Chapter 19
Collaborative Mobile Charging

Sheng Zhang and Jie Wu

Abstract Wireless power transfer attracts significant attention from both academia and industry. While most previous studies have primarily focused on optimizing charging sequences and/or durations for one or more mobile chargers, we concentrate on collaboration between mobile chargers. In this context, what we mean by "collaboration" is that chargers can exchange energy. We will first show the collaboration benefit in terms of charging coverage and energy usage effectiveness, then we will show how to design collaboration-based scheduling algorithms using a simple yet representative setting. Finally, we will demonstrate several extensions in which previous assumptions are relaxed one by one. We also use simulations to evaluate our theoretical findings and investigate the effects of important design parameters.

19.1 Introduction

Wireless power transfer [1] provides a promising means of replenishing battery-powered devices in ad hoc communication networks, and thus supports various novel applications. Armed with wireless power transfer, many existing studies [2–4] have envisioned employing mobile vehicles, robots, and even helicopters carrying high volume batteries as *mobile chargers* to periodically deliver energy to rechargeable devices in target areas. The optimization goals of these studies include maximizing the lifetime of the underlying network [5], optimizing the efficiency of charging scheduling [3], energy provisioning [6], minimizing total charging delay [2], optimizing the coordination of multiple mobile chargers [7], minimizing maximum radiation point [8], etc.

S. Zhang (✉)
State Key Laboratory for Novel Software Technology,
Nanjing University, 163 Xianlin Avenue, Nanjing 210023, China
e-mail: sheng@nju.edu.cn

J. Wu
Department of Computer and Information Sciences,
Temple University, 1925 N. 12th Street, Philadelphia, PA 19122, USA
e-mail: jiewu@temple.edu

S. Nikoletseas et al. (eds.), *Wireless Power Transfer Algorithms, Technologies and Applications in Ad Hoc Communication Networks*,
DOI 10.1007/978-3-319-46810-5_19

However, we found that most of the existing work has hardly considered the limited energy of mobile chargers, and has usually assumed that a mobile charger has a sufficient amount of energy to not only replenish an entire target network, but also to make a round-trip back to the base station. This model is invalidated when a dedicated charger (even that with a full battery) cannot reach a particularly remote area.

In this chapter, we introduce a novel charging paradigm, i.e., collaborative mobile charging paradigm [9], which allows energy transfer between mobile chargers. By careful selection of the location of and the amount of energy transferred at each rendezvous point, not only is the energy usage effectiveness improved, but the charging coverage is also enlarged.

We will first discuss related work in Sect. 19.2, and show the collaboration benefit in terms of charging coverage and energy usage effectiveness in Sect. 19.3, then we present how to design collaboration-based scheduling algorithms using a simple yet representative setting in Sect. 19.4. Finally, we demonstrate several extensions in which previous assumptions are relaxed one by one in Sect. 19.5, where we also use simulations to evaluate our theoretical findings and investigate the effects of important design parameters. Concluding remarks are given in Sect. 19.6.

Nikola Tesla (July 10, 1856−January 7, 1943) conducted the first experiments in wireless power transfer as early as the 1890s: an incandescent light bulb was successfully powered using a coil receiver that was in resonance with a nearby magnifying transmitter [10]. Recently, Kurs et al. experimentally demonstrated that energy can be efficiently transmitted between magnetically resonant objects without any interconnecting conductors, e.g., powering a 60 W light bulb, which is 2 m away, with approximately 40 % efficiency [1]. This technology has led to the development of several commercial products, e.g., Intel developed the wireless identification and sensing platform (WISP) for battery-free monitoring [11]; 30+ kinds of popular phones are beginning to embrace wireless charging [12]; and even vehicles [13] and unmanned planes [14] are now supporting wireless charging. It is predicted that the wireless charging market will be worth $13.78 billion by 2020 [15].

19.2 Related Work

There are a number of approaches that are useful in extending the sustainability and applicability of battery-power devices, e.g., sensors, RFIDs, and vehicles. These methods can be classified into two broad types: energy harvesting and energy conservation. The former extracts environmental energy for supporting energy-hungry

devices, and the latter resorts to energy-aware mechanisms for conserving energy during normal operations of devices.

For example, in energy harvesting, Kansal et al. [16] incorporated proactive learning on environmental parameters into performance adaption, and also provided performance-aware systematic methods to systematically utilize environmental energy [17]; solar energy harvesting was taken into consideration when making routing decisions [18]; Cammarano et al. [19] developed accurate prediction models for solar and wind energy harvesting.

With regard to energy conservation, Wang et al. [20] proposed using resource-rich mobile nodes as sinks/relays to balance the unbalanced energy usages; Dunkels et al. [21] took cross-layer information-sharing into consideration; Bhattacharya et al. [22] proposed caching mutable data at some locations to control the data retrieval rate, for the purpose of slowing down the energy consumption rate. Note that, a combination of energy conservation and wireless power transfer can further improve the energy usage effectiveness.

Wireless power transfer has been a viable topic in the area of wireless networks and mobile computing in recent years. A practical wireless recharge model is derived in [6] on top of WISP. The performance of multi-device simultaneous charging is investigated in [23]. For stationary chargers, a charger placement framework is proposed in [6] to ensure that each device receives sufficient energy for continuous operation; a joint optimization of charger placement and power allocation is considered in [24]; how to obtain the maximum electromagnetic radiation point in a given plane is studied in [8]; quality of energy provisioning for mobile nodes, given their spatial distribution, is investigated in [25].

For mobile chargers, existing studies have considered various decision variables and objectives. To maximize network lifetime, charging sequence and packet routing are optimized in [5, 26], while charger velocity is optimized in [27]; to maximize the ratio of the charger's vacation time (i.e., time spent at the home service station) over the cycle time, travelling path and stop schedules are optimized in [3, 4]; to maximize energy usage effectiveness, collaboration between mobile chargers is optimized in [28, 29]; to minimize the total charging delay, stop locations and durations are optimized in [2]; NDN-based energy monitoring and reporting protocols are designed in [30] with a special focus on scheduling mobile chargers for multiple concurrent emergencies; to simultaneously minimize charger travel distance and charging delay, synchronized charging sequences based on multiple nested tours are optimized in [31]; given heterogenous charging frequencies of sensors, how to schedule multiple charging rounds to minimize total moving distance of mobile chargers is studied in [32].

19.3 Motivation

The motivation of our design is to illustrate the benefits of collaboration in terms of charging coverage and energy usage effectiveness.

19.3.1 Notations

In order to clearly present the motivational examples, we first introduce a few necessary notations.

We consider N stationary sensor nodes distributed along a straight line. These nodes are uniformly distributed, i.e., a unit distance apart. A base station is located to the left of the wireless sensor network (WSN). The base station also serves as the service station that replenishes mobile chargers. Without loss of generality, we denote the sensor nodes from left to right by s_1, s_2, ..., and s_N. The battery capacity of each sensor node is b. Each node consumes energy for sensing, data reception, and transmission. We represent the average energy consumption rate of each sensor node as r. The recharging cycle of a sensor node is defined as the time period during which a sensor node with a full battery can survive without being charged. Letting τ be the recharging cycle of each sensor node, we have $\tau = b/r$.

We assume that mobile chargers are homogeneous: for every charger, the battery capacity is P, the travelling speed is v, and energy consumed by travelling one unit distance is c. Both travelling and wireless charging share the same battery of a mobile charger. The i-th mobile charger is denoted by C_i. A mobile charger starts from the base station with a full battery; after it finishes its charging task, it must return to the base station to be serviced (e.g., recharging its own battery).

As the wireless power transfer efficiency decreases rapidly when the distance between charger and rechargeable device increases [6], we assume that a charger can transfer energy to a sensor node only when they are very close to each other; in this case, we assume that the wireless power transfer is perfect, i.e., there is no energy loss during energy transfer. We denote by $C_i \xrightarrow{e} C_j$ (resp. $C_i \xrightarrow{e} s_j$) the event that charger C_i transfers e units of energy to charger C_j (resp. sensor s_j).

19.3.2 Collaboration Benefit

Coverage. Figure 19.1 shows an example of scheduling chargers without collaboration, where there are 6 sensor nodes and the distance between two consecutive nodes is 1 m. The battery capacities of each node and each charger are 2 and 40 J (J for Joule), respectively; the travelling cost is 3 J/m. In the figure, two chargers are used: C_1 charges s_1 to s_4 and returns to the base station, and C_2 charges s_5 to s_6 and returns to the base station.

We note that even given an infinite number of chargers, the maximum coverage cannot exceed 6 sensors, since any charger must return to the base station for the next round of scheduling (i.e., $\frac{P}{2c} = \frac{40}{2\times 3} < 7$). However, we will shortly see in Theorem 19.2 that, the coverage of well-designed collaboration-based scheduling can be infinite.

