Adaptive Mobile Chargers Scheduling Scheme based on AHP-MCDM for WRSN

Kondwani Makanda, Ammar Hawbani, Xingfu Wang, Abdulbary Naji, Ahmed Al-Dubai, Liang Zhao, and Saeed Hamood Alsamhi

Abstract—Wireless Sensor Networks (WSNs) are used to sense and monitor physical conditions in various services and applications. However, there are a number of challenges in deploying WSNs, especially those pertaining to energy replenishment. Using the current solutions, when a significant number of sensors need to replenish their energy, this would be costly in terms of time, efforts and resources. Thus, this paper aims to solve this problem by efficiently deploying wireless power transfer technologies and scheduling Mobile Charging Vehicles (MCVs) in WRSN. The proposed method deploys multi-criteria decision-making (i.e., Analytical Hierarchy Process (AHP)) to schedule the charging tasks. To the best of our knowledge, this paper is the first to depend solely on AHP in MCVs scheduling. The paper demonstrates the validity of the proposed method by illustrating that the matrices that are created are within the accepted values of consistency ratio. In addition, the paper proposes a method of partitioning the values of our criteria to avoid the problem of different criteria having different measurement units. Unlike existing works, the paper aims to schedule an MCV for charging based on both the distance and residual energy of the sensor. The proposed method exhibits superiority in terms of the average remaining energy available in the system, having the shortest queue length, shorter MCV response time, shorter charging duration, and shorter queue waiting time against the state-of-the-art methods. Our study paves the way for next generation efficient charging and MCV scheduling.

Index Terms—Multi-criteria decision-making, Analytic hierarchy process, pairwise comparison matrix, scheduling scheme, Wireless sensor network;

1 Introduction

Having a perpetual network is one of the most ultimate goals of WSN [1]. In an ideal scenario, we aim to have a network that does not run out of energy, but in practice, that is unfortunately not the case. The energy of a sensor can run out due to depletion of the sensor's battery or lack of enough energy to replenish the exhausted sensors.

Mainly, there are two major ways to replenish the energy of a sensor: battery swapping and wireless power transfer (WPT).

- **Battery swapping** is the process of physically replacing an exhausted battery with a new one. This is a tiresome and costly process, especially when thousands of sensors need battery replacement. In some cases, it is not ideal when the sensors are in hard-to-reach or dangerous places, such as sensors embedded in infrastructure monitoring systems or rugged terrains like mountains [2].
- **Wireless power transfer** is a technology that enables energy to be transferred from a source to a sensor's battery wirelessly [3]. WPT technologies such as inductive or magnetic coupling allow power transfer from high voltage to low voltage and WPT technologies are more efficient, more convenient, and more scalable than battery swapping [4].

Developments and advances in WPT has given rise to ways of wirelessly charging different electrical devices. These devices include mobile phones and electric vehicles [5]. Sensors in wireless sensor networks can also be charged using WPT technologies. Mobile charging vehicles (MCV) are also used to recharge sensors, and these are used in combination with wireless power transfer technologies.

In addition to battery swapping and WPT, there is another way to replenish the energy of sensors: energy harvesting. Energy harvesting is the process of converting ambient energy into electrical energy that can be used to power electronic devices and keep sensors in operation [6]. There are a number of different energy harvesting technologies, including: solar, wind, and vibrations. These energy sources are usually available in the vicinity of the sensor [7]. Therefore, the energy harvester should be within this vicinity [8]. However, the energy from these sources may not be available all the time [9]. For example, solar energy is not available at night, and wind energy is not available during calm weather. In contrast, WPT is a more reliable and versatile way to power sensors since WPT can be used to power sensors in a variety of environments, regardless of the weather conditions. In contrast, the advantages of WPT include being reliable, safe, versatile, dependable and also being able to overcome problems of unpredictable weather conditions and harsh terrain.

In an environment where resources are limited, it is important to ensure that these resources are used efficiently [10].

K. Makanda, W. Xingfu, A. Naji are with the School of Computer Science and Technology in the University of Science and Technology of China, China, Anhui, Hefei, 230026. E-mail: kmakanda@mail.ustc.edu.cn, wangxfu@ustc.edu.cn, abdulbary@mail.ustc.edu.cn;

Ammar Hawbani and Liang Zhao are with the school of computer science, Shenyang Aerospace University, Shenyang 110136, China. E-mail: anmande@ustc.edu.cn;lzhao@sau.edu.cn;

Ahmed Al-Dubai is with the computing school of Edinburgh Napier University, United Kingdom. E-mail: a.al-dubai@napier.ac.uk;

Saeed H. A. is with Insight Centre for Data Analytics, University of Galway, Galway, Ireland, and Faculty of Engineering, IBB University, Ibb, Yemen. E-mail: saeed.alsamhi@insight-centre.org;
Ammar Hawbani and Xingfu Wang are the corresponding authors.

Resource scheduling is a technique that helps to ensure that resources are used fairly and efficiently and that resource starvation is avoided. The resources that are being used should guarantee the quality of the services being provided [11].

Two critical resources considered in this paper are spatial and residual charging energy. Spatial resources refer to the distance between sensors and the MCV. As the MCV travels through the network, the distance between the MCV and the sensors changes. This makes the spatial resource dynamic, and it determines the amount of energy the MCV will use to travel to the sensors. The shorter the distance, the less energy the MCV will need to consume to travel to the sensor. Residual energy is the amount of energy that is left in a sensor's battery. The residual energy of a sensor affects how many sensors can be charged by the MCV. Allocating these resources efficiently will result in better network performance.

In this paper, we propose a method for selecting a sensor to be charged based on both the distance and residual energy of the sensor. The proposed method uses the AHP taking into account the different criteria that affect the operations of the sensor. Based on the overall operations of the sensors, weights of each criterion are generated. These weights are then used to select the sensor that should be charged. The use of multiple criteria prevents reliance on only one criterion (mostly residual energy of sensor) as the only criteria to select a sensor for charging. This is because different criteria may be more important in different scenarios. For example, the distance between the sensor and the MCV may be more important in a situation where the MCV has limited travel energy, while the residual energy of the sensor may be more important in a situation where the MCV has plenty of energy.

To avoid confusion that due to the criteria use different units, we propose a method of partitioning our criteria into different ranges that we can fit in the values of the AHP fundamental scale. This ensures that all criteria are on a comparable scale, and that the weights of the criteria are meaningful.

1.1 Motivation

This study is motivated by the following shortcomings found in state-of-the-art solutions for WRSN MCV scheduling.

There is lack of research that purely uses AHP MCDM in WSN. To the best of our knowledge this is the first study that only uses AHP in MCV scheduling. Other methods that we have found use this method in conjunction with other methods like using TOPSIS and fuzzy AHP.

The use of only AHP was necessitated by the following reasons: computational power of sensors is mostly limited and as such they are not capable of handling very complex and huge models or methods. For real-world implementation we need to reduce computational power to the bare minimum that can be handled by the nodes. This lead us to simplify the process and only use AHP to minimize the computation complexity and power usage of the sensors.

Data Aggregation: efficiently combine data from multiple sensors before transmission, reducing communication overhead. Having a few criteria transmitted within the network will also preserve sen power. That is why we considered the two criteria which have a huge impact on sensor charging decision. The other reason is that to reduce latency, few criteria need to be used as smaller data packets are efficient in minimizing delays in data transmission.

Light algorithms and methods also help to improve network performance. This is done by having faster response times. Light algorithms will also help in quick data processing and communication thereby enabling real-time applications. They also enhance reliability, i.e., lower complexity makes the algorithms less susceptible to errors thereby improving data integrity.

WSN are inherently limited in computation resources and introducing more complexity by addition of other methods makes it infeasible to implement the methods in the real world environment.

1.2 Contributions

This study designs an MCV scheduling scheme that takes advantage of two fundamental properties of sensors: namely, distance and residual energy. These two properties are jointly used in AHP to determine priorities for selecting a high-priority sensor for charging. The main contributions of this paper are summarized as follows:

- The paper studies charging scheduling, formulating the problem using the AHP in WRSN mobile charging vehicle scheduling. This is different from the existing works that deploy fuzzy logic in dealing with multiple criteria decision-making problems.
- We develop a greedy algorithm for partitioning all the real values of distance and residual energy into AHP fundamental scale. Since we will generate real residual energy and distance values, we present a method that describes how these values can be partitioned into the corresponding AHP fundamental values.
- Our experiments and results indicate that the proposed scheme outperforms other existing counterparts in terms of the average remaining energy available in the network, reduction of queue length, reduction in response time, and waiting queue time, paving the way towards for more effective scheduling and charging policies in WRSN.

The paper is organized as follows; in Section 2 we present Related Works. Section 3 discusses the System Description. Multi-criteria Decision Making related analysis is presented in Section 4. The proposed scheme is discussed in Section 5. The simulation environment is addressed in Section 6. In Section 6.3, we present the Experimental Results and Analysis, and lastly, Section 7 presents the Conclusion.

