A Fuzzy Logic-Based On-Demand Charging Algorithm for Wireless Rechargeable Sensor Networks With Multiple Chargers

Abhinav Tomar[✉], Member, IEEE, Lalatendu Muduli[✉], and Prasanta K. Jana[✉], Senior Member, IEEE

Abstract—Mobile chargers have greatly promoted the wireless rechargeable sensor networks (WRSNs). While most recent works have focused on recharging the WRSNs in an on-demand fashion, little attention has been paid on joint consideration of multiple mobile chargers (MCs) and multi-node energy transfer for determining the charging schedule of energy-hungry nodes. Moreover, most of the schemes leave out the contemplation of multiple network attributes while making scheduling decisions and even they overlook the issue of ill-timed charging response to the nodes with uneven energy consumption rates. In this paper, we address the aforesaid issues together and propose a novel scheduling scheme for on-demand charging in WRSNs. We first present an efficient network partitioning method for distributing the MCs so as to evenly balance their workload. We next adopt the fuzzy logic which blends various network attributes for determining the charging schedule of the MCs. We also formulate an expression to determine the charging threshold for the nodes that vary depending on their energy consumption rate. Extensive simulations are conducted to demonstrate the effectiveness and competitiveness of our scheme. The comparison results reveal that the proposed scheme improves charging performance compared to the state-of-the-art schemes with respect to various performance metrics.

Index Terms—Fuzzy logic, wireless rechargeable sensor networks, mobile charger, on-demand, charging schedule

1 INTRODUCTION

WIRELESS sensor networks (WSNs) have gained substantial research interest due to wide range of applications, including precision agriculture, habitat and environmental monitoring, IoT-based smart cities, industrial manufacturing, and so on [1]. However, energy constraint is the most critical issue as sensor nodes are small battery-powered that restricts their perpetual operation. Therefore, many energy saving [2], [3], [4] and energy harvesting schemes [5], [6] have been proposed. While the former schemes can only increase the operation time of a WSN to some extent, the latter ones are not always feasible as they supply energy from unpredictable environmental sources. The recent wireless energy transfer (WET) technology [7] and rechargeable lithium battery [8] have become very reliable to provide energy supply to the sensor nodes in a controllable and sustainable manner. As a result, wireless rechargeable sensor networks (WRSNs) have gained enormous research attention. In a WRSN, nodes are recharged by employing one or more mobile chargers (MCs) before their energy depletion and thus network can ideally work forever.

The MCs in a WRSN usually follow two basic schemes, periodic charging [9], [10] and on-demand charging [11], [12], [13]. In periodic charging, schedule of an MC is known

a priori and the MC roams accordingly around the network in a periodic fashion. However, due to heterogeneous and dynamic energy consumption rate of the nodes, the fixed charging schedule is not optimal. This leads to high node failure rate and diminishes the charging performance. On the contrary, an MC in on-demand charging can take real time charging decision to fulfill energy demand of the sensor nodes. This makes the on-demand scheme susceptible to deal with high uncertainty in energy consumption of the nodes. However, taking into account the battery deadlines of requesting nodes, determining their efficient charging schedule is very challenging and thus it is the most emphasized issue in the current research of WRSNs. Many on-demand schemes have been developed till date. However, most of them [11], [12], [13], [14], [15], [16] focus on the use of a single MC to recharge only one sensor node at a time. This can not be overlooked because a single MC with finite power supply is not enough to recharge all the nodes, especially in dense WRSNs. At the same time, multi-node charging [17] of the nodes reduces waiting time incurred per node, thereby improving charging efficiency. To this end, this paper proposes an on-demand multi-node charging scheme with multiple MCs to address the following fundamental issues: 1) How to distribute the multiple MCs in the network without overlapping their charging regions so that their charging workload are evenly balanced? 2) How to ensure that a sensor node will not be recharged by more than one MC at any time? 3) How to make wise and efficient scheduling decisions by blending multiple network attributes, such as residual energy (RE), distance to MC (D), energy consumption rate (ECR), and critical node density

• The authors are with the Department of Computer Science & Engineering, Indian Institute of Technology (ISM), Dhanbad, Jharkhand 826004, India. E-mail: {profession.abhinav, lalatendu.bapi123}@gmail.com, prasantajana@yahoo.com.

Manuscript received 2 Sept. 2019; revised 23 Mar. 2020; accepted 21 Apr. 2020. Date of publication 0 . 0000; date of current version 0 . 0000.

(Corresponding author: Abhinav Tomar.)

Digital Object Identifier no. 10.1109/TMC.2020.2990419

Fig. 1. System architecture showing an instance of charging scheduling problem: (a) Multi-node charging with multiple MCs. (b) Scheduling decision based on multiple network attributes.

(CN)? A system architecture addressing all such issues is shown in Fig. 1. This can be noted from Fig. 1a that MC_1 , MC_2 , and MC_3 receive charging requests from 5, 3, and 2 nodes, respectively. This makes their charging load uneven which leads to poor energy usage efficiency. This can also be noted that the nodes S_i and S_j are within the charging range of each other. As a result, both can be recharged by either of both MC_1 and MC_2 through multi-node charging. While taking the charging decision, this situation must be avoided as radio interference caused by multiple MCs may result in a lower energy transfer efficiency. In addition to this, different network attributes may have different priorities while scheduling the nodes. This is because the network attributes RE, D, ECR, and CN are more critical for the nodes S_1 , S_2 , S_3 , and S_4 , respectively, as shown in Fig. 1b. So, an effective and wise decision is necessitated by coordinating different attributes to determine efficient charging schedule of the nodes.

This is noteworthy that there exist quite a few on-demand charging schemes [18], [19], [20], [21] which consider multi-node charging of the nodes with multiple MCs. The authors in [18], [19] formulated novel scheduling schemes for the longest delay minimization problem. They aimed at finding a closed charging tour for each MC, while ensuring that a sensor node cannot be simultaneously charged by two or more MCs. However, in their approach, charging load among the MCs was uneven and optimization of other important performance metrics, such as average charging latency and energy usage efficiency was relaxed. Han *et al.* [20] proposed a charging scheme which first groups the nodes into uneven clusters and then determines the charging schedule for the MCs by considering both the traveling distance and residual lifetime of the nodes. Nonetheless, balancing the charging load among the MCs was missing which should not be overlooked. Similarly, the author in [21] proposed a charging scheme which clusters the energy demands of nodes to balance the charging load fairly between the MCs. This scheme minimizes the charging delay by maximizing the number of nodes recharged at each sojourn point. Still, this scheme lacks the joint consideration of various aforementioned attributes while making the scheduling decisions. As a result, many energy-hungry nodes may die immaturely. Furthermore, all such aforesaid schemes make their charging schedule by setting a fixed charging threshold (beyond of which sensor nodes send their charging request), which is not likely to cope with heterogeneous rate of energy consumption of the nodes. This makes the MCs perform unnecessary visits of energy-sufficient nodes which prolongs the waiting time of energy-hungry nodes and causes their energy starvation.

Inspired by the above facts, our proposed scheduling scheme, which is based on fuzzy logic, incorporates following novel and salient features to address the aforementioned issues. First, the charging workload among the MCs is evenly distributed so as to ensure the charging fairness. Second, the Fuzzy Inference System (FIS) contemplates the aforementioned network attributes (i.e., RE, D, ECR, and CN) as inputs to make the scheduling decisions based on the pre-set fuzzy IF-THEN rules. Third, charging threshold for the nodes is dynamically set up in order to avoid untimely charging response by the MCs and thus improves the charging performance. Note that fuzzy logic system is a powerful tool for designing a charging scheduling scheme. Because, (i) it can integrate multiple parameters in a balanced manner to make real time decisions, even with incomplete and uncertain network information [22], and (ii) parameters used for scheduling decisions such as “lower residual energy”, “closer distance”, etc., make their fuzzy representation easy and realistic. To the best knowledge so far, we are the first to utilize the fuzzy logic for on-demand charging of the nodes using multiple MCs. We call our scheme the Fuzzy Logic-based Charging Schedule Determination (FLCSD). The contributions of this paper are as follows:

- Considering multiple energy constrained MCs, we first present a novel partitioning approach to distribute the MCs in the network such that the charging workload between them is evenly balanced.
- Then, an on-demand charging scheme is designed which utilizes the fuzzy logic for determining the charging schedule of the nodes by contemplating various network attributes.
- A formulation of calculating an adaptive threshold for the charging request is also presented with the aim of timely recharging the nodes with uneven and dynamic energy consumption rates.
- The proposed scheme is evaluated through extensive simulations and compared with two existing and similar works [20] and [21] in terms of energy usage efficiency, average charging latency and survival rate. The results are also statistically validated through analysis of variance (ANOVA) test.