Energy usage effectiveness. The energy consumed in replenishing target networks can be classified into three categories: *payload energy*, which is the energy eventually

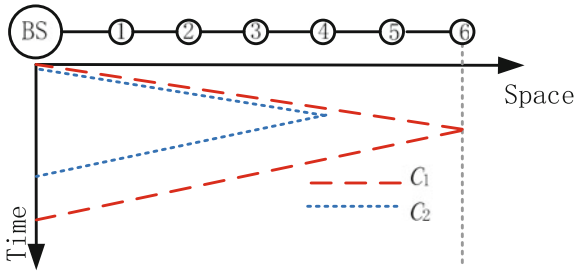


Fig. 19.1 Time-space view of scheduling chargers without collaboration, where $b = 2$ J, $P = 40$ J, and $c = 3$ J/m

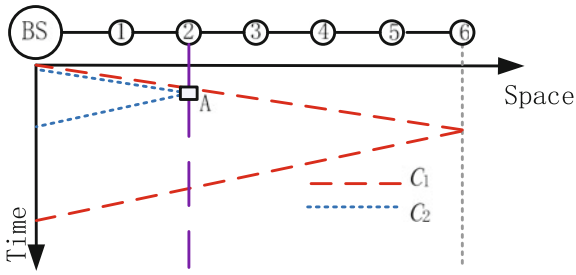


Fig. 19.2 Time-space view of scheduling chargers with collaboration, where $b = 2$ J, $P = 40$ J, and $c = 3$ J/m

obtained by rechargeable devices, *movement energy*, which is the energy used by mobile chargers for moving, and *loss energy*, which is the energy loss during wireless energy transfer. The energy usage effectiveness (EUE) is thus defined as the ratio of the amount of payload energy to the total energy.

In Fig. 19.1, the payload energy is 12, the movement energy is 60, and the loss energy is 0, and hence the EUE is $\frac{1}{6}$. In fact, this is also the highest EUE a scheduling without collaboration can achieve in this example. To delivery energy to s_6, one charger (C_1 in the example) must travel to s_6; due to capacity constraint, C_1 cannot charge more than two nodes (since $40 < 6 \times 2 \times 3 + 2 \times 3 = 42$); therefore, another charger (C_2) must travel to at least s_4.

However, collaboration can improve EUE, as shown in Fig. 19.2. C_2 charges s_1 to s_2, charges C_1 to its full battery at s_2, and returns to the base station. We can similarly calculate that, the EUE of this scheduling is $\frac{1}{5}$.

19.4 Design

In this section, we present the design of collaboration-based scheduling algorithms for a simple yet representative setting. First, we introduce problem formulation in Sect. 19.4.1 and scheduling examples in Sect. 19.4.2, then we present PushWait, an algorithm that achieves the highest EUE in Sect. 19.4.3, and lastly we give some notes in Sect. 19.4.4.

19.4.1 Problem Formulation

The scheduling of chargers is equivalent to deciding the time-space trajectories of chargers. In a feasible scheduling, any rechargeable device should not run out of energy, and every charger should be able to return to the base station. The *scheduling cycle* of a feasible scheduling is the time interval between two consecutive time points when each sensor has the same battery level. To evaluate the long-term energy efficiency of a scheduling, we only have to consider the energy usage in a scheduling cycle. In a scheduling cycle, we denote by E^{pl}, E^{mm}, and E^{ls} the amounts of payload energy, movement energy, and loss energy, respectively. The EUE is thus defined as

$$EUE = \frac{E^{pl}}{E^{pl} + E^{mm} + E^{ls}}. \tag{19.1}$$

Given these settings, our goal is to maximize the EUE.

For simplicity of presentation, we discuss our designs under two assumptions:

- *Short Duration (SD)*: the time for charging a sensor node to its full battery is negligible compared to the recharging cycle;
- *Long Cycle (LC)*: the recharging cycle of a sensor node is longer than a charging round, i.e., any two consecutive charging rounds have no intersections. Thus, mobile chargers can always accomplish a charging round, return to the base station, and wait for another charging round.

Our designs can be applied to settings without these two assumptions, see remarks in Sect. 19.4.4.

19.4.2 Scheduling Examples

We introduce three scheduling examples to motivate the algorithm design.

Suppose we have only 3 mobile chargers with $P = 80$ J, $c = 3$ J/m, and the battery capacity of each sensor node is $b = 2$ J. Since the total energy is fixed, and is 80 J $\times$ 3 = 240 J, EUE is maximized when E^{pl} is maximized. Therefore, in the following, we try to see how to cover the greatest number of sensor nodes using 3 chargers.

Figures 19.3, 19.4, and 19.5 show the time-space views of three simple scheduling heuristics. In the figures, we use L_i ($1 \leq i \leq 3$) to represent the farthest distance that C_i travels away from the base station. We also let L_4 be 0 for compatibility.

- *EqualShare*: each sensor node is jointly charged by all chargers. In our example, each mobile charger transfers $2/3$ unit of energy to each sensor node. Thus, 12 sensors can be covered, as shown in Fig. 19.3.

- *SolelyCharge*: each charger is responsible for a set of consecutive senor nodes, and each sensor node is assigned to one distinct charger. In our example, C_3 charges

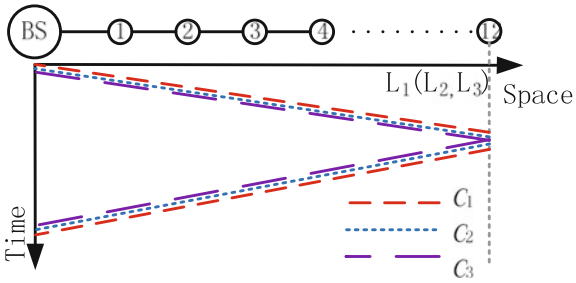


Fig. 19.3 EqualShare, $L_3 = L_2 = L_1 = 12$

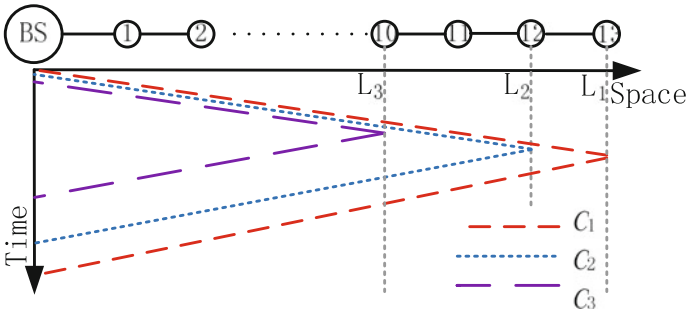


Fig. 19.4 SolelyCharge, $L_3 = 10, L_2 = 12, L_1 = 13$. C_2 has 4 J residual energy

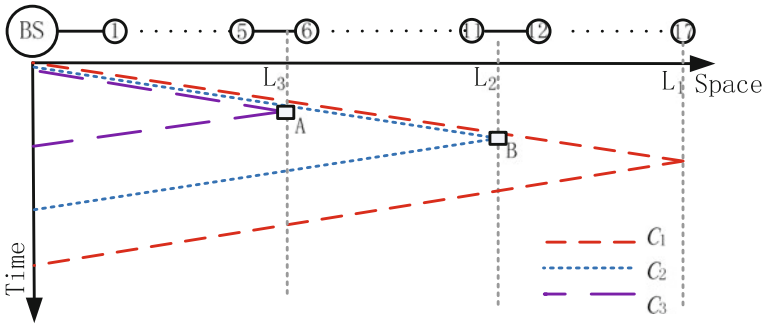


Fig. 19.5 CLCharge, $L_3 = 5\frac{5}{6}, L_2 = 11\frac{4}{9}, L_1 = 17$. C_1 has $\frac{1}{3}J$ residual energy

sensor nodes from L_4 to L_3; C_2 charges sensor nodes from L_3 to L_2; and C_1 charges sensor nodes from L_2 to L_1. The variables L_3, L_2, and L_1 are carefully chosen, so that each charger returns to the base station with exactly zero energy. Figure 19.4 demonstrates that 13 sensors can be covered.

- *CLCharge*: each charger is responsible for a set of consecutive senor nodes, each sensor node is assigned to one distinct charger, and energy transfer between chargers are utilized. In our example, C_3 charges sensors from the base station to L_3, then

transfers some energy to C_2 and C_1 at L_3, and finally returns to the base station. Here, L_3 is carefully chosen, such that (*i*) C_2 and C_1 have full batteries after C_3 transfers energy to them, and (*ii*) C_3 returns to the base station with exactly zero energy. In Fig. 19.5, 17 sensors can be covered.

In these scheduling examples, we make the following key observation that may provide some insight into the design of an optimal scheduling algorithm in the next subsection: when the extent of collaboration increases, the coverage increases. In EqualShare, chargers do not cooperate with each other, and they just charge sensor nodes one by one from the base station; in SolelyCharge, each charger is restricted to replenishing a set of consecutive nodes, and the target network is partitioned into disjoint intervals; finally, in CLCharge, intentional energy transfer between chargers is utilized to further enlarge the coverage.

19.4.3 PushWait

For a given target network, the payload energy is fixed, i.e., E^{pl} is fixed,; since we assume wireless power transfer is perfect, i.e., $E^{ls} = 0$, *we can maximize EUE through minimizing the sum of distances travelled by chargers, i.e., minimizing* E^{mm}.

How might we minimize the sum of the distances travelled by chargers? The key observation is that, *we can let as few mobile chargers as possible carry the residual energy of all chargers, and move forward*. CLCharge in Fig. 19.5 reflects this intuition: C_3 turns around at $L_3 = 5\frac{5}{6}$, which is smaller than 13 in Fig. 19.4. Therefore, the sum of the total travelling distances in CLCharge is less than that in SolelyCharge, leading to a higher EUE. The reason that we can safely let C_3 turn around at $L_3 = 5\frac{5}{6}$ is that C_2 and C_1 can carry the residual energy and move forward, instead of having all of the three not-full-battery mobile chargers move forward.

