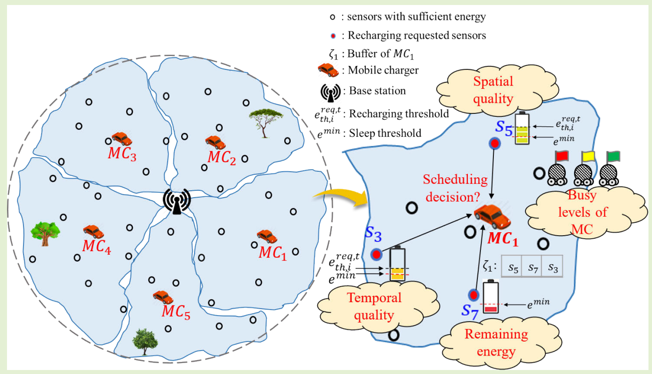
IEEE Sensors Council IEEE SENSORS JOURNAL, VOL. 23, NO. 11, 1 JUNE 2023

WLARS: Workload-Aware Recharge Scheduling Mechanism for Improving Surveillance Quality in Wireless Rechargeable Sensor Networks

Bhargavi Dande, Chih-Yung Chang, *Member, IEEE*, Shih-Jung Wu, and Diptendu Sinha Roy

***Abstract*—With radio frequency (RF), the mobile charger (MC) can wirelessly transmit energy to the sensor nodes in the network. The wireless energy transfer enhances the lifetime of sensor nodes in wireless rechargeable sensor networks (WRSNs). Most of the studies improved the efficiency of MC or perpetuated the lifetime of WRSNs. At the same time, none of the studies focused on considering the spatial and temporal surveillance qualities (STSQs) of each sensor recharged. Therefore, this article proposed an efficient energy recharge scheduling for MC, aiming to maximize the STSQ of the given network. Initially, sensor routing load (RL) is considered to partition the network to distribute the charging load of the MCs evenly. Second, the busy level of the MC is considered to send the recharging requests to the MC. Based on the busy level of MC, the sensors adjust the sensing rate frequency to manage their energy. Finally, cooperation between the neighboring MCs is proposed to reduce the waiting time for recharging requested sensors. The simulation results present that the proposed work yields the literature in terms of STSQ, traveling distance, average waiting time (AWT), and energy usage efficiency.**



***Index Terms*—Mobile charger (MC), recharging, routing load (RL), spatial and temporal qualities, wireless rechargeable sensor networks (WRSNs).**

I. Introduction

WIRELESS sensor networks (WSNs) are comprised of tiny sensors used in many applications, such as smart homes [1], industries [2], and transportation [3]. However, sensors bounded battery capacity constraints the development of WSNs. Therefore, recharging the sensors is one of the critical issues in WSNs. Many algorithms were proposed to recharge the sensors in the literature [4], [5], [6], [7]. In the literature, these algorithms were prorated into two categories, including offline and online recharging architectures. In the offline architecture [8], [9], [10], [11], [12], [13], the mobile charger (MC) moves to each sensor to recharge them. This type of architecture did not consider the dynamic recharging schedule for the old requests that have been made. It leads to an inefficient recharging schedule for the new requests since some sensors that sent the new requests might be very close to the recharging path of the old requests. If the recharging path can be dynamically reconstructed according to the new requests, some new requests can be satisfied using only a little overhead.

A few researchers proposed online recharging architectures [15], [16], [17], [18], [19], [20], [21], [22] to overcome the limitations of offline recharging architecture. These architectures consider the real-time energy consumption (EC) of sensors and dynamically change the path based on the sensor requests. The sensors in the online recharging architecture send a recharging request to the MC based on the threshold value, that is, the real-time EC of sensors is considered. In addition, assume that there is a sensor closer to the MC path, and it sends the recharging request. The MC includes the newly requested sensor in its path, reducing the MCs moving length and energy consumed. These online recharging architectures are suitable for high-demand applications.

Manuscript received 6 February 2023; revised 7 April 2023; accepted 10 April 2023. Date of publication 17 April 2023; date of current version 31 May 2023. This work was supported in part by the Ministry of Science and Technology of Taiwan under Grant MOST 111-2221-E-032-015 and Grant MOST 111-2221-E-032-014. The associate editor coordinating the review of this article and approving it for publication was Prof. Nitaigour P. Mahalik. *(Corresponding author: Chih-Yung Chang.)*

Bhargavi Dande is with the Department of Electrical Engineering, National Taiwan University, Taipei 10617, Taiwan (e-mail: bhargavidande@ntu.edu.tw).

Chih-Yung Chang and Shih-Jung Wu are with the Department of Computer Science and Information Engineering, Tamkang University, New Taipei 25137, Taiwan (e-mail: cychang@mail.tku.edu.tw; wushihjung@mail.tku.edu.tw).

Diptendu Sinha Roy is with the Department of Computer Science and Engineering, National Institute of Technology Meghalaya, Shillong 793003, India (e-mail: diptendu.sr@nitm.ac.in).

Digital Object Identifier 10.1109/JSEN.2023.3266483

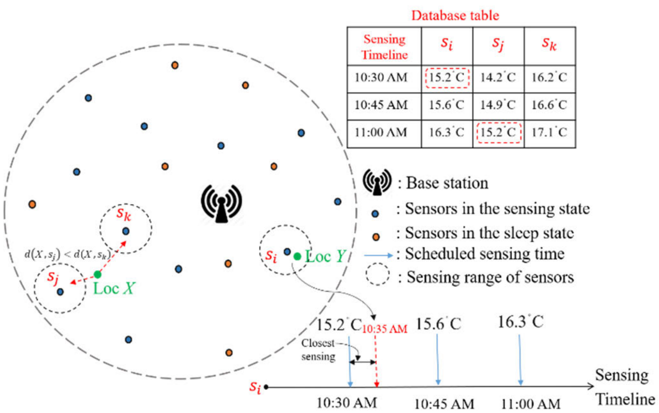


Fig. 1. Example to illustrate the STSQs representation.

Designing the threshold value in this type of architecture is very important. Our previous works [23], [24], [25], [26] considered the MCs recharging and moving times to determine the charging threshold value of sensors. As an extension of our previous study [26], this article designed the threshold value based on the queuing theory (QT).

A study [21] investigated the MC schedule based on the energy needed by each cluster in the network. In addition, this study also considered multicharger cooperation based on the distance between the clusters. Study [22] considered the mean waiting time and charging tours. This study adopted the fuzzy logic to schedule the MC. The cooperation between MCs is considered in this study based on remaining energy, sensor density, and critical node degree parameters to schedule the MC. However, these studies did not consider the network's spatial and temporal surveillance quality (STSQ). Different from the above-related works [21], [22], [23], [24], [25], and [26], this article considered the STSQ provided by each recharging requested sensor. Although the existing studies [27], [28], [29], and [30] considered the STSQ, the proposed workload-aware recharge scheduling (WLARS) algorithm measured STSQ based on the ant pheromone algorithm, which is different from the abovementioned studies.

A detailed description of STSQ calculation is presented in the following. The sensors in wireless rechargeable sensor networks (WRSNs) sense data according to the schedule and forward the sensed data to the base station (BS). For example, any user wants to calculate the "spatial surveillance quality (SSQ) in location X at 11:00 A.M." As shown in Fig. 1, there is no sensor deployed in location X, but the sensors s_j and s_k are closer to location X. Compared to the sensor s_k, we have $d(X, s_j) < d(X, s_k)$. Therefore, the BS will use the data collected by the closest sensor s_j at 11:00 A.M. for better SSQ accuracy.

Similarly, the temporal SQ is presented in the following. For instance, any user wants to know the "temporal SQ in location Y at time 10:35 A.M.?" As shown in Fig. 1, there is a sensor s_i deployed at location Y, but it is not sensing at 10:35 A.M. Therefore, the closest sensing time slot, i.e., 10:30 A.M. compared to 10:45 A.M., will represent location Y's temporal SQ for better temporal SQ, as shown in the database table. From the spatial and temporal quality calculation examples, it should be noted that SQ will be lower if the spatial and temporal distances are larger, which is like the concept of ant pheromone algorithm. This article proposed a WLARS mechanism for improving SQ in WRSNs, called WLARS. In this study, the network is partitioned based on the sensor's routing load (RL).

When the MC receives many recharging requests from sensors, it calculates the spatial surveillance contribution of each requested sensor. Then, it recharges the sensors with higher SSQ, aiming to improve the spatial quality. Then, the threshold value is determined based on the QT concept, and the sensor battery is constrained based on the MC busy level. In addition, this study also considered that MC could choose adaptive buffers such as offline and online recharging paths (i.e., hybrid mechanisms). Cooperation between neighboring MCs is also proposed, aiming to reduce the latency of the recharging sensors, which can improve both the STSQs. The contributions of the proposed WLARS are presented as follows.

A. Designing the Threshold Value Based on the QT Concept

In our previous work [26], the charging and moving times of the sensors on the path are considered to estimate the threshold value. However, it is not suitable for large-scale networks. Therefore, this study adopted the QT concept to determine the threshold value of sensors in large-scale networks.

B. Considering the Busy Levels of MC

This study considered the busy levels of MC to send the recharging request, that is to say, the sensor's battery is adjusted according to the busy level of MC by adjusting its sensing frequency. It will reduce the recharging requests received by the MC, reducing the network traffic and saving the EC for packet forwarding.

C. Considering the STSQs of the Network

Unlike the existing studies [21] and [22], this study considered the network's STSQ. In this study, the sensors with higher SSQ will be given higher priority for recharging, and sensors will adjust their sensing rate frequency to maximize the temporal SQ.

D. Considering the Adaptive Buffer and Cooperation (ABC) Between MCs

In this article, the hybrid (online and offline) mechanism, i.e., adaptive buffer for MC, is designed. The MC can choose its path according to the application type. The cooperation between the MCs is considered to reduce the charging latency of the recharging requested sensors.

The remainder of this article is organized as follows. Section II reviews the offline and online recharging architectures. The system model and problem statement are detailed in Section III. Sections IV and V present the proposed WLARS algorithm and the performance evaluations. Finally, Section VI presents the conclusion and future work of this study.

II. Related Works

In the literature, the recharging architectures of WRSNs were divided into two types: offline [8], [9], [10], [11], [12], [13], [14] and online [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26]. Some of the related works will be reviewed in the following.

A. Offline Recharging Architectures

In the offline recharging architecture, the MC constructs the predetermined charging schedule and recharges the sensors periodically, aiming to extend the lifetime of sensor nodes. Study [8] considered the MCs tour planning and depot positioning. In addition, this study aimed to reduce the number of MCs used. Study [9] proposed periodic charging and path planning of MC with time slot scheduling. This study aimed to reduce the unnecessary roaming of the MC based on the fine-grained node classification scheme. Similar to the study [9], periodic charging planning for MC considering the limited traveling energy is proposed in [10]. Another study [11] adopted a genetic algorithm to propose the charging scheme, where sensors were partially charged at each round. This study aimed to reduce the dead nodes and the charging time of the sensor.

A study [12] proposed a periodic recharging algorithm to maximize the recharging benefit. In addition, this study aimed at reducing the dead nodes and maximizing the energy efficiency of MC based on the distributed scheduling algorithm. Another study [13] proposed an intelligent charging scheme to select the appropriate charging node. This study constructed the path of MC and aimed to optimize the system utility. A study [14] proposed a recharging scheduling algorithm by considering the weight of the grid. This study prioritized the sensors with higher coverage contributions and recharged them to achieve higher SQ. Although the studies mentioned above [8], [9], [10], [11], [12], [13], [14] proposed periodic charging mechanisms, these architectures are unsuitable for on-demand applications.

