Int. j. inf. tecnol. (March 2022) 14(2):667–674
https://doi.org/10.1007/s41870-021-00837-1

ORIGINAL RESEARCH

On-demand charging planning for WRSNs based on weighted heuristic method

Naween Kumar[1] · Gandharba Swain[1] · S. Routray[2]

Received: 31 July 2021 / Accepted: 6 November 2021 / Published online: 6 January 2022

Abstract In wireless rechargeable sensor networks, wireless energy transfer technology is considered as a key technique for increasing sensor lifespan. It is widely used to successfully supply electric energy to sensors via mobile charging device(s). However, one of the most difficult problems in such networks is determining an optimal charge-schedule for the mobile charging element to charge the sensors. Sensor residual energies, as well as their spatial-temporal limits and other network factors, all have an effect on the charge-schedule. To address this issue, this paper proposes a novel heuristic charging scheme aimed at maximising sensor lifetimes. A multi-node charger is used in the proposed scheme, which can receive and charge multiple sensors at the same time. The proposed scheme is real-time, dynamic and determines a near-optimal charge-schedule using a weighted heuristic method. As network parameters, the weight function includes requesting sensors' residual energies, contribution-count values, and distances to charging element to quantify the sensors' charging scheduling priorities. Finally, simulation studies demonstrate the charging efficacy of the proposed scheme, demonstrating that it outperforms existing ones in terms of life-success ratio and energy utility ratio.

✉ Naween Kumar
naween@kluniversity.in

Gandharba Swain
gandharba.swain@kluniversity.in

S. Routray
sidheswar69@gmail.com

[1] Vaddeswaram, Guntur, Andhra Pradesh 522502, India

[2] Department of Computer Science and Engineering, School of Engineering, Indrashil University, Rajpur, Mehsana, Gujarat, India

Keywords Wireless rechargeable sensor networks · On-demand charging · Mobile charging element · Near-optimal charge-schedule · Weight function · Heuristic algorithm · Sensor lifetime

1 Introduction

Recently, a wireless sensor network (WSN) is being frequently utilized in military surveillance, disaster prediction [1], health-care monitoring [2], environment monitoring [3] and smart home applications [4], etc [5]. Depending on the application, sensors (*SNs*) collect various types of environmental data from their surroundings. However, because the *SNs* have constrained battery capacities, reliable network operation can be difficult or almost impossible. Consequently, conserving *SNs*' energies is critical for WSNs' long-term operation. When the *SNs*' batteries run out, replacing them may be a simple solution. Nonetheless, this strategy would be too costly and tedious.

Some WSN efforts are aimed at developing energy-efficient solutions to increase *SN* lifetime [6]. Even the most energy-efficient solutions, however, will ultimately deplete the *SNs*' batteries. Another approach is to harvest/charge the *SNs*. However, this is also a challenging issue in WSNs because it has an impact on *SNs* lives, especially when deploying a WSN for long-term monitoring. Due to dynamism, energy harvesting is extremely dependent on external conditions, implying that stable and reliable energy provisioning to *SNs* can't be guaranteed [7].

Wireless energy transfer (WET) has been shown in research studies to provide stable energy to the *SNs* [8]. As a result, it ensures the durability of WSNs based on tight coupling magnetic resonance technology. The authors in [9] first explained the efficacy of WET using strong coupling magnetic resonance. In ref. [10], the authors used WET with a WSN and named it a WRSN. In this network, a mobile charging element (*MCE*) can wirelessly charge *SNs* [11]. The WET has several advantages, including excellent power transmission efficiency, resilience to the outside environment, and no need for direct contact or a line of sight. To replenish energy in WRSNs, two charging methods are typically used: *periodic* [12, 13] and *on-demand* [14]. In the first charging method, the *MCE* provides energy to the *SNs* using a fixed charge-schedule, which may be unsuitable due to charging requirement uncertainties and energy consumption dynamism. On the contrary, in the second charging method, the *MCE* charges the low-energy *SNs* that have made charging requests to it. As a result, it is more advantageous in the WRSN research community [15]. *MCE* can charge *SN* in two modes in both charging methods: single-node charging mode [16] and multi-node charging mode [17]. *MCE* can only charge one node in the first mode, so charging efficiency is low; however, in the latter mode, it can charge multiple nodes simultaneously. It's worth mentioning that the network elements that influence *MCE*'s charging efficacy and network performances must be blended when determining the charge-schedule.

In such instances, charging strategies based on improved heuristic algorithms can be a very effective scheduling method. This is owing to its ease of implementation and ability to achieve the near-optimal solution fast and with less time complexity, making it more ideal for WSNs. Furthermore, it is capable of effectively combining and evaluating diverse network elements.

This paper proposes a charging scheme that employs an improved heuristic approach to find an on-demand charge-schedule. A charge request is sent to *MCE* by a *SN* when its remaining energy level is less than a given charging threshold. The charge-schedule is then determined using a heuristic approach for *MCE* to charge the SNs. The heuristic method uses a weight function for determining scheduling priorities of requesting *SNs*. The weight function includes three network elements as input parameters: residual energy, contribution-count, and distance to *MCE*.

