energy to sensor nodes wirelessly. In this paper, for the first time, we envision a novel charging paradigm: *collaborative mobile charging*, where mobile chargers are allowed to charge each other. We investigate the problem of scheduling multiple mobile chargers, which collaboratively recharge sensors, to maximize the ratio of the amount of payload energy to overhead energy, such that every sensor will not run out of energy. We first consider the uniform case where all sensors consume energy at the same rate, and propose a scheduling algorithm, *PushWait*, which is proven to be optimal in this case and can cover a one-dimensional WSN of infinite length. Then, in the non-uniform case, which is conjectured to be NP-hard, we first present two observations from space and time aspects to remove some impossible scheduling choices, and we propose our heuristic algorithm, *ClusterCharging*(β), which clusters sensors into groups and divides a scheduling cycle into charging rounds. Its approximation ratio is also presented. Extensive evaluations confirm the efficiency of our algorithms.**

***Index Terms*—Collaborative mobile charging, wireless energy transfer, wireless sensor networks.**

I. Introduction

Many applications of wireless sensor networks (WSNs) [1], such as structural health monitoring for the Golden Gate Bridge [2], agricultural rain-fed farming decisions [3], and forest fire detection [4], desire a long-lived WSN. However, sensor nodes are typically supplied by batteries that can only store a limited amount of energy, which has become the biggest impediment. Therefore, a lot of efforts, including energy conservation [5–7], energy harvesting [8, 9], and sensor reclamation [10], have been devoted to prolonging the lifetime of WSNs. However, energy conservation cannot compensate for energy depletion; energy harvesting is neither controllable nor predictable; sensor reclamation is costly and impractical when sensors are deployed in the deep ocean, on bridge surfaces, or in containers of hazardous materials.

We recently observed two particular breakthroughs in the areas of *wireless energy transfer* [11, 12] and *rechargeable lithium batteries* [13]. Wireless energy transfer is the transmission of electric energy from a power source to a receiver without any interconnecting conductors. Rechargeable lithium batteries with high energy density and high charge/discharge capability are identified in [13]. Armed with these two technologies, some studies [14–17] employed mobile vehicles of high volume batteries as mobile chargers to deliver energy to sensors. However, most of them [14–16] assume that a mobile charger has a sufficient amount of battery to cover the entire WSN and to make a round trip back to the base station. This model will become invalid when there is a remote area where even a dedicated charger with full battery energy cannot reach before running out of energy.

In this paper, we introduce a novel charging paradigm: *collaborative mobile charging*, where mobile chargers are allowed to charge each other. That is to say, multiple mobile chargers start from the base station with full energy, and after some time, some of them can intentionally gather at a *rendezvous point* to recharge others or to be recharged. We shall see that this collaborative paradigm not only enlarges the charging coverage, but also improves the energy efficiency since chargers in existing methods may return to the base station with residual energy. The scheduling problem of collaborative mobile charging in a general WSN is very complicated. As a first step, to initiate a meaningful study, this paper narrows the scope of this problem to a manageable extent: we consider one-dimensional (1-D) WSNs and leave 2-D as future work. The linear structure of 1-D WSNs can be utilized to reduce maintenance costs, increase routing efficiency, and improve network reliability [18]; therefore, these kind of WSNs have a broad array of applications, ranging from oil/gas/water pipeline monitoring [19] to driver-alert systems [18] to bridge and international border protection [20].

This paper focuses on the following problem: given a 1-D WSN and battery capacity constraints, how can we schedule multiple mobile chargers, which collaboratively recharge sensors, to maximize the ratio of the amount of payload energy to overhead energy, such that every sensor will not run out of energy? To gain a better understanding, we first consider the uniform case of this problem, where all sensors consume energy at the same rate, for which we propose an algorithm called *PushWait*. We prove the optimality of *PushWait* in this case. Then, in the non-uniform case of this problem, we conjecture that the problem becomes NP-hard and propose a heuristic algorithm called *ClusterCharging*(β) with guaranteed performance. We evaluate the performance of our algorithms with extensive simulations. The contributions of this paper are summarized as follows:

(1) To our best knowledge, we are the first to consider the collaborative mobile charging paradigm. By means of examples, theoretical analysis and experimental evaluations, this paper demonstrates the advantages of this novel paradigm in coverage and energy efficiency.

(2) For the uniform case of the scheduling problem, we propose a scheduling algorithm, *PushWait*, which is proven

to be optimal and can cover a 1-D WSN of any length. A variation of *PushWait* that uses dedicated chargers to substitute roundtrip chargers is also presented.

(3) For the non-uniform case, which is conjectured to be NP-hard, we first present two observations from space and time aspects to remove some impossible scheduling choices. Then, we propose our heuristic algorithm, *ClusterCharging*(β), which clusters sensors into groups and divides a scheduling cycle into charging rounds. Its approximation ratio is also presented.

II. Problem Description

A. Network Model

We consider a set of sensor nodes that are uniformly distributed, unit distance apart, along a one-dimensional straight line to the east of a base station (BS), as shown in Fig. 1. There are, in all, N sensor nodes, say $s_1, s_2, ..., s_N$. Sensor s_i consumes r_i amount of energy per unit time. All nodes are assumed to have the same battery capacity, say b. The *recharging cycle* of a sensor is defined as the time period that this sensor of full energy can survive without being charged. Denote the recharging cycle of s_i as τ_i; we have $\tau_i = b/r_i$.

B. Charging Model

A mobile charger (MC) has a maximum battery capacity of B and consumes c amount of energy per unit distance. The base station BS serves as data sink as well as the energy source. Mobile chargers start from the BS with full batteries, charge sensors, finally come back to the BS, and then get themselves recharged by the BS. Both the movement of the mobile chargers and the process of wireless charging share the same pool of battery energy.

The energy transfer efficiencies of BS-to-MC, MC-to-MC, and MC-to-sensor are all assumed to be 1, i.e., there is no energy loss. The corresponding charging time is negligible compared to the traveling time of mobile chargers.

Typically, the recharging cycle (or the lifetime) of a sensor is several months; while the time for a charger traveling from the BS to the farthest sensor in a WSN is usually several hours, or at most several days. Thus, in this paper, we assume that any two charging rounds have no intersection, i.e., mobile chargers can always accomplish a charging round, return to the BS, and wait for another charging round[1].

C. Performance Measure

When scheduling, we must decide the actions (such as, recharging a sensor or another charger, being charged, waiting, etc.) of each mobile charger in its time-space trajectory. A scheduling is said to be *feasible* if (i) all sensor nodes do not die, i.e., each sensor node will get charged before running out of energy, and (ii) all MCs are able to return to the BS to be serviced (e.g., replacing or recharging its battery).

[1]We make this assumption for the brevity of presentation. As we shall see shortly, when the recharging cycle is smaller than the MC round-trip time, pipeline-like parallel *PushWait* could still achieve its optimality.