B. Online Recharging Architectures

In these architectures, the MC considered the real-time requests from the sensors. The sensors can send the recharging request to the MC according to their current energy profiles. Then, the MC will dynamically change the recharging schedule and construct a new recharging path. A study [15] proposed a hybrid recharging mechanism by adapting the EC of the sensors in the online mode and optimizing the path of the MC in the offline mode. This study investigated reducing the charging duration of the sensors to minimize the charging latency and waiting time. Another study [16] proposed MC scheduling by reducing the path cost and the dead nodes while optimizing the charging time of each node. Study [17] proposed MC scheduling by prioritizing the sensor's contribution. In addition, this study also investigated on node deletion scheme to remove the nodes from the path, which did not enhance the charging performance. All the studies [15], [16], [17] designed the single MC scheduling.

TABLE I
Comparison of the Proposed and Related Works

Related works	No of MCs	Threshold value	Busy levels & adaptive buffer of MCs	The spatial and temporal surveillance qualities
[15]	*Single*	*Fixed*	×	×
[16]	*Single*	*Fixed*	×	×
[17]	*Single*	*Fixed*	×	×
[18]	*Multiple*	*Fixed*	×	×
[19]	*Multiple*	*Fixed*	×	×
[20]	*Multiple*	*Fixed*	×	×
[21]	*Multiple*	*Fixed*	×	×
[22]	*Multiple*	*Adaptive*	×	×
[23]	*Single*	*Adaptive*	×	×
[24]	*Single*	*Adaptive*	×	×
[25]	*Multiple*	*Adaptive*	×	×
[26]	*Single*	*Adaptive*	×	×
[27]	*Single*	*Fixed*	×	○
Proposed *WLARS*	*Multiple*	*Adaptive*	○	○

Few works [18], [19], [20], [21], [22], and [25] proposed multiple MC scheduling for large-scale WSNs. Study [18] proposed a multiple MC scheduling algorithm by considering the preemption in WRSNs to maximize the charging throughput. Besides, this study determined the threshold value of sensors by considering the speed of the MC, remaining energy, and queue length. The network is divided into clusters using the mean shift algorithm [19]. The clusters were prioritized by considering the number of nodes and their remaining energies. In addition, it assumed that the threshold value of all the sensors was fixed and predetermined. Similar to [19], study [20] proposed the charging schedule of MC with mean drift clustering by considering the density of sensor nodes in the network. Another study [21] investigated two threshold values: charging and waiting time thresholds. It proposed a multi-MCs cooperation algorithm for nonuniform network clustering. Besides, the number of MCs is determined according to the number of recharging requests received from the sensor nodes. Study [22] adopted fuzzy logic and proposed a charging schedule of MCs by considering the MCs sensor energy and density and location of MC. It considered the MCs remaining energy and average traveling time to determine the charging threshold value.

However, most studies [15], [16], [17], [18], [19], [20], [21] assumed that the charging threshold value is fixed. In the proposed WLARS algorithm, the sensor's STSQs are considered before MC makes the charging decision. In addition, this article determines the charging threshold values according to the QT concept. Furthermore, the busy levels and adaptive buffers of MCs are also considered in the proposed WLARS algorithm, while the studies [21] and [22] ignored them. Table I gives a comparison between the proposed and related works.

III. System Model and Problem Statement

This section presents the considered environment and formulates the problem statement of the proposed work.

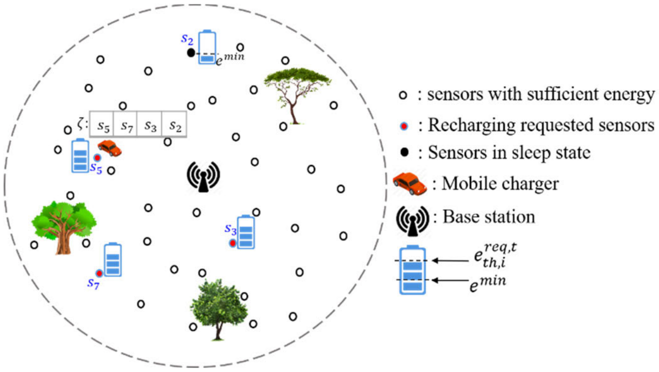


Fig. 2. Network environment of the proposed work.

A. Network Environment

Let $\mathcal{R}$ represent the circular monitoring region with radius r. There are n static sensors, $S = \{s_1, s_2, \ldots, s_n\}$, which have been randomly deployed in $\mathcal{R}$. Each sensor is equipped with a rechargeable battery and is responsible for sensing the environment and transmitting its readings to the BS based on the routing path. Each sensor consumes energy for performing sensing and communication tasks. Therefore, every sensor needs to be recharged before its energy is entirely exhausted. There is m (>1) homogeneous MC in the network, aiming to move to sensors and recharge them. However, the energy of MC is also limited. One stationary BS in the network collects the sensor data and recharges the MCs. Each MC should go to the BS before its energy exhaustion. All the MCs are initially located at the BS. When the BS receives the recharging request from the sensors, it will dispatch the MCs to provide the requested sensor with the charging service. To efficiently construct the path for MC to recharge the sensors, the BS might dispatch the MC after receiving a set of requests. If any sensor switches to a sleeping state due to energy exhaustion, the network will lead to coverage loss. This article aims to develop a workload-aware energy-recharging schedule for MCs to achieve maximal SQ and maintain a sustainable lifetime.

B. Recharging Model

Let $E_S^{\max}$ denote the maximum battery capacity of each sensor node. Due to uneven forwarding load, each sensor s_i in the network might have different energy discharging rates e_i^{disch}. In this article, two threshold values are designed for each sensor, namely, the recharging threshold $e_{\text{th},i}^{\text{req}}$ and sleep threshold $e^{\min}$. When the remaining energy of any sensor reaches $e_{\text{th},i}^{\text{req}}$, it will send out the recharging request. On the other hand, when the remaining energy of any sensor reaches $e^{\min}$, it will switch to a sleep state. Let ζ denote the buffer of the MC for maintaining the requests from all sensors. The network model of this work is shown in Fig. 2.

Let T denote the total time, which is further divided into h time slots $T = \{t_1, t_2, \ldots t_j, \ldots t_h\}$ where the length of each time slot t_i is denoted by t_{unit}. Let $d(l_i, l_M)$ denote the distance between the sensor s_i and MC. Let r^{crg} denote the charging radius of MC. Let $\phi_{i,j}^{\text{crg}}$ denote the Boolean variable to represent whether or not the MC is located in the charging range of s_i at time slot t_j. The value of $\phi_{i,j}^{\text{crg}}$ is determined according to the following:

$$\phi_{i,j}^{\text{crg}} = \begin{cases} 1 & d\left(l_i, l_M\right) \le r^{\text{crg}} \\ 0 & d\left(l_i, l_M\right) > r^{\text{crg}}. \end{cases} \tag{1}$$

This article uses the Friis transmission equation [31] as the wireless charging model. Let E_i^{crg} denote the total recharged energy of the sensor s_i. Let $P_{\text{crg}}^{\text{tx}}$ denote the transmission power of MC. Let $z_{\text{crg}}^{\text{tx}}$ and $z_{\text{crg}}^{\text{rx}}$ denote the transmitter and receiver antenna gains, respectively. If $\phi_{i,j}^{\text{crg}} = 1$, the value of E_i^{crg} is calculated according to the following:

$$E_i^{\text{crg}} = P_{\text{crg}}^{\text{tx}} \times \left(\frac{\lambda_{\text{crg}}}{4\pi\left(d\left(l_i, l_{\text{current}}\right)\right) + \beta_{\text{crg}}}\right)^2 \times \frac{z_{\text{crg}}^{\text{tx}} \times z_{\text{crg}}^{\text{rx}} \times \sigma_{\text{crg}}}{\psi_{\text{crg}}} \tag{2}$$

where λ_{crg} represents the wavelength of the radio frequency (RF) wave and σ_{crg} and ψ_{crg} denote the rectifier efficiency and the polarization loss, respectively. Once the sensor s_i is recharged, it has the energy to be consumed for sensing and transmission. Let $E_{S,j}$ denote the EC for performing the sensing task at the time slot t_j. The value of $E_{S,j}$ is given in the following:

$$E_{S,j} = e_s \lambda_{i,j} * \kappa \tag{3}$$

where e_s denotes the energy consumed by perceiving unit bit data (J/bit), $\lambda_{i,j}$ denotes the data perception rate of the sensor s_i at time slot t_j, and κ represents the number of bits in a packet.

Let sender s_i and receiver s_j be a communication pair. Assume that s_i sends κ bits to s_j and $d_{i,j}$ is the distance between s_i and s_j. Let $E_{T,j}$ represent the EC of s_i for transmitting κ bits to s_j at t_j

$$E_{T,j} = \kappa\varepsilon_1 + \kappa d_{i,j}^{\gamma}\varepsilon_2 \tag{4}$$

where ε_1 and ε_2 denote the EC per bit for the transmitting and amplifier circuit, respectively, and γ denotes the path loss exponent. Let $E_{R,j}$ denote the EC of s_i for receiving κ bits from s_j at t_j. Let $E_{i,j}^{\text{con}}$ be the total energy consumed by the sensor s_i at time slot t_j

$$E_{i,j}^{\text{con}} = E_{S,j} + E_{T,j} + E_{R,j}. \tag{5}$$

Let $\xi_{i,j}$ be the Boolean variable representing the sensor s_i that is sensing at t_j

$$\xi_{i,j} = \begin{cases} 1 & \text{if sensor } s_i \text{ sensing at time } t_j \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

Let $E_{i,j}^{\text{rem}}$ denote the remaining energy of the sensor s_i at t_j. The value of $E_{i,j}^{\text{rem}}$ can be evaluated according to the following:

$$E_{i,j}^{\text{rem}} = E_{i,j-1}^{\text{rem}} + E_{i,j}^{\text{crg}} * \phi_{i,j}^{\text{crg}} - E_{i,j}^{\text{con}} * \xi_{i,j}. \tag{7}$$

The problem statement is presented in Section III-C.

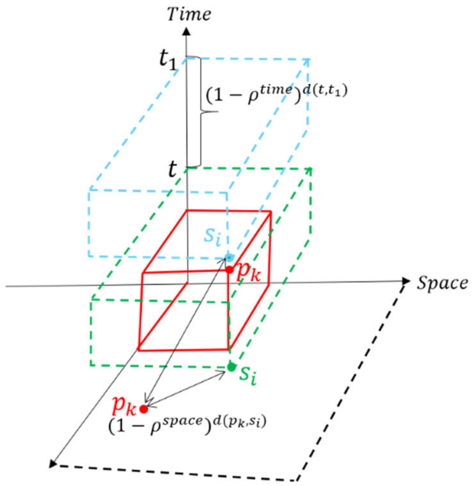


Fig. 3. Surveillance quality representation from space and time point of view.

C. Problem Statement

This section presents the definition of SQ. Then, the problem formulation, goal, and related constraints will be proposed. As shown in Fig. 3, the SQ can be examined from both space and time points of view. Let $P = \{p_1, \ldots, p_k \ldots, p_{|P|}\}$ be the set of points in the monitoring region $\mathcal{R}$, where $1 \le k \le |P|$. Fig. 3 shows an example of calculating the SQ of any point p_k in $\mathcal{R}$. The green and sky-blue dotted lines represent the sensing range of the sensor s_i at t and t_1 time slots. Let $\lambda_{p_k,i}$ be the Boolean variable representing the point p_k that falls in the sensing range of the sensor s_i. The notation $\lambda_{p_k,i}$ is given in the following:

$$\lambda_{p_k,i} = \begin{cases} 1, & \text{if } p_k \text{ falls in the sensing range of } s_i \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

This article adopts the ant pheromone algorithm to calculate the STSQs of all points in the network. Initially, the SSQ of point p_k is presented. If point p_k falls in the sensing range of the sensor $s_i(\lambda_{p_k,i} = 1)$, the SSQ of p_k is one. Similarly, if the point p_k does not fall in the sensing range of the sensor s_i, $(\lambda_{p_k,i} \neq 1)$, the SSQ is calculated based on the distance between the point p_k and sensor s_i. Let $d(p_k, s_i)$ be the spatial distance between point p_k and sensor s_i. According to the ant pheromone algorithm, the SSQ of point p_k, which does not belong to the sensing range of s_i, is $(1-\rho^{\text{space}})^{d(p_k,s_i)}$, where ρ^{space} be the loss rate of SSQ per unit distance.

