Design of Self-sustainable Wireless Sensor Networks with Energy Harvesting and Wireless Charging

PENGZHAN ZHOU, Department of Electrical and Computer Engineering, Stony Brook University
CONG WANG, Department of Computer Science, Old Dominion University
YUANYUAN YANG, Department of Electrical and Computer Engineering, Stony Brook University

Energy provisioning plays a key role in the sustainable operations of Wireless Sensor Networks (WSNs). Recent efforts deploy multi-source energy harvesting sensors to utilize ambient energy. Meanwhile, wireless charging is a reliable energy source not affected by spatial-temporal ambient dynamics. This article integrates multiple energy provisioning strategies and adaptive adjustment to accomplish self-sustainability under complex weather conditions. We design and optimize a three-tier framework with the first two tiers focusing on the planning problems of sensors with various types and distributed energy storage powered by environmental energy. Then we schedule the Mobile Chargers (MC) between different charging activities and propose an efficient, 4-factor approximation algorithm. Finally, we adaptively adjust the algorithms to capture real-time energy profiles and jointly optimize those correlated modules. Our extensive simulations demonstrate significant improvement of network lifetime (3×), increase of harvested energy (15%), reduction of network cost (30%), and the charging capability of MC by 100%.

CCS Concepts: • **Networks** → **Network types**; Wireless access networks;

Additional Key Words and Phrases: Multi-source energy harvesting, wireless charging, wireless sensor networks, energy self-sustainable.

ACM Reference format:
Pengzhan Zhou, Cong Wang, and Yuanyuan Yang. 2021. Design of Self-sustainable Wireless Sensor Networks with Energy Harvesting and Wireless Charging. *ACM Trans. Sen. Netw.* 17, 4, Article 45 (June 2021), 38 pages.
https://doi.org/10.1145/3459081

1 INTRODUCTION

Energy provisioning is critical in **wireless sensor networks (WSNs)**, since common approaches of energy conservation [1–3] do not address the problem at the source. As battery gets depleted ultimately, network service is disrupted. This creates new challenges in many applications, e.g., sensors scattered in large mountainous areas for wildfire monitoring in California [4]; sensors deployed in Fukushima nuclear reactors to monitor radiation levels after earthquake [5]; sensors surrounding volcanos to detect early signs of eruption in Indonesia. All these applications are

Authors' addresses: P. Zhou and Y. Yang, Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY, 11794-2350; emails: colinzpz@gmail.com, yuanyuan.yang@stonybrook.edu; C. Wang, Department of Computer Science, Old Dominion University, 3214 Engineering and Computer Science Building, Norfolk, VA, 23529; email: c1wang@odu.edu.

1550-4859/2021/06-ART45 $15.00
https://doi.org/10.1145/3459081

mission-critical and require continuous, uninterrupted services. Replacing sensor battery becomes infeasible in these hazardous environments. To this end, a key step is to make sensor networks self-sustainable on energy.

There exists two main strategies to power sensors: either through *energy harvesting* or *wireless charging*.[1] Ambient energy sources such as solar/wind typically enjoy higher power density and multiple sources can be combined for higher energy generation [6–8]. Yet, they are subject to the micro-climate variation that fluctuates rapidly due to spatial-temporal factors. For instance, cloud movements, sunlight angle, foliage shades, building obstructions, temperature and humidity all have impacts on the harvested energy. During rainy days, solar irradiance is limited; in hot and humid season, wind could completely stop for days. Thus, it is critical for the system to be cognizant of the spatial and temporal characteristics of the weather, and plan sensing, data processing, and communication accordingly. Under extreme circumstances, environmental energy is still not enough to guarantee robustness, even multiple sources are combined.

Wireless charging provides a more reliable energy source, as evidenced in Reference [9] that energy can be delivered via **Mobile Chargers (MCs)**. Since the MC typically serves one or a few sensors in the vicinity, scalability becomes an issue for large-scale networks. If a fraction of the sensors can become self-sufficient on energy (such as harvesting energy from the environment), then it is possible to cover the rest of the network by wireless-rechargeable sensors, thereby only one MC needed. From the perspective of operational cost, the MC should also replenish its own energy. Most of the previous works redirect the MC back to the base station with connection to the power grid [10]. However, such infrastructure is unavailable in ad hoc applications such as environmental and natural disaster monitoring. To address this challenge, energy **harvesting stations (HS)** can be deployed in the field to store enough energy with large harvesting equipment. When the MC depletes its energy, it recharges its own battery at the HS so the network no longer relies on electricity from the power grid. This way, we form a closed loop by employing the MC to deliver energy harvested from the stations to wireless-rechargeable sensors, and make the network self-sustained on energy.

This new framework integrates components from a variety of sensors and equipment, thus entailing a holistic approach to tackle a series of design and optimization challenges. In particular, we explore solutions to the following questions. First, how to determine the optimal combination of sensors harvesting different types of energy, regarding their energy profiles? Second, where to deploy HS to maximize energy output while minimizing moving cost of the MC? Third, how to schedule activities of the MC to respond to charging requests and replenish its own battery at HS? Finally, how to dynamically adjust the operations based on the real-time energy profile, and jointly optimize different modules in the framework?

We propose a suite of algorithms to address these issues in a systematic manner. First, we determine the percentages of different types of sensors by formulating a *sensor composition problem* and solve it optimally in polynomial time [11]. Second, we propose an algorithm that can partition sensing fields into equal-size regions, and deploy HS by jointly considering potential moving cost of the MC and spatial-temporal energy distributions. Third, we formulate the charging schedules of the MC into a variant of the *Interval Scheduling Problem* [12], and propose a three-factor approximation algorithm. To enhance the results from static optimization based on historical data, we further combine offline prediction with stochastic online algorithms to update the HS locations, sensor composition, and field division dynamically. Finally, we jointly optimize the composition

[1]Energy harvesting mainly harvests ambient energy while wireless charging is a special kind of energy harvesting via controlled inductance/RF radiation.

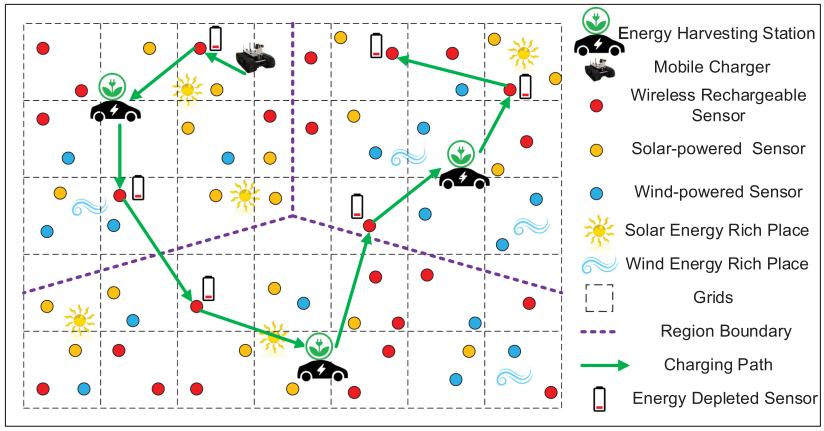


Fig. 1. An overview of network components.

of sensors with HS deployment as a multi-objective optimization problem. The contributions of this article are summarized below.

- We find the optimal combination of sensors given the ambient energy profiles, with the minimum deploying cost.
- We study the optimal placement of HS to achieve the maximal energy output and optimal sensor coverage.
- We propose a 4-approximation algorithm to plan MC's activity by jointly considering sensor charging and its own energy replenishment
- We design a stochastic online algorithm to jointly optimize and facilitate the interactions between different modules.
- Our evaluation indicates tremendous extensions of network lifetime (3×) and charging capability (2×), as well as additional cost savings (30%) and energy storage (15%). To the best of our knowledge, this is the first work that aims at the design and optimization of a *self-sustainable* WSN, powered by both multiple ambient energy source and wireless charging.

The rest of the article is organized as follows. Section 2 presents the network model, assumptions and motivations. Section 3 solves the optimal sensor composition problem. Section 4 investigates the optimal deployment of HS. Section 5 schedules activities of the MC. Section 6 dynamically adjusts network operations and performs joint optimization. Section 7 evaluates the performance. Section 8 studies literature and Section 9 concludes the article.

2 PRELIMINARY

This section presents an overview of the architecture, network components, assumptions, and the preliminary analysis that motivates this work.

2.1 Network Model and Assumptions

The network consists of a combination of sensors that can harvest energy from different sources. The ideal setting is to equip each sensor with all kinds of harvesting devices, at a high cost as well as increased system complexity. To remain under a reasonable budget, we assume each sensor is equipped with one energy harvesting device. Although it may not fully utilize the environmental energy, the proposed solution is more cost-effective for large networks and different types of

sensors can be deployed interchangeably to adapt to the local energy distribution. The main components in the network are defined in the following.

Energy Harvesting Sensors. We mainly consider three types of harvesting devices: *solar*, *wind*, *wireless-rechargeable* sensors. Specifically, to ensure efficiency, we adopt inductive-based wireless charging. We quote prices of sensors on the marketplace of Amazon and find their average prices of $50, $35, and $30 for solar, wind, and wireless-rechargeable sensors, respectively. The cost savings of forming different sensor composition can be used to establish energy harvesting stations, which provide energy for the MC.

Mobile Charger. The MC serves as a carrier to deliver energy from the HS to wireless-rechargeable sensors. It roams around the sensing field to receive energy requests, schedule and deliver energy to fulfill those requests. Once its own energy is about to deplete, it returns to one of the HS to recharge its own battery.

Energy Harvesting Stations. The HS is a simple device with two major components: a large capacity battery and solar-wind harvesting device. They can harvest different kinds of energy at the same time and store enough energy for the MC. Previous research assumes such facility is in the vicinity of the power grid. Unfortunately, it is not only sub-optimal for the MC (higher moving cost), but also limited in ad hoc environment with no access to the power infrastructure.

Figure 1 gives an example of a field divided into three regions with one HS in each. We first assume the micro-climate data are available, since it is not difficult to obtain in practice through some preliminary surveys. Then we account for the cases when such historical data is not available. We assume the HS have limited mobility that can change their locations infrequently.

2.2 Overview of Framework

Figure 2 describes the operations in the proposed system. Our strategy is to first come up with a solution for each module and then jointly optimize some of them together. The solution begins with an optimal planning problem: ❶ that finds the appropriate proportions of sensors to satisfy the task and energy demands, by solving a sensor composition problem. ❷ The sensing field is divided into regions of similar size, and ❸ HS are deployed to balance ambient energy and distance to wireless-rechargeable sensors. ❹ With the locations of HS, the MC schedules its activity of charging sensors and returns to HS for battery refill. After the optimization is done for each step, ❺ the prediction engine forecasts the energy profiles, and dynamically adjusts the modules between ❶ to ❸, based on the energy data collected. ❻ Tier 1 and tier 2 are jointly optimized using the number of wireless-rechargeable sensors that are active.

2.3 Preliminary Analysis

Although ambient energy such as wind and solar might provide spatial-temporal compensation to each other, in this subsection, we demonstrate that wireless-rechargeable sensors are still necessary to make the network self-sustained on energy. The preliminary results are derived based on the public dataset collected from NREL National Wind Technology Center (M2 Tower) [40] (from both solar irradiance and wind speed) located in Coal Creek Canyon, Colorado.

We calculate wind energy based on the speed from the dataset using [41],

$$P_{\text{wind}} = \frac{1}{2}\rho A v^3 C_p, \tag{1}$$

where P_{wind} is the output power of wind, ρ is the air density, A is the swept area of the wind turbine blades, v is the wind speed, and C_p is the power efficiency of converting wind energy to electricity. The C_p value is bounded by the *Betz Limit* of 0.59, and for the practical wind turbine, the C_p value usually lies in the range of ~0.1–0.3, and $C_p = 0.3$ is adopted here [42]. A is 150 cm^2. For solar

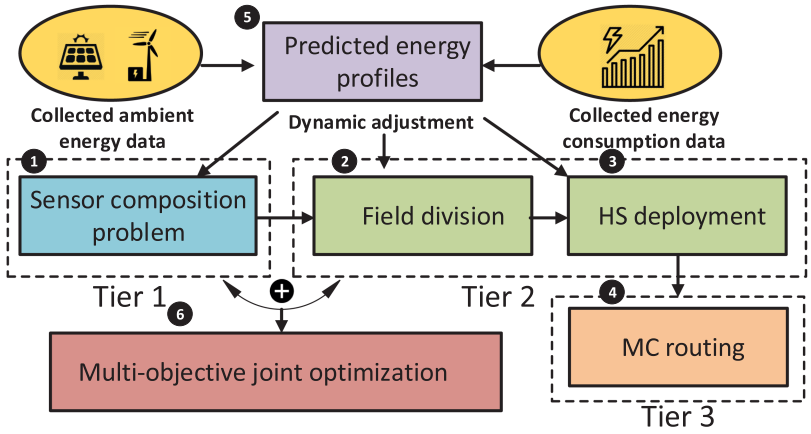


Fig. 2. System architecture of WSN with multi-source energy harvesting and wireless charging.

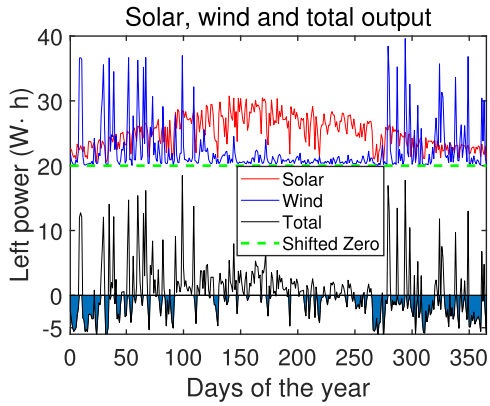
irradiance, the size of solar panel is set to 12 cm^2 (identical size of a sensor). The raw data of M2 Tower of the year 2017 is used, where solar radiance and the wind speed data is acquired at the height of 2 m (treated as the ground level).

We plot the energy evolution through a year. Figure 3(a) shows that the wind power lasts from winter to early spring, as well as autumn, but stops in summer. Interestingly, solar irradiance is able to fill the energy gap during the summer, and the expected weakness in winter is compensated by wind power. The black line computes their sum. To benchmark power consumption of sensors, we leverage the datasheet of Raspberry Pi 3 [43] and deduct this energy consumption from the sum of energy harvested. We can see that ambient energy does not always satisfy the energy demand as the deficit shown in shaded area. To guarantee network operations, the energy storage should remain positive; hence, depending on the ambient energy alone is not enough. If ambient energy is captured, stored by HS (with higher harvesting capabilities) and delivered to the wireless-rechargeable sensors via the MC, then the residual energy can be elevated to positive as shown in Figure 3(b). This would possibly allow the network to retain its functionality.

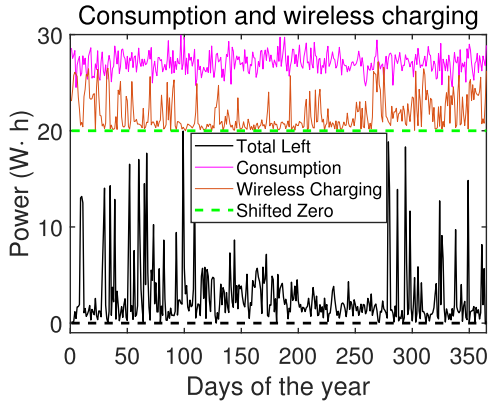
HS also has its own capacity. The HS adopted in the preliminary analysis is equipped with larger solar panel and wind turbine, with size of 600 cm^2 and swept area of 7500 cm^2, respectively. Such HS can generate an average of 442 W·h electricity per day in the area based on the weather data collected. Figure 3(c) shows the match/mismatch of energy demands from different number of sensors to the energy harvested by one HS across the entire year. We can see that an HS can handle up to 200 sensors, and barely satisfy the overwhelming energy demands from 300 sensors. Thus, the deployments of HS should also grow regarding the scale of the network.

