

國立中興大學資訊科學與工程學系  
碩士學位論文

在 WRSN 環境下使用 NSGA-II 演算法佈置無線  
充電站

Using Non-dominated Sorting Genetic Algorithm-  
II for Charging Station Deployment in Wireless  
Rechargeable Sensor Networks



指導教授： 高勝助 Shang-Juh Kao

研 究 生： 胡翔喻 Shiang-Yu Hu

中華民國 一百零九年七月

國立中興大學 資訊科學與工程學系

碩士學位論文

題目：在 WRSN 環境下使用 NSGA-II 演算法佈置無線充電站

Using Non-dominated Sorting Genetic Algorithm-II for Charging  
Station Deployment in Wireless Rechargeable Sensor Networks

姓名：胡翔喻 學號：7107056054

經 口 試 通 過 特 此 證 明

論文指導教授

高 福 勁

論文考試委員

廖 宜 恩

張 華 民

高 福 勁

中華民國

109 年

7 月

1 日

## 中文摘要

雖然無線感測網路 (WSN) 可以使用戶更方便地訪問信息，但是仍然存在一些固有的問題。在 WSN 中每個傳感器都感知各種信息，然後將其傳輸到中繼節點。然而這些動作都會消耗能量，但是傳感器的能量受到電池容量的限制。隨著無線充電技術的發展，可充式無線感測網路 (WRSN) 引起了很多關注。在 WRSN 中，無線充電站可以為傳感器的電池充電，以便它們可以持續運行。由於無線充電站價格昂貴且充電距離有限，因此如何部署最少數量的充電站以覆蓋所有傳感器並滿足所有傳感器的能源需求是一個重要且具有挑戰性的問題。

關於這個問題，大多數研究集中在減少充電站的數量上。它們沒有考慮傳感器與充電站之間的距離。實際上，當傳感器與充電站之間的距離減小時，充電效率將提高。因此充電站無需頻繁補充傳感器的電源。因此，本文通過同時考慮充電站的數量和傳感器與充電站之間的距離，提出了一種新的部署策略。我們將提出的策略轉化為一個多目標問題，並採用非支配排序遺傳演算法-II (NSGA-II) 來解決此問題。借助 NSGA-II 方法，生成的解決方案可以避免陷入單一目標。在我們的方法中，我們將充電站的所有可能位置編碼為一組基因，稱為染色體。通過交叉和突變階段，生成了不同的染色體，以保持下一代解決方案的多樣性。在選擇階段，使用非支配的排序方案以避免陷入局部最優解。

為了驗證所提出策略的可行性，我們使用 Python 編程語言實現了模擬環境。傳感器隨機分佈在 20 X 15 平方公尺的室內環境中。我們根據充電站的數量和總體充電功率，將所提出的方法與基於模擬退火的充電算法 (SABC) 和基於裁員及模擬退火的充電算法 (LSABC) 進行了比較。模擬結果表示，與使用 SABC 和 LSABC 方法獲得的總充電功率相比，在相同的充電站數量下，使用我們的方法獲得的總充電功率分別高出 5% 和 8%。而在相同的傳感器數量下使用 NSGA-II 獲得的充電站數量分別比使用 SABC 和 LSABC 方法獲得的充電站數量分別少 6% 和 1%。

**關鍵字：**可充式無線感測網路，無線充電站布置，非支配排序遺傳演算法-II

## Abstract

While wireless sensor networks (WSNs) allow user access to information more convenient, there are still some inherent problems. In WSNs, each sensor node senses various kinds of information and then transfers them to relay nodes. All of those actions will consume energy, but the energy of sensors is limited by battery capacity. With the development of wireless charging technology, wireless rechargeable sensor networks (WRSNs) have gained a lot of attention. In WRSNs, wireless charging stations can recharge the batteries of sensor nodes so that they can operate sustainably. Since wireless charging stations are costly and have limited charging distances, how to deploy the minimal number of charging stations to cover all sensor nodes and satisfy the energy requirements of all sensor nodes are important and challenging issues.

Regarding these issues, most of the researches focus on reducing the number of charging stations. They do not take the distance between sensor node and charging station into account. Actually, when the distance between sensor node and charging station decreases, the charging efficiency will be increased. Consequently, the charging stations do not need to replenish the sensor nodes' power frequently. Therefore, this thesis proposes a new deploy strategy by taking the number of charging stations and the distance between sensor node and charging station into account simultaneously. We formulate the proposed strategy into a multi-objective problem and employ a non-dominated sorting genetic algorithm-II (NSGA-II) to solve this problem. With the help of NSGA-II method, the generated solutions can avoid falling into a single objective. In our method, we encode all possible locations of charging stations into a set of genes, which is known as a chromosome. Through the phases of crossover and mutation, different chromosomes are generated for maintaining the diversity of the solutions of

next generation. In the selection phase, non-dominated sorting scheme is used to avoid falling into local optimal solutions.

To verify the feasibility of proposed strategy, we implement the simulation environment by using Python programming language. The sensor nodes are distributed randomly on an indoor environment of 20 X 15 square meters. We compare the proposed approach to the simulated annealing-based charging algorithm (SABC) and the layoff simulated annealing-based charging algorithm (LSABC) in terms of the number of charging stations and the overall charging power. The simulation results revealed that the overall charging power obtained using the proposed approach is 5% and 8% higher than that obtained using SABC and LSABC approaches. Moreover, the number of charging stations obtained using NSGA-II is 6% and 1% less than that obtained using SABC and LSABC approaches, respectively.

**Keywords:** Wireless Rechargeable Sensor Networks, Wireless Charging Stations Deployment, NSGA-II

# Contents

中文摘要.....	i
Abstract.....	ii
Contents .....	iv
List of figures.....	v
List of tables.....	vi
Chapter 1. Introduction.....	1
1.1 Research Motivation .....	1
1.2 Contributions.....	3
Chapter 2. Related Work.....	4
Chapter 3. System Model and Problem Formulation .....	7
Chapter 4. NSGA-II based Charging Station Deployment.....	9
4.1 Representation and Initialization .....	10
4.2 Crossover .....	12
4.3 Mutation.....	12
4.4 Selection.....	13
4.4.1 Non-dominated Sorting.....	15
4.4.2 Crowding Distance.....	17
Chapter 5. Simulation and Performance Evaluation.....	21
5.1 Experimental Setting.....	21
5.2 NSGA-II Convergence Experiment.....	23
5.3 Simulation Results .....	25
Chapter 6. Conclusions.....	29
References.....	30

## List of figures

Figure 1. Typical example of WRSN in outdoor environment.....	4
Figure 2. Typical example of WRSN in indoor environment.....	5
Figure 3. 20 x 15 indoor environment .....	7
Figure 4. Flowchart of the proposed mechanism.....	10
Figure 5. Example of a chromosome coding .....	11
Figure 6. Four chromosomes are created in initialize phase.....	11
Figure 7. Example of a crossover .....	12
Figure 8. An example of mutation .....	13
Figure 9. An example of fitness value .....	18
Figure 10. An example of non-dominated sorting .....	19
Figure 11. An example of crowding distance .....	19
Figure 12. Effects of the crossover rate on NSGA-II convergence .....	23
Figure 13. Effects of the mutation rate on NSGA-II convergence .....	24
Figure 14. The number of charging station with sensor nodes increasing from 25 to 125.....	25
Figure 15. The average energy received of each sensor node with sensor nodes increasing from 50 to 125 .....	26
Figure 16. Charging station deployment for different methods under 50 sensor nodes .....	28
Figure 17. Charging station deployment for different methods under 100 sensor nodes .....	28
Figure 18. Charging station deployment for different methods under 125 sensor nodes .....	28