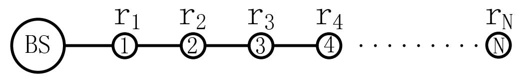


Fig. 1: Problem description

We define the *scheduling cycle* of a scheduling to be the time interval between two consecutive points of time when all sensors are fully charged. Although this definition seems strange at a first glance, we shall see its generality. It can be applied to the uniform case problem (Section III), where the scheduling cycle equals the recharging cycle of each sensor, and it can also be applied to the non-uniform case problem (Section IV), where the scheduling cycle contains more than one recharging cycle of a sensor.

In a scheduling cycle, denote the energy eventually obtained by sensors as *payload energy* ($E^{payload}$), and the energy consumed by MCs′ movements as *overhead energy* ($E^{overhead}$). The efficiency *ratio* of the scheduling can be defined as:

$$ratio = E^{payload}/E^{overhead} \tag{1}$$

A feasible scheduling cyclically charges sensor nodes to make a sensor network long-lived, so this definition characterizes the long-term efficiency of a scheduling well.

D. Scheduling as an Optimization Problem

***Problem** 1:* (Collaborative mobile charging scheduling problem ($CMCS$)) Given a 1-D WSN with parameters b and r_i, how can we find a feasible scheduling of chargers, with parameters c and B, so as to maximize the ratio defined above.

In order to have a better understanding of the *CMCS* problem, we first consider a uniform case where all sensors consume energy at the same rate in Section III; then, we study the non-uniform case in Section IV with the knowledge obtained from the uniform case.

III. *CMCS* with uniform energy consumption rate

In the uniform case, sensors consume energy at the same rate, which is denoted as r, then all sensors have the same recharging cycle, i.e., $\tau = b/r$. This uniformity makes the scheduling become simple: in each recharging cycle, we let the MCs charge all of the sensors and then wait for the next recharging cycle. Therefore, in this case, the scheduling cycle equals the recharging cycle. We then have $E^{payload} = N \cdot b$ is fixed, so the objective of maximizing the ratio of $E^{payload}$ to $E^{overhead}$ is reduced to minimizing $E^{overhead}$.

A. Motivational Examples

We use the following examples to demonstrate the benefits of collaborative mobile charging and to motivate our algorithm design. Three different scheduling schemes are shown in Fig. 2. The former two schemes do not consider collaboration, while the third does. We denote K as the number of MCs, L_i ($1 \le i \le K$) as the farthest point that MC_i reaches, and also let $L_{K+1} = 0$ for compatibility. Figs. 2(a), 2(b), and 2(c) illustrate the time-space view as well as the maximum coverage of these three scheduling schemes, respectively. The settings are $B = 80J, b = 2J, c = 3J/m$, and $K = 3$.

Scheme I: Each MC charges each sensor an mount of b/K energy, and each sensor is charged by all passing MCs. As shown in Fig. 2(a), 12 sensors can be covered.

Scheme II: Each sensor is charged by only one MC. MC_i ($1 \leq i \leq K$) charges sensors from L_{i+1} to L_i, its residual energy at L_i is just enough for it to return to the BS. Fig. 2(b) shows the entire process, where 13 sensors can be covered.

Scheme III: Each sensor is charged by only one MC. MC_i ($2 \leq i \leq K$) charges sensors from L_{i+1} to L_i, and it transfers energy to MC_{i-1}, MC_{i-2},..., and MC_1 until they are at their full energy capacity at L_i, and then it just has enough energy to return to the BS. Fig. 2(c) illustrates this scheme, where MC_3 charges MC_2 and MC_1 at A, and MC_2 charges MC_1 at B. This time, 17 sensors can be covered.

In summary, given a fixed number of MCs, scheme III can cover more sensors than Schemes I and II; collaboration makes scheduling more energy-efficient in the sense that scheduling with collaboration consumes less $E^{overhead}$ than scheduling without collaboration to deliver the same amount of $E^{payload}$.

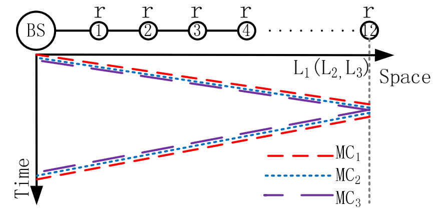
B. *PushWait*

Recall the objective of our scheduling is to minimize $E^{overhead}$, which is consumed by MCs' movement. The basic idea of *PushWait* is to use as less MCs as possible to carry the residual energy of all MCs through letting some MCs charge others at some rendezvous points. *PushWait* is illustrated as:

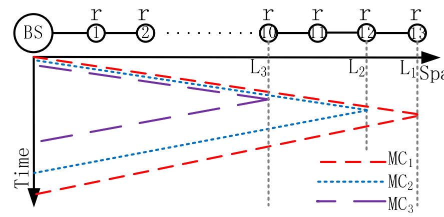
- MC_i charges sensors between L_{i+1} and L_i to their full batteries. At L_i, MC_i transfers energy to MC_{i-1}, MC_{i-2},..., and MC_1 until they are at their full energy capacity. Then MC_i waits at L_i, and all of the other $i-1$ MCs keep moving forward.
- After MC_{i-1}, MC_{i-2},..., and MC_1 return to L_i, where MC_i waits for them, MC_i evenly distributes its residual energy among i MCs (including MC_i). This will make them just have enough energy to return to L_{i+1}.

In *PushWait*, each MC_i follows the iterative process below: starts from the BS with full battery, gets fully charged at locations L_K, L_{K-1},..., and L_{i+1}, charges sensor nodes between L_{i+1} and L_i, charges MC_{i-1}, MC_{i-2},..., and MC_1 at L_i, waits for these MCs to return, then evenly distributes its residual energy among these i MCs (including MC_i itself) and moves towards the BS. The reason of naming this scheduling after "*PushWait*" is clear: from the point of view of MC_i, it pushes the other MCs to move forward and waits for their return.

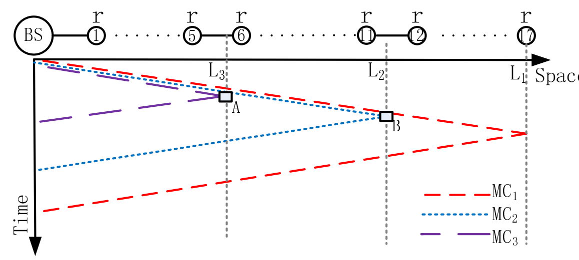
Fig. 2(d) depicts the time-space view of applying *PushWait* to the aforementioned settings. Three chargers start from the BS with $80J$ energy; when they reach L_3, both MC_1 and MC_2 have $80-3\cdot L_3 = 70J$ energy, while MC_3 has $80-3\cdot L_3-3\cdot b = 64J$ energy because it recharges s_1, s_2, and s_3; then, MC_3 charges both MC_1 and MC_2 to their full battery. After this, MC_3 waits at L_3 with $64-10-10 = 44J$ energy; similarly, after MC_2 charges sensors from s_4 to s_9, and charges MC_1 to its full battery at L_2, it waits at L_2 with $34J$ energy; when MC_1 returns to L_2, as the reader can verify, it will have zero energy; MC_2 then charges MC_1 to half of its residual energy, i.e., $17J$, which is just enough for MC_1 and MC_2 to reach