3 SENSOR COMPOSITION PROBLEM

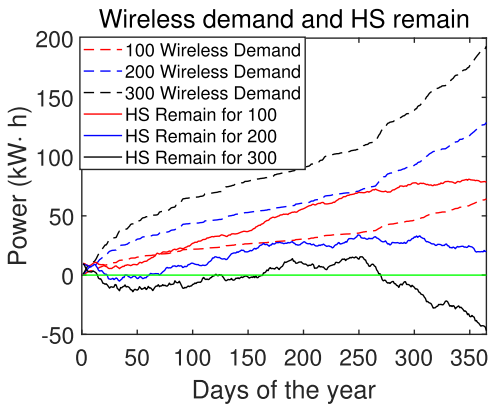
In this section, we first study the planning problem defined as the **Sensor Composition Problem (SCP)**. Before going into the technical details, we summarize those commonly used notations used in the following sections by Table 1. Different types of sensors usually have diverse manufacturing costs from our preliminary studies. Hence, we need to determine the ratio of different types of sensors to be deployed at various locations depending on the energy income (e.g., solar, wind, and wireless charging) and their cost. The objective of SCP is to minimize the total cost of sensors while making sure network energy is sufficient to support sensor activity. The inputs are the historical energy distributions (from surveyed data). Sensors work together to conduct different



(a) ambient energy only



(b) with wireless-rechargeable sensors



(c) residual energy of HS

Fig. 3. Energy status in the network.

tasks such as sensing, computing and communication. For example, the task of monitoring a target can be simultaneously performed by multiple sensors, whose sensing range covers the target. In Reference [44], the authors build a new prototype based on RFID tags to monitor the gesture of human body with great accuracy.

Table 1. List of Important Notations

Notation	Definition
y_{uv}	Grid in the uth row and vth column
n_{uv}	Number of sensors in y_{uv}
η_z^{uv}	Percentage of zth type sensors in y_{uv}
c_z	Manufacturing cost of one zth type sensor
t_k	kth time slot
τ	Flow link in SCP
q	Number of HS
$\mathcal{G}_l$	lth region
λ	Locations of HS
a	Region-size precision index
ξ	Size deviation ratio
x_i	ith charging request

To facilitate analysis, the sensing field is divided into grids.[2] y_{uv} represents the coordinate of the uth row and vth column. n_{uv} is the number of sensors to be deployed in the grid y_{uv}. There are l types of sensors ($l = 3$), where the percentages of each type of sensors are, η_1^{uv} for solar, η_2^{uv} for wind and η_3^{uv} for wireless-rechargeable sensors. Without loss of generality, the formulation here considers any l, which can be more than 3 depending on the number of available energy sources. $p_z(y_{uv}, t_k)$ is the power income from the zth energy source at location y_{uv} and time t_k. The grid partitions the field with the minimum granularity, e.g., 100 m $\times$ 100 m, so energy distribution is approximately uniform in each grid. Let $C(y_{uv}, t_k)$ denote the required energy at location y_{uv} at time t_k, composed of sensing, computing, communication and data transmission. The condition to ensure the sum of energy income is greater than or equal to the energy consumed at any arbitrary time is given below, which is not optimal in terms of manufacturing cost, since the batteries of sensors can restore the extra energy and utilize it in face of insufficient ambient energy,

$$\sum_{z=1}^{l} n_{uv}\eta_z^{uv}p_z(y_{uv}, t_k) \geq C(y_{uv}, t_k), \forall u, v, k. \tag{2}$$

To find the optimal composition of sensors while maintaining energy balance, we formulate SCP into the *network flow problem*, which finds the feasible flows meeting the demands of the sink, and involves the least cost from the sensor nodes. Shown in Figure 4, the *Source* represents all kinds of energy income, and the *Sink* represents all energy consumption tasks. There are n sensors and m tasks, where the index of sensors and tasks are denoted by i and j separately. The sensors are represented by the nodes directly connected with the source and the tasks are represented by the nodes directly connected with the sink in Figure 4. The m tasks include sensing, computing and data transmission in different grids y_{uv}. In the formulated network flow problem, there exists a link between sensor i and task j only if the task can be conducted by the sensor. For instance, for the task of monitoring a target, only the sensors within the sensing range from the target can establish links with the monitoring task. The differences of sensors are only the energy sources in the considered scenario, which can be treated functionally equivalent if only they can complete the task. In the grid y_{uv}, the flow of the link is denoted as τ_{ijz}^{uv} (z denotes the sensor type). $\tau_{ijz}^{uv}(t_k)$ is the energy consumption of type-z sensor i used for the purpose of completing task j at location

[2]Henceforth, the term *grid* refers to the square representing a specific position.

y_{uv} and time slot t_k. The energy consumption of task j is E_j^{uv}. The optimization problem can be formulated as

$$\mathbf{P1}: \quad \min \sum_{z=1}^{l} n_{uv} \eta_z^{uv} c_z. \tag{3}$$

Subject to

$$\sum_j \tau_{ijz}^{uv}(t_k) \le p_z(y_{uv}, t_k), \forall\ i, k, z, \tag{4}$$

$$\tau_{ijz}^{uv}(t_k) \ge 0, \forall\ i, j, k, z, \tag{5}$$

$$\sum_i \sum_{z=1}^{l} \tau_{ijz}^{uv}(t_k) \ge E_j^{uv}(t_k), \forall\ j, k. \tag{6}$$

Equation (3) minimizes the total manufacturing costs composed of $l = 3$ types of sensors, where η_z^{uv} is the optimization variable, n_{uv} is a fixed number satisfying Equation (2) and c_z is the given cost of deploying a sensor of the zth type. Equation (4) ensures all the tasks conducted on sensor i of type z do not consume more energy than the harvested energy $p_z(y_{uv}, t_k)$ at any time t_k. Equation (5) ensures that the energy consumed in each link must be positive. Equation (6) ensures that the energy being assigned to complete task j is more than its energy requirement $E_j^{uv}(t_k)$. $p_z(y_{uv}, t_k)$ can be derived from the historical data or offline survey conducted by the MC. The constant n_{uv} is derived by solving Equation (2) with all combinations of η_z^{uv}. The granularity of any η_z^{uv} is 1%, inferring that total combinations of η_z^{uv} is $\binom{l+99}{l-1}$ (e.g., $\binom{102}{2}$ when l=3). The smallest n_{uv} is found among all the results and used as the input of P1. A day is slotted into equal intervals t_k. The problem solves for each grid y_{uv} in the field. For given n_{uv} and η_z^{uv}, P1 is a *maximum flow problem*, which can be optimally solved by *MPM* algorithm [11]. By plugging different values of n_{uv} and η_z^{uv}, we can find the values with minimum cost according to Equation (3), which has a feasible solution for the problem. For each computation, MPM algorithm takes $O((n+m)^3)$ time. For given number of n sensors, they can be divided into three different categories, where $\binom{n+2}{2}$ number of combinations exist and each of which is plugged into Equation (3) and tested. Since $\binom{n+2}{2}$ is in the order of $O(n^2)$, the total time complexity is $O((n+m)^3 n^2)$.

Discussion. While the proposed mechanism can derive the optimal sensor composition for each grid, it also has some drawback. Since this is an offline approach that depends on the historical data of energy profiles, it may not adapt to the online scenario, where unexpected weather conditions appear. To address this, the dynamic sensor composition is introduced in Section 7.1.

4 DEPLOYMENT OF ENERGY HARVESTING STATION

The energy harvested by HS is essential to support the wireless-rechargeable sensors. Similarly to energy-harvesting sensors, their locations also determine the amount of energy they can harvest as well as the moving cost of the MC to reach them. Therefore, the goal is to find locations that are energy-rich and easily accessible by the MC. The energy-rich locations can be found from historical data. In addition to energy income, HS should be deployed close to places with more energy demands, so as to reduce moving cost of the MC. Therefore, both energy distribution and charging demands from MC should be jointly considered.

4.1 Division of Sensing Field

Given q HS satisfying the energy demands, the field of any arbitrary shape is divided into q similar-size regions represented by $\{\mathcal{G}_l\}$. The equivalent size would minimize the energy variance between different regions. This way, the network can avoid coverage holes due to energy depletion.

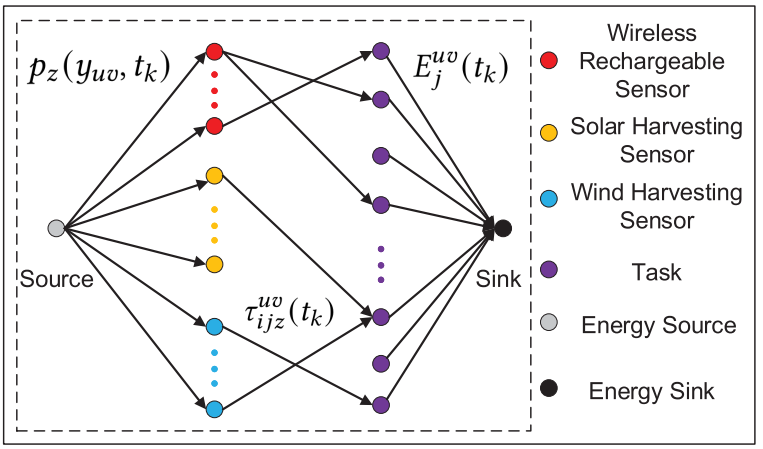


Fig. 4. Formulation of SCP into a network flow problem.

There are some trivial ways to divide the field. Take a square area for instance, it can be conveniently split into q equivalent rectangles. However, this method has some drawbacks: (1) nonuniform traveling distance in each rectangle. The distance from the center of the rectangle to the edge varies according to the directions emitted from the center. Traveling distance of the MC from sensors to the HS is unbalanced, which causes charging delay to those sensors far from the HS; (2) Uneven locations of HS (centroid of the rectangles). Their locations tend to concentrate near the center of the square, which leads to extended traveling time to service some nodes near the boundaries. Of course, an ideal case is a circular field, whereas for rectangular ones, the distances from the centroid are even larger than the maximum distance (i.e., the radius) in the circular field. Yet, circular division would leave some inevitable gaps if the overlapping of regions are not allowed. The sensors in the overlapping areas may increase the workload of multiple mobile chargers simultaneously, which reduces the charging capability of one MC. Based on these observations, the strategy should generate regions as isotropic as possible (i.e., split a field of arbitrary shape into identical sizes, which are also uniform in all orientations).

We propose a mechanism as described below. For better illustration, we take a square field as an example, whereas the general procedure can be applied to any arbitrary shape. The objective is to divide it into k^2 equal-size square grids, $k = \lceil a\sqrt{q} \rceil$, where a is an input integer called precision index. Larger a results finer granularity. Then the side length for most of the smaller square region is derived as $\lfloor k/\sqrt{q} \rfloor = \lfloor \lceil a \cdot \sqrt{q} \rceil / \sqrt{q} \rfloor$. This means that for most of the small square regions, the side length is $\lfloor k/\sqrt{q} \rfloor$ times the grid length, and each square region contains $\lfloor k/\sqrt{q} \rfloor^2$ grids. Thus, we have $\lfloor \frac{k}{\sqrt{q}} \rfloor = a$.

Next, the square field is divided into q regions. Starting from the grid on the top left corner of the field, the algorithm goes to the right by a grids, goes downward by a grids, and generates the first square region (contains a^2 grids). Then the next one is derived by going to the right. The process repeats until no more complete square region found on the right direction. The process continues downward until there is no more complete square region. For the remaining grids, starting from the grid on the bottom left corner of the field, the first a^2 adjacent grids form another region. The same process also starts from the top right corner. This process continues until the number of the remaining unassigned grids falls within the interval between $[a^2, 2a^2 - 1]$. Finally, the remaining grids form the last region. The mechanism is summarized in Algorithm 1. The proposed algorithm

ALGORITHM 1: Field division algorithm

Input: Number of regions q, precision index a, the field.
Output: q regions $\{\mathcal{G}_l\}$ dividing the field.
$k \leftarrow \lceil a\sqrt{q} \rceil$; divide field into k^2 grids $\{y_{uv}\}$, $\mathcal{Y} \leftarrow \{y_{uv}\}$. $T \leftarrow \{y_{uv} : \underset{y_{uv} \in \mathcal{Y}, \forall u}{\arg\min} \{v\}\}$, $L \leftarrow \{y_{uv} : \underset{y_{uv} \in \mathcal{Y}, \forall v}{\arg\min} \{u\}\}$, $l \leftarrow 1$.
$y_{ij} = \underset{y_{uv} \in T}{\arg\min}\{u\}$.
if $y_{i(j+a-1)} \in \mathcal{Y}$ **then**
    $\mathcal{G}_l \leftarrow \{y_{uv} : i \le u \le i + a - 1, j \le v \le j + a - 1\}$, $\mathcal{Y} \leftarrow \mathcal{Y} \setminus \mathcal{G}_l$, $l \leftarrow l + 1$, $T \leftarrow \{y_{uv} : \underset{y_{uv} \in \mathcal{Y}, \forall u}{\arg\min} \{v\}\}$,
    jump to **line 4**
$y_{ij} = \underset{y_{uv} \in L}{\arg\min}\{v\}$.
if $y_{i(j+a-1)} \in \mathcal{Y}$ ***And*** $y_{(i+a-1)j} \in \mathcal{Y}$ **then**
    jump to **line 4**
while $|\mathcal{Y}| > k^2 - (a-1)q$ **do**
    while $|\mathcal{G}_l| < a^2$ **do**
        $\mathcal{G}_l \leftarrow \mathcal{G}_l \bigcup \{y_{ij}\}$, $\mathcal{Y} \leftarrow \mathcal{Y} \setminus \{\mathcal{G}_l\}$, $T \leftarrow \{y_{uv} : \underset{y_{uv} \in \mathcal{Y}, \forall u}{\arg\min} \{v\}\}$, $y_{ij} = \underset{y_{uv} \in T}{\arg\min}\{u\}$.
    $l \leftarrow l + 1$.
    while $|\mathcal{G}_l| < a^2$ **do**
        $\mathcal{G}_l \leftarrow \mathcal{G}_l \bigcup \{y_{ij}\}$, $\mathcal{Y} \leftarrow \mathcal{Y} \setminus \{\mathcal{G}_l\}$, $L \leftarrow \{y_{uv} : \underset{y_{uv} \in \mathcal{Y}, \forall v}{\arg\min} \{u\}\}$, $y_{ij} = \underset{y_{uv} \in L}{\arg\min}\{v\}$.
$l \leftarrow l + 1$, $\mathcal{G}_l \leftarrow \mathcal{Y}$.

tends to derive more square fields, which balances the traveling distance of MC within the region and the distribution of the region centroid.

The algorithm produces most regions in square shape, some rectangles, and at most one with irregular shape. For quantitative evaluation, we further define an index ξ called *Size Deviation Ratio* as the ratio between the difference of largest and smallest region size and the mean,

$$\xi = \frac{\text{Max} - \text{Min}}{\text{Mean}}. \tag{7}$$

For the example of square field (grid area has unit 1), Min $= a^2$, and Max $= \lceil a\sqrt{q} \rceil^2 - (q-1)a^2$, and Average $= \lceil a\sqrt{q} \rceil^2 / q$. Hence, ξ is

$$\xi = q - \frac{a^2 q^2}{\lceil a\sqrt{q} \rceil^2}. \tag{8}$$

4.2 Examples of Regular and Irregular Fields

Figure 5 gives an example of the mechanism for a regular field. The field is divided into seven regions. $a = 3$, so the field contains 8×8 grids. The digits in the figure represent the order of the region being generated. The first four regions are squares. The next three regions of irregular shape contain the remaining unassigned grids. Note that the regions derived by the algorithm have comparable sizes. The size deviation ratio is 11% for this example.

