Abdalaziz Sawwan, Jie Wu. Energy-Efficient Minimum Mobile Charger Coverage for Wireless Sensor Networks. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 37(1): 1–20 January 2022.

Energy-Efficient Minimum Mobile Charger Coverage for Wireless Sensor Networks

Abdalaziz Sawwan*, Jie Wu, *Fellow, IEEE*

Department of Computer and Information Sciences, Temple University, Philadelphia 19122, USA

E-mail: abdalaziz.sawwan@temple.edu; jiewu@temple.edu
Received October 25, 2021; accepted December 27, 2021.

Abstract Sustaining an operational wireless sensor network (WSNs) is challenging due to the persistent need of the battery-powered sensors to be charged from time to time. The procedure of exploiting mobile chargers (MCs) that traverse to the fixed sensors of the network and wirelessly transfer energy in an efficient matter has been considered widely as a promising way to tackle that challenge. An optimization problem, called the mobile charger coverage problem, arises naturally to keep all of the sensors alive with an objective of determining both the minimum number of MCs required meeting the sensor recharge frequency and the schedule of these MCs. It's shown that this optimization problem becomes NP-hard in high-dimensional spaces. Moreover, the special case of homogeneous recharge frequency of the sensors has already been proven to have a tractable algorithm if we consider the 1-dimensional space, whether that space is a line or a ring. In this work, we seek to find a delicate border between the tractable and intractable problem space. Specifically, we study the special case of heterogeneous sensors that take frequencies of 1's and 2's (lifetimes of 1 and 0.5 time units) on a line, conjecture its NP-hardness, propose a novel brute-force optimal algorithm, and present a linear-time greedy algorithm that gives a 1.5-approximation solution for the problem. Afterwards, we introduce the energy optimization problem of the MCs with minimized number and solve it optimally. Comprehensive simulation is conducted to verify the efficiency of using our proposed algorithms that minimize the number of MCs.

Keywords Cooperative charging, linear networks, energy optimization, mobile chargers, wireless charging, wireless sensor networks.

1 Introduction

The employment of wireless sensor networks (WSNs) that utilize *mobile chargers* (MCs) for sustaining the sensors alive in the network has been growing in recent years. The recent advancement of the technologies used for wireless charging makes if practical to utilize MCs in order to sustain the sensors alive. One of the problems that face WSNs is the *mobile charger coverage problem*. This is the optimization problem that has the objective of minimizing the number of MCs used to charge the sensors in the network so that specific requirements, including the sensors' charging requirements, with other constraints are satisfied. The solution of this optimization problem needs to include the trajectories of these MCs.

In this work, we study the mobile charger coverage problem for a 1-dimensional (1-D) line specific heterogeneous WSN, construct an optimal solution, and propose an approximation algorithm for it. There is a specific frequency of time for each one of the sensors at which the sensor needs to be charged by having an MC visit it. We assume an instant full-charging of the sensors

Regular Paper

This research was supported in part by NSF grants CNS 2128378, CNS 2107014, CNS 1824440, CNS 1828363, CNS 1757533, CNS 1629746, and CNS 1651947

*Corresponding Author

once the MC visits the sensor's location. Moreover, the speed of the MC may not exceed a certain limit, which is the maximum speed $v_{\max}$. The MCs are assumed to have limitless capability of charging.

Formalizing the general mobile charger coverage problem, we consider a multi-dimensional space with a distribution of sensor nodes $S = \{s_i\}$ with an assigned fixed location x_i for every sensor s_i. We will denote the sensors interchangeably by their names s_i and their locations x_i. Each one of these sensor nodes x_i needs to be visited by an MC node from the set of deployed $MC = \{MC_i\}$ at a given frequency f_i, i.e. x_i has to be visited by one of the MCs no more than $1/f_i$ after the previous visit occurred at x_i. An optimization problem arises to determine the minimum number of MCs needed to satisfy the charging requirement of the sensors, the MCs' coverage areas, and their velocities at every moment. A homogeneous mobile charger coverage problem is the problem where the frequencies are equal for all of the sensors. Heterogeneous mobile charger coverage problem is the name of all other mobile charger coverage problems.

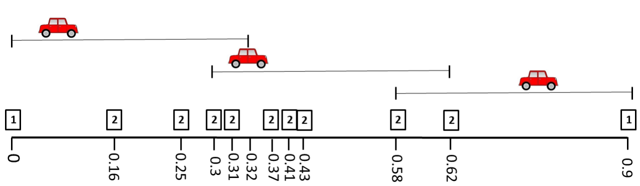

Fig.1. Toy example for the problem showing an optimal MC-solution.

Figure 1 shows an example of a heterogeneous WSN problem with allowed frequencies of 1's and 2's. The sensors of frequency 1 and frequency 2 are denoted as boxed 1's and 2's. We will call these sensors 1-sensors and 2-sensors, respectively. In the example, we see an optimal solution for a linear WSN of eleven sensors distributed at the locations shown in the figure.

Wu *et al.* [1] have come up with an optimal solution for the homogeneous mobile charger coverage problem for both the 1-D ring and 1-D line distributions of sensors. Furthermore, they showed that the solution for a line distribution has at most one MC more than the number of MCs in the solution of the same distribution on a ring. Hence, we focus our efforts in this work to consider 1-D line distributions of sensors. Their optimal solution to solve the homogeneous line problem is done by simply scheduling k MCs to cover non-overlapping fixed intervals of length 0.5 so that all of the sensors are covered, assuming, without loss of generality, the maximum speed of the MCs to be one unit distance per unit, and the frequencies of the sensors to be 1. In addition to that, they have started the investigation of the heterogeneous problem by proposing an approximation algorithm with a factor of 2 that solves the problem for a line distribution of sensors with frequencies $f_i \in \{1, 2, \dots, k\}$ by greedily assigning MCs with non-overlapping coverage areas that go back and forth as far as possible at maximum speed while completely supplying the demand of all of the sensors in their coverage areas.

Here, we raise a concern about the delicate border in the mobile charger coverage problem between the intractable and tractable solution for it and try to fill this gap by studying the 1-D linear heterogeneous WSN problem with frequencies of 1's and 2's, we will call this heterogeneous distribution of sensors *(1, 2)*-WSN. In order to consider general frequencies, all frequencies bounded by the frequency 2^i $(i = 0, 1, 2, ...)$ can be grouped together. This means that frequencies in the range $[2^i, 2^{i+1}]$ can be considered to form a virtual independent network i as discussed in [2].

Furthermore, the energy consumption of mobile charging poses many challenges in the context of WSNs [3]. The most impactful factor on energy consumption is the control of the speed of the MC in its trajectory. Since in this work we only consider the case where the MCs need to fully maintain the sensors to be always charged, i.e., so that they never run out of battery. Hence, given a specific coverage area for each MC, the speed of it would be the important part to consider in order to minimize the energy consumption of a WSN. The hardness of the problem of minimizing the energy consumed from the motion of the MCs comes from the possible non-uniform general distributions of sensors throughout the network combined with the different possible trajectories and coverage areas of the MCs. This results in the need of determining whether the MC should move faster or slower at a specific location and time in order to minimize the total energy consumed from the motion of the MCs in the whole WSN while maintaining the sensors alive all of the time.

Our results are summarized as follows:

- An optimal solution for $(1, 2)$-WSNs' mobile charger coverage problem. This solution exhausts a set of solutions with specific properties and chooses the optimal one from them. Also, we conjecture the NP-hardness of this problem.
- An approximation solution with an improved approximation ratio of 1.5 for $(1, 2)$-WSNs, an enhancement to this solution, and an analytical extension for the previous approximation solution.
- An optimal trajectory for the minimized number of MCs that guarantees minimum energy consumption by the motion of the MCs in the WSN.
- A comprehensive simulation to verify the closeness of our approximation solutions to the optimal one in different distributions of sensors. The distributions were chosen to model different real-life scenarios.

The remainder of the work is organized as follows. In Section II, some related works are reviewed. In Section III, the optimal solution for the $(1, 2)$-WSN problem is proposed, and the NP-hardness of the problem is conjectured. In Section IV, a greedy algorithm with an approximation ratio of 1.5 for the $(1, 2)$-WSN is proposed, an enhancement for this solution is demonstrated, and an analytical expansion for the previously proposed 2-approximation general algorithm is performed. Section V shows the optimal trajectory of the minimized number of MCs that minimized the energy consumption by their motion. In Section VI, simulation results are presented to compare the different proposed solutions. Finally, Section VII gives the conclusion.

2 Related Work

The breakthrough of the employment of strong magnetic resonances in wireless energy transfer technology [4] gave a reliable way to provide the sensors in WSNs with power [5]. The wireless energy transfer technology has many commercial applications [6]. Research has been conducted on wireless energy charging by applying MCs to charge sensors in WSNs [7]. Wu *et al.* [8] have formulated the mobile charging problem which allows cooperative charging of sensors by MCs in a way that guarantees none of the sensors will eventually run out of energy, which is the same constraint we have in our work.

The same problem has been formulated with many variable parameters: considering the MCs with limited energy capacity or with unlimited energy capacity so that the MCs themselves need to be recharged periodically [9, 1, 10], considering the demand of the sensors to be deadline-based or frequency-based [11, 12], and considering the charging of the sensors to be instant once they are visited or gradual in which a charging time is needed [13, 14]. In our model, we consider the MCs to have an infinite amount of energy, charging instantly once they visit the sensors that demand their chargings on a frequency base. We assume the charging time takes zero time. If the actual charging time takes t units of time, it can be converted to our proposed model by adding distance based on the maximum velocity of the charging unit, as discussed in [14].