Can we improve CLCharge? We notice that, in Fig. 19.4, when C_2 (resp. C_1) reaches L_3 on its way back to the base station, it has a positive amount of residual energy for safely returning to the base station. In other words, C_2 (resp. C_1) carries this particular part of energy during its travelling from L_3 to L_2 (resp. L_1), and finally to L_3 again. How about letting C_3 stop moving forward at a place L_3' which is closer to the base station than $L_3 = 5\frac{5}{6}$? In doing so, C_3 can wait at a place with sufficient energy to support C_2 and C_1's travelling from L_3' to the base station.

We therefore design PushWait, the optimal scheduling algorithm that achieves the highest EUE. Formally, suppose that we require M chargers to cover a given WSN, then each charger C_i in PushWait follows the iterative process below.

1. C_i starts from the base station with a full battery; it then gets fully charged at locations L_M, L_{M-1},..., and L_{i+1}.
2. C_i charges sensor nodes between L_{i+1} and L_i. When it arrives at L_i, it charges C_{i-1}, C_{i-2},..., and C_1, such that these $(i-1)$ chargers' batteries are full.
3. C_i waits at L_i. When all of C_1, C_2,..., and C_{i-1} return to L_i, it evenly distributes its residual energy among these i chargers (including C_i itself).

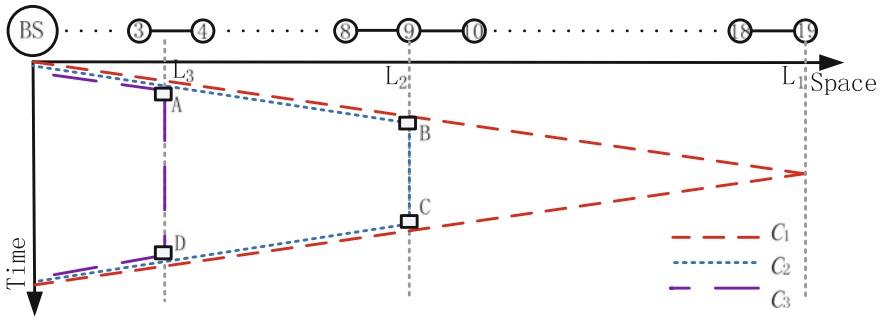


Fig. 19.6 Time-space view of PushWait with the same settings as in Figs. 19.3, 19.4, and 19.5. We have $L_3 = 3\frac{1}{3}$, $L_2 = 9$, $L_1 = 19$. C_3 has 14 J residual energy

4. On C_i's way back to the base station, it gets charged at locations L_{i+1}, L_{i+2},..., and L_M, which makes it have just enough energy to return to the base station.

The reason for naming this scheduling after "PushWait" is clear: each charger "pushes" some other chargers to move forward, and "waits" for their returns.

Figure 19.6 shows the time-space view of PushWait with the same settings as in Figs. 19.3, 19.4, and 19.5. The running details are as follows. Mobile chargers C_1, C_2, and C_3 start from the base station with $P = 80$ J energy. At $L_3 = 3\frac{1}{3}$, both C_1 and C_2 have $80 - 3 \cdot L_3 = 70$ J energy, while C_3 has $80 - 3 \cdot L_3 - 2 \times 3 = 64$ J energy, because it charges nodes s_1, s_2, and s_3. Then, we let $C_3 \overset{10}{\rightarrow} C_2$ and $C_3 \overset{10}{\rightarrow} C_1$. After this, C_3 waits at L_3 with 44 J energy. Similarly, after C_2 charges nodes from s_4 to s_9, and charges C_1 to its full battery at $L_2 = 9$, C_2 waits at L_2 with 34 J energy. When C_1 returns to L_2 after charging nodes from s_{10} to s_{19}, as the reader can verify, it has exactly 0 energy. Then, we let $C_2 \overset{17}{\rightarrow} C_1$. Note that, 17 J energy is just enough for C_1 or C_2 to move from L_2 to L_3. At L_3, we let $C_3 \overset{10}{\rightarrow} C_2$ and $C_3 \overset{10}{\rightarrow} C_1$. Again, note that, 10 J energy is just enough for C_1 or C_2 to move from L_3 to the base station. When they return to the base station, only C_3 has 14 J residual energy. PushWait covers 19 sensor nodes.

19.4.3.1 Parameters

We present how to determine L_i ($1 \le i \le M$) to maximize EUE in this subsection. Remember that, sensor nodes are uniformly distributed, and therefore, we make the following approximation: given d' distance, we have approximately d'/d sensors.

Let us analyze the interval between L_{i+1} and L_i. C_i gets fully charged at L_{i+1} and reaches L_{i+1} with 0 energy on its way back to the base station. The full battery P is used up for the following reasons: (*i*) C_i charges sensors between L_{i+1} and L_i, (*ii*) C_i moves from L_{i+1} to L_i, (*iii*) C_i transfers some energy to C_1, C_2, ..., and C_{i-1} at L_i for the first time. Note that these $i - 1$ chargers are fully charged at L_{i+1}, and

thus the energy transferred to them at L_i is exactly the energy consumed by their travellings from L_{i+1} to L_i; (*iv*) C_i transfers some energy to C_1, C_2, ..., and C_{i-1} at L_i for the second time, which is just enough for them to travel from L_i to L_{i+1}, and (*v*) C_i moves from L_i to L_{i+1}. Combining the above reasons together, we have the following equations.

$$2icd(L_i - L_{i+1}) + b(L_i - L_{i+1})/d = P(1 \leq i < M) \tag{19.2}$$

$$2Mcd(L_M - 0) + b(L_M - 0)/d \leq P \tag{19.3}$$

The second formula is an inequality, since PushWait cannot always use up exactly the total energy of M chargers. We then have:

$$L_1 = Nd \tag{19.4}$$

$$L_i = Nd - \sum_{j=1}^{i-1} \frac{Pd}{2cd^2j + b}(2 \leq i \leq M) \tag{19.5}$$

The number of chargers M can be determined by: $L_M > 0$, $L_{M+1} \leq 0$. We further have $E^{pl} = Nb$ and $E^{mm} = 2cd \sum_{i=1}^{M} L_i$. The duration of a charging round is Nd/v, and the scheduling cycle is equal to the recharging cycle, i.e., τ.

19.4.3.2 Properties

Theorem 19.1 (Optimality of PushWait) *For the settings described in Sect. 19.4, PushWait is optimal in terms of EUE.*

Proof Denote by $Distance(alg)$ the sum of travelling distances by all mobile chargers in a scheduling algorithm alg. As we mentioned before, it is sufficient to prove that $Distance(PushWait)$ is the minimum. Suppose that PushWait requires M mobile chargers to replenish the given WSN. We prove the theorem by mathematical induction on M.

$M = 1$. $Distance(PushWait) = 2L_1$, where L_1 is the length of the given WSN. We note that any scheduling algorithm $anyalg$ must have at least one charger to charge the farthest sensor node in the WSN, therefore, $Distance(anyalg) \geq 2L_1 = Distance(PushWait)$.

$M = 2$. (By contradiction) Suppose that PushWait is not optimal, and the optimal scheduling algorithm is OPT_2. Since one charger cannot cover the entire WSN, there are at least two chargers in OPT_2. One of them, say C', must charge the farthest sensor, and thus it moves at least $2L_1$ distance. By definition, we should have $Distance(OPT_2) < Distance(PushWait) = 2L_1 + 2L_2$. Therefore, all the other chargers in OPT_2 cannot travel as far as L_2. However, according to our calculation of L_2 in PushWait, a charger with a full battery at L_2 can only charge the sensors between L_2 and L_1 and return to L_2 with 0 energy; then we know C' in OPT_2 can, by

no means, reach L_1: a contradiction! Therefore, no such OPT_2 exists, and PushWait is optimal.

I.H.: PushWait is optimal for any $M < n$.

$M = n$. (By contradiction) Suppose that PushWait is not optimal, and the optimal scheduling algorithm is OPT_n. Imagine that a virtual base station BS' is located at L_n, then, OPT_n and PushWait require Q and $(n-1)P$ energy, respectively, to cover the sensors between L_n and L_1. By the induction hypothesis, $Q > (n-1)P$. Then, the task of OPT_n is to cover the sensors from the base station to L_n and to deliver Q energy to L_n. It is then straightforward to see that OPT_n requires at least n chargers to reach L_n; otherwise, the total residual energy of less than n chargers at L_n is definitely less than $(n-1)P$. Since $Q > (n-1)P$, OPT_n consumes more energy than PushWait: a contradiction! No such OPT_n exists, and PushWait is optimal. □

Theorem 19.2 (Coverage) *The maximum coverages of EqualShare, SolelyCharge, CLCharge, and PushWait are $P/2c$, $P/2c$, P/c, and infinity, respectively.*

The detailed proofs of Theorems 19.2, 19.3, 19.4, and 19.5 can be found in [9, 29].

Our problem resembles the banana-eating camel problem [33] to a limited extent. A farmer has 3,000 bananas that will be sold at a market 1,000 miles away. He has only a camel that can carry at most 1,000 bananas at a time but will eat 1 banana to refuel for each mile it walks. Is there any method to delivery any bananas to the market? If yes, how? There are two major differences between our problem and this one: first, energy can only be exchanged between chargers, while bananas could be placed on the ground; second, only one camel is involved, implying there is no collaboration.