The method is also applicable to irregular shapes. The precision index a can be adjusted larger to mitigate boundary effects. Figure 6 shows another example of applying the algorithm for irregular field. $a = 7$, and regions 1 to 6 are derived based on square regions, and regions 7 to 9 are derived based on irregular regions. The square shape region for irregular field is defined as the region that contains at least one square of side length equal to $\lfloor k/\sqrt{q} \rfloor$ times grid lengths. If some grids are

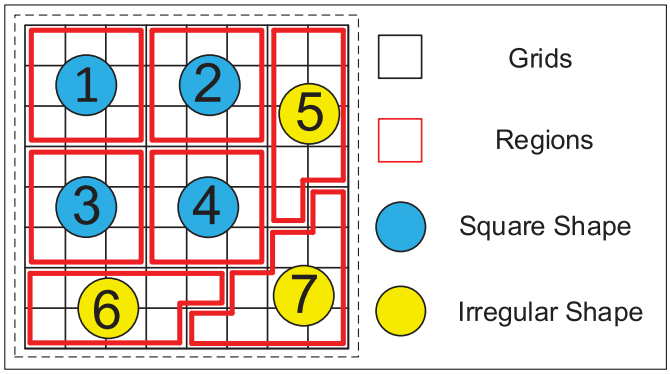


Fig. 5. Division of 8×8 square field into seven regions.

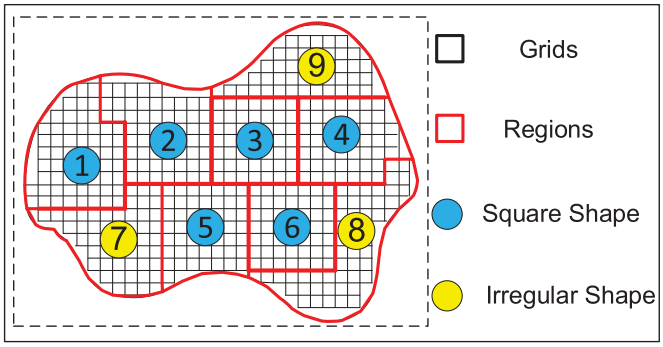


Fig. 6. Division of irregular field into nine regions.

unable to be included in the first pass, then they are assigned to adjacent regions. Similarly, those irregular areas not included in any grid are also merged into adjacent regions. Using this method, nine regions are derived with similar size, and the size deviation ratio is 16%.

Note that, due to the uncertainty of the irregular shapes of the field, starting from the top-left corner may not always derive the best partition. Therefore, all grids on the left edge of the field are applied as the starting points and their results are compared to derive the optimal. Among all the partitions, the ones with the largest number of square shapes are chosen and the deviation parameter ξ is used to break the tie (the lowest ξ is preferred). Algorithm 1 traverses all the grids and has the time complexity is $O(k^2)$. For irregular field, as much as $O(k^2)$ times of algorithm 1 need to be conducted to find the best starting point, thus resulting in $O(k^4)$ time complexity.

For the regular field, Figure 7 demonstrates the evolution of ξ with two variables q and a. For different number of regions q, as a increases, the deviation ratio is trending down (with some fluctuations). It is not monotonically decreasing, because the ceiling function introduces some rounding jitters. Having a finer granularity (larger a) is more likely to generate similar-size regions. When $q = 9$, the ratio $\xi = 0$. Since $\lfloor a\sqrt{9} \rfloor = 3a$, the region can be exactly divided into $9a^2$ equal-size square regions. Other than $q = 9$, larger q values also results larger ratios, since the difficulty of dividing the field increases with larger q. For $5 \leq q \leq 15$, $a \geq 35$ can ensure the ratio is always smaller than 20%.

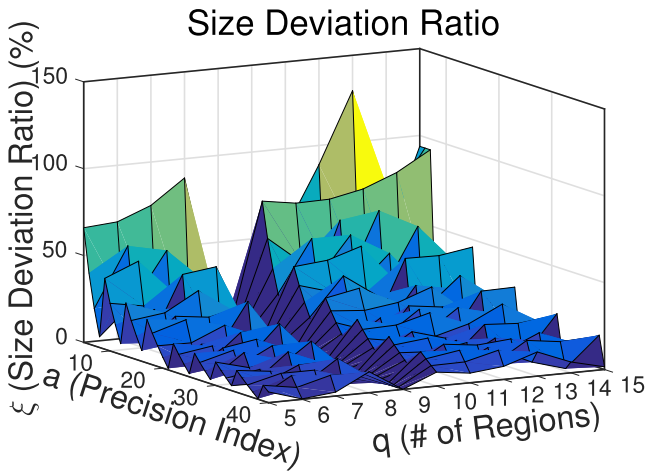


Fig. 7. Relations between size deviation ratio (ξ) with precision index (a) and number of regions (q).

4.3 Deployment of HS

After the division, HS are ready to be deployed into the q regions. Described in Section 3, the total amount and percentages of different sensors are determined independently for each grid y_{uv}, thus the density ρ_{uv} of wireless-rechargeable sensors in grid y_{uv} is not homogeneous for each grid. Using the centroid can achieve the largest coverage of the sensor groups in each grid [48] as well as balance the charging latency to reach those nodes from the centroid. For region $\mathcal{G}_l$, the centroid C_l of $\mathcal{G}_l$ is $C_l = \sum_{(u,v)\in\mathcal{G}_l} y_{uv}\rho_{uv} / \sum_{(u,v)\in\mathcal{G}_l} \rho_{uv}$. The location of HS should be close to the centroid C_l and the energy-rich places,

$$\arg\max_{\lambda} \left(\alpha(e_1(\lambda) + e_2(\lambda)) - ||\lambda - C_l||_2\right), \forall\lambda \in \mathcal{G}_l, \tag{9}$$

where the optimization variable is the deployment location, and the optimization objective is to maximize the summation of the harvested energy and the traveling convenience. $e_i(\lambda)$ denotes the expectation of the ith type energy at location x (from historical energy profiles). $||\lambda - C_l||_2$ is the deviation between λ and C_l (in proportion to the Euclidean distance). α is a scaling parameter with the unit of m/J to balance the two. The optimal location λ^* in region $\mathcal{G}_l$ that maximizes the sum of Equation (9) is selected as the location of the HS. It jointly considers the potential moving cost and energy distribution to maximize energy efficiency. The parameter α is a user-input of the network, which reflects the importance of these two components. α can be raised if the power of the harvested energy is more important for the user; otherwise, α can be reduced if the saving of the traveling cost is more crucial. Due to the spatial energy distributions being highly nonlinear, it is difficult to represent the harvestable energy in a function format and derive an iterative optimization method. Fortunately, the grid divisions are finite and the neighboring regions might have identical energy income. To solve Equation (9), we can perform an efficient grid search by enumerating these locations, and pick the one with the largest output λ^*. Since the candidate locations are limited, the complexity is in the order of $O(S)$, where S is the area of the field.

Discussion. The proposed field division algorithm is heuristic, which does not have a performance bound on the number of derived square shapes and the size deviation ratio ξ. Meanwhile, the number q of regions is a given input for the algorithm, which has a significant influence on the divisions. q may be adjusted as a parameter to derive the optimal division.

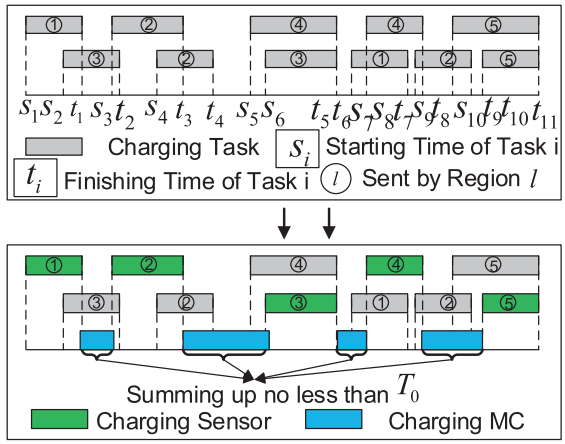


Fig. 8. Joint scheduling of sensor charging and energy replenishing by MC.

5 SCHEDULING SENSOR CHARGING AND BATTERY REPLENISHMENT FOR MOBILE CHARGER

The MC charges sensors to satisfy their energy demands as well as replenishes its own energy at the HS once the energy is depleted. This section studies scheduling of such activities.

5.1 Group Interval Scheduling Maximization

Charging requests from sensors are from various locations in the field due to dynamic spatial-temporal energy distribution. Wireless-rechargeable sensors play a key role in this situation to maintain network operation. The objective of the MC is to satisfy the charging requests from as many grids as possible instead of fulfill all the charging requests in one grid at once. Due to the introduction of multi-source energy harvesting, the charging requests of all MCs need not to be fulfilled immediately to maintain the network performance. By completing partial charging requests, the covering area of one MC can be significantly extended, which has been validated in the previous work [9]. Meanwhile, the size of each grid can be adjusted. If the percentage of serviced wireless rechargeable sensors in each grid is low, then this percentage can always be increased by decreasing the grid size in the tradeoff of the area covered by MC. Compared with previous charging mechanism [47], which tends to fulfill all the charging requests, the MC in our scenario could dispatch the energy to a wider range, saving deploying cost of more MCs.

In the meanwhile, the MC should determine an appropriate time to replenish its own battery at HS. The MC is not bundled with any specific HS, which can serve any MC if it is requested and has enough energy storage. Note that, the sensors report the time when they need to start charging based on the expected lifetime calculated via their current energy status and energy consumption rate. The MC refills its own energy at HS beforehand considering the number of charging requests to be fulfilled in the next time period.

All the energy requests contained during a time interval T are the input for the scheduling of the activities of MC in an offline manner. Once the energy of a sensor drops below the threshold, the MC does not immediately charge the sensor, but includes the sensor into those to be scheduled in the next round. A charging task x_i starts at time s_i and ends at t_i. Meanwhile, the MC needs to spend T_0 time to refill its own battery in T. The traveling distance of MC is important; however, since it takes much less compared to the charging time, it is neglected in the formulation of P2 to improve the tractability of the problem. For example, moving on a 100 m $\times$ 100 m region with a

speed of 100 m/min results 1.5 min on the longest diagonal movement. Compared to that battery charging usually takes 30–90 mins, such minimum time expense is ignored to simplify the problem. As shown in Figure 8, schedule of charging tasks may have temporal conflict with each other. Due to limited wireless charging range, the MC can only respond to one request at a time. To charge as many groups as possible, during T time, the MC finds the maximum number of grids containing non-conflicting charging tasks and leaves at least T_0 time for recharge. The problem is formulated as follows:

$$\mathbf{P2}: \quad \max_{x_i} |\mathcal{A}|, \tag{10}$$

Subject to

$$x_i = 0 \text{ or } 1, \forall i, \tag{11}$$

$$s_i \le t_i, \forall i, \tag{12}$$

$$\mathcal{R} = \{i | x_i = 1, \forall i\}, \tag{13}$$

$$t_i \le s_j, \text{ or } t_j \le s_i, \forall i, j \in \mathcal{R}, \tag{14}$$

$$\mathcal{A} = \{\mathcal{G}_l | i \in \mathcal{G}_l, \forall i \in \mathcal{R}\}, \tag{15}$$

$$T_0 \cdot P_r \ge \sum_{i \in \mathcal{R}} (t_i - s_i) \cdot P_c - E_{mc}, \tag{16}$$

$$T_0 + \sum_{i \in \mathcal{R}} (t_i - s_i) \le T. \tag{17}$$

The optimization variable x_i is the indicator showing whether task i is chosen or not. The objective is to maximize the number of grids $|\mathcal{A}|$ being covered by the MC. $\mathcal{R}$ is the sequence of the recharging tasks chosen by MC, and $\mathcal{A}$ is the set of grids where $\mathcal{R}$ originates. (14) ensures any two tasks are non-overlapping. P_r and P_c are the charging rates at HS and sensors, repectively. E_{mc} is the residual energy of the MC. P2 maximizes the number of grids to be charged in T, while constraint (16) ensures the energy replenished to the MC is larger than the energy demand of sensors, and constraint (17) imposes the time spent on charging sensors and recharge time is within T.

The optimization problem is a variation of **Group Interval Scheduling Problem (GISP)** [49]. It considers groups of tasks, where each task x_i is represented by an interval indicating its starting time s_i and finishing time t_i. A subset of all the intervals are considered to be compatible if any two of them do not have any overlapping with each other. If an interval is chosen, then it is the representative of the group containing it. GISP aims to find the subset of compatible intervals with the maximum coverage of different groups, i.e., maximize the number of groups (grids) having at least one representative in the derived subset.

Different from GISP, P2 leaves T_0 in addition to the charging tasks. The constraints (16) and (17) give the upper and lower bound for T_0. Combining them gives the following constraint:

$$\sum_{i \in \mathcal{R}} (t_i - s_i) \le \frac{T + E_{MC}/P_r}{1 + P_c/P_r}. \tag{18}$$

Therefore, if the summation of compatible intervals of different charging tasks is not larger than the right-hand side of Equation (18), then it is a feasible solution for P2. The new problem is an extension to GISP, and we call it **Reserved Group Interval Scheduling Problem (R-GISP)**.

Reserved Group Interval Scheduling Problem. Interval i starts at s_i and ends at t_i ($s_i \le t_i$), which belongs to a group G_j (i.e., interval i represents group G_j), and $|t_i - s_i|$ is the length of the interval. Two intervals are compatible if they do not overlap. For a set of intervals, R-GISP looks for the set of compatible intervals, which can represent the maximum number of groups, and whose

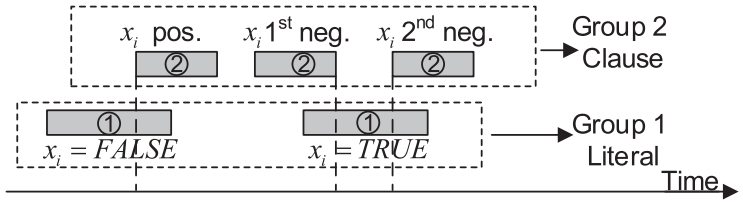


Fig. 9. The construction of GISP based on Boolean Satisfiability Problem.

length summation is less than or equal to a given constant. The R-GISP is formally defined in the following:

$$\mathbf{R-GISP}: \quad \max_{x_i} |\Psi|, \tag{19}$$

Subject to

$$x_i = 0 \text{ or } 1, \forall i, \tag{20}$$
$$s_i \leq t_i, \forall i, \tag{21}$$
$$\mathcal{R} = \{i | x_i = 1, \forall i\}, \tag{22}$$
$$t_i \leq s_j, \text{ or } t_j \leq s_i, \forall i, j \in \mathcal{R}, \tag{23}$$
$$\Psi = \{G_k | i \in G_k, \forall i \in \mathcal{R}\}, \tag{24}$$
$$\sum_{i \in \mathcal{R}} (t_i - s_i) \leq C. \tag{25}$$

x_i is the indicator showing whether interval i is chosen or not. $\mathcal{R}$ is the set of all chosen intervals. Equation (14) ensures any two intervals are compatible. Ψ is the set of groups represented by all intervals in $\mathcal{R}$. Equation (25) requires the summation of the length of all chosen intervals is no greater than a constant C.

P2 is closely related to the R-GISP. Each charging request (interval) belongs to one of the grids (groups). $\mathcal{A}$ is the set of grids whose charging requests are covered by MC. P2 maximizes the cardinality of the set $\mathcal{A}$ while satisfying the energy constraint, which is exactly the same as the definition of R-GISP.

NP-Hardness. R-GISP is NP-hard.

Proof. To prove that R-GISP is NP-hard, we first prove GISP is NP-hard and GISP can be reduced to R-GISP.

A special case of GISP is NP-hard. A special case of GISP solves the problem whether there is such compatible set that contains at least one representative from each group, i.e., the size of the groups being represented is equal to the number of groups. This special case is called **Group Interval Scheduling Decision Problem (GISDP)**. To understand the difficulty of R-GISP, the proof of the NP-hardness of GISDP is briefly illustrated below.

