Charging Task Scheduling for Directional Wireless Charger Networks

Haipeng Dai, *Member, IEEE*, Ke Sun, Alex X. Liu, *Fellow, IEEE*, Lijun Zhang, *Member, IEEE*, Jiaqi Zheng, and Guihai Chen, *Member, IEEE*

Abstract—This paper studies the problem of cHarging tAsk Scheduling for direcTional wireless chargEr networks (HASTE), i.e., given a set of rotatable directional wireless chargers on a 2D area and a series of offline (online) charging tasks, scheduling the orientations of all the chargers with time in a centralized offline (distributed online) fashion to maximize the overall charging utility for all the tasks. We prove that HASTE is NP-hard. Then, we prove that a relaxed version of HASTE falls within the realm of maximizing a submodular function subject to a partition matroid constraint, and propose a centralized offline algorithm that achieves $(1-\rho)(1-\frac{1}{e})$ approximation ratio to address HASTE where ρ is the switching delay of chargers. Further, we propose a distributed online algorithm and prove it achieves $\frac{1}{2}(1-\rho)(1-\frac{1}{e})$ competitive ratio. We conduct simulations and field experiments on a testbed consisting of eight off-the-shelf power transmitters and 8 rechargeable sensor nodes. The results show that our distributed online algorithm achieves 92.97 percent of the optimal charging utility, and outperforms the comparison algorithms by up to 15.28 percent in terms of charging utility.

Index Terms—Charging task, scheduling, directional wireless chargers

1 Introduction

1.1 Motivation and Problem Statement

The last decade has witnessed the rapid development of Wireless Power Transfer (WPT) technology, which enjoys huge advantages such as no contact, reliable power supply, and ease of maintenance compared to traditional wired power supply technologies. WPT technology has numerous applications, including wireless identification and sensing platform (WISP) [1], wireless rechargeable sensor networks [2], electric vehicles [3], solar power satellites [4], and wireless powered drone aircraft [5], *etc.* As per the record provided by Wireless Power Consortium, an organization dedicated to promote standardization of WPT, the number of registered WPT products from its 214 member companies, including IT leaders Samsung, Philips, LG, and Huawei, has surged to 848 [6]. By a recent report, 35 percent of consumers in the United States have used WPT products [7].

Directional wireless charger network, which consists of static directional wireless chargers, is one of the critical topics for WPT technology. To begin with, it is well-known that directional charging is more energy efficient than omnidirectional charging. Unlike omnidirectional charging which broadcasts the electromagnetic waves equally in all directions regardless of the locations of the rechargeable devices, directional charging concentrates radiated energy in the directions of the rechargeable devices (i.e., energy beamforming), and thus enhances the power intensity in the intended directions [8]. For this reason, directional antennas for WPT are widely adopted in applications such as millimeter wave cellular networks [9], [10], [11], [12], wireless rechargeable sensor networks [13], and wireless charging systems adopting the simultaneous wireless information and power transfer technology [14], [15], and are also studied in [16], [17], [18]. Further, static chargers are more preferable than mobile chargers in some scenarios. First, using static chargers is a more robust and timely way to handle unexpected arrived charging tasks in an online manner, such as urgent charging requests caused by accidental energy depletion of existing sensor nodes or new nodes join, than using mobile chargers, because mobile chargers may need to travel a long distance for handling tasks. Second, static chargers can also serve as data collectors, which allows fast and efficient data collection than using mobile chargers. Third, it is more cost-efficient for some applications where, for example, sensor nodes form multiple clusters with long distance between them. Moreover, from a long term view, purchasing wireless chargers is a one-time investment and can be amortized over time, while using mobile chargers usually require much higher energy cost and human cost than maintaining static chargers, and such cost constantly accumulates over time. Fourth, there have emerged a lot of on-the-shelf products based on wireless power transfer technologies [19], [20], [21], and they offer solutions for popular applications such as charging at coffee shops, security systems, smart home, and in-vehicle charging. These applications require dedicated static chargers.

In this paper, we consider the problem of cHarging tAsk Scheduling for direcTional wireless chargEr networks (HASTE) aiming for maximizing the overall charging utility of offline/online charging tasks. We adopt the directional charging model for wireless chargers and rechargeable

• *The authors are with the Department of Computer Science and Technology, Nanjing University, Nanjing 210093, China. E-mail: {haipengdai, zlj, gchen}@nju.edu.cn, kesun@smail.nju.edu.cn, {alexliu360, jiaqi369}@gmail.com.*

Manuscript received 28 Aug. 2018; revised 5 May 2020; accepted 8 May 2020. Date of publication 25 May 2020; date of current version 1 Oct. 2021. (Corresponding author: Haipeng Dai.)
Digital Object Identifier no. 10.1109/TMC.2020.2997602

devices, which captures the characteristics of power transmitters and receivers equipped with directional antennas. In this model, the power charging area for a charger and the power receiving area for a device are modeled as sectors. A rechargeable device can be charged via wireless by a charger with non-zero power if and only if they are located in each other's covered sector. All wireless chargers can freely adjust its orientation in $[0, 2\pi)$ while rechargeable devices cannot. Moreover, a charging task initiated by a rechargeable device consists of five elements: the position and orientation of its associated device, the release time and end time of the task, and its required charging energy. To evaluate the effectiveness of wireless charging for a task, we define the task's charging utility as a linear and bounded function with its harvested energy from its release time to its end time.

With these models, we consider two scenarios for charging task scheduling, i.e., offline and online. In the offline scenario, information for all charging tasks is known a priori, and thereby the scheduling policies for all chargers at any moment can be determined beforehand. To accommodate practical concerns, we assume that each charger needs an amount of time for switching its orientation, which we call *switching delay*. In the online scenario, charging tasks stochastically arrive, and chargers reschedule their orientations in realtime. Nevertheless, in addition to *switching delay*, each charger needs an additional amount of time for recomputing the scheduling policies with negotiating with neighboring chargers, which we call *rescheduling delay*. To avoid global management effort and reduce update cost, we desire a distributed and local algorithm which is scalable with network size. For both scenarios, we want to dynamically schedule the orientations of chargers as time goes on such that the overall weighted charging utility for all charging tasks is maximized. Moreover, we stress that chargers can be either in the working mode for the offline scenario or in that for the online scenario, but cannot switch between these two different statuses. To sum up, we state our problem HASTE as follows. Given a set of rotatable directional wireless chargers on a 2D area and a series of offline (online) charging tasks, scheduling the orientations of all the chargers with time in a centralized offline (distributed online) fashion to maximize the overall charging utility for all the tasks.

1.2 Prior Art

On one hand, there exist numerous literatures [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37] studying on the mobile charging case where one single or multiple chargers travel in a field to charge wireless rechargeable devices to guarantee their normal working. They are fundamentally different from ours as we consider static chargers.

On the other hand, the other works consider wireless charger networks consisted of static wireless chargers, but nearly none of them investigate charging task scheduling. In particular, most of them focus on scheduling issues in coarse granularity rather than task levels, such as those overlooking the harmful effect of high electromagnetic radiation (EMR) [16], [17], [38], [39], [40], [41] and those taking the EMR safety into consideration [42], [43], [44], [45], [46], [47], [48]. To the best of our knowledge, there is only one work [49], [50] investigate wireless charging task scheduling issue for omnidirectional wireless chargers in offline scenarios, which are fundamentally different from our paper. In the conference version of this paper [51], we initiated the first study on scheduling wireless charging tasks for directional wireless chargers and designing online algorithms.

1.3 Key Technical Challenges

We are faced with three major challenges to address HASTE. The first challenge is that HASTE is non-linear and is NP-hard. HASTE is nonlinear because that the orientation of chargers can be freely scheduled; a task can be either covered by a charger and have a certain constant power increment or not with no power increment, which has the flavor of 0-1 integer programming; the charging utility function is linear but bounded, let alone that we extend our results to the case where the utility function is a general concave function. In addition, by reducing from the classical NP-hard separate assignment problem, we prove that HASTE is NP-hard.

The second challenge is how to design an efficient centralized offline algorithm for HASTE in the offline scenario while considering the switching delay of chargers. The switching delay happens if and only if a charger's next intended orientation is different from its current orientation, which implies that the switching delay as well as its caused performance loss is history-dependent. Moreover, the performance loss is difficult to evaluate as there are potentially multiple tasks are affected by a charger's switching delay, and the charging utility function for tasks is non-linear.

The third challenge is how to design an efficient distributed online algorithm for HASTE in the online scenario where all chargers are asynchronous and the rescheduling delay needs to be considered. To the best of our knowledge, there are neither existing distributed online algorithms directly applicable to our problem even when the rescheduling delay is omitted, nor existing online algorithms that deal with the case in our considered scenario with rescheduling delay being concerned for which the response is delayed and the algorithm is not truly "online".

1.4 Proposed Approach

To address the first challenge, we propose that rather than considering all possible orientations in $[0, 2\pi)$ for chargers, we can safely consider a limited number of orientations for them without causing performance loss, and therefore, extract the so-called "dominant task sets" as the corresponding sets of covered tasks. Then, we neglect the switching delay for wireless chargers, and thus reformulate the original continuous optimization problem into a discrete optimization problem HASTE-R. Further, we prove that the reformulated problem is exactly a problem of maximizing a submodular function subject to a partition matroid constraint, which greatly facilitates approximation algorithm design.

To address the second challenge, based on the theoretical results obtained by addressing the first challenge, we can either use a simple greedy algorithm that achieves $\frac{1}{2}$

approximation ratio [52] or a randomized algorithm with the optimal approximation guarantee, namely, $1-\frac{1}{e}$ approximation ratio [53]. Nevertheless, as the former is not good enough and the latter is too computationally demanding, we tailor the TABULARGREEDY algorithm proposed in [54], [55] to address HASTE-R as it can achieve an approximation ratio between $\frac{1}{2}$ and $1-\frac{1}{e}$ ($1-\frac{1}{e}$ as default in our setting) depending on the value of a control parameter and resulting in different time complexity. Further, to bound the performance loss of switching delay, we exploit the concavity of the utility function and consider all the caused performance loss for all impacted tasks in the worst case, and prove that the switching delay introduces a constant factor of $1-\rho$ in the ultimate achieved approximation ratio, i.e., $(1-\rho)(1-\frac{1}{e})$, of the proposed algorithm, where ρ is the switching delay.

To address the third challenge, we propose a distributed online algorithm based on the proposed centralized offline algorithm to HASTE. We prove that if the rescheduling delay is neglected, as long as the local executions of a charger and its neighbors are in order and repeat regularly with time, the achieved global charging utility is the same as that of the centralized offline algorithm. Further, by leveraging the concavity of the utility function and the submodularity of the objective function in HASTE, we bound the performance loss of scheduling delay, and prove that our distributed online algorithm achieves $\frac{1}{2}(1-\rho)(1-\frac{1}{e})$ competitive ratio.

1.5 Evaluation Results

We conducted simulations and field experiments to evaluate our proposed algorithms. Our simulation results show that our proposed distributed online algorithm can achieve 92.97 percent of the optimal charging utility which corroborates our theoretical findings, outperform the other two comparison algorithms by 10.96 percent. We implemented our algorithms on a testbed consisting of 8 off-the-shelf TX91501 power transmitters produced by [19] and 8 rechargeable sensor nodes associated with 8 charging tasks. Our experimental results show that our distributed online algorithm outperforms the comparison algorithms by up to 15.28 percent on average, and 29.63 percent at most.

2 Related Work

In this section, we briefly review related works regarding wireless charging.

First, there exist some literatures focus on mobile charging scenarios where one single or multiple chargers travel in a field to charge rechargeable devices deployed there to make them work perpetually, which are fundamentally different from ours. [22], [23], [24], [25], [26] study the charging efficiency issues of wireless chargers, e.g., Zhang *et al.* presented an optimal scheme for multiple mobile chargers to charge a linear WSN while the ratio of truly charged energy to wasted energy is maximized. [27], [28], [29] concentrate on reducing the service delay of mobile chargers, e.g., Fu *et al.* considered the problem of minimizing the overall charging delay of a single mobile charger by planning its charging route and charging strategy [27]. [30], [31], [32], [33], [34], [35], [36], [37] pay attention to the overall network performance such as data routing, event monitoring, data collection, and task assignment. For instance, Shi *et al.* proposed to use a single mobile charger to improve data collection performance and the charger's working time in a charging time period [30], [31]. We refer readers to survey [56] for more related works.

Second, the other works are dedicated to wireless charger networks consisted of static wireless chargers, but nearly none of them consider charging task scheduling. First, most of them study scheduling issues in coarse granularity rather than task levels. On one hand, some works (e.g., [16], [17], [38], [39], [40], [41]) overlook the detrimental effect of the electromagnetic radiation to human health. For instance, He *et al.* considered the triangular deployment problem of wireless chargers [38]. They attempted to minimize the number of chargers while rechargeable tags can receive sufficient power. In addition, we first proposed the directional charging problem based on empirical experimental results, and investigated the ominidirectional charging problem using directional chargers in [16], the wireless charger placement problem for directional charging in [17], [39], [40], [41]. On the other hand, other literatures [42], [43], [44], [45], [46], [47], [48] take the EMR safety into consideration, and guarantee that the EMR intensity at any point in the area does not exceed a predefined EMR threshold. For instance, we presented and studied how to schedule non-adjustable chargers [42], [43] and adjustable chargers [44], [45] to maximize the charging utility for chargers under the EMR safety constraint. Nikoletseas *et al.* [46] considered more practical constraints such as the energy limitations of chargers and devices, the non-linear constraints in the time domain, and their goal is to optimize the amount of energy transferred from chargers to devices and truly utilized. Moreover, we reported a wireless charger placement scheme that ensures EMR safety in [47]. Second, to the best of our knowledge, there is only one work [49], [50] that study the wireless charging task scheduling. Nevertheless, [49], [50] consider omnidirectional wireless chargers whose charging power is adjustable and focus on offline scenarios, which are fundamentally different from our paper. Moreover, we launched the first study on scheduling wireless charging tasks for directional wireless chargers and designing online algorithms in the conference version of this paper [51].