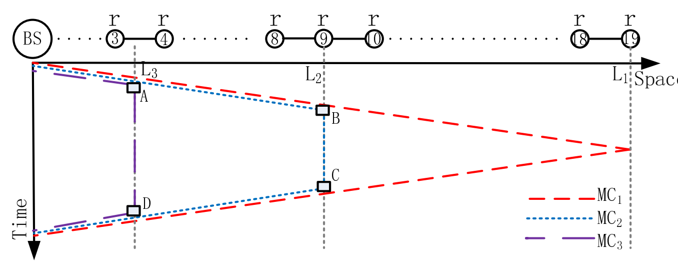
(a) Scheme I without collaboration, $L_1 = L_2 = L_3 = 12$



(b) Scheme II without collaboration, $L_1 = 13, L_2 = 12, L_3 = 10$



(c) Scheme III with collaboration, $L_1 = 17, L_2 = 11\frac{4}{9}, L_3 = 5\frac{5}{6}$



(d) *PushWait*, $L_1 = 19, L_2 = 9, L_3 = 3\frac{1}{3}$

Fig. 2: Time-space view of four different schemes where $B = 80J, b = 2J$ and $c = 3J/m$. (J=Joule)

L_3; at L_3, MC_3 then charges MC_1 and MC_2 with $10J$ energy, which is just enough for them return to the BS; at the BS, only MC_3 has $14J$ residual energy. We see that *PushWait* achieves the best result: 19 sensors can be covered.

C. *Rendezvous Points*

To make *PushWait* work, L_i ($1 \leq i \leq K$) should be chosen carefully to guarantee that MC_1, MC_2,..., and MC_{i-1} have zero energy when they return to L_i. Let's take the interval between L_{i+1} and L_i as an example to illustrate how to determine the values of these K rendezvous points.

MC_i gets fully charged at L_{i+1} and comes back to L_{i+1} with zero energy. The energy consumption of the full battery B includes the following five parts: (i) energy transferred to the sensors between L_{i+1} and L_i; (ii) energy consumed by MC_i to travel from L_{i+1} to L_i; (iii) energy transferred to MC_1, MC_2,

..., and MC_{i-1} at L_i for the first time. Note that these $i-1$ MCs are fully charged at L_{i+1}, thus the energy transferred to them at L_i is exactly the energy consumed by them to travel from L_{i+1} to L_i; (iv) energy consumed by MC_i to travel from L_i to L_{i+1}; and (v) energy transferred to MC_1, MC_2, ..., and MC_{i-1} at L_i for the second time, which is exactly the energy used by them to travel from L_i to L_{i+1}.

Therefore, we have the following equations.

$$\begin{cases} 2 \cdot c \cdot (L_1 - L_2) \cdot 1 + b \cdot (L_1 - L_2) = B \\ 2 \cdot c \cdot (L_{i-1} - L_i) \cdot (i-1) + b \cdot (L_{i-1} - L_i) = B \quad (2 \le i \le K) \\ 2 \cdot c \cdot (L_K - 0) \cdot K + b \cdot (L_K - 0) \le B \end{cases} \tag{2}$$

The last formula is an inequality, since *PushWait* cannot use up the total amount of energy of K MCs precisely. It is straightforward to see that:

$$\begin{cases} L_1 = N \\ L_i = N - \sum_{j=1}^{i-1} \frac{B}{2 \cdot c \cdot d \cdot j + b} \quad (2 \le i \le K) \end{cases} \tag{3}$$

K can be determined by: $L_K > 0$, $L_{K+1} \le 0$. Then, we have:

$$\begin{cases} E^{payload} = N \cdot b \\ E^{overhead} = 2c \cdot \sum_{i=1}^{K} L_i \end{cases}$$

We note in passing that, as MC_K may have some residual energy when it returns to the BS, we can further improve *PushWait* through the following trick. Let another MC' stay at L_K to collect the residual energy that MC_K would take back to the BS. In doing so, after enough scheduling cycles, only $(K-1)$ MCs are required to start from the BS in the subsequent scheduling cycle.

Fig. 2(d) shows an example, MC_3 has $14J$ residual energy when it comes back to the BS; another MC' can be used to collect 14J energy at L_3; after five scheduling cycles, MC' will have 70J energy. As the reader can verify, in the sixth scheduling cycle, only two MCs are needed.

D. Optimality

***Theorem** 1:* For the uniform case of the *CMCS* problem, *PushWait* achieves the maximum ratio of $E^{payload}$ to $E^{overhead}$.

Proof: Given a 1-D WSN where sensors consume energy at the same rate, $E^{payload}$ is fixed in a recharging cycle. Hence, it is sufficient to prove that *PushWait* uses the minimum $E^{overhead}$, which is proportional to the distance traveled by all of the MCs. Denote the distance traveled by all of the MCs in a scheduling scheme as *Distance(scheme)*. Suppose that *PushWait* requires K MCs to charge the given WSN. We prove the theorem by induction on K.

Base cases: K=1 and K=2

$K = 1$: this case is trivial. *Distance(PushWait)* $= 2 \cdot L_1$. Note that L_1 equals the length of the given WSN. Any scheduling scheme must have at least one MC to reach the farthest sensor in the WSN and then turn back, thus *Distance(anyscheme)* $\ge 2 \cdot L_1 =$ *Distance(PushWait)*.

$K = 2$: (by contradiction) suppose that *PushWait* is not optimal, and the optimal scheduling scheme is OPT. As one MC is not enough to cover the entire WSN, there are at least two MCs in the OPT. One of them, say MC', must reach the farthest sensor, thus it travels $2 \cdot L_1$ distance. Since OPT is the optimal scheduling scheme, i.e., $Distance(OPT) <$ *Distance(PushWait)* $= 2 \cdot L_1 + 2 \cdot L_2$. Hence, all of the other MCs in OPT should not reach L_2; otherwise, OPT is not optimal. However, according to our calculation of L_2 in *PushWait*, a fully charged MC at L_2 only charges the sensors from L_2 to L_1 and returns to L_2 with zero energy, then we know MC' in OPT can by no means reach L_1. A contradiction! Therefore, no such OPT exists. *PushWait* is optimal.

I.H.: PushWait is optimal when $K = n$.

$K = n + 1$: (by contradiction) suppose that *PushWait* is not optimal, and there are $n+1$ rendezvous points L_{n+1}, L_n, ...,L_1 in *PushWait*. The optimal scheduling scheme is OPT.

We can divide the WSN into two parts, the BS-to-L_{n+1} part, and the L_{n+1}-to-L_1 part. Suppose that a virtual base station BS' is located at L_{n+1}. *PushWait* needs precisely $n \cdot B$ energy to cover the sensors between L_{n+1} and L_1. By I.H., OPT will require more than $n \cdot B$ energy to cover the same part. Denote this energy as $Q > n \cdot B$.