A special version of **Boolean Satisfiability Problem (BSP)** contains a group of literals $\mathcal{X} = \{x_1, x_2, x_3, \ldots, x_g\}$ and a group of clauses $C = \{c_1, c_2, c_3, \ldots, c_h\}$. Any literal can be TRUE or FALSE. Each clause contains three literals, e.g., $c_1 = x_1 \vee x_2 \vee \neg x_5$, where the symbol $\vee$ here means OR. x_1 and x_2 are represented positively in this clause, and x_5 is represented negatively in this clause using notation $\neg x_5$. For this example, if either x_1 is TRUE or x_2 is TRUE or x_5 is FALSE, then the clause c_1 is TRUE. For each literal in $\mathcal{X}$, it is only positively represented at most once and negatively represented at most twice overall in C. BSP asks whether there exists such assignment to the TRUE

ALGORITHM 2: Earliest Finishing First Algorithm

Input: A number of n charging requests $\mathcal{X} = \{x_i\}$, $x_i = [s_i, t_i]$, distributed in grids $\{y_{uv}\}$, an empty set $\mathcal{R}$.
Output: Charging and energy replenishment sequence $\mathcal{R}$ for MC.
$\widehat{\mathcal{X}} \leftarrow \mathcal{X}$
while $\widehat{\mathcal{X}} \neq \phi$ **do**
  $f = \arg\min_{x_i \in \mathcal{X}} \{t_i\}$, $x_f \in y_{uv}$, $\mathcal{R} \leftarrow \mathcal{R} \bigcup \{x_f\}$;
  $\widehat{\mathcal{X}} \leftarrow \widehat{\mathcal{X}} \setminus \left\{\{x_j : x_j \cap x_f \neq \phi, \forall j\} \bigcup \{x_j : x_j \in y_{uv}, \forall j\}\right\}$.
if $\sum_{i \in \mathcal{R}}(t_i - s_i) \leq [T + E_{MC}/P_r]/[1 + P_c/P_]$ **then**
  $\mathcal{X} \leftarrow \mathcal{X} \setminus \mathcal{R}$; **jump** to **line 3**.
if $\sum_{i \in \mathcal{R}}(t_i - s_i) > [T + E_{MC}/P_r]/[1 + P_c/P_]$ **then**
  while $\sum_{i \in \mathcal{R}}(t_i - s_i) > \frac{T + E_{MC}/P_r}{1 + P_c/P_r}$ **do**
    $e \leftarrow \arg\max_{x_i \in \mathcal{R}}(t_i - s_i)$; $\mathcal{R} \leftarrow \mathcal{R} \setminus \{x_e\}$
MC charges sensors following the charging sequence derived in $\mathcal{R}$.
MC replenishes its energy at the closest HS with enough energy storage during spare time.

of FALSE status of literals $\mathcal{X}$ such that $c_1 \wedge c_2 \wedge c_3 \wedge \ldots \wedge c_h$ is TRUE. The symbol $\wedge$ here means AND, therefore, $c_1 \wedge c_2 \wedge c_3 \wedge \ldots \wedge c_h$ is TRUE if and only if every clause is TRUE.

As shown in Figure 9, GISDP can be constructed from the BSP. For example, for a literal x_i, whether it is true or false corresponds to two intervals in group 1; for a clause c_j, depending on the positive or negative representation and the times of the literal appearing in the clause, group 2 contains 3 intervals. It can be proved that, the feasible assignments for the BSP are one to one correspondence with the maximum compatible set for GISDP. Since this special BSP is NP-hard [50], GISDP is also NP-hard.

GISDP is a special case of GISP where the maximum groups being represented is equal to the total number of groups so GISP is NP-hard. GISP removes constraint (18) from R-GISP (i.e., extends the right-hand side of Equation (18) to ∞) so GISP is a special case to R-GISP thus R-GISP is NP-hard. □

5.2 Earliest Finishing First Algorithm

Due to the NP-hardness of R-GISP, it is not possible to find an optimal solution in polynomial time unless P = NP. Hence, we seek solutions that can achieve an approximated ratio within the optimal one in polynomial time.

We first propose a greedy algorithm as the baseline shown in Algorithm 2. Each time, the charging task x_i with the earliest deadline t_i is chosen. All charging tasks intersecting with it are removed. Charging tasks from the same grid are also removed. This process iterates until no task is left. Second, Equation (18) is the necessary condition for continuing the Algorithm 2. Therefore, for those intervals chosen after the first step, the algorithm sums them up and compares the summation with $(T + E_{MC}/P_r)/(1 + P_c/P_r)$, which is the maximum charging time. If the summation is smaller than the maximum charging time, indicating the intervals chosen in step one are compatible with energy constraints, then it removes all the intervals overlapping with them to assure no conflict exists between chosen intervals. For the remaining intervals, the algorithm performs the first step again. If the summation is larger than the maximum charging time, then it removes the longest interval from the covering set until the summation of the remaining intervals is not larger than the maximum charging time. In the last step, for a given period T, MC performs the charging tasks in $\mathcal{R}$ according to the order of the starting time. MC replenishes its own energy at the closest HS having enough energy when idle. When MC chooses the charging station, it requests the

ALGORITHM 3: 4-Approx Shortest Interval First Algorithm

Input: A number of n charging requests $\mathcal{X} = \{x_i\}$, $x_i = [s_i, t_i]$, distributed in grids $\{y_{uv}\}$, an empty set $\mathcal{R}$.
Output: Charging and energy replenishment sequence $\mathcal{R}$ for MC.
while $\sum_{\forall x_i \in \mathcal{R}} |t_i - s_i| \leq [T + E_{MC}/P_r]/[1 + P_c/P_]$ *and* $\mathcal{X} \neq \phi$ **do**
    $x_f \leftarrow \arg\min_{x_i \in \mathcal{X}} |t_i - s_i|, x_f \in y_{uv}, \quad \mathcal{R} \leftarrow \mathcal{R} \bigcup \{x_f\}$
    $\mathcal{X} \leftarrow \mathcal{X} \setminus \left\{\{x_j : x_j \cap x_f \neq \phi, \forall j\} \bigcup \{x_j : x_j \in y_{uv}, \forall j\}\right\}$.
$\mathcal{R} \leftarrow \mathcal{R} \setminus \{$the last interval put into $\mathcal{R}\}$.
MC charges sensors following the charging sequence derived in $\mathcal{R}$.
MC replenishes its energy at the closest HS with enough energy storage during spare time.

current status of the energy storage of each HS, and the chosen station reserves enough energy for the MC to ensure the availability of energy replenishment.

An example is shown in Figure 8. Eleven charging requests from 5 different regions are received. Solving the algorithm yields that 5 requests from different regions are met so as to achieve the best region coverage. T_0 time is left for the energy replenishment of MC.

The complexity of the algorithm is analyzed below. Assume there are n charging requests from m grids. Sorting requires $O(n \log n)$ time. First, finding the intervals intersecting with the chosen one takes $O(n)$ time with at most $O(m)$ times, and removing the intervals takes $O(n)$ time. Therefore, the time complexity for the first step is $O(n \log n) + O(nm)$. For the second part, summing up the intervals and comparing with the maximum charging time takes $O(n)$ time, and this process takes at most m times, therefore $O(nm)$ in total. The procedure is repeated at most n/m times and the total time complexity is $O(n^2 \log n/m)$.

5.3 4-Approximation Algorithm

We further propose a new algorithm with 4-approximation ratio as shown in Algorithm 3. As far as we know, this is the first approximation algorithm with constant ratio for the R-GISP problem. The previous earliest finishing first algorithm does not have a theoretical bound of performance, because choosing the tasks with the earliest finishing time does not consider the duration of tasks at all. Without a performance guarantee, the ratio between the optimal solution and the result derived is unbounded (can be arbitrarily large in the worst case), and we cannot gain any insight about the optimal solution from the results either.

Instead of looking for the earliest finishing time, the new algorithm seeks the shortest interval. For a set of tasks, the algorithm picks the one with the shortest duration, and removes the tasks intersecting with the chosen task from the set of tasks. If the total length of chosen tasks is smaller than $(T + E_{mc}/P_r)/(1 + P_c/P_r)$, then continue the process for the remaining set until the largest set of tasks whose total length is smaller than the preset limit $\frac{T+E_{MC}/P_r}{1+P_c/P_r}$.

Approximation Ratio. The Shortest Interval First Algorithm has 4-factor approximation for R-GISP.

Proof. Denote the optimal solution for R-GISP as *OPT*, and the solution derived by the algorithm as *SIF*. According to Algorithm 3, once a charging task is chosen, all the charging tasks belonging to the same grid cell as the chosen one are removed from the pending list. Therefore, there is no two intervals in *SIF* belonging to the same grid, which infers that $|SIF|$ is equal to the number of grids covered by applying the algorithm. If *OPT* contains more than two intervals belonging to the same grid, then the extra intervals can always be removed from the solution and a new optimal value can be derived with exactly one interval chosen in each grid while maintaining $|OPT|$. Due to the observations, it is sufficient to prove the approximation ratio of the number of intervals between *SIF* and *OPT* to show the same ratio holds for the number of covered grids.

The function f maps any interval $I \in OPT$ to an interval in SIF according to the following rule,

$$f(I) = \begin{cases} \text{❶: } I, \quad \textbf{if } I \in SIF; \\ \text{❷: Shortest interval in } SIF \text{ intersecting with } I, \\ \textbf{if } I \notin SIF \text{ and } I \text{ intersects with intervals in } SIF; \\ \text{❸: The longest interval in } SIF, \ \textbf{if } I \notin SIF \text{ and } I \\ \text{intersects with no interval in SIF.} \end{cases}$$

The above mapping function from OPT to SIF has three cases. For any interval J in SIF, if $J \in OPT$, then there is only one interval in OPT that could be mapped to SIF according to the compatible property of intervals in OPT.

If $J \notin OPT$ but J intersects with intervals in OPT, then there are at most $2 + X$ intervals in OPT that could be mapped to J via function $f(I)$. As shown in Figure 10, X denotes the number of intervals $I' \in OPT$ shorter than J and totally covered by J. Since J is the shortest interval chosen in each round, the shorter interval I' could exist but is not chosen by SIF only if there has already been an interval J' chosen by SIF, which belongs to the same grid containing I'. In such case, I' cannot be chosen by SIF anymore, since there is at most one interval to be chosen in each grid cell. Therefore, each I' corresponds to one unique interval J' in SIF, and the summation of such X number is no more than $|SIF|$.

For the longest interval J_{max} in SIF, some number of intervals in OPT are mapped to it according to the third case of $f(I)$, which does not intersect with any interval in SIF. If such interval I belongs to a grid already covered by SIF, then there always exists one interval of SIF in the same grid that is shorter than I, since algorithm 3 always chooses the shortest non-overlapping interval in a new grid. If such interval I belongs to a grid not covered by SIF, then adding I to SIF will violate the energy requirement Equation (18) due to the termination condition of SIF. Assuming the number of such I's is larger than or equal to $|SIF| + 1$, the total duration of all these intervals cannot be smaller than the duration of SIF adding up the shortest non-overlapping interval in a new grid (i.e., the $(|SIF|+1)$-th grid to be covered). However, the algorithm 3 terminates for not being able to find any non-overlapping interval that satisfies the energy requirement Equation (18). Therefore, it is inferred that the total duration of OPT violates constraint Equation (18), which yields an infeasible solution. Therefore, the number of such I's cannot be larger than $|SIF|$.

For any interval (besides of the longest) in SIF, $f(I)$ maps at most $2 + X$ intervals in OPT to it, and the summation of all such X is no greater than $|SIF|$. For the longest interval J_{max} in SIF, $f(I)$ maps at most $|SIF|$ intervals to it. Since any interval in OPT is mapped to an interval in SIF, $|OPT| \leq 4|SIF|$. □

A more intuitive way to understand the performance bound is illustrated as follows. Algorithm 2 chooses the interval with the earliest finishing time, whose duration has no guarantee. In the worst case, completing the first chosen task will use up all the time, causing the performance to be unbounded. Contrarily, the SIF Algorithm focuses on the interval duration and tends to choose the shortest non-overlapping interval from each grid. If there exists an optimal solution having no performance bound compared with SIF Algorithm, then it necessitates numerous small intervals that cannot be chosen by the SIF algorithm due to overlapping. However, as shown in Figure 10 one interval chosen by SIF algorithm can overlap with at most $2 + X$ non-overlapping smaller intervals. Note that, each I' interval corresponds to one unique interval J' in SIF. Therefore, there cannot be arbitrarily many intervals missed by the SIF algorithm due to overlapping. This intuitively explains the existence of the performance bound and infers the 4-approximation ratio.

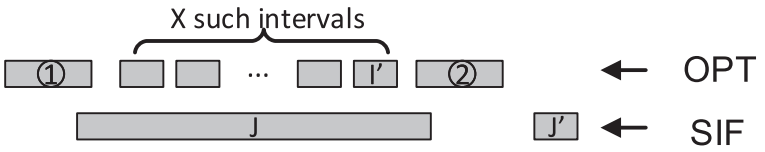


Fig. 10. Intuitive explanation of the approximation ratio of SIF algorithm.

The cardinality of the completed charing tasks of one MC is proportional to the area of the sensor networks covered by MCs for fixed sensor density. Since Algorithm 2 has no guaranteed performance bound compared with the optimal, the serviced network area can be arbitrarily small. With a guaranteed approximation ratio of 3, Algorithm 3 ensures that the serviced area of the network is at least one third of the optimal.

The time complexity is analyzed below. For n charging requests, sorting them, finding the shortest interval and removing the intersecting ones take $O(n \log n)$, and this process is conducted at most n times. The total time complexity is $O(n^2 \log n)$, which is $O(\frac{\log n}{\log n/m})$ times of Algorithm 2, but with the constant approximation ratio.

Discussion. The proposed algorithm aims to maximize the number of grids (area) covered by MC. However, for grids containing a large amount of charging requests, their requests may not be satisfied immediately via MC charging. While the proposed algorithms focuses more on the global perspective, the delay of charging for some local requests is the tradeoff. Moreover, the algorithm focuses on the routing of a single MC, which can be extended to schedule multiple MCs. Like [57], NDN-based energy aggregation and gathering protocols can be applied to satisfy the recharging needs for multiple mobile chargers. Equation (14) can be further relaxed only if the number of overlapping tasks is constrained by the number of deployed MCs.

6 DYNAMIC SCHEDULING

The previous sections have discussed optimizations based on empirical observations. However, the dynamic nature of those ambient energy sources may render the previously "optimized" solutions sub-optimal, when real-time status deviates from the historical average. This section provides an extension to account for the inherent dynamics during runtime.

6.1 Prediction-based Dynamic Adjustment

The (static) sensor composition and location are studied in the context across a large timescale (months or years). Nevertheless, some applications may impose non-stopped service requirement during a short timescale (e.g., flooding that lasts for days). Meanwhile, these situations may lead to high deviations from the average energy profile, that requires timely re-arrangement; otherwise, the unmatched energy distribution and deployment plan would easily cause inefficiency or even waste of ambient energy. Therefore, it is necessary to extend the proposed methods to consider the dynamic scenarios.

We leverage prediction algorithms to forecast energy income in the short-term and adjust network planning accordingly. The adjustment is based on two criteria, the predicted harvestable energy and energy consumption of various tasks in the next time period. State-of-the-art machine learning algorithms such as **Long Short Term Memory (LSTM)** network [58] can be applied for sequence learning (we benchmark its performance against two other statistical methods in the evaluations). Based on the energy forecast, the dynamic schemes for the HS position, sensor composition, and field division are studied.

ALGORITHM 4: Stochastic Adjustment Algorithm

Input: Predicted energy consumption $E(t_{k+1})$, accumulated energy consumption $\widetilde{E}(t_k)$, predicted harvestable energy $p_z(t_{k+1})$, accumulated harvestable energy $\widetilde{p_z}(t_k)$ for the previous time period, total number of deployed sensors n, and the sensor composition for the last time period $\eta_z(t_k)$, set of sensors $\mathcal{N} = \{n\}$.
Output: Dynamic strategies of working modes for sensors.
for $n \in \mathcal{N}$ **do**
  for *setting={offline, online }* **do**
    if *setting=offline* **then**
      $E \leftarrow E(t_{k+1}), p \leftarrow p_z(t_{k+1})$
    else
      $E \leftarrow \widetilde{E}(t_k), p \leftarrow \widetilde{p_z}(t_k)$
    Based on E and p, derive the percentages $\eta_z(t_{k+1})$ for zth type sensors by solving the Sensor Composition Problem.
    if $\eta_z(t_{k+1}) < \eta_z(t_k)$ **then**
      Sort zth type sensors in the ascending order of the remained energy;
      Change the first $\lfloor n \cdot (\eta_z(t_{k+1}) - \eta_z(t_k)) \rfloor$ sensors in the order to model 1.
    else
      $w_z(t_{k+1}) \leftarrow \frac{\eta_i(t_{k+1})}{\eta_i(t_k)} \cdot w_i(t_k)$.
      Change all zth type sensors to the largest working mode with power lower than $w_z(t_{k+1})$;
      $w_z(t_{k+1}) \leftarrow$ the power of the chosen working mode.
    $i \leftarrow$ offline working mode, $j \leftarrow$ online working mode.
  $PR_{\text{off}} \leftarrow \frac{1}{|i-j|+1}$, $PR_{\text{on}} \leftarrow 1 - \frac{1}{|i-j|+1}$;
  with probability PR_{off}, change the mode of sensors to mode i, and with probability PR_{on}, change the mode of sensors to mode j.