3 Problem Formulation

3.1 Preliminaries

Suppose there is a set of directional wireless chargers $S=\{s_1,\ldots,s_n\}$ located in a 2D plane Ω, which can continuously rotate with orientation angle within $[0\ 2\pi)$. Suppose there are also some rechargeable devices located in Ω, which either keep static or dynamically join or leave the wireless charger network. These rechargeable devices launch (wireless) charging tasks and sending them to wireless chargers now and then, and the chargers accordingly schedule their orientations to serve the tasks. Formally, charging tasks are defined by a five-tuple $\mathcal{T}_j=<o_j,\phi_j,t_r^j,t_e^j,E_j>$ where o_j denotes the position of the rechargeable device that raises the task, ϕ_j is the orientation of the device, t_r^j and t_e^j are the release time and end time of the task, and E_j is required charging energy. We adopt a discrete time model for which the time is divided into multiple

TABLE 1
Notations and Symbols Used in This Paper

Symbol	Description
s_i	The i_{th} directional wireless charger, or its position
n	Number of directional wireless chargers
θ_i ($\theta_i(t)$)	Orientation of charger s_i (its function with time t)
$\theta_{i,k}$	The value of $\theta_i(t)$ at the k_{th} time slot if charger s_i is not switching
$\mathcal{T}_j$	The j_{th} charging task
o_j	Position of the rechargeable device that raises charging task $\mathcal{T}_j$, or the j_{th} rechargeable device
ϕ_j	Orientation of the rechargeable device that raises charging task $\mathcal{T}_j$, or the orientation of device o_j
t_r^j (t_e^j)	Release time (end time) of charging task $\mathcal{T}_j$
E_j	Required charging energy of charging task $\mathcal{T}_j$
m	Number of charging tasks
A_s	Charging angle of chargers
A_o	Receiving angle of devices
T_s	Duration of a time slot
$P_r(.)$	Charging power function
α, β	Constants in the charging model
D	Radius of charging/receiving area
ρ	Switching delay
τ	Rescheduling delay
$\mathcal{U}(.)$	Charging utility function
w_j	Weight of charging task $\mathcal{T}_j$
$\boldsymbol{\mathcal{T}}_i$	Set of charging tasks that cover charger s_i
Γ_i (Γ_i^p)	Set of dominant task sets for charger s_i (the p_{th} dominant task set in Γ_i)
$\Gamma_{i,k}$ ($\Gamma_{i,k}^p$)	Set of dominant task sets for charger s_i at the k_{th} time slot (the p_{th} dominant task set in $\Gamma_{i,k}$)
K	Number of considered time slots for all tasks
C	Number of colors
$N(s_i)$	Neighbors of charger s_i (two chargers are neighbors to each other if and only if they cover at least one charging task in common)
K_i	Number of considered time slots for all tasks observed by charger s_i

slots with uniform duration T_s. For simplicity, we assume that t_r^j is exactly at the beginning of a time slot while t_e^j is at the end of a time slot. We summarize the notations used in this paper in Table 1.

We adopt the general and practical directional charging model proposed in [16], [17], [18]. As Fig. 1 shows, a charger s_i with working orientation denoted by vector $\overrightarrow{r_{\theta_i}}$ can only charge devices in a *charging area* in the shape of a sector with *charging angle* A_s and radius D. A rechargeable device o_j with orientation denoted by vector $\overrightarrow{r_{\phi_j}}$ can only receive non-zero power in a *receiving area* in the shape of a sector with *receiving angle* A_o and radius D. The charging power from s_i to o_j is given by

$$P_r(s_i, \theta_i, o_j, \phi_j) = \begin{cases} \frac{\alpha}{(\|s_i o_j\| + \beta)^2}, & 0 \leq \|s_i o_j\| \leq D, \\ & \overrightarrow{s_i o_j} \cdot \overrightarrow{r_{\theta_i}} - \|s_i o_j\| cos(A_s/2) \geq 0, \\ & and \quad \overrightarrow{o_j s_i} \cdot \overrightarrow{r_{\phi_j}} - \|o_j s_i\| cos(A_o/2) \geq 0. \\ 0, & otherwise \end{cases},$$

where α and β are two known constants determined by hardware parameters of chargers as well as surrounding environment [16], [17], [18], $\|s_i o_j\|$ is the distance between s_i and o_j, A_s and A_o are respectively the charging angle of chargers and the receiving angle of devices, $\overrightarrow{r_{\theta_i}}$ and $\overrightarrow{r_{\phi_j}}$ are respectively the unit vectors denoting the orientations of the charger and the device. Further, if a device o_j is covered by more than one directional wireless chargers, its received power is the sum of the received power from all chargers [16], [17]. Note that there is another directional charging model proposed in [57], which is more practical as it considers the anisotropic energy receiving property of rechargeable sensors. We plane to consider it in our future work.

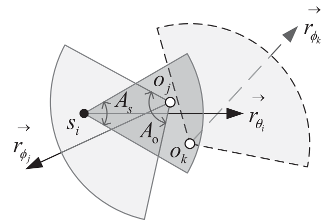

Fig. 1. Directional charging model (o_j can receive power from s_i while o_k cannot).

A charger can either keep its orientation unchanged during the same time slot, or switch its orientation in the starting ρ ($0 < \rho < 1$) portion of a time slot, which we call *switching delay*, and keep static in the rest $1 - \rho$ portion of the time slot. We argue that this assumption makes sense because typically a charging task can last up to tens of minutes or even more than an hour, the duration of time slots can be set to a few minutes, and the switching time for commercial rotatable heads or cradles [58] on which the chargers are mounted or soft switching of smart antennas of chargers [59], [60] is commonly a few seconds or even shorter. We assume that a charger stops emitting power during its switching. For convenience of exposition, we define $\theta_i = \Phi$ for a charger during its switching process, and further define $P_r(s_i, \Phi, o_j, \phi_j) = 0$. In the offline case, we assume the information for all charging tasks are known a prior, then the scheduling policies for all time slots for each charger are determined beforehand. In the online case, we assume the charging tasks stochastically arrive, and chargers recompute their scheduling policies in an on-the-fly fashion. Especially, we assume each charger needs τ ($\tau \in \mathbb{Z}_+$) number of time slots, which we name as *rescheduling delay*, for negotiation with neighboring chargers and computation to update its future scheduling policies, and then, if necessary, starts switching with a delay of ρ time slot. Typically, the rescheduling delay is expected to be much less than the duration of charging tasks. In this paper, we assume the latter is at least two times that of the former, i.e., $t_e^j - t_r^j \geq 2\tau T_s$ for any task $\mathcal{T}_j$, where T_s is the duration of a time slot.

3.2 Charging Utility Model

We adopt a linear and bounded charging utility model for harvested energy for a task, which is similar to the charging utility model for received power proposed in [17]. That is, the charging utility for a task is first proportional to the harvested energy of its associated device, and then reaches a constant if the harvested energy exceeds a predetermined threshold, i.e.,

$$\mathcal{U}(x) = \begin{cases} \frac{1}{E_j} \cdot x, & x \leq E_j \\ 1, & x > E_j \end{cases}. \tag{1}$$

where E_j is the required charging energy of charging task $\mathcal{T}_j$.

3.3 Problem Formulation and Hardness Analysis

Let $\theta_i(t)$ ($\theta_i : \mathbb{R}_{\geq 0} \mapsto \{[0\ 2\pi) \cup \Phi\}$) be the function of orientation for charger s_i with time t. Suppose the value of $\theta_i(t)$ at the k_{th} time slot is $\theta_{i,k}$ if charger s_i is not switching; otherwise, $\theta_i(t)$ is set to Φ and the charging power of s_i is zero. Then, for a charging task $\mathcal{T}_j$, its harvested power at time t is given by $\sum_{i=1}^{n} P_r(s_i, \theta_i(t), o_j, \phi_j)$, and its aggregate harvested energy during its whole life is $\int_{t_r^j}^{t_e^j} \sum_{i=1}^{n} P_r(s_i, \theta_i(t), o_j, \phi_j)dt$. And the overall (weighted) charging utility is $\sum_{j=1}^{m} w_j \cdot \mathcal{U}(\int_{t_r^j}^{t_e^j} \sum_{i=1}^{n} P_r(s_i, \theta_i(t), o_j, \phi_j)dt)$ where w_j is the weight of charging task $\mathcal{T}_j$.

Our task is to determine the decision variables $\theta_{i,k}$ defined in $\theta_i(t)$ for all the chargers so that the overall charging utility is maximized. With all above, we define the problem of cHarging tAsk Scheduling for direcTional wireless chargEr networks as follows.

$$(\mathbf{P1}) \quad \max_{\theta_{i,k}} \quad \overline{\mathcal{U}} = \sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\int_{t_r^j}^{t_e^j} \sum_{i=1}^{n} P_r(s_i, \theta_i(t), o_j, \phi_j)dt\right)$$

$$s.t. \quad \theta_i(t) = \begin{cases} \begin{cases} \Phi,\ kT_s < t \leq (k+\rho)T_s \\ \theta_{i,k}, (k+\rho)T_s < t \leq (k+1)T_s \end{cases} & , \theta_{i,k} \neq \theta_{i,k-1} \\ \theta_{i,k}, kT_s < t(k+1)T_s, & otherwise \end{cases}$$

$$where\ k \in \mathbf{Z}_0^+,\ and\ \theta_i(0) = \Phi$$

$$0 \leq \theta_{i,k} < 2\pi.$$

The following theorem shows the complexity of HASTE.

Theorem 3.1. *HASTE is NP-hard.*

Proof. Due to space limit, we only sketch the proof here. Suppose $\rho \to 0$, $t_r^j = 0$ and $t_e^j = T_s$ for all charging tasks, which means each task simply occupies the first time slot and we only need to consider one round scheduling in this time slot. Moreover, suppose the required charging energy for each task E_j is so small that as long as a task is covered by a charger, it certainly obtains an amount of energy greater then E_j and therefore achieves a charging utility of w_j in the overall charging utility. Besides, though the orientation of chargers can be freely chosen in $[0\ 2\pi)$, its covered sets of charging tasks can be enumerated in a fixed number of steps and are limited, as we will see in Algorithm 1. Consequently, with the above settings, our problem changes to choosing the orientation for each charger among its candidate choices such that the overall charging utility of all tasks is maximized. We can prove this simplified problem is NP-hard by reducing from the classical NP-hard separate assignment problem [61], which is defined as follows: given a set of bins and a set of items to pack in each bin, a value for assigning item j to bin i, and a separate packing constraint for each bin, i.e., for bin i, a family I_i of subsets of items that fit in bin i, packing items into bins to maximize the aggregate value. Here we can regard each charger as a bin, each task as an item, each set of covered tasks for a candidate orientation of charger s_i as a subset in the family I_i for bin i, the achieved utility of a task as the value for assigning this item to a bin, and therefore, we can reduce any instance of the separate assignment problem to the considered simplified problem. As the separate assignment problem is NP-hard [61], HASTE is also NP-hard. □

4 Problem Reformulation

In this section, considering the complexity of the formulation **P1** of HASTE, we reformulate HASTE to make it tractable. In particular, we first propose a dominant task sets extraction algorithm for chargers to reduce the continuous solution space for orientations of chargers to a discrete one with limited choices. Then, we consider a relaxed version of HASTE, i.e., HASTE-R, and prove it falls into the realm of maximizing a submodular function subject to a partition matroid constraint, which assists the further algorithm design.

Algorithm 1. Dominant Task Sets Extraction

Input: The wireless charger s_i, all charging tasks $\{\mathcal{T}_j\}_{j=1}^{m}$
Output: All dominant task sets
1: Find the subset of charging tasks in $\{\mathcal{T}_j\}_{j=1}^{m}$ that cover s_i, say $\mathcal{T}_i$;
2: Initialize the orientation of the charger to 0;
3: Rotate the charger anticlockwise to cover the tasks in $\mathcal{T}_i$ one by one until there is some covered task is going to be uncovered. During the rotating process, if the rotated angle is larger than 2π, then terminate;
4: Add the current covered set of tasks to the collection of dominant task sets;
5: Rotate the charger anticlockwise until a new task in $\mathcal{T}_i$ is included in the covered set. During the rotating process, if the rotated angle is larger than 2π, then terminate. If not, goto Line 3.

4.1 Extraction of Dominant Task Sets

Though each charger can continuously rotate within $[0\ 2\pi)$, we do NOT need to consider all possible orientations. Instead, we only need to care about the possible sets of covered tasks, whose number is obviously limited for any given charger. Further, among these sets we only need to consider the following specific ones.

Definition 4.1 (dominant task set). *Given a set of tasks $\mathcal{T}_i^1$ covered by a charger s_i with some orientation, if there doesn't exist another set of tasks $\mathcal{T}_i^2$ covered by s_i with some other orientation such that $\mathcal{T}_i^1 \subset \mathcal{T}_i^2$, then $\mathcal{T}_i^1$ is a* dominant task set.

We describe our algorithm for extracting dominant task sets in Algorithm 1. Basically, the considered charger rotates for 2π and extracts the dominant task sets one by one. We use a toy example for illustration. As shown in Fig. 2a, the charger first covers task $\mathcal{T}_1$, then rotates to cover tasks $\mathcal{T}_2$ and $\mathcal{T}_3$ sequentially. Further, $\mathcal{T}_4$ cannot be added in the current covered set as otherwise $\{\mathcal{T}_1, \mathcal{T}_2\}$ will be missed, and therefore, $\{\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3\}$ is a dominant task set. Then, the charger continues to cover $\mathcal{T}_4$ by removing $\mathcal{T}_1$ and $\mathcal{T}_2$ from the current set, as shown in Fig. 2b. Similarly, as $\mathcal{T}_5$ cannot be covered by the charger without missing $\mathcal{T}_3$, $\{\mathcal{T}_3, \mathcal{T}_4\}$ is added as a dominant task set. Algorithm 1 proceeds until the charger rotates for 2π, as depicted in Figs. 2c and 2d.