The following presents the temporal surveillance quality (TSQ) calculation of point p_k. If it is the case $(\xi_{i,t} = 1)$, the TSQ of point p_k is one. On the other hand, if s_i is not sensing at time $t(\xi_{i,t} \neq 1)$, the TSQ is calculated based on the most recent sensing of s_i. Assume that the most recent sensing time of s_i is t_1, $(\xi_{i,t_1} = 1)$. Then, the TSQ of the point p_k is based on the distance between the time slots t and t_1. Let $d(t, t_1)$ be the time difference between t and t_1 in terms of time slots, that is to say, the TSQ of p_k is $(1-\rho^{\text{time}})^{d(t,t_1)}$, where ρ^{time} denotes the loss rate of TSQ per unit of time.

Let $\delta_{p_k,t}$ be the STSQ of point p_k at t. The four possible cases of $\delta_{p_k,t}$ are shown in the following (9), as shown at the bottom of the page.

According to (9), the STSQ of all the points in $\mathcal{R}$ can be derived, that is to say, the total SQ of the network can be calculated based on the STSQ of all the points in $\mathcal{R}$ overall time instants $t \in T$. Let x be a schedule. Let Q^x denote the total SQ of the network. The value of Q^x is given in the following:

$$Q^x = \sum_{p_k \in R} \sum_{t \in T} \delta_{p_k,t}. \tag{10}$$

Let Φ denote the set of all possible schedules. Given a schedule $x \in \Phi$, the proposed algorithm aims to maximize the STSQ of all the points $p \in \mathcal{R}$ overall time instants $t \in T$. Let Q^{best} be the best schedule with maximal STSQ. The objective function is given in (11).

1) Objective Function:

$$Q^{\text{best}} = \max_{x \in \Phi} \sum_{p_k \in R} \sum_{t \in T} \delta_{p_k,t}. \tag{11}$$

Expression (11) should be achieved under the following constraints. Let $\Omega^{\text{req}}_{i,j}$ be a Boolean variable to represent the sensor s_i requested a charging request at t_j

$$\Omega^{\text{req}}_{i,j} = \begin{cases} 1, & E^{\text{rem}}_{i,j} < e^{\text{req}}_{\text{th},i} \\ 0, & E^{\text{rem}}_{i,j} > e^{\text{req}}_{\text{th},i}. \end{cases} \tag{12}$$

Let $\Omega^{\text{wrk}}_{i,j}$ be a Boolean variable denoting that the sensor s_i is awake/sleep at t_j. We have

$$\Omega^{\text{wrk}}_{i,j} = \begin{cases} 1, & E^{\text{rem}}_{i,j} > e^{\min}_i \quad (\text{working}) \\ 0, & E^{\text{rem}}_{i,j} < e^{\min}_i \quad (\text{sleeping}). \end{cases} \tag{13}$$

The first constraint is to ensure that each sensor s_i must send the recharging request to MC. The recharging request constraint is given in (14).

2) Recharging Request Constraint:

$$\Omega^{\text{req}}_{i,j} \ge \Omega^{\text{wrk}}_{i,j} \quad \forall s_i \in S \ \forall t_j \in T. \tag{14}$$

The second constraint is the sensor energy recharging constraint. This constraint ensures that each sensor battery is fully recharged when the MC visits. Recall that $E^{\max}_S$ denotes the maximum battery capacity of each sensor node. Let $T_i = [t^1_i, t^h_i]$ be the time period that MC is located in the

$$\delta_{p_k,t} = \begin{cases} 1, & \text{if } \lambda_{p_k,i} = 1, \xi_{i,t} = 1 \\ \left(1-\rho^{\text{time}}\right)^{d(t,t_1)}, & \text{if } \lambda_{p_k,i} = 1, \xi_{i,t} \neq 1 \\ (1-\rho^{\text{space}})^{d(p_k,s_i)}, & \text{if } \lambda_{p_k,i} \neq 1, \xi_{i,t} = 1 \\ (1-\rho^{\text{space}})^{d(p_k,s_i)} \times \left(1-\rho^{\text{time}}\right)^{d(t,t_1)}, & \text{if } \lambda_{p_k,i} \neq 1, \xi_{i,t} \neq 1 \end{cases} \tag{9}$$

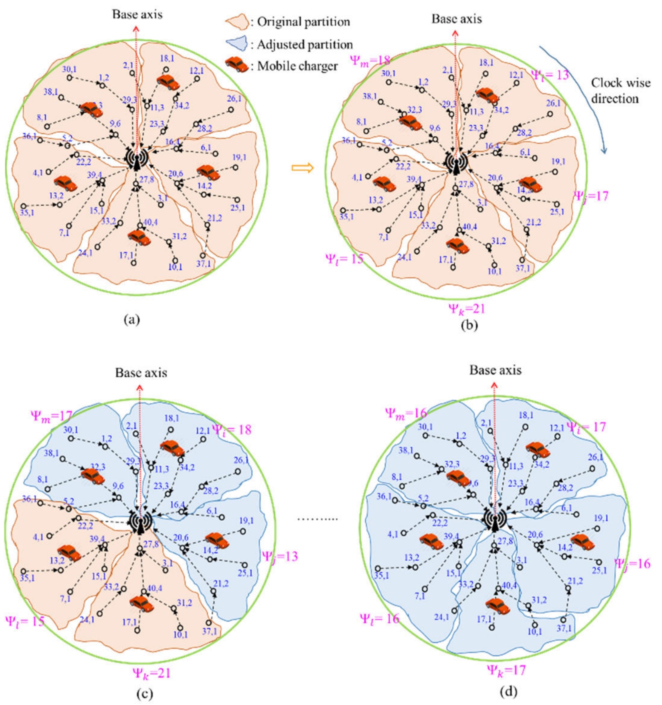


Fig. 4. (a) Equally partitioned subregions. (b) BS calculates RL of each subregion. (c) BS compares the neighboring RL values, i.e., Ψ_i with Ψ_j and Ψ_k. (d) Adjusted partitions based on RL.

charging range of s_i. Recall that $E_{i,t}^{\text{crg}}$ denote the total obtained energy of s_i from the MC at t. The sensor energy recharging constraint guarantees that each sensor can be fully recharged, given in (15).

3) Sensor Energy Recharging Constraint:

$$\int_{t_i^1}^{t_i^h} E_{i,t}^{\text{crg}} dt \geq E_S^{\max} \quad \text{where} \quad 0 \leq i \leq n. \tag{15}$$

IV. Algorithm Description

A WLARS mechanism is proposed in this article. The proposed three phases are network planning (NP) phase, recharging request determination (RRD) phase, and ABC phase. The NP phase partitions the network into many subregions based on the sensor's RL, aiming to distribute the charging load of the MCs evenly. Each sensor determines its threshold value based on the QT in the RRD phase. In the final phase, the adaptive buffer of the MC and cooperation between the neighboring MCs will be presented.

A. Network Planning (NP) Phase

This phase evenly partitions the sensors into fan-shaped subregions such that the charging load of each MC is balanced. Assume that the number of MCs in the network is m. In the network initialization, the BS partitions the region into equal-sized m subregions according to the base axis. Assuming that $m = 5$, the region is partitioned into five equal-sized subregions, as shown in Fig. 4(a).

In WSN, sensor nodes continuously execute sensing and data forwarding operations in each round. Sensor nodes may have different ECs due to their uneven RLs. By applying any standard routing algorithm, the RL of all sensors in the network can be known. The density of sensors increases more data, which should be forwarded. It leads to the sensors with low remaining energy.

Let Ψ_i denote the RL of the ith subregion. After partitioning the region into m subregions, the RL of each partition is calculated by BS, as shown in Fig. 4(b). Then, the RL of each subregion is compared with the RL of neighboring subregions in a clockwise direction.

Let Ψ_i, Ψ_j, and Ψ_m denote RL values of three neighbors, as shown in Fig. 4(b). The BS will compare Ψ_i with Ψ_j. If $\Psi_i < \Psi_j$, the ith subregion will include some sensors that belong to the jth subregion. Similarly, if $\Psi_i > \Psi_j$, the jth subregion will include some sensors that belong to the ith subregion. The scenario of $\Psi_j < \Psi_k$ is shown in Fig. 4(c). All the above operations are similar when comparing Ψ_i and Ψ_m. The BS repeats the above process for all the remaining subregions (i.e., from Ψ_j to Ψ_m). The adjusted partitions are shown in Fig. 4(d).

B. Recharging Request Determination (RRD) Phase

The discharging rate of each sensor highly depends on its forwarding load. Therefore, the on-demand charging policy is developed in this article. The value of $e^{\min}$ is used for basic operations, which are predetermined. In this phase, the value of $e_{\text{th},i}^{\text{req},t}$ is calculated in detail.

All the requests from the sensors in each subregion are stored in the queue of the corresponding MC. This concept is similar to the concept of QT. This article considers the MCs as servers and the sensors as the clients. In this article, $M/G/1{:}/N/N$ QT model is adopted to calculate $e_{\text{th},i}^{\text{req},t}$. Let λ and μ denote the arrival and service rates of the requested sensors in the network, respectively. The number of servers assigned to each subregion is one. The queue length is N, while the population of the network is N sensors. Therefore, the Kendall–Lee notation ($M/G/1{:}/N/N$) is denoted as $\lambda/\mu/1{:}/N/N$. Let L_q and L_s denote the average lengths of the queue and system, respectively. Let W_q and W_s denote the average waiting times (AWTs) in queue and system, respectively. Assume that each MC knows all the parameters L_q, L_s, W_q, and W_s. Let M^k denote the kth MC. The following calculates the threshold value of sensors that belongs to M^k.

Let $L_q^{t,k}$ be the queue size of kth MC at current time t. During the network initialization, each M^k broadcasts its schedule $L_q^{t,k} = \{\emptyset\}$ to its corresponding region. Let s_i^k be the sensor s_i belonging to M^k. Assume that sensor s_i^k wants to send the request to M^k. Let r denote the radius of the whole network. Let v denote the speed of the MC. Let $T^{\max}$ denote the time required for MC to travel from one location to another in its region. The notation $T^{\max}$ is calculated in the following:

$$T^{\max} = \frac{2r}{v}. \tag{16}$$

Let $e_{i,t}^{\text{disch}}$ be the discharging rate of s_i^k at time t and $e_{\text{th},i}^{\text{req},t}$ denote the threshold value of s_i^k at time t. The value

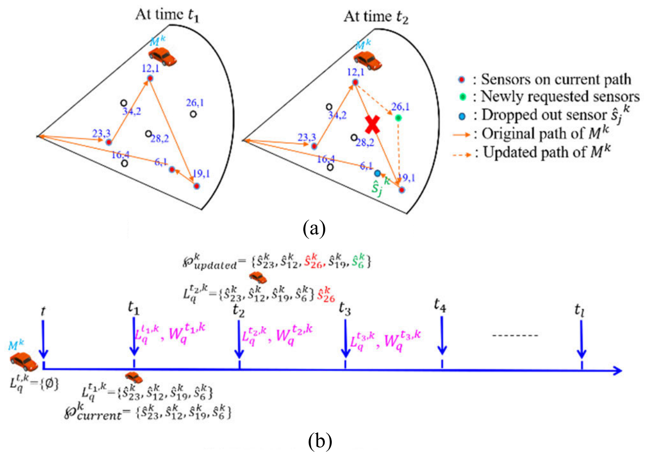


Fig. 5. Example of path construction and scheduling of M^k. (a) Example path of M^k at time t_1 and t_2. (b) Timeline schedule of M^k.