2 Related Works

Recently, a lot of research in WRSN has been conducted. Clearly, the research in this area is still in its infancy. The exisiting works focus on improving the performance of WRSNs. The main objective of most studies is to maximize the network lifetime and keep the nodes alive as long as possible. To achieve this, different charging scheduling schemes are utilized. These charging schemes are broadly classified into deterministic and nondeterministic, offline path planning and online path planning [12] or period charging and on-demand charging [13]. A comprehensive list of on-demand charging methods can be found in [14]. These details related to whether the charging decision is centralized or distributed, fully or partially charging of sensors, using single or multiple MCVs etc. In a similar manner, [14], studied an on-demand sensor charging by taking into consideration the benefits of partial charging. They also studied the distribution control in large-scale wireless rechargeable sensor networks, and how heterogeneity the energy in the sensors are. To achieve this, they used a game theory approach as a 0-1 integer linear programming in order to maximize the profit of MCVs.

In [15], they proposed a deep reinforcement learning-based mobile charger scheduling method, namely, dynamic partial mobile charger scheduling using deep-Q-networks. This method learns from the network environment's dynamics and decides the charging duration of each sensor in the MCV's trajectory. Using deep-Q-networks, they solved the MCV's schedule in order to find the best sensor to charge as well as the path that the MCV has to take. The aim of this method is to reduce the waiting time of the sensor for mobile charging vehicles and resulting in minimizing the number of dead sensors. The limitation of this method is the increase in the distance traveled by the MCV.

In [16], they proposed a charger scheduling method that uses residual energy, distance to MCV, energy consumption rate, and neighborhood energy priority to schedule an MCV. In this method, they used limited energy in MCVs and the sensor's energy consumption when scheduling the MCVs. The shortfalls in this literature include, they did not have the step-by-step method they used to create the pairwise matrices. Similarly, they did not show how the priorities of individual sensors are derived at. The other weakness that we found is that the way the consistency ratio was calculated was not shown, apart from mentioning that they used processes similar to [17]. The last shortfall is that this method is only applicable to a single MCV.

In [18], they proposed the Nearest-Job-Next with Preemption (NJNP). This is based on on-demand mobile charging in which an individual sensor requests charging from the mobile charger. This method was evaluated based on throughput and charging latency. The proposed method achieves the best performance in both throughput and charging latency but has drawbacks which include having a starvation problem. This is where nodes that are far away from the MCV may not be able to get the energy that they deserve. This problem can be solved by minimizing the distance covered by the MCV, which can help improve the charging utility of the network [19]. The other problem with this method is that as the number of sensors in the network increase, the charging latency and inactive time ratio of sensors increase due to the heavy workload exerted on the MCV.

In [20], they proposed an online multi MCV charging scheme. In this method, MCVs collaborate in the charging schedule planning. This method jointly uses temporal and spatial data to make a charging decision. Sensors that have the shortest distance and closer charging deadlines are the ones that have the highest priority. This method achieves a higher survival rate, throughput, queue length, and charging rate. Unfortunately, the method suffers from having a higher communication overhead since when a sensor is not charged by an MCV, it sends a recharge request to a neighbouring MCV. Sending frequent request will result in higher energy consumption as communication is one of largest energy consumers in wireless sensor networks [21].

Other related works have been added in the supplemental file.

In this work, we focus on selecting a sensor to be charged based on multiple criteria. This will schedule an MCV to visit a sensor once the defined criteria have been met. Our proposal involves dynamic charging where the next to-be-charged sensor depends on the priority of the current attributes of the sensor. The proposed method is also a one-to-one charging where a single MCV charges one sensor at a time. Another characteristic of the proposed method is that, the MCV will fully charge a sensor before proceeding to the next one. Multiple MCVs, will be used as opposed to using a single MCV.

3 SYSTEM DESCRIPTION

Given a wireless sensor network S, where $S = \{s_1, s_2, \ldots, s_n\}$ denotes a set of sensors deployed in a field of interest for sensing and collecting data. Each node has a rechargeable battery with E_{cap} capacity. The network is heterogeneous, i.e., the network uses two different types of sensors, these are temperature, and humidity sensors. Each sensor has a different energy consumption rate $e_{con_{s_i}}$. $e_{res_{s_i}}$ is the residual energy of sensor s_i. τ_{lt} is the amount of time a sensor is expected to continue operating before it fails. This is dependent on the amount of energy remaining in the sensor and the energy consumption rate of the sensor. This is given as $\tau_{lt} = e_{res_{s_i}}/e_{con_{s_i}}$ [20].

When e_{res} is equal to or below a given threshold value, i.e. $e_{res} \leq e_{thre}$, a charge request is sent to the base station (BS). The BS as a sink collects the sensor requests in a recharge request set or queue S_{req}. The BS acts as a central hub that has complete knowledge of the network, i.e., the BS will be used to perform a number of functions. These functions include: recharging the MCVs, performing computations for scheduling MCVs for sensor recharging tours, providing communication between sensors and the MCVs and also communication between MCVs and BS, acting as a gateway between the whole system and users, and storing data collected from the sensors. It will also keep track of which MCV has been assigned to which sensor.

The distance between two sensors s_i and s_j is given by the Euclidean distance $d(s_i, s_j)$. In this paper, a scheduling algorithm based on the multi-criteria decision is utilized where $1 < |S_{req}| \leq |S|$. The criteria are the sensor's residual energy and distance. In this paper, we will also briefly address the attributes that the MCV has that might also influence the sensor to be charged. Notations used in this paper are given in Table 1.

3.1 MCV Properties

The MCV has two important attributes that should be studied. These attributes are: the MCV travel energy ($E_{MCV_{trav}}$) and MCV residual charging energy ($E_{MCV_{res}}$). $E_{MCV_{res}}$ is the energy that is used to charge sensors. The MCV residual charging energy and traveling energies are separate; that is, the MCV will have two energy sources, one battery for charging sensors and another battery to be used as energy

Table 1: Notations and their meaning.

Notation	Meaning
S	A set of all sensors in the network
S_{req}	Set of sensors requesting recharge
S_{sub}	Subset of S_{req}
$E_{cap_{s_i}}$	Energy capacity of sensor s_i
$e_{con_{s_i}}$	Energy consumption rate of sensor s_i
$e_{res_{s_i}}$	Residual energy of sensor s_i
$d(s_i, s_j)$	Euclidean distance between sensors s_i and s_j
τ	Amount of time the sensor has been in operation
τ_{lt}	Sensor lifetime
τ_{s_i}	The time sensor s_i has been operating
τ_{total}	Total time spent by MCV charging sensors
S_{dead}	The number of dead sensors in a period of time
e_{charg}	The charging rate of the sensor
$\tau_{charg_{s_i}}$	The charging time of sensor s_i
v_{MCV}	The speed of the MCV
τ_{MCV}	Time of MCV to travel and charge sensors
$e_{con_{MCV_t}}$	Energy consumption rate of MCV at time t
$E_{MCV_{init}}$	MCV initial energy used for charging sensors
$E_{MCV_{trav}}$	MCV traveling energy
$E_{MCV_{res}}$	Residual charging energy of MCV for charging sensors

for traveling. These two energy sources determine whether a sensor gets charged or not. Three scenarios can force an MCV not to charge a sensor, (1) if $E_{MCV_{trav}}$ is less than the energy needed to reach a sensor and return to the BS, (2) if $E_{MCV_{res}}$ is less than the energy needed to charge a sensor, and (3) when the MCV has finished recharging sensors.

3.1.1 MCV Travel Energy

For the MCV to travel around its charging tour, it needs energy. If this energy is unavailable or insufficient for the MCV to recharge the sensors and return to the BS, the MCV will not travel. Given a charging route $\{BS \xrightarrow[e_1]{} s_1 \xrightarrow[e_2]{} s_2 \xrightarrow[e_3]{} \cdots \xrightarrow[e_n]{} s_n \xrightarrow[e_{n+1}]{} BS\}$ with corresponding energies e_i needed to travel between sensors, then the minimum MCV energy required to travel around the charging tour is given by Equation (1).

$$E_{MCV_{trav}} \geq \sum_{i=1}^{n+1} e_i \tag{1}$$

For $s_i \in S_{req}$, if $E_{MCV_{trav}} < \sum_{i=1}^{n+1} e_i$, then the MCV will not travel. The time the MCV has been moving from one sensor to another and the time it is charging the sensors. τ_{MCV} is defined by Equation (2) [19].

$$\tau_{MCV} = \sum_{\substack{i,j=1 \\ i \neq j}}^{n} \frac{d(s_i, s_j)}{v_{MCV}} + \sum_{i=1}^{n} \tau_{s_i}. \tag{2}$$

where v_{MCV} denotes the speed of the MCV in meters per second, and τ_{s_i} denotes the time taken to charge sensor s_i in seconds.

3.1.2 MCV Residual Charging Energy

The residual energy of the mobile charger also plays an important role in the sensor charging decision problem. The residual energy of the MCV will have a profound impact in terms of whether a sensor is charged or not. In a very large network, the MCV will charge sensors up to a point where its energy level allows it to, that is if $E_{MCV_{res}} \geq (E_{cap_{s_i}} - e_{res_{s_i}})$. After that point, the MCV will have to return back to the BS to replenish its energy, even if some sensors are still waiting to be charged.