The objective function to be optimized has some variances in the literature too. Some studied the problem trying to minimize the total distance a constant number of MCs travel [15]; others tried to minimize the maximum distance traveled by any one of the MCs [16]; while others studied minimizing the total power consumed [8], and others studied the case in which maximizing the charging throughput itself is concerned [17] . In our work, our objective is to minimize the number of MCs needed to keep the sensors alive, similar to the model Wu *et al.* have studied [8].

While we consider an instant full charging of the sensors as some did [12], some have considered the problem of charging the sensors to a partial capacity with a charging rate constraint [18, 19]. Finally, it is worth mentioning that even though 1-D [1], 2-D [20, 21, 22], and 3-D [23] WSNs have been studied for this problem, our work of studying the 1-D case remains novel since we try to investigate the NP-hardness boundary of the problem. For multi-dimensional instances of sensor distributions, we can use various dimension reduction processes, e.g., from 2-D to 1-D by constructing a spanning tree and then finding a Hamiltonian path around the tree.

Regarding the energy consumption of the WSN consideration, research has been conducted in order to explore how we can optimize the energy consumption of the network. Wu *et al.* [12] considered a more general model regarding energy that the MCs collaborate with each other in order to recharge not only the network's sensors, but also recharge each other in a way that relatively larger WSNs, where the base station is far from the sensors, could be completely served. Furthermore, Wang *et al.* [24] have conducted research with the aim of minimizing the traveling energy cost of multiple MCs while satisfying the charging requirement of all of the sensors in the WSN. Dai *et al.* [25] have studied the minimum number of possible MCs that are energy-constrained with their recharging trajectories while maintaining the sensors alive all of the time. Madhja *et al.* [26] investigated appropriate MC-coordination techniques with the consideration of energy-efficient techniques by demonstrating some novel protocols for energy-efficient charging coordination processes that are central and distributed. Shu *et al.* [3] have worked on evaluating the optimal velocity control to charge sensors of the WSN in an energy-efficient way considering the energy consumption of the sensors. In contrast, we propose the optimal energy-efficient trajectories of the minimized number of MCs that ensure the lowest cost of the traveling MCs given a simple quadratic energy model. This work is an extension to our previous paper [27].

3 Optimal Solution for $(1, 2)$-WSN

We approach the $(1, 2)$-WSN problem with the assumption that the maximum speed of MCs is one unit distance per unit time.

3.1 Subspace reduction

In this subsection, we reduce the search space for the MC-solution into one that has at least one optimal solution by imposing some restrictions for the possible solution. We will call our target optimal solution $\mathcal{O}$. The restrictions on the search space are:

- In the optimal solution $\mathcal{O}$, the trajectories of the MCs are end-to-end. This property holds after reducing any optimal solution to $\mathcal{O}$. This reduction can be made by replacing any detour by an extension of the MC‘‘s trajectory to one side of its coverage area. A detour is when an MC traverses from point A to point B in a time more than the minimum possible time such that the MC does not visit point B more than once or reach the end of its coverage area.

  To prove this, we call the difference in time between the detour and the minimum possible time

τ. By replacing this detour by a maximum-speed trajectory and traversing by τ time after one of the edges of the coverage area without detouring nor violating the frequency requirement of any sensor, an equivalent optimal solution will be obtained from the previous one.

- The MCs never go to the left of the leftmost sensor in $\mathcal{O}$; if there is an optimal solution in which the MCs cover an area to the left of the leftmost sensor, it can be reduced to another optimal solution in which the trajectories of the MCs are bounded from the left by the leftmost sensor.

- The MCs never meet each other. First, we perform a reduction in which the MCs never pass each other. This reduction can be made simply by swapping the velocities (direction and speed) of any two MCs passing each other. After that, if any two MCs meet, we simply apply a shift in time to the trajectory of one of them so that they do not meet.

Applying these restrictions to our search space guarantees that there will be at least one optimal solution $\mathcal{O}$ in the reduced search space. The optimal solution $\mathcal{O}$ has the following properties:

Property 1: The optimal solution $\mathcal{O}$ has the leftmost uncovered sensor completely supplied by exactly one MC.

This is a direct corollary from reducing any optimal solution to one ($\mathcal{O}$) in which MCs never meet under the imposed restrictions. Hence, we can not make two MCs supply the leftmost uncovered sensor without meeting.

Property 2: No sensor is supplied by more than two MCs in the optimal solution $\mathcal{O}$.

Since every MC will be deployed mainly to fully supply the demand of the leftmost sensor of the remaining distribution of sensors, the maximum length of any coverage area of any MC will not exceed 0.5, which requires supplying every sensor in the coverage area with energy at least once every unit of time. This means we would not need more than two MCs to supply any sensor of frequency 1 or 2.

Property 3: An MC‘‘s starting point is always more than 0.25 away from the starting point of the previous MC in $\mathcal{O}$.

We know from property 1 and property 2 that the optimal solution $\mathcal{O}$ has all of the sensors in the first 0.5 distance units completely supplied by at most two MCs. Thus, alleviating the resulting problem as much as possible will be achieved by making the next MC *able to* reach as distant away as possible away, and that only happens if the starting point of the coverage area of the next MC is after at least 0.25 distance units of the starting point of the coverage area of the previous MC. This means that the next MC will not visit the sensors in the first 0.25 distance units.

3.2 Algorithm overview

In this subsection, we show the high-level of an algorithm that searches for all of the solutions in the reduced search space with the restrictions, then picks the one of them that uses the least number of MCs, which is $\mathcal{O}$.

Constructing all of the solutions with these properties for our *(1, 2)*-WSN, where the leftmost uncovered sensor location is x_1, will be as follows: First, deploy an MC that completely supplies the sensors in $[x_1, x_1 + 0.25]$. Completely supplying them directly implies that **1)** the start point of the coverage area of the MC is x_1, and that **2)** the endpoint of the coverage area of the MC is in $[x_1 + 0.25, y_{\text{last}}]$ where y_{last} is $0.25 +$ the location of the first 2-sensor in $[x_1, x_1 + 0.25]$ if there is any, or $y_{\text{last}} = x_1 + 0.5$ if there is no 2-sensor in $[x_1, x_1 + 0.25]$. The exact possible locations of the endpoint are discussed in the next subsection. Second, eliminate all of the completely supplied sensors, and then repeat the process for the new distribution of sensors calling the leftmost uncovered sensor x_1. We will call the visited 2-sensors in $(x_1 + 0.25, y_{\text{last}}]$ *partially-supplied* sensors, since they are not completely supplied and will be addressed collaboratively with the next MC in an overlapping region.

What makes this problem hard is what we will call *the constraint of overlaps*, which states that if two MCs supply a 2-sensor collaboratively in their overlapping region, then the coverage areas of the two MCs have to be equal. This natural constraint arises from the fact that if the two coverage areas are not equal, then the time gap between their visits to the 2-sensor in the overlapping area will keep changing until it eventually reaches more than 0.5, which would mean that the concerned 2-sensor in the overlapping area is not supplied properly.[1]

Determining the endpoint of the coverage area of any new deployed MC is the hardest part. We find that the possible locations of the best endpoint are limited: if there is no 2-sensor in $(x_1 + 0.25, y_{\text{last}}]$, then choosing the endpoint to be y_{last} will always be the best, but if there exists at least one 2-sensor in $(x_1 + 0.25, y_{\text{last}}]$ and x_1 is not a partially-supplied sensor, we will be left with two options:

- **Option 1**: Have all of the 2-sensors in $(x_1 + 0.25, y_{\text{last}}]$ supplied collaboratively with the next MC by having them in an overlap region between the two MCs.
- **Option 2**: **1)** Define $y_{\text{partition}}$ to be a point in $[y_{\text{first}}, y_{\text{last}}]$, where y_{first} is the first 2-sensor in $(x_1 + 0.25, y_{\text{last}}]$, **2)** have the 2-sensors in $[y_{\text{first}}, y_{\text{partition}}]$ supplied collaboratively with the next MC by having them in an overlap region between them, and **3)** have the 2-sensors in $(y_{\text{partition}}, y_{\text{last}}]$ supplied completely by the next MC.

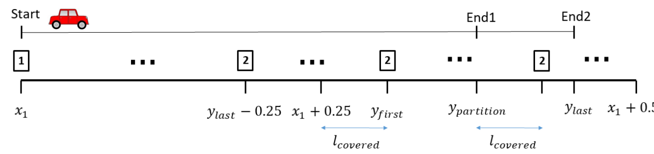

Fig.2. Deploying an MC with no previously 'visited' sensors.

Figure 2 shows the two options of the endpoints when deploying an MC in the case where the sensor x_1 is not partially-supplied.

Considering the *constraint of overlaps*, the always-best choice for the endpoint under option 1 will be y_{last} and the always-best choice under option 2 will be $y_{\text{partition}}$, where $y_{\text{partition}} = max(x_i) - l_{\text{covered}}$, x_i is a 2-sensor location in $[x_1 + 0.25, y_{\text{last}}]$ that satisfies the following condition: There is no 2-sensor in $(x_i - l_{\text{covered}}, x_i)$, and $l_{\text{covered}} = y_{\text{first}} - (x_1 + 0.25)$. However, if $max(x_i) = y_{\text{first}}$, then we set $y_{\text{partition}} = y_{\text{first}}$, and if there is no 2-sensor in $(x_1 + 0.25, y_{\text{last}}]$, we set $y_{\text{partition}} = y_{\text{last}}$.