Dynamic HS Position. The new position of HS is jointly computed based on the predicted energy income and consumption, and re-located to the new position following

$$\arg\max_{\lambda} \left(\alpha(\overline{e_1}(\lambda, t_k) + \overline{e_2}(\lambda, t_k)) - ||\lambda - C_l(t_k)||_2\right), \forall \lambda \in \mathcal{G}_l, \tag{26}$$

where $\overline{e_1}(\lambda, t_k)$ and $\overline{e_2}(\lambda, t_k)$ are the average predicted solar and wind energy for the t_kth period. $C_l(t_k) = \sum_{(u,v)\in\mathcal{G}_l} y_{uv}\rho_{uv}(t_k) / \sum_{(u,v)\in\mathcal{G}_l} \rho_{uv}(t_k)$, where $C_l(t_k)$ is the dynamic center of region $\mathcal{G}_l$ at time t_k, since the percentage of wireless-rechargeable sensors $\rho_{uv}(t_k)$ also change dynamically. The process of solving Equation (26) is given in Section 4.3. The runtime overhead of this process is proportional to the frequency f of the dynamic positioning. Solving (26) has the time complexity of $O(S)$, so the total time complexity is in the order of $O(fS)$.

Dynamic Sensor Composition. According to the predicted energy profile, the sensor composition problem can be solved by applying the method proposed in Section 3. The modes of sensors are classified into several categories according to the power consumption of different states [43]. If the percentage of one kind of sensors is larger than the previous percentage, then some of the abundant sensors are turned into sleeping mode to save energy, while the energy harvester can still collect energy. The sensors temporarily wake up to utilize the harvested energy, enable more efficient use of the currently abundant ambient energy.

For fixed energy consumptions and the deploying cost of sensors, solving the sensor composition problem should reach appropriate percentages of sensors regarding the energy income, i.e., higher percentage of one kind of sensors if the corresponding harvestable energy is higher; lower percentage if the energy is less. Therefore, if $\eta_i(t_{k+1}) < \eta_i(t_k)$ (the percentage $\eta_i(t_{k+1})$ of ith type of sensors for the next time period is smaller than the percentage for the previous period),

then the extra sensors turn into sleeping mode in the ascending order of the remained energy. If $\eta_i(t_{k+1}) \geq \eta_i(t_k)$, then the working power of these sensors can be increased to $\frac{\eta_i(t_{k+1})}{\eta_i(t_k)} \cdot w_i(t_k)$. Compare the updated power of sensors with the power of each working mode, and the sensors are turned into the highest working mode whose power is no larger than $\frac{\eta_i(t_{k+1})}{\eta_i(t_k)} \cdot w_i(t_k)$. These calculated working modes are the offline adjustment strategies, and the sensors follow this strategy in the next period. This process needs to compare $\eta_i(t_k)$ with $\eta_i(t_{k+1})$ for z types of energy sources in each time interval. Therefore, the total runtime overhead is $O(zf(n+m)^3n^2)$.

Stochastic Online Adjustment. Note that the machine learning algorithms may perform poorly facing unexpected changes of ambient energy and urgent tasks. For example, the data analytics reflect obvious increment of prediction errors for solar irradiance during summer due to frequent thunder storms. Similarly, wind power also exhibits unforeseen increment of wind speed as a normal deviation from the seasonal trend. To better utilize the real time with predictions for better decisions, we further develop a stochastic online algorithm that can alternate between the offline and online data. The offline and online modes are the dynamic working modes of the sensors determined based on the predicted and actual energy consumption and harvestable energy.

Note that it is not designed in an optimal sense that should respond perfectly to abrupt changes (either looking backwards or forwards), but with certain probabilistic measures to avoid being trapped at a previous optimum, and make necessary explorations of the new energy patterns.

Based on the cumulative power of harvested energy and energy consumption in the past period, the sensor composition problem is solved online. The offline mode is determined by the predicted data tending to reflect long-term pattern; while the online mode determined by the actual data tends to reflect abrupt changes. These two modes have their own advantages, which motivates the combinations of these two schemes. To coordinate the offline and online strategies, a probability is introduced, which indicates how likely sensors would change to the mode regulated by the policy. Different integer numbers represent certain working modes, where mode 1 represents the sleeping mode and larger number corresponds to working mode with higher power consumption. If the offline and the online mode for one sensor is i and j respectively ($1 \leq i, j \leq 5$ [43]), then the probability for offline and online strategy is

$$PR_{\text{off}} = \frac{1}{|i-j|+1}, \; PR_{\text{on}} = 1 - PR_{\text{off}}. \tag{27}$$

In the trivial case, if j is the same as i, either one is performed, because they are equivalent. If j is different from i, larger deviation of j from i corresponds to larger prediction error of energy, which increases the probability of selecting the online strategy. Small deviation of j from i means the prediction is precise, and the system tends to choose the offline strategy. For example, if the offline and the online mode is 1 and 2 respectively, indicating the tendency to reduce the power obeying either offline or online strategy, the sensors will be kept in low-power mode; if the offline and online mode is 1 and 5, respectively, indicating urgent requirement for the high-power mode based on the real-time data, the sensors will have much lower probability to remain idle. The algorithm is summarized in Algorithm 4 and evaluated in Section 7.6. Since the determination of the offline or online working mode takes $O(1)$ time, the time complexity for stochastic online adjustment is in the order of $O(zf(n+m)^3n^2)$.

Note that, the dynamic adjustment of sensor composition is realized via the change of sensors' working modes, which may not always be available when the desiring working mode exceeds the largest working mode. Under such circumstances, the network falls short achieve the optimal sensor composition, and the deficit can be resolved by adding more sensors to the network via redeployment as shown in References [39, 61].

ALGORITHM 5: Dynamic Field Division Algorithm

Input: The similar-size divisions $\{\mathcal{G}_l\}$ of the field, the sum of predicted harvestable energy $P_l(t_{k+1})$ for region $\mathcal{G}_l$, and the grids of the field $\mathcal{Y} = \{y_{uv}\}$.
Output: Dynamic field division based on the energy income.
Sort $P_l(t_{k+1})$; find the region $\mathcal{G}_H$ with the highest energy.
while $P_H(t_{k+1}) > P_l(t_{k+1}),\ \forall l \in \{$*Neighboring regions of* $\mathcal{G}_H\}$ **do**
    In the neighboring regions of $\mathcal{G}_H$, find the region $\mathcal{G}_L$ with the Lowest energy.
    Randomly pick y s.t. $y \in \mathcal{Y}$ and $y \in (\mathcal{G}_H \bigcap \mathcal{G}_L)$
    $\mathcal{G}_H \leftarrow \mathcal{G}_H \setminus \{y\}$, $\mathcal{G}_L \leftarrow \mathcal{G}_L \bigcup \{y\}$;
    Update $P_H(t_{k+1})$ and $P_L(t_{k+1})$.

Dynamic Field Division. Previously, Algorithm 1 can divide the field of arbitrary shapes into similar size sub-regions. To adapt to the dynamic adjustment of sensor compositions and make better usage of the varying energy distributions across the field, dynamic field division is proposed. The algorithm jointly considers the size and the energy income of each region, and aims to achieve the division with similar region size and energy input.

By applying Algorithm 1, the field is divided into similar size regions, and each region shares boundary with at least one neighboring region. Summing up the predicted energy income for each region finds the total harvested energy in the next time period. The algorithm then starts from the region with the highest energy income. Among the neighboring regions with the highest energy income, the one with the lowest energy income is chosen from the neighbors. The goal is to "dilute" the regions with high-energy concentration into the neighboring ones. A grid of the highest-energy region is randomly chosen on the contour between the highest-energy region and its lowest-energy neighbor, and swapped. The process is continued until the highest-energy region is no longer dominant in the neighborhood. After a round of this process, the new highest-energy region is chosen again, and repeated until there is no more region is available for swapping. The procedure is summarized in Algorithm 5.

The algorithm has time complexity $O(k^2)$. For q regions, sorting takes $O(q \log q)$, since each swapping process terminates until the highest-energy region is no longer the highest among its neighboring, then each grid can only be swapped at most once, and the total swapping takes time $O(k^2)$. Since $q \log q$ is smaller than k^2, the total time complexity of the algorithm is $O(k^2)$.

6.2 Joint Optimization

There are opportunities for further improvement depending on the energy income in the short-term. For example, when ambient energy is sufficient, the role of wireless-rechargeable sensors is weakened so an ideal solution is to get them recycled. Since it is difficult to retrieve those sensors from the outdoor environments, a reasonable way is to deactivate some of them via sending control signals to save the subsequent maintenance cost of the MC. However, deactivation reduces the utility of sensors. Thus, there exists a tension between these conflicting objectives in the framework between #1 sensor composition problem and #2 HS deployment. Here, we optimize them jointly, and adaptively activate/deactivate wireless-rechargeable sensors with the deployment of HS. Note that #3 of charge scheduling aims to maximize the number of grids being covered by MC, which is independent with #1,2, since there is no such variable that can affect # 3 and #1/#2 simultaneously. Hence, the joint optimization as shown in Figure 2 focuses on #1,2.

The utility of one region is defined as the manufacturing cost for all the sensors included in the area. Since the cost for the already deployed sensors cannot be retrieved, the joint optimization intends to maximize the usage of these sensors. Therefore, in the utility perspective, the percentage

of activated **wireless-rechargeable (WR)** sensors should be maximized (for #1). However, the objective of HS deployment that minimizes the traveling cost of MC should expect fewer WR sensors being activated. We formulate it into a multi-objective optimization problem. The first objective $f_1(\Theta)$ is the utility reduction if the set Θ of WR sensors is deactivated. The second objective $f_2(\Theta)$ is the traveling cost from the HS to all the WR sensors. The feasible set $\mathcal{N}$ comprised of all the WR sensors,

$$\mathbf{P3}: \quad \min_{\Theta \subset \mathcal{N}} (f_1(\Theta), f_2(\Theta)), \tag{28}$$

Subject to

$$f_1(\Theta) = \sum_{(u,v)\in\mathcal{G}_l} (n_{uv}\eta_3^{uv} - \theta_{uv}) \cdot c_3, \theta_{uv} \in \Theta, \tag{29}$$

$$f_2(\Theta) = \beta \left[||\lambda^* - C_l||_2 \sqrt{\frac{\sum_{(u,v)\in\mathcal{G}_l} \theta_{uv}}{\sum_{(u,v)\in\mathcal{G}_l} n_{uv}\eta_3^{uv}}} - \alpha(e_1(\lambda^*) + e_2(\lambda^*)) \right], \theta_{uv} \in \Theta. \tag{30}$$

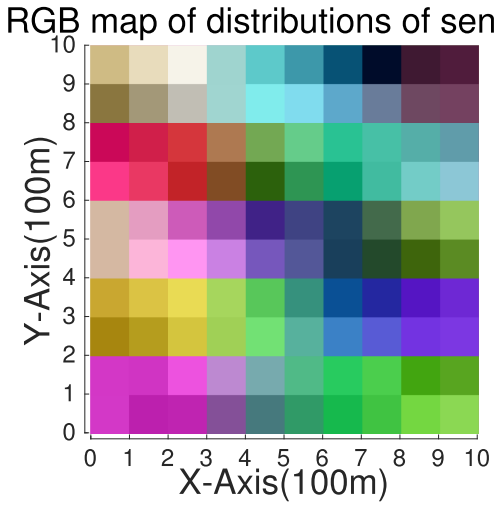
The optimization variable is the set of activated WR sensors $\Theta \in \mathcal{N}$, and the objective is to minimize the cost of utility waste $f_1(\Theta)$ and the traveling cost $f_2(\Theta)$ simultaneously. θ_{uv} is the number of activated WR sensors, and $n_{uv}\eta_3^{uv}$ is the number of deployed WR sensors, whose difference multiplying the cost c_3 of deploying a WR sensor gives the cost of utility waste. For a rectangular field, the shortest path traversing n sensors is derived as $\sqrt{2(n-2)D_1D_2} + 2(D_1 + D_2)$ [9, 53], where D_1 and D_2 is the side length of the field. Therefore, the ratio of the traveling distance for MC covering sensors in Θ to the distance for MC covering sensors in $\mathcal{N}$ can be approximated by $\sqrt{\frac{\sum_{(u,v)\in\mathcal{G}_l} \theta_{uv}}{\sum_{(u,v)\in\mathcal{G}_l} n_{uv}\eta_3^{uv}}}$ [9]. Multiplying the ratio with original traveling distance $||\lambda^* - C_l||_2$ and a scalar parameter β (in the unit of \$/m) gives the corresponding traveling distance denoted by $f_2(\Theta)$.

However, since the conflicting nature of $f_1(\Theta)$ and $f_2(\Theta)$, it is difficult to find a bi-optimal solution. Therefore, the Pareto optimality is considered instead, which represents the set of solutions that cannot be improved in any of the two objectives without degrading the other one. $f_1(\Theta)$ is proportional to the number of deactivated sensors. For the same number of sensors, removing the sensors furthest from the HS leads to Pareto optimality, because their charging cost dominates other sensors. If the number of wireless-rechargeable sensors is n, then the cardinality of the Pareto optimal set is also n. We can traverse this set and find the solution with the minimum summation of Equation (28). This step can finish within $O(n)$ time.

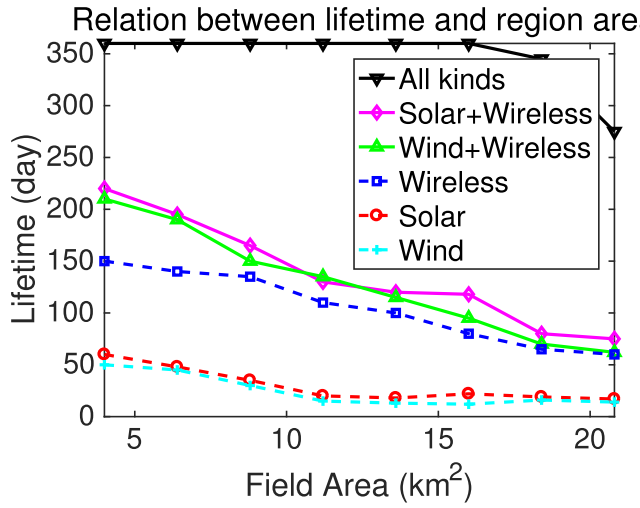
7 PERFORMANCE EVALUATIONS

We evaluate the performance of the self-sustained framework by a discrete-event simulator developed in MATLAB and compare with the previous work that depends on single energy source [51]. In the simulation, we use data trace of solar radiation from *SOLARGIS* [55] and wind power from *NREL* [56]

The sensing field has side length of $L = 2000$ m. Time is equally slotted (1 hour) and the average energy consumption rate of working sensor is 12 J/min. A typical sensing range r_s is 15 m. Wireless-rechargeable sensors have Li-Ion battery of 1200 mAh capacity and 3.7 V working voltage with $\Delta t = 30$ mins charging time from empty to full [33]. Solar/wind-powered sensors have batteries of 2, 150 mAh and 3.7 V working voltage. The average manufacturing costs of solar-powered, wind-powered and wireless-rechargeable sensors are quoted from Amazon as \$50, \$35 and \$30 apiece, respectively. The maximum energy harvesting power for solar-powered sensor and wind-powered sensor is 2 W·h and 1.5 W·h, respectively. The maximum energy harvesting power for HS are 2kW [45]. Note that, although available commercial HS [46] can generate as much as 3.3 kW with a size



(a) RGB map for the best composition of three kinds of sensors.