19.4.3.3 Performance

Following similar settings in [5], we assume that sensor nodes are powered by a 1.5 V 2000 mAh Alkaline rechargeable battery, then the battery capacity (b) is 1.5 V × 2 A × 3600 s = 10.8 KJ. The battery capacity of a mobile charger (P) is 2000 KJ; the moving speed of a charger (v) is 1 m/s; the charger's moving power consumption rate is 50 W, thus, the moving cost of a charger (c) is 50 J/m. We assume that sensor nodes are uniformly deployed over a 10 Km straight line. By default, the number of sensor nodes (N) is 400; the wireless charging efficiency (η_1) is 1.5 %; the charging efficiency between mobile chargers (η_2) is 30 %. We compare PushWait with EqualShare, SolelyCharge, CLCharge, and *GreedyPlus* [5]. The original version of GreedyPlus does not consider multiple chargers, and we tailored it to our scenarios: multiple mobile chargers are seen as one large charger, which adopts binary search to

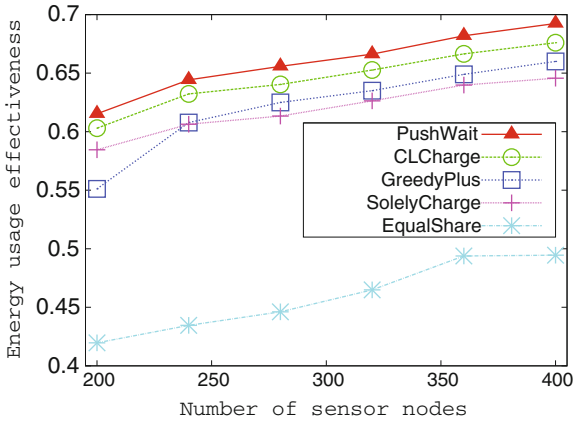


Fig. 19.7 Performance comparisons by varying the number of sensor nodes

find a suitable target network lifetime. Note that, a sensor may be recharged several times in a charging round in GreedyPlus.

Figure 19.7 shows the performance comparisons in scenario $K1K2K3$ by varying the number of sensor nodes. When the number of sensor nodes increases, since they are restricted to the 10 km long line, the density of sensor nodes also increases. This is so that mobile chargers can transfer energy to more sensor nodes without incurring much moving cost. According to this, all of the five algorithms perform better when the number of sensor nodes increases. The main reason for the relatively low EUE of EqualShare is that, every charger in EqualShare has to move to the farthest sensor node, and thus, the increase in the number of sensor nodes leads to an increase in the amount of overhead energy. In summary, as we theoretically demonstrated earlier, PushWait achieves the highest EUE among the five algorithms. CLCharge takes advantage of collaboration between chargers, and so has the second highest EUE. GreedyPlus greedily selects the next charging target, outperforming SolelyCharge and EqualShare. EqualShare has the worst performance. Comparisons between EqualShare, SoleCharge, and CLCharge can be found in [28].

19.4.4 Remarks

We now show how to apply PushWait to contexts without the short duration (SD) and low cycle (LC) assumptions.

Without SD. When the time for charging a sensor node to its full battery is not negligible compared to the recharging cycle, we use the first charging round to synchronize the battery levels of all sensor nodes. The purpose of synchronization is to make sure that, each sensor node s_i would require exactly b amount of energy when the mobile charger approaches it in the following charging rounds. Since the energy

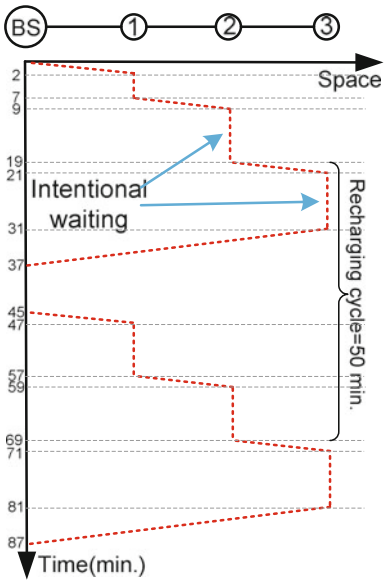


Fig. 19.8 Without the SD assumption. The first charging round (0 to 37 min) synchronizes the battery levels through intentionally waiting. In the figure, the distance between two consecutive sensor nodes is 10 m, $b = 2$ J, $P = 200$ J, $c = 3$ J/m, $v = 5$ m/min., and the charger can transfer 0.2 J to a node in 1 min

transfer rate is fixed, we can modify our scheduling algorithms by intentionally adding a fixed duration at each sensor node.

We use Fig. 19.8 for further explanation, where 3 sensor nodes are 10 m apart. The battery capacity of each node is 2 J, and the recharging cycle of each node is 50 min. Therefore, the energy consumption rate is 0.04 J/min. The charger can transfer 0.2 J energy to a node in 1 min. Therefore, it takes 10 min to transfer 2 J to a node.

Suppose that when the charger arrives at s_1 for the first time, it has 1.2 J residual energy; since $(2 - 1.2)/(0.2 - 0.04) = 5$, it takes 5 min for the charger to replenish s_1 to its full battery. When the charger arrives at s_2, it also has 1.2 J residual energy. Although the charger could finish charging s_2 in 5 min, *the charger should intentionally wait another 5 min before heading for* s_3. This should be done for the purpose of synchronizing energy levels among sensor nodes, as shown in Fig. 19.8. In doing so, in the following charging rounds, each sensor node will have exactly 0.4 J energy when the charger begins to recharge it, implying that it would take the same amount of time (i.e., 10 min) for the charger to recharge each node to its full battery. For instance, s_1, s_2, and s_3 have full batteries at the 7th, 19th, and 31st min, respectively; in the second charging round, they become fully charged at the 47th, 59th, and 71st min, respectively. We see that, the corresponding time interval for every sensor is 50 min, which is the recharging cycle of each sensor node.

Without LC. Generally speaking, when the recharging cycle of a sensor node is not longer than a charging round, we can adopt a pipeline-like solution. Figure 19.9 shows an example. Since the recharging cycle is 24 min, the 2nd (resp. 3rd) charging

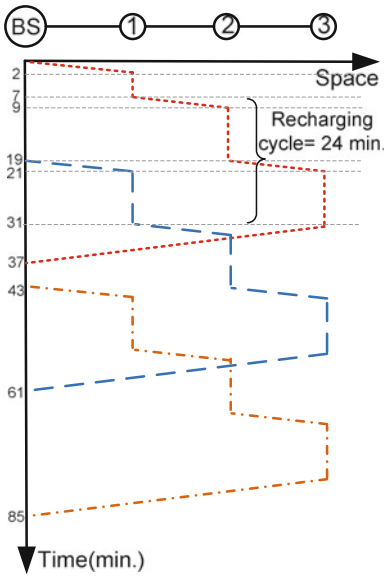


Fig. 19.9 Without the LC assumption. Pipeline-like periodic charging rounds are scheduled to cope with short recharging cycle (24 min < 42 min). In the figure, the distance between two consecutive sensor nodes is 10 m, $b = 2$ J, $P = 200$ J, $c = 3$ J/m, $v = 5$ m/min., and the charger can transfer 0.2 J to a node in 1 min

round has to start at the 19th (resp. 43rd) minute. It is not hard to see that, pipeline-like PushWait can still achieve optimality. An additional requirement of such a solution is that, it needs more chargers, e.g., 2 chargers are required in Fig. 19.9.

19.5 Extensions

Three main assumptions that are made in Sect. 19.4 include: ($K1$) all sensor nodes are distributed along a one-dimensional (1-D) line; ($K2$) the recharging cycles of all sensor nodes are the same; ($K3$) there is no energy loss during any energy transfers. In this section, we consider several scenarios that remove these conditions one by one, and finally investigate the mobile charging scheduling problem in general 2-dimensional (2-D) WSNs. We use $\overline{Kj}$ to indicate that Kj does not hold, $j \in \{1, 2, 3\}$. For example, $K1K2\overline{K3}$ represents the scenario in which $K1$ and $K2$ hold, while $K3$ does not hold.

The recharging cycle of s_i is denoted by τ_i. We denote by η_1 the wireless charging efficiency between a charger and a sensor node, i.e., a charger C consumes one unit of energy while a sensor can only receive η_1 units of energy. Similarly, denote by η_2 the efficiency between two chargers. For example, if $C_i \xrightarrow{e} C_j$ (resp. $C_i \xrightarrow{e} s_j$), then C_j (resp. s_j) receives only $\eta_2 e$ (resp. $\eta_1 e$) units of energy.

19.5.1 Energy Loss

We first study the scenario where $K3$ does not hold. Mobile chargers' collaboration helps PushWait achieve optimality; however, when energy loss during energy transfer is not negligible, collaboration increases E^{ls}, and hence, may impair the EUE of PushWait. We use the following example to illustrate this observation.

Figures 19.10, 19.11, 19.12 and 19.13 show the time-space views of four scheduling algorithms. The farthest distance that C_i moves away from the base station is determined via the same analysis as before. The settings are similar to those in Figs. 19.3, 19.4, 19.5 and 19.6, except that $\eta_1 = 0.5$ and $\eta_2 = 0.25$.