The rest of this paper is organized as follows. Section 2 presents a brief literature review of on-demand charging schemes. Section 3 introduces some preliminaries, including the overview of fuzzy logic and system description. In Section 4, the proposed FLCSD scheme is discussed in detail. Section 5 derives the simulation results followed by performance evaluations. Finally, the paper concludes in Section 6.

2 LITERATURE REVIEW

A plethora of research efforts have been devoted for developing the scheduling schemes in on-demand WRSNs. We briefly review some of them as follows.

The Nearest-Job-Next with Preemption (NJNP) scheme was proposed in [12] where the closest node from the MC is selected to be charged next. However, in case of new charging request from a more closer node, the MC is preempted. In case of dense networks, too much preemption of the MC causes premature death of the nodes far away from the MC, thereby making NJNP deficient. In [14], a charging scheme

was proposed by taking into account both the requesting time of nodes and their distance from the MC, but in case of high number of charging requests occurred at the same time, it works same as NJNP which is not preferable. The authors in [23], aimed to schedule multiple MCs for energy supply to the nodes so as to maximize the covering utility. However, they failed to consider the other objectives, such as maximization of the survival rate along with minimum charging latency. In [11], the researchers focused to develop a metaheuristic-based charging scheme for scheduling the nodes. However, these schemes did not consider the multi-node charging of the nodes, thereby reducing the energy usage efficiency. Liang *et al.* [24] minimized the number of MCs for recharging the nodes and presented an approximation algorithm for the same. The authors in [16], came up with an approximated solution by utilizing the concept of smallest enclosing disk and determined optimal sojourn points for the MC. The authors in [13] designed a temporal-spatial charging scheme in order to plan an optimal charging path with the goal of minimizing the node deaths. Nonetheless, the above discussed schemes fail to take into account the multi-node charging and heterogeneous energy consumption of the nodes which should be given further consideration since it reduces the energy usage efficiency. The authors in [25] addressed the charging scheduling problem by considering both temporal deadline and spatial closeness among the charging requests. Xu *et al.* [26] solved the charging scheduling problem with multiple MCs by minimizing their total traveling distance. However, they failed to evenly balance the workload between the MCs, thereby causing more node deaths. Ma *et al.* [27] proposed a framework for multi-node charging using a single MC that measures the charging utility gain of each charged sensor, subject to the energy capacity of the MC. Gharaei *et al.* [28] proposed a route optimization of wireless mobile charger (ROWMC) algorithm in which two MCs are used, one for balancing the remaining lifetime of nodes and the other for guarantying that the energy level of nodes does not fall below a threshold.

Recently, the authors in [18] and [19] have reported algorithms for finding the charging tours of given MCs, subject to that no sensor can be charged by more than one MC at any time. Mo *et al.* [29] have addressed the multiple MCs coordination problem with an aim to minimize the total energy consumption of MCs by adjusting their moving and charging time. Khelladi *et al.* [30] considered multi-node charging for energy supply to the demanding nodes but they did not serve the objective of minimizing the charging latency. Moreover, considering multiple network attributes for taking scheduling decisions of the nodes was also missing. The author in [21] used the mechanisms of [30] and proposed a charging scheme which first clusters the energy demands to efficiently balance the charging load between MCs and then charges the nodes following certain scheduling rules. Finally, it computes the charging schedule for each MC by applying Traveling Salesman Problem (TSP) algorithm. The scheme aimed to improve charging performance in terms of charging latency and node failures. Han *et al.* [20] proposed the uneven cluster-based mobile charging (UCMC) scheme which first groups the nodes into uneven clusters and then determines the charging schedule

TABLE 1
Summary of the Existing on-Demand Charging Schemes

Paper	Scheduling based on multiple network attributes	Multiple MCs	Multi-node charging	Charging workload balancing between MCs	Adaptive charging threshold
[18]	No	Yes	Yes	No	No
[19]	No	Yes	Yes	No	No
[20]	No	Yes	Yes	Yes	No
[21]	No	Yes	Yes	No	No
[23]	No	Yes	No	No	No
[24]	No	Yes	No	No	No
[26]	No	Yes	No	No	No
[27]	No	No	Yes	No	No
[29]	No	Yes	No	No	No
[30]	No	No	Yes	No	Yes
[31]	Yes	No	Yes	No	No
Our	Yes	Yes	Yes	Yes	Yes

for the MCs by considering both the traveling distance and residual lifetime of the nodes. However, multi-hop energy transmission in their scheme suffers from low charging efficiency. Recently, Tomar *et al.* [31] have proposed a fuzzy logic-based charging scheduling scheme, but for a single MC and thus it is infeasible for dense WRSNs. Note that the aforesaid schemes [13], [20], [21], [24], [26], [29], [31] consider uniform charging threshold for the nodes which is not feasible in dynamic networks because the rate of nodes' energy depletion vary dynamically with the time.

These limitations arise the need to further investigate the scheduling problem of the nodes. The comparisons between the proposed and the existing schemes with respect to five key attributes are presented in Table 1.

3 PRELIMINARIES

3.1 Overview of Fuzzy Logic

A fuzzy logic system has three functional units: *fuzzification*, *fuzzy logic controller*, and *defuzzification*. In fuzzification, the inputs in the crisp form are mapped to their fuzzy form using some widely used membership functions. The fuzzy logic controller performs decision making based on FIS and rule base (composed of "IF{set of inputs}-THEN{outputs}" linguistic statements). A fuzzy rule R_i with m -inputs and 1-output is expressed as follows:

$$R_i : \text{IF}(I_1 \text{ is } A_{i1} \text{ AND } I_2 \text{ is } A_{i2} \text{ AND } \dots \text{ AND } I_m \text{ is } A_{im}) \text{ THEN } (O \text{ is } B_i), \quad (1)$$

where $\{I_1, I_2, \dots, I_m\}$ denote the input linguistic variables and O is the output linguistic variable. Here, linguistic values for input and output variables are represented by $\{A_{i1}, A_{i2}, \dots, A_{im}\}$ and B_i , respectively.

For evaluating the fuzzy outputs, the FIS uses some membership function (μ) for reasoning. For example, the fuzzy output for rule R_i is evaluated (using μ and Eq. (1)) as follows:

$$\text{eval}(R_i) : \left(\mu_{A_{i1}}(I_1) \text{ AND } \mu_{A_{i2}}(I_2) \text{ AND } \dots \text{ AND } \mu_{A_{im}}(I_m) \right) \longrightarrow \mu_{B_i}(O). \quad (2)$$

Note that in fuzzy logic controller, conventional T-operators, such as MAX, MIN and Complement are used instead of combination operators, such as OR, AND, and NOT, respectively, and are expressed as follows:

$$\mu_{A_i}(I_i) \text{ AND } \mu_{A_j}(I_j) = \mu_{A_i \cap A_j}(I_{ij}) = \min(\mu_{A_i}(I_i), \mu_{A_j}(I_j)) \quad (3)$$

$$\mu_{A_i}(I_i) \text{ OR } \mu_{A_j}(I_j) = \mu_{A_i \cup A_j}(I_{ij}) = \max(\mu_{A_i}(I_i), \mu_{A_j}(I_j)) \quad (4)$$

$$\text{NOT}(\mu_{A_i}(I_i)) = \mu_{\bar{A}_i}(I_i) = 1 - \mu_{A_i}(I_i). \quad (5)$$

Finally, the aggregated fuzzy output is mapped to its crisp form using widely accepted defuzzification method (i.e., *center of gravity*), defined as follows:

$$\text{Center of gravity (COG)} = \frac{\sum_{j=1}^p z_j \cdot \mu_C(z_j)}{\sum_{j=1}^p \mu_C(z_j)}, \quad (6)$$

where $\mu_C(z_j)$ is the modified membership functions of fuzzy output set C and p is the number of points over which the modified fuzzy output is spread in X -axis. Note that Eqs. (2)–(6) have been utilized in the proposed FLCSD scheme.