At the onset of the network, the residual charging energy of MCV ($E_{MCV_{res}}$) is given by Equation (3) [16].

$$E_{MCV_{res}} = E_{MCV_{init}} - \left(\sum_{t=1}^{n} \left(e_{con_{s_i}} * \tau_{s_i} \right) \right) \tag{3}$$

where $e_{con_{s_i}}$ is the energy consumption rate of sensor s_i, and τ_{s_i} is the time sensor s_i has been in operation. If no sensor requests charging then $E_{MCV_{res}} = E_{MCV_{init}}$.

3.2 Sensor Properties

Sensors also have attributes that are of interest and highly influence the charging scheme design. There are several attributes or criteria that are important, and these are the distance of the sensor from the MCV, residual energy of the sensor, sensor charging time, and lastly, sensor energy consumption rate. The two attributes that are of interest in this paper are residual energy and the distance of the sensor to the MCV. A brief discussion of the four attributes and how they affect the MCV's scheduling is discussed in detail.

3.2.1 Distance between sensors

The euclidean distance between two sensors s_i and s_j is given as $d(s_i, s_j) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. The aim is to find the shortest distance covered by the MCV as it is traveling charging the sensors [22], maximizing the number of nodes charged per every MCV stop, and setting up of MCV energy replenishment depots along the MCV route, i.e., the distance between two sensors [23]. This problem has a computational complexity of $\mathcal{O}(n!)$ for n locations visited [24]. The total distance T_d covered by the MCV from BS to all the sensors in its charging path is given by Equation (4), and the total distance back to the base station is $T_d + d(s_n, BS)$ where T_d is given in Equation 4.

$$T_d = \sum_{i=0}^{n} d(BS, s_i) \tag{4}$$

3.2.2 Sensor Residual Energy

The residual energy of a sensor s_i given by $e_{res_{s_i}}$ is the remaining energy after a sensor has been in operation after τ seconds where τ_{s_i} is the time the sensor s_i has been operating. The sensor uses its energy to sense its environment, communication and computing tasks [25]. The $e_{res_{s_i}}$ is calculated by Equation (5), where $E_{cap_{s_i}}$ is the energy capacity of sensor s_i.

$$e_{res_{s_i}} = E_{cap_{s_i}} - \left(\sum_{t=1}^{n} \left(e_{con_{s_i}} * \tau_{s_i} \right) \right) \tag{5}$$

3.2.3 Sensor Charging Duration

Sensor charging time is the amount of time it takes the MCV to charge a sensor. This will ultimately determine how much time a mobile charger will spend on a particular sensor. The more time the MCV spends on a particular sensor, the less time it will be able to charge other sensors. This can result in other sensors not getting charged. These sensors will die, which in turn disrupts the operation of the whole network. The charging time $\tau_{charg_{s_i}}$ of sensor s_i is given by Equation

(6), where $E_{cap_{s_i}}$ denotes the battery capacity of the sensor s_i when full, and e_{charg} denotes the charging rate.

$$\tau_{charg_{s_i}} = \frac{(E_{cap_{s_i}} - e_{res_{s_i}})}{e_{charg}} \tag{6}$$

For sensors not to die, we need to make sure that $e_{charg} \geq e_{con_{s_i}}$. The total time τ_{total} spent by the mobile charger to charge all the sensors in a charge request queue is given by Equation (7).

$$\tau_{total} = \sum_{i=1}^{|S_{req}|} \tau_{charg_{s_i}} \tag{7}$$

3.2.4 Charging Rate

This is the rate the sensor is being charged given by e_{charg}. For a sensor to be fully charged, we need to make sure that $e_{charg} \geq e_{con_{s_i}}$. If $e_{charg} < e_{con_{s_i}}$, then the sensor will exhaust its residual energy and die.

4 Multi-criteria Decision Making (MCDM)

MCDM is a process used to solve problems with multiple criteria, such as charging or scheduling schemes, by aggregating the criteria to determine the best option. In WSN, selecting which sensor to charge is a challenge that involves considering multiple criteria to determine the optimal choice [5].

Many researchers use residual energy as the sole metric for determining which sensor to charge [26]. Once sensors have sent a recharge request to the BS, a recharge request set is created, and a charging route is planned, which represents a charger scheduling and path planning problem [19].

Rather than relying on a single criterion to select a sensor for charging, this paper considers multiple criteria and aggregates the scheduling decision to determine which sensor should be recharged first. Specifically, the system takes into account the distance between the MCV and the sensor as well as the sensor's residual energy. Reasons for selecting these two criteria are explained next.

Residual energy was selected for the following reasons:

- **Network lifetime:** Network lifetime determines the amount of time the network can operate before batteries run out of power. Therefore, energy consumption is a major factor in determining the network lifetime.
- **Performance:** Energy affects performance because, having no energy, sensors will not be able to transmit data, process data, sensing, and communication. Once sensors are not able to perform these functions, then the performance of the network will be negatively affected.

Distance was selected as a criterion for the following reasons:

- **Energy consumption:** If two sensors are further apart, then they will use more energy to communicate between them. This is because the signal travels a much greater distance, which means the signal has a much higher likelihood of being attenuated or lost.
- **Data rate:** Two sensors located further apart have a lower data rate they can support. The reason for this is that the signal is spread over a wider area which means there is less bandwidth available for each sensor. For efficient data transmission, dynamic data routing is better than static routing in WSN [27].
- **Latency:** The further apart the sensors are, the higher the latency between them. The reason is that it will take a long time for the signal to travel between the two nodes.
- **Reliability:** Sensors that are located further apart have less reliable communication between them. This is because there is a greater chance of the signal being lost or attenuated.

For these reasons, we decided to choose residual energy and distance as criteria that we need to use in the proposed method. The proposed AHP MCDM hierarchy system is described as follows:

- **Alternatives**: $S = \{s_1, s_2, \cdots, s_n\}$ a set of n potential alternatives (sensors) to be selected for charging.
- **Criteria**: A set of criteria $A = \{a_1, a_2, \}$ where a_1 is the Euclidean distance between a sensor and the MCV, and a_2 is the residual energy.
- **Goal**: the aggregation of performance of alternatives s_i on criterion a_j is given by $g_j(a_i)$ $\forall$ $s_i \in S$ and $a_i \in A$.

4.1 Analytic Hierarchy Process

AHP is a structured technique for organizing, analyzing and making complex decisions [28]. These hierarchies help one make judgments about complex problems [29]. AHP allows one to transform a problem having different scales and units like meters and kilograms to a single scale problem by making sure that the criteria are independent of each other [29].

Given a set of n alternatives or sensors $S = \{s_1, s_2, \ldots, s_n\}$. We need to make a decision as to which sensor will be selected for charging based on several criteria (m) where m is a finite set. The finite set S contains our alternatives, i.e., which sensor will be selected for charging. The set of objectives will be limited to a maximum of three sensors, i.e., $|S_{req}| \leq 3$ as 7 is the optimal value [30]. For a criterion k, the AHP generates a weight ω_k. For an alternative j, a score s_{kj} for objective k is obtained.

The total score of the alternative j is aggregated by Equation (8).

$$\sum_{k=1}^{m} \sum_{j=1}^{n} \omega_k s_{kj} \tag{8}$$

Based on the total score, a decision will be made as to which sensor should be selected for charging; that is, a sensor with the highest score will be selected, or a sensor with the lowest score, or any selection criteria can be applied.

The AHP involves a pairwise comparison matrix where objectives are compared with each other [31]. In our case, the criteria that will be used are the distance between the sensors and the MCV and the sensor's residual energy. These criteria are the ones that are used to evaluate alternatives.

Since in AHP, the goal is defined at the highest level of the hierarchy, in this case, the goal is defined as "being able to select a sensor to charge based on multiple criteria or MCV charging scheduling." The alternatives, in this case, are the sensors at the lowest level of the hierarchy.

Figure 1 shows the hierarchy used in making decisions in this paper, where the highest level gives the goal. The objectives are on the second level and lastly, the alternatives which are the sensors.

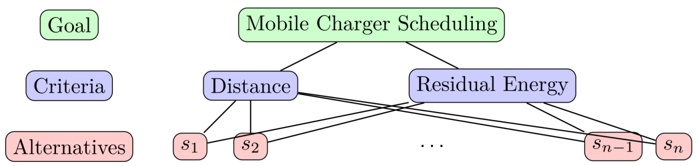


Figure 1: AHP Hierarchy

The hierarchies influence each other; from top to bottom. The lowest level has alternatives; these are influenced by the level immediately above them. In Figure 1 having three levels, the higher level influences on the middle level with weight of ω matrix and middle level influences on the lower level is the score matrix S. This, therefore, means the influence of the higher level on the lower level is the matrix multiplication $S\omega$.

Based on AHP, we will use the method outlined in [32]. After a positive reciprocal pairwise matrix ($A = [a_{i,j}]_{nxn}$) has been created, we need to generate priorities for the alternatives and criteria.