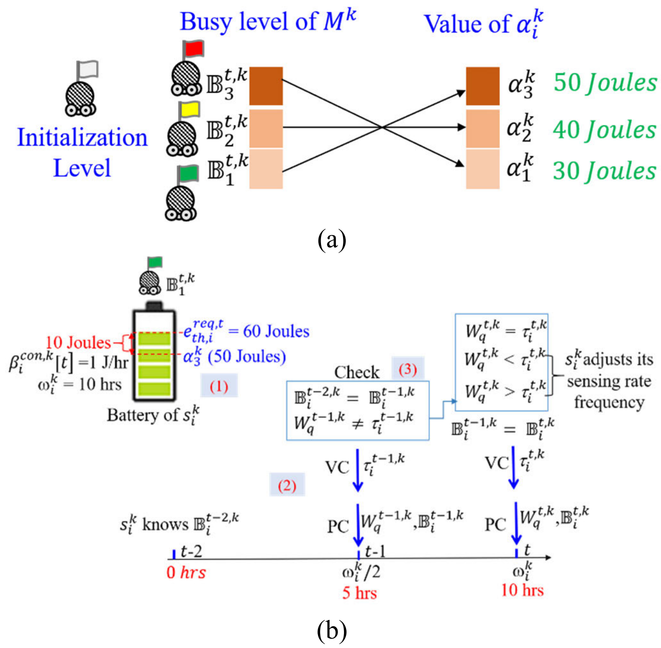


Fig. 6. (a) Busy levels of M^k and (b) duration check of s_i^k.

of $e_{\text{th},i}^{\text{req},t}$ is shown in the following:

$$e_{\text{th},i}^{\text{req},t} = e^{\min} + \left(T^{\max} * L_q^{t,k} + W_q^{t,k}\right) * e_{i,t}^{\text{disch}} \tag{17}$$

where $W_q^{t,k}$ denotes the AWT of the kth MC at current time t. The value of $W_q^{t,k}$ will be determined according to (22) of phase III. Fig. 5 presents an example to illustrate the above-discussed operation. Fig. 5(a) and (b) shows an example of the recharging path and the timeline schedule of M^k. Fig. 5(b) shows that during the network initialization, $L_q^{t,k} = \{\emptyset\}$. Let $\wp_{\text{current}}^k$ and $\wp_{\text{updated}}^k$ denote the current and updated paths of M^k, respectively. Assume that M^k receives the recharging requests from the sensors $\hat{s}_{23}^k$, $\hat{s}_{12}^k$, $\hat{s}_{19}^k$, and $\hat{s}_6^k$ at time t_1. Then, M^k forms the shortest Hamiltonian cycle for $\hat{s}_{23}^k$, $\hat{s}_{12}^k$, $\hat{s}_{19}^k$, and $\hat{s}_6^k$. Whenever M^k moves to each sensor, $L_q^{t_1,k}$ and $W_q^{t_1,k}$ at a corresponding time will be broadcasted using a quorum system. Assume that M^k is charging $\hat{s}_{23}^k$ at time t_2 and sensor $\hat{s}_{26}^k$ sends the recharging request to M^k at t_2. M^k will update its path by inserting $\hat{s}_{26}^k$ to the path. Assume that M^k cannot support its energy for recharging the sensor $\hat{s}_6^k$. Thus, $\hat{s}_6^k$ will be the dropped-out sensor. How to recharge the dropped-out sensor will be shown in phase III. After recharging each sensor, M^k will check its queue to know whether any new requests have been received in the local networks, aiming to utilize the path of M^k.

According to (17), after a specific system running time, the value of $e_{\text{th},i}^{\text{req},t}$ will be equal to $E_{\max}$, i.e., $e_{\text{th},i}^{\text{req},t} = E_{\max}$. This situation makes every sensor send the recharging request, leading to the bad situation that M^k is very busy. In this article, the sensors will utilize their battery according to the status of M^k. Let α_i^k denote the ith value of α, which belongs to M^k. In this article, α_i^k is used to control the sensor battery. The busy level of MC is introduced to design the operations of this phase. The busy level of an MC refers to the level of demand for MC charging services from the sensor nodes in the network. It can be defined as the ratio of the time the MC spends charging the sensor nodes to the total time it spends within the network. In other words, the busy level of the MC indicates how occupied it is with providing charging services to the sensor nodes.

In QT, the busy level of an MC can be defined as the utilization rate of the charger. The utilization rate is the fraction of the time the charger is busy serving requests to charge sensor nodes. In QT, the utilization rate can be calculated by dividing the time the charger is busy for charging sensor nodes by the total time period considered. The busy level or utilization rate is an important parameter in QT as it impacts the performance of the system, including the AWT and queue length.

Let $\mathbb{B}_i^{t,k}$ denote the ith busy level of M^k at time t. The higher value of i in $\mathbb{B}_i^{t,k}$, the busier M^k is, which is shown in Fig. 6(a). All the sensors in the corresponding region of M^k are assumed to know its busy level, and the value of α is adjusted by them according to $\mathbb{B}_i^{t,k}$.

Let ζ^k denote the queue of M^k. M^k stays in initialization status (M^k with a white flag), as shown in Fig. 6(a). Once the network is initialized, the status of M^k will be changed. If M^k is in status $\mathbb{B}_1^{t,k}$, this means that M^k status is free, and there are no requested sensors in ζ^k. Based on the quorum, each sensor can know the busy level of M^k ($\mathbb{B}_1^{t,k}$) along with $L_q^{t_1,k}$ and $W_q^{t_1,k}$. Based on the busy level of M^k ($\mathbb{B}_1^{t,k}$), the sensor knows the value of α_i^k. For instance, if M^k is in $\mathbb{B}_1^{t,k}$, $\mathbb{B}_2^{t,k}$, and $\mathbb{B}_3^{t,k}$, the values of α_1^k, α_2^k, and α_3^k are 50, 40, and 30 J, respectively. Then, each sensor s_i^k calculates its threshold value $e_{\text{th},i}^{\text{req},t}$. If $e_{\text{th},i}^{\text{req},t} \geq \alpha_i^k$, according to the current status of M^k, then the sensor will not send the recharging request. Thus, we restrict the sensor's battery for sending the recharging request, aiming to reduce the number of requests sent to M^k. In the worst case, at some time point, there might be sensors that send the recharging request to M^k when its $e_{\text{th},i}^{\text{req},t} = 100$ J, where $E_{\max} = 100$ J. To solve this issue, we assume that any sensor s_i^k whose $e_{\text{th},i}^{\text{req},t} \geq \alpha_1^k$ (50 J) will not send out its recharging request. Each sensor will perform the duration check. The procedure of duration check is detailed in the following. The duration check is divided into two types: physical check (PC) and virtual check (VC). The PC uses the quorum to know the waiting time of M^k at the corresponding time ($W_q^{t,k}$). Let $\tau_i^{t,k}$ denote the remaining lifetime of s_i^k at time t. The sensor s_i^k should VC use its energy to know its remaining lifetime ($\tau_i^{t,k}$). To know its $\tau_i^{t,k}$, the sensor s_i^k should estimate its EC rate. The value of $\tau_i^{t,k}$ depends on its EC rate, while the EC rate may always vary because the forwarding load of s_i^k changes with time. Let $\beta_i^{\text{con},k}$ denote the EC rate of the sensor s_i^k. The calculation of $\beta_i^{\text{con},k}$ is based on the prediction formula as shown in (18).

The prediction formula is divided into two parts, including historical and recent EC rates. Let Ω denote the fixed time period to calculate $\beta_i^{\text{con},k}$. Recall that $E_{i,t}^{\text{rem}}$ denotes the remaining energy of the sensor s_i at time t. Similarly, $E_{i,t}^{\text{rem},k}$ and $E_{i,t-\Omega}^{\text{rem},k}$ denote the remaining energy of the sensor s_i^k at time t and $t-\Omega$, respectively. Therefore, the recent EC rate during the time period $[t, t-\Omega]$ is $((\rho * (E_{i,t-\Omega}^{\text{rem},k} - E_{i,t}^{\text{rem},k}))/\Omega)$. The historical EC rate of s_i^k is expressed by $\beta_i^{\text{con},k}[t-\Omega]$, where ρ is the coefficient to adjust the historical and recent EC rates. When the value of ρ is larger, $\beta_i^{\text{con},k}$ is more dependent on the recent EC rate. Let $\beta_i^{\text{con},k}[t]$ denote the EC rate of the sensor s_i^k from the initial time t_0 to time t. The value of $\beta_i^{\text{con},k}[t]$ can be determined by applying as follows:

$$\beta_i^{\text{con},k}[t] = (1-\rho)\,\beta_i^{\text{con},k}[t-\Omega] + \frac{\rho * \left(E_{i,t-\Omega}^{\text{rem},k} - E_{i,t}^{\text{rem},k}\right)}{\Omega}. \tag{18}$$

According to (18), assume that $\beta_i^{\text{con},k}[t]$ is 1 J/h. Assume that M^k stays in $\mathbb{B}_1^{t,k}$ level and $e_{\text{th},i}^{\text{req},t} \geq \alpha_3^k$ (i.e., 60 J > 50 J), which means 10 J greater than α_3^k as denoted by (1) in Fig. 6(b). Therefore, the energy of the sensor s_i^k (10 J) will support for 10 h since $\beta_i^{\text{con},k}[t]$ is 1 J/h. Then, the sensor s_i^k will perform a duration check, denoted by (2) in Fig. 6(b). Let ω_i^k denote the time duration (10 h), where the energy of the sensor s_i^k will support. The value of ω_i^k is known based on $\beta_i^{\text{con},k}[t]$. As shown in Fig. 6(b), $\omega_i^k = 10$ h. The sensor performs PC and VC at $\omega_i^k/2$ and ω_i^k h, respectively. At $\omega_i^k/2$ h, the sensor s_i^k will perform PC to know $W_q^{t-1,k}$ and $\mathbb{B}_i^{t-1,k}$ and VC its $\tau_i^{t-1,k}$. Then, the sensor will check whether or not $\mathbb{B}_i^{t-2,k} = \mathbb{B}_i^{t-1,k}$ and $W_q^{t-1,k} = \tau_i^{t-1,k}$, denoted by (3) in Fig. 6(b). If $\mathbb{B}_i^{t-2,k} = \mathbb{B}_i^{t-1,k}$ and $W_q^{t-1,k} \neq \tau_i^{t-1,k}$, the sensor s_i^k will again perform PC to know $W_q^{t,k}$, $\mathbb{B}_i^{t,k}$, and VC its $\tau_i^{t,k}$ at ω_i^k h. After performing the duration check, three cases may occur at ω_i^k h, as shown in the following.

1) *Best Case Criterion:* At ω_i^k h, if $\mathbb{B}_i^{t-1,k} = \mathbb{B}_i^{t,k}$ and $W_q^{t,k} = \tau_i^{t,k}$, the sensor s_i^k will send the recharging request to M^k.
2) *Average Case Criterion:* At ω_i^k h, if $\mathbb{B}_i^{t-1,k} = \mathbb{B}_i^{t,k}$ and $W_q^{t,k} < \tau_i^{t,k}$, the sensor s_i^k will increase its sensing rate frequency.
3) *Worst Case Criterion:* At ω_i^k h, if $\mathbb{B}_i^{t-1,k} = \mathbb{B}_i^{t,k}$ and $W_q^{t,k} > \tau_i^{t,k}$, the sensor s_i^k will decrease its sensing rate frequency.