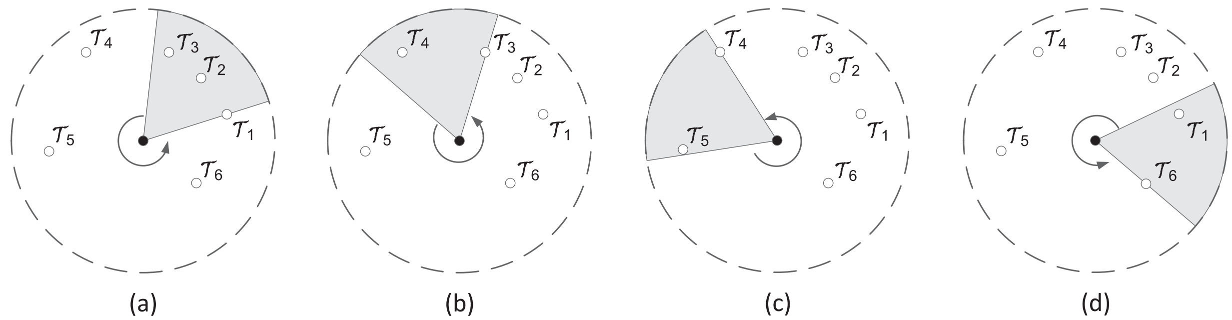

Fig. 2. A toy example of dominant task sets extraction.

After all, the obtained dominant task sets are $\{\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3\}, \{\mathcal{T}_3, \mathcal{T}_4\}, \{\mathcal{T}_4, \mathcal{T}_5\}$ and $\{\mathcal{T}_6, \mathcal{T}_1\}$.

4.2 Problem Relaxation and Reformulation

As the switching delay is hard to be analyzed for optimization, we first consider a relaxed version of HASTE, HASTE-R for short, by neglecting the switching delay of all chargers, and then analyze HASTE. We will bound the performance loss for the relaxation in our proposed algorithms.

Suppose the obtained set of dominant task sets for charger s_i is Γ_i, the p_{th} dominant task set in Γ_i is Γ_i^p. Let $x_{i,k}^p$ be a binary indicator denoting whether the p_{th} dominant task set in Γ_i in the k_{th} time slot is selected or not. For convenience of expression, we define

$$P_r(s_i, o_j) = \begin{cases} \frac{\alpha}{(||s_i o_j|| + \beta)^2}, & 0 \le ||s_i o_j|| \le D, \\ 0, & otherwise. \end{cases}$$

Moreover, we abuse the notation slightly by defining $\Gamma_i^p \ni o_j$ as $\exists \mathcal{T}_{j'} \in \Gamma_i^p | \mathcal{T}_{j'}.o_{j'} = o_j$. That is, there exists a charing task $\mathcal{T}_{j'}$ in Γ_i^p and its associated position of rechargeable device is o_j. Then, the problem HASTE-R can be formulated as

$$(\mathbf{RP1})$$

$$\max_{x_{i,k}^p} \quad \overline{\mathcal{U}}_R = \sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\sum_{k=t_r^j/T_s+1}^{t_e^j/T_s} \sum_{\substack{\Gamma_i^p \ni o_j, \\ i\in[n],\, p\in[|\Gamma_i|]}} x_{i,k}^p P_r(s_i, o_j) T_s \right)$$

$$s.t. \quad \sum_{p=1}^{|\Gamma_i|} x_{i,k}^p = 1, \ (x_{i,k}^p \in \{0,1\}),$$

where $x_{i,k}^p$s are the decision variables, Γ_i^p is the p_{th} dominant task set in Γ_i.

Clearly, **RP1** is a combinatorial optimization problem. To facilitate further analysis, we first give the following definitions.

Definition 4.2 [62] (submodular set function). *Let S be a finite ground set. A real-valued set function $f : 2^S \to \mathbb{R}$ is normalized, monotonic and submodular if and only if it satisfies the following conditions, respectively:*

1) $f(\emptyset) = 0$;
2) $f(A \cup \{e\}) - f(A) \ge 0$ *for any* $A \subseteq S$ *and* $e \in S \backslash A$;
3) $f(A \cup \{e\}) - f(A) \ge f(B \cup \{e\}) - f(B)$ *for any* $A \subseteq B \subseteq S$ *and* $e \in S \backslash B$.

Definition 4.3 [62] (matroid). *A* matroid $\mathcal{M}$ *is a strategy* $\mathcal{M} = (S, L)$ *where S is a finite ground set, $L \subseteq 2^S$ is a collection of independent sets, such that*

1) $\emptyset \in L$;
2) *if* $X \subseteq Y \in L$, *then* $X \in L$;
3) *if* $X, Y \in L$, *and* $|X| < |Y|$, *then* $\exists y \in Y \backslash X$, $X \cup \{y\} \in L$.

Definition 4.4 [62] (partition matroid). *Given* $S = \bigcup_{i=1}^{k} S_i'$ *is the disjoint union of k sets,* $l_1, l_2, \ldots, l_k$ *are positive integers, a* partition matroid $\mathcal{M} = (S, \mathcal{I})$ *is a matroid where* $\mathcal{I} = \{X \subset S : |X \cap S_i'| \le l_i \text{ for } i \in [k]\}$.

We will show that the problem **RP1** fits perfectly in the realm of maximizing a monotone submodular function subject to a partition matroid. First, we define $\Gamma_{i,k} = \Gamma_i$ $(k \in [K])$ as the set of dominant task sets for charger s_i at the k_{th} time slot, where K is the total number of time slots and the notation $[n] = \{1, 2, \ldots, n\}$. Then, we define $\Theta_{i,k}^p$ as the corresponding scheduling policy for $\Gamma_{i,k}^p$, i.e., the orientation that covers $\Gamma_{i,k}^p = \Gamma_i^p$, for charger s_i at the k_{th} time slot, define $\mathbf{\Theta}_{i,k} = \{\mathbf{\Theta}_{i,k}^p\}_{p\in[|\Gamma_{i,k}|]}$ as the set of scheduling policies for s_i at the k_{th} time slot, and define a ground set of all scheduling policies $S = \{\mathbf{\Theta}_{i,k}\}_{i\in[n],k\in[K]}$. Further, we define the scheduling policies for all chargers at all K time slots as X, which is subject to $|X \cap \mathbf{\Theta}_{i,k}| \le 1$. Therefore, as $\mathbf{\Theta}_{i,k}$s are disjoint sets, we write the independent sets as

$$\mathcal{I} = \{X \subseteq S : |X \cap \mathbf{\Theta}_{i,k}| \le 1 \text{ for } i \in [n], k \in [K]\}. \tag{2}$$

Besides, it can be easily proved that $\mathcal{M} = \{S, \mathcal{I}\}$ is a matroid by verifying the three properties proposed in Definition 4.3. Thus we have the following lemma.

Lemma 4.1. *The constraint in the scheduling problem* **RP1** *can be written as a partition matroid on the ground set S.*

Accordingly, problem **RP1** can be rewritten as

$$(\mathbf{RP2})$$

$$\max_{X} f(X) = \sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\sum_{k=t_r^j/T_s+1}^{t_e^j/T_s} \sum_{\substack{\Gamma_{i,k}^p \ni o_j,\, i\in[n], \\ p\in\{p|\Theta_{i,k}^p = X\cap\mathbf{\Theta}_{i,k}\}}} P_r(s_i, o_j) T_s \right)$$

$$s.t. \quad X \in \mathcal{I},$$

where X is the decision variable, and $f(X)$ $(f : 2^S \to \mathbb{R}_{\ge 0})$ is the objective function. Note that we abuse the notation slightly, and here $\Gamma_{i,k}^p \ni o_j$ means $\exists \mathcal{T}_{j'} \in \Gamma_{i,k}^p | \mathcal{T}_{j'}.o_{j'} = o_j$.

For **RP2**, we have the following critical lemma.

Lemma 4.2. *The objective function $f(X)$ in* **RP2** *is a monotone submodular set function.*

Proof. By Definition 4.2, we need to check whether $f(X)$ satisfies the three listed conditions.

First, when there are no active scheduling policies, i.e., $X = \varnothing$, the received energy for any task is zero, then we have $f(X) = 0$.

Second, let A be a set of scheduling strategies in S and $e \in S \backslash A$. For simplicity, define

$$g(X, j) = \mathcal{U}\left(\sum_{k=t_r^j/T_s+1}^{t_e^j/T_s} \sum_{\substack{\Gamma_{i,k}^p \ni o_j,\, i\in[n],\\ p\in\{p|\Theta_{i,k}^p = X\cap\boldsymbol{\Theta}_{i,k}\}}} P_r(s_i, o_j) T_s \right), \tag{3}$$

as the achieved utility for task $\mathcal{T}_j$. It is easy to see that $g(A\cup\{e\}, j) - g(A, j) \geq 0$ because there are possibly more chargers cover task $\mathcal{T}_j$ as all possible dominant task sets that cover $\mathcal{T}_j$, i.e., $\Gamma_{i,k}^p$ $(i \in [n],\ p \in \{p|\Theta_{i,k}^p = A \cap \boldsymbol{\Theta}_{i,k}\})$ would be enlarged as A becomes $A \cup \{e\}$, and the utility function $\mathcal{U}(.)$ is non-decreasing. Hence we have

$$f(A\cup\{e\}) - f(A) = \sum_{j=1}^{m} w_j \cdot [g(A\cup\{e\}, j) - g(A, j)] \geq 0. \tag{4}$$

Third, let A and B be two sets such that $A \subseteq B \subseteq S$ and element $e \in S\backslash B$. On one hand, it is easy to see that

$$\begin{aligned} &\sum_{k=\frac{t_r^j}{T_s}+1}^{\frac{t_e^j}{T_s}} \sum_{\substack{\Gamma_{i,k}^p\ni o_j,\\ i\in[n],\, p\in P_1}} P_r(s_i, o_j)T_s - \sum_{k=\frac{t_r^j}{T_s}+1}^{\frac{t_e^j}{T_s}} \sum_{\substack{\Gamma_{i,k}^p\ni o_j,\\ i\in[n],\, p\in P_2}} P_r(s_i, o_j)T_s \\ =& \sum_{k=\frac{t_r^j}{T_s}+1}^{\frac{t_e^j}{T_s}} \sum_{\substack{\Gamma_{i,k}^p\ni o_j,\\ i\in[n],\, p\in P_3}} P_r(s_i, o_j)T_s - \sum_{k=\frac{t_r^j}{T_s}+1}^{\frac{t_e^j}{T_s}} \sum_{\substack{\Gamma_{i,k}^p\ni o_j,\\ i\in[n],\, p\in P_4}} P_r(s_i, o_j)T_s, \end{aligned} \tag{5}$$

where $P_1 = \{p|\Theta_{i,k}^p = \{A\cup e\}\cap\boldsymbol{\Theta}_{i,k}\}$, $P_2 = \{p|\Theta_{i,k}^p = A\cap \boldsymbol{\Theta}_{i,k}\}$, $P_3 = \{p|\Theta_{i,k}^p = \{B\cup e\}\cap\boldsymbol{\Theta}_{i,k}\}$, and $P_4 = \{p|\Theta_{i,k}^p = B\cap\boldsymbol{\Theta}_{i,k}\}$. On the other hand, it is clear that

$$(\mathcal{U}(x_1+\Delta x) - \mathcal{U}(x_1)) - (\mathcal{U}(x_2+\Delta x) - \mathcal{U}(x_2)) \geq 0, \tag{6}$$

for any $x_2 \geq x_1 \geq 0$ and $\Delta x \geq 0$ due to the concavity of the charging utility function $\mathcal{U}(.)$.

Consequently, we have $[g(A\cup\{e\}, j) - g(A, i, q)] - [g(B\cup\{e\}, j) - g(B, j)] \geq 0$, and therefore,

$$\begin{aligned} &[f(A\cup\{e\}) - f(A)] - [f(B\cup\{e\}) - f(B)] \\ =&\sum_{j=1}^{m} w_j \cdot \{[g(A\cup\{e\}, j) - g(A, j)] - [g(B\cup\{e\}, j) - g(B, j)]\} \\ \geq& 0. \end{aligned} \tag{7}$$

In summary, we conclude that $f(X)$ is a monotone submodular set function. This completes the proof. □

5 Centralized Offline Algorithm

In this section, we propose a centralized offline algorithm to address HASTE in the offline scenario. We note that in this case, the information for all charging tasks is known beforehand, and thereby the scheduling policies for all chargers at any time can be determined a priori.

5.1 Algorithm Description

After proved that HASTE-R is a problem of maximizing a submodular function under a partition matroid constraint, we can resort to existing schemes to address HASTE-R. For example, we can use a simple greedy algorithm to find a solution that achieves $\frac{1}{2}$ approximation ratio according to the classical results presented in [52]. Moreover, [53] proposes a randomized algorithm with optimal approximation guarantees, namely, $1-\frac{1}{e}$ approximation ratio. Nevertheless, it is too computationally demanding to practically implement. In this paper, we tailor the TABULARGREEDY algorithm proposed in [54], [55] to address HASTE-R as it can achieve an approximation ratio between $\frac{1}{2}$ and $1-\frac{1}{e}$ depending on the value of a control parameter and resulting in different time complexity, which provides flexibility in practical applications.

Algorithm 2. Centralized Offline Algorithm to HASTE

Input: Integer C, set of scheduling policies $\boldsymbol{\Theta}_{i,k}$ for charger s_i $(i\in[n], k\in[K])$, objective function $f(.)$
Output: Scheduling policies for all chargers X

$Q \leftarrow \varnothing$;
for *all* $c \in [C]$ **do**
  for *all* $i \in [n]$, $k \in [K]$ **do**
    $e_{i,k,c} \leftarrow \arg\max_{x\in\boldsymbol{\Theta}_{i,k}\times\{c\}} \mathbb{F}(Q+x)$;
    $Q \leftarrow Q \cup e_{i,k,c}$;
for *all* $i \in [n]$, $k \in [K]$ **do**
  Choose $c_{i,k}$ uniformly at random from $[C]$;
$X \leftarrow \texttt{sample}_{\vec{c}}(Q)$, where $\vec{c} = (c_{1,1}, \ldots, c_{n,1}, \ldots, c_{1,K}, \ldots, c_{n,K})$.
return X

We first propose some useful concepts in our context, which also capture the essential elements in the TABULARGREEDY algorithm, to facilitate understanding our algorithm.