If the sensor x_1 is partially-supplied, then the endpoint of the next MC's coverage area will lie under two other options:

- **Option 3**: The endpoint is $x_1 + 0.25$ (we will not make use of the overlap.)
- **Option 4**: The endpoint is $x_1 + l$, where l is the coverage area of the previous MC (we will make use of the overlap.)

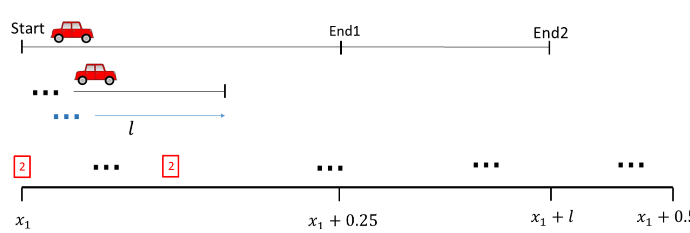

Fig.3. Deploying an MC with previously visited sensors.

Figure 3, which illustrates the partially-supplied sensors in the red color, shows these two options. This

[1]The time gap eventually becomes greater than 0.25 by reaching $min\{2 \times \text{Area}_1, 2 \times \text{Area}_2\}$ if $\frac{\text{Area}_1}{\text{Area}_2}$ is rational. If the ratio of their coverage areas is not rational, then it reaches $min\{2 \times \text{Area}_1, 2 \times \text{Area}_2\} - \delta,\ 0 < \delta < \epsilon\ \forall \epsilon > 0$.

means that when we deploy a new MC, there will be only one possible start point of its coverage area, which is the leftmost uncovered sensor, and a maximum of two possible options of its endpoint: option 1 and option 2 if there is no partially-supplied in the remaining distribution, or option 3 and option 4 if x_1 is partially-supplied.

Furthermore, in the case of having partially-supplied 2-sensors in the remaining distribution, choosing option 4 in the special case where there is a non-visited 2-sensor in $(x_1, x_1 + l - 0.25)$ will result in a solution that does not have property 3. Hence, we exclude option 4 in this case.

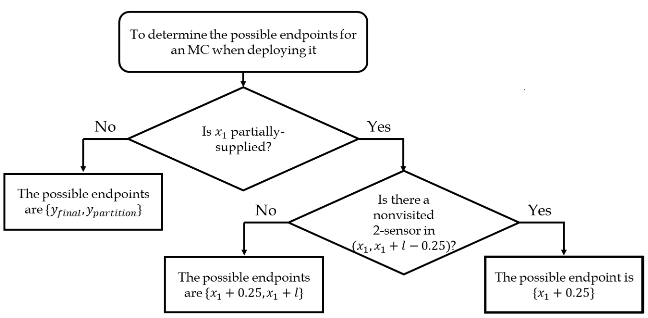

Fig.4. An illustration of criterion $\mathcal{C}$ of how to determine the endpoints.

Now, we have everything set up to define a criterion, $\mathcal{C}$, to choose the set of possible endpoints for the MC from three cases: if x_1 is not partially-supplied, there will be two possible endpoints (option 1 and option 2), if x_1 is partially-supplied and there is a nonvisited 2-sensor in $(x_1, x_1 + l - 0.25)$, then there will be one possible endpoint (option 3), and lastly, there will be two possible endpoints (option 3 and option 4) if there is no 2-sensor in $(x_1, x_1 + l - 0.25)$. Figure 4 shows criterion $\mathcal{C}$.

Algorithm 1 produces the set of the possible endpoints for any new MC to be deployed to cover an area starting from the leftmost remaining uncovered sensor x_1.

3.3 Algorithm design

At this point, we tackled the hardness of the problem: *Where should the endpoint of the coverage area of the next MC be determined?* Even though we know for sure where the MC's coverage area starts (it will start from the leftmost uncovered sensor), determining where the previous one ends remains hard. Algorithm 1 is designed so that the produced MC-solutions hold the properties of our reduced search space.

Algorithm 2 Search space for the optimal solution $\mathcal{O}$

Input: Sensor locations $\{x_1, x_2, \ldots, x_n\}$ and frequencies $\{f_1, f_2, \ldots, f_n\}$, $f_k \in \{1, 2\}$.
Output: The set of possible MC-solutions including $\mathcal{O}$.
Initialization: All sensors are unvisited.
$S = \{\}$ //The search space of the MC-solutions.
$l = 0$ //The last deployed MC's coverage area.
Optimal$(x_1, x_2, \ldots, x_n, l, S)$:
1: **if** all sensors are completely supplied **then**
$S = S \cup \{\text{Last generated MC-solution}\}$.
return
2: Call Algorithm 1 to determine $Possible\_Endpoints$.
3: **for** each k in $Possible\_Endpoints$ **do**
4: Generate an MC that covers $[x_1, k]$, add it to the current MC-solution, and let $l = k - x_1$.
5: Eliminate all sensors in $[x_1, x_1 + 0.25]$ and 1-sensors in $[x_1 + 0.25, k]$.
6: Annotate the 2-sensors in $(x_1 + 0.25, k]$ as 'visited'.
7: Call **Optimal**$(x_1, x_2, \ldots, x_n, l, S)$ where x_1 is the leftmost sensor.

The main question that holds and forces us to exhaust all possible MC-solutions is *how can we make our next deployed MC contribute in supplying the other sensors in the (1, 2)-WSN in a way that makes the remaining distribution of sensors need as less MCs as possible?* Algorithm 2 exhausts all of the possible MC-solutions with properties 1‘‘3.

The MC-solution in S, which is produced by Algorithm 2, with the least number of MCs is the optimal solution $\mathcal{O}$ and has a time complexity of $O(d \times 16^L)$.

Proof. Algorithm 2 does nothing but exhaust all of the MC-solutions with properties 1‘‘3 in our subspace.

Since $\mathcal{O}$ has these properties, choosing the MC-solution with the least number of MCs from all of the generated solutions with these properties gives us $\mathcal{O}$. □

We conjecture that the $(1, 2)$-WSN problem is an NP-hard problem. Analysing the complexity of our brute-force algorithm (Algorithm 2) shows that the recursive call inside the loop is equivalent to a maximum number of nested loops of $L/0.25$, where every loop has a maximum of two iterations. Each iteration takes $O(n/0.25)$ to find $max(x_i)$ in order to calculate $y_{\text{partition}}$, where n is the number of sensors in the search region of $max(x_i)$. This search region is bounded by 0.25. This means that in the worst-case scenario, the algorithm takes $O(d\times 2^{L/0.25}) = O(d\times 16^L)$ time, where d is the maximum sensor-density of any region of length 0.25 in the given linear WSN. d is bounded by $n/0.25$.

4 Approximation Solutions for $(1, 2)$-WSN

In this section, we propose our new greedy approximation algorithm and an enhancement for it in the first part. In the second part, we perform an analytical expansion for a previous general approximation algorithm.

4.1 A novel approximation algorithm

First, we will set up a lower bound for the optimal solution $\mathcal{O}$. The lower bound is going to be determined by seeking the optimal solution Ω of the problem after lifting *the constraint of overlaps*. We will assume that 2-sensors are now satisfied if they are supplied collaboratively by two MCs of different coverage areas, these coverage areas shall not exceed 0.5. Then we will propose an approximation solution for the original problem which produces a number of MCs that is bounded by 1.5 of the lower bound produced by Ω.

Even after lifting *the constraint of overlaps*, the optimal solution of the resulting alleviated problem Ω still has the properties 1“3. The optimal solution Ω is produced as follows: deploying an MC covering the area starting from the leftmost sensor and ending as far as possible while completely supplying the sensors in the first 0.25 distance. We may treat the 2-sensors visited by this sensor but not completely supplied (the visited 2-sensors after the 0.25 distance) as 1-sensors for the next MC. Continuing to deploy the MCs at this manner produces the lower bound of the optimal solution Ω.

Algorithm 3 Greedy 1.5-approximation solution

Input: Sensor locations $\{x_1, x_2, \ldots, x_n\}$ and frequencies $\{f_1, f_2, \ldots, f_n\}$, $f_k \in \{1, 2\}$.
Output: A 1.5-approximation MC-solution.
Initialization: $i = 0$ //The MCs' indexes.
1: **While** there is a non-zero leftmost sensor x **do**
2: $i = i + 1$.
3: **if** there is a leftmost 2-sensor x' in $[x, x + 0.25]$ **then**
Generate MC_i that covers $[x, x' + 0.25]$.
4: **else**
Generate MC_i that covers $[x, x + 0.5]$.
5: Eliminate the sensors in $[x, x + 0.25]$.
6: Subtract 1 from visited sensors in $(x + 0.25, x + 0.5]$.
7: for every MC_{2i}, generate an additional MC that covers the same area MC_{2i} covers.

The MC-solution, which is produced by Algorithm 3, is an approximation solution with a ratio of 1.5 and has a time complexity of $O(L)$.

Proof. After producing Ω, simply addressing the sensors in the overlapping regions (i.e., considering the *the constraint of overlaps* again) by deploying additional MCs for them gives us the 1.5-approximation solution. Algorithm 3 produces this solution; lines 1-6 of it produce Ω and line 7 generates the additional MCs that address any possible overlap. The number of these additional MCs cannot exceed half the number of MCs in Ω. This confirms our approximation ratio of 1.5.