3.2 System Description

Without loss of generality, we envision a WRSN consisting of (i) N number of sensor nodes, randomly distributed over a 2D monitoring region, (ii) k (> 1) number of homogeneous MCs, and (iii) an immobile base station (BS) or service station which not only collects sensory data but also responsible for maintenance and other services of the MCs. The BS is in charge of scheduling the MCs and also able to recharge the battery of an MC in negligible time. Each node is equipped with a rechargeable battery, having maximum energy capacity E_{max} . We assume that the BS knows the location of each node and an event occurs at every node with the same possibility. Moreover, the nodes continuously sense the environment and send the sensed data to the BS in each data collection round. The network is also assumed to have no isolated node. Each MC is powered with energy capacity E_{mc} that can be used to provide multi-node energy transfer to the nearby nodes (i.e., nodes within the charging range) and also for its moving purpose. The working process of the system is as follows. Each node continuously keeps track of its remaining energy. When the energy level of a node falls below its charging threshold (which vary depending on its energy consumption rate (see Section 4.2)), it sends out a charging request to the assigned MC. The charging request messages contain ID and location information of the node. Based on the network information, the MC determines the charging schedule of the nodes. A charging schedule $C_{schedule}$ is the visiting order of requesting nodes by which an MC recharges them. Mathematically, it can be shown as follows:

$$C_{schedule} = \{BS \rightarrow s_i \rightarrow s_{i+1} \rightarrow \dots \rightarrow s_r\}, \quad (7)$$

where node s_{i+1} is scheduled to be recharged after s_i , and r denotes the total requesting nodes in the request queue (Q_{req}), i.e., $|Q_{req}|$. We use the same energy model as in [21], in which nodes may consume energy for transmitting, receiving, and computing tasks. However, energy consumption profiles of the nodes are not known in advance. Sensor nodes may have diverse energy consumption due to their uneven traffic loads,

TABLE 2
Important Notations

Notation	Description
s_i	Sensor node i
$N(i)$	Number of nodes in the i th region
$N_{coverage}(i)$	Number of nodes traversed by an MC in a recharging tour
E_{max}	Battery capacity of the nodes
E_{mc}	Battery capacity of the MCs
$D(s_i, mc)$	Distance between s_i and an MC
$D_{max}(BS)$	Maximum possible distance of a node from the BS
$CN(s_i)$	Critical node density of s_i
$RE(s_i)$	Residual energy of s_i
$RE(mc)$	Residual energy of an MC
$ECR(s_i)$	Energy consumption rate of s_i
E_{move}	Moving energy consumption of the MCs per meter distance
D_{max}	Maximum distance in the network between any two nodes
CN_{max}	Maximum critical node density of any node
RE_{max}	Maximum residual energy of any node
ECR_{max}	Maximum energy consumption rate of any node
Q_{req}	Queue in which charging requests are stored
$P_r(s_i)$	Charging energy received per second by s_i from an MC
$C_{thres}(s_i)$	Charging threshold of s_i
$T_{coverage}(i)$	Time to recharge $N_{coverage}(i)$ nodes during a charging tour
$SP(s_i)$	Scheduling probability of s_i
η	Charging energy efficiency rate
ω	Charging threshold of the MCs

depending on the underlying application or routing protocol. This means that the sensor nodes have heterogeneous and dynamic rate of energy consumption.

Note that all the MCs are assumed to know this dynamic rate of energy consumption of the nodes periodically using the technique of energy notification messages as described in [32]. Following the charging model in [21], the charging energy received by s_i (i.e., $P_r(s_i)$) is calculated as

$$P_r(s_i) = \begin{cases} \eta \times c_{rate}, & \text{if } D(s_i, mc) \leq C_r \\ 0, & \text{if } D(s_i, mc) > C_r \end{cases}, \quad (8)$$

where $D(s_i, mc)$ is distance between s_i and an MC, c_{rate} is the charging rate of the MCs, and η ($0 < \eta < 1$) denotes the charging energy efficiency rate. Since an MC usually has a small charging range C_r (e.g., $C_r \approx 2.7$ meters in [17]), we assume that there is no difference in the recharging energy among the nodes in terms of η and $D(s_i, mc)$. In case of multi-node charging, total stopping time of the MC (i.e., $t_{charge}(CN(s_i))$) for charging the nearby nodes, is calculated as

$$t_{charge}(CN(s_i)) = \max_{s_i \in CN(s_i)}(t_{charge}(s_i)), \quad (9)$$

where $t_{charge}(s_i)$ is the time taken by an MC to charge s_i , i.e., $t_{charge}(s_i) = E_{max}/P_r(s_i)$. We do not consider the handling of routing the charging requests in the paper. For simplicity, the routing time of charging requests is assumed to be negligible in comparison to the traveling and charging time of the MC [30]. Table 2 summarizes the important notations used in the paper.

4 PROPOSED SCHEME

The proposed scheme has the following basic operational steps: (1) after network initialization, it first partitions the network to evenly distribute the charging load of sensor

Fig. 2. Flowchart of the FLCSD scheme.

nodes among the MCs, (2) it then estimates a dynamic charging threshold for nodes based on their heterogeneous energy consumption rate, and (3) finally, it evaluates various network attributes jointly using fuzzy logic in order to determine most suitable next-to-be-charged node. All these aforementioned steps are presented in the form of a flow chart as shown in Fig. 2.

4.1 Network Partitioning

The goal of network partitioning is to assign the MCs to k different regions such that each MC belongs to one region only. The main idea is to evenly distribute the traffic load of the nodes in network so as to balance the charging load among the MCs. This provides in-time charging response to the nodes with uneven energy consumption rates. Note that the proposed approach is not affected by the size as well as the density

of the network. Besides, the BS performs network partitioning periodically by following the proposed approach so as to effectively deal with the network dynamics over the time.

The traffic loads of all the nodes are known to be a priori by following any standard routing algorithm. By traffic load of a node, we mean the total number of data packets forwarded by that node to the BS in one data collection round. The BS is assumed to be located at the center of the network. Let TTL be the total traffic load of network and RTL be the regional traffic load of a particular region. In general, there are two following conditions for an efficient network partitioning:

- (i) If $\text{mod}(TTL, k) == 0$, the network partitioning will perform efficient traffic load balancing when the RTL of each region is TTL/k .
- (ii) If $\text{mod}(TTL, k) \neq 0$, the network partitioning will perform efficient traffic load balancing when each of the $\text{mod}(TTL, k)$ regions has $RTL = (TTL/k) + 1$ and remaining regions have $RTL = TTL/k$.

Our network partitioning approach works as follows. For the sake of first partitioned region, a base axis line is considered from the BS which intersects the rectangular network horizontally (see Fig. 3b). Then, the angular distance of all the nodes from the base axis line is computed. The angular distance of the nodes is calculated by the formulas given in Fig. 3c. Next, all the angular distances are sorted in the non-decreasing order. Now, one by one all the nodes are selected in the non-decreasing order of their angular distances until $RTL \leq \lfloor \frac{TTL}{k} \rfloor$. All the nodes, satisfying this condition, belong to the first region. Likewise, this process is repeated until all nodes are covered and k number of regions are formed. Mathematically, it can be explained as follows. Let $AD(s_i)$ be the angular distance of i th node from the base axis line and $R'(m)$ be the m th region in the network. T denotes the set of nodes which have their angular distances in non-decreasing order, i.e., $T = \{s_1, s_2, \dots, s_N\}$, where $AD(s_1) \leq AD(s_2) \leq \dots \leq AD(s_N)$. Suppose l number of nodes from set T are already distributed to m number of regions ($m < k$), then if there exist j number of nodes $\{s_{l+1}, s_{l+2}, \dots, s_{l+j}\}$ from the remaining nodes, such that $RTL (= \sum_{c=l+1}^{l+j} TL_c) \leq \lfloor \frac{TTL}{k} \rfloor$, then $\{s_{l+1}, s_{l+2}, \dots, s_{l+j}\} \in R'(m+1)$, where TL_c is the traffic load of c th node. This means the nodes $\{s_{l+1}, s_{l+2}, \dots, s_{l+j}\}$ belong to the region next to $R'(m)$.

Fig. 3. Network partitioning approach with $k = 5$ showing, (a) network scenario with 25 nodes, (b) five partitions with $RTL = 8, 6, 6, 7$, and 13 , and (c) calculation of the angular distances of nodes from base axis line.