On the other hand, if $\mathbb{B}_i^{t-2,k} \neq \mathbb{B}_i^{t-1,k}$ or $\mathbb{B}_i^{t-1,k} \neq \mathbb{B}_i^{t,k}$ at $\omega_i^k/2$ or ω_i^k h, the value of α_i^k is adjusted according to $\mathbb{B}_i^{t,k}$ at the corresponding time.

The determination of sensing rate frequency and its adjustment are shown. Let $\mathcal{S}_{i,t}^{\text{freq},k}$ be the sensing rate frequency of the sensor s_i^k at time t. Recall that $E_{i,t}^{\text{rem},k}$ denotes the remaining energy of s_i^k at time t. Recall that $\beta_{i,t}^{\text{con},k}$ and $W_q^{t,k}$ are the EC rate of s_i^k and waiting time of M^k at time t, respectively

$$\mathcal{S}_{i,t}^{\text{freq},k} = \frac{E_{i,t}^{\text{rem},k}}{\beta_{i,t}^{\text{con}} * W_q^{t,k}}. \tag{19}$$

The following presents an example of decreasing $\mathcal{S}_{i,t}^{\text{freq},k}$ if the worst case occurs, i.e., $W_q^{t,k} > \tau_i^{t,k}$. Assume that $E_{i,t}^{\text{rem},k} = 2$ J, $\beta_{i,t}^{\text{con}} = 1$ J/h, and $W_q^{t,k} = 4$ h. Since its $\mathrm{S}_{i,t}^{\text{freq},k} = 1$ J/1 h, regardless of $W_q^{t,k}$, s_i^k is scheduled in the first two time slots, as shown in the upper part of Fig. 7(a), that is to say, the sensor s_i^k is scheduled without adjusting its $\mathcal{S}_{i,t}^{\text{freq},k}$. Assume that $E_{i,t}^{\text{rem},k} = 2$ J, $\beta_{i,t}^{\text{con}} = 1$ J/h, and $W_q^{t,k} = 4$ h. Substituting these parameters in (19), $\mathcal{S}_{i,t}^{\text{freq},k} = 1$ J/2h. The lower part of Fig. 7(a) shows the updated schedule of s_i^k. At the current time t, once $\mathcal{S}_{i,t}^{\text{freq},k}$ is finished, s_i^k will check the busy level of M^k, i.e., ($\mathbb{B}_i^{t-1,k} = \mathbb{B}_i^{t,k}$). If $\mathbb{B}_i^{t-1,k} = \mathbb{B}_i^{t,k}$, s_i^k will send the recharging request to M^k. Otherwise, the value of α_i^k is adjusted according to $\mathbb{B}_i^{t,k}$.

The above procedure is similar to increase $\mathcal{S}_{i,t}^{\text{freq},k}$ if the average case occurs, i.e., $W_q^{t,k} < \tau_i^{t,k}$. Fig. 7(b) presents an example when s_i^k increases its $\mathcal{S}_{i,t}^{\text{freq},k}$. Therefore, the sensor s_i^k can increase/decrease its sensing frequency according to the waiting time of M^k, aiming to reduce the number of recharging requests sent to M^k.

In phase II, the sensors can utilize their battery based on the busy level of MC. During the execution of phase II, the sensor state is not static. To clearly illustrate the sensor states, the sensor state diagram of a sensor s_i^k is shown in Fig. 8. The MC adaptive buffer and recharging scheduling are discussed in the following phase.

C. Adaptive Buffer and Cooperation (ABC) Phase

This article uses the recharging path construction proposed in our previous work [26], named multiple passer-by update (MPU) strategies for scheduling the MC. In this phase, the SSQ benefit (SSQB) is considered for executing the recharging operation. Although the sensor battery is constrained by the value of α_i^k and the recharging requests are sent according to $\mathbb{B}_i^{t,k}$, at a certain time point t, there might be many recharging requests stored in the buffer of MC, considering the scalability of the network. This situation causes some sensors to exhaust their energy because of the higher charging latency. This article assumes that MC has an adaptive buffer to solve this issue.

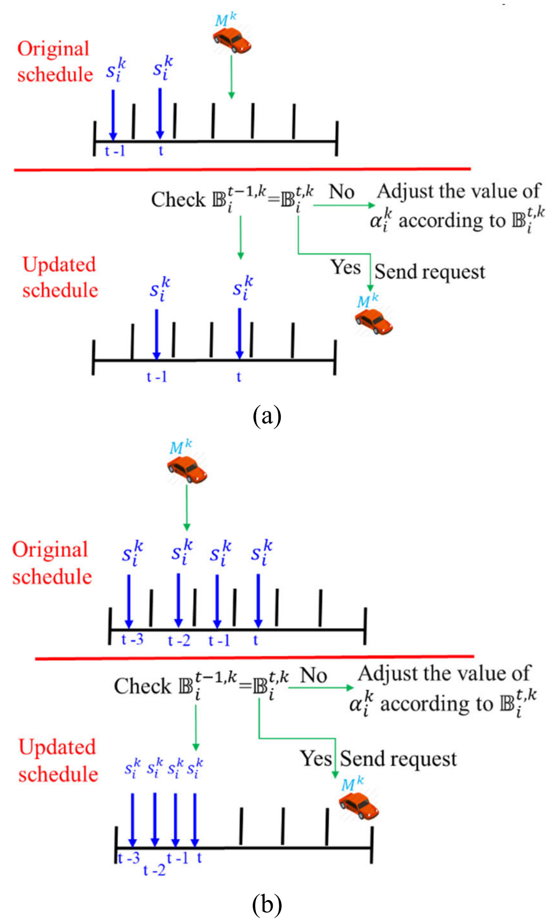


Fig. 7. Sensing rate frequency adjustment of the sensor s_i^k. (a) Reducing and (b) increasing the sensing rate frequency.

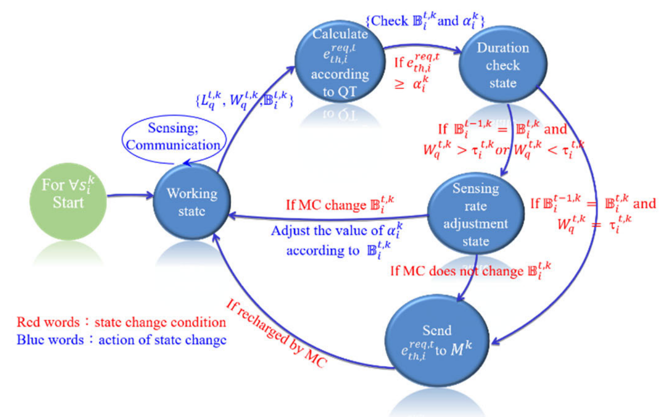


Fig. 8. State diagram of a sensor s_i^k.

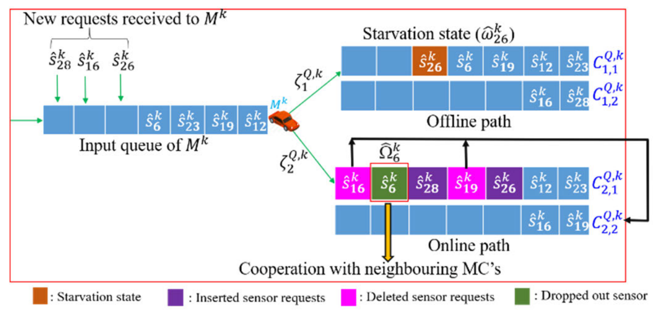


Fig. 9. Adaptive buffer of M^k.

The following discusses the adaptive buffer of M^k. It is assumed that M^k buffer has two priority queues.

Let $\zeta_1^{Q,k}$ denote the queue for the offline recharging mechanism. Let $\hat{s}_i^k$ denote that sensor s_i^k has already sent the recharging request to M^k and has a lower chance of recharging for a long time; it stays in the starvation state. Let $\hat{\omega}_i^k$ denote the starvation state of $\hat{s}_i^k$. Let $C_{1,1}^{Q,k}$ denote the first cycle of M^k when adopting $\zeta_1^{Q,k}$. In $\zeta_1^{Q,k}$, while inserting the new requests of sensors that have higher SSQB, if any sensor switches to starvation state ($\hat{\omega}_i^k$), M^k will close its buffer (i.e., $C_{1,1}^{Q,k}$) and all the new requests are stored in the second buffer $C_{1,2}^{Q,k}$, as shown in Fig. 9.

Let $\zeta_2^{Q,k}$ denote the queue for the online recharging mechanism. In this $\zeta_2^{Q,k}$, although sensor $\hat{s}_i^k \in M^k$ sends the recharging request very late, if it is closer to the path of M^k, based on SSQB, it could be inserted to $\wp_{\text{current}}^k$, aiming to optimize $\wp_{\text{current}}^k$. Since each sensor forwarding load dynamic changes, the charging load of each MC might be different. The energy of M^k might not support recharging all the sensors in $C_{2,1}^{Q,k}$. Therefore, there is cooperation between neighboring MCs. An example of the adaptive buffer of M^k is shown in Fig. 9.

Initially, all the sensors are stored in sequence in the input queue. Based on the SSQB, the sensor requests are rearranged in $\zeta_1^{Q,k}$ and $\zeta_2^{Q,k}$, as shown in Fig. 9. In the offline path ($\zeta_1^{Q,k}$), assume that $\hat{s}_{26}^k$ switches to the starvation state in $C_{1,1}^{Q,k}$. M^k closes its buffer and the new sensor requests, such as $\hat{s}_{28}^k$ and $\hat{s}_{16}^k$ that are stored in $C_{1,2}^{Q,k}$. On the other hand, in the online path ($\zeta_2^{Q,k}$), $\hat{s}_{26}^k$ and $\hat{s}_{28}^k$ are inserted to $C_{2,1}^{Q,k}$, while $\hat{s}_{19}^k$ and $\hat{s}_{16}^k$ are deleted from $C_{2,1}^{Q,k}$ by considering the SSQB. Besides, M^k might not support recharging a few sensors, which are stored at the last position of the buffer, although their SSQB is higher. Let $\hat{\Omega}_i^k$ denote the ith dropped out the sensor of M^k. As shown in Fig. 9, $\hat{\Omega}_6^k$ is the dropped-out sensor. There is cooperation between the neighboring MCs to recharge the dropped-out sensor ($\hat{\Omega}_6^k$).

In the following, the cooperation between the neighboring MCs is discussed. The MCs can share the charging load with neighbors to achieve more charging efficiency. As shown in Fig. 10, assume that M^k can communicate with neighboring M^{k-1} and M^{k+1} to recharge $\hat{\Omega}_i^k$. Both M^{k-1} and M^{k+1} include $\hat{\Omega}_i^k$ to their schedule and checks the following two conditions. Let $W_{q,i}^{t,k-1}$ denote the waiting time of M^{k-1} for executing the recharging of $\hat{s}_i^{k-1}$ at time t. Let t_i^{crg} denote the recharging time of the sensor s_i. Let q_M^{crg} be the rate at which the MC recharges. Let t_i^{mov} be the moving time of the sensor s_i. Recall that e_i^{disch} denotes the discharging rate of sensor s_i. The values of t_i^{crg} and t_i^{mov} are calculated in the following equations:

$$t_i^{\text{crg}} = \frac{E_S^{\max} - \left(E_i^{\text{rem}} - e_i^{\text{disch}} * t_i^{\text{mov}}\right)}{q_M^{\text{crg}}} \quad 1 \le i \le n \tag{20}$$

$$t_i^{\text{mov}} = t_{i-1}^{\text{mov}} + t_{i-1}^{\text{crg}} + \frac{d_{i-1,i}}{v}, \quad 1 \le i \le n \tag{21}$$

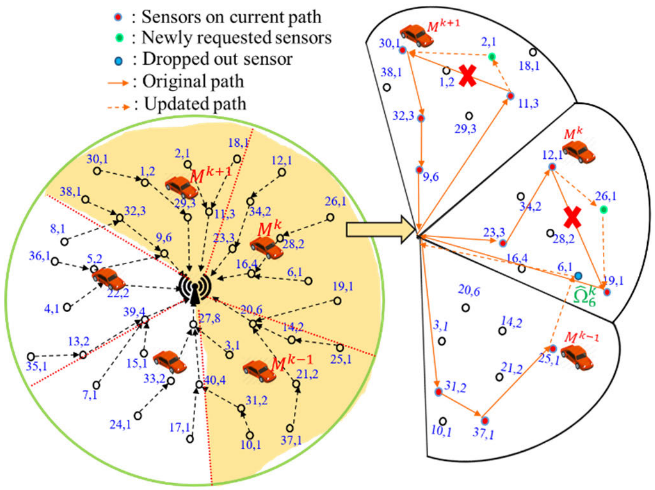


Fig. 10. Collaborative schedule between the neighboring MCs.

where $d_{i-1,i}$ denotes the distance between s_{i-1} and s_i and s_0 is BS. The following illustrates the procedure of M^{k-1} to include $\hat{\Omega}_i^k$ to its schedule.