Various studies on on-demand charging schemes with *MCE*(*s*) have been conducted. In comparison to other algorithms, the proposed algorithm's potential features make it more efficient. For dynamic WRSNs, the proposed heuristic scheme finds an efficient charge-schedule. In this scheme, charge-requests from low-energy *SNs* are collected first. Following that, the scheme computes a charge-schedule based on a weight function that considers various network factors. The weight function improves the fairness of scheduling decisions by using real-time network data as input parameters. Furthermore, despite the limited *MCE* battery capacity, the proposed scheme can handle several charge-requests concurrently based on real-time weight values. Consequently, scalability of the proposed scheme is increased. In this work, the *MCE* is having multi-node charging feature [18, 19], which minimizes on-demand charging delay. The following are the primary contributions.

- Propose a weighted heuristic scheme GOCS for determining a near-optimal preemptive charge-schedule to extend the lifetime of *SNs*.
- The *GOCS* scheme assigns requesting *SNs*' weights based on their residual energies, contribution-count values, and Euclidean distances from the *MCE*, and preferentially selects the requesting *SNs* that have the least weight to provide charging service.
- Simulation tests comparing the proposed GOCS scheme to existing ones in terms of two network performance metrics: life-success ratio and energy utility ratio.

The rest of work is laid out as follows: The literature study is surveyed in Sect. 2. The system model with preliminaries and proposed problem definition are defined in Sect. 3. Section 4 provides the proposed scheme's description. In Sect. 5 simulation study is presented to compare the proposed scheme to baseline schemes. The proposed work is summarized in Sect. 6.

2 Literature survey

In this section, some of the most important on-demand charging algorithms pertinent to this paper, are surveyed. In [14, 20], The *MCE* serves charge requests in compliance with FCFS policy. However, to charge requesting *SNs*, the *MCE* may necessitate excessive back-and-forth motions, leading to a longer charging delay. To address this issue, the study [21] included NJNP approach to find a charge schedule based on a geographical limit. In NJNP, a charge-schedule is designed based on the distance between *MCE* and the requesting *SNs*, i.e. *MCE* serves the *SN* that is closest to it first. However, if *MCE* receives a new charge-request from a nearby *SN* while charging, the *MCE* is pre-empted. However, due to the ad-hoc nature of *SN* deployment, requests can come from any *SN*, resulting in the premature death of many low-energy *SNs* that are far away from the *MCE*. This phenomenon causes a lower life-success ratio. In ref. [22–24], a charge-schedule is designed by proposing reinforcement learning-based charging method that uses only one network attribute, *SN* remaining energy.

Addressing this issue, in ref. [25], the *SNs*' charge-requests are served based on spatial-temporal constraints. in ref. [26], the *SNs*' charge-requests are served based on the distance between *MCE* and the requesting *SNs* and their variable energy consumption rates. Using the smallest enclosing disks concept, an approximation approach is introduced in ref. [27] to identify optimal sojourn locations for minimizing charging delay. In ref. [15], the authors aimed at optimizing covering utility by scheduling several *MCEs*. The *MCE* may charge a requesting *SN* while traveling. Their proposed algorithms were unable to significantly increase the network's life-success ratio. But, in prior studies [15, 25], the *MCE* did not have a multi-node charging feature and could not evaluate numerous network factors to create the schedule, which is something that must be overlooked if charging efficiency is to be significantly increased. In ref. [28], the authors also employed a multi-node *MCE*. Their goals were to shorten the *MCE*'s tour-length and create a charge-schedule based on SNs' charging utility gains. The amount of energy acquired by *SNs* from the *MCE* determines their charging utility gains. They did not, however, make use of numerous network properties for scheduling purposes (Table 1).

The aforementioned issues highlight the necessity to dig deeper into the charging problems and develop a charging scheme having the following features like (1) utilizing a multi-node *MCE* for simultaneous charging of SNs, (2) allowing *MCE* to preempt if a new charge-request with a higher weightage is received, and (3) combining various network features for making decisions. The scope of this work is to improve the life-success ratio as well as the energy utility ratio.

3 Network model and problem formulation

A WRSN has n rechargeable *SNs* in $S = \{S_1, S_2, \cdots, S_n\}$, a base-station *BS*, a mobile charging element *MCE* with traveling speed V_x and battery capacity BC_{me}. For the *MCE*, the *BS* functions as a charging station. At initially, the *MCE* is completely charged and located at the *BS*. The *SNs* with rechargeable battery capacity BC_{max} are dispersed at random across the network. Their energies are depleted when they are receiving and sending data, or in listening or sleeping states. Each sensor S_i's average energy-consuming rate is denoted by $E_{con}(S_i)$. Furthermore, when moving across the network or charging the *SNs*, the *MCE* loses energy. The *MCE*'s charging range is denoted by C_r.