## List of tables

Table 1. Simulation parameters.....	22
Table 2. Charging Station vs. Charging Efficiency in different number of sensor nodes .....	27





# Chapter 1. Introduction

## 1.1 Research Motivation

With the rapid development of Internet of Things (IoT) [1], all devices are gradually able to communicate via the Internet. In order to get more information, the technology of wireless sensor network (WSN), which is composed of wireless sensor nodes and relay nodes, is used extensively. WSN technology can be used in various applications because it has many advantages, such as lower cost, scalability, reliability, accuracy, flexibility, and ease of deployment. As wireless sensor nodes become smarter, smaller and cheaper, billions of wireless sensor nodes are deployed in many application scenarios. For example, sensor nodes can be used to detect, locate or track enemy movements. In addition, it can sense and detect the environment to predict disasters in advance. Moreover, sensor nodes can monitor a patient's health [2]. In the security, sensors can provide vigilant surveillance and increase alertness to potential terrorist attacks [3]. In the future, wireless sensor networks will eventually realize automatic monitoring of forest fires, avalanches, hurricanes, transportation, hospitals, etc. The wide ranges of potential applications for wireless sensor networks have made WSN becoming a fast-growing multi-purpose network [4].

Although the WSN allows users access information conveniently, there are some inherent problems. For example, in WSN, each sensor node senses various kinds of information and then transfer them to relay nodes. All actions consume energy, but the energy of sensor node is limited by battery capacity. When the energy of sensor node is exhausted, they may cause obstacles in the operation of WSN. To solve this network lifetime's problem, wireless rechargeable sensor network (WRSN) [5-7], which is

composed of charging stations, sensor nodes and relay nodes, is a promising approach.

Since wireless charging stations are costly and have limited charging distances, how to deploy the minimal number of charging stations to cover all sensor nodes and satisfy the energy requirements of all sensor nodes are important and challenging issues. Regarding these issues, most of the researches focus on reducing the number of charging stations. But they do not consider the charging efficiency of each sensor node under the same coverage of charging station. Actually, when the distance between sensor node and charging station decreases, the charging efficiency will be increased. Consequently, the charging stations do not need to replenish the sensor nodes' power frequently. Therefore, this thesis proposes a new deploy strategy by taking the number of charging stations and the distance between sensor node and charging station into account simultaneously. We formulate the proposed strategy into a multi-objective problem and employ a non-dominated sorting genetic algorithm-II (NSGA-II) [8] to solve this problem. Note that many approaches have proposed for searching the optimal solution of multi-objective problem, such as Vector Evaluated Genetic Algorithm (VEGA) [9], Weight-based Genetic Algorithm (WBGA) [10] and Multi-objective GA (MOGA) [11]. However, NSGA-II can find the optimal solution better because NSGA-II simultaneously optimize multiple assignment objective instead of searching for possible assignments based on a single composite values.

## 1.2 Contributions

The contributions of this thesis are listed as follows:

- (1) We propose an deploy algorithm by taking both the number of charging station and the distance between the charging station and the sensor node in a WRSN into account simultaneously.
- (2) We formulate the proposed strategy into a multi-objective problem.
- (3) A modified NSGA-II algorithm is proposed to solve a charging station deployment problem.
- (4) A new NSGA-II chromosome coding is provided for problem formulation.

The remaining portion of this thesis is organized as follows. Recent related studies are discussed in chapter 2. Chapter 3 describes the system model and presents the problem formulation. The NSGA-II charging station deployment algorithm is introduced in chapter 4. In chapter 5, we conduct a simulation to verify the applicability of proposed approach and compare it with other prominent methods. Finally, the conclusions are provided in chapter 6.

## Chapter 2. Related Work

In this chapter, we briefly introduce the concept of wireless rechargeable sensor networks and present various methods for the deployment of wireless charging station.

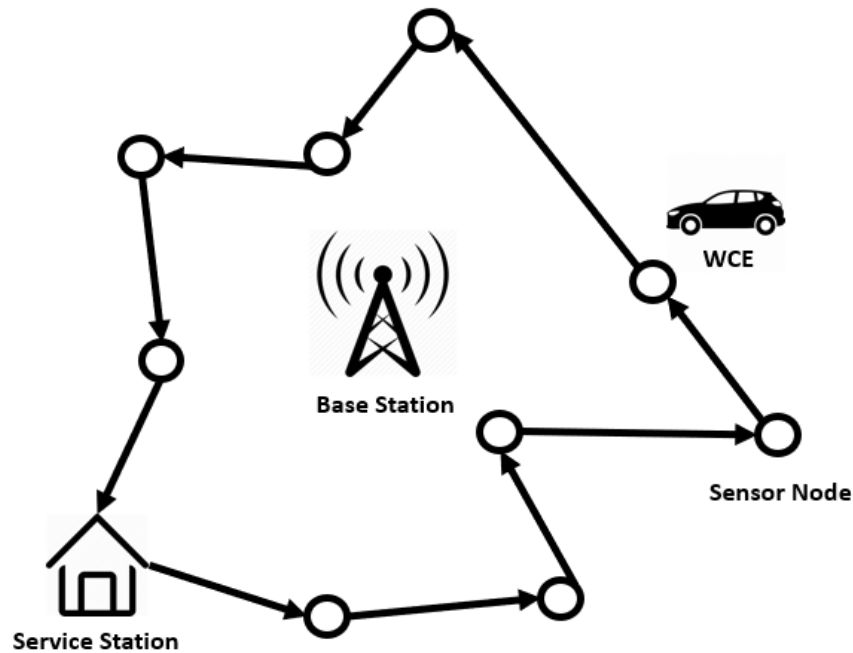


Figure 1. Typical example of WRSN in outdoor environment

The environment of WRSN can be broadly divided into two categories: indoor and outdoor. A typical example of WRSN in outdoor environment is depicted in Figure 1. From the figure 1 we can see that a vehicle carries a wireless charging equipment (WCE) and travels along with the prior path planning to charge the power of sensor nodes. Because the distance of the vehicle's movement and the lifecycle of sensor nodes are limited, how to plan the vehicle's motion path is an important issue. The vehicle motion path planning needs to ensure that sensor nodes do not fail to deplete the WSN due to energy depletion, and charge as more sensor nodes as possible under the limited energy of WCE. In order to allow the WCE to travel further distance, Zhang *et al.* [12] proposed a Push-Wait mechanism, where WCEs are allowed to intentionally transfer

energy between themselves. However, this approach needs too many WCEs. Liu *et al.* [13] proposed a Push-Shuttle-Back mechanism to allow that the WCE can go back to base station halfway for replenishing energy. With this mechanism, the energy loss in the movement and charging processes between WCEs can be reduced. In addition, the number of WCEs also can be reduced. In terms of path planning, Lyu *et al.* [14] proposed a periodic charging planning for a mobile WCE with limited traveling energy. This periodic charging planning ensures that the energy of the nodes in the WRSN varies periodically and that nodes perpetually fail to die. The authors proposed a Hybrid Particle Swarm Optimization Genetic Algorithm (HPSOGA) to solve this NP-hard problem.