Analysing the time complexity of Algorithm 3 shows that its run-time is linearly proportional to $L/0.25$ as the number of iterations is upper-bounded by $O(L/0.25) = O(L)$, where L is the whole length of the linear WSN. □

We may enhance the solution produced by Algorithm 3 by deploying the additional MCs only when needed instead of deploying them for all even-numbered MCs. That may be done by making the last line of the algorithm produce additional MCs to only cover the overlaps of different-lengths coverage areas. Even though this improvement generally reduces the number of additional MCs, the approximation factor of the solution remains 1.5.

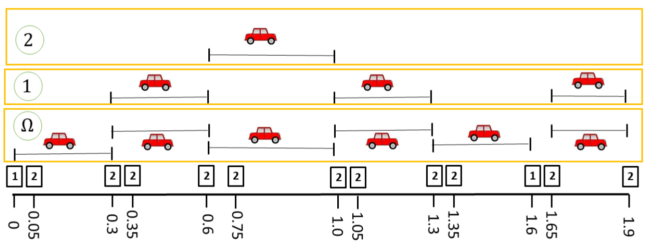

Fig.5. The lower bound of the optimal, the 1.5-approximation algorithm, and the enhanced 1.5-approximation algorithm.

Figure 5 shows the MCs generated by lines 1-6 of the approximation algorithm (Ω), the additional MCs added by line 7 (1), and the additional MCs added by the enhanced approximation algorithm which only assigns MCs to address the overlaps between two coverage areas of different length (2).

4.2 New analysis for the 2-approximation solution

Algorithm 4 General greedy solution

Input: Uncovered sensor locations $\{x_1, x_2, \ldots, x_n\}$ and frequencies $\{f_1, f_2, \ldots, f_n\}$, $f_k \in \mathbb{R}$.
Output: A 2-approximation MC-solution.
if $n = 0$ **then return**.
Generate an MC that goes back and forth as far as possible at a full speed to cover sensors at $\{x_1, \ldots x_{i-1}\}$.
Recursively call Algorithm 4 for $\{x_i, \ldots x_n\}$.

Wu [1] have proposed Algorithm 4 as an approximation algorithm to generate an MC-solution for heterogeneous WSNs with any frequencies. They have proved that this algorithm produces a solution with an approximation ratio of 2. In this section here, we prove that this approximation ratio becomes tighter as it reaches 1.5 for $(1, 2)$-WSNs.

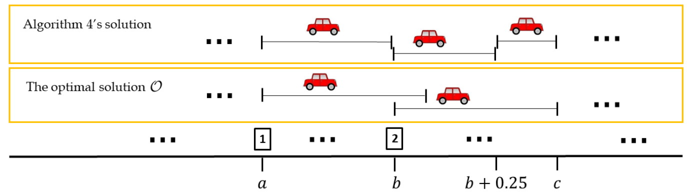

Fig.6. The optimal solution $\mathcal{O}$ and Algorithm 4's approximation solution.

The MC-solution, which is produced by Algorithm 4, is an approximation solution with a ratio of 1.5 when applied to the $(1, 2)$-WSN and has a time complexity of $O(L)$.

Proof. Considering the optimal solution $\mathcal{O}$, we know from property 2 that the optimal solution $\mathcal{O}$ may have overlaps between the coverage areas of no more than two MCs. It is trivial to show that if, for some distribution of sensors, $\mathcal{O}$ has no overlaps, then Algorithm 4 (as well as Algorithm 3) produces the same optimal solution. However, when the optimal solution $\mathcal{O}$ has two MCs with an overlapping region as shown in Figure 6, Algorithm 4 produces exactly three MCs in order to be able to cover the same region the two overlapping MCs cover.

The three MCs will be produced by Algorithm 4 in the following order: the first MC will be deployed to cover the sensors in the region $[a, b)$, then the second one will be deployed to cover the sensors in the region $[b, b + 0.25]$. The third MC will cover any remaining sensor in the region $(b + 0.25, c]$.

In the worst-case scenario, when the optimal solution O has each MC to cover a region with an overlap with exactly one other MC, where this region is significantly far from other coverage areas, Algorithm 4 produces three MCs for each two overlapping MCs. Hence, an approximation ratio of 1.5.

The run-time of the algorithm has an upper-bound of $L/0.25$; for any WSN with sensors of frequencies of

1's and 2's and a given length L, the number of iterations will not exceed $*L/0.25$, even for a dense distribution of 2-sensors. This upper bound of the worst-case scenario gives us a fairly tight upper bound for the time complexity of Algorithm 4, which is $O(L)$. $\square$

5 Optimizing Energy

In this section, we optimize the trajectories taken by the minimized number of MCs that the previous algorithms produce for the objective of minimizing the energy consumed.

5.1 Energy Model and The Basic Optimization Problem

It is clear that the MCs and their trajectories produced by the optimal solution and the approximation solutions do waste energy in their trajectories most of the time, that is due to the fact that their speed is fixed to the maximum all of the time. However, having the degree of freedom of varying the speed at which each MC performs would enable us to optimize the energy by consumed by the motion of the MCs. In this context, we do not consider the energy consumed by the batteries of the sensors because the rate of consuming energy of the sensors, and hence the rate of the energy required for charging them does not change with any of the variables that are under our control (i.e., the number of used MCs and their trajectories).

When we mention the term trajectory in this context, we mean both the set of locations, i.e. the interval, the the MC covers and the velocity (speed and direction) of the MC at each point. We allow sudden changes in the speed or/and direction.

The mathematical model for the energy consumption by the motion of the MCs that we will use in this optimization problem is the simple practical model that is commonly used for moving vehicles which comes mainly from the consideration of the friction force. Our simplified mathematical model of the rate of energy consumption and the energy consumption of an MC traversing at a constant speed v, a specific distance d, and at the corresponding time $t = \frac{d}{v}$ are shown in Equations (1-2), which considers the most dominant term of the general high-degree polynomial that describes the energy consumption rate in terms of speed [28]. In spite of that, the final result of this section would be valid too for any strongly-convex function that describes the rate of energy consumption.

$$P = \alpha v^2 \tag{1}$$

$$E = \alpha v^2 \times \frac{d}{v} = \alpha \frac{d^2}{t} \tag{2}$$

where α is a characteristic constant related to the mass of the MC, friction, and other surrounding factors. E is the total energy consumed by the MC where $|\frac{dE}{dt}| = |P|$.

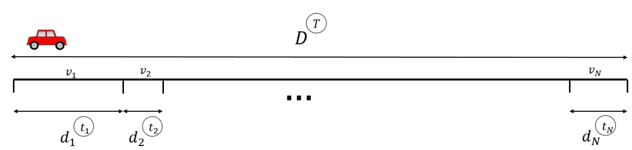

Fig.7. The illustration of the division of distance and time in the optimization problem, the MC travels in N constant speed values on N regions.

To this extent, we define our energy consumption optimization problem for an MC that has to cover a distance D at time of at most T, where the maximum speed $v_{\max} \geq \frac{D}{T}$, in Equations (3-6).

$$\min_{d_1,\ldots,d_N,t_1,\ldots,t_N} \alpha \sum\nolimits_{i=1}^{N} \frac{d_i^2}{t_i} \tag{3}$$

s.t.

$$\sum\nolimits_{i=1}^{N} d_i = D \tag{4}$$

$$\sum\nolimits_{i=1}^{N} t_i = T \tag{5}$$

$$\frac{d_i}{t_i} \leq v_{\max} \quad \forall i \in [1, N], \quad i \in \mathbb{N} \tag{6}$$

where each distance segment d_i is traversed in time t_i at a constant speed $v_i = \frac{d_i}{t_i}$. Figure 7 illustrates the division specified in the optimization problem.

The solution of this optimization problem is simple; which equivalent to satisfying the conditions under the premise of minimizing the maximum speed the MC travels by. Hence, the direct solution of the optimization problem is shown in Equation (7).

$$\frac{d_i}{t_i} = \frac{D}{T} \quad \forall i \in [1, N], \quad i \in \mathbb{N} \tag{7}$$

The most energy-efficient trajectory for an MC with specific sensors to cover is the trajectory whose maximum speed value on any interval is minimized while satisfying the constraints.

Proof. The result follows directly from the optimization problem solution in Equation (7) after considering all sub-intervals as the whole coverage area at which the energy is optimized. □

The solution is to have the MC traveling into a uniform speed that is equal to the minimum allowed speed under the condition. For example, if an MC with $v_{\max} = 1$ speed unit needs to travel a distance of 10 distance units in at least a time of 20 time units, the optimized-energy solution is to travel at speed of 0.5 speed units rather than, for example, traveling at the maximum speed then resting at zero speed for the remaining time.

5.2 Optimizing Energy With No Overlaps

Now, we consider an MC that is covering a set of sensors with different frequencies such that no sensor is charged collaboratively with an another MC, i.e., no overlaps.

We will use the result from the previous subsection to apply it to the MC covering an area of sensors with no overlaps. In order to do that, we need to restrict ourselves to the following three conclusions from the previous subsection:

1. Our objective will be to make the MC travel in the slowest speed possible given that the MC completely satisfies the charging requirements of all of the sensors within its area. Follows directly from Theorem 4.

2. It is always more energy-efficient to shrink the area of coverage of the MC to cover as little as possible, i.e., exactly from the leftmost to the rightmost sensors with no additional area more than that.

3. The empty area between any two adjacent sensors must be traveled by a uniform speed that is the minimum possible one so that all of the constraints are satisfied. Follows directly from Theorem 4.

Now, we have everything needed to introduce the algorithm for determining the most energy-efficient speed values of an MC covering a set of sensors without an overlap. We will do that by going through the example illustrated in Figure 8.