Fig. 4. Network partitioning approach with $k = 5$ showing, (a) Base axis at 0° , (b) Base axis at 60° , (c) Base axis at 120° , (d) Base axis at 180° , (e) Base axis at 240° , and (f) Base axis at 300° .

For the sake of illustrating the network partitioning approach, we consider a random network scenario shown in Fig. 3a, in which 25 nodes are distributed randomly. In figure, nodes are represented along with their ID and traffic load. The TTL of the network is 40. By assuming $k = 5$, the RTL should be $\lfloor \frac{40}{5} \rfloor = 8$. Now, we pick nodes one by one in non-decreasing order of their angular distances until their RTL becomes 8 ($RTL \leq 8$). The nodes 5, 8, 15, 19, and 22 are selected for the first region as their $RTL = 8$. This process is further repeated in a similar manner until all the five regions are formed. The final partitions are shown in Fig. 3b with $RTL = 8, 6, 6, 7$, and 13, respectively. It is evident from the above example that a sparse WRSN may have a case when RTL is not evenly balanced among the regions. Moreover, as the network density increases, considering different base axis lines may result in better traffic load distribution. In view of the above facts, our approach considers different base axis lines at different angular rotations while partitioning the network.

We consider another dense network scenario where base axis is rotated by 60° to create $(\frac{360^\circ}{60^\circ}) = 6$ different partitioning instances which is shown in Figs. 4a, 4b, 4c, 4d, 4e, and 4f. Let TLD be the total load deviation in the network and is defined as

$$TLD = \sum_{i=1}^k |RTL_{avg} - RTL_i|, \quad (10)$$

where RTL_i and RTL_{avg} denote the RTL of i th region and the average RTL of all the regions, respectively. Among all the partitioning instances (Figs. 4a, 4b, 4c, 4d, 4e, and 4f),

we select the instance having minimum TLD which is most efficient partitioning as shown in Fig. 4f with $TLD = 1.6$.

It can be further observed that in a highly dense WRSN, a slight change in the base axis may lead to different partitioning results. Thus, our approach suggests to change the base axis by 1° for each partition and then checks for the best partitioning results with almost equal RTL distribution. There can be some cases in which more than one instances have same minimum TLD , then the instance having minimum TDD is selected. Here, TDD is the total density deviation in the network and is defined as

$$TDD = \sum_{i=1}^k |RND_{avg} - RND_i|, \quad (11)$$

where RND_i denotes the regional node density, i.e., total number of nodes in i th region, and RND_{avg} denotes the average node density of all the regions.

In case of more than one instances having minimum TDD , any instance with minimum TDD can be selected. The TLD of all the $(\frac{360^\circ}{1^\circ}) = 360$ instances for a random network scenario is highlighted in Fig. 5a. In figure, the highlighted red colored boxes show the range of rotated base axis angles, where partitioning instances are created with minimum $TLD (= 1.6)$.

Here, we have shown only two instances (Figs. 5b and 5c) having minimum TLD . The Fig. 5c is chosen to be final partitioning as it has minimum $TDD = 4$. In Fig. 5c, it can be observed that traffic load is evenly distributed among all the five partitioned regions. At the same time, each partition of this instance has optimum node density distribution as

Fig. 5. Network partitioning approach with $k = 5$ showing, (a) variation of TLD with angle of rotation, (b) the partitioning instance with TLD = 1.6 and TDD = 6, (c) the partitioning instance with TLD = 1.6 and TDD = 4.

compared to other instances. After partitioning of the network, the BS assigns one MC in each region and notifies the nodes about ID of the MC assigned to them for sending the charging requests. Besides, each MC knows the location of nodes in its respective region.

4.2 Threshold Formulation of Charging Request

It is inarguable that unpredictable events occur in the network sporadically and thus energy consumption rates of nodes vary significantly. In such condition, fixed charging threshold for the nodes is not suitable for dynamic networks. A very low threshold results in response delay of charging requests, thereby causing more number of nodes to have premature death. On the other hand, too high threshold degrades the charging efficiency since too much time is spent on moving due to more frequent charging requests. Thus, formulating a reasonable charging threshold for the nodes is worth exploring so as to improve the charging performance in the network. Inspired by the above facts, an adaptive approach for calculating the charging threshold is proposed as follows.

Let us consider i th region in the network, where $N(i)$ be the total number of nodes, and $N_{coverage}(i)$ be the number of nodes which can be traversed by an MC in a recharging tour. Let $E_{coverage}$ be the total energy consumed by the MC in order to visit $N_{coverage}(i)$ nodes which is defined as

$$E_{coverage} = (N_{coverage}(i) + 1) \times D_{average} \times E_{move}, \quad (12)$$

where

$$D_{average} = \frac{1}{N(i) \times (N(i) + 1)} \sum_{x=1}^{N(i)+1} \sum_{y=1, y \neq x}^{N(i)+1} D(s_x, s_y). \quad (13)$$

Here, $D_{average}$ is the average euclidean distance between any two nodes, $(N_{coverage}(i) + 1)$ represents $N_{coverage}(i)$ nodes including the BS, $D(s_x, s_y)$ denotes the euclidean distance between node s_x and s_y , and E_{move} is the energy consumption of MC per meter traveling distance. Next, $N_{coverage}(i)$ can be computed as follows:

$$N_{coverage}(i) = \left\lfloor \frac{E_{mc} - E_{coverage}}{E_{max}} \right\rfloor, \quad (14)$$

where E_{mc} and E_{max} are the battery capacity of the MCs and the nodes, respectively. In Eq. (14), the value of $E_{coverage}$ is substituted from Eq. (12) to calculate $N_{coverage}(i)$. Then, the total time ($T_{coverage}(i)$) taken by the MC to recharge $N_{coverage}(i)$ nodes during its tour is computed as follows:

$$T_{coverage}(i) = (N_{coverage}(i) + 1) \times \left(\frac{D_{average}}{v_{mc}} \right) + N_{coverage}(i) \times \left(\frac{E_{max}}{\eta \times C_{rate}} \right). \quad (15)$$

For the sake of satisfying all the energy demands, the MC needs to complete $\left\lceil \frac{N(i)}{N_{coverage}(i)} \right\rceil$ charging tours. Thus, the maximum waiting time of a node can be computed as $T_{coverage}(i) \times \left\lceil \frac{N(i)}{N_{coverage}(i)} \right\rceil$. In view of the above facts, the charging threshold of s_i (i.e., $C_{thres}(s_i)$) is calculated as follows:

$$C_{thres}(s_i) = ECR(s_i) \times T_{coverage}(i) \times \left\lceil \frac{N(i)}{N_{coverage}(i)} \right\rceil. \quad (16)$$

4.3 Charging Schedule Determination

As discussed earlier, each MC computes the charging schedule of its regional nodes. When an MC receives the charging requests of its regional nodes, it stores them into Q_{req} . Based on the network information, the MC stores the values of all the network attributes. Note that the critical node density for a node can be computed as the total number of requesting nodes which fall within the charging range of that node. Thereupon, scheduling probability of each requesting node is computed by applying the FIS in order to determine the most suitable next-to-be-charged node. The MC visits that node and recharges it along with the other requesting nodes within the charging range. After completion of charging, the MC deletes the recharged nodes from the Q_{req} . Then, the MC again updates the network information of remaining nodes and computes the next most suitable next-to-be-charged node. Likewise, all the requesting nodes are scheduled and recharged one by one iteratively. Thus, we can say that charging schedule determination follows a real-time and dynamic process. If in-between, new charging requests are triggered, then they wait until previous charging schedule is completed. Before serving a scheduled node s_i , an MC considers the following two cases.

Case 1: Let $D_{max}(BS)$ be the maximum possible distance of a node from the BS. If MC's residual energy is higher than threshold value ω (where $\omega \geq E_{move} \times D_{max}(BS)$), then it continues to recharge the scheduled nodes.

Case 2: If MC has remaining energy less than the ω , nodes in Q_{req} are judged whether they meet the following constraint

Fig. 6. The FIS of FLCSD scheme: 4 inputs, 1 output, 81 rules.

Fig. 7. Fuzzy input membership functions: (a) Residual energy, (b) Distance to MC, (c) Critical node density, and (d) Energy consumption rate.

$$E_{move} \times D(s_i, BS) \leq RE(mc)(t) - (E_{move} \times D(s_i, mc)) - (E_{max} - RE(s_i)(t + t')), \quad (17)$$

where $RE(mc)(t)$ denotes the residual energy of MC at time t , t' is the time when MC reaches the node s_i , $RE(s_i)(t + t')$ is the remaining energy of s_i at time $(t + t')$.