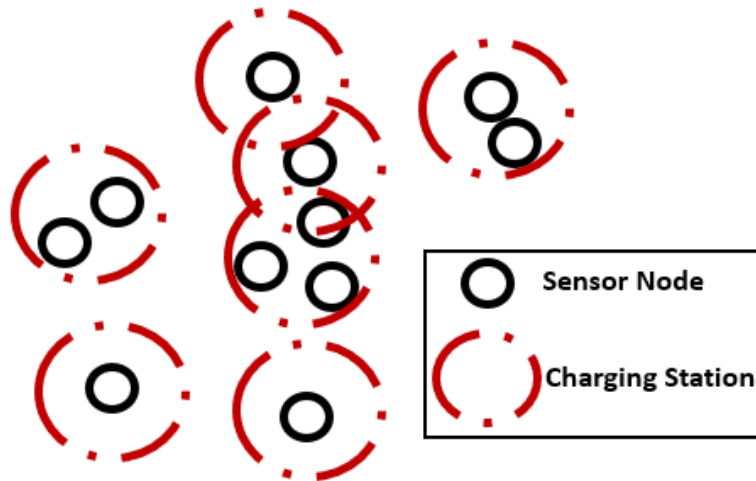


Figure 2. Typical example of WRSN in indoor environment

A typical example of WRSN in indoor environment is shown in Figure 2. In indoor environment of WRSN, the charging station deployment needs to consider the sensor node position, radio frequency interference and charging efficiency. A good charging stations deployment intends to minimize the number of deployed charging station under the requirement of covering all sensor nodes. Jian *et al.* [15] proposed a movable-charger-based algorithm (MCBA), which is using overlapping area of charging antenna

covered area, to find out some candidate locations deploying of charging stations and then record them to a set. With the set of candidate locations, the authors employed a greedy algorithm to search the final deploying location of charging stations. It chooses the candidate position which cover the most sensor nodes. According to this rule, greedy algorithm is difficult to find better solution because it is easier to fall into local optimum. Similarly, based on the candidate set obtained from MCBA, Chien *et al.* [16] proposed a simulated anneal (SA) algorithm to find the final charging station locations. The main concept of SABC is that he can accept candidate nodes with a small number of sensor nodes. At the beginning of the process, a charging station is randomly selected from the candidate nodes, and the temperature is determined during each iteration to decide whether to add a charging station or replace the original charging station. This method has more directions for finding a solution, so there is a higher chance of finding a good solution. However, during the SABC iteration process, unnecessary solutions are often found repeatedly. If unnecessary solutions can be eliminated in the process, the convergence of the algorithm can be accelerated. Furthermore, Chien *et al.* [17] used the layoff algorithm to eliminate the unnecessary solutions during SA iterations. Different from SABC, LSABC initially treats all candidate nodes as all placements, and then randomly selects one candidate point for each iteration to eliminate. If the result is better, it replaces the original solution. If not, the candidate point is retrenched. Consequently, the computation time can be reduced.

### Chapter 3. System Model and Problem Formulation

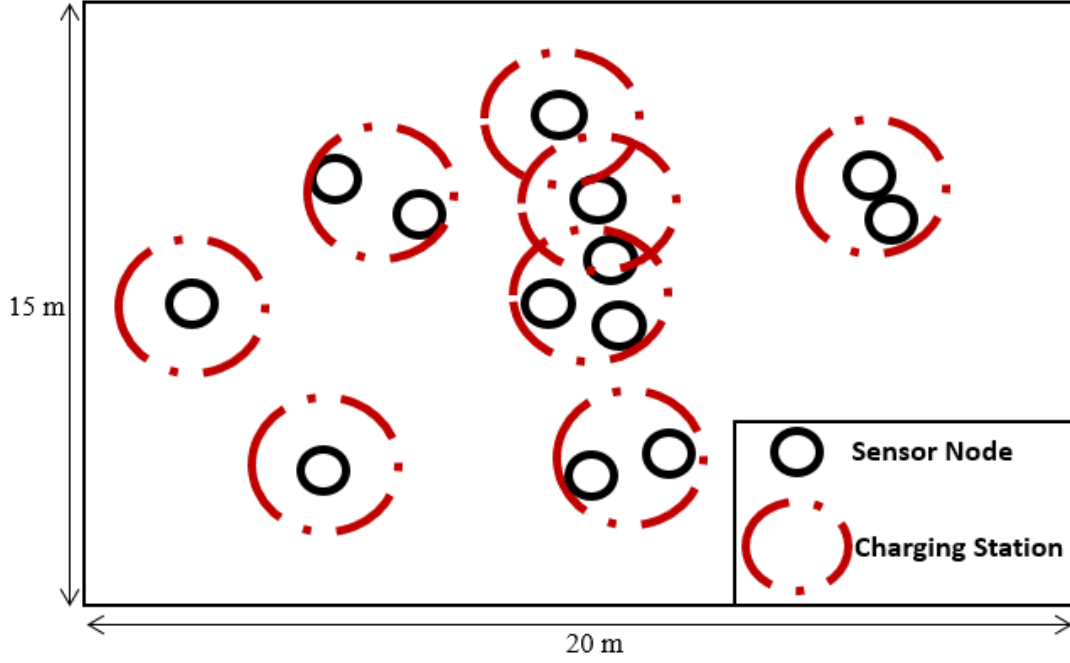


Figure 3. 20 x 15 indoor environment

In this study, we construct a 20 X 15 square meters indoor environment with  $n$  wireless rechargeable sensor nodes and  $m$  wireless directional charging stations which is shown in Figure 3. Let  $S = \{s_1, \dots, s_n\}$  be the set of all the wireless rechargeable sensor nodes and  $C = \{c_1, \dots, c_m\}$  be the set of all the charging stations. Those sensor nodes are randomly distributed in the three-dimensional region. Every directional charging station is identical and can charge a circular area with  $R$ -radius. Denote  $u_j$  as the number of sensor nodes covered by charging station  $c_j$  and  $U$  as the set of  $u_j$ , where each  $u_j$  is greater than equal to one. Because the efficiency of power transmission decreases as the distance of power transmission increases, the sensor nodes which are out of this  $R$ -radius circle cannot receive the effective power. A sensor node may be covered by multiple charging stations and receives power from multiple

charging stations simultaneously. This thesis aims to minimize the number of charging stations and maximize the overall charging capacity. To do that, we define the following problem.

$$\begin{cases} \text{minimize } m \\ \text{maximize } C_{rate} = \frac{\sum_{i=1}^n s_i}{n} \\ \text{maximize } P = \sum_{i=1}^n p_i \end{cases} \quad (1)$$

subject to  $d_{t,i} \leq E, \forall u_j \geq 1$ .

Note that  $s_i$  represents the sensor node  $i$  which is covered by any charging station,  $C_{rate}$  represents the cover ratio of total charging stations. If sensor node  $i$  is covered by any charging station,  $s_i$  is equal to 1, otherwise,  $s_i$  is equal to 0. For the sensor node  $i$ , the total amount power received from multiple charging stations is denoted as  $p_i$ .  $E$  denotes the effectual charging distance which is defined by charging equipment.  $d_{t,i}$  denotes the distance between sensor node  $i$  and charging station  $t$ .

To measure  $p_i$ , the following model which is provided by the company Powercast [18] is employed:

$$p_i = \sum_{t=0}^T p_{t,i} g_t g_i \left( \frac{\lambda}{4\pi R_{t,i}} \right)^2, \quad (2)$$

,where  $p_{t,i}$  represents the power which transfer from charging station  $t$  to the sensor node  $i$ ;  $g_t$  denotes the value of antenna gains of charging station  $t$ ;  $g_i$  represents the value of antenna gains of sensor node  $i$ ;  $\lambda$  denotes the wavelength of radio frequency (RF);  $R_{t,i}$  denotes the distance between sensor node  $i$  and charging station  $t$ .