Starting from Figure 8 (a), an MC that covers five sensors with five different frequencies (the solution in this subsection is valid for any frequency values) is illustrated. From conclusion 3, we know that each one of the areas between different sensors will be covered on a constant speed, and since we try to minimize those uniform speed values, we consider the individual case for each sensor so that each one of them is having the MC traveling at the minimum possible speed so that it visits the edges of the whole interval. This results into the case where for each sensor, the minimum uniform speed that the MC can use while going back and forth to the edge of the coverage area, whether to the left or to the right side, uses the maximum allowed value of time away from each sensor, which is basically $\frac{1}{f_i}$.

For example, if we consider sensor s_2, to the right side, we will have a total distance of $2\sum_{i=2}^{4} d_i$, and a maximum allowed time of $\frac{1}{f_2}$. Hence, we will have the minimum possible speed of the MC to traverse uniformly to the right of sensor s_2 is $(2\sum_{i=2}^{4} d_i)/(\frac{1}{f_2}) =$

$2f_2\sum_{i=2}^{4} d_i$. Applying the same procedure to determine the minimum possible speed to the left and right of each sensor gives us the values in Figure 8 (a).

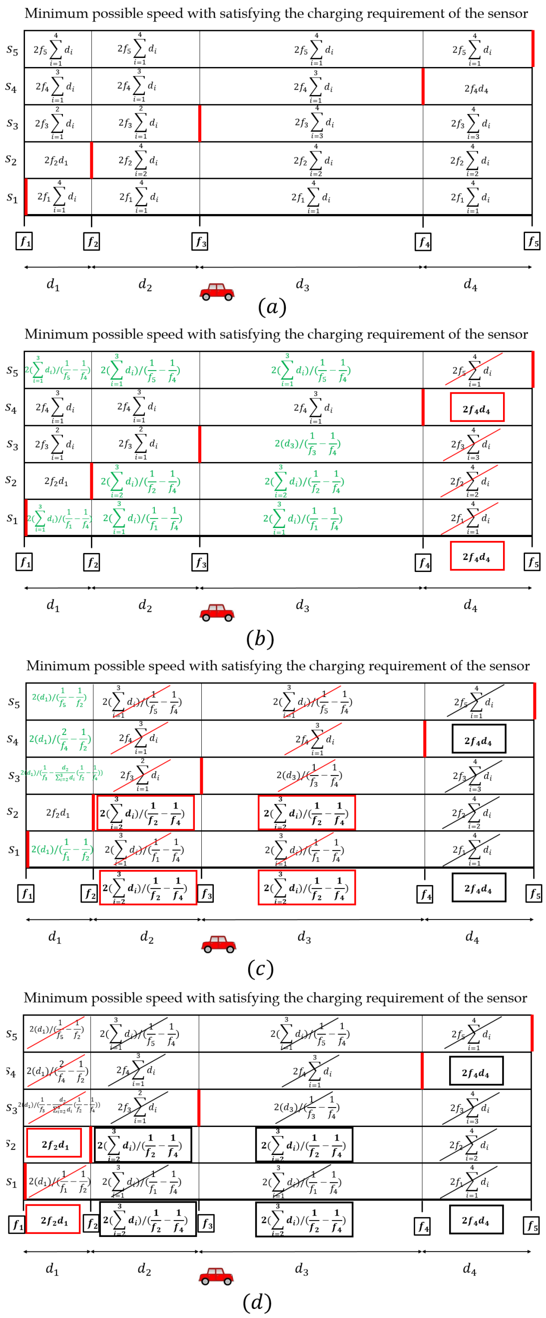

Fig.8. An example of the problem of optimizing speed. (a) shows the initial settings, where each sensor has its minimum possible speed on each region determined. (b) shows the first assignment of the *bottleneck speed* which is the maximum of all speed values in the table. It also shows the updated loosened minimum possible values for the other sensors in green. (c) shows the assignment of the second *bottleneck speed* which turns out to cover more than one region. (d) shows the last step at which the last *bottleneck speed* value was assigned to the last region.

The next step after constructing the table in Figure (8) will be to determine the *bottleneck sensor*, which is the sensor at the current phase of the solution which has the highest value of minimum possible speed. we call the maximum value of all of the minimum possible speed values in the table the *bottleneck speed*. There may be more than one *bottleneck sensor* in the same time when the same maximum value in the table appears for different sensors.

In our example here, the maximum value in Figure (a) is $2f_4d_4$. After determining this maximum value, we will have the case that the highest speed was used at a specific region, which may give more time for other sensors (exclusively to the right or to the left to the sensor depending on where the highest speed was determined) in a way that can further reduce their corresponding minimum speeds. For example, after determining the *bottleneck sensor* to be s_4 at the first step, the MC will spend a total time of $\frac{1}{f_4}$ on d_4. This, in our example, would give us a reduction in sensor s_2 values. Instead of requiring to travel the distance (back and forth) of $2\sum_{i=2}^{4} d_i$ to the right of it in maximum time of $\frac{1}{f_2}$, the requirement now will be to travel the remaining distance to the right, which equals $2\sum_{i=2}^{3} d_i$ in a maximum total time of $\frac{1}{f_2} - \frac{1}{f_4}$, this will result in an update of the minimum possible speed values of sensors s_2 in the table so that we will have a new smaller possible speed which is $2\sum_{i=2}^{3} d_i/(\frac{1}{f_2} - \frac{1}{f_4})$ as shown in Figure 8 (b).

In our example here, the maximum value in Figure (a) is $2f_4d_4$. After determining this maximum value, we will have the case that the highest speed was used at a specific region, which may give more time for other sensors (exclusively to the right or to the left to the sensor depending on where the highest speed was determined) in a way that can further reduce their corresponding minimum speeds. For example, after determining the *bottleneck sensor* to be s_4 at the first step, the MC will spend a total time of $\frac{1}{f_4}$ on d_4. This, in our example, would give us a reduction in sensor s_2 values. Instead

of requiring to travel the distance (back and forth) of $2\sum_{i=2}^{4} d_i$ to the right of it in maximum time of $\frac{1}{f_2}$, the requirement now will be to travel the remaining distance to the right, which equals $2\sum_{i=2}^{3} d_i$ in a maximum total time of $\frac{1}{f_2} - \frac{1}{f_4}$, this will result in an update of the minimum possible speed values of sensors s_2 in the table so that we will have a new smaller possible speed which is $2\sum_{i=2}^{3} d_i/(\frac{1}{f_2} - \frac{1}{f_4})$ as shown in Figure 8 (b).

The same process is done again by picking the highest minimum possible speed value of the remaining values in the table after the update, which is in our example $2\sum_{i=2}^{3} d_i/(\frac{1}{f_2} - \frac{1}{f_4})$. This will set the MC to traverse in this speed on both region d_2 and region d_3. Now, in order to update the other values, we, again, just consider the speed values to the same side of the sensor which has a portion of the region in it determined. This means, for example, that sensor s_1 will have its value at d_1 updated because it is on the right of the sensor, the same side at which a new speed was determined.

Evaluating the update again, consider sensor s_3, nowm to the left of the sensor, the MC must traverse a total distance of $2\sum_{i=1}^{2} d_i$ in a maximum time of $\frac{1}{f_3}$. However, after setting the MC to move at speed $2\sum_{i=2}^{3} d_i/(\frac{1}{f_2} - \frac{1}{f_4})$ on region d_2, which takes time of $2d_2/(2\sum_{i=2}^{3} d_i/(\frac{1}{f_2} - \frac{1}{f_4})) = \frac{d_2}{\sum_{i=2}^{3} d_i}(\frac{1}{f_2} - \frac{1}{f_4})$. Hence, for sensor s_3, the region d_1 now has to be traveled with a maximum total time of $\frac{1}{f_3} - (\frac{d_2}{\sum_{i=2}^{3} d_i}(\frac{1}{f_2} - \frac{1}{f_4}))$, which means that the minimum possible updated speed for region d_1 in order to satisfy the charging requirement of sensor s_3 will be $2d_1/(\frac{1}{f_3} - \frac{d_2}{\sum_{i=2}^{3} d_i}(\frac{1}{f_2} - \frac{1}{f_4}))$. Figure 8 (c) shows the updates at this step of the solution.

Figure 8 (d) shows the last step of determining the speed at the last region, which is d_1 in our example, considering the highest minimum possible value to be $2f_2d_1$ between the remaining updated possible speed values for region d_1.

Algorithm 5 Determining the energy-optimized trajectory of an MC with no overlaps

---

Input: Sensor locations $\{x_1, x_2, \ldots, x_n\}$ and frequencies $\{f_1, f_2, \ldots, f_n\}$, $f_k \in \mathbb{R}$ covered completely by one MC.
Output: The energy efficient trajectory for the MC.
1: Specify regions d_i to be $[x_i, x_{i+1}]$ $\forall i \in [1, n-1]$, $\quad i \in \mathbb{N}$.
2: **While** there is a region d_i with no assigned speed **do**
3: Calculate the minimum possible speed for each individual sensor on all remaining regions d_i.
4: Determine the *bottleneck sensor* with the highest speed of all the minimum possible speeds: the *bottleneck speed*.
5: Assign the determined *bottleneck speed* on its regions.
6: **Return** the assigned speed values on their regions.

Algorithm 5 shows the general way of determining the energy-efficient trajectory of an MC that covers the sensors in its region alone. It's worth mentioning that this algorithm works for any range of frequencies.

The MC trajectory produced by Algorithm 5 is an energy-optimal trajectory. The time complexity is $O(L \times d^2)$.