(b) Comparison of lifetime for different sensor compositions vs. field sizes.

Fig. 11. Optimal sensor composition and network lifetime.

of 4.9×2.7 m^2 solely depending on solar energy, HS may not assure enough energy supply for all weather conditions.

We assume re-locating HS consumes energy at 11.5 kW·h/100 km, which is on the same level of electrical vehicles [59]. The MC moves at a speed of 100 m/min at an energy consumption rate of 5 J/m. When the percentage of remaining energy for wireless-rechargeable sensor is lower than a threshold of 20%, they send out requests for recharge. The simulation time is 360 days.

7.1 Sensor Compositions and Lifetime

First, we evaluate the optimal composition of three types of sensors (solar, wind and wireless-rechargeable) for the minimum total cost and evaluate the network lifetime compared with network of one energy source. To obtain the energy consumption of different tasks at different time

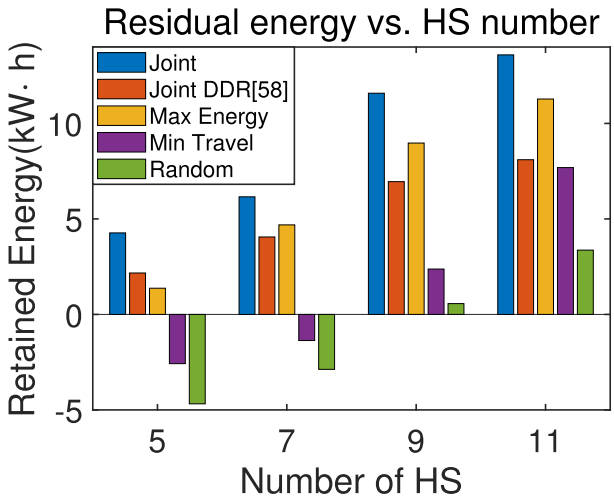
slots in the network, we develop a (high-level) simulator of the wireless sensor network using MATLAB.[3] Energy consumption of sensors is dominated by sensing/computing/communications. Here, we aim to monitor static targets. For example, once a target appears in the field, all the sensors within the sensing range of the target are turned on, with different energy consumption rates. The energy consumption of the task of monitoring this target is derived by the duration of appearance multiplying the sensing power and the number of sensors around it. Similarly, we can derive the energy consumption of other tasks like computing and communication. For the same type of tasks, the average energy consumption is used as the input of the formulated sensor composition problem together with the timestamps. The appearance of the targets follow poisson distribution. We rely on the off-the-shelf Arduino IoT series as the sensing platform: for sensing, the working voltage and current is 5 V and 15 mA (e.g., ultrasonic sensor), consuming power at 75 mW/h [62]; for computation, it features a low-power ARM Cortex-M0 processor with an average power consumption at 11.2 μW/MHz [63], and the maximum frequency of the processor is 50 MHz [64], so the estimated energy consumption of computing is 2016 mW/h. Due to frequency scaling, the actual power consumption could be lower. The communication module consumes around 12 mA at the max voltage of 3.6 V during transmission at 0 dBm [65], which is 43 mW/h.

Figure 11(a) demonstrates the number and percentages of three kinds of sensors for each grid by applying RGB heatmap. The color of each grid is determined by an $[R\ G\ B]$ vector, where R, G, B represent the solar-powered, wind-powered, and wireless-rechargeable sensors, respectively. The values are proportional to the number of corresponding sensors in each grid, which is normalized by the total number of sensors of their kinds. In other words, the color of each grid is the total number of sensors deployed and the ratios of each kind. For example, when [1 0 0] is red, only solar-powered sensors are deployed in the grid; when [0 1 1] is cyan, the same number of wind-powered sensors and wireless-rechargeable sensors are deployed with no solar-powered sensor.

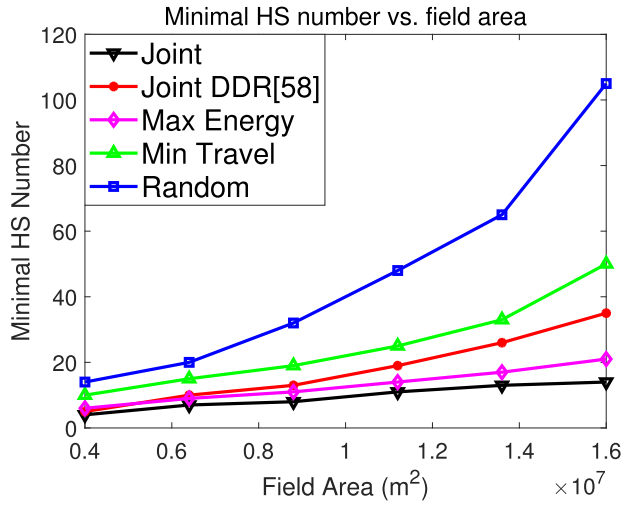
From Figure 11(a), it is observed that for the region $[100 \sim 400; 0 \sim 200]$ (i.e., the area in 100 m $\leq X \leq$ 400 m and 0 m $\leq Y \leq$ 200 m), similar number of solar sensors and wireless sensors and almost no wind sensors are deployed, since those areas have solar energy, whereas lack wind energy; for the region $[100 \sim 400; 600 \sim 800]$, most of the sensors are solar-powered, since those areas have abundant solar energy and no wind energy; for the region $[600 \sim 900; 700 \sim 900]$, wind sensors and wireless sensors have similar number while not many solar sensor, since those areas are abundant of wind energy and lacking of solar energy; for the region $[600 \sim 900; 900 \sim 1000]$, there are many sensors of all three kinds, since those areas are lacking of both solar and wind energy; for the region $[0 \sim 300; 900 \sim 1000]$, the numbers of different sensors are similar, since those areas are sufficient to provide all kinds of energy. The simulation demonstrates that our framework precisely selects different ratios of sensors reflecting the ambient energy distribution while minimizing the total manufacturing cost.

Figure 11(b) compares network lifetime, which is defined as the time expansion until the first energy depletion occurs; otherwise, the lifetime lasts the entire simulation time (360 days). Note that some energy depleted nodes would temporarily turn into sleep mode and wait for energy refill from the renewable energy or wireless charging. Other nodes with energy can still execute the network tasks and maintain operation. We compare our framework with some previous works [9, 33, 51, 52]. "All kinds" means all three energy sources are used referring to the mechanism proposed in this article. "Solar+Wireless" [33] means only solar and wireless-rechargeable sensors are used. "Wind+Wireless" means only wind and wireless-rechargeable sensors are used. "Wireless" [9],

[3]Since we generally focus on the performance metrics over a long run, such simulator can take experimental traces, datasets and hardware readings and is sufficient for our evaluations here.



(a) Residual energy at each HS for different deployment methods.

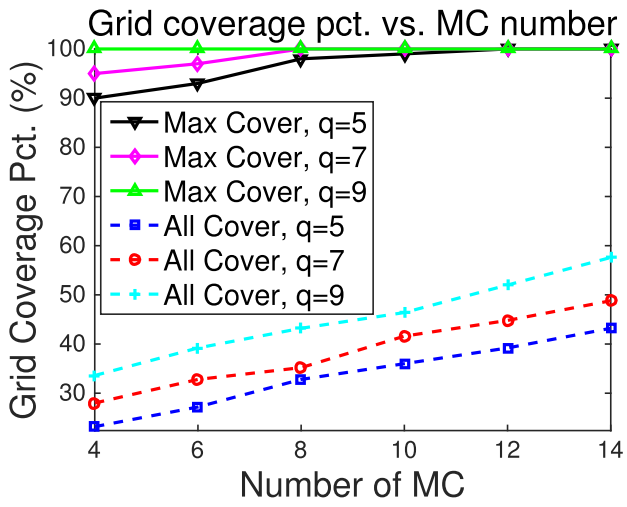


(b) Minimal number of HS to achieve energy balance.

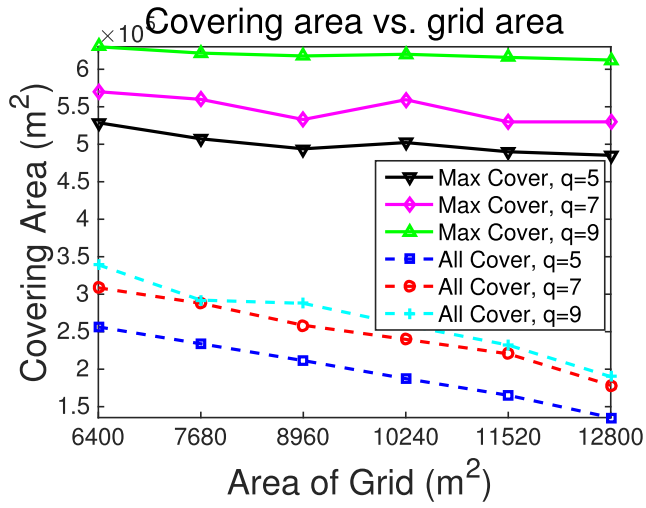
Fig. 12. Residual energy and minimal number of HS.

"Solar" [51], and "Wind" [52] represent the previous works that solely depend on one energy source. We alternate through all the combinations and their optimal compositions are computed by the MPM algorithm [11].

The results indicate having all kinds of sensors is able to maintain network lifetime over 360 days (for most field sizes). It still supports 273 running days for a giant area of 20.8 km^2 (5 times of the original size). Two types of sensors have shorter lifetime but still much longer than the single type. The difference among "Solar+Wireless," "Wind+Wireless," and "Wireless" becomes smaller when the area increases. This is because, relying on a single energy source is unstable when it suddenly becomes unavailable, that is more likely for larger field sizes. The lifetime of single energy harvesting sensor is the worst, which lasts for only about 20 days even for small area of 1.1 km^2. Our framework achieves at least 3 times longer lifetime compared with traditional



(a) Grid coverage percentage for different covering algorithms.



(b) Covering area of one MC for different covering algorithms.

Fig. 13. Grid coverage percentage and covering area by MC.

network of a single energy source. Meanwhile, the network depending on solar energy usually enjoys longer lifetime, since energy density of solar radiation is larger and the radiation is more consistent than wind. It is also observed that combining more than 2 types of energy harvesting sensors with the wireless ones is more beneficial for extending network lifetime.

7.2 Energy Output and Traveling Cost of MC

In this subsection, we evaluate network energy status using different methods to deploy HS and the number of HS needed to satisfy the energy demands of MC. Figure 12(a) demonstrates the total residual energy of q HS for different deploying strategies. The residual energy is the average energy remaining of the q HS over 360 days. It is difference of the harvested energy and the energy dissipated to replenish the MCs.

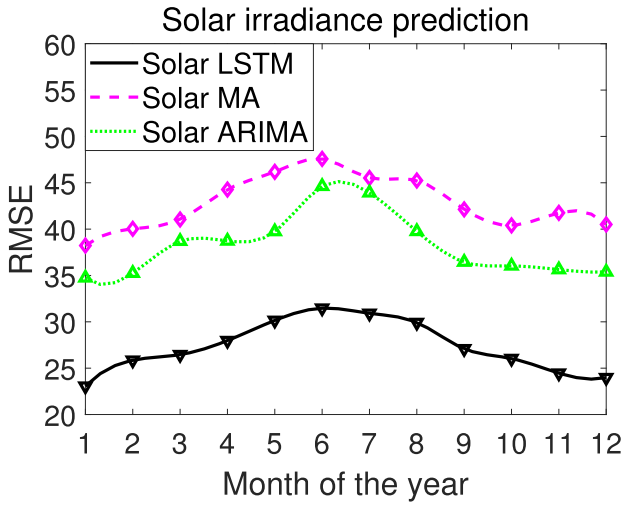
Figure 12 compares a few possible strategies of deploying HS. "Joint" is proposed in Section 4, which is compared with "Joint DDR," applying another field division algorithm named DDR proposed in [60]. The DDR mechanism divides the field into a series of equal-size quadrilaterals. Note that, "Joint" and "Joint DDR" both jointly consider the energy output and the traveling cost of MC, while the only difference is the field division. "Max Energy" deploys HS at the position with the maximum energy. "Min Travel" deploys HS at the centroid of each region. "Random" deploys q HS randomly in the field following a uniform distribution. We can see that residual energy grows with an increasing number of HS for all cases. For all four q values, "Joint" always has the maximum energy storage while "Random" is the worst. For $q = 5, 7$, the residual energy of "Min Travel" and "Random" is less than 0. When q is small, it is more important to wisely pick the locations of HS, since one false deployed HS is less likely to be compensated by other HS. "Max Energy" is always better than "Min Travel," since the harvested energy is much larger. The difference between the strategies becomes less obvious when q increases, because reduced region size weakened the impact from location choices. "Joint DDR" performs relatively better for small q values, since the influence of field division is less obvious with small q's. By balancing the charging demands via carefully dividing the field into as many as similar-size square shapes, the proposed field division algorithm increases the retained energy of HS by at least 51%.

Figure 12(b) shows the relation between the minimal number of HS needed to maintain energy balance. For a field of certain area, the minimal number of HS needed to maintain energy balance is evaluated for different deploying strategies. Maintaining energy balance needs to ensure the continuous operation. It shows that the minimal number of HS for "Joint" increases almost linearly with the increase of field size. The number for "Max Energy" is larger than "Joint" but still linear. "Min Travel" and "Random" increase much faster. "Random" grows the fastest. "Joint DDR" approaches "Joint" for small field area and increases rapidly as the field area expands, requiring 145% more HS at the end, which validates again that the proposed field division algorithm performs significantly better for large area. This is because random deployment tends to leave coverage holes for some regions and cause network disruption.

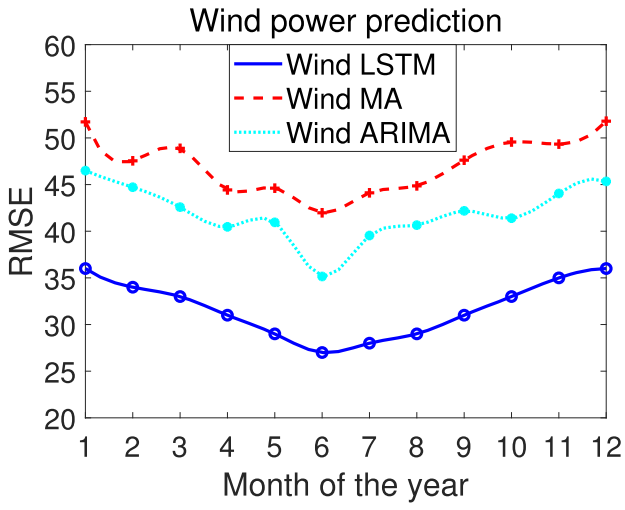
7.3 Network Coverage and Charging Capability

Finally, we evaluate network coverage and charging capability of an MC by comparing with the previous work that charges all sensors in Reference [47]. Figure 13(a) demonstrates coverage percentage with the number of MCs in terms of grids. The coverage percentage is the ratio of grids covered by MC to the total number of grids. "Max Cover" applies the maximum coverage charging algorithm. It jointly schedules the activities of charging sensors and replenishing the energy of MC for the maximum coverage rate of different grids. "All Cover" is the previous approach in Reference [47], which fulfills all charging requests in one grid first before moving to the next one. It is observed that our algorithm exhibits significant improvement for the coverage percentage of grids. "Max Cover" achieves 100% coverage ratio while "All Cover" achieves at most 58%. The coverage ratio increases for our algorithm when the number of HS increases, since larger q means less traveling cost for MC to replenish its own energy and also smaller size of grids. "Max Cover" does not achieve 100% sometimes with a few MCs, because of large energy request number vs. charging capability.

Figure 13(b) shows the covering capability of one MC for different charging algorithms. Covering capability is represented by the largest area one MC can serve (timely response to all the requests). As shown in Figure 13(b), the proposed "Max Cover" algorithm yields over 2 times charging capability for one MC than the "All Cover" algorithm, i.e., the charging capability is doubled. The charging area is not affected much by the increase of the grid size, whereas it is not the case for "All Cover." This is because larger grid means more sensors to be charged in one grid if "All



(a) Solar irradiance.