1) Sum the values in each column of the pairwise matrix A as in Equation (9).

$$A_{ij} = \sum_{i=1}^{n} a_{ij} \tag{9}$$

2) Generate a normalized pairwise matrix by dividing each element of A by the sum of its column as given in Equation (10).

$$\begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1n} \\ q_{21} & q_{22} & \cdots & q_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ q_{n1} & q_{n2} & \cdots & q_{nn} \end{bmatrix}, Q_{ij} = \frac{A_{ij}}{\sum_{i=1}^{n} A_{ij}} \tag{10}$$

3) To generate the weighted matrix, divide the sum of the normalized column of the matrix by the number of the criteria used (n in our case). This is given in Equation (11).

$$\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}, W_{ij} = \frac{\sum_{j=1}^{n} Q_{ij}}{n} \tag{11}$$

Equations (9), (10) and (11) allow the generation of a normalized matrix of priorities.

After generating the weight vectors, the next stage involves consistency analysis. This helps in determining whether our matrix A is consistent or not. The steps involved are as follows:

1) Generate a consistent vector by right multiplying the pairwise A with the weight vector as in Equations (12).

$$\begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1n} \\ q_{21} & q_{22} & \cdots & q_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ q_{n1} & q_{n2} & \cdots & q_{nn} \end{bmatrix} \begin{bmatrix} w_{11} \\ w_{21} \\ \vdots \\ w_{n1} \end{bmatrix} = \begin{bmatrix} qv_{11} \\ qv_{21} \\ \vdots \\ qv_{n1} \end{bmatrix} \tag{12}$$

2) Divide the weighted sum vector with each criterion weight. This is given in Equation (13).

$$\begin{aligned} qv_{11} &= \frac{1}{w_{11}} \left[\begin{array}{cccc} q_{11}w_{11}+ & q_{12}w_{21}+ & \cdots + & q_{1n}w_{n1} \end{array} \right] \\ qv_{21} &= \frac{1}{w_{21}} \left[\begin{array}{cccc} q_{21}w_{11}+ & q_{12}w_{21}+ & \cdots + & q_{1n}w_{n1} \end{array} \right] \\ & \quad\;\; \vdots \qquad\qquad \vdots \qquad\qquad \ddots \qquad\qquad \vdots \\ qv_{n1} &= \frac{1}{w_{n1}} \left[\begin{array}{cccc} q_{n1}w_{11}+ & q_{12}w_{21}+ & \cdots + & q_{1n}w_{n1} \end{array} \right] \end{aligned} \tag{13}$$

3) Calculate the largest eigenvalue of matrix A i.e., λ_{max} is calculated as in Equation (14).

$$\lambda_{max} = \sum_{i=1}^{n} qv_{ij} \tag{14}$$

4) Consistency Index $(C.I.)$ is calculated using Equation (15), where n is the eigenvalue of matrix A.

$$C.I. = \frac{\lambda_{max} - n}{n - 1} \tag{15}$$

5) Calculate the Consistency Ratio $(C.R.)$ which is the ratio of $C.I.$ to $R.I.$ i.e., using Equation (16), where $R.I.$ is the value obtained from Table 2 [33]. If $C.R. < 0.10$, the matrix is considered consistent otherwise, it's not consistent.

$$C.R. = \frac{C.I.}{R.I.}. \tag{16}$$

5 Proposed Scheme

This section discusses the proposed method and its divided into two sub-sections. Sub-section 5.1 presents a brief description of the energy utilization of MCV and sensors while 5.2 discusses the process of creating pairwise comparison matrices.

5.1 MCV and Sensor Energy Utilization

When a sensor's residual energy is below a given threshold, that particular sensor will be added to the charge request queue S_{req}. The sensor distance to the MCV will then be calculated. Based on these two properties, comparison matrices will be created. These will be used in calculating the priorities of each sensor. Using Equations (9) through (15), a sensor to be charged will be selected based on highest priority.

In literature, most authors assume that the MCV has unlimited energy. In reality, the MCV also has limited energy to travel and the energy it uses to charge the sensors. Since the energy of the MCV is also limited, it means that if the travel energy of the MCV is less than the energy needed to

Number of Criteria (n)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Random Consistency Indices (R.I.)	0.0	0.0	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49	1.51	1.54	1.56	1.57	1.59

Table 2: Average Random Consistency Index (R.I.)

reach a particular sensor, charge the sensor and return to the BS. The sensor will either not be recharged, and the MCV will also die on the way or, the sensor will be able to be recharged by the MCV, but the MCV will not be able to return to the BS for itself to be recharged. Either way, the MCV will die, which will disrupt the whole network. When the MCV is stationary charging the sensor, the travel energy will be zero. If the MCV residual energy is less than the energy needed to recharge a sensor, then the MCV will not be able to travel and recharge the sensors, i.e., the two attributes, initial or residual energy ($E_{MCV_{init}}$) and travel energy ($E_{MCV_{trav}}$) are important in determining whether the sensor is charged or not.

Given binary value 0 representing ”no adequate energy” and 1 representing ”adequate energy”. An MCV can only start a charging tour if both values are 1. This is similar to a binary AND operation as shown in Table 3.

Table 3: MCV travel decision

$E_{MCV_{init}}$	$E_{MCV_{trav}}$	$(E_{MCV_{init}} \wedge E_{MCV_{trav}})$/Recharge?
0	0	No
0	1	No
1	0	No
1	1	Yes

The minimum residual charging energy of the MCV, before it starts its journey, is given by Equation (3)

so that the MCV can have enough energy to charge all the sensors in S_{req}. Equation (1) gives the minimum travel energy. The minimal travel energy is the energy needed for the MCV to travel around the charging tour.

This will result in the MCV recharging all the sensors and returning to the BS.

5.2 Pairwise Comparison Matrices

The pairwise comparison matrix is made by comparing the criteria with each other and also the alternatives at the lowest level with each other. After creating a pairwise comparison matrix, we need to make sure that the matrix is consistent and reciprocal. A matrix $\mathcal{X}$ is said to be consistent if for $a_{im} \in \mathcal{X}$ and $a_{mj} \in \mathcal{X}$ then $a_{im} \cdot a_{mj} = a_{ij} \in \mathcal{X}$ otherwise the matrix is not consistent [34]. a_{ij} is the relative preference of the criterion i to j. When the matrix is inconsistent, the comparison must be revised until it becomes consistent, although this may not be successful. For the matrix $\mathcal{X}$ to be reciprocal then the elements $a_{ij} = 1/a_{ji} \rightarrow a_{ij}a_{ji} = 1$ and that the diagonal matrix is 1 i.e. $a_{ii} = 1$ since $a_{ij}a_{ji} = 1$.

For n criteria, we have $\frac{n(n-1)}{2}$ comparisons excluding reciprocal values and self-comparison criteria. Based on this, we should be able to create matrices for distance and residual energy.

5.2.1 *Distance Pairwise Comparison Matrix*

To create a distance pairwise comparison matrix D, we measure the euclidean distance from MCV to sensors. For example given two sensors s_i and s_j, if $d(MCV, s_i) < d(MCV, s_j)$ then we assign the value $d(MCV, s_i)$ on the row s_{ij} and the reciprocal $1/d(MCV, s_i)$ on the row s_{ji}. If $d(MCV, s_i) > d(MCV, s_j)$, then on row s_{ij} will have the value $1/d(MCV, s_i)$ and $d(MCV, s_j)$ will be allocated at s_{ji}. Otherwise, if $d(MCV, s_i) = d(MCV, s_j)$ then $s_{ij} = s_{ji} = 1$. The values assigned to the rows s_{ij} or s_{ji} are taken from Table 4 based on the scale created.

Algorithm 1 Distance Pairwise Comparison Matrix Generator

```
Input : Sensor Distance from in S_req
Output : Distance Pairwise Comparison Matrix
if d(MCV, s_i) < d(MCV, s_j) then
    s_ij ← d(MCV, s_i)
    s_ji ← 1/d(MCV, s_i)
else if d(MCV, s_i) > d(MCV, s_j) then
    s_ij ← 1/d(MCV, s_i)
    s_ji ← d(MCV, s_i)
else
    s_ij ← s_ji ← 1
end if
```

The priority of sensors based on distance is calculated using Equations 9 through 11 and consistency ratio is calculated using Equations 12 through 15. The time complexity of Algorithm 1 is provided in the Supplementary file.

5.2.2 *Residual Energy Comparison Matrix*

When creating a comparison matrix based on residual energy, we need to make a comparison based on the current residual energy of the compared sensors. Given two sensors s_i and s_j, if $e_{res_{s_i}} > e_{res_{s_j}}$ then $s_{ij} = 1/e_{res_{s_i}}$. If $e_{res_{s_i}} < e_{res_{s_j}}$, then $s_{ij} = e_{res_{s_i}}$ and $s_{ji} = 1/e_{res_{s_i}}$ and lastly if $e_{res_{s_i}} = e_{res_{s_j}}$ then $s_{ij} = s_{ji} = 1$. This means that the higher the residual energy value, the lower the value that will be assigned from Table 4. Similarly, the lower the residual energy, the higher the value to be assigned from Table 4. The values $e_{res_{s_i}}$ and $e_{res_{s_j}}$ will be replaced with values from the modified fundamental scale of absolute numbers from Table 4.