According to the proposed model, each sensor says S_i continuously monitors its residual energy $RE(S_i)$ and sends a request to *MCE* when $RE(S_i) < E_{th}$, where E_{th} denotes a energy threshold limit. The charge-request for a *SN* S_i includes its location $LOC(S_i)$, residual energy $RE(s_i)$, and a time-stamp value $TS(S_i)$. It can be denoted as $CH(s_i) = \{LOC(S_i), RE(S_i), TS(S_i)\}$. The *MCE* stores such requests in a queue called CHQ_{req}. Then, to provide services, a charge-schedule CH_{sch} is determined. The *MCE* can go to charge a requesting *SN* if it has adequate energy before returning to the *BS*. Otherwise, the *MCE* recharges itself by returning to the *BS*. Following that, process the remaining *SNs* for charging. The *MCE* recharge time is presumed to be insignificant. Through multi-node charging,

Table 1 Summary of on-demand charging scheduling schemes

ID:year	Network-attributes	Charging principle	Goals
[14]:2018	Temporal	First-Come-First-Serve	Reducing delay
[15]:2014	Point-of-interests, sensor positions	Peicewise covering utility	Maximizing coverage
[16]:2017	Temporal-spatial	Global-path-optimization	Network lifetime
[17]:2017	Residual energy, neighboring sensors, distance to MC	Ordering of sojourn locations with bounded delay	Increasing survival ratio, Energy efficiency utilization
[20]:2010	Temporal	First-Come-First-Serve	Reducing delay
[21]:2014	Spatial	Nearest-Job-Next-with Preemption	Throughput, charging latency
[27]:2015	Sensor positions, charging power vector	Ordering of sojourn locations	Reducing charging delay
[25]:2018	Temporal-spatial	Gravitational search	Reducing delay
[28]:2018	Charging energy obtained	Charging utility gain	Maximizing charging utility
[22]:2020	Residual ebergy	Q (Reinforcement) learning	Maximizing target coverage
[23]:2021	Residual ebergy	Fuzzy Q learning	Increasing survival ratio
[24]:2021	Residual ebergy	Reinforcement learning	Increasing charging effectiveness
[26]:2018	Temporal and energy consumption rate	Weight factor	Increasing charging effectiveness

MCE may charge several $SNs \in CHQ_{req}$ at the same time [28]. In comparison to *MCE*'s traversing time, its response time is considered negligible. At all times, the *MCE* is fully well aware of *SNs*' residual energy levels. The actual charging rate for S_i is $ACH_{rate}(S_i) = CH_{rate} \times \eta$, where CH_{rate} is charging power output of *MCE*. The term η ranges from 0 to 1. Since, in this work, $C_r \approx 2.7m$, η is set to 0.68 [29]. The Lifetime of SN S_i $(lf_{time}(S_i))$ is the time required for consuming the residual energy $RE(S_i)$ of *SN* S_i, defined as $lf_{time}(S_i) = \frac{RE(S_i)}{E_{con}(S_i)}$. Furthermore, the service time of S_i $(T_c(S_i))$ is the time spent by *MCE* to travel and completely charge S_i, expressed as, $T_c(S_i) = \frac{D(S_i,me)}{V_x} + \frac{E_{max}-(RE(S_i)-\frac{D(S_i,me)}{V_x}\times E_{con}(S_i))}{ACH_{rate}(S_i)-E_{con}(S_i)}$. Here, $\frac{D(S_i,mc)}{V_x}$ is *MCE*'s travel time to reach S_i, $(RE(S_i) - \frac{D(rs_i,me)}{V_x} \times E_{con}(S_i))$ is the updated residual energy of SN S_i when the *MCE* reaches at SN S_i. The required full charging time for the SN S_i is as follows. $\frac{E_{max}-(RE(S_i)-\frac{D(S_i,me)}{V_x}\times E_{con}(S_i))}{(ACH_{rate}(S_i)-E_{con}(S_i))}$A *SN* S_i is considered to be charged successfully if $T_c(S_i) \leq lf_{time}(S_i)$, else it is not. In both circumstances, *MCE* would charge S_i.

In this work, the states of a SN $(S_i))$ is classified into three categories : critical $(S_i(crt))$, normal $(S_i(nrm))$, and dead $(S_i(dead))$. If a *SN* S_i requests a charge from *MCE*, it enters its critical state and expressed as $S_i(crt) = 1$; otherwise $S_i(crt) = 0$. If service time $T_c(S_i)$ of a SN S_i is less than or equal to its lifetime $lf_{time}(S_i)$, it enters its normal state, expressed as $S_i(nrm) = 1$; otherwise $S_i(nrm) = 0$. Furthermore, the dead state of SN S_i is expressed as $S_i(nrm) = 1$ if $RE(S_i) = 0$; otherwise $S_i(nrm) = 1$.