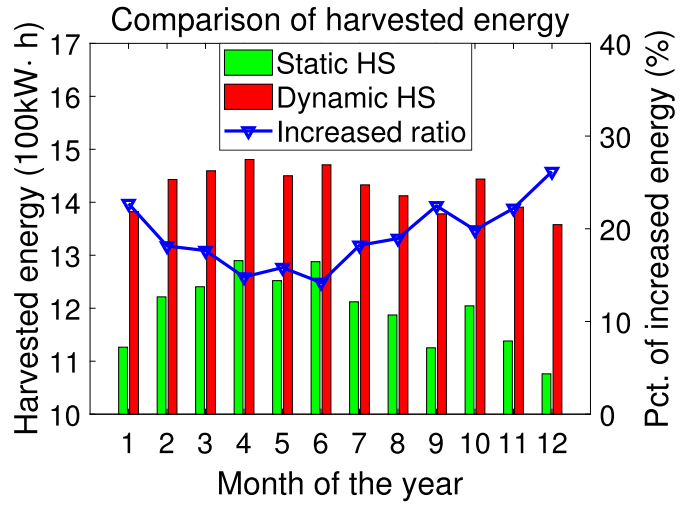
(b) Wind power.

Fig. 14. Prediction performance for energy output.

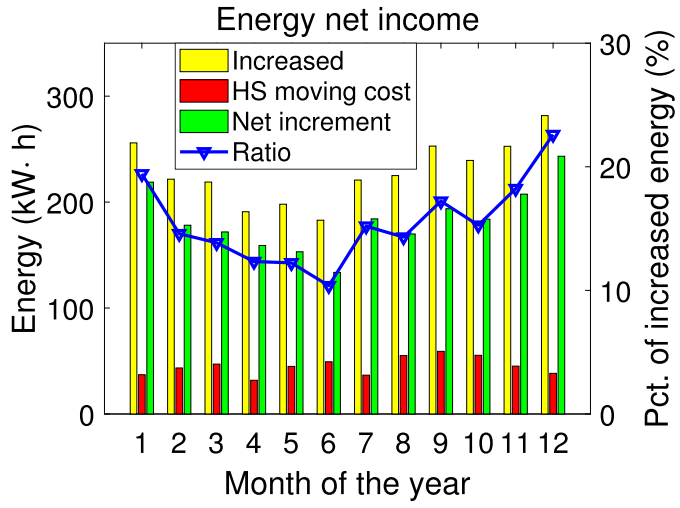
Cover" is applied. This would then postpone charging in other grids. When q increases, both algorithms have larger covering area, because more HS means shorter traveling distance for the MC, therefore it can spend more time in charge as well as enlarge its covering area.

7.4 Energy Prediction

We compare three energy prediction methods for the solar and wind harvesting income within a year in Figure 14. LSTM is a popular recurrent neural network [58] that is widely used for sequence learning in time series prediction, language models, and so on. It is compared with another two popular methods of MA and ARIMA via the metrics of **Root Mean Square Error (RMSE)**. The LSTM has two hidden layers and 6 backward time steps, MA has window size of 1 and ARIMA has lag order 6. Both figures in Figure 14 show that the LSTM is the best with smallest average RMSE



(a) Increased energy.



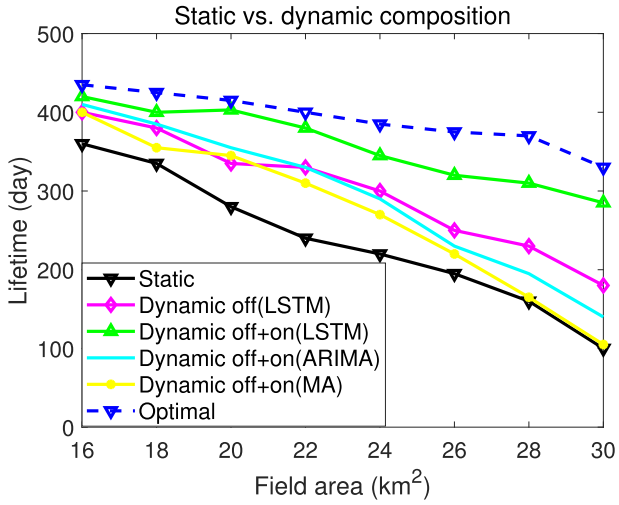
(b) Net income of energy.

Fig. 15. The increased energy harvested through dynamic HS position

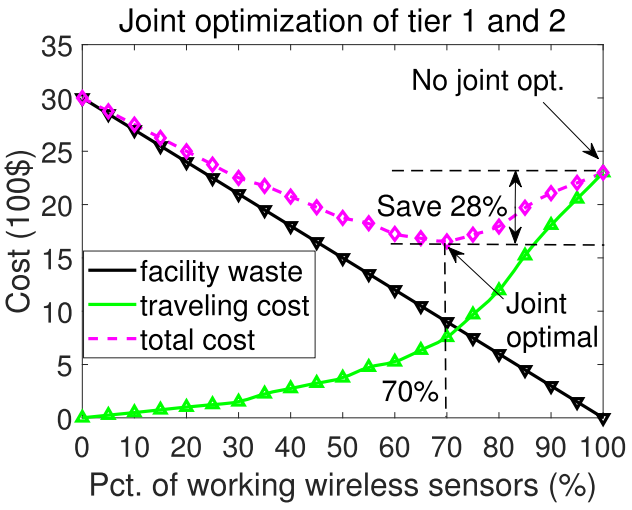
of 27.3. The maximum RMSE is observed during summer and winter for solar and wind energy respectively, which is considered to be caused by fiercer fluctuations of solar irradiance and wind speed in these seasons.

7.5 Dynamic HS Placement

We evaluate the additional energy harvested by dynamically adjusting the HS compared with the static deployment [54] for an extension of 500 days. Figure 15(a) shows that the harvested energy increases by an average of 19%. Note that, the increased percentage is inversely proportional to the amount of energy harvested. For example, in the April and June, the total harvested energy is the most but the corresponding percentages are the lowest, since the optimal placement of HS has lower impact when the ambient energy is abundant. Since the moving of HS also costs energy,



(a) Lifetime for different schemes.



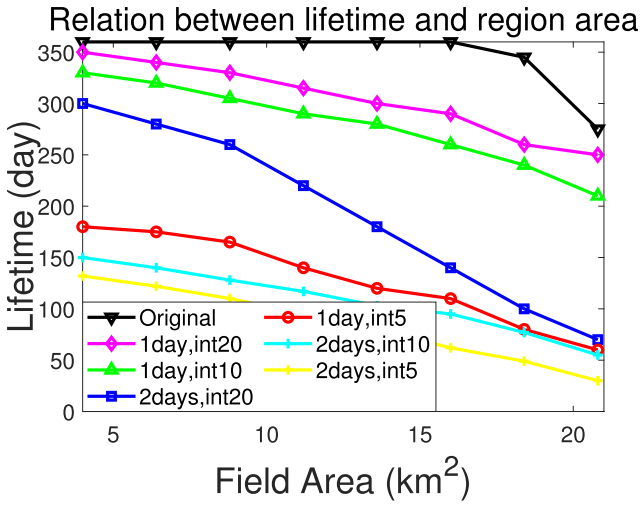
(b) Joint optimization.

Fig. 16. Comparison of static and dynamic sensor composition adjustment.

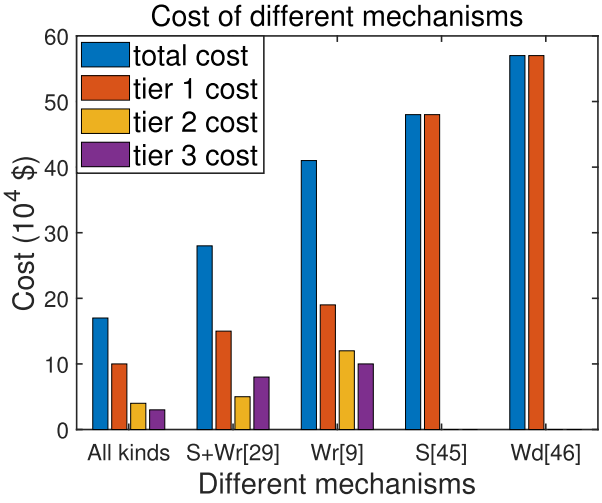
Figure 15(b) compares the increased energy via dynamic positioning with the cost of moving HS and the net income of energy. After deducted by the moving cost, the average percentage of energy increment is 15%, and achieves the maximum of 22.6% increase in December and the minimum of 10.4% in June. We can see that dynamic placement of HS fitting to the energy distribution is beneficial to the energy output across the year.

7.6 Dynamic Sensor Composition and Joint Optimization

Figure 16 evaluates the performance of dynamic sensor composition and joint optimization. The optimal sensor composition is derived based on the energy income and consumptions. Since we adopt average consumption rates, the optimality then depends on how well the prediction engine performs. It is demonstrated that the state-of-the-art methods like LSTM achieve minimum testing



(a) Lifetime of the networks for synthetic data testing extreme weather conditions.



(b) Comparison of costs for different mechanisms considering all three tiers of optimization.

Fig. 17. Lifetime of the network and the comparison of costs.

error on the simple time serial data. Thus, sensor composition can be considered as a near-optimal solution here. The resilience of the stochastic adjustment algorithm to the prediction errors is evaluated by applying ARIMA and MA as the prediction engines. In accordance with Figure 14, higher prediction errors result in shorter lifetime of the network. Meanwhile, the performance tends to be impeded more significantly for larger area, since large-scale network is less resilient to the prediction errors. Compared with the optimal (i.e., no prediction error), the algorithm based on LSTM loses 7.9% lifetime while ARIMA and MA loses lifetime by 25.5% and 30.8%, respectively.

The results demonstrate that making dynamic adjustment in the network offers further benefits to extend network lifetime. Figure 16(b) shows the results of joint optimization, where the cost due to a waste of facility[4] and the traveling are unified in $ and treated equally with the same

[4]Since the sensors are already deployed, the removals of them from the network cause the waste of the established facilities.

weights. The average optimal percentage of working wireless-rechargeable sensors is 70%, saving 28% cost compared with the case of static sensor composition. Note that, the cost increases as the percentage drops further, since the waste of already established facility dominates in this scenario.

7.7 Network Robustness under Extreme Weather Conditions and the Cost of Different Mechanisms

In this subsection, we further test the performance of networks under extreme weather conditions as shown in Figure 17(a), where the harvestable energy is absent continuously and periodically. Moreover, in Figure 17(b), a more comprehensive evaluation of the cost related with all three tiers of the optimization is conducted, which compares the proposed framework in this article with four other benchmarks.

As shown in Figure 17(a), the lifetime of the network is tested for the original realistic data [55, 56] and the synthetic data. The extreme weather conditions such as continuously rainy days are simulated via the synthetic data, modified based on the realistic data. In the legend of the figure, "1 day, int 20" means the power of the harvestable energy is set to 0 kW·h for 1 whole day for every interval of 20 days. The other legends are defined in the same way. This figure verifies that, the continuous absence of ambient energy does affect the lifetime of the network, while the lifetime reduces further for higher frequency of the absence. The frequencies of the energy absence are 0, 0.05, 0.1, 0.1, 0.2, 0.2, and 0.4, which is derived by taking the ratio of the absent days and the period. Note that, if the energy is absent for 2 days, the lifetime reduces as much as 70%. However, for "1 day, int 20" case, the reduction is only 10%.

Figure 17(b) compares the tier 1 manufacturing cost, tier 2 traveling cost, and tier 3 MC cost. Note that, tier 3 aims to maximize the area being covered by MC, whose performance is evaluated by the number of MC needed for satisfying all the energy requests. The unit cost for one MC is set to 1000 $ [31, 32], considering the cost of building the automatic battery swap system, the Field Programmable Gate Array scheduling the mobile computing tasks, and the wireless charging system. The proposed mechanism in this article is compared with benchmarks of other mechanisms as mentioned in Section 7.1. The shown cost represents the minimal required cost for the corresponding network running for at least one month. "All kinds" achieves the minimal total cost among the five mechanisms, and combining different sources together induces lower cost compared with sole energy source. The proposed mechanism can save at least 40% total cost. The mechanism solely depending on wireless charging has lower cost compared with the mechanisms solely depending on solar or wind energy, since the wireless charging is more reliable and consistent. This simulation indicates the potential of further reducing the cost by integrating more harvestable energy sources.

8 RELATED WORK

8.1 Energy Conservation

There is a plethora of literatures on energy conservation that improves energy efficiency using duty cycles, opportunistic data transmission and energy-efficient data collection. Here, we discuss some representative works [1, 2, 13–16]. In Reference [13], a new MAC protocol is proposed to reduce the energy consumption while balancing the energy conservation and contention latency. In Reference [14], a tracking system is implemented that can adaptively adjust the sensitivity to balance surveillance performance and energy-awareness. Energy conservation is explored in Reference [15] to jointly consider coverage and connectivity. In Reference [1], an opportunistic data transmission strategy is proposed to reduce the latency and redundancy for the low-duty-cycle sensors. In Reference [2], an on-demand prediction mechanism is proposed to forecast the wake-up time of sensors, so as to reduce the energy consumption while idling.

Data gathering also plays a key role in network-wide energy consumption. In Reference [16], both centralized and distributed data collection schemes are studied to achieve energy-efficient, delay-aware and lifetime-balancing. Yet, these works only focus on extending the battery lifetime rather than providing continuous operation.

8.2 Energy Provisioning

Wireless Charging. As a promising way to transmit electromagnetic energy to sensors via the air without interconnecting cords, wireless charging technologies can be classified into inductive coupling, magnetic resonance coupling, and RF radiation [17–19]. Inductive coupling [17] delivers power with the shortest range (millimeters to centimeters), but is the least hazardous to human health. Magnetic resonance coupling [18] has more complicated structure, but can charge multiple devices simultaneously without requiring the line-of-sight in a medium range (centimeters to meters). RF radiation [19] can charge mobile devices in a long range (meters to kilometers), but it has low charging efficiency and requires line-of-sight.

For the static sensors deployed in complex environment (i.e., lacking of line-of-sight conditions), their batteries can be efficiently replenished with inductive or magnetic coupling charging, by employing an MC with wireless-charging coils and high-capacity battery [3, 9, 20–25]. In Reference [3], sensors are partially charged such that more sensors can be charged within unit time, thus jointly increases the sensor lifetime and reduces the traveling cost of the MC. In Reference [9], the energy provisioning issues of WSN are studied to keep track of mobile targets with high precision. In Reference [20], a real-time energy status protocol is developed to schedule wireless charging based on residual lifetime and moving distance. In Reference [21], the optimal charging bundle of MC is generated to balance the charging efficiency and the trajectory distance. In Reference [22], the MC selectively charges sensors to extend the lifetime of the network and coverage of the MC while assuring the target k-coverage. In Reference [23], the energy provisioning problem is addressed by studying the deployment of static wireless-rechargeable tags and regulation of the mobile tags. In Reference [24], authors choose the charging sensors, charing time, and activation time to maximize the quality of monitoring for stochastic event based on known probability of arriving distribution. From the safety perspective, energy successfully transferred through the air is balanced with electromagnetic radiation to avoid health hazard [25].

As discussed in the survey [26], the dispatch strategies of MC can be classified as offline/online planing; single/multiple planning; centralized/distributed planning. In Reference [27], the authors derive the necessary and sufficient conditions for a single MC to maintain the infinite network lifetime considering the periodic patterns of residual energy levels of sensors. In Reference [28], for multiple MCs, authors proposes a polynomial algorithm to determine the scheduling and the minimal number of required MCs for both homogeneous and heterogenous charging frequencies. Different from the previous two article addressing offline planning, in Reference [29], an algorithm always serving the request with the smallest sum of the traveling time and charging time is proposed to serve the arriving requests in an online manner. To reduce the energy consumption of acquiring global knowledge and increase the scalability, in Reference [30], the MC finds the path following the fastest descent rate of the charger's battery levels, which is performed distributedly. These works consider wireless energy as the sole source to the WSNs, which does not benefit from harvestable energy and lacks self-sustainability.