Taking Fig. 19.10 for example, C_3 can cover only 8 sensors, because $8c + 8c + 2 \times 8\eta_1 = 80$; C_1 can cover only s_{12}, because it cannot return to the base station if it

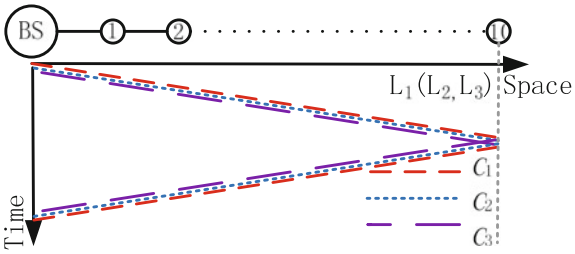


Fig. 19.10 EqualShare, $L_3 = L_2 = L_1 = 10$

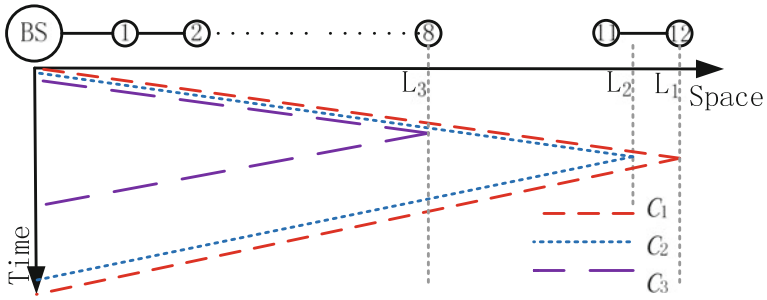
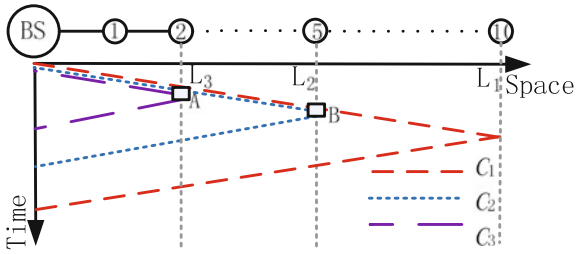


Fig. 19.11 SolelyCharge, $L_3 = 8$, $L_2 = 11\frac{1}{3}$, $L_1 = 12$. C_1 has 4 J residual energy

Fig. 19.12 CLCharge, $L_3 = 2, L_2 = 5, L_1 = 10$. C_1 has 4 J residual energy



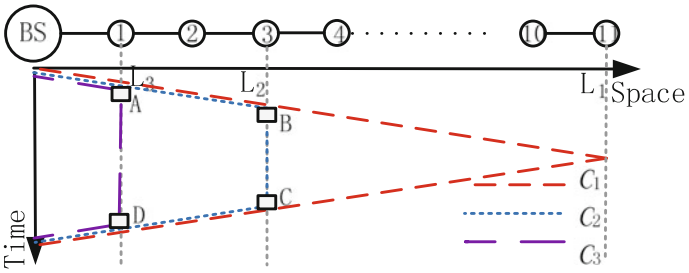


Fig. 19.13 PushWait, $L_3 = 1, L_2 = 3, L_1 = 11$. C_3 has 10 J residual energy

covers s_{13} as well. When all three chargers return to the base station, only C_1 has 4 J residual energy. The numbers of sensor nodes that can be covered in four algorithms are 10, 12, 10, and 11, respectively; their EUEs are $\frac{10\times 2}{240-20} \approx 0.091$, $\frac{12\times 2}{240-4} \approx 0.102$, $\frac{10\times 2}{240-4} \approx 0.085$, and $\frac{11\times 2}{240-10} \approx 0.096$, respectively.

19.5.1.1 ηPushWait

Recall that PushWait is optimal for scenario $K1K2K3$; however, in scenario $K1K2\overline{K3}$, due to the energy loss between chargers, its EUE is only the second highest, while SolelyCharge achieves the highest EUE. This example suggests to us that SolelyCharge may perform better than PushWait for $K1K2\overline{K3}$.

Theorem 19.3 (Optimality of SolelyCharge) *If collaboration among chargers is not permitted, SolelyCharge is optimal in terms of EUE for* $K1K2\overline{K3}$*,*

Though SolelyCharge is optimal if collaboration is not allowed, it has limited coverage (see Theorem 19.2). PushWait is not optimal for scenario $K1K2\overline{K3}$, but it can cover a 1-dimensional WSN of infinite length.

We therefore propose combining SolelyCharge with PushWait to construct our solution η**PushWait**, which is better than either of them individually. Denote by $cg(alg, M)$ the coverage of a scheduling algorithm alg with M mobile chargers. For example, in scenario $K1K2\overline{K3}$, if we assume that, $\forall 1 \le i \le N, b_i = b$ and $x_i = i \cdot d$, following a similar analysis as in Sect. 19.4.3, we have

$$cg(SolelyCharge, M) = \sum_{i=1}^{M} \frac{\eta_1 d b^{i-1} P}{(2\eta_1 c d^2 + b)^i} \tag{19.6}$$

$$cg(PushWait, M) = \sum_{i=0}^{M-1} \frac{\eta_1 \eta_2 d P}{\eta_2 b + 2\eta_1 c d^2 (\eta_2 + i)} \tag{19.7}$$

Given a WSN and mobile chargers that satisfy $K1K2\overline{K3}$, let M' be the largest value of M that ensures $cg(SolelyCharge, M) \ge cg(PushWait, M)$, i.e.,

$$M' = \arg \max_{cg(SolelyCharge,M) \geq cg(PushWait,M)} M. \tag{19.8}$$

Then, ηPushWait can be constructed as follows. If the length of the given WSN is not greater than $cg(SolelyCharge, M')$, we use SolelyCharge. Otherwise, we have the following strategy: we use SolelyCharge to charge sensors between the base station and $cg(SolelyCharge, m)$, and use PushWait to charge the remaining sensors, where m $(1 \leq m \leq M')$ is a positive integer that maximizes the EUE of such a strategy.

19.5.1.2 Performance

We evaluate the impact of charging efficiencies in this subsection. Figure 19.14 shows the case where η_2 is fixed, i.e., the charging efficiency between chargers is fixed. When we increase η_1, the energy loss during wireless charging becomes smaller, so the EUE of each algorithm gets larger.

Figure 19.15 shows the case where η_2 is fixed. There are three interesting observations. First, since there is no energy transfer between chargers in SolelyCharge,

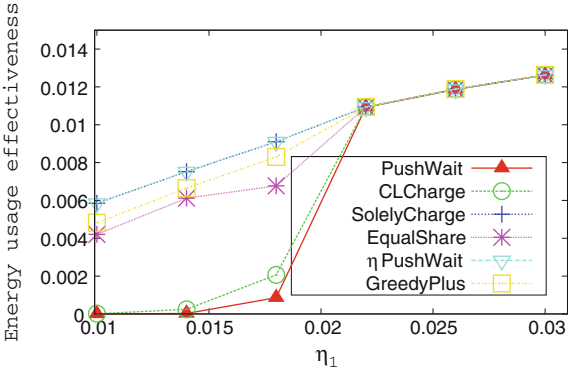


Fig. 19.14 Performance comparisons in scenario $K1K2\overline{K3}$ by varying η_1 while keeping $\eta_2 = 0.3$

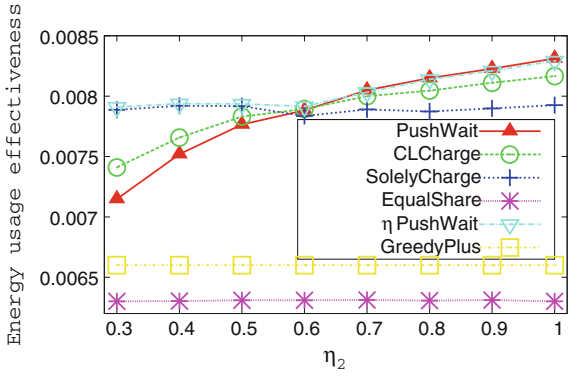


Fig. 19.15 Performance comparisons in scenario $K1K2\overline{K3}$ by varying η_2 while keeping $\eta_1 = 0.015$

GreedyPlus and EqualShare, their EUEs remain unchanged when η_2 increases. Second, CLCharge has a higher and lower EUE than PushWait when $\eta_2 \leq 0.6$ and $\eta_2 \geq 0.6$, respectively. The rationale behind this phenomenon is that, the total energy exchanged between chargers in CLCharge is less than that in PushWait. Thus, CLCharge may perform better than PushWait if η_2 is small. Third, ηPushWait always has the best performance, because it takes advantage of SolelyCharge when $\eta_2 \leq 0.6$, and takes advantage of PushWait when $\eta_2 > 0.6$.

19.5.2 Different Recharging Cycles

We consider $K1\overline{K2}K3$ in this subsection. When sensor nodes have different recharging cycles, a scheduling cycle may include multiple charging rounds, which greatly complicates the scheduling problem. Figure 19.16 shows a scheduling example in scenario $K1\overline{K2}K3$. There are six sensor nodes in the WSN: $\tau_1 = 2$, $\tau_2 = 4$, $\tau_3 = 3$, $\tau_4 = 7$, $\tau_5 = 6$, and $\tau_6 = 5$. All sensor nodes are initialized to their full batteries at time 0. At time 1, we plan to charge s_1, s_2, and s_3. Since $\tau_2 = 4$ and s_2 has a full battery at time 0, s_2 needs only $b/4$ energy at time 1. So we employ PushWait to deliver $b/2$, $b/4$, and $b/3$ energy to s_1, s_2, and s_3, respectively, at time 1. In the scheduling example, there are also charging rounds at time points 2 and 4. From time 5, the three charging rounds between time points 1 and 4 are repeated. Two important questions can be raised for such a scheduling example.

First and foremost, how are we to go about characterizing long-term EUE? Since the scheduling cycle is 4, we can use the EUE within a scheduling cycle to exactly represent the long-term EUE. Second, how might we define the scheduling cycle? As we mentioned before, it is defined as the time interval between two consecutive time points when all sensors are fully charged. For example, the WSN in the figure is fully charged at time points 0, 4, 8, and so on, so the scheduling cycle is 4.