- *S-C tuple*. An S-C tuple is a tuple of a scheduling policy for a charger at a time slot and a color from a palette $[C]$ of C colors (note that here color and palette have no concrete meaning, and they are only used to assist sampling). A set $Q \subseteq S\times[C]$ consists of S-C tuples which can be regarded as labeling each scheduling policy for a charger with one or more colors.
- *S-C tuple sampling function*. We associate with each partition $\boldsymbol{\Theta}_{i,k}$ a color $c_{i,k}$. For any set $Q\subseteq S\times[C]$ and vector $\vec{c} = (c_{1,1}, \ldots, c_{n,1}, \ldots, c_{1,K}, \ldots, c_{n,K})$, we define S-C tuple sampling function as

$$\texttt{sample}_{\vec{c}}(Q) = \bigcup_{i\in[n],k\in[K]} \{x\in\boldsymbol{\Theta}_{i,k} : (x, c_{i,k})\in Q\}. \tag{8}$$

  In other words, $\texttt{sample}_{\vec{c}}(Q)$ returns a set containing each item x that is exactly labeled with the color $c_{i,k}$ assigned by $\vec{c}$ to the partition $\boldsymbol{\Theta}_{i,k}$ that contains x.
- *Expected charging utility function after S-C tuple sampling*. It is defined as $\mathbb{F}(Q) = \mathbb{E}(f(\texttt{sample}_{\vec{c}}(Q)))$ as the

expected value of $f(\texttt{sample}_{\vec{c}}(Q))$ when each color $c_{i,k}$ in $\vec{c}$ is selected uniformly at random from $[C]$.

We present our centralized offline algorithm in Algorithm 2. We can see that at each step in the two-level loop, Algorithm 2 greedily optimizes $\mathbb{F}(Q)$.

5.2 Theoretical Analysis

Following Theorem 2 in [55], we have the following lemma.

Lemma 5.1. *Algorithm 2 achieves* $1-(1-\frac{1}{C})^C - \binom{nK}{2}C^{-1}$ *approximation ratio for HASTE-R.*

Obviously, when $C \rightarrow +\infty$, the approximation ratio approaches $1-\frac{1}{e}$. Further, when $C=1$, there is only one possible choice for $\vec{c}$, and TABULARGREEDY is simply the locally greedy algorithm that achieves $\frac{1}{2}$ approximation ratio [52]. For simplicity, we assume $C \rightarrow +\infty$ and say Algorithm 2 achieves $1-\frac{1}{e}$ approximation ratio for HASTE-R.

Theorem 5.1. *Algorithm 2 achieves* $(1-\rho)(1-\frac{1}{e})$ *approximation ratio for HASTE, and its time complexity is* $O(C(nmK)^2)$ *where* ρ *is the switching delay,* C*,* n*, and* m *are the color number, charger number, and task number, respectively,* K *is the number of considered time slots for all tasks.*

Proof. Suppose the optimal charging utility for HASTE is $\overline{\mathcal{U}}^*$, and that for HASTE-R is $\overline{\mathcal{U}}_R^*$. Apparently, we have

$$\overline{\mathcal{U}}_R^* \geq \overline{\mathcal{U}}^*. \tag{9}$$

Further, suppose the output X of Algorithm 2 achieves overall charging utility $\overline{\mathcal{U}}_R$ for HASTE-R, i.e.,

$$\overline{\mathcal{U}}_R = \sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\sum_{k=t_r^j/T_s+1}^{t_e^j/T_s} \sum_{\substack{\Gamma_{i,k}^p \ni o_j,\, i\in[n],\\ p\in\{p|\Theta_{i,k}^p = X\cap\mathbf{\Theta}_{i,k}\}}} P_r(s_i,o_j)T_s\right),$$

and achieves $\overline{\mathcal{U}}$ ($\overline{\mathcal{U}} \leq \overline{\mathcal{U}}_R$) for HASTE by taking the switching delay into consideration. Consider the worst case, i.e., every charger needs to rotate at the beginning of each time slot and lead to switching delay, which results in a time duration of $(1-\rho)T_s$ for effective charging in each time slot for all tasks, then we have

$$\begin{aligned}\overline{\mathcal{U}} &\geq \sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\sum_{k=t_r^j/T_s+1}^{t_e^j/T_s} \sum_{\substack{\Gamma_{i,k}^p \ni o_j,\, i\in[n],\\ p\in\{p|\Theta_{i,k}^p = X\cap\mathbf{\Theta}_{i,k}\}}} P_r(s_i,o_j)(1-\rho)T_s\right)\\ &\geq (1-\rho)\sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\sum_{k=t_r^j/T_s+1}^{t_e^j/T_s} \sum_{\substack{\Gamma_{i,k}^p \ni o_j,\, i\in[n],\\ p\in\{p|\Theta_{i,k}^p = X\cap\mathbf{\Theta}_{i,k}\}}} P_r(s_i,o_j)T_s\right)\\ &= (1-\rho)\overline{\mathcal{U}}_R.\end{aligned} \tag{10}$$

Note that the second inequality in the above formula is due to the concavity of the charging utility function. Following Lemma 5.1 and letting $C \rightarrow +\infty$, we have

$$\overline{\mathcal{U}}_R \geq \left(1-\frac{1}{e}\right)\overline{\mathcal{U}}_R^*. \tag{11}$$

Combining Equs. (9), (10), and (11) we obtain

$$\overline{\mathcal{U}} \geq (1-\rho)(1-\frac{1}{e})\overline{\mathcal{U}}^*, \tag{12}$$

which indicates that Algorithm 2 achieves $(1-\rho)(1-\frac{1}{e})$ approximation ratio.

For time complexity, it is clear that the computation for Q is the dominating part. The computation inside the two-level loop involves testing all possible scheduling policies, which is $O(m)$ in the worst case when the considered charger covers all m tasks. Moreover, computing $\mathbb{F}(Q+x)$ needs $O(nmK)$ computational cost. Thus, considering all CnK loops, the overall time complexity for computing Q is $O(C(nmK)^2)$, so does the time complexity of Algorithm 2. This completes the proof. □

6 Distributed Online Algorithm

In this section, we propose a distributed online algorithm to address HASTE in the online scenario. Note that in this case, charging tasks stochastically arrive, and chargers reschedule their orientations in realtime. Moreover, chargers can be either in the working mode for the offline scenario or in that for the online scenario, and cannot switch between these two different statuses.

We face two main challenges. First, we need to adapt the centralized offline algorithm to HASTE, whose relaxed version HASTE-R is a submodular function maximization problem, to cater to the distributed online scenario where all chargers are asynchronous and charging tasks randomly arrive. Nevertheless, to the best of our knowledge, there are no distributed online schemes for maximizing a submodular function with or without constraints. Second, the response of each charger has a delay of up to $\tau+\rho$ time slots, that is, τ number of time slots for computation and negotiation with neighboring chargers and, possibly, plus ρ time slot for switching delay. This setting is fundamentally different from existing ones of online scheduling problems and invalidates traditional online algorithms. We address these challenges by proposing a distributed online algorithm that achieves $\frac{1}{2}(1-\rho)(1-\frac{1}{e})$ competitive ratio.

6.1 Algorithm Description

We design our distributed online algorithm for HASTE based on our proposed centralized offline algorithm, and partially borrow the idea of the distributed algorithm in [63].

First, we present some concepts to assist analysis.

- *Neighbors of a charger*. We say two chargers are neighbors to each other if and only if they cover at least one charging task in common. We assume that the communication range of wireless chargers is at least twice of their charging range, and therefore, the neighboring wireless chargers can communicate with each other. The set of neighbors of charger s_i is denoted as $N(s_i)$.

- *Local charging utility function*. The local charging utility function for charger s_i is defined as the aggregated charging utility of all charging tasks that can be charged by s_i, i.e., $\boldsymbol{\mathcal{T}}_i$. Denote by X_i as the set of scheduling policies of s_i, and $\mathcal{X}_i$ the set of scheduling polices of s_i and its neighbors $N(s_i)$, we can formally express the local charging utility function for HASTE-R as $f_i : \cup_{s_{i'}\in\{s_i\}\cup N(s_i), k\in[K_i]} \boldsymbol{\Theta}_{i',k} \mapsto \mathbb{R}_{\geq 0}$ as

$$f_i(X_i) = \sum_{\mathcal{T}_j\in\boldsymbol{\mathcal{T}}_i} w_j \mathcal{U}\Bigg(\sum_{k=t_r^j/T_s+1}^{t_e^j/T_s} \sum_{\substack{\Gamma^p_{i,k}\ni o_j,\, s_{i'}\in\{s_i\}\cup N(s_i),\\ p\in\{p|\Theta^p_{i,k}=\mathcal{X}_i\cap\boldsymbol{\Theta}_{i,k}\}}} P_r(s_{i'}, o_j)T_s \Bigg),$$

  where K_i is the number of considered time slots for all tasks $\boldsymbol{\mathcal{T}}_i$ observed by charger s_i.
- *Local expected charging utility function after S-C tuple sampling*. Similar to the expected charging utility function after S-C tuple sampling defined in Section 5.1, we define $\mathbb{F}_i(Q_i) = \mathbb{E}(f_i(\mathtt{sample}_{\vec{c}}(Q_i)))$ as the expected value of $f_i(\mathtt{sample}_{\vec{c}}(Q_i))$ when each color $c_{i,k}$ in $\vec{c}$ is selected uniformly at random from $[C]$.
- *Control message*. The control message exchanged between wireless chargers is expressed as $msg(ID, TIM, COL, CMD, \Delta\mathbb{F}_i^*(Q_i), e_i^{k*})$. The field ID is the charger ID; TIM is the index of the time slots; COL is an integer between 1 and C, which stands for the parameter c in the centralized offline algorithm; CMD can be UPD which indicates an update command; and $\Delta\mathbb{F}_i^{k*}(Q_i)$ is the "maximum" marginal increment for the local expected charging utility function after S-C tuple sampling for charger s_i for all possible scheduling policies at the k_{th} time slot, and e_i^{k*} is the corresponding scheduling policy.

We show our distributed online algorithm in Algorithm 3, which is invoked at charger s_i upon arrival of new charging tasks that can be charged by s_i. Each charger accordingly updates the set of charging tasks $\boldsymbol{\mathcal{T}}_i$, all possible scheduling policies in all K_i time slots $\boldsymbol{\Theta}_{i,k}$, and the local charging utility function $f_i(.)$. Then, each charger s_i enumerates all C colors in all K_i time slots. For each color c at the k_{th} time slot, s_i computes $\Delta\mathbb{F}_i^{k*}(Q_i)$ and the corresponding scheduling policy e_i^{k*}, and broadcasts them to its neighbors. Note that $\Delta\mathbb{F}_i^{k*}(Q_i)$ for charger s_i is obtained by greedily choosing the scheduling policies that yield the maximum additional local expected charging utility in all K_i time slots in an increasing order, and therefore, e_i^{k*} is a set of K_i scheduling policies. Meanwhile, s_i receives the control messages sent from its neighbors. If it collects the messages from all its neighbors and finds that it has the maximum value of $\Delta\mathbb{F}_i^{k*}(Q_i)$ (if there are two or more chargers have the same value of $\Delta\mathbb{F}_i^{k*}(Q_i)$ which leads to a tie, we break it based on the IDs of these chargers), s_i adds the S-C tuple (e_i^{k*}, c) to its global S-C tuple set Q_i, and broadcasts the update command to its surrounding neighbors. Otherwise, if it receives an update command from one of its neighbors, s_i updates the stored scheduling policy for the neighbor, recomputes $\Delta\mathbb{F}_i^{k*}(Q_i)$ and e_i^{k*}, and repeats the above negotiation procedure. After traversing all C colors for all K_i time slots, Algorithm 3 obtains a set of S-C tuples Q_i, and applies a sampling function on Q_i to get a solution X_i.