Proof. Algorithm 5 guarantees that three conclusions from the previous subsection are satisfied while maintaining the smallest possible speed for each region applied while delivering the charging requirement of all of the sensors. Since the algorithm focuses on satisfying the charging requirement of the *bottleneck sensor* while loosening the minimum possible speed values for all of the sensors afterwards, that guarantees that each region d_i will have the minimum possible speed after guaranteeing that the maximum speed assigned so far on the trajectory is minimized as much as the constraints allow, then the second maximum speed, then the third one, and so on. Which proves the theorem as a result of Theorem 4.

Analysing the time complexity of Algorithm 5 shows that it scans at most all of the regions between the sensors where for each one of those scans, it performs a number of operations that is equal to the number of sensors in that region to calculate their minimum possible speed values. The number of those operations for each scan is upper-bounded by $O(d)$. On the other hand, the number of those scans is upper-bounded by $O(L \times d)$, where L is the whole length of the linear WSN and d is the maximum density of sensors on a region of length $\frac{v_{\max}}{2f_{\max}}$. Hence, the total time complexity of Algorithm 5 will be $O(L \times d^2)$ □

5.3 Optimizing Energy With Overlaps

Unlike the previous subsection, this subsection discusses the optimal energy trajectory where the WSN consists of only 1-sensors and 2-sensors. Furthermore, the overlaps in the MC-solutions that we are trying to optimize their trajectories energy consumption are between no more than two MCs.

In order to discuss this problem, we focus our discussion on at least three MCs, where the second one overlaps with both the first and third one in order to satisfy the charging requirement of the cooperatively-maintained 2-sensors in the overlapping areas. The toy example in Figure 1 is a good illustration for our focus. The solution of three overlapping MCs can be extended to any number of overlapping MCs.

The solution of this problem is simple; we first relax the problem by lifting *the constraint of overlaps*. Hence, we can consider each MC individually without worrying about satisfying the condition that the time spend on one cycle of an MC has to be exactly the same as the adjacent one. This allows us to consider the range of covered sensors of the MC just like the previous subsection after reducing the charging requirement of the cooperatively-maintained 2-sensors to treat them as 1-sensors.

After determining the energy-optimal trajectory for each MC in the group of MCs that are overlapping with each other, we need to make all of the MCs take the same time for each cycle to satisfy *the constraint of overlaps*. Thus, we simply calculate the total time taken by the energy-optimal trajectory of each MC in the group of MCs that are overlapping with each other, and then we determine the one that takes the least time, then we make all of the other MCs take the exact same minimum time by increasing the speed values applied on some regions.

To this extent, we need to determine how exactly to adjust the speed values of the MC that is taking an energy-optimal trajectory so that the total time is decreased to equal the time determined by the MC that takes the least time. Following Theorem 3, the least energy-costly increase of the speed values of the MCs are the ones that are applied to the regions with least speed values. Hence the process will be simply to increase the speed taken by the MC on the region that uses the minimum speed until either the new reduced total time is reached or until the speed value of this minimum region equals the speed value of the region with the second minimum speed. In the latter case, we repeat again by increasing the speed value of regions that use the new minimum speed value and so on.

By that, we guarantee that decreasing the time an MC takes to finish its cycle costs as less energy as possible to reach the required minimized total time that does satisfy *the constraint of overlaps*. The process does not break any constraint since we start from the beginning with a solution, i.e., charging requirements of all sensors are satisfied by the MCs covering them.

6 Simulation

In this section, we conduct simulations to evaluate the effectiveness of the algorithms discussed in this work. It is worth mentioning that the energy consumption of the solutions produced by the proposed algorithms do not need a simulation because all of them keep the MCs run at constant maximum speed. Hence, the energy consumption rate is constant that can be evaluated directly under our simplified model $P = \alpha v_{\max}^2$.

6.1 Experimental Settings

In our simulations, the frequencies of the sensors (f_k) follow the Bernoulli distribution with a certain probability for each of the two possible frequencies to occur. The distribution of the locations of the sensors was considered to follow one of three different scenarios. The first scenario distributes the sensors uniformly on the given line segment of length l after placing a sensor on each edge of the line segment. We will call this distribution the *uniform distribution* of sensors.

The second distribution of the locations is a cluster distribution, where k sensors are distributed uniformly as the centers of the clusters, then two sensors are placed on the two edges of the line WSN of length l, then the remaining sensors are divided equally into k groups (if there is a shortage in the remaining number of sensors to be divided equally into k groups, all of the shortage is applied to one random cluster). The remaining sensors are distributed around the k center of clusters on a normal distribution $N(x_{\text{cluster}}, \sigma^2)$, where each group of sensors has its own x_{cluster} as the mean of their locations, and a certain standard deviation σ. We will call this distribution the *cluster distribution* of sensors.

The third distribution is a mixture between the latter two distributions, where a certain percentage of sensors are distributed to follow the uniform distribution, while the rest of them are distributed to follow the cluster distribution. We will call this distribution the *mixed distribution* of sensors.

Our choice to choose such distributions comes from their practical emulation of real-world situations, whether for WSNs or border patrolling applications. This gives us a vast number of parameters: the first one is the percentage of the 2-sensors; the second one is the probability distribution which the locations of the sensors follow. Each one of those distributions has its own additional parameters. The uniform distribution has an additional two parameters: the length of the WSN and the total number of sensors. The cluster distribution has an additional four parameters: the length of the WSN, the total number of sensors, the number of clusters, and the standard deviation of the sensors in the clusters. The mixed distribution has an additional five parameters: the length of the WSN, the total number of sensors, the number of clusters, the standard deviation of the sensors in the clusters, and the percentage of the uniformly distributed sensors.

6.2 Algorithm Comparison

We consider various settings to compare the performance of the four algorithms: the algorithm that produces the optimal solution $\mathcal{O}$, our proposed 1.5-approximation greedy algorithm (Algorithm 3), the enhanced version of our 1.5-approximation algorithm, and Wu‘‘s greedy algorithm for general line heterogeneous WSNs (Algorithm 4). Because of the exponential time-complexity of our optimal algorithm, we include small-scale scenarios of limited lengths and numbers of sensors.

6.3 Experimental Results

We can observe from the first three plots in Figure 9 that for the uniform distribution, fixing all of the parameters but the length of the WSN shows that the number of MCs grows almost linearly with the increase of the length of the WSN. The enhanced algorithm and the Wu's 2-approximation general algorithm both give results very close to the optimal solution under a constant number of sensors. Furthermore, it is clear that the dominance by 1-sensors favors the enhanced 1.5-approximation algorithm over the 2-approximation algorithm.

Observing the last three plots in Figure 9, in which the density of the sensors varies under a fixed length of the WSN, we see that the 2-approximation algorithm behaves very closely to the optimal algorithm with different percentages of 2-sensors as opposed to

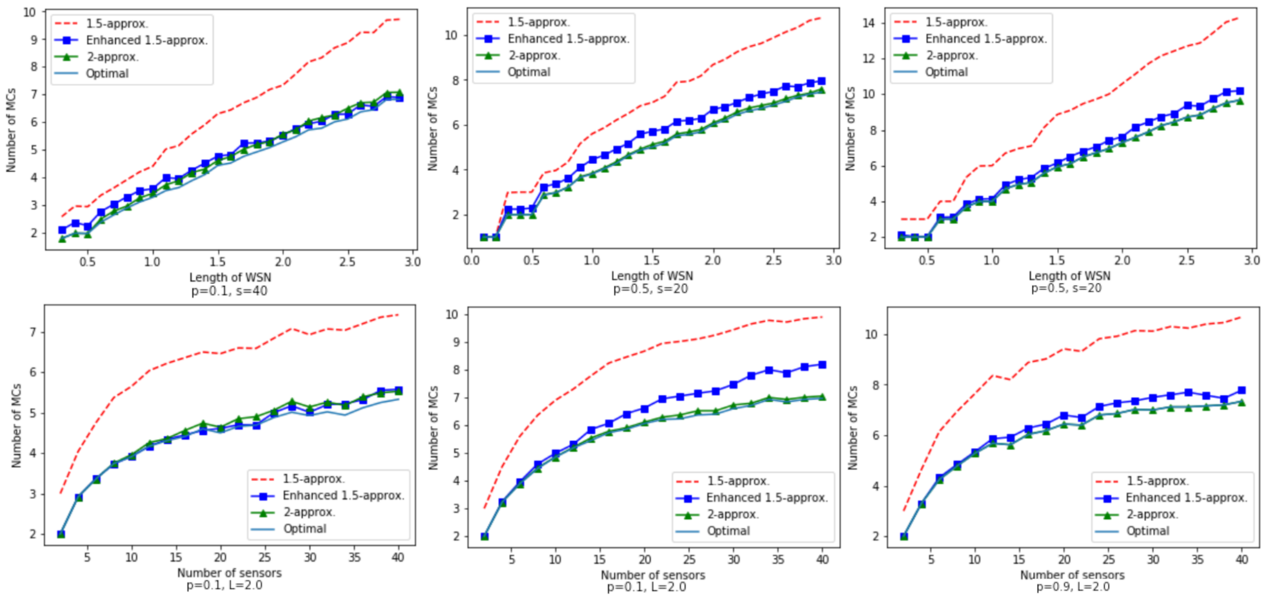

Fig.9. The results of the algorithms under the uniform distribution with various percentages of 2-sensors. s is the number of sensors, p is the percentage of the 2-sensors, and L is the length of the WSN.

the enhanced algorithm that depends significantly on the percentage of the 2-sensors deployed.