Fig. 8. Fuzzy output membership function: Scheduling probability.

In summary, the MC first examines if after recharging the next scheduled node, it has sufficient remaining energy for reaching the BS. Nodes which satisfy the constraint (Eq. (17)) are chosen as the candidate nodes from which the next node is scheduled for recharging by following the proposed approach. If no requesting node satisfies the above constraint, then the MC do not entertain any other charging request and instantly returns to the BS for its recharging.

Next, we elucidate the process of determining the next-to-be-charged node using the FLCSD scheme. The FIS in our scheme consists of 4-inputs and 1-output fuzzy linguistic variables, namely *residual energy* (RE), *distance to MC* (D), *critical node density* (CN), *energy consumption rate* (ECR), and *scheduling probability* (SP), respectively. The design of MAMDANI FIS is presented in Fig. 6 and its pseudo-code is delineated in Procedure FUZZY. Figs. 7 and 8 depict trapezoidal and triangular membership functions and show that these fuzzy variables are represented by more than one fuzzy linguistic values. The input and output fuzzy variables and their corresponding linguistic values are presented in Tables 3 and 4, respectively.

The proposed FLCSD scheme uses following trapezoidal and triangular membership functions

$$Trapezoidal(x; a, b, c, d) = \begin{cases} 0 & , x \leq a \\ \frac{x-a}{b-a} & , a \leq x \leq b \\ 1 & , b \leq x \leq c \\ \frac{d-x}{d-c} & , c \leq x \leq d \\ 0 & , d \leq x \end{cases} \quad (18)$$

$$Triangular(x; a, b, c) = \begin{cases} 0 & , x \leq a \\ \frac{x-a}{b-a} & , a \leq x \leq b \\ \frac{c-x}{c-b} & , b \leq x \leq c \\ 0 & , c \leq x \end{cases} \quad (19)$$

TABLE 3
Input Variables With Their Linguistic Values and Corresponding Membership Function Ranges

Input variable	Linguistic values	Membership function range
Residual energy (RE)	low (L)	$[0, 0, 0.3 * E_{max}, 0.6 * E_{max}]$
	medium (M)	$[0.3 * E_{max}, 0.6 * E_{max}, 0.9 * E_{max}]$
	high (H)	$[0.6 * E_{max}, 0.9 * E_{max}, E_{max}, E_{max}]$
Distance to MC (D)	near (N)	$[0, 0, 0.25 * D_{max}, 0.35 * D_{max}]$
	average (A)	$[0.25 * D_{max}, 0.35 * D_{max}, 0.65 * D_{max}, 0.75 * D_{max}]$
	far (F)	$[0.65 * D_{max}, 0.75 * D_{max}, D_{max}, D_{max}]$
Critical node density (CN)	few (F)	$[0, 0, 0.25 * CN_{max}, 0.35 * CN_{max}]$
	average (A)	$[0.25 * CN_{max}, 0.35 * CN_{max}, 0.65 * CN_{max}, 0.75 * CN_{max}]$
	many (M)	$[0.65 * CN_{max}, 0.75 * CN_{max}, CN_{max}, CN_{max}]$
Energy consumption rate (ECR)	low (L)	$[0, 0, 0.3 * ECR_{max}, 0.6 * ECR_{max}]$
	medium (M)	$[0.3 * ECR_{max}, 0.6 * ECR_{max}, 0.9 * ECR_{max}]$
	high (H)	$[0.6 * ECR_{max}, 0.9 * ECR_{max}, ECR_{max}, ECR_{max}]$

TABLE 4
Output Variable With its Linguistic Values and Membership Function Ranges

Output variable	Linguistic values	Membership function range
Scheduling probability (SP)	very low (VL)	[0, 0, 0.1, 0.15]
	low (L)	[0.1, 0.15, 0.35, 0.4]
	moderate (M)	[0.35, 0.4, 0.6, 0.65]
	high (H)	[0.6, 0.65, 0.85, 0.9]
	very high (VH)	[0.85, 0.9, 1, 1]

where x is the crisp input, and a, b, c, d are membership function ranges of the fuzzy variables. Note that these values are independent of network size and node count in the proposed scheme.

The complete rule set in the proposed scheme, consists of $3^4 = 81$ rules as presented in Table 5. We now transform the crisp data inputs to the fuzzy inputs using the Trapezoidal and Triangular membership functions as given in Eqs. (18) and (19), respectively. Let us consider a random charging request for a node in the Q_{req} , e.g., $RE = 1.52 \text{ KJ}$, $D = 32 \text{ m}$, $CN = 1$, and $ECR = 0.02 \text{ J/s}$. Here, we consider $E_{max} = 10.8 \text{ KJ}$, $D_{max} = 125.09 \text{ m}$, $CN_{max} = 5$, and $ECR_{max} = 0.08 \text{ J/s}$. Then, the fuzzy inputs of different linguistic variables are calculated from the aforesaid crisp inputs using Eqs. (18) and (19) as presented in Table 6.

Now, the fuzzy outputs for each fuzzy rule are evaluated from fuzzy inputs (using Eqs. (2)–(5)). The evaluated values

TABLE 6
Calculation of Fuzzy Inputs From Crisp Inputs Using Input Fuzzy Membership Functions

$low(RE)=1$	$near(D)=0.942$	$few(CN)=1$	$low(ECR)=1$
$medium(RE)=0$	$average(D)=0.058$	$average(CN)=0$	$medium(ECR)=0$
$high(RE)=0$	$far(D)=0$	$many(CN)=0$	$high(ECR)=0$

for all the fuzzy rules are presented in Table 7. For example, the fuzzy output for rule R_1 is calculated as follows:

$$\Rightarrow eval(R_1) = \min(low(RE), near(D), few(CN), low(ECR)) \\ = moderate(SP) = 0.942.$$

The aggregated fuzzy output is generated from the modified fuzzy outputs in Table 7. The fuzzy membership ranges of SP are span over the intervals, such as $VL=0.1, 0.1029, 0.3971, 0.4$, $M=[0.35, 0.3971, 0.6029, 0.65]$, $H=0.6, 0.6029, 0.85, 0.9$, and $VH=0.85, 0.9, 1, 1$. Finally, the crisp output (i.e., defuzzification point) is calculated as 0.509 (see Fig. 9) from the aggregated fuzzy output by using *center of gravity* defuzzification method (Eq. (6)). The computed crisp output for a node is the value of its scheduling probability, i.e., SP value. It is to note here that the node having the highest SP value is scheduled as the next-to-be-charged node. In case of more than one node having same SP value, the node closest to the MC is given higher priority for recharging the next. The pseudo-code of the proposed scheme is shown in Algorithm 1.