Therefore, the task of *PushWait* is to cover the sensors from BS to L_{n+1} and to deliver $n \cdot B$ energy to L_{n+1}. According to *PushWait*, $n+1$ MCs that start from the BS can accomplish this task. Correspondingly, the task of OPT is to cover the sensors from BS to L_{n+1} and to deliver Q energy to L_{n+1}. We know that OPT requires at least $n+1$ MCs to reach L_{n+1} (otherwise, the total residual energy of less than $n+1$ MCs at L_{n+1} is definitely less than $n \cdot B$).

Considering $Q > n \cdot B$, *PushWait* is optimal. ■

Remarks: This theorem still holds in some some general settings, for example, the distance between adjacent sensors is non-uniform, the sensor battery capacity is non-uniform, the recharging cycle is smaller than MC round-trip time, and so on. The corresponding proofs follow a similar routine as the above proof and are left to the reader.

E. Coverage

***Theorem** 2:* Given infinite MCs, the maximum numbers of sensors that can be covered by scheme I, II, III, and *PushWait* are $< B/2c$, $< B/2c$, $< B/c$, and infinite, respectively.

Proof: Scheme I: Each MC needs to contribute b/K amount of energy to each sensor. When K approaches infinity, the share b/K approaches 0. However, every MC still needs to return to the BS, thus the maximum number of sensors that can be covered by this scheme is less than $B/2c$.

Scheme II: When i increases, MC_i needs to travel a longer distance to reach the sensors that it should cover. Also, every MC needs to return to the BS, so the maximum number of sensors that can be covered by this scheme is less than $B/2c$.

Scheme III: When an MC begins to turn back, it can no longer get energy from others. Thus, the maximum number of sensors that can be covered is less than B/c.

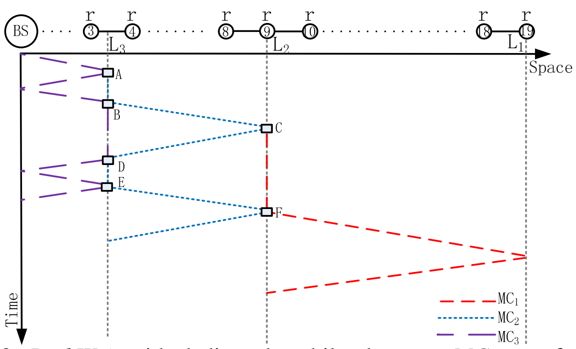


Fig. 3: *PushWait* with dedicated mobile chargers. MC_3 transfers 54 J energy to MC_2 at A; MC_3 transfers 26 J energy to MC_2 at B; MC_2 transfers 34 J energy to MC_1 at C; MC_3 transfers 34 J energy to MC_2 at D; MC_3 transfers 46 J energy to MC_2 at E; MC_2 transfers 46 J energy to MC_1 at F.

PushWait: According to Equ. (3), we have:

$$L_i - L_{i+1} = \frac{B}{2 \cdot c \cdot i + b}, \forall i \geq 1 \tag{4}$$

Then the distance covered by K MCs is:

$$\sum_{i=1}^{K} \frac{B}{2 \cdot c \cdot i + b} > \sum_{i=i_0}^{K} \frac{B}{2 \cdot c \cdot i + b} (\text{let } 2 \cdot c \cdot i_0 \geq b)$$
$$> \sum_{i=i_0}^{K} \frac{B}{4 \cdot c \cdot i} = \frac{B}{4 \cdot c} \sum_{i=i_0}^{K} \frac{1}{i} (\text{harmonic series})$$

which approaches infinity as K approaches infinity. ■

F. A Variation of PushWait: Dedicated Chargers

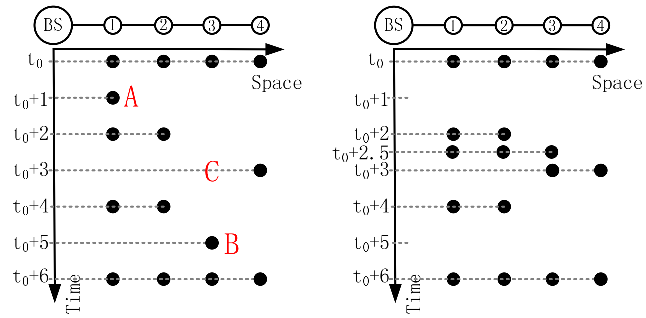
In *PushWait*, all of the MCs start from the BS and return to the BS in a scheduling cycle. It looks complex in terms of oprations. In fact, a clean and simple variation of *PushWait* is to have MC_i travel between L_{i+1} and L_i without going back to the BS, which is illustrated in Fig. 3. The parameters are the same as in Fig. 2. MC_3 starts from the BS, charges sensors s_1, s_2, and s_3, arrives at L_3, where it can transfer 54 J energy to MC_2 (point A). As MC_2 is not fully charged, it waits at point A for the next arrival of MC_3, and so on. It is easy to see that *PushWait* with dedicated mobile chargers achieves the same energy efficiency ratio, i.e., $E^{payload}/E^{overhead}$.

The drawback of this variation is that it finishes the charging of the entire network with a longer time compared to *PushWait*, as MC_i needs to wait for MC_{i+1} to deliver energy to it. For example, MC_2 needs to wait from A to B and then from D to E, MC_1 needs to wait from C to F.

The advantages of this variation are twofold. Firstly, this variation simplifies the scheduling by just letting a dedicated MC be responsible for an area of sensors. Secondly, when there are many types of MCs with different capacities, it is easy to apply *PushWait* to this situation by just expanding each dedicated area.

IV. *CMCS* with non-uniform energy consumption rates

In this section, we employ the results from the last section to develop a scheduling algorithm for the non-uniform case of the *CMCS* problem. We first give two examples to help readers understand the problem and see the need to design a heuristic scheduling algorithm to cope with the NP-hardness of this problem. Then, we present two observations that can remove some impossible scheduling choices, after which we present our heuristic, *ClusterCharging*(β), and its approximation ratio.



(a) Scheme IV (b) Scheme V

Fig. 4: Time-space view of two scheduling schemes. In the figure, $\tau_1 = 4, \tau_2 = 2, \tau_3 = 5, \tau_4 = 3$. The black point "•" indicates the time when the corresponding sensor is recharged.