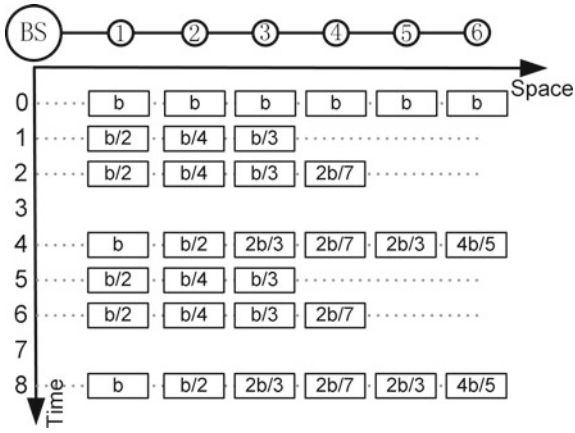


Fig. 19.16 A scheduling example for $K1\overline{K2}K3$ where $\tau_1 = 2$, $\tau_2 = 4$, $\tau_3 = 3$, $\tau_4 = 7$, $\tau_5 = 6$, and $\tau_6 = 5$. For instance, at time 2, we use PushWait to deliver $b/2$, $b/4$, $b/3$, and $2b/7$ energy to s_1, s_2, s_3, and s_4, respectively. The scheduling cycle of this example is 4

Second, when should we plan to make a charging round? The next theorem tells us that, we only need to start a charging round when there is at least one dying sensor node. For example, the charging round at time 1 in Fig. 19.16 is redundant, since no sensor nodes will run out of energy if the charging round is cancelled.

Theorem 19.4 (Necessary condition) *Given a node s that is x_s distance away from the base station, the battery capacity of s is b; using PushWait to deliver b energy to s one time achieves a higher EUE than using PushWait twice.*

At one extreme, when we plan to recharge a sensor node, we want to transfer as much energy as possible to it, so as to increase the payload energy. Based on this intuition, we start a charging round only when there is at least one dying sensor node, and in this charging round, we only charge the dying sensor nodes. With the same settings as in Fig. 19.16, Figure 19.17 shows such an example: a sensor node is charged only when it is dying. At another extreme, when there is a charging round, we want to charge as many sensor nodes as possible, so as to increase the payload energy. Figure 19.18 demonstrates this extreme case.

In fact, these two design options compete with each other; thus, we strive to strike a balance between them, and propose our solution ClusterCharging(β).

19.5.2.1 ClusterCharging(β)

We first sort sensor nodes in decreasing order of their recharging cycles, then we divide them into groups such that the ratio of the maximum recharging cycle to the minimum recharging cycle in each group is not greater than a given threshold, say β. Then, we start a charging round only when there is at least one dying sensor node, and in this charging round, we employ PushWait to charge all sensor nodes in a group on the condition that this group contains at least one dying sensor node. Note that, in each charging round, different sensor nodes may need different amounts of energy, e.g., s_1, s_2, and s_3 require $b/2$, $b/4$, and $b/3$ energy, respectively, at time 1 in Fig. 19.16. Remember that PushWait can still achieve its optimality in each round, due to Theorem 19.1.

Let us take Figs. 19.17, 19.18, 19.19 and 19.20 for example. "$\beta = 1$" represents the extreme case in Fig. 19.17, where each sensor node, itself, forms a group. Similarly, "$\beta = +\infty$" represents the other extreme case in Fig. 19.18, where all sensor nodes form a single group. In Fig. 19.18, we consider sensor nodes in decreasing order of their recharging cycles, i.e., s_1, s_3, s_2, s_6, s_5, and s_4. First, s_1 forms a group $\{s_1\}$; we then attempt to put s_3 into $\{s_1\}$, since $\tau_3/\tau_1 = 1.5 < \beta = 2$, it is feasible for them to be in the same group; the group $\{s_1, s_3\}$ can also accommodate s_2; when we want to put s_6 into $\{s_1, s_3, s_2\}$, as $\tau_6/\tau_1 > \beta$, s_6 forms a new group, and so on. In Fig. 19.19, β is set to 3, resulting in two groups, i.e., $\{s_1, s_2, s_3, s_5, s_6\}$, and $\{s_4\}$.

Different values of β lead to different EUEs of ClusterCharging(β), and the optimal value of β varies with the parameters of a given MCS problem. Therefore, for a given MCS problem that satisfies $K1\overline{K2}K3$, we maximize the EUE

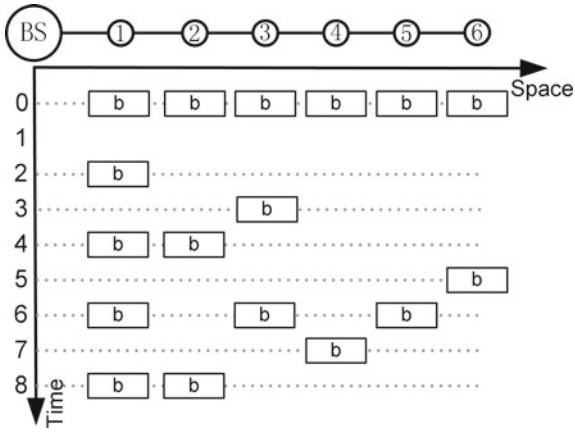


Fig. 19.17 $\beta = 1$, there are six groups; each sensor, itself, forms a group. Illustrations of ClusterCharging(β) for $K1\overline{K2}K3$ with the same settings as in Fig. 19.16

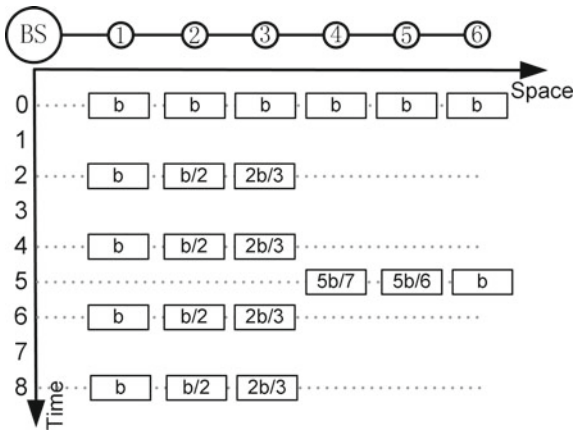


Fig. 19.18 $\beta = 2$, there are two groups: $\{s_1, s_2, s_3\}$, and $\{s_4, s_5, s_6\}$

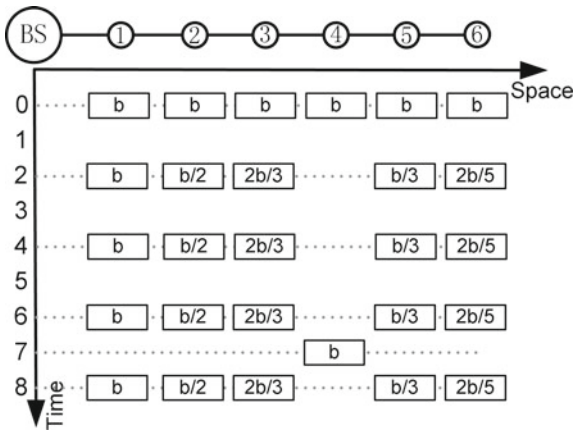


Fig. 19.19 $\beta = 3$, there are two groups: $\{s_1, s_2, s_3, s_5, s_6\}$, and $\{s_4\}$

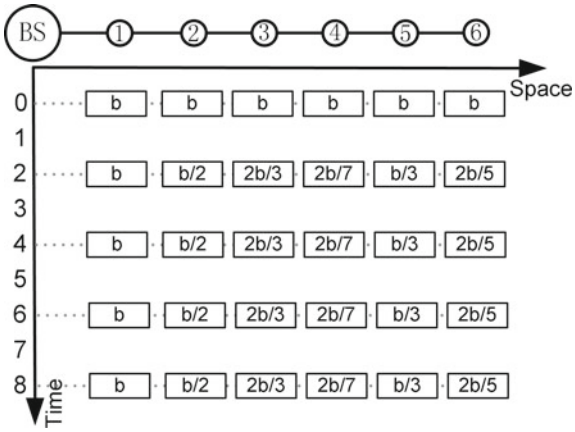


Fig. 19.20 $\beta = +\infty$, there is one group that contains all sensors

of ClusterCharging(β) by searching the optimal β in range $[1, \frac{\tau_{max}}{\tau_{min}} + 1]$, where $\tau_{min} = min_{1 \le i \le N} \tau_i$, and $\tau_{max} = max_{1 \le i \le N} \tau_i$.

Theorem 19.5 (Performance guarantee of ClusterCharging(β)) *The approximation ratio of ClusterCharging(β) for $K1\overline{K2}K3$ is*

$$\frac{b_{min}(2cx_N + \sum_{i=1}^{N} b_i)}{P\tau_{max} k \sum_{i=1}^{N} b_i} \tag{19.9}$$

where $b_{min} = min_{i=1}^{N} b_i$, and

$$k = argmin_{(\sum_{i=1}^{k} \frac{1}{i} \ge \frac{2cx_N \tau_{max} + b_{min}}{P\tau_{max}})} k \tag{19.10}$$

For scenario $K1\overline{K2K3}$, we design an algorithm called η*ClusterCharging*(β), in which sensor nodes are divided into groups in a similar way as ClusterCharging(β). However, in each charging round, we employ ηPushWait instead of PushWait to replenish sensor nodes.