TABLE 5
Fuzzy IF-THEN Rules for Scheduling Probability Determination

R#	Inputs				Output SP	R#	Inputs				Output SP	R#	Inputs				Output SP
	RE	D	CN	ECR			RE	D	CN	ECR			RE	D	CN	ECR	
R_1	L	N	F	L	M	R_{28}	M	N	F	L	L	R_{55}	H	N	F	L	VL
R_2	L	N	F	M	M	R_{29}	M	N	F	M	M	R_{56}	H	N	F	M	VL
R_3	L	N	F	H	VH	R_{30}	M	N	F	H	H	R_{57}	H	N	F	H	L
R_4	L	N	A	L	M	R_{31}	M	N	A	L	M	R_{58}	H	N	A	L	VL
R_5	L	N	A	M	H	R_{32}	M	N	A	M	H	R_{59}	H	N	A	M	L
R_6	L	N	A	H	VH	R_{33}	M	N	A	H	H	R_{60}	H	N	A	H	M
R_7	L	N	M	L	H	R_{34}	M	N	M	L	M	R_{61}	H	N	M	L	L
R_8	L	N	M	M	VH	R_{35}	M	N	M	M	H	R_{62}	H	N	M	M	M
R_9	L	N	M	H	VH	R_{36}	M	N	M	H	VH	R_{63}	H	N	M	H	H
R_{10}	L	A	F	L	L	R_{37}	M	A	F	L	L	R_{64}	H	A	F	L	VL
R_{11}	L	A	F	M	L	R_{38}	M	A	F	M	L	R_{65}	H	A	F	M	VL
R_{12}	L	A	F	H	M	R_{39}	M	A	F	H	H	R_{66}	H	A	F	H	L
R_{13}	L	A	A	L	L	R_{40}	M	A	A	L	M	R_{67}	H	A	A	L	VL
R_{14}	L	A	A	M	M	R_{41}	M	A	A	M	M	R_{68}	H	A	A	M	L
R_{15}	L	A	A	H	H	R_{42}	M	A	A	H	H	R_{69}	H	A	A	H	M
R_{16}	L	A	M	L	M	R_{43}	M	A	M	L	M	R_{70}	H	A	M	L	L
R_{17}	L	A	M	M	H	R_{44}	M	A	M	M	M	R_{71}	H	A	M	M	L
R_{18}	L	A	M	H	VH	R_{45}	M	A	M	H	H	R_{72}	H	A	M	H	M
R_{19}	L	F	F	L	L	R_{46}	M	F	F	L	L	R_{73}	H	F	F	L	VL
R_{20}	L	F	F	M	L	R_{47}	M	F	F	M	L	R_{74}	H	F	F	M	VL
R_{21}	L	F	F	H	M	R_{48}	M	F	F	H	M	R_{75}	H	F	F	H	L
R_{22}	L	F	A	L	L	R_{49}	M	F	A	L	M	R_{76}	H	F	A	L	VL
R_{23}	L	F	A	M	L	R_{50}	M	F	A	M	M	R_{77}	H	F	A	M	L
R_{24}	L	F	A	H	H	R_{51}	M	F	A	H	H	R_{78}	H	F	A	H	L
R_{25}	L	F	M	L	L	R_{52}	M	F	M	L	M	R_{79}	H	F	M	L	L
R_{26}	L	F	M	M	M	R_{53}	M	F	M	M	M	R_{80}	H	F	M	M	L
R_{27}	L	F	M	H	H	R_{54}	M	F	M	H	H	R_{81}	H	F	M	H	M

TABLE 7
Fuzzy Rules Evaluation

$R_{\#}$	$eval(R_{\#})$	$R_{\#}$	$eval(R_{\#})$	$R_{\#}$	$eval(R_{\#})$	$R_{\#}$	$eval(R_{\#})$	$R_{\#}$	$eval(R_{\#})$
R_1	0.942	R_{15}	0	R_{29}	0	R_{43}	0	R_{57}	0
R_2	0	R_{16}	0	R_{30}	0	R_{44}	0	R_{58}	0
R_3	0	R_{17}	0	R_{31}	0	R_{45}	0	R_{59}	0
R_4	0	R_{18}	0	R_{32}	0	R_{46}	0	R_{60}	0
R_5	0	R_{19}	0	R_{33}	0	R_{47}	0	R_{61}	0
R_6	0	R_{20}	0	R_{34}	0	R_{48}	0	R_{62}	0
R_7	0	R_{21}	0	R_{35}	0	R_{49}	0	R_{63}	0
R_8	0	R_{22}	0	R_{36}	0	R_{50}	0	R_{64}	0
R_9	0	R_{23}	0	R_{37}	0	R_{51}	0	R_{65}	0
R_{10}	0.058	R_{24}	0	R_{38}	0	R_{52}	0	R_{66}	0
R_{11}	0	R_{25}	0	R_{39}	0	R_{53}	0	R_{67}	0
R_{12}	0	R_{26}	0	R_{40}	0	R_{54}	0	R_{68}	-
R_{13}	0	R_{27}	0	R_{41}	0	R_{55}	0	R_{69}	-
R_{14}	0	R_{28}	0	R_{42}	0	R_{56}	0	R_{70}	-

Algorithm 1. Charging Schedule Determination

Input: Q_{req} : queue containing ID of each requesting node;
 RE_{max} ; D_{max} ; CN_{max} ; ECR_{max}
Output: $C_{schedule}$: charging schedule

- 1 MC stores information of requesting nodes in Q_{req} $\triangleright Q_{req}$
- contains $[RE(s_i), D(s_i, mc), CN(s_i), ECR(s_i)]$, $\forall s_i \in Q_{req}$
- 2 **while** $Q_{req} \neq \text{NULL}$ **do**
- 3 $\text{FUZZY}(RE_{max}, D_{max}, CN_{max}, ECR_{max}, Q_{req})$;
- 4 Select s_i with highest $SP(s_i)$ value $\triangleright SP(s_i)$ is the
- scheduling probability of s_i ;
- 5 $C_{schedule} \leftarrow C_{schedule} \cup s_i$;
- 6 Delete s_i from Q_{req} ;
- 7 Update $[RE(s_j), D(s_j, mc), CN(s_j), ECR(s_j)]$ for all remain-
- ing nodes $s_j \in Q_{req}$;

Procedure FUZZY($RE_{max}, D_{max}, CN_{max}, ECR_{max}, Q_{req}$)

Input: Q_{req} (stored crisp inputs); $r: |Q_{req}|$; q : number of
fuzzy linguistic variables; m : number of fuzzy lin-
guistic values
Output: SP : set of scheduling probabilities (crisp output)

- 1 Define input and output fuzzy linguistic variables;
- 2 Define fuzzy linguistic values for each linguistic variables;
- 3 **for** $i \leftarrow 1$ **to** m^q **do**
- 4 Establish fuzzy rule R_i $\triangleright$ using Eq. (1);
- 5 $SP \leftarrow \phi$ $\triangleright$ Initialize SP ;
- 6 **for** $i \leftarrow 1$ **to** r **do**
- 7 **for** $j \leftarrow 1$ **to** q **do**
- 8 **for** $k \leftarrow 1$ **to** m **do**
- 9 Mapping crisp input $[RE(s_i), D(s_i, mc), CN(s_i),$
 $ECR(s_i)]$ to fuzzy input using input membership
functions;
- 10 $\text{Aggregate} \leftarrow \phi$ $\triangleright$ Initialize aggregated output set ;
- 11 **for** $i \leftarrow 1$ **to** m^q **do**
- 12 Evaluate fuzzy rule R_i , i.e., $eval(R_i)$ $\triangleright$ using Eqs. (2)–(5);
- 13 $\text{Aggregate} \leftarrow \text{Aggregate} \cup eval(R_i)$ $\triangleright$ Aggregate each
- evaluated fuzzy outputs;
- 14 Calculate crisp output $SP(s_i)$ $\triangleright$ using Eq. (6);
- 15 $SP \leftarrow SP \cup SP(s_i)$
- 16 **return** SP

Fig. 9. Aggregated fuzzy output.

5 SIMULATION ANALYSIS

To testify the benefits of the proposed scheme, extensive simulations have been carried out. The charging schemes proposed in [20] and [21] have been used for comparing and evaluating the proposed FLCSD scheme. Both the schemes [20] and [21] follow an on-demand multi-node charging architecture, employing multiple MCs. We have used the scheme [20] as UCMC and the scheme [21] as OMC_TSP in the interest of depicting the performance results. For fair comparisons, it is assumed that there is no aggregated electromagnetic radiation (EMR) between nearby MCs. We mainly focus on three performance metrics, namely *energy usage efficiency*, *survival rate*, and *average charging latency* in order to judge the efficacy of the proposed scheme.

5.1 Simulation Environment

We have considered a dense WRSN with 400 to 1000 nodes randomly placed in $100 \times 100 \text{ m}^2$ region. The BS is located at the center of the network. We have assumed the following parametric values for the simulation. Each node has power capacity of 10.8 KJ [27]. The maximum energy of each MC is 4000 KJ [27]. Each MC charges at the rate of 5 J/s [21] with $\eta \approx 0.68$ [17]. Each MC has moving speed of v_{mc} with per meter energy consumption of 600 J [24], [27]. The nodes randomly consume between 0.01 to 0.05 J/s for sensing, communication, and computation tasks in order to emulate their non-predictable energy consumption [21]. The simulations are performed in MATLAB version R2016a using a system of 4 GB RAM and Intel Core 2 Duo processor. We have considered the average of 20 random simulation instances for depicting the result in each figure for the sake of fair comparisons. The important simulation parameters with their values are listed in Table 8.

5.2 Performance Evaluation

In this section, we assess the impact of various network parameters (i.e., *number of nodes*, *simulation time*, and *moving speed of MC*) on the performance metrics.