## Chapter 4. NSGA-II based Charging Station Deployment

In proposed method, all candidate deploying locations of charging station are discovered by applying MCBA [15] and then record them to a set. Based on this set, a modified NSGA-II scheme are performed to find a solution for the charging station deployment problem. Proposed by Deb *et al.* [8] on the basis of NSGA, NSGA-II improves the nondominated sorting algorithm and reduces the computational of parents and children with elitist strategy, introduces the crowded comparison operator to improve diversity of solutions, and avoids the use of niched operators. We encode all candidate locations of charging station into a set of genes, which is known as a chromosome or an individual. In our scheme, four primary phases are performed – initialization phase, crossover phase, mutation phase and selection phase. Individuals are created randomly in the initialization phase. In the crossover phase, genes are copied and are delivered to offspring. In the mutation phase, genes change their information content. Through the phases of crossover and mutation, different chromosomes are generated for maintaining the diversity of the next generation of solutions. In the selection phase, a modified non-dominated sorting scheme is used to preserve the diversity of different objectives. After many generations, the strongest genes are obtained. The chromosomes are updated continually in the main loop until the stopping criterion is met. The flowchart of the proposed NSGA-II approach is displayed in Figure 4.

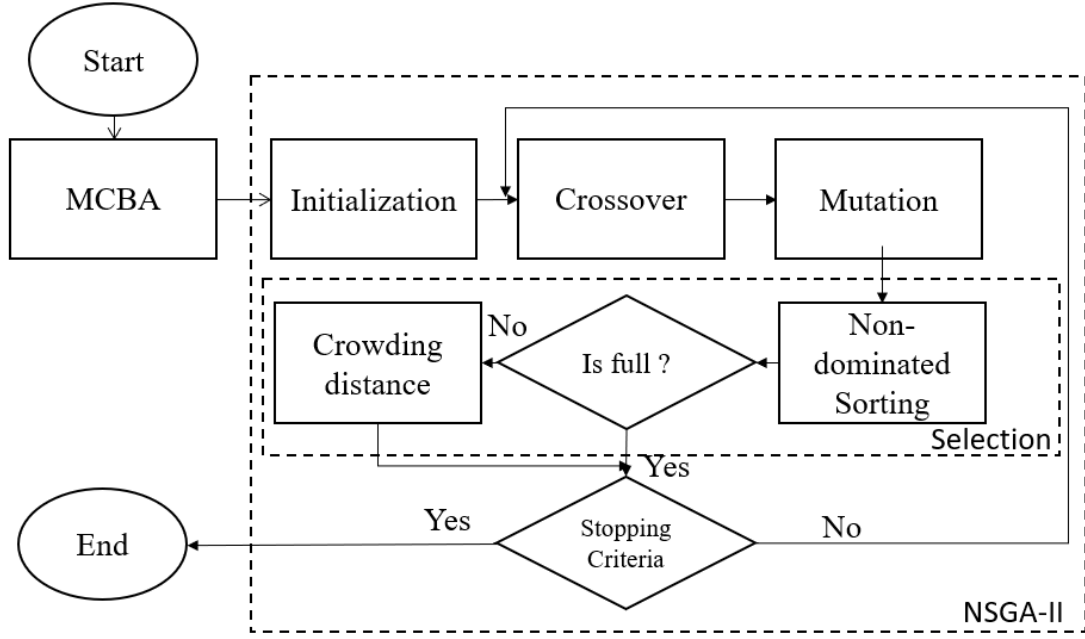


Figure 4. Flowchart of the proposed mechanism

#### 4.1 Representation and Initialization

In the proposed approach, we assume that each sensor node can be covered by multiple charging stations and receives power from multiple charging stations simultaneously. The WRSN deployment space is regarded as a cuboid with the dimension  $(L, E, H)$ , where  $L$  is the length of cuboid,  $W$  is width of cuboid and  $H$  is height of cuboid. In order to efficiently avoid the interference from obstacles and consider the actual situation, we deploy the charging station at ceiling as well as the sensor nodes are randomly distributed in the floor. On the basis of these assumptions, a sensor node is represented as a point  $s_i$  with coordinates  $(x_i, y_i, z_i \text{ with } z_i = 0)$  and a charging station is denoted as a point  $c_j$  with coordinates  $(x_j, y_j, z_j \text{ with } z_j = 0)$  in three-dimensional space. By applying the MCBA algorithm, we can get a set  $V$  which includes all the places where the charging station may be deployed. Each candidate point is considered as a genetic cell.

A gene, which is a component of a chromosome, has two indicators,  $\alpha$  and  $\beta$ . Note that  $\alpha$  stores the candidate position of charging station with  $(x, y)$  and  $\beta$  represents this candidate point is deployed or not, where  $\beta = 1$  means this candidate point is deployed and  $\beta = 0$  represents the candidate point is not deployed. Figure 5 presents the process of transforming a candidate set into a chromosome. The coding example for a WRSN includes five sensor nodes and candidate locations.

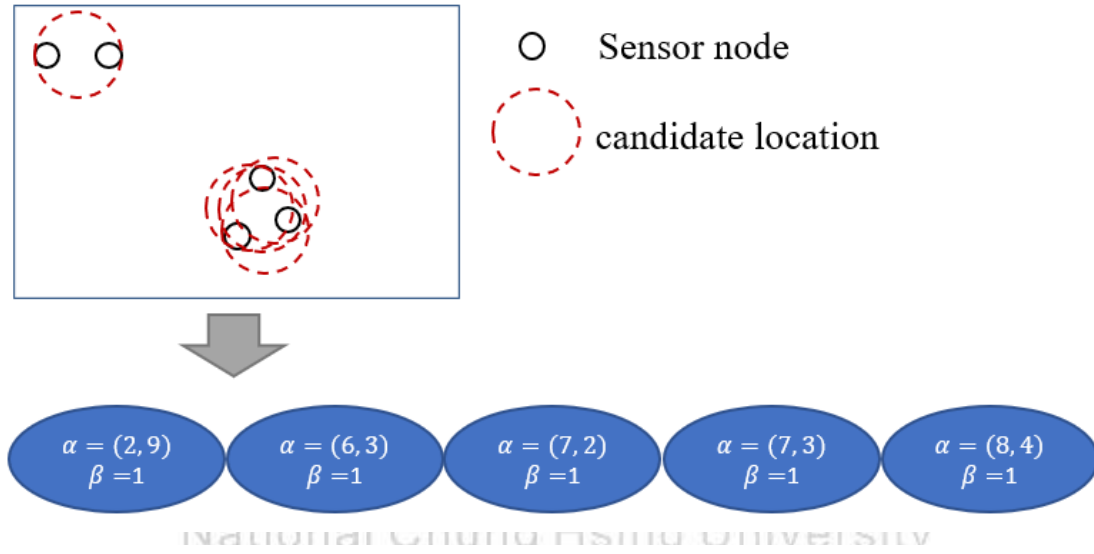


Figure 5. Example of a chromosome coding

In the initialization phase,  $N_i$  individuals are created randomly. For example, in Figure 6, we randomly generate four chromosomes.