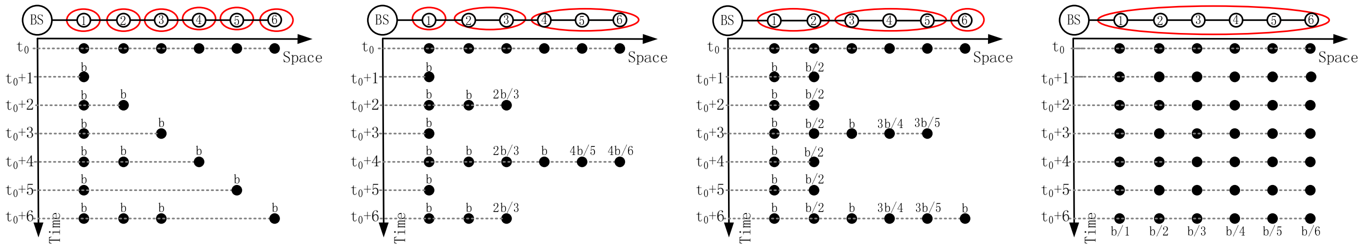
A. Examples of Scheduling Schemes

Denote $\tau_i = b/r_i$ as the recharging cycle of sensor s_i. To avoid the messy details and focus on the main problem, we assume that τ_i is an integer (in fact, τ_i is typically large enough for us to let $\tau_i = \lfloor \tau_i \rfloor$). Fig. 4 shows two feasible scheduling schemes. At time t_0, four sensors are full of energy; as $\tau_1 = 4$, s_1 should be recharged no later than time $t_0 + 4$, otherwise it will die; similar statements hold for other sensors. Recall our definition of scheduling cycle in Section II-C; in these examples, the consecutive time points when all sensors get fully charged are t_0 and $t_0 + 6$. Therefore, the scheduling cycle of both of Scheme IV and V is 6, and there are 6 rounds of charging in a scheduling cycle in both of them.

We notice from these examples that the number of possible scheduling solutions could be extremely large, because any charging round in a solution has exponential choices of a set of sensors to recharge. To find the optimal scheduling for a given WSN, we must determine both the length of the scheduling cycle and the set of sensors to be recharged in each round. With all that said, we conjecture that the non-uniform case of the *CMCS* problem is NP-hard. In the following subsections, we will present a heuristic with an approximation ratio after introducing two observations.

B. Observation from Space Aspect

In Fig. 4(a), $\tau_1 = 4 > \tau_2 = 2$, whenever we recharge s_2, we can recharge s_1 incidentally. Considering that the objective is to maximize the ratio of payload energy to overhead energy, we see that there is no need to recharge s_1 individually, i.e., the recharging at point A in Fig. 4(a) is not cost-efficient. In doing so, we can take τ_1 as 2. For the same reason, we can take τ_3 as 3, and recharge s_3 at time $t_0 + 3$ (as point C shows) instead of $t_0 + 5$ (as point B shows). This observation enables us to only consider the following setting in the rest of this paper: $\tau_1 \leq \tau_2 \leq \cdots \leq \tau_N$.


(a) $\beta = 1$, there are six groups; each sensor itself forms a group.

(b) $\beta = 2$, there are three groups: (s_1), (s_2,s_3), and (s_4,s_5,s_6).

(c) $\beta = 3$, there are three groups: (s_1, s_2), (s_3,s_4,s_5), and (s_6).

(d) $\beta = \infty$, there is only one group that contains all sensors.

Fig. 5: Time-space view of *ClusterCharing*(β) where $\tau_i = i$. The number near a black point is the amount of energy that the corresponding sensor should be charged in that charging round.

C. Observation from Time Aspect

The following theorem indicates that we only need to start a charging round when there is at least one sensor node that will die if we do not. For example, in Fig. 4(b), we only plan possible rounds at $t = t_0 + i$, where i is an integer; we do not need to start the round at $t_0 + 2.5$ because, at that time, there is no sensor node that will die if it is not charged.

***Theorem** 3:* Given a base station BS and a sensor node s, with a battery capacity of b, that are d distance apart, then it is better to deliver b amount of energy to s using *PushWait* one time than twice (the total energy s gets is still b).

Proof: Note that d is not restricted to being small, thus one MC may not be enough. Suppose that k MCs are needed to deliver b amount of energy to s using *PushWait* one time; according to Equ. (4), k should satisfy:

$$\frac{B}{2kc} + \frac{B}{2(k-1)c} + \cdots + \frac{B-b}{2c} = \frac{B}{2c}\sum_{i=1}^{i=k}\frac{1}{i} - \frac{b}{2c} = d$$

Equivalently:

$$\sum_{i=1}^{k}\frac{1}{i} = \frac{2cd+b}{B} \tag{5}$$

Similarly, if we use *PushWait* twice, suppose that k_1 MCs are needed to deliver ϵ amount of energy to s for the first time, and k_2 MCs are needed to deliver $b - \epsilon$ amount of energy to s for the second time, then k_1 and k_2 should satisfy:

$$\frac{B}{2k_1c} + \frac{B}{2(k_1-1)c} + \cdots + \frac{B-\epsilon}{2c} = \frac{B}{2c}\sum_{i=1}^{i=k_1}\frac{1}{i} - \frac{\epsilon}{2c} = d$$

$$\frac{B}{2k_2c} + \frac{B}{2(k_2-1)c} + \cdots + \frac{B-(b-\epsilon)}{2c} = \frac{B}{2c}\sum_{i=1}^{i=k_2}\frac{1}{i} - \frac{b-\epsilon}{2c} = d$$

Equivalently:

$$\sum_{i=1}^{k_1}\frac{1}{i} = \frac{2cd+\epsilon}{B}, \quad \sum_{i=1}^{k_2}\frac{1}{i} = \frac{2cd+b-\epsilon}{B} \tag{6}$$

Then it is sufficient to prove that $k_1 + k_2 < k$ cannot be true, subject to Eqs. (5) and (6). The harmonic series [21] can be represented as:

$$\sum_{i=1}^{k}\frac{1}{i} \approx lnk + \frac{1}{2k} + \gamma \tag{7}$$

where γ is the Euler-Mascheroni constant. As we know, if k_1+k_2 is fixed, $\sum_{i=1}^{k_1} 1/i + \sum_{i=1}^{k_2} 1/i$ achieves its maximum when $k_1 = k_2$. Therefore, we let $k_1 = k_2 = k/2$ to see what condition k should satisfy to ensure Eqs. (5) and (6). By combining Eqs. (5), (6), and (7), we have:

$$k = \frac{B}{2ln2 \cdot B - b} < \frac{B}{2 \cdot B - B} = 1 \tag{8}$$

which is impossible. Alas, we prove that $k_1 + k_2 > k$, indicating that using *PushWait* one time is more cost-efficient. ■

It is worth mentioning that the above proof is based on the following assumption: the total amount of energy of k MCs, k_1 MCs, or k_2 MCs is completely used up. The worst case is when the energy of k MCs is completely used up while there are two MCs among k_1 and k_2 MCs whose energy is nearly unused. Then, the two times of *PushWait* costs the total energy of $k_1 - 1 + k_2 - 1 = k_1 + k_2 - 2$ MCs. As $k_1 + k_2 \geq k + 1$, the only bad situation is $k_1 + k_2 = k + 1$, which is rare compared to all possible cases.

D. ClusterCharging(β)

1) Basic idea: The basic idea of *ClusterCharging*(β) is to cluster sensors into groups such that the ratio of the maximum recharging cycle to the minimum recharging cycle in each group is less than the clustering threshold β. In each charging round, our heuristic selects the groups that satisfy the following condition as the charging targets, and employs *PushWait*[2] to recharge the sensors in these groups. The condition for a group to be selected is that there is at least one sensor in this group that is going to die if our heuristic does not recharge it.

Take Fig. 5 for example. In Fig. 5(a), $\beta = 1$, thus each sensor itself forms a group; each group (or sensor) gets recharged only before running out of energy.