19.5.2.2 Performance

We evaluate ClusterCharging(β) under different values of β: $\beta = 1$ to 5 in Fig. 19.21 and $\beta = 1$ to 3 in Fig. 19.22. We notice that, ClusterCharging(β) with three different β's perform almost the same in Fig. 19.21, while ClusterCharging(3) outperforms the other two algorithms in Fig. 19.22. The main reason is that, the relative gap between recharging cycles in Fig. 19.21 is large, while the relative gap in Fig. 19.22 is small. For example, if we use ClusterCharging(5) to replenish the WSN in Fig. 19.21, then the energy we have to transfer to each sensor node varies from 1.8 KJ ($b/6$) to 10.8 KJ (b). We can see that, some sensor nodes just need a small amount of energy.

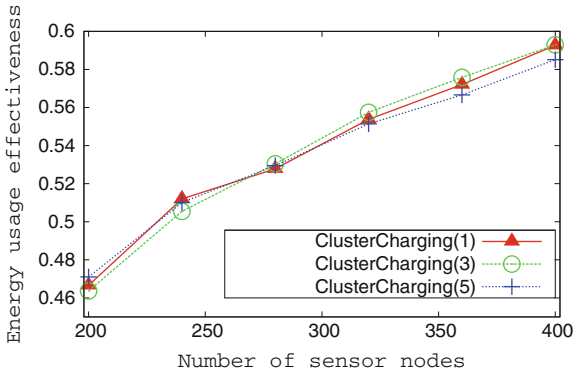


Fig. 19.21 Performance comparisons in scenario $K1\overline{K2}K3$ when all τ_i's are uniformly generated between 1 and 6

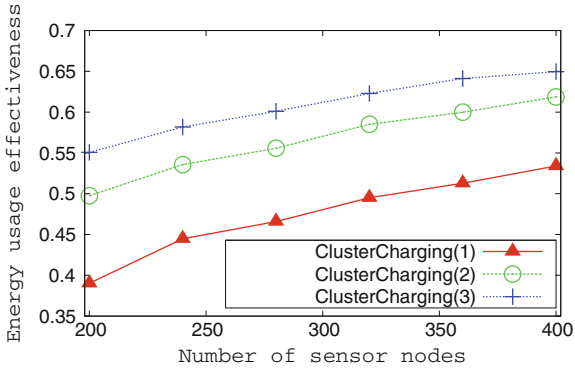


Fig. 19.22 Performance comparisons in scenario $K1\overline{K2}K3$ when all τ_i's are uniformly generated between 3 and 8

However, if we use ClusterCharging(5) to replenish the WSN in Fig. 19.22 (note that, ClusterCharging(5) is equivalent to ClusterCharging(3) for the setting in Fig. 19.22), then the energy we have to transfer to each sensor node varies only from 4.05 KJ ($3b/8$) to 10.8 KJ (b). Therefore, ClusterCharging(5) can have the best performance in Fig. 19.22.

19.5.3 2-D Networks + Different Recharging Cycles + Energy Loss

Different recharging cycles force us to divide sensor nodes into groups, and energy loss makes us combine SolelyCharge with PushWait. What challenges does 2-D pose to us? The answer is that, given a set of sensor nodes to be replenished in a charging round, we must discover how to determine the charging sequence, so as to minimize the overhead energy.

Before presenting our design, we introduce some notations. In a charging round, denote by S the set of sensor nodes that are going to be charged. We construct

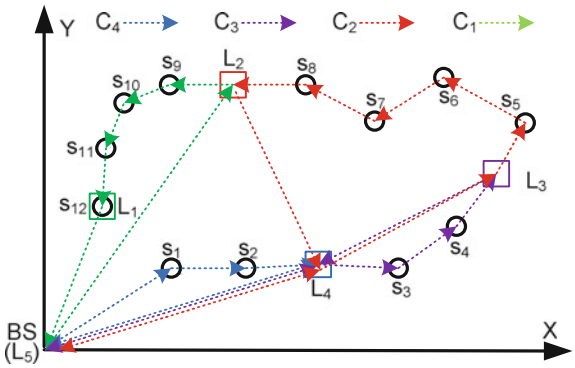


Fig. 19.23 Mobile chargers are no longer restricted to moving along a Hamiltonian cycle; they can take shortcuts to improve EUE when possible

a complete graph $G[S]$ with vertices being $S \bigcup \{BS\}$, and edge weights being the Euclidean distance between two corresponding vertices.

We denote the minimum weight Hamiltonian cycle of $G[S]$ by H. Denote by $d_H(P_1, P_2)$ the sum of Euclidean distances of line segments between two positions P_1 and P_2 on H. For example, $d_H(L_4, L_3) = d(L_4, s_3) + d(s_3, s_4) + d(s_4, L_3)$ in Fig. 19.23.

19.5.3.1 HηClusterCharging(β)

In HηClusterCharging(β), we first divide sensor nodes into groups, and plan a charging round when there is at least one dying sensor node; in each charging round, we try to find the minimum weight Hamiltonian cycle in the complete graph on the corresponding set of sensors and the base station. Lastly, we apply ηPushWait to the Hamiltonian cycle, and further improve the results through shortcutting. More specifically, HηClusterCharging(β) works as follows:

1. Sort sensor nodes in decreasing order of their recharging cycles, then divide sensor nodes into groups with respect to a threshold β, as in ClusterCharging(β).
2. Decide the charging round plan, that is, decide the set of sensor nodes S that should be replenished in each charging round. We apply the following steps, i.e., 3–5, to each charging round.
3. Construct a complete graph $G[S]$ and use the minimum spanning tree-based heuristic [34] to generate a Hamilton cycle H in $G[S]$.
4. Randomly choose a direction for H. Suppose that we start from the base station and visit sensor nodes following the chosen direction along H. Without loss of generality, denote the sequence of sensor nodes we visit by s_1, s_2, ..., and $s_{|S|}$. We apply ηPushWait to H, which can be seen as a 1-dimensional manifold [35], and thus, we can obtain the number of required chargers M and the farthest positions that chargers will reach, i.e., L_1, L_2, ..., and L_M. Again, without loss of generality, we let $L_{M+1} = (0, 0)$. (Note that, since we apply ηPushWait to this 1-D manifold, each L_i will be located on an edge between two consecutive sensor nodes or at

the location of a sensor node; particularly, L_1 is located at the same location with the farthest sensor node, i.e., $s_{|S|}$.)

5. We improve ηPushWait through shortcutting. In this step, we only present how chargers take shortcuts and do not elaborate on the energy transfers between mobile chargers, for the sake of presentation brevity.

   a. For charger C_M, it charges the sensor nodes between the base station and L_M, transfers energy to the other chargers at L_M, and waits at L_M for the other chargers' return. When C_M finishes its charging task, it can take a shortcut: it directly returns to the base station.
   b. For charger C_i $(1 \leq i \leq M-1)$, denote the current position of C_i as L_g. Before it finishes charging the sensor nodes between L_{i+1} and L_i
      i. When $i+2 \leq g \leq M+1$, it directly takes a shortcut to $L_{j_{min}}$, where j_{min} satisfies:

$$j_{min} = argmin_{(d(L_g,L_j) \leq d_H(L_{j+1},L_j), i+1 \leq j \leq g-1)} j \tag{19.11}$$

      ii. When $g = i+1$, it begins to charge the sensor nodes between L_{i+1} and L_i.
      iii. After this, on its way back to the base station: for $i \leq g \leq M$, it directly takes a shortcut to $L_{j_{max}}$, where j_{max} satisfies:

$$j_{max} = argmax_{(d(L_g,L_j) \leq d_H(L_g,L_{g+1}), g+1 \leq j \leq M+1)} j \tag{19.12}$$

19.5.3.2 Running Example

Figure 19.23 shows when and how a charger can take a shortcut. Suppose that we have to replenish 12 sensor nodes in a charging round. Without loss of generality, the Hamiltonian cycle H we find is $BS - s_1 - s_2 - ... - s_{12} - BS$. After applying ηPushWait to this cycle, we know that, this round requires four chargers, and the farthest position of each charger is L_i $(1 \leq i \leq 4)$.

C_4 is responsible for replenishing sensor nodes between L_5 (base station) and L_4. It can only take a shortcut after it completes its task, and thus, its trajectory is $BS \rightarrow s_1 \rightarrow s_2 \rightarrow L_4 \Rightarrow BS$, where "$\rightarrow$" denotes a path segment along H, and "$\Rightarrow$" denotes a shortcut.

When C_3 starts from the base station, since $g = 5$, the situation satisfies case 5.2.1 in HηClusterCharging(β), and we can determine $j_{min} = 4$, so it directly moves to L_4. Then, the situation satisfies case 5.2.2, and thus, C_3 charges sensor nodes between L_4 and L_3. The situation begins to satisfy case 5.2.3 when it arrives at L_3, as $d(L_3, L_5) > d_H(L_3, L_4)$, it does not have enough energy to move directly to the base station. (Please keep in mind that, according to ηPushWait, if C_3 moves along H to L_4, it would have 0 energy at L_4.) Thus, the trajectory of C_3 is $BS \Rightarrow L_4 \rightarrow s_3 \rightarrow s_4 \rightarrow L_3 \Rightarrow L_4 \Rightarrow BS$.