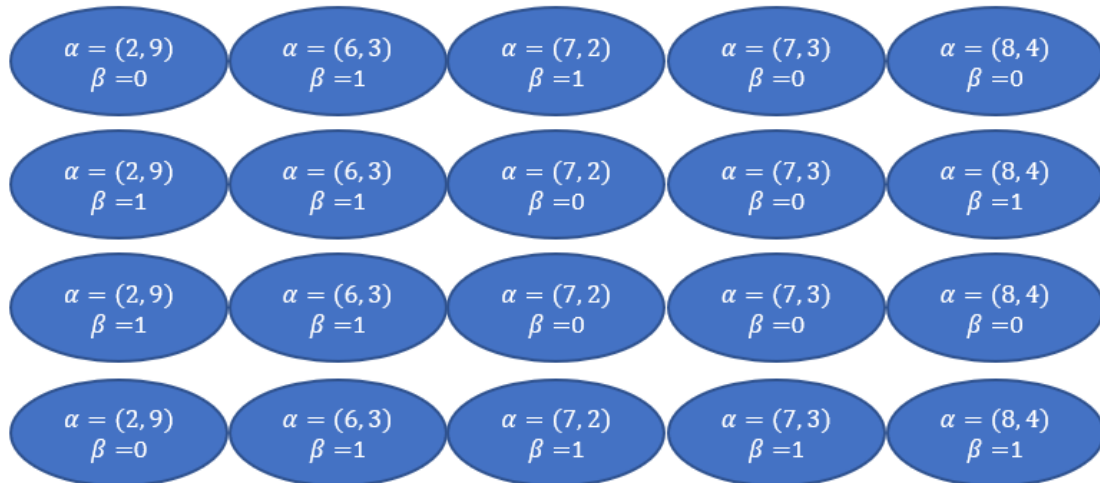


Figure 6. Four chromosomes are created in initialize phase

## 4.2 Crossover

In the crossover phase, switching point is randomly selected. At each iteration, parents selected after replication are exchanged for genetic, so that the new offspring can retain some of the characteristics of the parents. In order to avoid a large number of genetic changes caused by single-point mating, in this study, two-point mating was used. The process is determined by the crossover rate  $r_c$ , which is a floating number between zero and one. If the generated random number is less than  $r_c$ , two replicating parents are selected randomly. Two crossover point on this chromosome are also generated randomly and then exchange the genetic at the crossover point. An example of the crossover is presented in Figure 7.

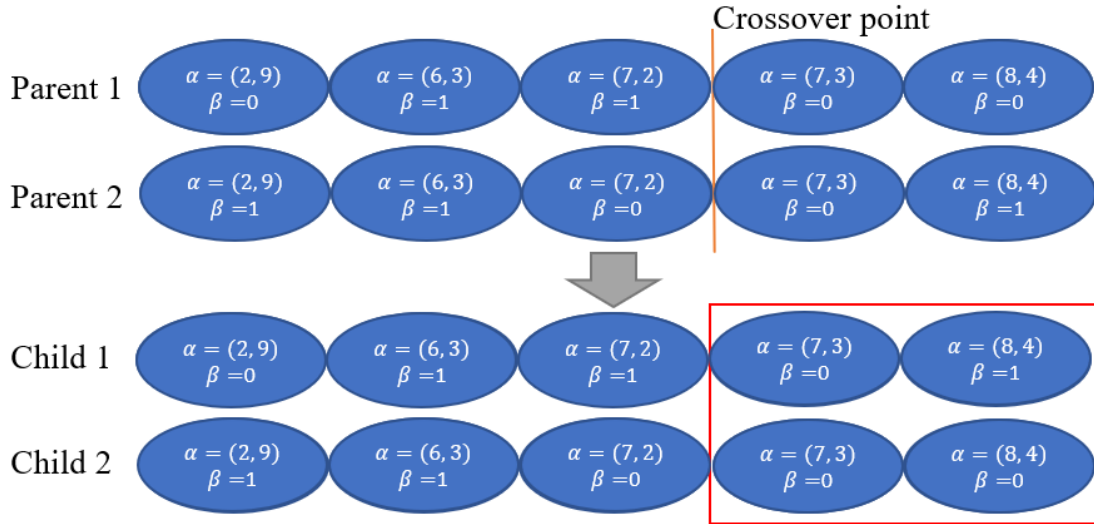


Figure 7. Example of a crossover

## 4.3 Mutation

The purpose of mutation is to generate the genetic that have not appeared in the parents, in order to prevent the best solution for falling into the local optimal solution. In this study, single-point mutation was used. Similar to the crossover phase, the mutation phase is determined by the mutation rate  $r_m$ , which is a floating number between zero and one. If the generated random number is less than  $r_m$ , then one gene

of the chromosome is randomly selected for mutation. In this case, if the selected point is 0, it will become 1. The opposite if the selected point is 1, it will become 0. An example of mutation is presented in Figure 8.

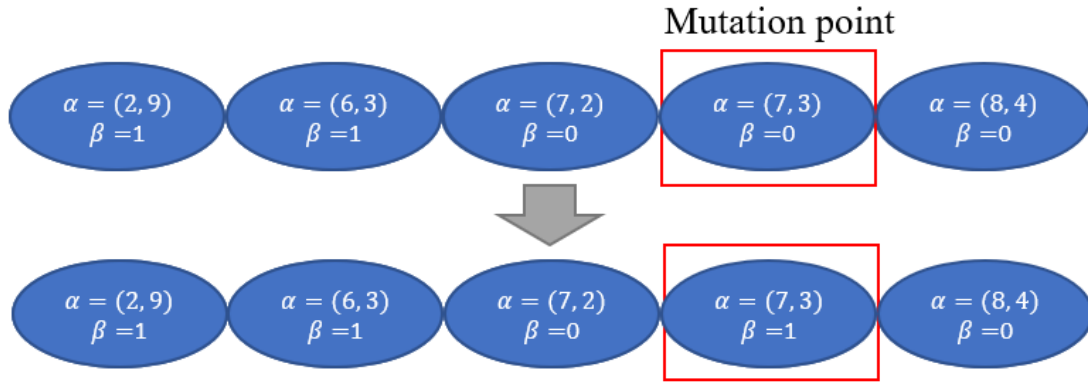


Figure 8. An example of mutation

#### 4.4 Selection

In the selection phase, all chromosomes are decoded to obtain their information, such as the location of candidate nodes, the location is deployed or not. These parameters are used to calculate the fitness value through fitness function.

In fitness function, three metrics are considered, the number of charging stations, the total coverage of charging stations and the sum of received energy. To know the number of charging stations for each chromosome, the minimize  $m$  is given by

$$\text{minimize } m = \sum_{i=1}^l \beta_i, \quad (3)$$

where  $l$  is the number of genes in chromosome,  $\beta_i$  equal to 1 or 0.

The total coverage of charging stations  $C_{rate}$  is used to obtain the charging stations' cover ratio. It's expressed as

$$\text{maximize } C_{rate} = \frac{\sum_{i=1}^n s_i}{n}, \quad (4)$$

where  $s_i$  is the sensor node  $i$  covered by any charging station.

We can use the Friis transmission equation [19] to get the energy received by each sensor node. The sum of the received energy is

$$\text{maximize } P = \sum_{i=1}^n p_i, \quad (5)$$

where  $p_i$  represents sensor node  $i$  received the total amount power from multiple charging stations. We can obtain it value through Eq. (2).