Hybrid Energy. Upon realizing the limitations of wireless energy (potential hazards/close range), the work in Reference [33] takes a step back to consider a hybrid combination with ambient energy [23, 34, 35]. In particular, the network consists of wireless-rechargeable sensors with cluster heads powered by solar [33], which aims to mitigate the dynamics in ambient energy source with controllable wireless energy, as well as improve the power density on some heavily loaded sensors.

In additional to solar energy, other ambient sources can be exploited and possibly complement each other [34, 35]. In Reference [34], a multi-source power unit is designed to charge sensors with maximum efficiency and can be easily adapted to any combination of harvestable energy. In Reference [35], a communication system is designed to achieve the connection between RF-powered sensors and the Internet. However, it lacks a general framework that considers multiple energy sources (ambient/wireless charging). In contrast, this work can be extended to account for all known methods for energy provisioning.

Energy Management. Some previous works [36–39] have considered the energy management of energy harvesting sensor networks in a dynamic manner. In Reference [36], authors considered a more sophisticated characterization for the energy of harvested sensor network and the varying harvesting opportunity among different nodes to ameliorate the energy management problem. In Reference [37], the authors studied the energy management of wearable IoT devices depending on harvestable light and body heat energy to achieve the anticipated energy output. In Reference [38], an adaptive duty-cycling mechanism is proposed to perform the energy neutral operation for sensors and provide sufficient energy to them in face of the dynamically temporal nature of harvestable energy. In Reference [39], an adaptive power conservation protocol is proposed to realize efficient data propagation for heterogenous sensor networks accepting redeployment of sensors. However, these works cannot address the energy management of spatially distributed heterogenous energy and generate the maximal energy output with the minimal manufacturing cost.

9 CONCLUSIONS

In this article, we propose a new self-sustained WSN via integrating multi-source energy harvesting with wireless charging. First, we derive the optimal composition of different types of sensors by solving a maximum flow problem to minimize the budget. Second, we jointly consider both energy distribution and moving cost to deploy HS in the field of any arbitrary shape. Third, we propose a scheduling algorithm for the MC to coordinate wireless charging and its own energy replenishment at HS, prove its NP-hardness and propose a 4-approximation algorithm. We propose a stochastic algorithm to dynamically place HS, and jointly optimize different objectives. Finally, we demonstrate that the framework can significantly extend network lifetime, adapt to dynamic energy distribution and improve energy efficiency through extensive simulations.

REFERENCES

[1] S. Guo, L. He, Y. Gu, B. Jiang, and T. He. 2014. Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links. *IEEE Trans. Comput.* 63, 11 (Jul. 2013), 2787–2802.

[2] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson. 2011. PW-MAC: An energy-efficient predictive-wakeup MAC protocol for wireless sensor networks. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'11).* 1305–1313.

[3] W. Xu, W. Liang, X. Jia, Z. Xu, Z. Li, and Y. Liu. 2018. Maximizing sensor lifetime with the minimal service cost of a mobile charger in wireless sensor networks. *IEEE Trans. Mobile Comput.* 17, 11 (Mar. 2018), 2564–2577.

[4] California Fire Statistics. Retrieved from http://cdfdata.fire.ca.gov/incidents/incidents_stats.

[5] S. Takahashi, 2014. *Radiation Monitoring and Dose Estimation of the Fukushima Nuclear Accident.* Springer.

[6] J. Bernal-Agustín and R. Dufo-López. 2009. Simulation and optimization of stand-alone hybrid renewable energy systems. *Renew. Sust. Energy Rev.* 13, 8 (Oct. 2009), 2111–2118.

[7] A. Fahrenbruch and B. R. Bube, 2012. *Fundamentals of Solar Cells: Photovoltaic Solar Energy Conversion.* Elsevier.

[8] G. Cicia, C. Luigi, D. Teresa, and A. Palladino. 2012. Fossil energy versus nuclear, wind, solar and agricultural biomass: Insights from an Italian national survey. *Energy Policy* 42 (Mar. 2012), 59–66.

[9] P. Zhou, C. Wang, and Y. Yang. 2018. Static and mobile target k-coverage in wireless rechargeable sensor networks. *IEEE Trans. Mobile Comput.* 18, 10 (Sep. 2018), 2430–2445.

[10] F. Akhtar and M. Rehmani. 2015. Energy replenishment using renewable and traditional energy resources for sustainable wireless sensor networks: A. review. *Renew. Sust. Energy Rev.* 45 (May 2015), 769–784.

[11] V. Malhotra, M. Kumar, and S. Maheshwari. 1978. An $O(|V|^3)$ algorithm for finding maximum flows in networks. *Inf. Process. Lett.* 7, 6 (Oct. 1978), 277–278.
[12] F. Spieksma, 1999. On the approximability of an interval scheduling problem. *J. Schedul.* 2, 5 (Sep. 1999), 215–227.
[13] W. Ye, J. Heidemann, and D. Estrin. 2002. An energy-efficient MAC protocol for wireless sensor networks. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'02)*. 1567–1576.
[14] T. He, S. Krishnamurthy, J. A. Stankovic, T. F. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. Krogh. 2004. Energy-Efficient Surveillance System Using Wireless Sensor Networks. In *Proceedings of the ACM International Conference on Mobile Systems, Applications, and Services (MobiSys'04)*. 270–283.
[15] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill. 2003. Integrated coverage and connectivity configuration in wireless sensor networks. In *Proceedings of the ACM Conference on Embedded Networked Sensor Systems (Sensys'03)*. 28–39.
[16] Y. Yao, Q. Cao, and A. Vasilakos. 2015. EDAL: An energy-efficient, delay-aware, and lifetime-balancing data collection protocol for heterogeneous wireless sensor networks *IEEE/ACM Trans. Netw.* 23, 3 (Mar. 2014), 810–823.
[17] X. Wei, Z. Wang, and H. Dai. 2014. A critical review of wireless power transfer via strongly coupled magnetic resonances. *Energies* 7, 7 (Jul. 2014), 4316–4341.
[18] X. Li, C. Tsui, and W. Ki. 2015. A 13.56 MHz wireless power transfer system with reconfigurable resonant regulating rectifier and wireless power control for implantable medical devices. *IEEE J. Solid State Circ.* 50, 4 (Feb. 2015), 978–989.
[19] X. Lu, P. Wang, D. Niyato, and E. Hossain. 2014. Dynamic spectrum access in cognitive radio networks with RF energy harvesting. *IEEE Wireless Commun.* 21, 3 (Jun. 2014), 102–110.
[20] C. Wang, J. Li, F. Ye, and Y. Yang. 2014. Mobile data gathering with load balanced clustering and dual data uploading in wireless sensor networks *IEEE Trans. Mobile Comput.* 14, 4 (Jul. 2014), 770–785.
[21] N. Wang, J. Wu, and H. Dai. 2019. Bundle charging: Wireless charging energy minimization in dense wireless sensor networks. In *Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS'19)*. 810–820.
[22] P. Zhou, C. Wang, and Y. Yang. 2017. Leveraging target k-coverage in wireless rechargeable sensor networks. In *Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS'17)*. 1291–1300.
[23] S. He, J. Chen, F. Jiang, D. Yau, G. Xing, and Y. Sun. 2012. Energy provisioning in wireless rechargeable sensor networks. *IEEE Trans. Mobile Comput.* 12, 10 (Sep. 2012), 1931-1942.
[24] H. Dai, Q. Ma, X. Wu, G. Chen, D. K. Yau, S. Tang, X. Li, and C. Tian. 2018. Chase: Charging and scheduling scheme for stochastic event capture in wireless rechargeable sensor networks. *IEEE Trans. Mobile Comput.* 19, 1 (Dec. 2018), 44–59.
[25] S. Nikoletseas, T. P. Raptis, and C. Raptopoulos. 2015. Low radiation efficient wireless energy transfer in wireless distributed systems. In *Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS'15)*. 196–204.
[26] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han. 2015. Wireless charging technologies: Fundamentals, standards, and network applications. *IEEE Commun. Surv. Tutor.* 18, 2 (Nov. 2015), 1413–1452.
[27] L. Xie, Y. Shi, Y. Hou, and H. Sherali. 2012. Making sensor networks immortal: An energy-renewal approach with wireless power transfer. *IEEE/ACM Trans. Netw.* 20, 6 (Feb. 2012), 1748–1761.
[28] R. Beigel, J. Wu, and H. Zheng. 2014. On optimal scheduling of multiple mobile chargers in wireless sensor networks. In *Proceedings of the 1st International Workshop on Mobile Sensing, Computing and Communication*. 1–6.
[29] X. Ren, Xiaojiang, W. Liang, and W. Xu. 2014. Maximizing charging throughput in rechargeable sensor networks. *Proceedings of the International Conference on Computer Communications and Networks (ICCCN'14)*. 1–8.
[30] C. M. Angelopoulos, S. Nikoletseas, and T. P. Raptis. 2014. Wireless energy transfer in sensor networks with adaptive, limited knowledge protocols. *Comput. Netw.* 70 (Sep. 2014), 113–141.
[31] J. Li, C. Wang, and Y. Yang. 2017. A general purpose testbed for mobile data gathering in wireless sensor networks and a case study. In *Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS'17)*. 1694–1703.
[32] T. Liu, B. Wu, H. Wu, and J. Peng. 2016. Low-cost collaborative mobile charging for large-scale wireless sensor networks. *IEEE Trans. Mobile Comput.* 16, 8 (Oct. 2016), 2213–2227.
[33] C. Wang, J. Li, Y. Yang, and F. Ye. 2016. A hybrid framework combining solar energy harvesting and wireless charging for wireless sensor networks. *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'16)*. 1–9.
[34] M. Magno, S. Marinkovic, D. Brunelli, E. Popovici, B. O'Flynn, and L. Benini. 2012 , Smart power unit with ultra low power radio trigger capabilities for wireless sensor networks. In *Proceedings of the IEEE Design, Automation and Test in Europe Conference (DATE'12)*. 75–80.
[35] B. Kellogg, A. Parks, S. Gollakota, J. Smith, and D. Wetherall. 2014. Wi-Fi backscatter: Internet connectivity for RF-powered devices. In *Proceedings of the ACM Special Interest Group on Data Communication Conference (SIGCOMM'14)*. 607–618.

[36] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava. 2007. Power management in energy harvesting sensor networks. *ACM Trans. Embed. Comput. Syst.* 6, 4 (Sep. 2007), 32-es.
[37] G. Bhat, J. Park, and U. Y. Ogras. 2017. Near-optimal energy allocation for self-powered wearable systems. In *Proceedings of the IEEE 36th International Conference on Computer-Aided Design (ICCAD'17)*. 368–375.
[38] C. M. Vigorito, D. Ganesan, and A. G. Barto. 2007. Adaptive control of duty cycling in energy-harvesting wireless sensor networks. In *Proceedings of the 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*. 21–30.
[39] I. Chatzigiannakis, A. Kinalis, and S. Nikoletseas. 2005. An adaptive power conservation scheme for heterogeneous wireless sensor networks with node redeployment. In *Proceedings of the Symposium on Parallelism in Algorithms and Architectures (ACM SPAA'05)*. 96–105.
[40] D. Jager and A. Andreas. NREL Report No DA-5500-56489. NREL National Wind Technology Center (NWTC): M2 Tower, Boulder, Colorado (Data).
[41] T. Senjyu, R. Sakamoto, N. Urasaki, T. Funabashi, H. Fujita, and H. Sekine. 2006. Output power leveling of wind turbine generator for all operating regions by pitch angle control. *IEEE Trans. Energy Convers.* 21, 2 (Jun. 2006), 467–475.
[42] M. Hansen. 2015. *Aerodynamics of Wind Turbines*. Routledge.
[43] Power Consumption Benchmarks. Retrieved from http://www.pidramble.com/wiki/benchmarks/power-consumption.
[44] Y. Zhang, S. Chen, Y. Zhou, and Y. Fang. 2018. Using wireless tags to monitor bodily oscillation. In *Proceedings of the IEEE International Conference on Mobile Ad-Hoc and Smart Systems (MASS'18)*. 211–219.
[45] V. Khare, S. Nema, and P. Baredar. 2016. Solar-wind hybrid renewable energy system: A review. *Renew. Sust. Energy Rev.* 58 (May 2016), 23–33.
[46] EnvisionSolar. Retrieved from http://www.envisionsolar.com.
[47] C. Wang, S. Guo, and Y. Yang. 2016. An optimization framework for mobile data collection in energy-harvesting wireless sensor networks. *IEEE Trans. Mobile Comput.* 15, 12 (Feb. 2016), 2969–2986.
[48] H. Song. 2008. A method of mobile base station placement for High Altitude Platform based network with geographical clustering of mobile ground nodes. In *IEEE International Multiconference on Computer Science and Information Technology*. 869–876
[49] R. Van, M. Matthias, R. Niedermeier, and M. Weller. 2015. Interval scheduling and colorful independent sets. *J. Sched.* 18, 5 (Oct. 2015), 449–469.
[50] C. Papadimitriou, and K. Steiglitz. 1998. *Combinatorial Optimization: Algorithms and Complexity*. Dover.
[51] B. Gaudette, H. Vinay, V. Sarma, and K. Marwan. 2012. Optimal range assignment in solar powered active wireless sensor networks. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'12)*. 2354–2362.
[52] S. Kosunalp. 2017. An energy prediction algorithm for wind-powered wireless sensor networks with energy harvesting. *Energy* 139 (Nov. 2017), 1275–1280.
[53] N. Sharma, J. Gummeson, and D. Irwin. 2010. Cloudy computing: Leveraging weather forecasts in energy harvesting sensor systems. In *Proceedings of the IEEE International Conference on Sensing, Communication and Networking (SECON'10)*. 1–.
[54] P. Zhou, C. Wang, and Y. Yang. 2019. Self-sustainable sensor networks with multi-source energy harvesting and wireless charging. in *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'19)*. 1828–1836.
[55] Solargis. Retrieved from http://solargis.com.
[56] National Renewable Energy Laboratory. Retrieved from http://www.nrel.gov.
[57] C. Wang, J. Li, F. Ye, and Y. Yang. 2013. Multi-vehicle coordination for wireless energy replenishment in sensor networks. In *Proceedings of the IEEE International Parallel & Distributed Processing Symposium (IPDPS'13)*. 1101–1111.
[58] S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neur. Comput.* 9, 8 (Nov. 1997), 1735–1780.
[59] S. Hanley. 2018. Hyundai & Kia to begin offering solar sunroofs after 2019. Retrieved from https://cleantechnica.com/2018/11/06/hyundai-kia-to-begin-offering-solar-sun-roofs-after-/2019.
[60] K. Latif, N. Javaid, M. N. Saqib, Z. A. Khan, and N. Alrajeh. 2016. Energy consumption model for density controlled divide-and-rule scheme for energy efficient routing in wireless sensor networks. *Int. J. Ad Hoc Ubiq. Comput.* 21, 2 (2016), 130–139.
[61] R. Lee, K. Chen, S. Chiang, C. Lai, H. Liu, and M. Wei. 2006. A backup routing with wireless sensor network for bridge monitoring system. In *Proceedings of the 4th IEEE Annual Communication Networks and Services Research Conference (CNSR'06)*. 5–9.
[62] Arduino-Ultrasonic Sensor. Retrieved from https://www.tutorialspoint.com/arduino/arduino_ultrasonic_sensor.htm.
[63] ARM's Most Energy Efficient Cortex-M Class Processor. Retrieved from https://www.silabs.com/products/mcu/32-bit/arm-cortex-m0-32-bit-microcontroller.

[64] LPC1102/1104 32-bit ARM Cortex-M0 microcontroller product data sheet. Retrieved from https://www.nxp.com/docs/en/data-sheet/LPC1102_1104.pdf.
[65] How nRF24L01 Wireless Module Works and Interface with Arduino. Retrieved from https://lastminuteengineers.com/nrf24l01-arduino-wireless-communication/.

Received May 2020; revised November 2020; accepted March 2021