The performance parameters for evaluating the proposed work are the Life-Success ratio (LR_{ratio}) and the Energy Utility Ratio (EUE_{ratio}). The life-success ratio (LR_{ratio}) is a ratio of the total number of *SNs* successfully charged by the *MCE* to the total number of requesting *SNs*, expressed as $LR_{ratio} = \frac{\sum_{i=1}^{n} S_i(nrm).S_i(crt)}{\sum_{i=1}^{n} S_i(ctr)}$. The energy utility ratio (EUE_{ratio}) is a ratio of total energy obtained by *SNs* from *MCE* to total energy depleted by the *MCE* for traversing and charging during time *T*, and is defined as, $EUE_{eff} = \frac{\sum_{i=1}^{n} E_{max}-RE(S_i)+T_c(S_i)\times E_{con}(S_i)}{E(T)} \times S_i(crt)$where $E(T)$ denotes energy depleted by *MCE* within time *T*. For clear understanding, consider the following WRSN example: a charge-request queue $CH_{req} = \{S_1, S_2, S_3, \ \ldots, S_5\}$. Let $vp_1 = LOC(S_1), vp_2 = LOC(S_2)$, and $vp_3 = LOC(S_3)$ are used to visit the *SNs* S_1, S_2, and S_3 respectively. The visiting point $vp_4 = LOC(S_4)$ is used to visit S_4 and S_5 at a time. Let *MCE* initially have BC_{me} = 20 *J* and its initial location is at *BS*. Let the schedule be $CH_{sch} = BS \rightarrow S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4, S_5 \rightarrow BS$.

Figure 1 depicts one such example, in which S_1, S_2,S_4, and *S*5 are classified as normal, while *S*3 are classified as dead. Even though it is marked as dead, the *MCE* will charge it. This figure contains five tables, one for each $SN \in CH_{req}$. Each table has the following information: the *SN*'s amount of energy obtained, the *MCE*'s traversing time to go from the previous visiting location to a targeted visiting location, and the *SN*'s charging time. The total energy obtained by the *SNs* from *MCE* is (26 *J* + 24 *J* + 14.6 *J* + 22.4 *J* + 11.2 *J*) = 98.2 *J*. Thus, the energy utility ratio $EU_{eff} = \frac{17.4}{100}$ = 0.982. Also, as only four *SNs* are charged successfully among 5 *SNs*, the life-success ratio is LR_{ratio}= 0.8. The notations used throughout the proposed work is listed in Table 2.

3.1 Problem definition

A set of charge-requests denoted by $CHQ_{req} = \{CR(S_i), \ldots, CR(S_m)\}$ is received by *MCE*, the goal is to discover a charge-schedule CH_{sch} for the *MCE* to improve the network life. To put it another way, a charging algorithm must be used to improve the network's life-success ratio and energy utility ratio in order to extend its life. This is known as *Computing charge-schedule for enhancing sensor-lifetime (CSSL)* problem.

4 Proposed solution

The proposed heuristic-based charging solution called *G*enerated-on-demand charge-scheduling with pre-emption (or *G*OCS), is detailed here to solve the proposed *C*SSL problem. The *MCE*, in this algorithm, buffers charging-request messages from low-energy *SNs*. It prioritizes them by using a weight function. Specifically, each requesting *SN* is assigned a weight based on its residual energy, contribution-count value, and Euclidean distance from the *MCE*'s current position. The contribution-count of a requesting *SN* is based on its neighbors which have also made requests to the *MCE*. Then, the proposed

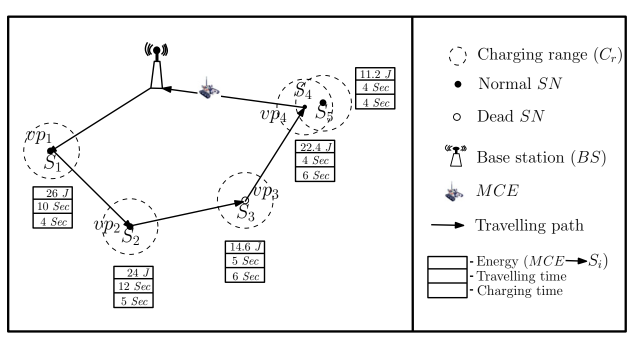


Fig. 1 An example of charge-schedule

Table 2 Summary of notations

Notation	Description
S	Collection of SNs, where $\lvert S\rvert = n$
C_r	*MCE*'s charging range
BC_{max}	SNs' battery energy limit
BC_{me}	*MCE*'s battery energy limit
V_x	*MCE*'s maximum speed
E_{th}	Energy threshold to make a charge request for SNs
CHQ_{req}	A queue for buffering SNs' charge requests.
CH_{sch}	A schedule for *MCE* to charge SNs as output.
CH_{rate}	*MCE*'s energy charging rate
$CR(S_i)$	A charge request of S_i to the *MCE*
$ACH_{rate}(S_i)$	Actual charging rate for SN S_i
$RE(S_i)$	Remaining energy of a SN S_i
$E_{con}(S_i)$	Average energy depletion rate of SN S_i
$lf_{time}(S_i)$	$lf_{time}(S_i) = \frac{RE(S_i)}{E_{con}(S_i)}$
$T_c(S_i)$	Total service time required for *MCE* to charge S_i
LR_{ratio}	Life-success ratio
EUE_{ratio}	Energy utility ratio
$N(S_i)$	Neighbor set of SN S_i
$RNS(S_i)$	requesting neighbor set of a SN S_i
$CC(S_i)$	Contribution-count of a SN S_i
$W(S_i)$	Weight value of SN S_i

scheme aims to find an efficient charge-schedule (CH_{sch}) as an output to optimize the network's life-success ratio (LR_{ratio}) and energy utility ratio (EUE_{eff}). The following are a few more terminology used, followed by the proposed scheme's detailed description.