1) *Waiting Time Criterion:* M^{k-1} checks the waiting time criterion of $\hat{\Omega}_i^k$, i.e.,

$$W_{q,i}^{t,k-1} = \begin{cases} t_i^{\text{mov}}, & i = 1 \\ W_{q,i-1}^{t,k-1} + t_{i-1}^{\text{crg}} + t_i^{\text{mov}}, & 2 \le i \le n. \end{cases} \quad (22)$$

2) *Remaining Energy Criterion:* The sensors $\hat{\Omega}_i^k$ should satisfy

$$E_{i,k}^{\text{rem}} - e_{i,k}^{\text{disch}} * W_{q,i}^{t,k-1} > 0. \quad (23)$$

The first condition is to check whether the waiting time of M^{k-1} satisfies, and the second condition is to check whether $\hat{\Omega}_i^k$ is alive until M^{k-1} comes to recharge it. All the above procedure is similar for M^{k+1} to include $\hat{\Omega}_i^k$ to its schedule. According to the above two conditions shown in (22) and (23), there might be three kinds of outcomes.

The first one is that only M^{k-1} satisfies the above two conditions, and therefore, M^{k-1} will include $\hat{\Omega}_i^k$ to its schedule and recharge it. The second outcome is both M^{k-1} and M^{k+1} is too busy and does not satisfy the two conditions; then, $\hat{\Omega}_i^k$ request will be abandoned. The third outcome is both M^{k-1} and M^{k+1} that satisfy the above two conditions. In this case, M^*, which has a shorter waiting time to recharge $\hat{\Omega}_i^k$, will include it in its schedule. Let $S_{\hat{\Omega}_i^k}^{\text{best}}$ denote the best schedule for $\hat{\Omega}_i^k$ to recharge

$$S_{\hat{\Omega}_i^k}^{\text{best}} = \begin{cases} M^{k-1}, & \text{if } W_{q,i}^{t,k+1} > W_{q,i}^{t,k-1} \\ M^{k+1}, & \text{otherwise}. \end{cases} \quad (24)$$

As shown in Fig. 10, M^k is not able to recharge $\hat{s}_6^k$. Therefore, the dropped-out sensor is $\hat{\Omega}_6^k$. M^k communicates with the neighboring M^{k-1} and M^{k+1} to include $\hat{\Omega}_6^k$ to their schedules to recharge it. Both M^{k-1} and M^{k+1} include $\hat{\Omega}_6^k$ to its schedule and check the two conditions shown in (22) and (23). Assume that $W_{q,6}^{t,k+1} > W_{q,6}^{t,k-1}$. Therefore, $S_{\hat{\Omega}_6^k}^{\text{best}} = M^{k-1}$. Thus, the cooperative scheduling increases the probability that a sensor with the highest SSQB is successfully recharged.

TABLE II
SIMULATION PARAMETERS

Parameters	Values
Simulator	Matlab R2019b
Network size	$600m$ x $400m$
Sensor nodes	300 - 700
Deployment	Random
Sensing range	$5-20m$
Sleep threshold e^{min}	0.3- 0.7J
Discharging rate e_i^{disch}	0.05J/s
E_S^{max}	3.6kJ
v	1 - $8m/s$
q_M^{crg}	3-15 J/s

V. PERFORMANCE EVALUATIONS

The WLARS algorithm is compared with the recharge scheduling algorithm for maximizing spatial and temporal qualities (RS-STQ) [26], multi-charger cooperation algorithm (MCCA) [21] and fuzzy logic-based charging schedule determination (FLCSD) [22] algorithms in terms of the analytic results. The RS-STQ considered the SSQ and TSQ of the sensors and constructed the path of the MC. The MCCA algorithm proposed a charging schedule based on the energy required by each cluster. The FLCSD algorithm scheduled the multiple MCs based on fuzzy logic by considering parameters, such as remaining energy, distance, node density, and EC rate.

A. Simulation Environment

The size of the considered monitoring region is 600×400 m. The sensors are randomly deployed in the given area. The sensors deployed in the region are 300–700. Table II presents the parameter settings of the simulation.

B. Simulation Results

Fig. 11 evaluates the STSQ for the proposed WLARS (MPU) and the existing three algorithms. The terms WLARS (MPU) and RS-STQ (MPU) mean that the simulation is conducted by adopting the MPU strategy for WLARS (MPU) and RS-STQ algorithms. In Fig. 11, the STSQ is measured by changing the deployed sensor nodes and $e^{\min}$. The value of $e^{\min}$ varies from 0.3 to 0.7 J. As shown in Fig. 11, the STSQ increases with sensors. It is because the distance between the sensors is decreased. Thus, the MC spends most of its energy recharging the sensors instead of its movement. On the other hand, the STSQ of four algorithms decreases with $e^{\min}$. It occurs because a higher $e^{\min}$ makes the sensor send the recharging request very quickly. Therefore, the charging load of each MC increases, leading to a lower value of STSQ. The proposed WLARS (MPU) performance is higher than the existing algorithms. It occurs because the proposed WLARS (MPU) is considered the SSQ. Besides, the sensing frequency of sensors is adjusted according to the busy level of MC, improving the TSQ. Therefore, the STSQ of the proposed

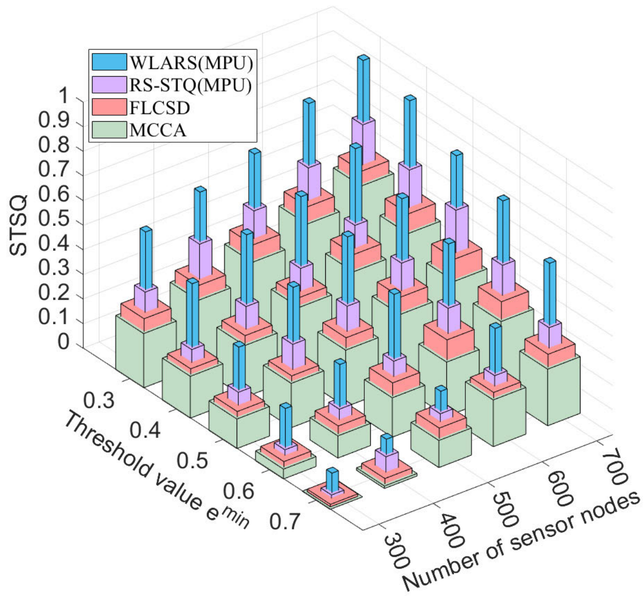


Fig. 11. Performance comparison of STSQ for the proposed WLARS (MPU) and existing RS-STQ (MPU), FLCSD, and MCCA algorithms.

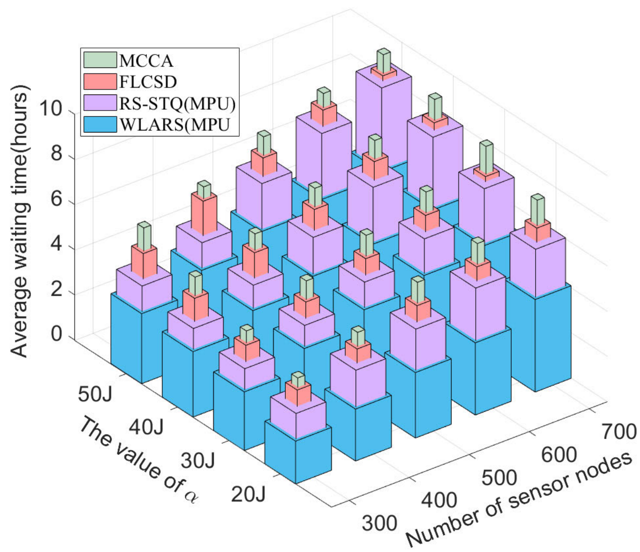


Fig. 12. Comparison of AWT for the proposed WLARS (MPU) and existing RS-STQ (MPU), FLCSD, and MCCA algorithms.

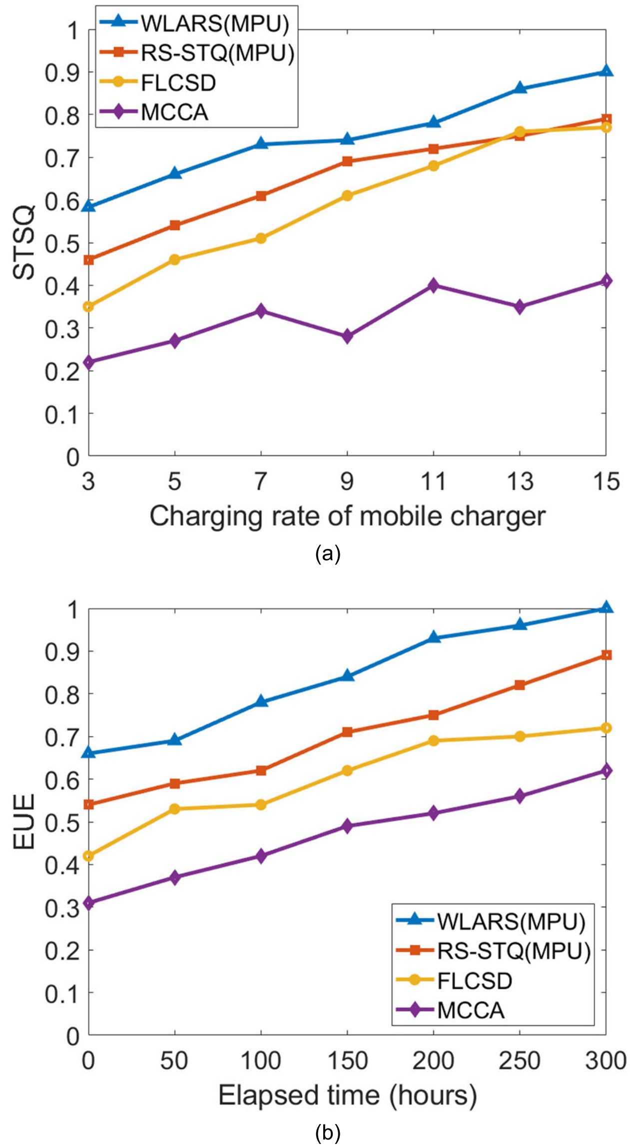


Fig. 13. Performance comparison of STSQ and EUE for the proposed WLARS (MPU) and existing RS-STQ (MPU), FLCSD, and MCCA algorithms. (a) Charging rate of MC versus STSQ. (b) Elapsed time versus EUE.

WLARS (MPU) is higher than those of the other three algorithms.