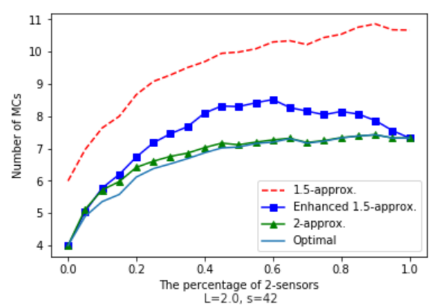

Fig.10. The behavior of the algorithms with uniform distribution under varying percentage of 2-sensors and fixed other parameters.

Figure 10 shows how the different algorithms behave for a fixed number of sensors and a fixed length of the WSN. At the two edges where we have homogeneous WSN, the enhanced and 2-approximation algorithms give exactly the same result as the optimal solution. In general, the 2-approximation algorithm performs better under these settings and very close to the optimal algorithm. However, for the settings where we have 1-sensors dominate the 2-sensors, the enhanced algorithm very closely beats the 2-approximation algorithm. The normal 1.5-approximation algorithm performs very closely to its upper bound due to the blind deployment of the additional MCs to address the overlaps whether they are needed or not.

We may observe from Figure 11 that the behavior of the algorithms under the cluster distribution, the first three plots show that, under low standard deviation (dense clusters), the enhanced and the 2-approximation general algorithms perform very closely to the optimal algorithm. As the clusters become more loose, the enhanced algorithm performs more poorly than the 2-approximation algorithm which does not get affected heavily by the standard deviation parameter. The third plot shows behavior close to the behavior in uniform distribution due to the high standard deviation.

The second three plots in Figure 11 show how the number of clusters affects the needed number of MCs under different standard deviations; the tighter the clusters are (have lower standard deviation), the more their number correlates more strongly with the needed number of MCs. For low standard deviation, the number of MCs increases almost-linearly with the number of clusters under fixed other parameters. With a higher standard deviation, an increase in the number of clus-

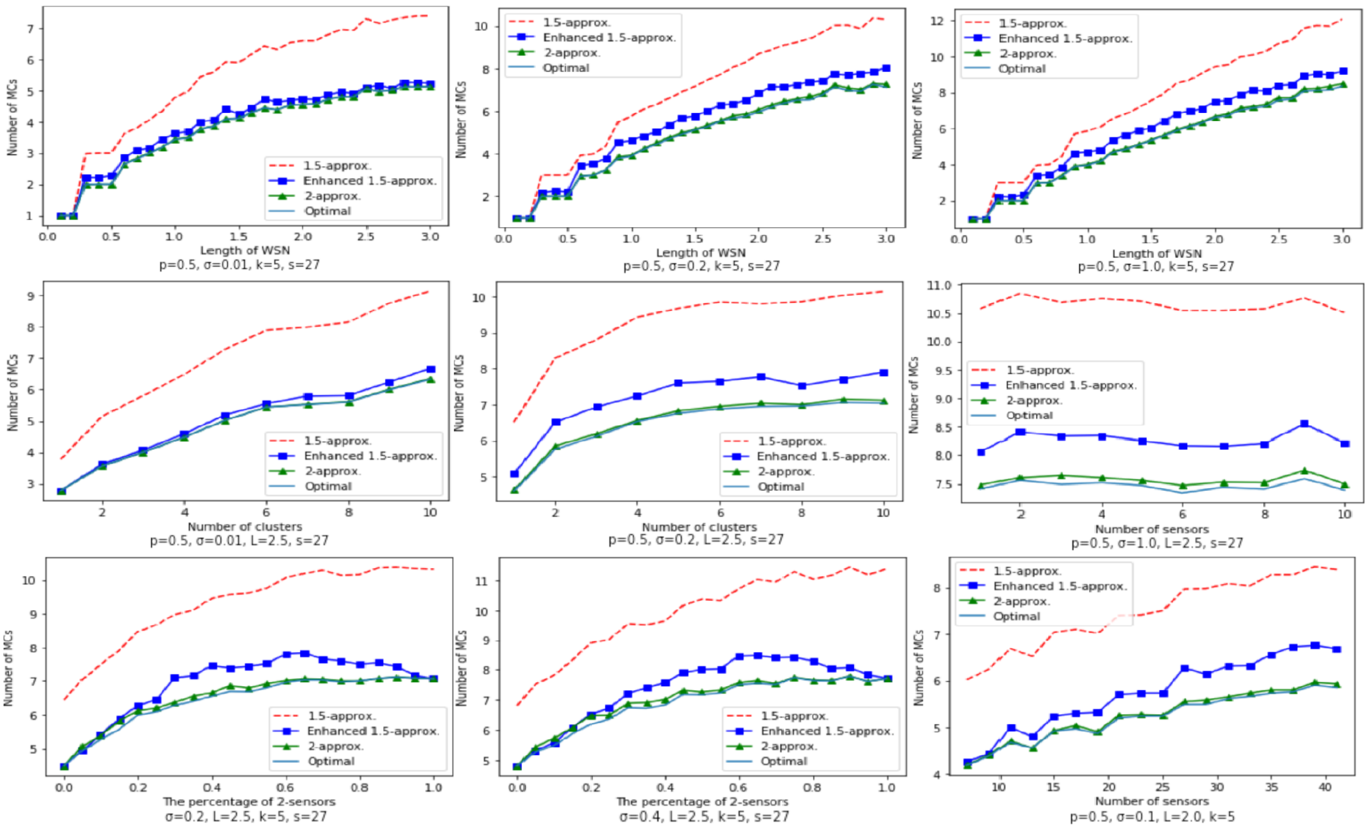

Fig. 11. The results of the algorithms under the cluster distribution with various standard deviations. p is the percentage of the 2-sensors, σ is the standard deviation, L is the length of the WSN, k is the number of clusters, and s is the number of sensors.

ters affects the needed number of MCs less. For very high standard deviation values, the number of clusters does not affect the outcome of the algorithms. Furthermore, it is clear that the 2-approximation algorithm outperforms the enhanced 1.5-approximation algorithm under the cluster distribution. The last line of the plots in Figure 11 shows again how the 2-approximation algorithm performs very closely to the optimal solution. This is due to the fact that this algorithm is actually exactly the same as the first six lines of Algorithm 3 (which produce the lower bound of the optimal solution Ω) except that in line 6, it does not subtract 1 from the 2-sensors in $(x + 0.25, x + 0.5]$.

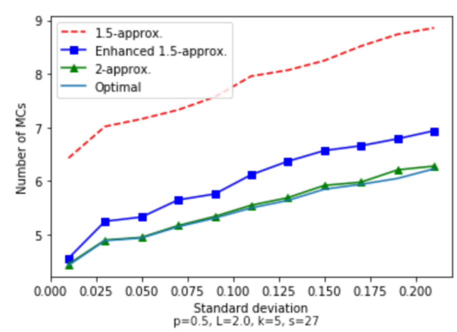

Fig.12. The behavior of the algorithms with cluster distribution under varying standard deviation with fixed other parameters.

Figure 12 shows how the number of MCs, for a fixed number of clusters with fixed length of WSN and number of sensors, correlates directly with the standard deviation, and how under various standard deviations, the 2-approximation algorithm remains very close to the optimal solution.

In Figure 13, the behavior of the algorithms under the mixed distribution is shown. The first plot

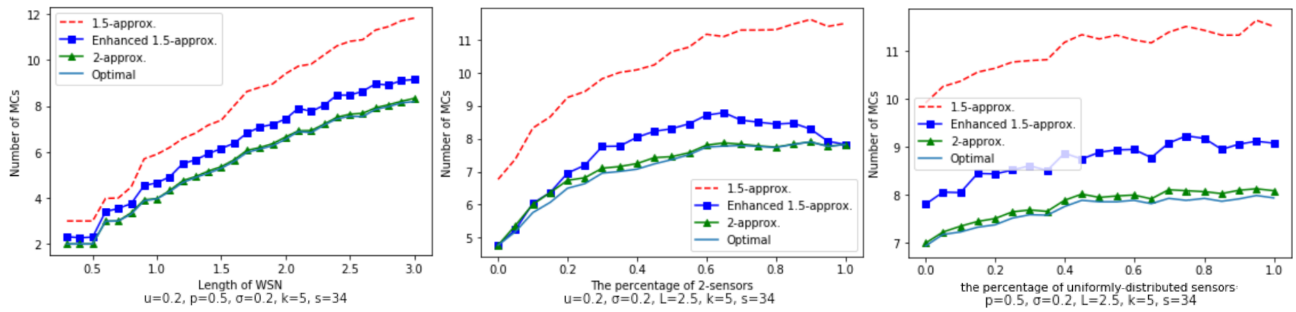

Fig.13. The results of the algorithms under the mixed distribution with different settings, u is the percentage of uniformly-distributed sensors, p is the percentage of the 2-sensors, σ is the standard deviation, L is the length of the WSN, k is the number of clusters, and s is the number of sensors.

shows that the length of the WSN affects the number of MCs almost as linearly as the previous two distributions. From the different distributions and various parameters settings, the previously proposed general 2-approximation algorithm outperforms the enhanced 1.5-approximation algorithm by around 10%, where the original 1.5-approximation algorithm produces a solution that is always close to the upper bound of 1.5. The 2-approximation algorithm remains very close to the optimal solution in all of the observed scenarios due to its closeness to the algorithm that produces the optimal solution's lower bound.