Algorithm 3. Distributed Online Algorithm to HASTE (at each wireless charger s_i)

Input: Neighbor set $N(s_i)$
Output: Scheduling policy X_i

```
Update the set of charging tasks that can cover charger s_i, i.e., T_i to include the new arrived tasks;
Compute the dominant task sets and determine all possible scheduling policies Θ_{i,k};
Exchange the information of dominant task sets and scheduling policies with the neighbors, and thus derive the local charging utility function f_i(.);
Q_i ← ∅;
for k from 1 to K_i do
    for c from 1 to C do
        Calculate ΔF_i^{k*}(Q_i) and obtain e_i^{k*};
        Broadcast msg(i, k, c, NULL, ΔF_i^{k*}(Q_i), e_i^{k*});
        while ΔF_i^{k*}(Q_i) > 0 do
            if ΔF_j^{k*}(Q_j) of all neighbors s_j ∈ N(s_i) are collected and all their colors are equal to c, and ΔF_i^{k*}(Q_i) is larger than any of them then
                Q_i ← Q_i ∪ (e_i^{k*}, c);
                Broadcasts msg(i, k, c, UPD, ΔF_i^{k*}(Q_i), e_i^{k*});
                break;
            if msg(j, k, c, UPD, ΔF_j^{k*}(Q_j), e_j^{k*}) is received then
                Update the stored scheduling policy of its neighbor s_j at the k_th time slots to e_j^{k*};
                Calculate ΔF_i^{k*}(Q_i) and obtain e_i^{k*};
                Broadcast msg(i, k, c, NULL, ΔF_i^{k*}(Q_i), e_i^{k*});
                continue;
            if msg(j, k, c, NULL, ΔF_j^{k*}(Q_j), e_j^{k*}) is received then
                Update ΔF_j^{k*}(Q_j) and e_j^{k*} for the neighbor s_j;
                continue;
for c from 1 to C do
    Choose c_k^i uniformly at random from [C];
X_i ← sample_c(Q_i), where c = (c_1^i, ..., c_{K_i}^i).
return X_i
```

6.2 Theoretical Analysis

Theorem 6.1. *Algorithm 3 achieves $\frac{1}{2}(1-\rho)(1-\frac{1}{e})$ competitive ratio for HASTE, and its time complexity is $O(C(|N(s_i)||\boldsymbol{\mathcal{T}}_i|K_i)^2)$, its communication cost is $O(CK_i(|N(s_i)|)^2)$ where ρ is the switching delay, C is the number of colors, $N(s_i)$ is the set of neighbors of charger s_i, $\boldsymbol{\mathcal{T}}_i$ is the set of tasks that can cover s_i, K_i is the number of considered time slots for all tasks in $\boldsymbol{\mathcal{T}}_i$.*

Proof. We first analyze the competitive ratio. To begin with, we ignore the rescheduling delay of chargers. Different from the centralized offline algorithm described in Algorithm 2 that is executed in a well-ordered sequence, the online algorithm is conducted in a totally asynchronous manner among wireless chargers. Nevertheless, we prove that we can organize the scheduling policies determination processes at all chargers in a global order. First, as the processes of determining scheduling policies for difference colors $c \in [C]$ are in different loops as shown in Algorithm 2, we can equivalently think of the processes

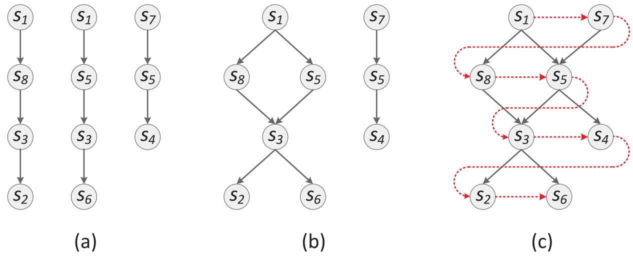

Fig. 3. An example of directed acyclic graph construction.

of determining scheduling policies for difference colors being isolated from each other and executed in order. For each color, it is clear that the process of determining scheduling policies for a charger s_i and its neighbors is executed in order, which can be expressed as a directed chain with a directed edge between s_i and s_j indicating that the scheduling policies of s_i is determined just left behind that of s_j. For instance, suppose the observed order of determining scheduling policies for s_1, s_3, and s_5 are $s_1 \rightarrow s_8 \rightarrow s_3 \rightarrow s_2$, $s_1 \rightarrow s_5 \rightarrow s_3 \rightarrow s_6$, and $s_7 \rightarrow s_5 \rightarrow s_4$, respectively, then we can plot their order chains as in Fig. 3a. Next, we combine these chains by merging the same nodes. For example, Fig. 3b illustrates the resulted directed graph when we combine two directed chains corresponding to s_1 and s_3 by merging the two nodes for s_1 and s_3. Similarly, we can further combine the directed chain of s_5 by merging the node for s_5 as shown in Fig. 3c. After all, we can obtain a directed graph G, which must be acyclic, i.e., with no directed cycles, as otherwise we can always find a charger s_i determining its scheduling policies ahead of itself and thus a contradiction arises. Consequently, we can apply some topological sorting algorithm, such as the well-known linear time topological sorting algorithm presented in [64], to order all the chargers. For example, the red dotted lines in Fig. 3c connecting all the nodes indicate a topological sort of $s_1 \rightarrow s_7 \rightarrow s_8 \rightarrow s_5 \rightarrow s_3 \rightarrow s_4 \rightarrow s_2 \rightarrow s_6$.

Second, clearly the "maximum" marginal increment for the local expected charging utility function after S-C tuple sampling for charger s_i, i.e., $\Delta\mathbb{F}_i^{k*}(Q_i)$, computed by each charger is exactly equal to the "maximum" marginal increment for the global expected charging utility function after S-C tuple sampling because the increased charging utility exactly stems from the affected tasks covered by charger s_i. Then, all chargers can be regarded as sequentially determining their scheduling policies based on the global knowledge of the expected charging utility function after S-C tuple sampling, just as that in the centralized offline algorithm.

Third, in Algorithm 3, the loop for enumerating all time slots is outside the loop for enumerating all colors. This is critical for online algorithm design because as such, the process of being interrupted by arrivals of new charging tasks, recomputing the new scheduling policies and carrying out these new polices for Algorithm 3 can be equivalently viewed as the fluent process with all charging tasks are known a priori. Nevertheless, one may notice that Algorithm 3 differs from Algorithm 2 in that the latter has the loop for enumerating all time slots being inside the loop for enumerating all colors, then does it make any difference in the ultimate performance guarantee? Our answer is negative. Briefly speaking, the TABULARGREEDY algorithm, upon which Algorithm 2 is based, is essentially the locally greedy algorithm for selecting $C \times K$ items that maximize a submodular function [54], [55] for which the order for selection does not matter. We omit the detail analysis to save space.

To sum up, we claim that Algorithm 3 achieves the same performance as Algorithm 2.

Next, we consider rescheduling delay, and first, we neglect the switching delay as for HASTE-R. Suppose the global solution X based on the outputs X_i of Algorithm 3 for all chargers achieves charging utility $\overline{\mathcal{U}}_R$ for HASTE-R, i.e.,

$$\overline{\mathcal{U}}_R = \sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\sum_{k=t_r^j/T_s+1}^{t_e^j/T_s} \sum_{\substack{\Gamma_{i,k}^p \ni o_j,\, i\in[n],\\ p\in\{p|\Theta_{i,k}^p = X\cap\mathbf{\Theta}_{i,k}\}}} P_r(s_i,o_j)T_s\right).$$

Due to rescheduling delay, the reaction of each charger for a newly arrived charging task is delayed for $\tau \cdot T_s$ time. Therefore, it can be equivalently considered that there is no rescheduling delay for chargers under the setting where the first τ time slots of all the charging tasks are "cut off". Suppose X achieves overall charging utility $\overline{\mathcal{U}}_R'$ for this setting, i.e.,

$$\overline{\mathcal{U}}_R' = \sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\sum_{k=t_r^j/T_s+\tau+1}^{t_e^j/T_s} \sum_{\substack{\Gamma_{i,k}^p \ni o_j,\, i\in[n],\\ p\in\{p|\Theta_{i,k}^p = X\cap\mathbf{\Theta}_{i,k}\}}} P_r(s_i,o_j)T_s\right).$$

Obviously, we have $\overline{\mathcal{U}}_R \geq \overline{\mathcal{U}}_R'$ as each task misses the opportunity to harvest charging power at its first τ time slots. Assume the optimal overall charging utility for the above setting is $\overline{\mathcal{U}}_R'^*$, then we have

$$\overline{\mathcal{U}}_R \geq \overline{\mathcal{U}}_R' \geq \left(1 - \frac{1}{e}\right)\overline{\mathcal{U}}_R'^*. \tag{13}$$

Further, assume that the optimal overall charging utility for HASTE-R is $\overline{\mathcal{U}}_R^*$ and its corresponding solution is X^*. Due to the concavity of the charging utility function, we have

$$\begin{aligned}
\overline{\mathcal{U}}_R^* &= \sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\sum_{k=t_r^j/T_s+1}^{t_e^j/T_s} \sum_{\substack{\Gamma_{i,k}^p \ni o_j,\, i\in[n],\\ p\in\{p|\Theta_{i,k}^p = X^*\cap\mathbf{\Theta}_{i,k}\}}} P_r(s_i,o_j)T_s\right)\\
&\leq \sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\sum_{k=t_r^j/T_s+1}^{k=t_r^j/T_s+\tau} \sum_{\substack{\Gamma_{i,k}^p \ni o_j,\, i\in[n],\\ p\in\{p|\Theta_{i,k}^p = X^*\cap\mathbf{\Theta}_{i,k}\}}} P_r(s_i,o_j)T_s\right)\\
&\quad + \sum_{j=1}^{m} w_j \cdot \mathcal{U}\left(\sum_{k=t_r^j/T_s+\tau+1}^{t_e^j/T_s} \sum_{\substack{\Gamma_{i,k}^p \ni o_j,\, i\in[n],\\ p\in\{p|\Theta_{i,k}^p = X^*\cap\mathbf{\Theta}_{i,k}\}}} P_r(s_i,o_j)T_s\right)\\
&\leq \overline{\mathcal{U}}_R^{*1} + \overline{\mathcal{U}}_R^{*2}.
\end{aligned} \tag{14}$$

Note that we denote by $\overline{\mathcal{U}}_R^{*1}$ and $\overline{\mathcal{U}}_R^{*2}$ the first and second terms at the right side of the second inequality. First, we have

$$\overline{\mathcal{U}}_R^{*2} \leq \overline{\mathcal{U}}_R^{\prime *}, \tag{15}$$

as the latter is optimal under the same setting. Second, recall that all the charging tasks have a duration of at least $2\tau T_s$ where τ is the switching delay, which indicates $t_e^j/T_s - (t_r^j/T_s + \tau + 1) + 1 \geq (t_r^j/T_s + \tau) - (t_r^j/T_s + 1) + 1$. Thus, the duration of each task regarding $\overline{\mathcal{U}}_R^{*2}$ is greater than or equal to that of the corresponding task regarding $\overline{\mathcal{U}}_R^{*1}$. Notice that we can move the starting time points of all tasks regarding $\overline{\mathcal{U}}_R^{*1}$ for τ time slots along the time dimension to make them aligned with the corresponding tasks regarding $\overline{\mathcal{U}}_R^{*2}$, we have

$$\overline{\mathcal{U}}_R^{*1} \leq \overline{\mathcal{U}}_R^{\prime *}. \tag{16}$$

Combining Equs. (13), (14), (15), and (16), we obtain

$$\overline{\mathcal{U}}_R \geq \frac{1}{2}(1 - \frac{1}{e})\overline{\mathcal{U}}_R^{*}. \tag{17}$$

Thus, Algorithm 3 achieves $\frac{1}{2}(1 - \frac{1}{e})$ competitive ratio.

Last, by similar analysis on switching delay as in the proof to Theorem 5.1, the achieved competitive ratio of Algorithm 3 is $\frac{1}{2}(1 - \rho)(1 - \frac{1}{e})$.

The time complexity analysis is similar to that in the proof to Algorithm 5.1, we omit it to save space. For communication cost, it is clear that there are in total CK_i loops, and in each loop, there are $O(|N(s_i)|)$ rounds to determine a local a°maximum$a\pm$ marginal increment for the local expected charging utility function after S-C tuple sampling for a charger and its neighbors. Each round in turn needs $O(|N(s_i)|)$ times of message sending and receiving. To sum up, the total communication cost is $O(CK_i(|N(s_i)|)^2)$. This completes the proof. □

7 Simulation Results

In this section, we perform simulations to evaluate the performance of the proposed centralized offline and distributed online algorithms to HASTE.

7.1 Evaluation Setup

Unless otherwise stated, we use the following setup in our simulations. The considered field is a $50\,m \times 50\,m$ square area, and wireless chargers and charging tasks are uniformly distributed in this filed. We set $\alpha = 10000$, $\beta = 40$, $D = 20\,m$, $n = 50$, $m = 200$, $w_j = \frac{1}{200}$, $T_s = 1\,min$, $\rho = \frac{1}{12}$, $\tau = 1$, $A_s = \pi/3$, $A_o = \pi/3$, respectively. The required charging energy and duration of charging tasks are randomly selected in $[5kJ\ 20kJ]$ and $[10min\ 120min]$, respectively. If we choose 3.7 Volts for the voltage, the required charging energy is selected in $[375mAh\ 1500mAh]$. Therefore, the simulation setup is reasonable for the required battery of wireless sensor and mobile devices. To cover the area of $50\,m \times 50\,m$, we chose $n = 50$ wireless chargers. Note that if the number of wireless chargers is too small, some charging tasks area will not be covered, which will make HASTE meaningless. Conversely, if the number of wireless chargers is too large, the charging utility will close to 1.0. Besides, each data point in the figures in this section stands for an averaging result for 100 random topologies.

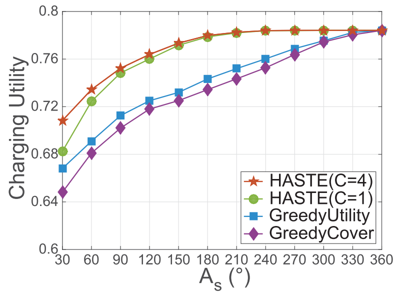

Fig. 4. A_s versus charging utility (centralized offline algorithm).

7.2 Baseline Setup

As there are no existing schemes for scheduling charging tasks in directional wireless charger networks, we propose two algorithms named GreedyUtility and GreedyCover for comparison. For GreedyUtility, each charger greedily picks the orientation that leads to maximum charging utility while ignoring the scheduling policies of its neighboring chargers. For GreedyCover, the only difference compared with GreedyUtility is that each charger greedily selects the orientation that covers the maximum number of charging tasks. Apparently, both of these algorithms can be easily implemented in a distributed way by letting each charger execute them locally.

7.3 Centralized Offline Algorithm Evaluation

7.3.1 Impact of Charging Angle A_s

Our simulation results show that on average HASTE outperforms GreedyUtility and GreedyCover by 2.67 and 3.40 percent (at most 4.34 and 6.03 percent), respectively, in terms of A_s. Fig. 4 shows that the charging utilities of HASTE, GreedyUtility, and GreedyCover steadily increase with the charging angle of chargers A_s, and achieve the same maximum overall charging utility when $A_s = 360°$. Note that for simplicity, we still use HASTE to denote our proposed centralized offline algorithm or distributed online algorithm to HASTE in all simulation figures if no confusion arises. This observation is consistent with our intuition as the larger the charging angle, the larger the chance that a charger can cover more charging tasks even with the same orientation, and all the chargers cover the same set of tasks regardless of their orientations when $A_s = 360°$ and thus make no difference in the performance for the three algorithms. Moreover, the solution for HASTE with the color number $C = 4$ always outperforms that with $C = 1$, and has a performance gain of 0.39 percent on average (at most 2.59 percent).