Fig. 12 compares the AWT of the proposed WLARS (MPU) with the existing three algorithms by varying the number of sensor nodes and the value of. Let $\hat{s}_p^k$ denote the $\hat{p}$ number of recharging requests received by M^k. Then, the AWT of all the sensors, which are waiting to recharge by M^k at time t, denoted by $W_{q,t}^{\text{wait},k}$ is shown in the following:

$$W_{q,t}^{\text{wait},k} = \frac{\sum_{i=1}^{|\hat{p}|} W_{q,i}^{t,k}}{|\hat{p}|}. \tag{25}$$

As shown in Fig. 12, the AWT increases with sensor nodes. It occurs because the number of recharging requested sensors increases with the number of deployed sensors, leading to the increase of AWT. On the other hand, the AWT decreases with α. It is due to higher α making the sensors send recharge requests to the MC frequently. Thus, MC might receive many requests, leading to a higher AWT. The AWT of WLARS (MPU) is very low because it considers the MCs cooperation based on the charging load.

Fig. 13(a) compares the STSQ by varying the charging rate of MC. The charging rate ranges from 3 to 15 J/s. As shown in Fig. 13(a), the STSQ of four algorithms increases with the charging rate of MC. It occurs because if the charging rate increases, the MC can recharge more sensors, leading to a higher STSQ. The proposed WLARS (MPU) considers the busy levels of MC and adjusts the sensing frequency of the sensors accordingly, leading to higher performance. This strategy controls the sensor's battery to send the recharging request. Therefore, all the requests of the other sensors will be recharged timely because of the less charging load.

Fig. 13(b) compares the energy usage effectiveness (EUE) by varying the elapsed time for the proposed WLARS (MPU),

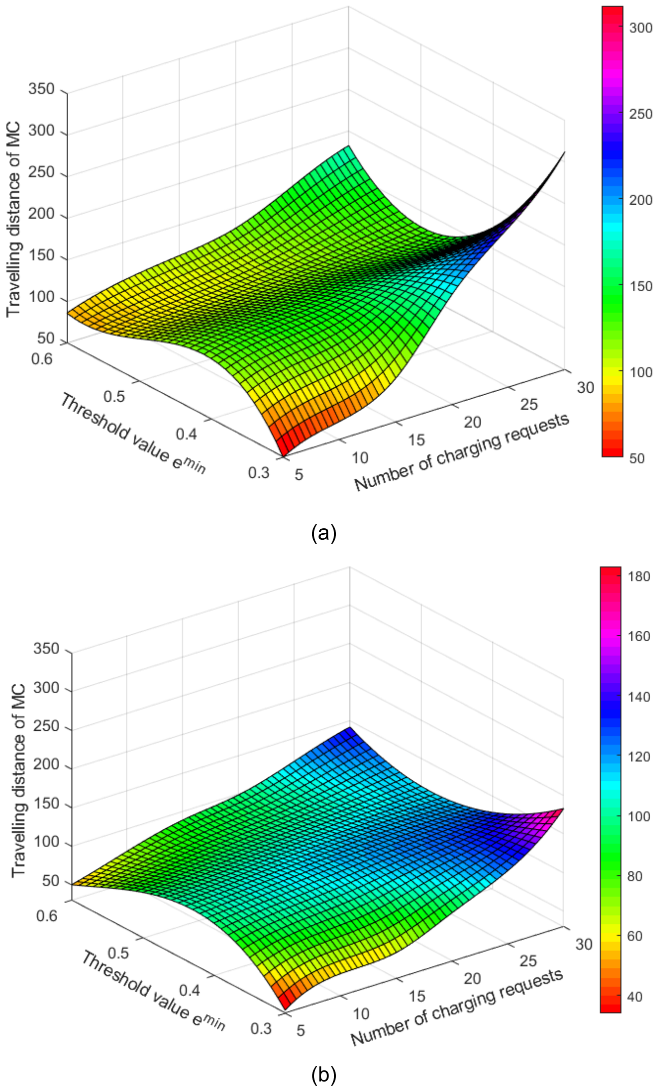


Fig. 14. Comparison of traveling distance of MC for the proposed WLARS (MPU) algorithm. Traveling distance of MC when the value of α is (a) 50 J and (b) 20 J.

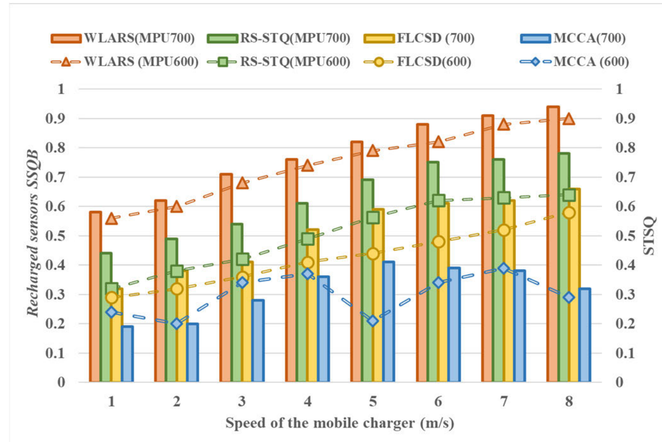


Fig. 15. Performance comparison of recharged sensors SSQB and STSQ by varying the speed of the MC.

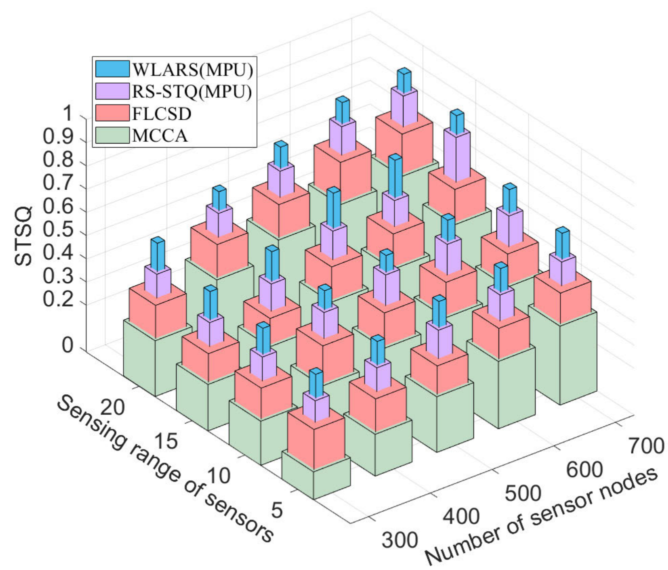


Fig. 16. Performance comparison of STSQ for the proposed WLARS (MPU) and existing RS-STQ (MPU), FLCSD, and MCCA algorithms.

RS-STQ (MPU), FLCSD, and MCCA algorithms. Let E_{crg}^k and E_{mov}^k denote the energies of M^k, which are consumed for charging and moving, respectively. The EUE of all four algorithms increases with the elapsed time. It occurs because as the elapsed time increases, the MC can recharge a greater number of sensors, leading to a higher EUE

$$\text{EUE} = \frac{E_{\text{crg}}^k}{E_{\text{crg}}^k + E_{\text{mov}}^k}. \tag{26}$$

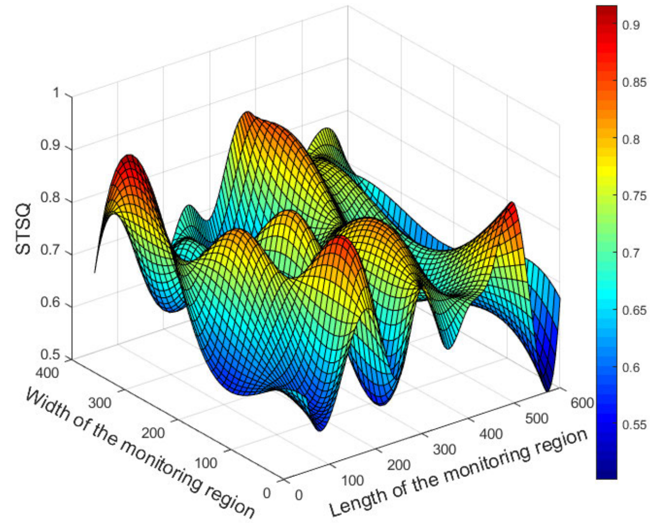
The MC recharges the closest sensors with higher STSQ, and the neighboring MCs will recharge the farthest sensors, leading to higher WLARS (MPU) performance. The MC will travel a shorter distance to recharge more sensors as time passes.

Fig. 14 compares the traveling distance of MC. The requests vary, ranging from 5 to 30. The value of $e^{\min}$ is set at 0.3–0.6 J. In Fig. 14(a) and (b), the value of α is set at 50 and 20 J, respectively. As shown in Fig. 14(a) and (b), if the MC receives more recharging requests, it will travel a long distance to recharge all the requested sensors. The traveling distance of MC increases with the value of $e^{\min}$. It occurs because a higher value of $e^{\min}$ leads the sensor to send the recharging request to the MC very quickly. Comparing Fig. 14(a) and (b), the traveling distance of MC in Fig. 14(b) is very small. It occurs because the lower value of α means that the MC might receive fewer recharging requests compared to Fig. 14(a), leading to a smaller traveling distance.

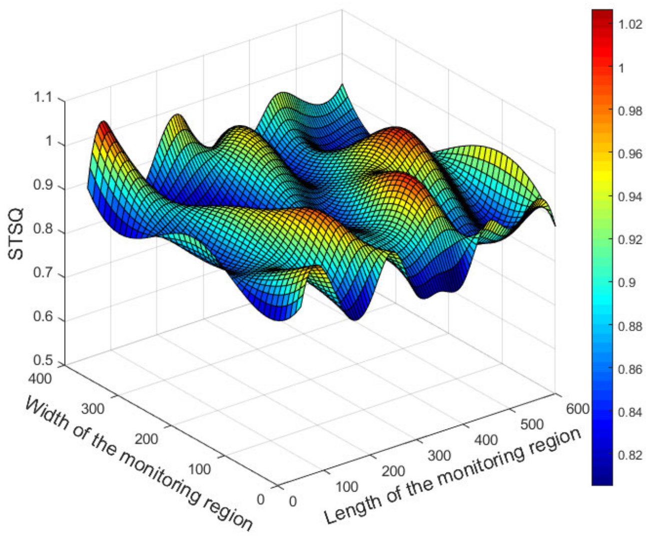
Fig. 15 compares the SSQB and STSQ. Let Ω_i be the length of the working time slot of s_i. Let c_i^x denote the SSQB of $\hat{s}_i$ at time slot $t_x \in T$, i.e.,

$$\text{SSQB} = \frac{\sum_{i=1}^{n} \sum_{j=t_1}^{t_h} \left(\phi_{i,j}^{\text{crg}} \times \sum_{x=t}^{t+\Omega_i} c_i^x \right) \Big)}{R}. \tag{27}$$

Let WLARS (MPU700) and WLARS (MPU600) denote different simulation environments that the sensors are deployed



(a)



(b)

Fig. 17. Performance comparison of STSQ for the proposed WLARS (MPU) algorithm. (a) Comparison of STSQ without considering the busy levels of MC. (b) Comparison of STSQ with considering the busy levels of MC.

as 700 and 600 for the WLARS (MPU) algorithm, respectively. The speed of MC is varied, ranging from 1 to 8 m/s.

As shown in Fig. 15, the SSQB of the recharged sensors increases with the speed of MC for the WLARS (MPU) algorithm. This is due to the high speed of MC recharges more sensors, leading to a higher SSQB. The SSQB of the recharged sensors for the existing RS-STQ (MPU) is slightly increasing. Besides, the performances of the FLCSD and MCCA are not very stable. It occurs because they did not consider the SSQB of the recharged sensors. On the other hand, the STSQ decreases with the speed of the MC. It is due to the lower speed of MC reducing the number of sensors recharged, leading to a lower STSQ.

Fig. 16 compares the STSQ of WLARS. As shown in Fig. 16, the STSQ increases with the sensor nodes. It is due to the sensor density, which might lead to many sensors sending recharging requests to the MC. Thus, recharge many sensors, leading to a higher STSQ. The STSQ decreases with the sensing range. It occurs because a higher sensing range will contribute more to the spatial quality.