In Fig. 5(b), $\beta = 2$. Since $\tau_1 = 1$ and $\tau_2 = 2$, $\tau_2/\tau_1 \geq 2 = \beta$, then s_1 itself forms a group. Also $\tau_3/\tau_2 < \beta$ and $\tau_4/\tau_2 \geq \beta$, thus sensors s_2 and s_3 form a group, and so forth. In summary, when $\beta = 2$, there are three groups, (s_1), (s_2,s_3), and (s_4,s_5,s_6). At time $t_0 + 1$, only the first group is recharged; at time $t_0 + 2$,

[2]Note that, in each round of this scenario, different sensors may need to be charged a different amount of energy. For example, at time $t_0 + 4$ in Fig. 5(b), the six sensors need b, b, $2b/3$, b, $4b/5$, and $4b/6$ amount of energy, respectively. *PushWait* still achieves the optimality in each round according to the remarks in Section III-D.

as s_2 is going to die if it is not recharged, the second group together with the first group are recharged; at time $t_0 + 3$, all groups are selected as the charging targets.

Similarly, when $\beta = 3$ in Fig. 5(c), there are three groups, (s_1, s_2), (s_3,s_4,s_5), and (s_6). When $\beta = \infty$ in Fig. 5(d), there is only one group that contains all sensors.

2) Scheduling Cycle: $\beta = 1$. *ClusterCharing*(1) lazily charges each sensor just before it runs out of energy. The scheduling cycle is the least common multiple of τ_1,...,τ_N. Denote it as *lcm*. Then, the $E^{payload}$ in a scheduling cycle is:

$$E^{payload} = \frac{lcm}{\tau_1}b + \frac{lcm}{\tau_2}b + \cdots + \frac{lcm}{\tau_N}b = \sum_{i=1}^{N}(r_i \cdot lcm)$$

$\beta = 2, 3, ..., n$. Suppose that there are x groups; since the ratio of the maximum recharging cycle to the minimum recharging cycle in each group is less than β, we have $\tau_1 \cdot \beta^x \leq \tau_N$, then we know $x = \lfloor log_\beta(\tau_N/\tau_1)\rfloor$. Thus, the scheduling cycle of *ClusterCharging*(β) is:

$$\beta^{\lfloor log_\beta(\frac{\tau_N}{\tau_1})\rfloor} \cdot \tau_1$$

For example, the scheduling cycles in Figs. 5(b) and 5(c) are 4 and 6, respectively. Correspondingly, we can calculate $E^{payload}$ and $E^{overhead}$ in a scheduling cycle using *PushWait*.

$\beta = \infty$. *ClusterCharging*(∞) charges all sensors to their full battery capacity every τ_1 time, i.e., the minimum recharging cycle among all sensor nodes. Obviously, the scheduling cycle is τ_1. Then, the $E^{payload}$ in a scheduling cycle is:

$$E^{payload} = \frac{\tau_1}{\tau_1}b + \frac{\tau_1}{\tau_2}b + \cdots + \frac{\tau_1}{\tau_N}b = \sum_{i=1}^{N}\frac{r_i}{r_1}b$$

Different values of β lead to different performances. In our simulations, we will show that when the parameters of the problem instance change, the optimal β also changes.

3) Approximation Ratio of ClusterCharging(β): Denote by *ratio*(S) the ratio of $E^{payload}$ to $E^{overhead}$ in a scheduling scheme S. Denote by *OPT* the optimal scheduling scheme for the non-uniform case of the *CMCS* problem.

***Theorem** 4:*

$$\frac{ratio(ClusterCharging(\beta))}{ratio(OPT)} > \frac{2c}{k_{min}B\tau_N - b}$$

where,

$$k_{min} = argmin_{(\sum_{i=1}^{k}\frac{1}{i} \geq \frac{2c\tau_N + b}{B\tau_N})}k$$

Proof: The main line of this proof is to construct a scheme S_1 so that $ratio(S_1) < ratio(ClusterCharing(\beta))$, and to construct another scheme S_2 so that $ratio(OPT) < ratio(S_2)$. We then have $ratio(ClusterCharing(\beta))/ratio(OPT) > ratio(S_1)/ratio(S_2)$.

Constructing S_1. Consider the following charging round: we use *MC*s to charge only one sensor, which is at the farthest point of the WSN and only needs the least possible energy amount, i.e., b/τ_N. In this round, $E^{payload} = b/\tau_N$ and $E^{overhead}$ can be obtained as follows. The number of *MC*s used in this round is: $k_{min} = argmin_k(\sum_{i=1}^{k} 1/i \geq (2c\tau_N + b)/B\tau_N)$ (similar to Equ. (5)). Therefore, $E^{overhead} = k_{min} \cdot B - b/\tau_N$. As this round is the worst round we can imagine, we have $ratio(S_1) > E^{payload}/E^{overhead} = b/(k_{min}B\tau_N - b)$.

Constructing S_2. Suppose that an *MC* has an infinite amount of energy, then we only need one *MC* in each round. Obviously, $ratio(OPT) < ratio(S_2)$. Remember that whenever a sensor needs to be recharged, this *MC* must travel at least one unit distance to recharge it and then travel at least another one unit distance to come back to its original point. Therefore, $ratio(S_2) < b/2c$.

The theorem follows immediately. ■

V. Performance Evaluation

In this section, we primarily focus on evaluating *ClusterCharging*(β) in different settings with respect to various parameters, and will not evaluate *PushWait* since *PushWait* provides the optimal solution for the uniform case of the *CMCS* problem. We first introduce the evaluation settings, then present the results.

A. Evaluation Setup

In order to see the impact of the recharging cycles, τ_1, τ_2,..., and τ_N, on the performance of *ClusterCharging*(β), we use two different settings to generate these cycles.

Random-Setting: The recharging cycles are randomly generated from a bounded range, i.e., $[lbound, ubound] = [2, 8]$. We then sort them to guarantee that $\tau_1 \leq \tau_2 \leq \cdots \leq \tau_N$. For evaluations based on this setting, we ran experiments 100 times and averaged the results.

Power-Setting: The recharging cycles are generated based on a power function, i.e., $\tau_i = base^{\lfloor (i+1)/2\rfloor} = 2^{\lfloor (i+1)/base\rfloor}$.

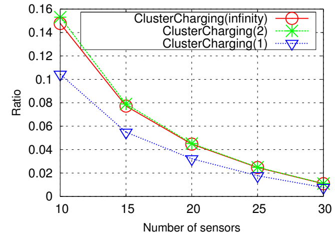
These two settings reflect two extremes of the mathematical variances of recharging cycles: *Random-Setting* generates cycles with a small variance, while *Power-Setting* generates cycles with a relatively large variance. Therefore, we can observe the impacts of the non-uniform recharging cycles on our proposed heuristic more clearly.

In each setting, we try to evaluate the effects of the number of sensors, N, the energy cost per unit distance, c, the battery capacity of a sensor node, b, and the battery capacity of a mobile charger, B, separately. We are also interested in the impacts of the bounded range and the power function in each setting. Hence, we ran experiments with the *ubound* varying from 4 to 12 while keeping $lbound = 2$, and we ran experiments with *base* varying from 2 to 6.