Definition 1 (Charging-overlapped *SNs*): If the Euclidean distance between two *SNs* S_i and S_j is less than C_r, they are said to be charging-overlapped. Furthermore, the set of *SNs* that are charging-overlapped with S_i is called as its neighbor set $(N(S_i))$.

Definition 2 (Requesting-neighbor set of sensor S_i $(RNS(S_i))$): The requesting-neighbor set of S_i is the set of requesting $SNs \in CHQ_{req}$ that are charging-overlapped with the sensor S_i. Its contribution-count is the size of its requesting-neighbor set i.e., $CC(S_i) = \lvert RNS(S_i)\rvert$.

In Fig. 2, the dotted circles denotes charging regions of the *MCE* for the *SNs* in $S = \{S_1, S_2, \ldots, S_7\}$. In this Fig., S_1 is charging-overlapped with S_2, S_2 is charging-overlapped with both S_1 and S_3, S_5 is charging-overlapped with S_6, and S_4's charging region is isolated from others. The Neighbor sets of $SNs \in S$ are $N(S_1) = \{S_2\}, N(S_2) = \{S_1, S_3\}, N(S_3) = \{S_2\}, N(S_4) = \{\emptyset\}, N(S_5) = \{S_6\}$, and $N(S_6) = \{S_5\}$.

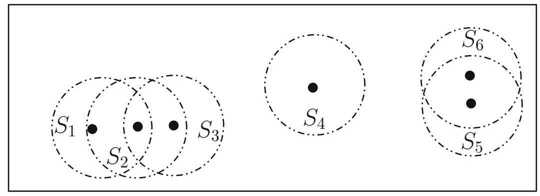


Fig. 2 An example of *MCE*'s charging range for the *SNs*

Let the *SNs* S_2, S_3, S_4 and S_6 have low-energy and send request for charging to the *MCE* i.e. $CHQ_{req} = \{S_2, S_3, S_4, S_6\}$. In this case, the requesting-neighbor sets of the requesting sensors $SNs \in CHQ_{req}$ are $RNS(S_2) = \{S_3\}, RNS(S_3) = \{S_2\}, RNS(S_4) = \{\emptyset\}$, and $RNS(S_6) = \{\emptyset\}$.

After storing charge-requests from low-energy *SNs* in a queue CHQ_{req}, the *MCE* first computes a contribution-count $CC(S_i)$ for each requesting sensor $S_i \in CHQ_{req}$. Based on the information obtained from the charging-request message, it later updates the remaining information like remaining energy $(RE(S_i))$ and distance to *MCE* ($D(me, S_i)$: distance from the *MCE*'s current position to the SN S_i) for all requesting $S_i \in CHQ_{req}$. The *MCE* uses a weighted scheduling function to compute a charge-schedule (CH_{sch}). The *MCE* roams the network and provides service to the requesting *SNs*. If new charge-requests arise in the meanwhile, they are likewise stored in the same queue CHQ_{req} and would be served under the same charging-schedule CH_{sch}. The *MCE* switch to an incoming-requesting *SN* if the new requesting *SN* has a minimum weight value among all values of $SNs \in CHQ_{req}$. Before providing service to a requesting sensor S_i, the *MCE* checks to see if its residual energy is adequate to reach *BS* after charging it. Otherwise, if possible, it charges at the nearest requesting $SN \in CHQ_{req}$ and immediately visits the *BS* for energy replenishment. The following sub-section discusses the heuristic weighted scheduling function.

4.1 Weighted scheduling function

Here, we explain how the GOCS scheme computes a weight value based on the contribution-count $CC(S_i)$, residual energy $RE(S_i)$, and distance to *MCE* $D(me, S_i)$ corresponding to each requesting $SN\ Si \in CHQ_{req}$. The GOCS prioritizes requesting *SNs* with the minimum weight as the first sensor to be charged by the *MCE*. Let $W(S_i)$ denotes a weight value of a *SN* S_i, then its weight is computed as:

$$W(S_i) = \frac{D(me, S_i) * RE(S_i)}{1 + CC(S_i)}. \quad (1)$$

According to the weight function, requesting $SNs \in CHQ_{req}$ with less residual energy, smaller distance from

MCE, and larger contribution-count receives the minimum weight. As a result, requesting $SNs \in CHQ_{req}$ that are nearby to the *MCE*, have multiple requesting-neighboring *SNs*, and have the least residual energy have a better chance of obtaining energy from the *MCE*. The amount of energy spent by *MCE* is proportional to the Euclidean distance between the *MCE* and a requesting *SN*, as well as the number of requesting neighbors and the amount of obtained energy from the *MCE*. As a result, visiting the requested *SN* with minimum weight reduces the *MCE*'s travel distance as well as the life-success ratio, thereby minimizing energy consumption. A special case occurs when multiple *SNs* have the same weight. In this case, the remaining energy of the *SNs* with the same weight value must be compared, and the *SN* with the least remaining energy must be chosen as the next charging node.