7 Conclusion

In this work, we establish an investigation for the NP-hardness boundaries between the tractable and intractable solutions of the mobile charger coverage problem. The schedule of the least possible number of mobile chargers for heterogeneous line wireless sensor networks, with sensors of frequencies 1's and 2's, is studied. We obtain the optimal solution for this problem by exhausting all of the possible solutions with specific properties, and conjecture the NP-hardness of it. A 1.5-approximation algorithm, an enhancement of this approximation, and an analytical expansion for a previously proposed general 2-approximation algorithm are done. Simulation results compare the optimal solution with our approximation algorithms and the previous general approximation algorithm. Furthermore, we introduce a polynomial-time algorithm to determine the energy-optimal trajectories that the MCs of a WSN with any sensor charging frequencies may have, with another one designed for our specific problem. The simulation shows that in practical set-ups, our enhanced algorithm performs 10% less than the 2-approximation algorithm, which remains very close to the optimal solution. In future work, we will try to study different line WSNs of different frequency ranges, prove the NP-hardness of this problem, and come up with better approximations.

References

[1] R. Beigel, J. Wu, and H. Zheng. "On optimal scheduling of multiple mobile chargers in wireless sensor networks." In Proceedings of the first international workshop on mobile sensing, computing and communication (MSCC '14), 1“6, 2014.

[2] H. Zheng, and J. Wu. "Cooperative Wireless Charging Vehicle Scheduling." In IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS) pp. 224-232, 2017.

[3] Y. Shu, H. Yousefi, P. Cheng, J. Chen, Y. J. Gu, T. He, and K. G. Shin. "Near-optimal velocity control for mobile charging in wireless rechargeable sensor networks." In IEEE Transactions on Mobile Computing, pp. 1699-1713, 2015.

[4] A. Kurs, A. Karalis, R. Moffatt, JD. Joannopoulos, P. Fisher, M. Soljacic. "Wireless power transfer via strongly coupled magnetic resonances." In Science, 83-6, 2007.

[5] W. Xu, W. Liang, X. Jia, Z. Xu, Z. Li, and Y. Liu. “Maximizing Sensor Lifetime with the Minimal Service Cost of a Mobile Charger in Wireless Sensor Networks.” In IEEE Transactions on Mobile Computing, vol. 17, no. 11, pp. 2564-2577, 2018.

[6] PRIMOVE e-Mobility Solution. [Online]. Available: http://primove.bombardier.com

[7] Y. Ma, W. Liang, and W. Xu. “Charging Utility Maximization in Wireless Rechargeable Sensor Networks by Charging Multiple Sensors Simultaneously.” In IEEE/ACM Transactions on Networking, vol. 26, no. 04, pp. 1591-1604, 2018.

[8] M. Wu, Y. Dongdong, K. Jiawen, Z. Haochuan, and Y. Rong. “Optimal and Cooperative Energy Replenishment in Mobile Rechargeable Networks.” In IEEE 83rd Vehicular Technology Conference (VTC Spring), pp. 1-5, 2016.

[9] W. Liang, Z. Xu, W. Xu, J. Shi, G. Mao, and SK. Das. “Approximation Algorithms for Charging Reward Maximization in Rechargeable Sensor Networks via a Mobile Charger.” In IEEE/ACM Transactions on Networking, vol. 25, no. 05, pp. 3161-3174, 2017.

[10] W. Xu, W. Liang, X. Jia, Z. Xu, Z. Li, and Y. Liu. “Maximizing Sensor Lifetime with the Minimal Service Cost of a Mobile Charger in Wireless Sensor Networks.” In IEEE Transactions on Mobile Computing vol. 17, no. 11, pp. 2564‘‘2577, 2018.

[11] P. Yang, T. Wu, H. Dai, X. Rao, X. Wang, P. Wan, and X. He. “MORE: Multi-node Mobile Charging Scheduling for Deadline Constraints.” In ACM Transactions on Sensor Networks vol. 17, no. 01, Article 7, 2020.

[12] S. Zhang, J. Wu, and S. Lu. “Collaborative Mobile Charging.” In IEEE Transactions on Computers, vol. 64, no. 03, pp. 654-667, 2015.

[13] N. Wang, J. Wu, and H. Dai. “Bundle Charging: Wireless Charging Energy Minimization in Dense Wireless Sensor Networks.” In IEEE 39th International Conference on Distributed Computing Systems (ICDCS), pp. 810-820, 2019.

[14] L. He, L. Kong, Y. Gu, J. Pan, and T. Zhu. “Evaluating the On-Demand Mobile Charging in Wireless Sensor Networks.” In IEEE Transactions on Mobile Computing, vol. 14, no. 09, pp. 1861-1875, 2015.

[15] Y. Ma, W. Liang, and W. Xu. “Charging Utility Maximization in Wireless Rechargeable Sensor Networks by Charging Multiple Sensors Simultaneously.” In IEEE/ACM Transactions on Networking, vol. 26, no. 04, pp. 1591-1604, 2018.

[16] W. Xu, W. Liang, and X. Lin. “Approximation Algorithms for Min-Max Cycle Cover Problems.” In IEEE Transactions on Computers, vol. 64, no. 03, pp. 600-613, 2015.

[17] M. Wu, D. Ye, J. Kang, H. Zhang, and R. Yu. “Optimal and Cooperative Energy Replenishment in Mobile Rechargeable Networks.” In IEEE 83rd Vehicular Technology Conference (VTC Spring), pp. 1-5, 2016.

[18] W. Xu, W. Liang, X. Jia, and Z. Xu. “Maximizing Sensor Lifetime in a Rechargeable Sensor Network via Partial Energy Charging on Sensors.” In IEEE 13th International Conference on Sensing, Communication, and Networking (SECON), pp. 1-9, 2016.

[19] S. Banoth, P. Donta, and T. Amgoth. “Dynamic mobile charger scheduling with partial charging strategy for WSNs using deep-Q-networks.” In Neural Computing and Applications, pp.1-13, 2021.

[20] F. H. Tseng, H. H. Cho, and C. F. Lai. “Mobile Charger Planning for Wireless Rechargeable Sensor Network Based on Ant Colony Optimization.” In Advances in Computer Science and Ubiquitous Computing. Lecture Notes in Electrical Engineering, Springer, vol. 715, 2021.

[21] S. Zhang, Z. Qian, J. Wu, F. Kong, and S. Lu. “Optimizing itinerary selection and charging association for mobile chargers.” In IEEE Transactions on Mobile Computing, vol. 16, no. 10, pp. 2833‘‘2846, 2016.

[22] F. Sangare, Y. Xiao, D. Niyato, and Z. Han. “Mobile charging in wireless-powered sensor networks: Optimal scheduling and experimental implementation.” In IEEE Transactions on Vehicular Technology, vol. 66, no. 08, pp. 7400‘‘7410, 2017.

[23] C. Lin, C. Guo, H. Dai, L. Wang, and G. Wu. “Near Optimal Charging Scheduling for 3-D Wireless Rechargeable Sensor Networks with Energy Constraints.” In IEEE 39th International Conference on Distributed Computing Systems (ICDCS), pp. 624-633, 2019.

[24] C. Wang, J. Li, F. Ye, and Y. Yang, “Multi-vehicle coordination for wireless energy replenishment in sensor networks.” In Proceedings IEEE 27th International Symposium on Parallel and Distributed Processing, pp. 1101‘‘1111, 2013.

[25] H. Dai, X. Wu, G. Chen, L. Xu, and S. Lin, “Minimizing the number of mobile chargers for large-scale wireless rechargeable sensor networks.” Computer Communication, vol. 46, pp. 54‘‘65, 2014.

[26] A. Madhja, S. Nikoletseas, and T. P. Raptis, “Efficient, distributed coordination of multiple mobile chargers in sensor networks.” In Proceedings of the 16th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, pp. 101‘‘108, 2013.

[27] A. Sawwan, and J. Wu, “Mobile Charger Coverage Problem for Specific Heterogeneous Wireless Sensor Networks.” Proc. of 18th IEEE International Conference on Mobile Ad Hoc and Smart Systems (MASS 2021). 2021.

[28] R. Galvin. “Energy consumption effects of speed and acceleration in electric vehicles: Laboratory case studies and implications for drivers and policymakers.” In Transportation Research Part D: Transport and Environment, pp. 234-248, 2017.

Abdalaziz Sawwan is a second-year Ph.D. student in Computer and Information Sciences at Temple University. Sawwan received his bachelor's degree in Electrical Engineering from the University of Jordan in 2020. His current research interests include mobile charging and wireless networks, routing protocols, and age of information.

Jie Wu is the director of the Center for Networked Computing and Laura H. Carnell professor at Temple University. He also serves as the director of International Affairs at College of Science and Technology. He served as chair of Department of Computer and Information Sciences from the summer of 2009 to the summer of 2016 and associate vice provost for International Affairs from the fall of 2015 to the summer of 2017. Prior to joining Temple University, he was a program director at the National Science Foundation and was a distinguished professor at Florida Atlantic University. His current research interests include mobile computing and wireless networks, routing protocols, cloud and green computing, network trust and security, and social network applications. Dr. Wu regularly publishes in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including IEEE Transactions on Mobile Computing, IEEE Transactions on Service Computing, Journal of Parallel and Distributed Computing, and Journal of Computer Science and Technology. Dr. Wu was general co-chair for IEEE MASS 2006, IEEE IPDPS 2008, IEEE ICDCS 2013, ACM MobiHoc 2014, ICPP 2016, and IEEE CNS 2016, as well as program cochair for IEEE INFOCOM 2011 and CCF CNCC 2013. He was an IEEE Computer Society Distinguished Visitor, ACM Distinguished Speaker, and chair for the IEEE Technical Committee on Distributed Processing (TCDP). Dr. Wu is a fellow of the AAAS and a fellow of the IEEE. He is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award.