5.2.1 Energy Usage Efficiency

Energy usage efficiency (EUE) is defined as the ratio between the energy obtained by nodes and the overall energy transferred from the BS to the MCs [33]. The EUE plays a vital role in achieving the promising charging performance. The higher EUE confirms that more number of nodes are getting charged in time. It is observed from Figs. 10a, 10b, and 10c that the EUE of FLCSD is relatively high with that of other

TABLE 8
Simulation Parameters

Parameter	Value
Network region	$100 \times 100 \text{ m}^2$
Number of nodes (N)	400 – 1000
Number of MCs (k)	5
Battery capacity of nodes (E_{max})	10.8 KJ
Battery capacity of MCs (E_{mc})	4000 KJ
Moving energy consumption rate (E_{move})	600 J/m
Moving speed of MC (v_{mc})	1 – 7 m/s
Charging range (C_r)	2.7 m
Charging rate of MCs (c_{rate})	5 J/s
Charging energy efficiency rate (η)	0.68
Simulation time	200 hours

two schemes, with increasing number of nodes, simulation time, and moving speed of the MC, respectively. This demonstrates the high performance of proposed scheme as compared to other ones. This can be understood with the fact that fuzzy logic in the FLCSD tends to make wise and efficient scheduling decisions which provides charging opportunity to more number of nodes in time. Thus, it maximizes the amount of energy transferred to the nodes. On the other hand, unlike FLCSD, in UCMC and OMC_TSP, the charging load of MCs is not efficiently balanced and thus they fail to cope with the heterogeneous and dynamic energy consumption of the nodes. Moreover, the efficient selection criteria for choosing the most suitable node for recharging is also missing in these schemes, as a result, lesser nodes get recharged in time, thereby reducing the EUE. Further, we can observe from Figs. 10a, 10b, and 10c that the EUE of all the schemes shows increasing trend while the value of corresponding network parameters is also increasing. This is reasonable because more number of charging requests are delivered to

the MCs with the increased number of nodes (see Fig. 10a). This ensures that more energy is obtained by the nodes while satisfying the requests. On an average, the EUE of the FLCSD is 7 and 12 percent higher than the UCMC and the OMC_TSP, respectively. Similarly, as the simulation time goes on (in Fig. 10b) and speed of the MC increases (in Fig. 10c), more number of charging requests are eventually served by the MCs, thereby increasing the overall obtained energy by the nodes and so EUE.

5.2.2 Survival Rate

Survival rate (SR) is defined as the ratio of the number of successfully charged nodes by the MCs and the total number of requesting nodes. The comparison results for SR with varying simulation time and moving speed of MC are shown in Figs. 10d and 11a, respectively. A high SR in the network guarantees the connectivity and stability of the network, thus it is vital for improving the charging performance.

The FLCSD performs scheduling of the nodes by blending multiple factors together so as to respond more requests. Also due to the load balanced network partitioning, each MC takes shorter time to complete the recharging tasks, thus more nodes would get charged and so the SR increases. It is clear from the Figs. 10d and 11a that the SR in the FLCSD is always higher than that of the UCMC and the OMC_TSP. Furthermore, in Fig. 10d, with the simulation going on, the SR gets decreased. The reason is that, the increase in the simulation time leads to a bigger number of charging requests which impose a heavier load on the MC for fulfilling the charging requests. Therefore, more exhausted nodes will appear. Furthermore, the survival rates of compared schemes sharply decreases than that of the proposed FLCSD scheme as observed in Fig. 10d. This is because, the compared schemes ignore making scheduling

Fig. 10. Performance comparison of (a) EUE versus number of nodes. (b) EUE versus simulation time. (c) EUE versus moving speed of MC. (d) Survival rate versus simulation time.

Fig. 11. Performance comparison of (a) Survival rate versus moving speed of MC. (b) ACL versus number of nodes. (c) ACL versus simulation time. (d) ACL versus moving speed of MC.

TABLE 9
Outcomes of ANOVA

Performance metric	F -statistic	p -value	F -critical
Energy usage efficiency	12.51752712	3.11493E-05	3.158842719
Survival rate	116.5302981	2.56652E-20	3.168245967
Average charging latency	13.39063491	1.70901E-05	3.158842719

decisions by incorporating multiple network attributes and consider fixed charging threshold for the nodes. This overlooks the exigency of recharging, making the nodes with more critical residual energy frequently die in case of large number of charging requests. Similarly, in Fig. 11a, the SR of all the simulated schemes increases when increasing moving speed of the MC. The rationale is that faster the MC moves, more requesting nodes the MC can serve. Hence, total number of successfully charged nodes increases, which makes the survival rate of all these schemes rise up.

5.2.3 Average Charging Latency

Average charging latency (ACL) is defined as the average time interval since the arrival time of a charging request to its completion time by the MC. The average charging latency is less, which means other critical nodes have to wait for a shorter time period, thereby preventing them to die earlier due to energy exhaustion.

It is clear to see in Figs. 11b and 11c that the ACL of FLCSD scheme is always less than that of others with varying number of nodes and simulation time, respectively. The reason is clear as the MCs have to travel a lesser distance because a lot of sensors are charged through multi-node charging; second, requesting nodes are evenly distributed in the network by balancing their traffic load, thereby reducing the charging burden on the MCs and also shortening the response time for recharging the nodes. As a result, the MC tends to serve more number of requests while reducing the ACL in the network. Moreover, adaptive charging thresholds help nodes to send their charging requests in time without much delaying their charging response time. On the other hand, state-of-the-art schemes fail to consider above factors and make inefficient scheduling decisions. Moreover, charging workload of each MC remain unbalanced in the network and thus the ACL dramatically increases. Fig. 11d plots the ACL obtained by the simulated schemes for different moving speed of the MC. We can easily see that the ACL decreases as the speed of MC increases but the decreasing paces are different. This is due to the fact that if the speed of MC is higher, it takes less traveling time to serve the requesting nodes and this decreases the ACL. At the beginning, when moving speed of the MC are $\{1, 2, 3\}$ m/s, the ACL of all the schemes decreases at the same rate. As the speed of MC further increases, the ACL of the FLCSD scheme decreases more rapidly. Moreover, the ACL of FLCSD is recorded the lowest, i.e., 6 percent lower than the UCMC and 9 percent lower than the OMC_TSP.

5.2.4 Statistical Validation

In this section, we check the statistical significance of the simulation results through hypothesis testing. We use analysis of variance (ANOVA) [34] to accomplish the same. ANOVA

Fig. 12. Confidence interval plots for FLCSD, UCMC, and OMC_TSP schemes in terms of (a) EUE, (b) SR, and (c) ACL.

is a statistical method, mainly used to differentiate between two or more means. More specifically, it helps to judge whether a null hypothesis (H_0 : the given groups have the same means), can be rejected. Let F -statistic be the outcome of the ANOVA test. There are two important conditions for rejecting a null hypothesis, one is p -value should not be higher than the significance level (α) and the other is F -statistic must be greater than F -critical. We have performed the ANOVA test on each performance metric with $\alpha = 0.05$ by considering 20 samples of each scheme under the same network instance. We define the null hypothesis as " $H_0 : \mu_{FLCSD} = \mu_{UCMC} = \mu_{OMC_TSP}$ " and the alternative hypothesis as " $H_1 : \mu_{FLCSD} \neq \mu_{UCMC} \neq \mu_{OMC_TSP}$ ". The outcomes of ANOVA test in term of F -statistic, p -value, and F -critical are presented in Table 9. On analyzing the results, we can observe that both the aforementioned conditions are satisfied (i.e., the F -statistic is greater than the F -critical and p -value is much less than 0.05). Thus, we deduce that the null hypothesis is rejected. This means that there are statistically significant differences in the means of each performance metric for all the simulated schemes. We have also shown the 95 percent confidence interval of the means of the simulated schemes with respect to all the performance metrics. Fig. 12 depicts the interval plots using the normal distribution, where μ_1 , μ_2 , and μ_3 denote the mean of FLCSD, UCMC, and OMC_TSP, respectively.