Algorithm 2 Residual Energy Pairwise Comparison Matrix Generator

```
Input : Residual energy from sensors in S_q
Output : Residual Energy Pairwise Comparison Matrix
if e_res_si > e_res_sj then
    s_ij ← 1/e_res_si
    s_ji ← e_res_si
else if e_res_si > e_res_sj then
    s_ij ← 1/e_res_si
    s_ji ← 1/e_res_si
else
    s_ij ← s_ji ← 1
end if
```

The time complexity of Algorithm 2 is provided in the Supplementary file.

The priority of sensors based on residual energy is calculated using Equations 9 through 11 and consistency ratio is calculated using Equations 12 through 15.

5.2.3 Weights of Criterion and Alternatives

The importance of the criteria should be evaluated relative to the importance of the other criteria. This is done by normalization. Normalization will show the relative importance of the criterion [31].

After obtaining the weights of the alternatives using Equations (9) through (11), we will use the distributive mode of AHP to obtain the weights of the criteria. The distributive mode of AHP takes into account the fact that the weights of the criteria may depend on the alternatives. Alternatively, we can derive the weights of the criteria from the alternatives [31].

From Equation (11), weight for criterion i with respect to the goal is given by Equation (17).

$$\omega_i = \sum_{i=1}^{n} w_i. \tag{17}$$

The weight of the alternative with respect to the overall goal is the average of its weights with respect to the criteria given by Equation (18).

$$\frac{\sum_{i=1}^{n} \omega_i}{n}. \tag{18}$$

The sum of the weights $(\mathcal{W})$ of all criteria will be as in Equation (19).

$$\mathcal{W} = \sum_{i=1}^{n} \omega_i = 1. \tag{19}$$

5.2.4 Partition criteria values into corresponding AHP values

Given the closed interval of the values of our criteria as $[\alpha, \beta]$ where $\alpha < \beta$. The interval $[\alpha, \beta]$ will be partitioned into ζ equal partitions by Equation 20:

$$f(\alpha, \beta) = \frac{\beta - \alpha}{\zeta}. \tag{20}$$

We need to develop a method of partitioning the criteria into finite intervals to fit the AHP fundamental scale. Since all the attributes will have definite values, then these values can be put in an interval $[a, b]$ where $a < b$. The partition of this interval $[a, b]$ is given as sub-intervals

$$[x_0, x_1], [x_2, x_3], \cdots [x_{n-1}, x_n] \tag{21}$$

such that

$$a = x_0 < x_1 < x_2 < \cdots < n_{n-1} < x_n = b. \tag{22}$$

The associated AHP values are the tags of our interval. The interval will be $\sigma = \lceil \frac{\beta - \alpha}{n} \rceil$. We set $\alpha = 0 \implies \sigma = \lceil \frac{\beta}{n} \rceil$ because for all our criteria, the smallest value they can have is 0 and $\beta = max\{d(BS, s_1), d(BS, s_2), \cdots, d(BS, s_n)\}$. In our case, $n = 9$, since the AHP scale is divided into 9 intervals and β is the largest value of the criterion, we are fitting in the AHP fundamental scale interval. For example, given six sensors having the following distances (meters) from MCV, $\{40, 90, 124, 71, 5, 6\}$ from the MCV. Then the interval will be divided into sizes of $f(x) = \lceil \frac{124-0}{9} \rceil = 14m$ where $\beta = 124$ and $\alpha = 0$. The ceiling is used to make ensure that all the values in the range are included. The interval created will be dynamic. As the values of criteria change, so is the interval. This will enable every sensor to be allocated to a particular interval. This will result in a sensor being assigned a value in AHP fundamental scale. Partitions will also help in removing the units that our different criteria have.

To determine the priority of distance and residual energy, 16 comparison matrices regarding distance and residual energy can be generated. This will be helpful in determining the overall priority of a particular sensor concerning the goal. Due to a lack of expert knowledge about whether residual energy or distance should be given higher priority, in this paper, we gave distance and residual energy equal priorities, i.e., each has a priority of 50%. This will result in the overall priorities of the sensors being determined by the sensor's actual residual energy and distance at the particular time the priorities are being calculated.

The selection of a sensor to be charged works as follows, using AHP, each sensor sends its residual energy and distance to the BS station. Based on the data, the BS uses Algorithm 1 to create a distance comparison matrix. This will be used to get the sensor with the highest priority by only considering the distance. It also uses Algorithm 2 to generate a residual energy comparison matrix. The comparison matrix will then be used to calculate the sensor with the highest priority due to residual energy.

After creating the comparison matrices, Equation (9) through Equation (14) will be used to calculate the priorities. These priorities are then multiplied by the two criteria priorities and then added to find the overall priority in comparison with the overall goal. Equation (15) is used to calculate the consistency index as a component in the calculation of consistency ratio in Equation (16). To ensure that the consistency ratio is less than 10% for the results to be acceptable, a secondary queue will be created that only stores a maximum of three sensor attributes. This queue $S_{sub} \subset S_{req}$ and $|S_{sub}| \leq |S_{req}|$. A comparison matrix based on these three sensor attributes will be created and compared to reduce inconsistencies.

Consistency Ratio Challenge: When calculating our priorities, we need to make sure that the consistency ratio is within the required threshold of less than 10%. If the value is not less than 10%, the judgments should be revised [29]. In cases where we have one or two sensors in the queue, the consistency ratio will be within the required threshold. In cases of three or more sensors in S_{sub} (in our case, we will only consider three sensors to allow us to get the correct consistent ratio), then a systematic method needs to be developed that will allow us to get the necessary results. To make sure that we have the right consistency ratio whenever we create our comparison matrices, we propose Algorithm 3.

AHP Scale	1	2	$\cdots$	6	7	8	9
Distance	$[\beta, 8f(\alpha,\beta))$	$[8f(\alpha,\beta), 7f(\alpha,\beta))$	$\cdots$	$[4f(\alpha,\beta), 3f(\alpha,\beta))$	$[3f(\alpha,\beta), 2f(\alpha,\beta))$	$[2f(\alpha,\beta), f(\alpha,\beta))$	$[f(\alpha,\beta), \alpha]$
Residual Energy	$[\beta, 8f(\alpha,\beta))$	$[8f(\alpha,\beta), 7f(\alpha,\beta))$	$\cdots$	$[4f(\alpha,\beta), 3f(\alpha,\beta))$	$[3f(\alpha,\beta), 2f(\alpha,\beta))$	$[2f(\alpha,\beta), f(\alpha,\beta))$	$[f(\alpha,\beta), \alpha]$

Table 4: The AHP fundamental scale covering our criteria.

	$\mathcal{D}$	$\mathcal{RE}$
$\mathcal{D}$	1	n
$\mathcal{RE}$	1/n	1

	$\mathcal{D}$	$\mathcal{RE}$
$\mathcal{D}$	1	1/n
$\mathcal{RE}$	n	1

Table 5: Pairwise comparison matrix of distance and residual energy for $n \in [0, 9]$.

Algorithm 3 Required Consistency Ratio (CR) Calculation

Require $|S_{req}| > 0$
Ensure $C.I. \leftarrow \frac{\lambda_{max}-n}{n-1}$
Ensure $R.I. \leftarrow$ Table 2
Ensure $\Omega \leftarrow 0.1$
while $|S_{req}| \neq 0$ **do**
  if $C.R. > \Omega$ and $|S_{req}| > 3$ **then**
    $S_{sub} \backslash s_3$
    $S_{sub} \leftarrow s_{3+i}$ ▷ s_{3+i} with lowest R.E. and dist.
  else if $C.R.$ is $< \Omega$ and $|S_{req}| \leq 3$ **then**
    $C.R. \leftarrow \frac{C.I.}{R.I.}$
  else if $C.R. \leq \Omega$ **then**
    $MCV \leftarrow s_i$
  end if
  $|S_{req}| \leftarrow -1$
end while

The time complexity of Algorithm 3 is provided in the Supplementary file.

An example explaining the proposed method is provided in the supplementary file.

6 Simulation Results

6.1 Simulation Parameters

For our simulations, 100 sensors are used, covering an area of 500m x 500m. We used five MCVs. Each sensor has a battery capacity of 900 Joules and a constant recharge rate of 1.5 J/s. The sensor's energy consumption rate is set to within the range (0.02,1) J/s. The sensors that are used are temperature, and humidity sensor. The mobile charger has a battery capacity of 9500 Joules that is used to charge sensors. The sensors have a battery capacity of 900J. While the MCV's traveling energy is 15,000J [14]. The speed of the mobile charger is 5 m/s. Each experiment was simulated for a period of four hours. The summary of the simulation environment parameters is given in Table 6. In terms of communication and data transmission, we used TCP / IP protocol. Much as it has some weakness like point-to-point communication, responsiveness, and no event-driven [21]. We still used it because of its scalability, open standard, reliability, error detection and correction, routing and addressing, security, universality, interoperability, and security.