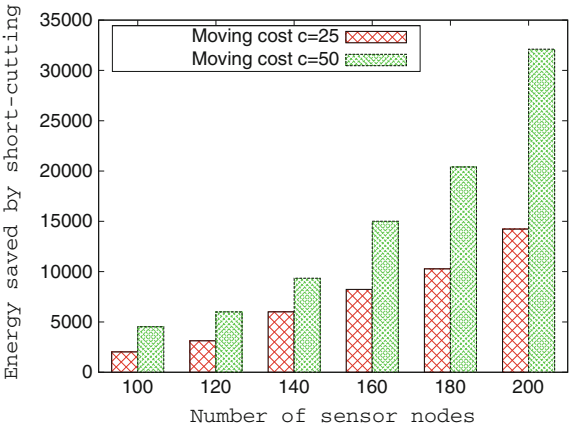


Fig. 19.24 The benefit of shortcutting

Following a similar argument, we can have the trajectory of C_2: $BS \Rightarrow L_4 \Rightarrow L_3 \rightarrow s_5 \rightarrow s_6 \rightarrow s_7 \rightarrow s_8 \rightarrow L_2 \Rightarrow L_4 \Rightarrow BS$.

When C_1 is at L_5, since $d(L_5, L_2) < d_H(L_3, L_2)$, we have $j_{min} = 2$, and it takes a shortcut to L_2. After charging sensor nodes between L_2 and L_1, it arrives at L_1. As $d(L_1, L_5) < d_H(L_1, L_2)$, we have $j_{max} = 5$, thus, it directly returns to the base station. The trajectory of C_1 is $BS \Rightarrow L_2 \rightarrow s_9 \rightarrow s_{10} \rightarrow s_{11} \rightarrow s_{12} \Rightarrow BS$.

19.5.3.3 Performance

We are interested in investigating how much benefit shortcutting brings about. The results are shown in Fig. 19.24. When the moving cost of a mobile charger (c) is fixed, if the number of sensor nodes increases, the energy saved by shortcutting also increases. This is reasonable, since an increase in the number of sensor nodes results in another increase in the number of mobile chargers required. Because of this, more chargers may take shortcuts when necessary. When the moving cost doubles, we find that the energy saved also doubles, or even triples. This is because, when moving cost increases, the number of chargers required also increases.

19.6 Concluding Remarks

In this chapter, we mainly present some recent work on collaborative mobile charging. Research results in this area can potentially be used in several mobile applications, including DARPA flying robots, and Google WiFi balloon. Through presenting this content, we aim to inspire readers to recognize the usefulness and importance of collaboration, incorporate it into their designs, and further elevate it.

Acknowledgments Jie Wu was supported in part by NSF grants CNS 1629746, CNS 1564128, CNS 1449860, CNS 1461932, CNS 1460971, CNS 1439672, CNS 1301774, and ECCS 1231461. Sheng Zhang was supported in part by NSFC (61502224, 61472181, 61472185), China Postdoctoral Science Foundation (2015M570434, 2016T90444), CCF-Tencent Open Research Fund (AGR20160104), Jiangsu NSF (BK20151390), and Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

1. Kurs, A., Karalis, A., Moffatt, R., Joannopoulos, J.D., Fisher, P.: Soljačić, M.: Wireless power transfer via strongly coupled magnetic resonances. Science **317**(5834), 83–86 (2007)
2. Fu, L., Cheng, P., Gu, Y., Chen, J., He, T.: Minimizing charging delay in wireless rechargeable sensor networks. In: Proceedings of IEEE INFOCOM 2013, pp. 2922–2930
3. Shi, Y., Xie, L., Hou, Y., Sherali, H.: On renewable sensor networks with wireless energy transfer. In: Proceedings of IEEE INFOCOM 2011, pp. 1350–1358
4. Xie, L., Shi, Y., Hou, Y.T., Lou, W., Sherali, H.D., Midkiff, S.F.: On renewable sensor networks with wireless energy transfer: The multi-node case. In: Proceedings of IEEE SECON 2012, pp. 10–18
5. Peng, Y., Li, Z., Zhang, W., Qiao, D.: Prolonging sensor network lifetime through wireless charging. In: Proceedings of IEEE RTSS 2010, pp. 129–139
6. He, S., Chen, J., Jiang, F., Yau, D.K., Xing, G., Sun, Y.: Energy provisioning in wireless rechargeable sensor networks. In: Proceedings of IEEE INFOCOM 2011, pp. 2006–2014
7. Madhja, A., Nikoletseas, S., Raptis, T.P.: Distributed wireless power transfer in sensor networks with multiple mobile chargers. Comput. Netw. **80**, 89–108 (2015)
8. Dai, H., Liu, Y., Chen, G., Wu, X., He, T.: Safe charging for wireless power transfer. In: Proceedings of IEEE INFOCOM 2014, pp. 1105–1113
9. Zhang, S., Wu, J., Lu, S.: Collaborative mobile charging for sensor networks. In: Proceedings of IEEE MASS 2012, pp. 84–92
10. Cheney, M., Glenn, J., Uth, R.: Tesla: Master of lightning. Barnes & Noble Publishing (1999)
11. Sample, A.P., Yeager, D.J., Powledge, P.S., Mamishev, A.V., Smith, J.R.: Design of an RFID-based battery-free programmable sensing platform. IEEE Trans. Instrum. Meas. **57**(11), 2608–2615 (2008)
12. Androidcentral. http://www.androidcentral.com/these-android-phones-can-handle-wireless-charging/
13. Tesla motors. http://www.teslamotors.com/
14. SHARP. http://www.friendsofcrc.ca/Projects/SHARP/sharp.html
15. Wireless charging market. http://www.marketsandmarkets.com/PressReleases/wireless-charging.asp
16. Kansal, A., Srivastava, M.B.: An environmental energy harvesting framework for sensor networks. In: Proceedings of IEEE ISLPED 2003, pp. 481–486
17. Kansal, A., Hsu, J., Zahedi, S., Srivastava, M.B.: Power management in energy harvesting sensor networks. ACM Trans. Embedded Comput. Syst. **6** (2007)
18. Voigt, T., Ritter, H., Schiller, J.: Utilizing solar power in wireless sensor networks. In: Proceedings of IEEE LCN 2003, pp. 416–422
19. Cammarano, A., Petrioli, C., Spenza, D.: Pro-energy: a novel energy prediction model for solar and wind energy-harvesting wireless sensor networks. In: Proceedings of IEE MASS 2012, pp. 75–83
20. Wang, W., Srinivasan, V., Chua, K.C.: Using mobile relays to prolong the lifetime of wireless sensor networks. In: Proceedings of ACM MobiCom 2005, pp. 270–283
21. Dunkels, A., Österlind, F., He, Z.: An adaptive communication architecture for wireless sensor networks. In: Proceedings of ACM SenSys 2007, pp. 335–349

22. Bhattacharya, S., Kim, H., Prabh, S., Abdelzaher, T.: Energy-conserving data placement and asynchronous multicast in wireless sensor networks. In: Proceedings of ACM MobiSys 2003, pp. 173–185
23. Tong, B., Li, Z., Wang, G., Zhang, W.: How wireless power charging technology affects sensor network deployment and routing. In: Proceedings of IEEE ICDCS 2010, pp. 438–447
24. Zhang, S., Qian, Z., Kong, F., Wu, J., Lu, S.: P^3: Joint optimization of charger placement and power allocation for wireless power transfer. In: Proceedings of IEEE INFOCOM 2015, pp. 2344–2352
25. Dai, H., Wu, X., Chen, G., Xu, L., Lin, S.: Minimizing the number of mobile chargers for large-scale wireless rechargeable sensor networks. Comput. Commun. **46**, 54–65 (2014)
26. Li, Z., Peng, Y., Zhang, W., Qiao, D.: Study of joint routing and wireless charging strategies in sensor networks. In: Proceedings of WASA 2010, pp. 125–135
27. Shu, Y., Yousefi, H., Cheng, P., Chen, J., Gu, Y., He, T., Shin, K.: Optimal velocity control for time-bounded mobile charging in wireless rechargeable sensor networks. IEEE Trans. Mobile Comput. **PP**(99), 1–1 (2015). doi:10.1109/TMC.2015.2473163
28. Wu, J.: Collaborative mobile charging and coverage. J. Comput. Sci. Technol. **29**(4), 550–561 (2014)
29. Zhang, S., Wu, J., Lu, S.: Collaborative mobile charging. IEEE Trans. Comput. **64**(3), 654–667 (2015)
30. Wang, C., Li, J., Ye, F., Yang, Y.: NETWRAP: an NDN based real-time wireless recharging framework for wireless sensor networks. IEEE Trans. Mob. Comput. **13**(6), 1283–1297 (2014). doi:10.1109/TMC.2013.2296515
31. Fu, L., He, L., Cheng, P., Gu, Y., Pan, J., Chen, J.: ESync: Energy synchronized mobile charging in rechargeable wireless sensor networks. IEEE Trans. Veh. Technol. **PP**(99), 1–1 (2015). doi:10.1109/TVT.2015.2481920
32. Xu, W., Liang, W., Lin, X., Mao, G.: Efficient scheduling of multiple mobile chargers for wireless sensor networks. IEEE Trans. Veh. Technol. **PP**(99), 1–1 (2015). doi:10.1109/TVT.2015.2496971
33. Bentham, J.: An introduction to the principles of morals and legislation. Courier Corporation (2007)
34. Vazirani, V.V.: Approximation Algorithms. Springer (2003)
35. Lee, J.M.: Introduction to Topological Manifolds. Springer (2000)