6 CONCLUSION

Taking into account the MC's limited power supply and multi-node energy transfer capability, we have investigated the charging scheduling problem, using multiple MCs in on-demand WRSNs. First, we have presented a novel network partitioning approach to distribute the MCs in a load balanced manner. Next, with an objective of avoiding the unfair charging response by the MCs, we have established a mathematical formulation to calculate the adaptive recharging thresholds for the nodes. Finally, we have adopted the Mamdani Fuzzy Inference System which determines the charging schedules of the nodes by wisely integrating

multiple attributes, such as residual energy, distance to MC, critical node density, and energy consumption rate. Rigorous simulations have been carried out to prove the supremacy of the proposed FLCSD scheme over the state-of-the-art schemes. The results have revealed that as compared to the state-of-the-art schemes, the FLCSD provides better stability by increasing the number of survived nodes and boosts the charging performance by achieving both low average charging latency and high energy usage efficiency. The simulation results are also validated through the statistical ANOVA test. However, the proposed scheme has not considered the coordination among the MCs for recharging the nodes as well as the MCs, which may further improve the charging performance, especially in large-scale WRSNs. We will try to incorporate this feature in our future work by exploring new multi-attribute decision making methods.

REFERENCES

- [1] M. Ayaz, M. Ammad-uddin, I. Baig, and E. M. Aggoune, "Wireless sensor's civil applications, prototypes, and future integration possibilities: A review," *IEEE Sensors J.*, vol. 18, no. 1, pp. 4–30, Jan. 2018.
- [2] P. Kar and S. Misra, "Reliable and efficient data acquisition in wireless sensor networks in the presence of transfaulity nodes," *IEEE Trans. Netw. Service Manag.*, vol. 13, no. 1, pp. 99–112, Mar. 2016.
- [3] S. Gao, H. Zhang, and S. K. Das, "Efficient data collection in wireless sensor networks with path-constrained mobile sinks," *IEEE Trans. Mobile Comput.*, vol. 10, no. 4, pp. 592–608, Apr. 2011.
- [4] B. Fateh and M. Govindarasu, "Joint scheduling of tasks and messages for energy minimization in interference-aware real-time sensor networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 1, pp. 86–98, Jan. 2015.
- [5] S. Guo, Y. Shi, Y. Yang, and B. Xiao, "Energy efficiency maximization in mobile wireless energy harvesting sensor networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 7, pp. 1524–1537, Jul. 2018.
- [6] A. Mehrabi and K. Kim, "General framework for network throughput maximization in sink-based energy harvesting wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 7, pp. 1881–1896, Jul. 2017.
- [7] A. Kurs, A. Karalis, R. Moffatt, J. D. Joannopoulos, P. Fisher, and M. Soljačić, "Wireless power transfer via strongly coupled magnetic resonances," *Science*, vol. 317, no. 5834, pp. 83–86, 2007.
- [8] K. Kang, Y. S. Meng, J. Bréger, C. P. Grey, and G. Ceder, "Electrodes with high power and high capacity for rechargeable lithium batteries," *Science*, vol. 311, no. 5763, pp. 977–980, 2006.
- [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, 2018.
- [10] L. Xie, Y. Shi, Y. T. Hou, W. Lou, H. D. Sherali, and S. F. Midkiff, "Multi-node wireless energy charging in sensor networks," *IEEE/ACM Trans. Netw.*, vol. 23, no. 2, pp. 437–450, Apr. 2015.
- [11] A. Kaswan, A. Tomar, and P. K. Jana, "An efficient scheduling scheme for mobile charger in on-demand wireless rechargeable sensor networks," *J. Netw. Comput. Appl.*, vol. 114, pp. 123–134, 2018.
- [12] L. He, L. Kong, Y. Gu, J. Pan, and T. Zhu, "Evaluating the on-demand mobile charging in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 9, pp. 1861–1875, Sep. 2015.
- [13] 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.
- [14] C. Lin, Z. Wang, D. Han, Y. Wu, C. W. Yu, and G. Wu, "TADP: Enabling temporal and distastial priority scheduling for on-demand charging architecture in wireless rechargeable sensor networks," *J. Syst. Archit.*, vol. 70, pp. 26–38, 2016.
- [15] Y. Shu *et al.*, "Near-optimal velocity control for mobile charging in wireless rechargeable sensor networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 7, pp. 1699–1713, Jul. 2016.
- [16] L. Fu, P. Cheng, Y. Gu, J. Chen, and T. He, "Optimal charging in wireless rechargeable sensor networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 1, pp. 278–291, Jan. 2016.
- [17] A. Kurs, R. Moffatt, and M. Soljačić, "Simultaneous mid-range power transfer to multiple devices," *Appl. Phys. Lett.*, vol. 96, no. 4, 2010, Art. no. 044102.
- [18] W. Xu, W. Liang, H. Kan, Y. Xu, and X. Zhang, "Minimizing the longest charge delay of multiple mobile chargers for wireless rechargeable sensor networks by charging multiple sensors simultaneously," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst.*, 2019, pp. 881–890.
- [19] W. Xu, W. Liang, X. Jia, H. Kan, Y. Xu, and X. Zhang, "Minimizing the maximum charging delay of multiple mobile chargers under the multi-node energy charging scheme," *IEEE Trans. Mobile Comput.*, to be published, doi: [10.1109/TMC.2020.2973979](https://doi.org/10.1109/TMC.2020.2973979).
- [20] G. Han, H. Guan, J. Wu, S. Chan, L. Shu, and W. Zhang, "An uneven cluster-based mobile charging algorithm for wireless rechargeable sensor networks," *IEEE Syst. J.*, vol. 13, no. 4, pp. 3747–3758, Dec. 2019.
- [21] T. Rault, "Avoiding radiation of on-demand multi-node energy charging with multiple mobile chargers," *Comput. Commun.*, vol. 134, pp. 42–51, 2019.
- [22] R. V. Kulkarni, A. Forster, and G. K. Venayagamoorthy, "Computational intelligence in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 13, no. 1, pp. 68–96, First Quarter 2011.
- [23] L. Jiang, X. Wu, G. Chen, and Y. Li, "Effective on-demand mobile charger scheduling for maximizing coverage in wireless rechargeable sensor networks," *Mobile Netw. Appl.*, vol. 19, no. 4, pp. 543–551, 2014.
- [24] W. Liang, W. Xu, X. Ren, X. Jia, and X. Lin, "Maintaining large-scale rechargeable sensor networks perpetually via multiple mobile charging vehicles," *ACM Trans. Sensor Netw.*, vol. 12, no. 2, 2016, Art. no. 14.
- [25] C. Lin, D. Han, J. Deng, and G. Wu, "P²S: A primary and passer-by scheduling algorithm for on-demand charging architecture in wireless rechargeable sensor networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 9, pp. 8047–8058, Sep. 2017.
- [26] W. Xu, W. Liang, X. Lin, and G. Mao, "Efficient scheduling of multiple mobile chargers for wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 9, pp. 7670–7683, Sep. 2016.
- [27] Y. Ma, W. Liang, and W. Xu, "Charging utility maximization in wireless rechargeable sensor networks by charging multiple sensors simultaneously," *IEEE/ACM Trans. Netw.*, vol. 26, no. 4, pp. 1591–1604, Aug. 2018.
- [28] N. Gharaei, Y. D. Al-Otaibi, S. A. Butt, S. J. Malebary, S. Rahim, and G. Sahar, "Energy-efficient tour optimization of wireless mobile chargers for rechargeable sensor networks," *IEEE Syst. J.*, to be published, doi: [10.1109/JSYST.2020.2968968](https://doi.org/10.1109/JSYST.2020.2968968).
- [29] L. Mo, A. Kritikakou, and S. He, "Energy-aware multiple mobile chargers coordination for wireless rechargeable sensor networks," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8202–8214, Oct. 2019.
- [30] L. Khelladi, D. Djenouri, M. Rossi, and N. Badache, "Efficient on-demand multi-node charging techniques for wireless sensor networks," *Comput. Commun.*, vol. 101, pp. 44–56, 2017.
- [31] A. Tomar, L. Muduli, and P. K. Jana, "An efficient scheduling scheme for on-demand mobile charging in wireless rechargeable sensor networks," *Pervasive Mobile Comput.*, vol. 59, 2019, Art. no. 101074.
- [32] J. Zhu, Y. Feng, M. Liu, G. Chen, and Y. Huang, "Adaptive online mobile charging for node failure avoidance in wireless rechargeable sensor networks," *Comput. Commun.*, vol. 126, pp. 28–37, 2018.
- [33] C. Lin *et al.*, "GTCCS: A game theoretical collaborative charging scheduling for on-demand charging architecture," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 124–1236, Dec. 2018.
- [34] K. E. Muller and B. A. Fetterman, *Regression and ANOVA: An Integrated Approach Using SAS Software*. Cary, NC, USA: SAS Institute, 2003.

► For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.