To validate the performance guarantee of our proposed centralized algorithm, we conduct simulations for a small-scale network with five chargers and ten tasks in a $10\,\text{m} \times 10\,\text{m}$ field and under the setting $T_s = 1\,\text{min}$, $\rho = \frac{1}{12}$, $\tau = 1$, $A_s = \pi/3$, $A_o = \pi/3$, respectively. The required charging energy and duration of charging tasks are randomly selected in $[200\,J\ 800\,kJ]$ and $[1\,\text{min}\ 5\,\text{min}]$, respectively. We compute the optimal solution by a brute-force algorithm

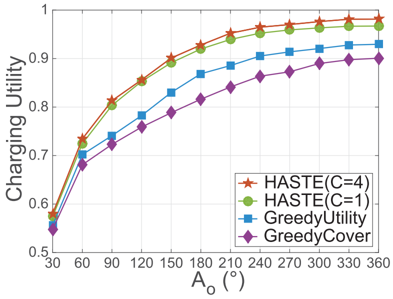

Fig. 5. A_o versus charging utility (centralized offline algorithm).

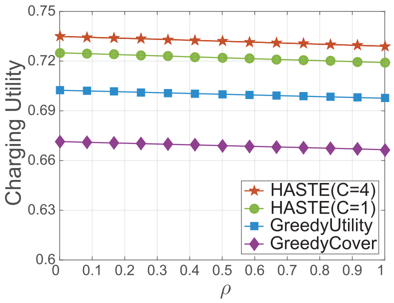

Fig. 6. ρ versus charging utility (centralized offline algorithm).

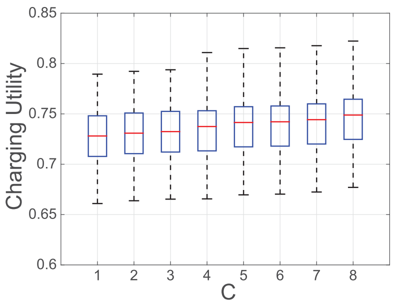

Fig. 7. C versus charging utility (centralized offline algorithm).

which enumerates all combinations of scheduling polices for chargers, and plot it in Fig. 8. We can verify that even for HASTE with $C = 1$, its charging utility is far greater than $(1-\rho)(1-\frac{1}{e}) \approx 0.579$ (at least 92.97 percent) of the optimal charging utility. This fact supports Theorem 5.1.

7.3.2 Impact of Receiving Angle A_o

Our simulation results show that on average HASTE outperforms GreedyUtility and GreedyCover by 5.63 and 8.81 percent (at most 7.36 and 11.27 percent), respectively, in terms of A_o. Fig. 5 shows that the charging utilities of the three algorithms increase monotonically with the receiving angle of devices A_o. This is because tasks with larger receiving angles can be charged with more potential chargers. Clearly, the increasing speeds of charging utilities for these algorithms are first fast and then become slow as A_o increases from 30° to 360°. On average, HASTE with $C = 4$ outperforms HASTE with $C = 1$ by 1.04 percent on average (at most 1.45 percent).

Further, we conduct small-scale simulations under the same setting in Section 7.3.1. We can see from Fig. 9 that the achieved charging utility for either $C = 1$ or $C = 4$ is very close to the optimal, specifically, it is at least 88.63 percent

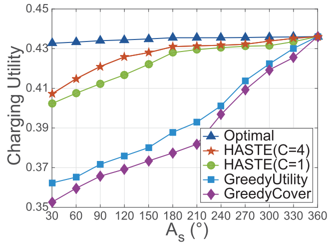

Fig. 8. A_s versus charging utility (small-scale networks).

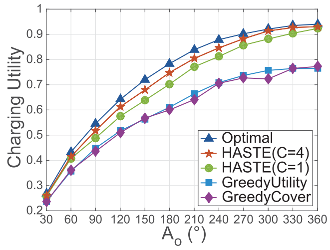

Fig. 9. A_o versus charging utility (small-scale networks).

($> \frac{1}{2}(1-\rho)(1-\frac{1}{e}) \approx 0.290$) of the latter. This finding corroborates Theorem 6.1.

Besides, though Figs. 4 and 5 show that the charging utility increases with growing A_s and A_o, it does not mean that omnidirectional WPT is superior to directional WPT. This is because in reality, with identical hardware settings and working power, directional WPT concentrates more radiated energy in the directions of the rechargeable devices via energy beamforming, which enhances the power intensity in the intended directions, or equivalently, enhance the whole energy efficiency [8]. Therefore, we can image that some other charging parameters will change accordingly when increasing A_s and A_o, such as a decreasing α, which is not reflected in Figs. 4 and 5.

7.3.3 Impact of Switching Delay ρ

Our simulation results show that on average HASTE outperforms GreedyUtility and GreedyCover by 3.20 and 6.30 percent (at most 3.25 and 6.34 percent), respectively, in terms of ρ. Not surprisingly, we observe in Fig. 6 that the charging utilities for all the algorithms smoothly decrease with an increasing switching delay ρ. HASTE with $C = 4$ outperforms HASTE with $C = 1$ by 0.99 percent (at most 1.00 percent). Note that even when $\rho = 1$, which means the switching delay is up to one time slot, the achieved charging utilities for all the algorithms just slightly degrade. The reason is that each charger keeps still most of the time and the orientation switching seldom happens, and therefore, the performance loss caused by switching is little.

7.3.4 Impact of Color Number C

Our simulation results show that on average the achieved charging utility of HASTE steadily increases with color number C. Fig. 7 shows the box plot of the charging utilities of HASTE. It can be seen that the average charging utility of HASTE increases by 3.29 percent when the color number C increases from 1 to 8. The maximum and minimum

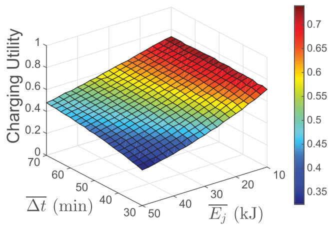

Fig. 10. Required charging energy & task duration versus charging utility (centralized offline algorithm).

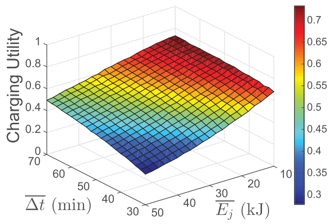

Fig. 11. Required charging energy & task duration versus charging utility (distributed online algorithm).

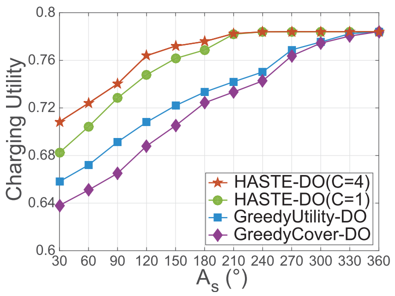

Fig. 12. A_s versus charging utility (distributed online algorithm).

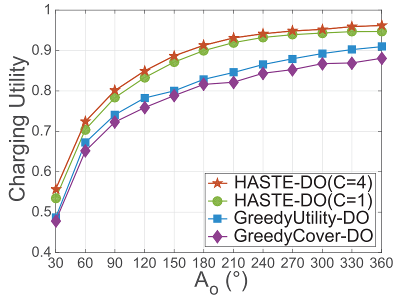

Fig. 13. A_o versus charging utility (distributed online algorithm).

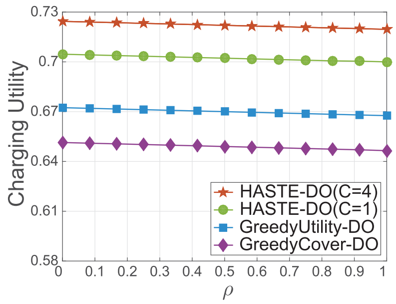

Fig. 14. ρ versus charging utility (distributed online algorithm).

charging utilities of HASTE also smoothly increase with C. The variance of charging utility for the eight colors is at most 8.56×10^{-3}.

7.3.5 Impact of Required Charging Energy and Task Duration

Our simulation results show that the achieved charging utility of HASTE steadily increases with a decreasing charging energy or an increasing task duration. We set the required charging energy being randomly selected from $[0.5\overline{E_j}\ 1.5\overline{E_j}]$, and task duration from $[0.5\overline{\Delta t}\ 1.5\overline{\Delta t}]$. Fig. 10 shows that when $\overline{E_j}$ decreases from $50\,kJ$ to $10\,kJ$ and $\overline{\Delta t}$ increases from $30\,\text{min}$ to $70\,\text{min}$, the overall charging utility increases by 44.28 percent. Moreover, the increasing speed for charging utility slows down when $\overline{E_j}$ is large or $\overline{\Delta t}$ is small, which indicates a marginal diminishing gain property.

7.4 Distributed Online Algorithm Evaluation

7.4.1 Impact of Charging Angle A_s

Our simulation results show that on average HASTE outperforms GreedyUtility and GreedyCover by 3.33 and 4.47 percent (at most 5.59 and 7.59 percent), respectively, in terms of A_s. We denote by HASTE-DO the distributed online algorithm for HASTE in the following figures. Fig. 12 demonstrates that the charging utilities of HASTE, GreedyUtility, and GreedyCover smoothly increase with the charging angle of chargers A_s, and reach the same maximum overall charging utility when $A_s = 360°$. This is a natural result because the larger the charging angle, the larger the chance that a charger can cover more charging tasks with the same orientation. Moreover, if $A_s = 360°$, each charger covers the same set of tasks regardless of its orientations, and therefore, the three algorithms have the same performance. The solution for HASTE with $C = 4$ always outperforms that with $C = 1$ with a gain of 0.77 percent on average (at most 2.59 percent). Besides, we can see that the charging utility for each of the three distributed online algorithms is less than that of its corresponding centralized offline algorithm.

7.4.2 Impact of Receiving Angle A_o

Our simulation results show that on average HASTE outperforms GreedyUtility and GreedyCover by 6.83 and 8.95 percent (at most 8.68 and 10.96 percent), respectively, in terms of A_o. Fig. 13 illustrates that the charging utilities of the three algorithms monotonically increase with the receiving angle of devices A_o. The reason is that tasks with larger receiving angles can potentially be charged by more chargers, and thus receive more energy on average. Moreover, it is clear that the increasing trends of charging utilities for these algorithms are first fast and then become slow as A_o increases from 30° to 360°. Besides, HASTE with $C = 4$ outperforms HASTE with $C = 1$ by 1.42 percent on average (at most 2.23 percent). Again, the charging utilities for the distributed online algorithms are less than their corresponding centralized offline version.

7.4.3 Impact of Switching Delay ρ

Our simulation results show that on average HASTE outperforms GreedyUtility and GreedyCover by 5.20 and 7.3 percent (at most 5.20 and 7.31 percent), respectively, in terms of ρ. Fig. 14 shows that the charging utilities for all the algorithms steadily

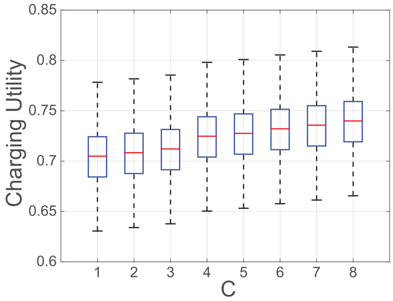

Fig. 15. C versus charging utility (distributed online algorithm).

decrease with switching delay ρ. Especially, HASTE with $C = 4$ outperforms HASTE with $C = 1$ by 1.98 percent. When the switching delay is even up to one time slot, i.e., $\rho = 1$, the achieved charging utilities for all the algorithms only slightly degrade compared with $\rho = 0$. This is because most chargers keep still most of the time, and thus the caused performance loss is little.

7.4.4 Impact of Color Number C

Our simulation results show that on average the achieved charging utility of HASTE steadily increases with color number C. Fig. 15 demonstrates the box plot of the charging utilities of HASTE when the color number C increases from 1 to 8. We can see that both of the maximum and minimum charging utilities of HASTE steadily increase with C. Moreover, on average the average charging utility of HASTE increases by 3.08 percent when the color number C increases by 1. Besides, the variance of charging utility for all the eight colors is at most 8.42×10^{-3}, which indicates the stable performance of our algorithm.

7.4.5 Impact of Required Charging Energy and Task Duration

Our simulation results show that the achieved charging utility of HASTE steadily increases with a decreasing charging energy or an increasing task duration. Similar to the setting for the centralized offline algorithm, we set the required charging energy being randomly selected from $[0.5\overline{E_j}\ 1.5\overline{E_j}]$, and task duration from $[0.5\overline{\Delta t}\ 1.5\overline{\Delta t}]$. Fig. 11 shows that when $\overline{E_j}$ downgrades from $50\,kJ$ to $10\,kJ$ and $\overline{\Delta t}$ rises from $30\,\text{min}$ to $70\,\text{min}$, the achieved charging utility increases by 45.47 percent. The increasing speed for charging utility decreases when $\overline{E_j}$ increases or $\overline{\Delta t}$ decreases, which demonstrates a marginal diminishing gain property.

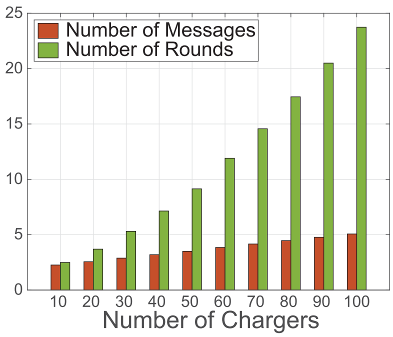

Fig. 16. Communication cost.

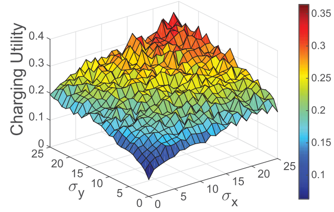

Fig. 17. Overall charging utility versus Guassian distribution variance.