Table 6: Simulation parameters

Parameter	Value
Size of network (m^2)	500 x 500
Number of nodes	100
Sensor Type	DHT11 (Temp. and Humidity Sensor
Number of MCV	5
Speed of MCV	5 m/s
MCV charging rate	1.5 J/s
MCV energy capacity	9500 J
MCV travelling energy	15,000J
Sensor energy capacity (Li-ion battery)	900 J
Initial energy of MCV	9500J
Weight of distance	[0,1]
Weight of residual energy	[0,1]
Simulation time	4 hours
Sensor's energy threshold	30-40Joules

6.2 Evaluation Metrics

The performance of the proposed method was evaluated based on the following metrics:

Average remaining energy: The average remaining energy of the sensors, denoted by ($\bar{\mu}$), in the network, is defined as the energy remaining in the system at a given point in time, mathematically expressed by Equation (23).

$$\bar{\mu} = \frac{1}{|S|}\sum_{i=1}^{|S|} e_{res_{s_i}}. \tag{23}$$

Dead Nodes: denoted by (D_n), it represents the number of nodes whose residual energy is zero at a given time period, expressed by Equation (24).

$$D_n = \begin{cases} 0, & \text{if } e_{res_{s_i}} > 0 \\ \sum_{i=0}^{|S_{req}|} s_i, & \text{if } e_{res_{s_i}} = 0 \end{cases} \tag{24}$$

Average Queue Length: denoted by Q_l, is given by Equation (25). The average queue length represents the average of recharge requests that are sent to the base station.

$$Q_l = \frac{1}{|S_{req}|}\sum_{i=1}^{|S_{req}|} s_i \tag{25}$$

where $|S_{req}| \leq |P_{MCV_1} \cup P_{MCV_2} \cup \cdots \cup P_{MCV_n}| = |S|$.

Response Time: The time the MCV takes to respond to the request from a sensor. Defined as the amount of time from the moment the sensor sends a request until the moment the MCV starts charging the sensor.

Survival Rate (ρ): is the ratio of the sensors that are alive in comparison to the sensors that die in a given period of time. As the charging rate increases, the survival rate also increases, mathematically expressed by Equation (26).

$$\rho = \frac{S_r}{|S|} \tag{26}$$

and

$$S_r = |S| - S_{dead} \tag{27}$$

where $|S|$ is the size of the queue and S_{dead} is the number of dead sensors in a given period of time.

Waiting Queue Time: is the time that the sensor spends waiting for it to be charged.

Charging Duration: is the time it takes the MCV to charge a sensor. This is given by Equation (6).

6.3 Experimental Results and Analysis

In this section, we present the evaluation results of our experiments and then analyze the results afterward. Note that the results presented are an average of ten independent runs of simulations. The simulator for the experiment is available at [1].

6.3.1 Experimental Results

The results are presented by varying the time. As time changes, we observe how the evaluation metrics behave. The simulations were run for a period of four hours. Our results were compared with two other methods, mTS [20] and IOCS [35] that uses residual energy and distance as criteria. The results of the experiments are shown in Figure 2 through Figure 8.

Average Remaining Energy (ARE). The evaluation results of average remaining energy by varying time is shown in Figure 2. At the beginning, the average remaining energy is high. Then, due to sensors performing operations like sensing the environment, computations, transmitting, and communication, the average remaining energy starts to drop. As time progresses, the average remaining energy stabilizes. Our method results in an average of around 5.91% increase compared with mTS and a 25.82% better performance compared with IOCS. The reason our method gives better ARE is that, our method has, faster response time due to short queues and also the sensors with the lowest energy and the shortest distance will have high priority. In IOCS a sensor that has large distance will result in having high priority due to Equation 7 in [35]. Whereas mTS, aims at maximizing energy usage efficiency.

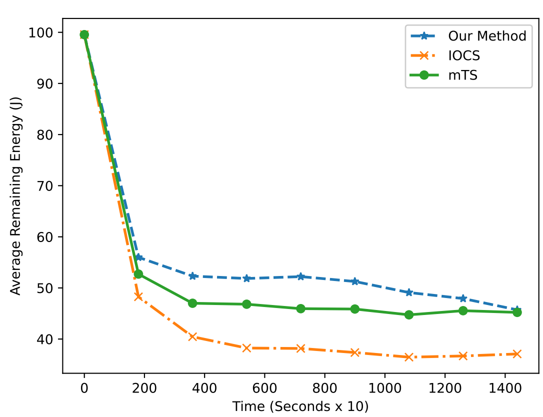


Figure 2: Average Remaining Energy.

Charging Duration. As depicted in Figure 3, charging duration for sensors increases as time passes by. This then stabilizes because energy depletion in sensors also stabilizes, resulting in near-constant charging duration. Our method results in a 10.56% improvement compared with mTS and an 11.75% improvement in comparison with IOCS. On average, this indicates that using our method, the MCV spends less time charging a sensor than the other two methods. Although all methods fully charge sensors. The reason our method has better performance is that, in our method, the allocation of sensors to the MCV is such that no MCV will be allocated more sensors than necessary, i.e., fair allocation of sensors to the MCVs resulting in the MCVs spending less time charging sensors. This is because when calculating the priorities, the allocation gives high priorities to sensors closer to the MCV and those about to die, i.e., those sensors having low energy. In IOCS, the influence of distance results in sensors requiring more charging time since as the MCV is traveling the sensor will still consume some energy and this can slightly increase the charging duration even though the MCV is within its charging area.

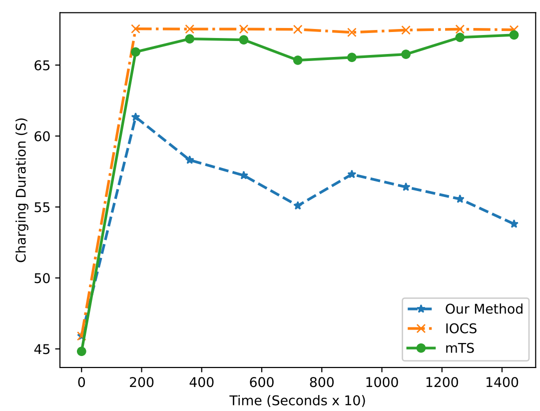


Figure 3: Charging Duration

Dead Nodes. Figure 4 show the results of the nodes that die as time varies. The number of dead nodes increases as time progresses. This is because the ratio of MCV to sensors is not one-to-one. As such, as the MCV is charging some sensors, more sensors will be added to the queue. In this context, our method shows an increase in the number of sensors that die up to 50.2% compared with mTS and an improvement of 47.01% in comparison with IOCS. For the comparison with mTS the recalculation of the consistency ratio causes delays in some cases in dispatching the MCV to charge sensors as priorities have to be recalculated so that they are within the 10% threshold and also the aggregation process of the priority has an effect in delaying dispatching of the MCV. Delays in dispatching an MCV result in some sensors dying. Whereas in IOCS, giving sensors located a longer distance high priority will result in more sensors dying as the MCV will need to travel a longer distance within its service area to reach an energy exhausted sensor.

Average Queue Length (AQL). We present the results of the AQL over time in Figure 5. In our case, the AQL decreases over time due to shorter charging duration and faster response time. Our method achieves an average of 45.74% improvement over mTS and 52.34% over IOCS. This

1. https://github.com/qondwani/wrsn

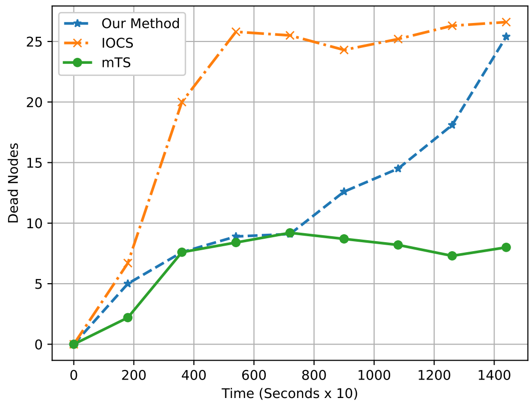


Figure 4: Dead Nodes

is because, according to our method, each MCV is allocated a queue. To charge a sensor, priority is calculated by on greedy algorithm of selecting a sensor dependent on residual energy and distance. The sensor with the highest priority is the one that is charged. So sensors with the lowest energy and closest to the MCV will have higher overall priority; this results in more sensors being charged and lower average queue length. In IOCS, much as each MCV has its own service area and queue to serve, as pointed out earlier, the priority is heavily influenced by the distance, the longer the distance the larger the priority. This is due to Equation 7 in [35]. This will affect the size of the queue in IOCS.