From the initial population  $N_i$  and the population  $Q_i$  obtained after the crossover and mutation phase, it becomes a  $N_i + Q_i$  population. In order to retain the first 50% of the population into the next generation, we must calculate the non-dominated sorting and the crowding distance. After non-dominated sorting, all chromosomes are classified into different levels and all levels rank by ascending order. If they are belonged to the same level, they will be ranked according to the crowding distance.

#### 4.4.1 Non-dominated Sorting

A non-dominated solution means that this solution cannot be dominated by other solutions. In other words, all objective function values of other solution can be greater than this solution. The main purpose of non-dominated ordering is to divide the initial population  $N_i$  into several groups of non-dominated solution sets according to the objective function value of each chromosome. A pseudocode for non-dominated sorting is presented in Algorithm 1. To perform this algorithm, we need to input a set of population  $P$  with three parameters  $N$ ,  $C_{rate}$  and  $E$ , where  $N$  represents the set of the number of charging station for each chromosome,  $C_{rate}$  denotes the set of the total coverage of charging stations for each chromosome and  $E$  represents the set of the sum of the received energy for each chromosome. For each solution we calculate two entities,  $n_p$  and  $S_p$ , where  $n_p$  represents the number of solutions which dominate the solution  $p$  and  $S_p$  denotes the set of solutions that the solution  $p$  dominates. In Algorithm 1, we first calculate the number of chromosomes dominated by other chromosomes (lines 1-17). Note that  $F$  is the nondominated front. If  $n_p$  is 0, which means that the chromosome is not dominated by other chromosomes, they are defined as level 1 and are removed from the population (lines 18-21). Next, we calculate the number of chromosomes dominated by other chromosomes from the remaining population.  $Q$  is used to store the members of next front. If the number of chromosomes dominated is 0, they are defined as level 2 (lines 23-33). Continue the process until the entire population is sorted.

---

**Algorithm 1:** Non-dominated Sorting

---

**Input** : list  $P[N, C_{rate}, E]$   
**Output:** list  $F$

```
1 for each  $p \in P$  do
2    $S_p = \emptyset$ 
3    $n_p = 0$ 
4   for each  $q \in P$  do
5     if  $q[C_{rate}] < p[C_{rate}]$  then
6        $S_p = S_p \cup q$ 
7     else if  $q[C_{rate}] = p[C_{rate}]$  then
8       if  $(q[N] < p[N] \text{ and } q[E] < p[E]) \text{ or } (q[N] < p[N] \text{ and } q[E] = p[E]) \text{ or } (q[N] = p[N] \text{ and } q[E] < p[E])$  then
9          $S_p = S_p \cup q$ 
10      else
11         $n_p = n_p + 1$ 
12      end if
13    else
14       $n_p = n_p + 1$ 
15    end if
16  end for
17 end for
18 if  $n_p = 0$  then
19    $p_{rank} = 1$ 
20    $F_1 = F_1 \cup p$ 
21 end if
22  $i = 1$ 
23 while  $F_i \neq \emptyset$  do
24    $Q = \emptyset$ 
25   for each  $q \in S_p$  do
26      $n_q = n_q - 1$ 
27     if  $n_q = 0$  then
28        $q_{rank} = i + 1$ 
29        $Q = Q \cup q$ 
30     end if
31      $i = i + 1$ 
32      $F_i = Q$ 
33   end for
34 end while
35 return  $F$ 
```

---



#### 4.4.2 Crowding Distance

According to the ranks sorted in the previous step, the chromosomes of the same rank are taken out and the crowding distance is calculated. The concept of crowding distance is the denseness between the chromosome and its surrounding chromosomes. When the crowding distance is smaller, it means that the chromosome falls in a relatively crowded range. If the crowding distance is larger, it means that the chromosome falls in a relatively loose range. The formula for the crowding distance of the  $i$ th chromosome is as follows:

$$CD_i = \sum_{o=1}^K \frac{P_o(x_{i+1}) - P_o(x_{i-1})}{P_o^{max}(x) - P_o^{min}(x)} \quad i = 2, 3, \dots, W - 1, \quad (6)$$

where  $K$  is the number of target formulas,  $o$  is the value of one of the objective functions,  $i$  is the number of chromosomes,  $W$  is the last chromosome in this level,  $P_o(x_{i+1})$  and  $P_o(x_{i-1})$  are the previous chromosome objective function value and the next chromosome objective function value of chromosome  $i$  under the  $o$ th target. In each level, the chromosome distributed at the two ends set their crowding distance to infinity. A pseudocode for crowding distance is presented in Algorithm 2.

---

<b>Algorithm 2:</b> Crowding distance	
<hr/>	
<b>Input</b> : list $P[N, C_{rate}, E], F$	
<b>Output:</b> list $P_{distance}$	
1	$l =  P $
2	<b>for</b> each $i \in P$ <b>do</b>
3	$P[i]_{distance} = 0$
4	<b>end for</b>
5	<b>for</b> each object $o$ <b>do</b>
6	$P = sort(P, o)$
7	$P[0]_{distance} = P[l - 1]_{distance} = \infty$
8	<b>for</b> $i = 1$ to $(l - 1)$ <b>do</b>
9	$P[i]_{distance} = P[i]_{distance} + (P[i + 1]_o - P[i - 1]_o) / (P_o^{max} - P_o^{min})$
10	<b>end for</b>
11	<b>end for</b>
12	<b>return</b> $P_{distance}$

---

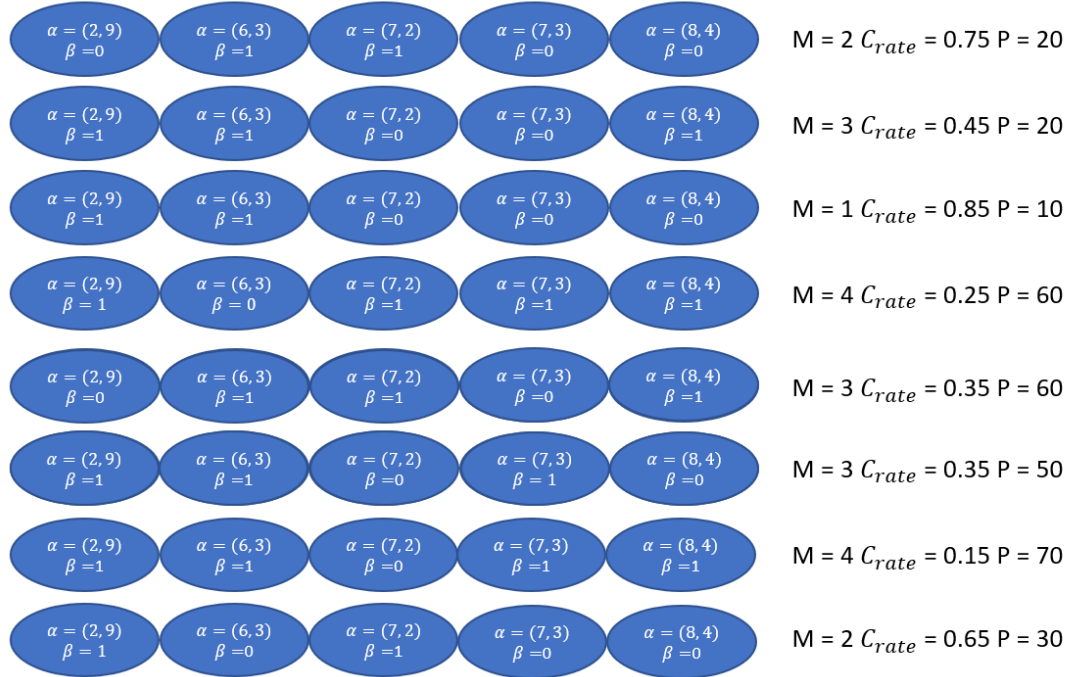


Figure 9. An example of fitness value

After the crossover and mutation phase, we get four new chromosomes and calculate the fitness value through the Eq. (3)(4)(5) together with the original chromosomes as shown in figure 9. Then we find the non-dominated set for all chromosomes and sort them according to the non-dominated set as shown in figure 10. Level 1 indicates that the chromosome is not dominated by any chromosome, level 2 indicates that the chromosome is dominated by a chromosome, and so on. In the next generation, we must pick out four new chromosomes, but we can see that there are 5 chromosomes in level 1, so we need to calculate the crowding distance of these five chromosomes.