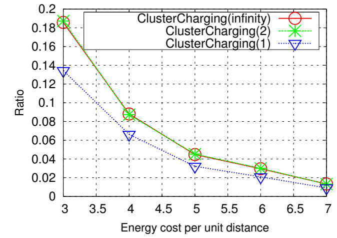
The optimal solution to the non-uniform case of the *CMCS* problem requires exhaustive searching, which is infeasible even when the number of sensors is a little large. Considering that the approximation ratio of *ClusterCharging*(β) is given, we do not implement the optimal solution for comparison.

B. Evaluation Results

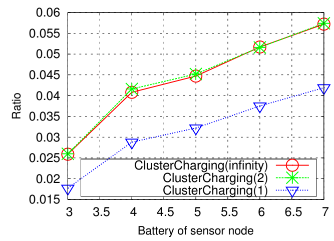
1) Random-Setting: Fig. 6 shows the results of different setups for the *Random-Setting*. In general, *ClusterCharging*(∞) (red line with circle markers) achieves almost the same performance as *ClusterCharging*(2) (green line with cross



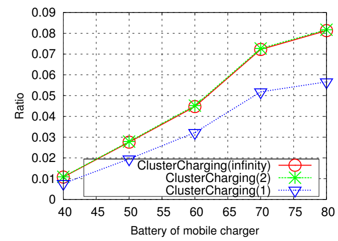
(a) Varying N. ($c = 5, b = 5, B = 60, lbound = 2, ubound = 8$)



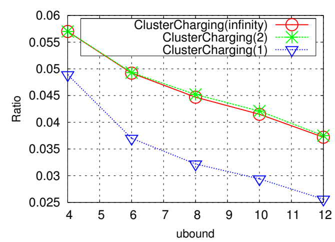
(b) Varying c. ($N = 20, b = 5, B = 60, lbound = 2, ubound = 8$)



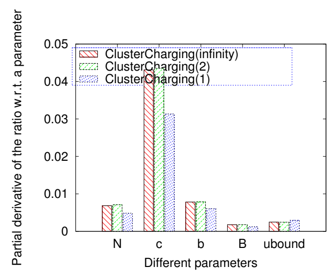
(c) Varying b. ($N = 20, c = 5, B = 60, lbound = 2, ubound = 8$)



(d) Varying B. ($N = 20, c = 5, b = 5, lbound = 2, ubound = 8$)

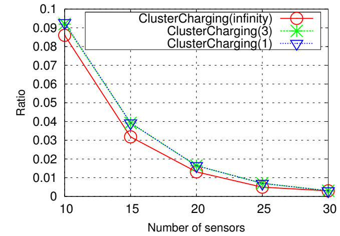


(e) Varying $ubound$.($N = 20, c = 5, b = 5, B = 60, lbound = 2$)

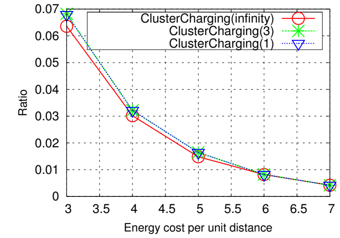


(f) Partial derivative of the ratio with respect to each parameter

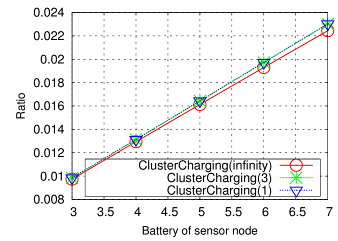
Fig. 6: Impact of various parameters in *Random-Setting*



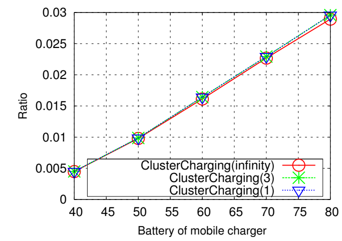
(a) Varying N. ($c = 5, b = 5, B = 60, base = 4$)



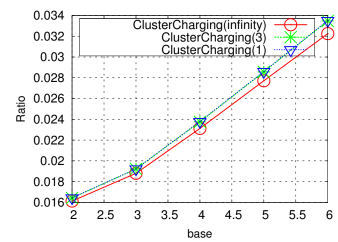
(b) Varying c. ($N = 20, b = 5, B = 60, base = 4$)



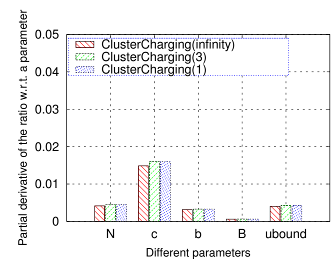
(c) Varying b. ($N = 20, c = 5, B = 60, base = 4$)



(d) Varying B. ($N = 20, c = 5, b = 5, base = 4$)



(e) Varying $base$. ($N = 20, c = 5, b = 5, B = 60$)



(f) Partial derivative of the ratio with respect to each parameter

Fig. 7: Impact of various parameters in *Power-Setting*

markers), and they outperform *ClusterCharging*(1) (blue line with square markers). In detail, when the other parameters are fixed, the performance metric, i.e., the ratio of $E^{payload}$ to $E^{overhead}$, goes down as the number of sensors increases (Fig. 6(a)), goes down as the energy cost per unit distance increases (Fig. 6(b)), goes up as the battery capacity of a sensor node increases (Fig. 6(c)), and goes up as the battery capacity of the *MC* increases (Fig. 6(d)).

In Fig. 6(e), when the *ubound* increases, the ratio increases. This is because a larger range incurs a larger variance of the recharging cycles, which leads to a longer scheduling cycle, and only a few sensors need to be charged in each round. This sparsity causes a performance reduction in each round.

Fig. 6(f) shows the partial derivative of the ratio with respect to each parameter in this setting. For example, when N increases by 1, the ratio of *ClusterCharging*(∞) decreases by 0.00684. We notice that the impact of c is the greatest; B and *ubound* have the least impacts on the ratio. This is reasonable, as the change of c influences every moving segment between any pair of adjacent sensor nodes.

2) Power-Setting: Fig. 7 shows the results of different setups for the *Power-Setting*. In general, *ClusterCharging*(1) (blue line with square markers) achieves almost the same performance as *ClusterCharging*(3) (green line with cross markers), while *ClusterCharging*(∞) (red line with circle markers) has the worst performance. Most of the observations from *Random-Setting* still hold in Figs. 7(a) to 7(d).

In Fig. 7(e), when the *base* increases, the ratio decreases. The main reason is that a larger *base* makes the length of the consecutive recharging cycles of the same value become longer, which further leads to a smaller variance. For example, if $base = 2$, the generated sequence is $\{1, 2, 2, 4, 4, 8, ...\}$; if $base = 4$, the sequence is $\{1, 1, 1, 4, 4, 4, 4, 16, ...\}$. Fig. 7(f) illustrates the partial derivative of the ratio with respect to each parameter in this setting. Like the partial derivatives in *Power-Setting*, c has the greatest impact on the ratio; B have the least impact.