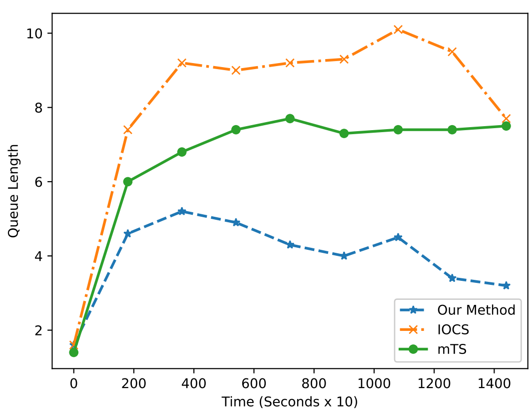


Figure 5: Queue Length

Response Time. The results for response time are given in Figure 6. As can be seen in Figure 6, initially, the response time is faster as fewer sensors send recharge requests to MCV. As time progresses, response time increases slightly due to some sensors being in the waiting queue for an MCV. Response time for the next sensor will depend on the residual energy of the current sensor and the charging rate. The higher the residual energy of the current sensor, the faster the next-to-charged response time will be. Our method had a faster response time of averaging 35.12% compared to mTS and 62.94% over IOCS. The response time is shorter because the AQL is shorter on average because by considering the low residual energy and also giving high priority to shorter distance, it means that the MCV will travel shorter distance to recharge a sensor as opposed to mTS and IOCS. which makes the MCV response to the recharge request faster, hence a faster response time.

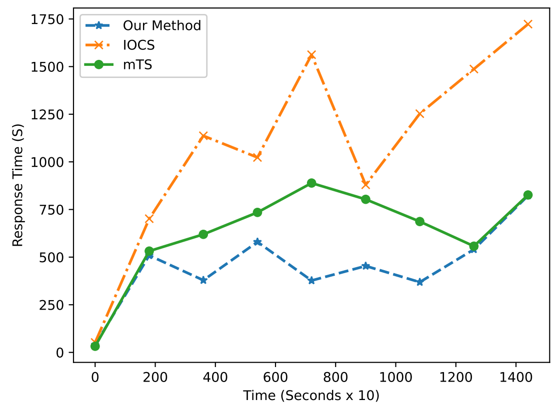


Figure 6: Response Time

Survival Rate. Figure 7 shows the survival rate results. It shows that the progression of time results in fewer sensors surviving. The survival rate stabilizes due to some sensors being charged and thereby preventing sensors from dying. In this context, mTS has better performance than our method. mTS achieves a better performance of 3.9% than our proposal and 12.31% better than IOCS. This is because one of the main aims of mTS is to maximize the sensors' survival rate regardless of other metrics. The other reason is that the time our method takes to calculate the consistency ratio affects the survival rate in comparison to mTS. In IOCS, the survival rate is affected by the time the MCV takes to travel to the nearest requesting sensor. The longer distance the MCV takes to travel affects the survivability of the sensors and in IOCS, the influence of distance is high resulting in some sensors dying.

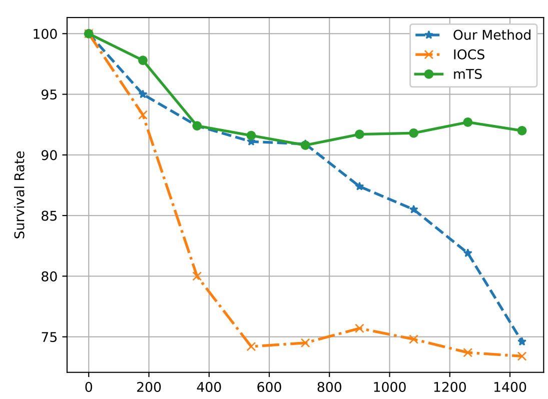


Figure 7: Survival Rate

Waiting Queue Time (WQT). Waiting queue time is related to response time and queue length. A shorter response time will result in the WQT being shorter. This similarly applies to the queue length in that the shorter the queue length, the shorter the WQT. In this context, our method shows an improvement of an average of 40.04% in comparison with mTS and 62.94% in comparison with IOCS as in Figure 8. This is because our method has a shorter response time, resulting in shorter WQT. In IOCS, the waiting queue time is high because the response time is high and also the queue length is high due to the reasons explained above. In our method, as the distance or residual energy gets smaller, the priority of the sensor gets high. Similarly, if both the distance and residual energy gets smaller, then the priority will increase, giving the sensors in our method high priority than the other methods we compared with. This is because, over time, the residual energy of the sensor has to decrease, resulting in the priority of the sensor increasing even if the distance is increasing, thereby giving the sensors in our method an opportunity of being charged by the MCV.

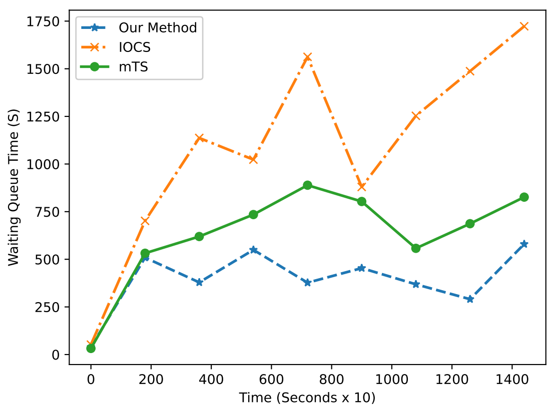


Figure 8: Waiting Queue Time

6.4 Challenges in Real-life Implementation

The challenges in implementing this work in real-time scenario include technical complexities, resource constraints, scalability, management complexities, and redundancy and fault tolerance. These have been explained in detail in supplementary file.

7 Conclusion

This study proposes a charging scheduling scheme based on MCDM. In particular, we used AHP to determine the next-to-be-charged sensor. Based on our research, we have found few works that used AHP in WRSN mobile charger scheduling and no research that has purely applied AHP to this problem. In our work, two criteria of distance and residual energy were used to decide sensor priority. We performed extensive experiments in order to compare our method with two other methods. The comparisons with the two other methods show an improvement in most of the metrics. Better results were observed in the overall average remaining energy in the system, charging duration, average queue length, response time, and waiting queue time. The survival rate was slightly better in mTS, with dead sensors being more in our method. The main challenge is ensuring that the consistency ratio is within the 10% threshold. This can result in the number of computations being high because after a sensor is recharged, for the MCV to select the next sensor, the priorities are recalculated to make sure that the priorities are less than 10%. In future work, we plan to consider increasing further the number of criteria of the sensor. We also plan to include MCV attributes such as traveling energy and recharging energy in MCV scheduling. Lastly, we plan to compare our method with fuzzy logic-based methods. This will help in understanding the difference in performance of the different multi-criteria methods and establish a concrete step towards the next generation scheduling and charging schemes.

Acknowledgments

This paper is supported in part by the Innovation Team and Talents Cultivation Program of the National Administration of Traditional Chinese Medicine (No. ZYYCXTD-D-202208), and in part by the National Natural Science Foundation of China under Grant 62372310, and in part by the Liaoning Province Applied Basic Research Program under Grant 2023JH2/101300194.Saeed work emanated from research conducted with the financial support of Science Foundation Ireland under Grant number SFI/12/RC/2289_P2.

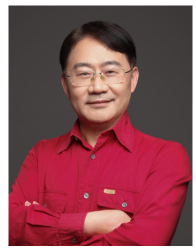
References

[1] L. Xie, Y. Shi, Y. T. Hou, and H. D. Sherali, "Making sensor networks immortal: An energy-renewal approach with wireless power transfer," IEEE/ACM Transactions on Networking, vol. 20, no. 6, pp. 1748–1761, 2012.
[2] O. Jonah and S. V. Georgakopoulos, "Wireless power transfer in concrete via strongly coupled magnetic resonance," IEEE Transactions on Antennas and Propagation, vol. 61, no. 3, pp. 1378–1384, 2013.
[3] C. T. Rim, "34 - wireless charging of electric vehicles," in Power Electronics Handbook (Fourth Edition) (M. H. Rashid, ed.), pp. 1113–1137, Butterworth-Heinemann, fourth edition ed., 2018.
[4] A. Madhja, S. Nikoletseas, and T. P. Raptis, "Distributed wireless power transfer in sensor networks with multiple mobile chargers," Computer Networks, vol. 80, pp. 89–108, 2015.
[5] H. Dai, Q. Ma, X. Wu, G. Chen, D. K. Y. Yau, S. Tang, X.-Y. Li, and C. Tian, "Chase: Charging and scheduling scheme for stochastic event capture in wireless rechargeable sensor networks," IEEE Transactions on Mobile Computing, vol. 19, no. 1, pp. 44–59, 2020.
[6] T. Wu, P. Yang, H. Dai, C. Xiang, X. Rao, J. Huang, and T. Ma, "Joint sensor selection and energy allocation for tasks-driven mobile charging in wireless rechargeable sensor networks," IEEE Internet of Things Journal, vol. 7, no. 12, pp. 11505–11523, 2020.
[7] H. A. Illias, N. S. Ishak, H. Mokhlis, and M. Z. Hossain, "IoT-based hybrid renewable energy harvesting system from water flow," in 2020 IEEE International Conference on Power and Energy (PECon), pp. 204–208, 2020.
[8] A. Jushi, A. Pegatoquet, and T. N. Le, "Wind energy harvesting for autonomous wireless sensor networks," in 2016 Euromicro Conference on Digital System Design (DSD), pp. 301–308, 2016.
[9] T. Ruan, Z. J. Chew, and M. Zhu, "Energy-aware approaches for energy harvesting powered wireless sensor nodes," IEEE Sensors Journal, vol. 17, no. 7, pp. 2165–2173, 2017.
[10] S. Tuli, S. Ilager, K. Ramamohanarao, and R. Buyya, "Dynamic scheduling for stochastic edge-cloud computing environments using A3C learning and residual recurrent neural networks," IEEE Transactions on Mobile Computing, vol. 21, no. 3, pp. 940–954, 2022.
[11] P. Liu, R. Berry, and M. Honig, "A fluid analysis of a utility-based wireless scheduling policy," IEEE Transactions on Information Theory, vol. 52, no. 7, pp. 2872–2889, 2006.
[12] L. He, Y. Gu, J. Pan, and T. Zhu, "On-demand charging in wireless sensor networks: Theories and applications," p. 28 – 36, 2013. Cited by: 118; Conference name: 10th IEEE International Conference on Mobile Ad-Hoc and Sensor Systems, MASS 2013; Conference date: 14 October 2013 through 16 October 2013; Conference code: 102400.