7.4.6 Communication Cost

Our simulation results show that the number of messages and the number of rounds for a time slot increase quadratically and linearly, respectively, with the number of chargers. We set the number of color C to 1, and plot the average numbers of messages and rounds in Algorithm 3 in Fig. 16. We can see that when the number of chargers increases from 10 to 100, the numbers of messages and rounds increase by 223.77 and 952.29 percent, respectively. The number of rounds linearly increases because the number of neighboring chargers linearly increases. Further, as the number of messages in each round also grows proportionally to the number of neighboring chargers, it thus grows quadratically with the number of neighboring chargers, or the number of chargers. This finding supports Theorem 6.1.

7.5 Insights

First, we investigate the impact of distribution of positions of charging tasks on the overall charging utility. Suppose there are 50 tasks distributed in a $50\,\text{m} \times 50\,\text{m}$ area, and $A_o = A_s = \pi/3$. The required charging energy and charging duration for all tasks are randomly chosen from $[5\,kJ\ 20\,kJ]$ and $[10\,\text{min}\ 120\,\text{min}]$, respectively. The positions of tasks are randomly generated following a 2D Gaussian distribution with both x- and y- coordinates obeying a Gaussian distribution with $\mu = 25$. Fig. 17 shows that generally the charging utility increases with either σ_x or σ_y, which indicates that the uniformness of tasks' distribution contributes to the overall charging utility. This is because with a higher degree of uniformness of positions, the phenomenon that some tasks are over-charged while the others are starved out can be effectively avoided, and according to the concavity of the charging utility function, the overall charging utility will be enhanced. Second, we study the impact of E_j on the individual charging utility of each charger. Compared with the above setting, we uniformly distribute 50 chargers and 200 tasks. The required charging energy is a random number in $[5\,kJ\ 100\,kJ]$. Fig. 18 shows that generally the charging utility first can achieve 1 for a small E_j, and then rapidly decreases when E_j continues growing. The maximum individual charging utility is approximately inversely proportional to E_j, as shown by the curve in Fig. 18. The reason is that

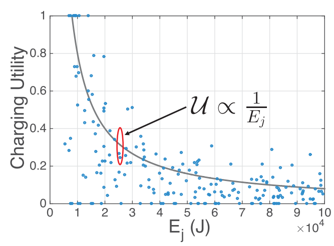

Fig. 18. Individual charging utility versus required charging energy.

Fig. 19. Testbed.

to achieve the same charging utility, a task with a higher required E_j needs a higher average charging power from its surrounding chargers, which is not cost efficient. Thus, higher E_j leads to lower charging utility.

8 Field Experiments

We have conducted field experiments to evaluate our scheme.

First of all, we implemented our proposed schemes on a small textbed which consists of 8 TX91501 power transmitters produced by Powercast [19] with charging angle of about 60°, 8 rechargeable sensor nodes with receiving angle of about 120°, and an AP that connects to a laptop for reporting data collected from the nodes as shown in Fig. 19. Each power transmitter is mounted on a rotatable platform atop a mobile robot, and thus can be freely rotated. Fig. 20 shows the topology of this testbed, where the 8 power transmitters are placed at the boundaries of a $2.4\,\mathrm{m} \times 2.4\,\mathrm{m}$ square area, and the 8 sensor nodes are placed inside the square area. We mark the orientation angle and the release and end time (in time slots) on the top of each task associated with a sensor node in Fig. 20. The required charging energy for all tasks is set to be in $[3\,J\ 5\,J]$. We set $\alpha = 41.93$, $\beta = 0.6428$, $D = 4\,m$, $\rho = \frac{1}{12}$, $\tau = 1$, $A_s = \pi/3$, $A_o = 2\pi/3$, $w_j = \frac{1}{8}$, based on our empirical results for the power transmitters, the sensor nodes, and the robot, and set $T_s = 1\,\mathrm{min}$.

Figs. 21 and 22 show the charging utility for each task for the three algorithms, i.e., HASTE (with $C = 4$), GreedyUtility, and GreedyCover, for the centralized offline and distributed online settings, respectively. We can observe that HASTE basically has the best charging utility for all tasks, and respectively outperforms GreedyUtility and GreedyCover by 4.67 and 12.74 percent on average, and by 16.68 and 24.83 percent at most in the centralized offline scenario; and by 5.62 and 12.38 percent on average, and by 19.52 and 22.10 percent at most in the distributed online scenario. Moreover, task 1 and task 6 have the largest two charging utility for both the algorithms as they have the largest two charging task duration.

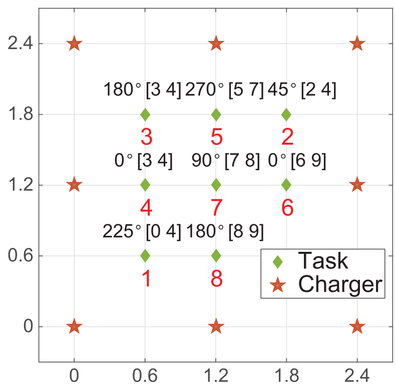

Fig. 20. Topology 1.

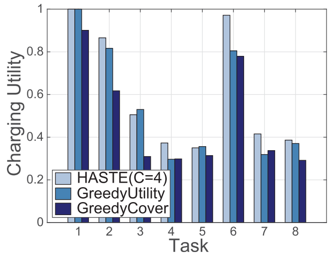

Fig. 21. Charging utility of 8 tasks for the centralized offline algorithms.

Next, we implemented our proposed schemes on a large testbed which consists of 16 TX91501 power transmitters and 20 rechargeable sensor nodes. Fig. 23 shows the topology of this large testbed, which is much more irregular than the small testbed as it is randomly generated. Similarly, Figs. 24 and 25 show that HASTE respectively outperforms GreedyUtility and GreedyCover by 4.38 and 10.12 percent on average, and by 13.27 and 23.60 percent at most in the centralized offline scenario; and by 6.04 and 15.28 percent on average, and by 22.58 and 29.63 percent at most for in the distributed online scenario.

9 Conclusion

The key novelty of this paper is on proposing the first scheduling algorithm for charging tasks in directional wireless charging networks. The key contributions of this paper are proposing a centralized offline algorithm that achieves $(1-\rho)(1-\frac{1}{e})$ approximation ratio where ρ denotes the switching delay, and a distributed online algorithm that achieves $\frac{1}{2}(1-\rho)(1-\frac{1}{e})$ competitive ratio, and conducting both simulations and field experiments for evaluation. The key technical depth of this paper is in transforming the problem into maximizing a submodular function subject to a partition matroid constraint, bounding the performance loss caused by the switching delay and proving the approximation ratio for the centralized offline algorithm, making the centralized offline algorithm distributed and bounding the performance loss caused by the rescheduling delay and proving the competitive ratio for the distributed online algorithm. Our simulation and field experimental results show that our proposed distributed online algorithm can achieve 92.97 percent of the optimal charging utility, outperform the other two comparison

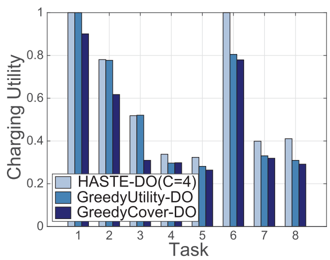

Fig. 22. Charging utility of 8 tasks for the distributed online algorithms.

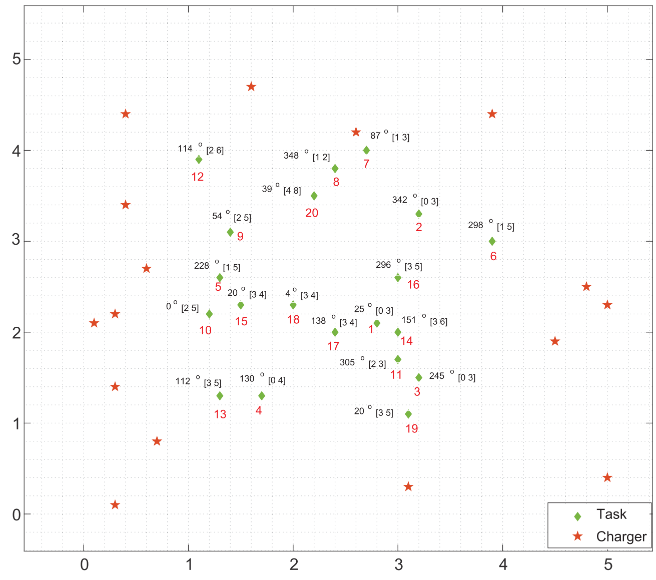

Fig. 23. Topology 2.

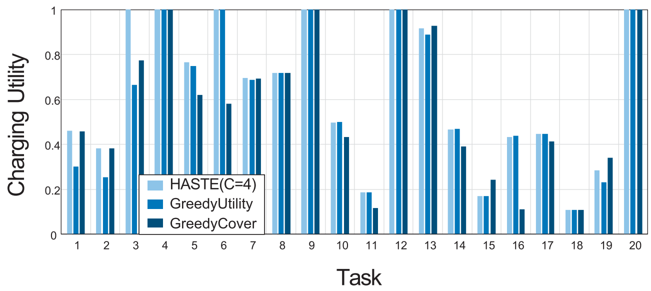

Fig. 24. Charging utility of 20 tasks for the centralized offline algorithms.

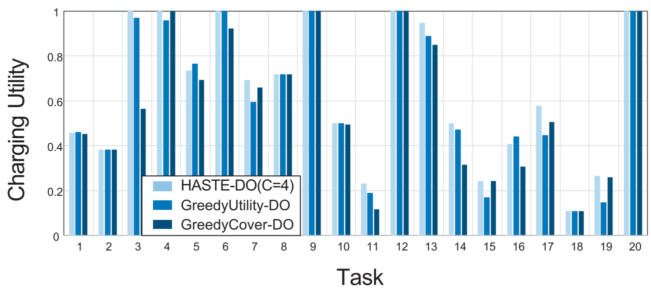

Fig. 25. Charging utility of 20 tasks for the distributed online algorithms.

algorithms, and its communication cost moderately increases as the charger number scales up.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61872178, in part by the Natural Science Foundation of Jiangsu Province under Grant No. BK20181251, in part by Supported by the Fundamental Research Funds for the Central Universities under Grant 14380062, in part by the open research fund of Key Lab of Broadband Wireless Communication and Sensor Network Technology (Nanjing University of Posts and Telecommunications), Ministry of Education, in part by the National Natural Science Foundation of China under Grant 61802172, Grant 61672276, Grant 61832005, Grant 61472184, Grant 61872173, and Grant 61872082, in part by the Key Research and Development Project of Jiangsu Province under Grant BE2015154 and Grant BE2016120, and the Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing University, and in part by the Jiangsu Innovation and Entrepreneurship (Shuangchuang) Program, the Guangdong Leading Talent Program No. 2016LJ06D658.

References

[1] A. P. Sample, D. J. Yeager, P. S. Powledge, A. V. Mamishev, and J. R. Smith, "Design of an RFID-based battery-free programmable sensing platform," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 11, pp. 2608–2615, Nov. 2008.

[2] L. Xie, Y. Shi, Y. T. Hou, and A. Lou, "Wireless power transfer and applications to sensor networks," *IEEE Wireless Commun.*, vol. 20, no. 4, pp. 140–145, Aug. 2013.

[3] S. Li and C. C. Mi, "Wireless power transfer for electric vehicle applications," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 3, no. 1, pp. 4–17, Mar. 2015.

[4] P. E. Glaser, "An overview of the solar power satellite option," *IEEE Trans. Microw. Theory Techn.*, vol. 40, no. 6, pp. 1230–1238, Jun. 1992.

[5] S. F. Bush, *Smart Grid: Communication-Enabled Intelligence for the Electric Power Grid*. Hoboken, NJ, USA: Wiley, 2014.

[6] 2018. [Online]. Available: https://www.wirelesspowerconsortium.com/

[7] 2018. [Online]. Available: https://www.wirelesspowerconsortium.com/blog/273/wireless-power-market-surge s-as-usage-leaps-forward

[8] Z. Wang, L. Duan, and R. Zhang, "Adaptively directional wireless power transfer for large-scale sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1785–1800, May 2016.

[9] L. Wang, M. Elkashlan, R. W. Heath, M. Di Renzo, and K.-K. Wong, "Millimeter wave power transfer and information transmission," in *Proc. IEEE Global Commun. Conf.*, 2015, pp. 1–6.

[10] N. Deng and M. Haenggi, "A novel approximate antenna pattern for directional antenna arrays," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 832–835, Oct. 2018.

[11] T. A. Khan and R. W. Heath, "Wireless power transfer in millimeter wave tactical networks," *IEEE Signal Process. Lett.*, vol. 24, no. 9, pp. 1284–1287, Sep. 2017.

[12] T. A. Khan and R. W. Heath Jr, "Wireless power transfer in millimeter wave," in *Wireless Information and Power Transfer: Theory and Practice*. Chichester, U.K.: John Wiley & Sons, Ltd., pp. 139–156, 2018.

[13] 2018. [Online]. Available: http://www.powercastco.com/pdf/p2110-evb-reva.pdf

[14] O. Georgiou, "Simultaneous wireless information and power transfer in cellular networks with directional antennas," *IEEE Commun. Lett.*, vol. 21, no. 4, pp. 885–888, Apr. 2017.

[15] L. Fan, Q. Wang, Y. Huang, and L. Yang, "Secure wireless information and power transfer based on tilt adaptation in 3-D massive MIMO systems," *IEEE Access*, vol. 7, pp. 5531–5540, 2019.

[16] H. Dai, X. Wang, A. X. X. Liu, F. Zhang, Y. Zhao, and G. Chen, "Omnidirectional chargability with directional antennas," in *Proc. IEEE 24th Int. Conf. Netw. Protocols*, 2016, pp. 1–10.

[17] H. Dai, X. Wang, A. X. X. Liu, F. Zhang, Y. Zhao, and G. Chen, "Optimizing wireless charger placement for directional charging," in *Proc. IEEE 24th Int. Conf. Netw. Protocols*, 2017, pp. 2922–2930.