It is worth noting that each charging schedule is selected only when the requesting *SN* and its neighbours (if any) are charged, or when a new charge request is made. There is only one charging request selected at each charging schedule. Because all network attributes, such as the distance between other *SNs* and the *MCE*, as well as the energy consumption rate and neighbouring *SNs*, may change. As a result, at the start of the next charging schedule, the previous order of the *SNs* should be updated. As previously stated, the estimation of remaining energy, distance to *MCE*, and neighbouring *SNs* is time-dependent, and when a charging request is served after determining the charge schedule, all network attributes may be changed, implying that all network attributes are subject to real-time change. As a result, the proposed GOCS scheme is both real-time and dynamic. The Algorithm 1 summaries how the GOCS scheme works.

Algorithm 1: GOCS Algorithm

Input: $S = \{S_1, S_2, \ldots, S_n\}$: A set of sensors ; C_r: Charging range; BC_{max}: Battery capacity of SNs; $\{RE(S_1), RE(S_2), \ldots, RE(S_n)\}$: residual energy of sensors; V_x: Traveling speed of MCE; BC_{me}: Battery capacity of MCE; $\{E_{con}(S_1), E_{con}(S_2), \ldots, E_{con}(S_n)\}$: Average energy depletion rate of sensors; $CHQreq$: A queue for buffering charge-requests.

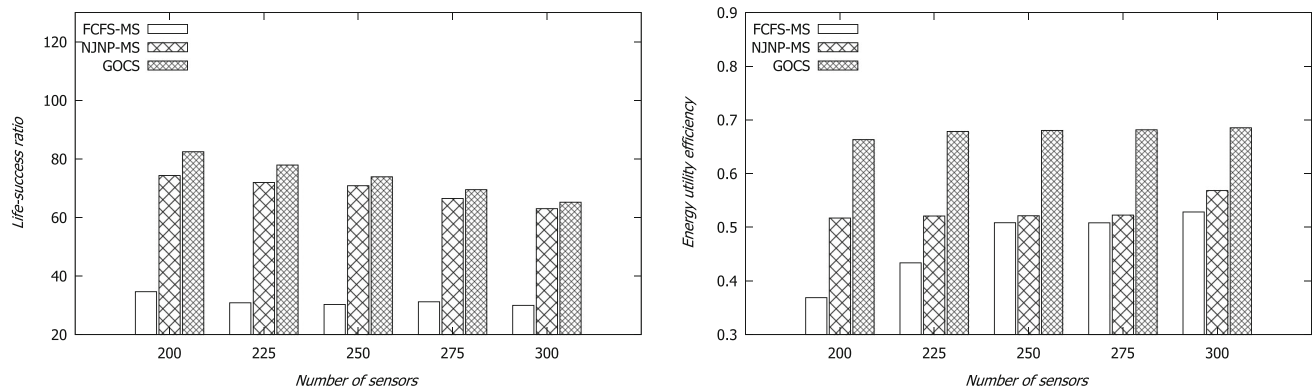
Output: CH_{sch}: a charge-schedule.

Low-energy SNs send charge-request messages to MCE;
The MCE stores information $[RE(S_i), D(me, rs_i), CC(S_i)]$ $\forall S_i \in CHQ_{req}$;
while $CHQ_{req} \neq \emptyset$ **do**
  Computing weights of the requesting sensors in CHQ_{req} using Eq. no. 1;
  Select the SN S_i with minimum weight value $W(S_i)$;
  $CH_{sch} \leftarrow CH_{sch} \cup S_i$ /* Include S_i in output schedule CH_{sch} */
  $CHQ_{req} \leftarrow CHQ_{req} \setminus S_i$; // Remove S_i from the queue CHQ_{req}
  If new charge-requests arise in meanwhile, they are stored by MCE into CHQ_{req};
  Update the information $[RE(S_j), D(me, S_j), CC(S_j)]$ of all remaining sensors in CHQ_{req};
return CH_{sch}