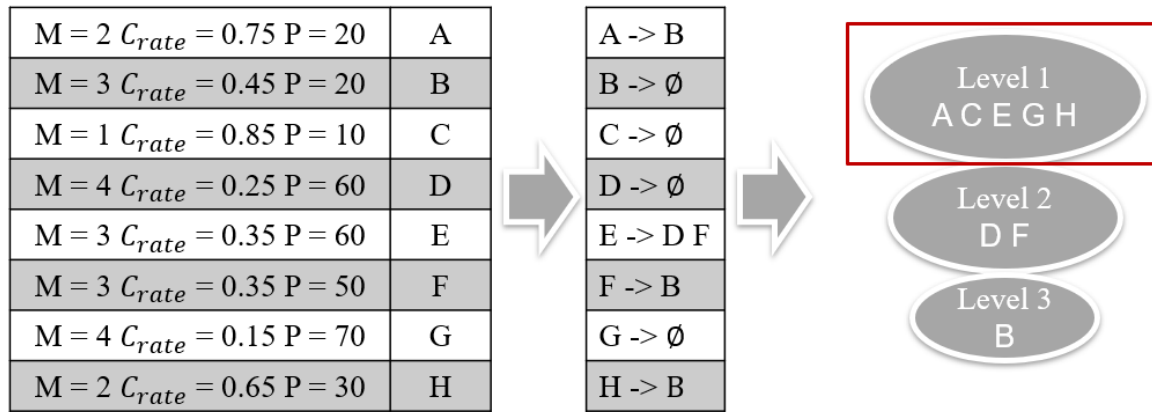


Figure 11. An example of non-dominated sorting

Before calculating the crowding distance, we must sort each target as shown in figure 11. After finishing the sorting, we can apply Eq. (6) to calculate the crowding distance.

For example, chromosome A =  $\frac{(2-1)}{(4-1)} + \frac{(0.85-0.65)}{(0.85-0.15)} + \frac{(30-10)}{(70-10)}$ . After the calculation, we take out the first four with large values.

The larger the value of the chromosome, the lower the similarity of the chromosome with other chromosomes

M = 2 $C_{rate} = 0.75$ P = 20	A
M = 1 $C_{rate} = 0.85$ P = 10	C
M = 3 $C_{rate} = 0.35$ P = 60	E
M = 4 $C_{rate} = 0.15$ P = 70	G
M = 2 $C_{rate} = 0.65$ P = 30	H

Sort	
M	C A H E G
	1 2 2 3 4
$C_{rate}$	G E H A C
	0.15 0.35 0.65 0.75 0.85
P	C A H E G
	10 20 30 60 70

Figure 10. An example of crowding distance

---

**Algorithm 3:** modified NSGA-II

---

**Input** :  $population, r_c, r_m, iter$   
**Output:**  $final$

```
1  $i = 1$ 
2  $parents = Initiation(population)$ 
3 while  $i \neq iter$  do
4    $offspring = Crossover(parents, r_c)$ 
5    $offspring = Mutation(offspring, r_m)$ 
6    $family = parents + offspring$ 
7    $P[N, C_{rate}, E] = Evaluation(family)$ 
8    $F = NondominatedSorting(P)$ 
9    $CrowdingDistance(F, P)$ 
10  Select the first 50% of chromosomes as the next generation after
    CrowdingDistance
11   $i = i + 1$ 
12 end while
13 Choose the smallest N as the final answer
14 return  $final$ 
```

---

The process of proposed NSGA-II is shown in Algorithm 3. In the input parameters,  $population$  represents the amount of initial chromosome population;  $iter$  denotes a value for stopping criteria;  $r_c$  and  $r_m$  represent the rate of crossover and mutation, respectively;  $final$  is the output solution and represents the deployment positions of charging stations.

## Chapter 5. Simulation and Performance Evaluation

To verify the feasibility of proposed strategy, we simulate the proposed approach in a cubic indoor environment,  $15 * 20 * 2.3 (m^3)$ , by using Python programming language. In section 5.1, we describe the experimental setting. NSGA-II convergence experiments are demonstrated in section 5.2 afterwards. Finally, we compare the proposed method with Jian *et al.*'s simulated annealing-based charging algorithm (SABC) approach and Jian *et al.*'s layoff simulated annealing-based charging algorithm (LSABC) approach in terms of the number of charging stations and the overall charging power in section 5.3.

### 5.1 Experimental Setting

In the simulation experiment, the sensor nodes are randomly deployed on the ground. For a charging station, the effectual charging distance is 3(m) and the angle  $\theta$  is set to  $30^\circ$  [20]. Number of sensor nodes are set, varying from 25 to 125 with increment of 25. According to [18], frequency of charging station is set as 915 MHz, the transmission power is set as 3 W EIPR and receiver antenna gain is set as 6 dBi. Note that as the same with other researches, we assume that the charging efficiency is not affected by the number of sensor nodes. In other words, the time required to charge multiple power-depleted sensor nodes is the same as to charge a single one [21, 22]. The size of the population almost does not affect the result of experiment but the calculation time will increase. Therefore, we chose a population of 10 as the experimental setting. Before performing the comparisons between the proposed approach and other approaches, the crossover rate and mutation rate should be identified for the proposed NSGA-II approach. After experiments, it is found that the algorithm will reach a stable convergence after 1000, so we set iteration to 1000. The parameters details are shown

in Table 1.

Table 1. Simulation parameters

Parameters	Setting
Size of venue	15 x 20 x 2.3 $m^3$
Number of sensor nodes	25 ~ 125
Effectual charging distance	3 $m$
Transmitted Power	3 $W$ EIRP
Receiver Antenna Gain	6 $dBi$
Frequency	915 $MHz$
Population	10
Crossover rate	1
Mutation rate	0.8
Iteration	1000

## 5.2 NSGA-II Convergence Experiment

We observe the convergence behavior of the number of charging stations under different crossover rate and mutation rate. In the experiment, the default values of  $r_c$ ,  $r_m$  and  $population$  were set as 1, 0.8 and 10, respectively. First, we varied the values of  $r_c$  from 0.2 to 1 with increments of 0.2 to observe the effects on the NSGA-II convergence. The simulation result is presented in Figure 12. From the figure, we observe that the number of charging stations is smallest when  $r_c$  is equal to 1. As a result, we set the crossover rate as 1. This means that every time you will go through the steps of crossover.