[18] X. Xu *et al.*, "Optimizing charging efficiency and maintaining sensor network perpetually in mobile directional charging," *Sensors*, vol. 19, no. 12, 2019, Art. no. 2657.

[19] 2018. [Online]. Available: www.powercastco.com

[20] 2019. [Online]. Available: https://blog.ossia.com/news/ossia-and-t-mobile-pilot-wireless-charging-asset-trackers-in-walmart-warehouses

[21] 2019. [Online]. Available: https://wi-charge.com/applications

[22] C. Wang, J. Li, F. Ye, and Y. Yang, "Multi-vehicle coordination for wireless energy replenishment in sensor networks," in *Proc. IEEE 27th Int. Symp. Parallel Distrib. Process.*, 2013, pp. 1101–1111.

[23] C. Wang, J. Li, F. Ye, and Y. Yang, "Recharging schedules for wireless sensor networks with vehicle movement costs and capacity constraints," in *Proc. 11th Annu. IEEE Int. Conf. Sens. Commun. Netw.*, 2014, pp. 468–476.

[24] W. Xu, W. Liang, X. Lin, G. Mao, and X. Ren, "Towards perpetual sensor networks via deploying multiple mobile wireless chargers," in *Proc. IEEE 43rd Int. Conf. Parallel Process.*, 2014, pp. 80–89.
[25] L. Chen *et al.*, "Charge me if you can: Charging path optimization and scheduling in mobile networks," in *Proc. 17th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2016, pp. 101–110.
[26] S. Nikoletseas, T. P. Raptis, and C. Raptopoulos, "Interactive wireless charging for energy balance," in *Proc. IEEE 36th Int. Conf. Distrib. Comput. Syst.*, 2016, pp. 262–270.
[27] L. Fu, P. Cheng, Y. Gu, J. Chen, and T. He, "Minimizing charging delay in wireless rechargeable sensor networks," in *Proc. IEEE Int. Conf. Comput. Commun.*, 2013, pp. 2922–2930.
[28] L. He *et al.*, "ESync: An energy synchronized charging protocol for rechargeable wireless sensor networks," in *Proc. 15th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2014, pp. 247–256.
[29] L. He, P. Cheng, Y. Gu, J. Pan, T. Zhu, and C. Liu, "Mobile-to-mobile energy replenishment in mission-critical robotic sensor networks," in *Proc. IEEE Conf. Comput. Commun.*, 2014, pp. 1195–1203.
[30] Y. Shi, L. Xie, Y. T. Hou, and H. D. Sherali, "On renewable sensor networks with wireless energy transfer," in *Proc. IEEE Conf. Comput. Commun.*, 2011, pp. 1350–1358.
[31] L. Xie, Y. Shi, Y. T. Hou, W. Lou, H. D. Sherali, and S. F. Midkiff, "On renewable sensor networks with wireless energy transfer: The multi-node case," in *Proc. 9th Annu. IEEE Commun. Soc. Conf. Sensor Mesh Ad Hoc Commun. Netw.*, 2012, pp. 10–18.
[32] S. Guo, C. Wang, and Y. Yang, "Mobile data gathering with wireless energy replenishment in rechargeable sensor networks," in *Proc. IEEE Conf. Comput. Commun.*, 2013, pp. 1932–1940.
[33] L. Xie, Y. Shi, Y. T. Hou, W. Lou, H. D. Sherali, and S. F. Midkiff, "Bundling mobile base station and wireless energy transfer: Modeling and optimization," in *Proc. IEEE Conf. Comput. Commun.*, 2013, pp. 1636–1644.
[34] L. Xie, Y. Shi, Y. T. Hou, W. Lou, H. D. Sherali, and S. F. Midkiff, "On traveling path and related problems for a mobile station in a rechargeable sensor network," in *Proc. 14th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2013, pp. 109–118.
[35] C. Wang, J. Li, F. Ye, and Y. Yang, "Improve charging capability for wireless rechargeable sensor networks using resonant repeaters," in *Proc. IEEE 35th Int. Conf. Distrib. Comput. Syst.*, 2015, pp. 133–142.
[36] C. Wang, J. Li, F. Ye, and Y. Yang, "A hybrid framework combining solar energy harvesting and wireless charging for wireless sensor networks," in *Proc. 35th Annu. IEEE Int. Conf. Comput. Commun.*, 2016, pp. 1–9.
[37] Z. Dong *et al.*, "Energy synchronized task assignment in rechargeable sensor networks," in *Proc. 13th Annu. IEEE Int. Conf. Sens. Commun. Netw.*, 2016, pp. 1–9.
[38] S. He, J. Chen, F. Jiang, D. K.Y. Yau, G. Xing, and Y. Sun, "Energy provisioning in wireless rechargeable sensor networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 10, pp. 1931–1942, Oct. 2013.
[39] H. Dai, X. Wang, A. X. Liu, H. Ma, G. Chen, and W. Dou, "Wireless charger placement for directional charging," *IEEE/ACM Trans. Netw.*, vol. 26, no. 4, pp. 1865–1878, Aug. 2018.
[40] N. Yu, H. Dai, A. X. Liu, and B. Tian *et al.*, "Placement of connected wireless chargers," in *Proc. IEEE Conf. Comput. Commun.*, 2018, pp. 387–395.
[41] X. Wang *et al.*, "Heterogeneous wireless charger placement with obstacles," in *Proc. 47th Int. Conf. Parallel Process.*, 2018, Art. no. 16.
[42] H. Dai, Y. Liu, G. Chen, X. Wu, and T. He, "Safe charging for wireless power transfer," in *Proc. IEEE Conf. Comput. Commun.*, 2014, pp. 1105–1113.
[43] H. Dai, Y. Liu, G. Chen, X. Wu, and T. He, "Safe charging for wireless power transfer," *IEEE/ACM Trans. Netw.*, vol. 25, no. 6, pp. 3531–3544, Dec. 2017.
[44] H. Dai, Y. Liu, G. Chen, X. Wu, and T. He, "SCAPE: Safe charging with adjustable power," in *Proc. IEEE 34th Int. Conf. Distrib. Comput. Syst.*, 2014, pp. 439–448.
[45] H. Dai, Y. Liu, G. Chen, X. Wu, and T. He, "SCAPE: Safe charging with adjustable power," *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, pp. 520–533, Feb 2018.
[46] S. Nikoletseas, T. P. Raptis, and C. Raptopoulos, "Low radiation efficient wireless energy transfer in wireless distributed systems," in *Proc. IEEE 35th Int. Conf. Distrib. Comput. Syst.*, 2015, pp. 196–204.
[47] H. Dai, Y. Liu, A. X. Liu, L. Kong, G. Chen, and T. He, "Radiation constrained wireless charger placement," in *Proc. 35th Annu. IEEE Int. Conf. Comput. Commun.*, 2016, pp. 1–9.
[48] H. Dai, Y. Liu, A. X. Liu, L. Kong, G. Chen, and T. He, "Robustly safe charging for wireless power transfer," in *Proc. IEEE Conf. Comput. Commun.*, 2018, pp. 378–386.
[49] H. Dai, Y. Liu, A. X. Liu, L. Kong, G. Chen, and T. He, "Radiation constrained scheduling of wireless charging tasks," in *Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2017, pp. 17–26.
[50] H. Dai, Y. Liu, A. X. Liu, L. Kong, G. Chen, and T. He, "Radiation constrained scheduling of wireless charging tasks," *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, pp. 314–327, Feb. 2018.
[51] H. Dai, Y. Liu, A. X. Liu, L. Kong, G. Chen, and T. He, "Charging task scheduling for directional wireless charger networks," in *Proc. 47th Int. Conf. Parallel Process.*, 2018, Art. no. 10.
[52] G. L. Nemhauser *et al.*, "An analysis of approximations for maximizing submodular set functions-I," *Math. Program.*, vol. 14, no. 1, pp. 265–294, 1978.
[53] G. Calinescu *et al.*, "Maximizing a monotone submodular function subject to a matroid constraint," *SIAM J. Comput.*, vol. 40, no. 6, pp. 1740–1766, 2011.
[54] M. Streeter *et al.*, "Online learning of assignments," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2009, pp. 1794–1802. [Online]. Available: http://papers.nips.cc/paper/3719-online-learning-of-assignments.pdf
[55] D. Golovin, A. Krause, and M. Streeter, "Online submodular maximization under a matroid constraint with application to learning assignments," 2014, *arXiv:1407.1082*.
[56] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Wireless charging technologies: Fundamentals, standards, and network applications," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1413–1452, Secondquarter 2016.
[57] C. Lin, Y. Zhou, F. Ma, J. Deng, L. Wang, and G. Wu, "Minimizing charging delay for directional charging in wireless rechargeable sensor networks," in *Proc. IEEE Conf. Comput. Commun.*, 2019, pp. 1819–1827.
[58] 2018. [Online]. Available: http://www.sevenoak.biz/pantiltheads/sk-ebh01.html
[59] R. Zhang and C. K. Ho, "MIMO broadcasting for simultaneous wireless information and power transfer," *IEEE Trans. Wireless Commun.*, vol. 12, no. 5, pp. 1989–2001, May 2013.
[60] Z. Ding *et al.*,"Application of smart antenna technologies in simultaneous wireless information and power transfer," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 86–93, Apr. 2015.
[61] L. Fleischer *et al.*, "Tight approximation algorithms for maximum general assignment problems," in *Proc. 17th Annu. ACM-SIAM Symp. Discrete Algorithm*, 2006, pp. 611–620.
[62] S. Fujishige, *Submodular Functions and Optimization*, vol. 58. Amsterdam, Netherlands: Elsevier, 2005.
[63] H. Dai *et al.*, "Practical scheduling for stochastic event capture in energy harvesting sensor networks," *Int. J. Sensor Netw.*, vol. 18, no. 1/2, pp. 85–100, 2015.
[64] T. H. Cormen, *Introduction to Algorithms*. Cambridge, MA, USA: MIT Press, 2009.

Haipeng Dai (Member, IEEE) received the BS degree from the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2010, and the PhD degree from the Department of Computer Science and Technology, Nanjing University, Nanjing, China, in 2014. His research interests include the areas of wireless charging, mobile computing, and data mining. He is an associate professor with the Department of Computer Science and Technology, Nanjing University, China. His research papers have been published in many prestigious conferences and journals such as ACM MobiSys, ACM MobiHoc, ACM VLDB, ACM SIGMETRICS, ACM UbiComp, IEEE INFOCOM, IEEE ICDCS, IEEE ICNP, IEEE SECON, IEEE IPSN, the *IEEE Journal on Selected Areas in Communications*, the *IEEE/ACM Transactions on Networking*, the *IEEE Transactions on Mobile Computing*, the *IEEE Transactions on Parallel & Distributed Systems*, and *ACM Transactions on Sensor Networks*. He is an ACM member. He serves/ed as poster chair of the IEEE ICNP'14, track chair of the ICCCN'19, TPC member of the ACM MobiHoc'20, IEEE INFOCOM'20, IEEE ICDCS'20, IEEE ICNP'14, IEEE IWQoS'19, IEEE IPDPS'20, IEEE MASS'18-19, IEEE ICC'14-18, IEEE ICCCN'15-18 and IEEE Globecom'14-18. He received Best Paper Award from IEEE ICNP'15, Best Paper Award Runner-up from IEEE SECON'18, and Best Paper Award Candidate from IEEE INFOCOM'17.

Ke Sun received the BS degree in computer science from the Nanjing University of Aeronautics and Astronautics, Jiangsu, China, in 2016. He is currently working toward the master's degree at Nanjing University, China. His research interests include the area of of mobile computing.

Alex X. Liu (Fellow, IEEE) received the PhD degree in computer science from The University of Texas at Austin, Austin, Texas, in 2006. He is currently a professor with the Department of Computer Science and Engineering, Michigan State University, East Lansing, Michigan. He received the IEEE & IFIP William C. Carter Award, in 2004, a National Science Foundation CAREER award, in 2009, and the Michigan State University Withrow Distinguished Scholar Award, in 2011. He has served as an editor for the *IEEE/ACM Transactions on Networking*, and he is currently an associate editor for the *IEEE Transactions on Dependable and Secure Computing* and the *IEEE Transactions on Mobile Computing*, and an area editor for the *Computer Communications*. He received best paper awards from ICNP-2012, SRDS-2012, and LISA-2010. His research interests include networking and security.

Lijun Zhang (Member, IEEE) received the BS and PhD degrees in software engineering and computer science from Zhejiang University, China, in 2007 and 2012, respectively. He is currently an associate professor of the Department of Computer Science and Technology, Nanjing University, China. Prior to joining Nanjing University, China, he was a postdoctoral researcher with the Department of Computer Science and Engineering, Michigan State University, East Lansing, Michigan. His research interests include machine learning, optimization, information retrieval, and data mining.

Jiaqi Zheng received the PhD degree from Nanjing University, China, in 2017. He is currently an assistant researcher from the Department of Computer Science and Technology, Nanjing University, China. His research interests include computer networking, particularly data center networks, SDN, and NFV. He was an assistant researcher with the City University of Hong Kong, Hong Kong, in 2015, and a visiting scholar in Temple University, Philadelphia, Pennsylvania, in 2016. He received the Best Paper Award from IEEE ICNP 2015 and Doctorial Dissertation Award from ACM SIGCOMM China, 2018. He is a member of ACM.

Guihai Chen (Member, IEEE) received the BS degree in computer software from Nanjing University, China, in 1984, the ME degree in computer applications from Southeast University, China, in 1987, and the PhD degree in computer science from the University of Hong Kong, Hong Kong, in 1997. He is currently a professor and the deputy chair of the Department of Computer Science, Nanjing University, China. He had been invited as a visiting professor by many foreign universities including Kyushu Institute of Technology, Japan, in 1998, University of Queensland, Australia, in 2000, and Wayne State University, Detroit, Michigan during September 2001 to August 2003. He has a wide range of research interests with focus on sensor networks, peer-to-peer computing, high-performance computer architecture, and combinatorics.