5 Performance evaluation

The proposed GOCS scheme's performance is reported in terms of life-success ratio (LR_{ratio}), and energy utility ratio (EUE_{ratio}) metrics. Simulated experiments in MATLAB are used to verify the results. The GOCS scheme is compared to two existing on-demand charging schemes: FCFS scheme [14], which schedules requesting *SNs* based on the arrival of charge requests, and the NJNP-MS scheme [21], which schedules requesting *SNs*' based on the spatial and temporal limits. Whereas, in GOCS scheme, the charge-schedule is determined by multiple factors such as spatial, temporal, and residual energy. To reflect the fact that *MCE* now has multi-node charging capability, we renamed the FCFS scheme to FCFS-MS. A WRSN with 200–300 sensors is considered for simulation set-up. The *SNs* are randomly distributed in a 100 × 100 m^2 region. The *BS* is located in the middle of the deployment region [21]. Each *SN* S_i has a battery capacity BC_{max} = 10.8 *kJ* [30], and a charging range C_r = 2.7 *m* [29]. Each *SN* S_i has remaining energy [0.1–10.8] *KJ* and the average rate of energy depletion [0.0001–0.05] *J/Sec*. The *MCE*'s battery capacity is BC_{me} = 4000 *kJ*, an its energy-charging rate is CH_{rate} = 5 *J/Sec*. The charging efficiency is set to η = 0.68 [29]. The *MCE* looses energy at the rate of 600 *J/m* [28] while travelling at speed 5 *m/Sec*. The simulation outcomes are averaged across 30 distinct random sensor-deployments.

The proposed GOCS scheme's performance is measured in terms of life-success ratio and energy utility ratio. The impact of the number of *SNs* on performance metrics is discussed in the subsequent sub-sections. The Life-success ratio (LR_{ratio}) is calculated based on the number of *SNs*. Figure 3(a) reports that, increasing the number of *SNs* reduces the LR_{ratio}. This is because as the number of *SNs* increased, so does the charge-request list's size $|CHQ_{req}|$, putting a higher load on *MCE* to deliver charging services



(a) Life-success ratio

(b) Energy utility ratio

Fig. 3 Performance of GOCS algorithm by varying number of *SNs*

to the requesting *SNs*. More *SNs* are likely to die as a result, reducing the LR_{ratio} over time. Furthermore, the GOCS scheme outperforms the others in terms of LR_{ratio}. This is since GOCS combines multiple network variables, leading to an increased number of requested *SNs* being charged. The FCFS-MS scheme, unlike the GCPS scheme, prioritizes recharging the requested *SNs* based solely on the time constraint. Due to a lack of energy, distance, and time constraints in charging-scheduling decisions, the requesting *SNs* may be more likely to die. Furthermore, hardly any of the existing schemes employ all three factors in association with the pre-emption capability.

Here, various charging methods' energy utility ratios (EUE_{ratio}) are compared by varying numbers of *SNs*. EUE_{ratio} increases as the number of *SNs* increases, as in Fig. 3(b). It's because of the increase in $|CHQ_{req}|$. As a result, a greater number of *SNs* will be able to charge, resulting in an increase of the amount of total energy obtained by *SNs*, increasing EUE_{ratio}'s value. It's also seen from Fig. 3(b) that GOCS algorithm succeeds in achieving the highest EUE_{ratio}. It is because GOCS algorithm employs multiple network variables with pre-emption to create an effective schedule, reducing the waiting time of requesting *SNs*.

6 Conclusion

In this work, a novel charging method termed GOCS is devised for on-demand charging in WRSNs using a mobile charging element (*MCE*). The GOCS scheme determines an efficient charge schedule for requesting *SNs* in dynamic WSNs. It creates a charge-schedule considering various network factors of incoming requests and then serves each one individually. It enables the *MCE* to serve a new requesting *SN* if that requesting *SN* has a minimum weight value. To quantify the *SNs*' scheduling priority, the weight function includes residual energy, contribution-count, and distance to *MCE*. Using experimental simulation, the GOCS's charging performances are evaluated concerning life-success ratio and energy utility ratio. The results report that the GOCS outperforms other existing schemes. However, in large-scale WRSNs, a single *MCE* is insufficient for charging the *SNs*. The use of multiple *MCE*s and their efficient coordination among themselves for recharging the *SNs* as well as themselves may enhance overall charging performance even more, particularly in large-scale WRSNs. So, in the future, we will strive to integrate this feature into our work by investigating a new weighted multi-attribute based heuristic method.

References

1. Furquim G, Jalali R, Pessin G, Pazzi RW, Ueyama J et al (2018) How to improve fault tolerance in disaster predictions: a case study about flash floods using iot, ml and real data. Sensors 18(3):907
2. Adame T, Bel A, Carreras A, Melia-Segui J, Oliver M, Pous R (2018) Cuidats: an rfid-wsn hybrid monitoring system for smart health care environments. Futur Gener Comput Syst 78:602–615
3. Verma S, Sood N, Sharma AK (2018) Design of a novel routing architecture for harsh environment monitoring in heterogeneous wsn. IET Wirel Sens Syst 8(6):284–294
4. Alaiad A, Zhou L (2017) Patients' adoption of wsn-based smart home healthcare systems: an integrated model of facilitators and barriers. IEEE Trans Prof Commun 60(1):4–23
5. Naga B, Sushmitha JR (2019) Analysis of wireless sensor application and architecture. *IJRTE* 7
6. Kumar N, Dash D (2020) Flow based efficient data gathering in wireless sensor network using path-constrained mobile sink. J Ambient Intell Humaniz Comput 11(3):1163–1175
7. Kumar N, Dash D (2021) Improving data-throughput in energy harvesting wireless sensor networks using a data mule-recent advances in computer science and communications. Recent Adv Comput Sci Commun 14:1213–1224