In summary, our simulations show that the proposed algorithms perform well in a variety of settings. Specifically, when the variance of the recharging cycles becomes larger, *ClusterCharging*(β) performs worse; *ClusterCharging*(1) and *ClusterCharging*(∞) are sensible to the variance of the recharging cycles, while *ClusterCharging*(β) with other value of β is robust in both settings.

VI. Related Work

Energy conservation, harvesting and node reclamation. Energy conservation was proposed to slow down the energy consumption rate. Bhattacharya *et al.* [5] proposed to cache mutable data at some locations to control data retrieval rate. Dunkels *et al.* [6] incorporated cross-layer information-sharing in the their proposed adaptive communication architecture. Wang *et al.* [7] proposed to use resource rich mobile nodes as sinks or relays to prolong the lifetime of WSNs. However, conservation cannot compensate for energy depletion in the end. Energy harvesting [8, 9] tries to harvest energy (such as solar, wind, and vibration) directly from the environment to replenish sensors, but it is neither controllable nor predictable, which hinders WSNs from providing the desired level of performance. Sensor reclamation [10] periodically replaces sensors of no or low energy with fully charged ones; however, it requires either human intervention or advanced robotic mechanisms, which can be costly in various situations.

Wireless energy transfer. The wireless power consortium [22] defines the inter-operability standards of wireless energy transfer based on magnetic induction. Kurs *et al.* [11] demonstrated it to be efficient and non-radiative in *Science*. Peng *et al.* [14] proposed the use of a mobile charger with sufficient energy to charge the entire network and formulated it as a TSP-like (Traveling salesman problem [23]) problem. Li *et al.* [15] took both mobile charger scheduling and touring into consideration with the assumption that the movement of a mobile charger costs zero energy. Tong *et al.* [17] found that: when the number of sensor nodes being charged simultaneously increases, the average power received at each sensor remains approximately the same. Using this observation, the authors tried to determine the optimal node deployment and routing strategies to improve energy efficiency. Shi *et al.* [16] also assumed that the mobile charger has unbounded energy and investigated the problem of periodically charging sensors to maximize the free time of the mobile charger over a cycle. In contrast to these works, we investigate the problem with a more realistic condition: a mobile charger may not have enough energy to cover the entire network.

VII. Conclusions and Future Work

This paper introduces a novel charging paradigm, i.e., collaborative mobile charging. We investigate the collaborative mobile charging scheduling problem in 1-D WSNs. We first consider the uniform case and propose an algorithm, *PushWait*, which is proven to be optimal in this case and can cover a 1-D WSN of any length. A variation of *PushWait* that uses dedicated chargers to substitute roundtrip chargers is also presented. We then develop a heuristic, *ClusterCharging*(β), with guaranteed performance for the non-uniform case. Extensive simulations validate the advantages of our algorithms.

Our future work will focus on two parts. One part involves investigating the impact of wireless transfer efficiency, which is assumed to be one in this paper. The other part involves extending our algorithms to 2-D networks.

Acknowledgments

This work is supported in part by the National NSF of China under Grant No. 61073028 and No. 61021062; Key Project of Jiangsu Research Program under Grant No.BE2010179; Jiangsu Natural Science Foundation under Grant No. BK2011510; the National 973 Basic Research Program of China under Grant No. 2009CB320705 and No. 2011CB302800; College graduate research and innovation project of Jiangsu under Grant No. CXZZ12_0055; US NSF grants ECCS 1128209, CNS 1065444, CCF 1028167, CNS 0948184, and CCF 0830289.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks, 2002*.

[2] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Health monitoring of civil infrastructures using wireless sensor networks," in *ACM/IEEE IPSN 2007*.

[3] J. Panchard, S. Rao, M. S. Sheshshayee, P. Papadimitratos, S. Kumar, and J.-P. Hubaux, "Wireless sensor networking for rain-fed farming decision support," in *ACM SIGCOMM Workshop on NSDR 2008*.

[4] M. Hefeeda and M. Bagheri, "Wireless sensor networks for early detection of forest fires," in *IEEE MASS 2007*.

[5] S. Bhattacharya, H. Kim, S. Prabh, and T. Abdelzaher, "Energy-conserving data placement and asynchronous multicast in wireless sensor networks," in *ACM Mobisys 2003*.

[6] A. Dunkels, F. Österlind, and Z. He, "An adaptive communication architecture for wireless sensor networks," in *ACM Sensys 2007*.

[7] W. Wang, V. Srinivasan, and K.-C. Chua, "Using mobile relays to prolong the lifetime of wireless sensor networks," in *ACM MobiCom 2005*.

[8] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *ACM Trans. Embed. Comput. Syst.*, vol. 6, Sept. 2007.

[9] X. Jiang, J. Polastre, and D. Culler, "Perpetual environmentally powered sensor networks," in *ACM/IEEE IPSN 2005*.

[10] B. Tong, G. Wang, W. Zhang, and C. Wang, "Node reclamation and replacement for long-lived sensor networks," *IEEE TPDS*, vol. 22, no. 9, pp. 1550 –1563, Sept. 2011.

[11] A. Kurs, A. Karalis, R. Moffatt, J. D. Joannopoulos, P. Fisher, and M. Soljačić, "Wireless power transfer via strongly coupled magnetic resonances," *Science*, vol. 317, no. 5834, pp. 83–86, 2007.

[12] B. Cannon, J. Hoburg, D. Stancil, and S. Goldstein, "Magnetic resonant coupling as a potential means for wireless power transfer to multiple small receivers," *IEEE Trans. on Power Electronics*, vol. 24, no. 7, pp. 1819 –1825, July 2009.

[13] K. Kang, Y. S. Meng, J. Bréger, C. P. Grey, and G. Ceder, "Electrodes with high power and high capacity for rechargeable lithium batteries," *Science*, vol. 311, no. 5763, pp. 977–980, 2006.

[14] Y. Peng, Z. Li, W. Zhang, and D. Qiao, "Prolonging sensor network lifetime through wireless charging," in *IEEE RTSS 2010*.

[15] Z. Li, Y. Peng, W. Zhang, and D. Qiao, "Study of joint routing and wireless charging strategies in sensor networks," in *WASA 2010*.

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

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

[18] I. Jawhar and N. Mohamed, "A hierarchical and topological classification of linear sensor networks," in *WTS 2009*.

[19] I. Jawhar, N. Mohamed, and K. Shuaib, "A framework for pipeline infrastructure monitoring using wireless sensor networks," in *WTS 2007*.

[20] A. D'Costa, V. Ramachandran, and A. M. Sayeed, "Distributed classification of gaussian space-time sources in wireless sensor networks," *IEEE JSAC*, vol. 22, no. 6, pp. 1026–1036, 2004.

[21] "Harmonic Series," *http://en.wikipedia.org/wiki/Harmonic_series*.

[22] "Wireless power consortium," *http://www.wirelesspowerconsortium.com/*.

[23] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. The MIT Press, 2001.