Fig. 17(a) evaluates the STSQ without considering the busy levels of MC, that is to say, the sensor's battery is not constrained with α. Similarly, Fig. 17(b) evaluates the STSQ of the monitoring region with the busy levels of MC, that is, the sensor battery is constrained with α, where sensors send the request according to the busy levels of MC. In Fig. 17(b), the value of α is set at 40 J.

The STSQ in Fig. 17(a) is very unstable compared to Fig. 17(b). It occurs without considering the busy levels of MC. This situation made the MC very busy, where some sensors were not recharged, leading to the unstable STSQ. On the other hand, when the busy levels are considered, the STSQ is very stable. It occurs because the sensor's battery is constrained by the value of α, and the MC receives very few recharging requests according to its busy level. Therefore, the MC can timely recharge all the sensors, leading to a stable STSQ.

VI. Conclusion

This article proposed a WLARS for MC to maximize the STSQ of the network. The three phases of the proposed work are NP, RRD, and ABC. The first phase partitioned the network, aiming to distribute the MCs charging load evenly. The second phase determines the sensor's threshold and considers the busy level of MC. Finally, the third phase considers the adaptive buffer of the MC and the cooperation between the neighboring MCs. The analytical results show that WLARS achieves a higher STSQ than the existing works. The future work of this study is that the BS dispatches the MCs based on the prediction of the arrival rate of the recharging requests of the sensors.

References

[1] M. Khan, J. Seo, and D. Kim, "Modeling of intelligent sensor duty cycling for smart home automation," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 3, pp. 2412–2421, Jul. 2022.

[2] M. A. Mughal, P. Shi, A. Ullah, K. Mahmood, M. Abid, and X. Luo, "Logical tree based secure rekeying management for smart devices groups in IoT enabled WSN," *IEEE Access*, vol. 7, pp. 76699–76711, 2019.

[3] S. Verma, S. Zeadally, S. Kaur, and A. K. Sharma, "Intelligent and secure clustering in wireless sensor network (WSN)-based intelligent transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 13473–13481, Aug. 2022.

[4] J. Chen, C. Yi, R. Wang, K. Zhu, and J. Cai, "Learning aided joint sensor activation and mobile charging vehicle scheduling for energy-efficient WRSN-based industrial IoT," *IEEE Trans. Veh. Technol.*, vol. 72, no. 4, pp. 5064–5078, Apr. 2023.

[5] N. Gharaei, Y. D. Al-Otaibi, S. Rahim, H. J. Alyamani, N. A. K. K. Khani, and S. J. Malebary, "Broker-based nodes recharging scheme for surveillance wireless rechargeable sensor networks," *IEEE Sensors J.*, vol. 21, no. 7, pp. 9242–9249, Apr. 2021.

[6] Q. Qian, A. Y. S. Pandiyan, and D. E. Boyle, "Optimal recharge scheduler for drone-to-sensor wireless power transfer," *IEEE Access*, vol. 9, pp. 59301–59312, 2021.

[7] Y. Kan, C.-Y. Chang, C.-H. Kuo, and D. Sinha Roy, "Coverage and connectivity aware energy charging mechanism using mobile charger for WRSNs," *IEEE Syst. J.*, vol. 16, no. 3, pp. 3993–4004, Sep. 2022.

[8] G. Jiang, S.-K. Lam, Y. Sun, L. Tu, and J. Wu, "Joint charging tour planning and depot positioning for wireless sensor networks using mobile chargers," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 2250–2266, Aug. 2017.

[9] M. Hu, Z. Chen, K. Peng, X. Ma, P. Zhou, and J. Liu, "Periodic charging for wireless sensor networks with multiple portable chargers," *IEEE Access*, vol. 7, pp. 2612–2623, 2019.

[10] Z. W. Lu et al., "Periodic charging planning for a mobile WCE in wireless rechargeable sensor networks based on hybrid PSO and GA algorithm," *Appl. Soft Comput.*, vol. 75, no. 1, pp. 388–403, Feb. 2019.

[11] T. T. Huong, P. Le Nguyen, H. T. T. Binh, K. Nguyenz, N. M. Hai, and L. T. Vinh, "Genetic algorithm-based periodic charging scheme for energy depletion avoidance in WRSNs," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, May 2020, pp. 1–6.

[12] C. Sha, D. Song, and R. Malekian, "A periodic and distributed energy supplement method based on maximum recharging benefit in sensor networks," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2649–2669, Feb. 2021.

[13] Y. Ren, A. Liu, X. Mao, and F. Li, "An intelligent charging scheme maximizing the utility for rechargeable network in smart city," *Pervas. Mobile Comput.*, vol. 77, Oct. 2021, Art. no. 101457.

[14] B. Dande, S.-Y. Chen, H.-C. Keh, S.-J. Yang, and D. S. Roy, "Coverage-aware recharging scheduling using mobile charger in wireless sensor networks," *IEEE Access*, vol. 9, pp. 87318–87331, 2021.

[15] Y. Feng, L. Guo, X. Fu, and N. Liu, "Efficient mobile energy replenishment scheme based on hybrid mode for wireless rechargeable sensor networks," *IEEE Sensors J.*, vol. 19, no. 21, pp. 10131–10143, Nov. 2019.

[16] C. Zhao, H. Zhang, F. Chen, S. Chen, C. Wu, and T. Wang, "Spatiotemporal charging scheduling in wireless rechargeable sensor networks," *Comput. Commun.*, vol. 152, pp. 155–170, Feb. 2020.

[17] K. Wang, L. Wang, M. S. Obaidat, C. Lin, and M. Alam, "Extending network lifetime for wireless rechargeable sensor network systems through partial charge," *IEEE Syst. J.*, vol. 15, no. 1, pp. 1307–1317, Mar. 2021.

[18] C. Lin, Y. Sun, K. Wang, Z. Chen, B. Xu, and G. Wu, "Double warning thresholds for preemptive charging scheduling in wireless rechargeable sensor networks," *Comput. Netw.*, vol. 148, pp. 72–87, Jan. 2019.

[19] G. Han, J. Wu, H. Wang, M. Guizani, J. A. Ansere, and W. Zhang, "A multicharger cooperative energy provision algorithm based on density clustering in the industrial Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 9165–9174, Oct. 2019.

[20] D. Lee, C. Lee, G. Jang, W. Na, and S. Cho, "Energy-efficient directional charging strategy for wireless rechargeable sensor networks," *IEEE Internet Things J.*, vol. 9, no. 19, pp. 19034–19048, Oct. 2022.

[21] G. Han, H. Wang, H. Guan, and M. Guizani, "A mobile charging algorithm based on multicharger cooperation in Internet of Things," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 684–694, Jan. 2021.

[22] A. Tomar, L. Muduli, and P. K. Jana, "A fuzzy logic-based on-demand charging algorithm for wireless rechargeable sensor networks with multiple chargers," *IEEE Trans. Mobile Comput.*, vol. 20, no. 9, pp. 2715–2727, Sep. 2021.

[23] B. Dande, C.-Y. Chang, and S.-J. Wu, "Mobile charger recharge scheduling algorithm based on data quality in wireless sensor networks," in *Proc. IEEE Int. Conf. Consum. Electron.-Taiwan*, Jul. 2022, pp. 37–38.

[24] B. Dande, C.-Y. Chang, and C.-D. Fan, "Mobile charger scheduling algorithm for energy recharging in wireless sensor networks," in *Proc. Int. Conf. Electron. Syst. Intell. Comput. (ICESIC)*, Apr. 2022, pp. 18–22.

[25] B. Dande, C.-Y. Chang, M.-Y. Su, and C.-D. Fan, "On-demand recharge scheduling algorithm in wireless sensor networks," in *Proc. Int. Conf. Electron. Syst. Intell. Comput. (ICESIC)*, Apr. 2022, pp. 13–17.

[26] B. Dande, C.-Y. Chang, C.-H. Kuo, and D. Sinha Roy, "RS-STQ: Recharge scheduling algorithm for maximizing spatial and temporal data accuracy," *IEEE Sensors J.*, vol. 22, no. 23, pp. 23565–23580, Dec. 2022.

[27] A. Tomar and P. K. Jana, "A multi-attribute decision making approach for on-demand charging scheduling in wireless rechargeable sensor networks," *Computing*, vol. 2020, pp. 1677–1701, Nov. 2020.

[28] D. Parker, M. Stojanovic, and C. Yu, "Exploiting temporal and spatial correlation in wireless sensor networks," in *Proc. Asilomar Conf. Signals, Syst. Comput.*, Nov. 2013, pp. 442–446.

[29] C. Lin, J. Zhou, C. Guo, H. Song, G. Wu, and M. S. Obaidat, "TSCA: A temporal–spatial real-time charging scheduling algorithm for on-demand architecture in wireless rechargeable sensor networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 1, pp. 211–224, Jan. 2018.

[30] Y. Deng, C. Han, J. Guo, and L. Sun, "Temporal and spatial nearest neighbor values based missing data imputation in wireless sensor networks," *Sensors*, vol. 21, no. 5, p. 1782, Mar. 2021.

[31] J. A. Shaw, "Radiometry and the Friis transmission equation," *Amer. J. Phys.*, vol. 81, no. 1, pp. 33–37, 2013.

Bhargavi Dande received the M.S. and Ph.D. degrees in computer science and information engineering from Tamkang University, New Taipei, Taiwan, in 2019 and 2022, respectively.

Since 2022, she has been working as a Postdoctoral Researcher with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan. She is currently working on developing decentralized control algorithms for optimal operation of electric vehicles integrated smart grid charging. Her research interests are the Internet of Things, wireless sensor networks, electric vehicle charging, and fog/edge computing.

Chih-Yung Chang (Member, IEEE) received the Ph.D. degree in computer science and information engineering from National Central University, Taoyuan, Taiwan, in 1995.

He is currently a Distinguished Professor with the Department of Computer Science and Information Engineering and the Department of Artificial Intelligence, Tamkang University, New Taipei, Taiwan. His current research interests include artificial intelligence, deep learning and machine learning, the Internet of Things, and wireless sensor networks.

Dr. Chang has served as an Associate Guest Editor for several SCI-indexed journals, including *International Journal of Ad Hoc and Ubiquitous Computing* (IJAHUC) since 2011, *Journal of Applied Science and Engineering* (JASE) since 2018, *International Journal of Distributed Sensor Networks* (IJDSN) from 2012 to 2014, *IET Communications* in 2011, *Telecommunication Systems* (TS) in 2010, *Journal of Information Science and Engineering* (JISE) in 2008, and *Journal of Internet Technology* (JIT) from 2004 to 2008. Dr. Chang is the Co-Chair of ACM SIGMOBILE in Taiwan.

Shih-Jung Wu received the M.S. and Ph.D. degrees from Tamkang University, New Taipei, Taiwan, in 2001 and 2006, respectively.

He is a Professor and the Dean of Student Affairs with Tamkang University. His current research interests include parallel computing, data mining, wireless networks, and artificial intelligence.

Diptendu Sinha Roy received the Ph.D. degree in engineering from the Birla Institute of Technology, Mesra, Ranchi, India, in 2010.

In 2016, he joined the Department of Computer Science and Engineering, National Institute of Technology (NIT) Meghalaya, Shillong, India, as an Associate Professor, where he has been serving as the Chair of the Department of Computer Science and Engineering since January 2017. Prior to his stint with NIT Meghalaya, he had served the Department of Computer Science and Engineering, National Institute of Science and Technology, Berhampur, India. His current research interests include software reliability, distributed and cloud computing, and the IoT, specifically applications of artificial intelligence/machine learning for smart integrated systems.

Dr. Sinha Roy is a member of the IEEE Computer Society.