8. Tian M, Jiao W, Liu J, Ma S (2019) A charging algorithm for the wireless rechargeable sensor network with imperfect charging channel and finite energy storage. Sensors 19(18):3887
9. Kurs A, Karalis A, Moffatt R, Joannopoulos JD, Fisher P, Soljačić M (2007) Wireless power transfer via strongly coupled magnetic resonances. Science 317(5834):83–86
10. Han L, Li L (2017) Integrated wireless communications and wireless power transfer: an overview. Phys Commun 25:555–563
11. Nikoletseas S, Raptis T, Souroulagkas A, Tsolovos D (2017) Wireless power transfer protocols in sensor networks: experiments and simulations. J Sens Actuator Netw 6(2):4
12. Lyu Z, Wei Z, Pan J, Chen H, Xia C, Han J, Shi L (2019) Periodic charging planning for a mobile WCE in wireless rechargeable sensor networks based on hybrid PSO and GA algorithm. Appl Soft Comput 75:388–403
13. Huong TT, Le Nguyen P, Binh HTT, Nguyenz K, Hai NM et al. (2020) Genetic algorithm-based periodic charging scheme for energy depletion avoidance in WRSNS. In: 2020 IEEE wireless communications and networking conference (WCNC), IEEE, p. 1–6
14. He L, Kong L, Tao J, Xu J, Pan J (2018) On-demand mobile data collection in cyber-physical systems. Wirel Commun Mob Comput. https://doi.org/10.1155/2018/5913981
15. Jiang L, Xiaobing W, Chen G, Li Y (2014) Effective on-demand mobile charger scheduling for maximizing coverage in wireless rechargeable sensor networks. Mob Netw Appl 19(4):543–551
16. Lin C, Zhou J, Guo C, Song H, Wu G, Obaidat MS (2017) Tsca: a temporal-spatial real-time charging scheduling algorithm for on-demand architecture in wireless rechargeable sensor networks. IEEE Trans Mob Comput 17(1):211–224
17. Khelladi L, Djenouri D, Rossi M, Badache N (2017) Efficient on-demand multi-node charging techniques for wireless sensor networks. Comput Commun 101:44–56
18. Kurs A, Moffatt R, Soljačić M (2010) Simultaneous mid-range power transfer to multiple devices. Appl Phys Lett 96(4):044102
19. CP Ricci (2019)Multi-mode rechargeable electric vehicle, US Patent 10,183,584
20. L He, Y Zhuang, J Pan, J Xu (2010) Evaluating on-demand data collection with mobile elements in wireless sensor networks. In IEEE 72nd vehicular technology conference-fall, Ottawa, ON, Canada, p. 1–5
21. He L, Kong L, Gu Y, Pan J, Zhu T (2014) Evaluating the on-demand mobile charging in wireless sensor networks. IEEE Trans Mob Comput 14(9):1861–1875
22. Le Nguyen P, Nguyen T-H, Nguyen K et al.(2020) Q-learning-based, optimized on-demand charging algorithm in WRSN. In: 19th international symposium on network computing and applications (NCA), IEEE, p. 1–8
23. Le Nguyen P, La VQ, Nguyen AD, Nguyen TH, Nguyen K et al (2021) An on-demand charging for connected target coverage in WRSNS using fuzzy logic and q-learning. Sensors 21(16):5520
24. Liu T, Baijun W, Wenzheng X, Cao X, Peng J, Hongyi W (2021) RLC: a reinforcement learning-based charging algorithm for mobile devices. ACM Trans Sensor Netw (TOSN) 17(4):1–23
25. Kaswan A, Tomar A, Jana PK (2018) An efficient scheduling scheme for mobile charger in on-demand wireless rechargeable sensor networks. J Netw Comput Appl 114:123–134
26. Zhong P, Zhang Y, Ma S, Kui X, Gao J (2018) RCSS: a real-time on-demand charging scheduling scheme for wireless rechargeable sensor networks. Sensors 18(5):1601
27. Lingkun F, Cheng P, Chen J, He T (2015) Optimal charging in wireless rechargeable sensor networks. IEEE Trans Veh Technol 65(1):278–291
28. Ma Y, Liang W, Wenzheng X (2018) Charging utility maximization in wireless rechargeable sensor networks by charging multiple sensors simultaneously. IEEE/ACM Trans Netw 26(4):1591–1604
29. Tomar A, Muduli L, Jana PK (2019) An efficient scheduling scheme for on-demand mobile charging in wireless rechargeable sensor networks. Pervasive Mob Comput 59:101074
30. Liang W, Wenzheng X, Ren X, Jia X, Lin X (2016) Maintaining large-scale rechargeable sensor networks perpetually via multiple mobile charging vehicles. ACM Trans Sens Netw (TOSN) 12(2):1–26