[13] L. Xie, Y. Shi, Y. T. Hou, W. Lou, H. D. Sherali, and S. F. Midkiff, "On renewable sensor networks with wireless energy transfer: The multi-node case," in 2012 9th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), pp. 10–18, 2012.
[14] A. Kaswan, P. K. Jana, M. Dash, A. Kumar, and B. P. Sinha, "DMCP: a distributed mobile charging protocol in wireless rechargeable sensor networks," ACM Trans. Sen. Netw., vol. 19, dec 2022.
[15] S. P. R. Banoth, P. K. Donta, and T. Amgoth, "Dynamic mobile charger scheduling with partial charging strategy for WSNs using deep-Q-networks," Neural Computing and Applications, vol. 33, no. 22, pp. 15267–15279, 2021.
[16] A. Tomar and P. K. Jana, "A multi-attribute decision making approach for on-demand charging scheduling in wireless rechargeable sensor networks," Computing, vol. 103, pp. 1677 – 1701, 2021.
[17] R. Saaty, "The analytic hierarchy process—what it is and how it is used," Mathematical Modelling, vol. 9, no. 3, pp. 161–176, 1987.
[18] L. He, L. Kong, Y. Gu, J. Pan, and T. Zhu, "Evaluating the on-demand mobile charging in wireless sensor networks," IEEE Transactions on Mobile Computing, vol. 14, no. 9, pp. 1861–1875, 2015.
[19] W. Ouyang, M. S. Obaidat, X. Liu, X. Long, W. Xu, and T. Liu, "Importance-different charging scheduling based on matroid theory for wireless rechargeable sensor networks," IEEE Transactions on Wireless Communications, vol. 20, no. 5, pp. 3284–3294, 2021.
[20] C. Lin, Z. Wang, J. Deng, L. Wang, J. Ren, and G. Wu, "mTS: temporal-and spatial-collaborative charging for wireless rechargeable sensor networks with multiple vehicles," in IEEE INFOCOM 2018 - IEEE Conference on Computer Communications, pp. 99–107, 2018.
[21] P. K. Donta, S. N. Srirama, T. Amgoth, and C. S. R. Annavarapu, "Survey on recent advances in IoT application layer protocols and machine learning scope for research directions," Digital Communications and Networks, vol. 8, no. 5, pp. 727–744, 2022.
[22] W. Xu, W. Liang, X. Lin, and G. Mao, "Efficient scheduling of multiple mobile chargers for wireless sensor networks," IEEE Transactions on Vehicular Technology, vol. 65, no. 9, pp. 7670–7683, 2016.
[23] X. Ding, W. Chen, Y. Wang, D. Li, and Y. Hong, "Efficient scheduling of a mobile charger in large-scale sensor networks," Theoretical Computer Science, vol. 840, pp. 219–233, 2020.
[24] L. Fu, P. Cheng, Y. Gu, J. Chen, and T. He, "Optimal charging in wireless rechargeable sensor networks," IEEE Transactions on Vehicular Technology, vol. 65, no. 1, pp. 278–291, 2016.
[25] A. Tomar, L. Muduli, and P. K. Jana, "A fuzzy logic-based on-demand charging algorithm for wireless rechargeable sensor networks with multiple chargers," IEEE Transactions on Mobile Computing, vol. 20, no. 9, pp. 2715–2727, 2021.
[26] 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 Transactions on Mobile Computing, vol. 20, no. 5, pp. 1846–1861, 2021.
[27] A. Kaswan, P. K. Jana, and S. K. Das, "A survey on mobile charging techniques in wireless rechargeable sensor networks," Commun. Surveys Tuts., vol. 24, p. 1750–1779, jul 2022.
[28] F. Sitorus, J. J. Cilliers, and P. R. Brito-Parada, "Multi-criteria decision making for the choice problem in mining and mineral processing: Applications and trends," Expert Systems with Applications, vol. 121, pp. 393–417, 2019.
[29] T. L. Saaty and L. G. Vargas, How to Make a Decision, pp. 1–21. Boston, MA: Springer US, 2012.
[30] T. Saaty and M. Ozdemir, "Why the magic number seven plus or minus two," Mathematical and Computer Modelling, vol. 38, no. 3, pp. 233–244, 2003.
[31] P. M. P. Constantin Zopounidis, Handbook of Multicriteria Analysis. Springer, Berlin, Heidelberg, 1 ed., 2010.
[32] M. Doumpos and E. Grigoroudis, "Multicriteria decision aid and artificial intelligence: Links, theory and applications," 2013.
[33] A. R. Al-shabeeb, "The use of AHP within GIS in selecting potential sites for water harvesting sites in the azraq basin—jordan," J. Geogr. Inf. Syst., vol. 08, no. 01, pp. 73–88, 2016.
[34] E. H. Forman and M. A. Selly, "Decision by objectives: How to convince others that you are right," 2001.
[35] Y. Dong, G. Bao, Y. Liu, M. Wei, Y. Huo, Z. Lou, Y. Wang, and C. Wang, "Instant on-demand charging strategy with multiple chargers in wireless rechargeable sensor networks," Ad Hoc Networks, vol. 136, p. 102964, 2022.

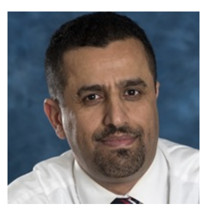
Kondwani Makanda received a Master of Computer Engineering from Kumoh National Institute of Technology, South Korea in 2015. Currently, he is pursuing his Ph.D. in the Department of Computer Science and Technology at the University of Science and Technology of China. His research interest include Wireless Rechargeable Sensor Networks.

Ammar Hawbani is full Professor at Shenyang Aerospace University, specializing in IoT, WSNs, WBANs, WMNs, VANETs, and SDN. Earned B.S., M.S., and Ph.D. from the University of Science and Technology of China. Served as a Postdoctoral Researcher at USTC and later as an Associate Researcher until 2023. Currently, holds the position of a Full Professor at the School of Computer Science.



Xingfu Wang received a B.S. degree in electronic and information engineering from the Beijing Normal University of China, Beijing, China, in 1988, and an M.S. degree in computer science from the University of Science and Technology of China, Hefei, China, in 1997. He is an Associate Professor with the School of Computer Science and Technology, University of Science and Technology of China. His current research interests include information security, data management, and WSN.

Abdulbary Naji received a B.Eng. degree in communication engineering from Taizz University, Taizz, Yemen, in 2018. Currently, he is an M.S. student in the Computer science department at the University of Science and Technology of China (USTC), Hefei, China. His research interests include Remote Sensing, Wireless Communication, SDR, FPGAs, communication algorithms design, SDN, IoT, WSNs,and WRSN.



Ahmed Al-Dubai [SM] received the Ph.D. degree in Computing from the University of Glasgow in 2004. He is currently Professor of Networking and Communication Algorithms in the School of Computing, Engineering and Built Environment at Edinburgh Napier University, UK. His research interests include Communication Algorithms, Internet of Things, and Future Internet.

Liang Zhao (S'09–M'17) is a Professor at Shenyang Aerospace University, China. He received his Ph.D. degree from the School of Computing at Edinburgh Napier University in 2011. Before joining Shenyang Aerospace University, he worked as associate senior researcher in Hitachi (China) Research and Development Corporation from 2012 to 2014. He is also a JSPS Fellow. His research interests include ITS, VANET, WMN and SDN.

SAEED Hamood ALSAMHI received the B.Eng. degree from the Department of Electronic Engineering, IBB University, Yemen, in 2009, and the M.Tech. degree in communication systems and a Ph.D. degree from the Department of Electronics Engineering, Indian Institute of Technology (Banaras Hindu University), IIT (BHU), Varanasi, India, in 2012 and 2015. Since 2019, he has been an Assistant Professor at Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen. In 2020, he worked as MSCA SMART 4.0 FELLOW at Athlone Institute of Technology, Athlone, Ireland. Currently, he is working as a research fellow at the Insight Centre for Data Analytics, University of Galway, Galway, Ireland.