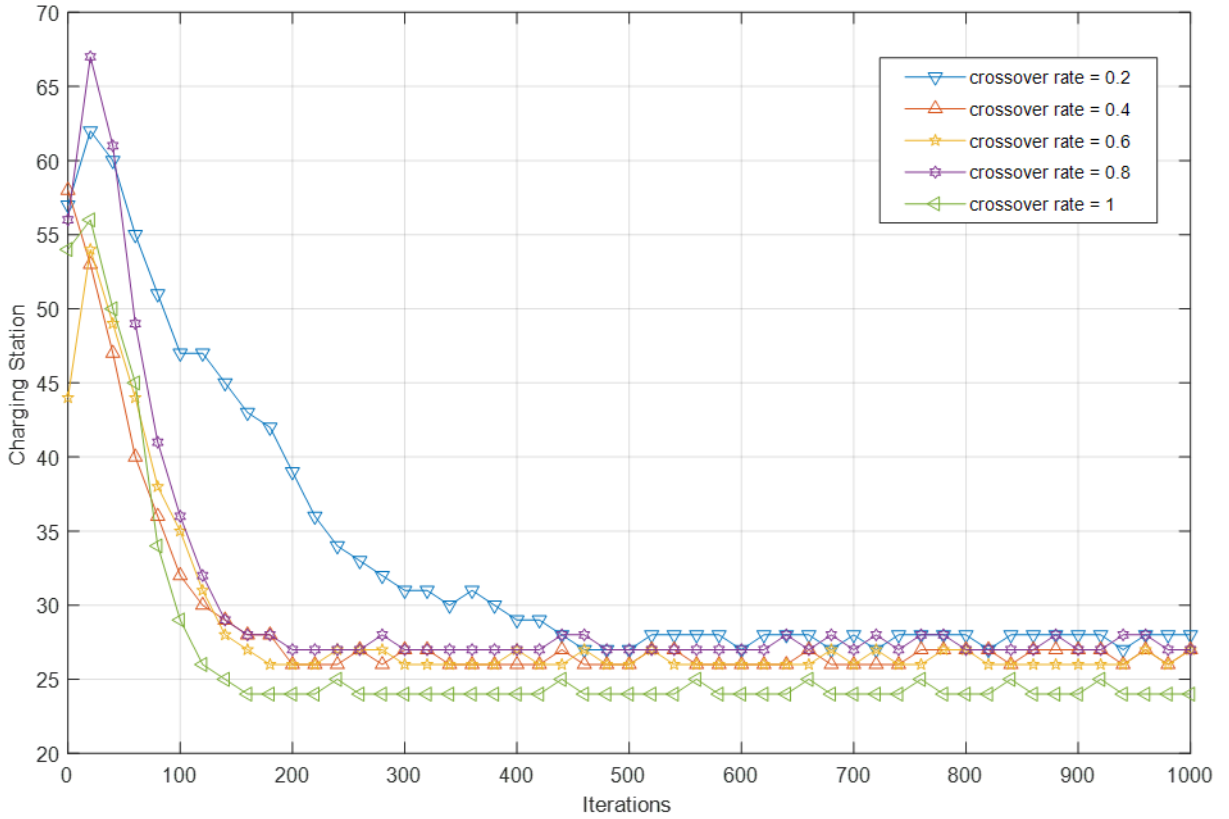


Figure 12. Effects of the crossover rate on NSGA-II convergence

Next, we varied the values of  $r_m$  from 0.2 to 1 with increments of 0.2 to observe the effects on the NSGA-II convergence. The simulation result is presented in Figure 13. From the figure, we observe that the number of charging stations is smallest as  $r_m$  is equal to 0.8. As a result, we set the mutation rate as 0.8.

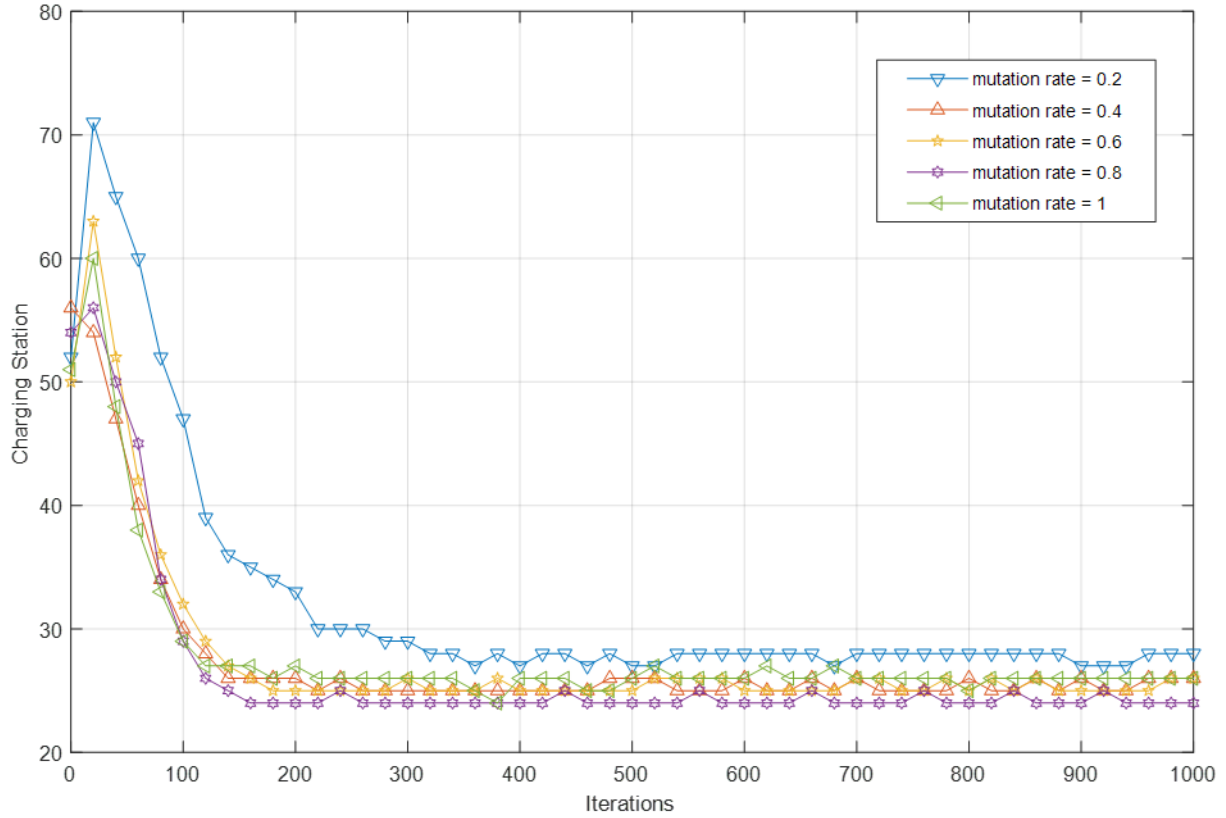


Figure 13. Effects of the mutation rate on NSGA-II convergence



### 5.3 Simulation Results

In our experiments, the simulations were executed 30 times. We measured the average number of charging stations and the average received energy of every sensor node. The Figure 14 shows the comparison of number of chargers with NSGA-II, LSABC and SABC. X-axis represents the number of sensor nodes, and Y-axis represents the number of charging stations. Obviously, the number of charging station of NSGA-II and LSABC is lower than SABC and this phenomenon become evident increasingly when the number of sensor nodes is increased. This is because NSGA-II uses the mechanisms of crossover and mutation and LSABC uses the layoff algorithm to avoid falling into local optimum.

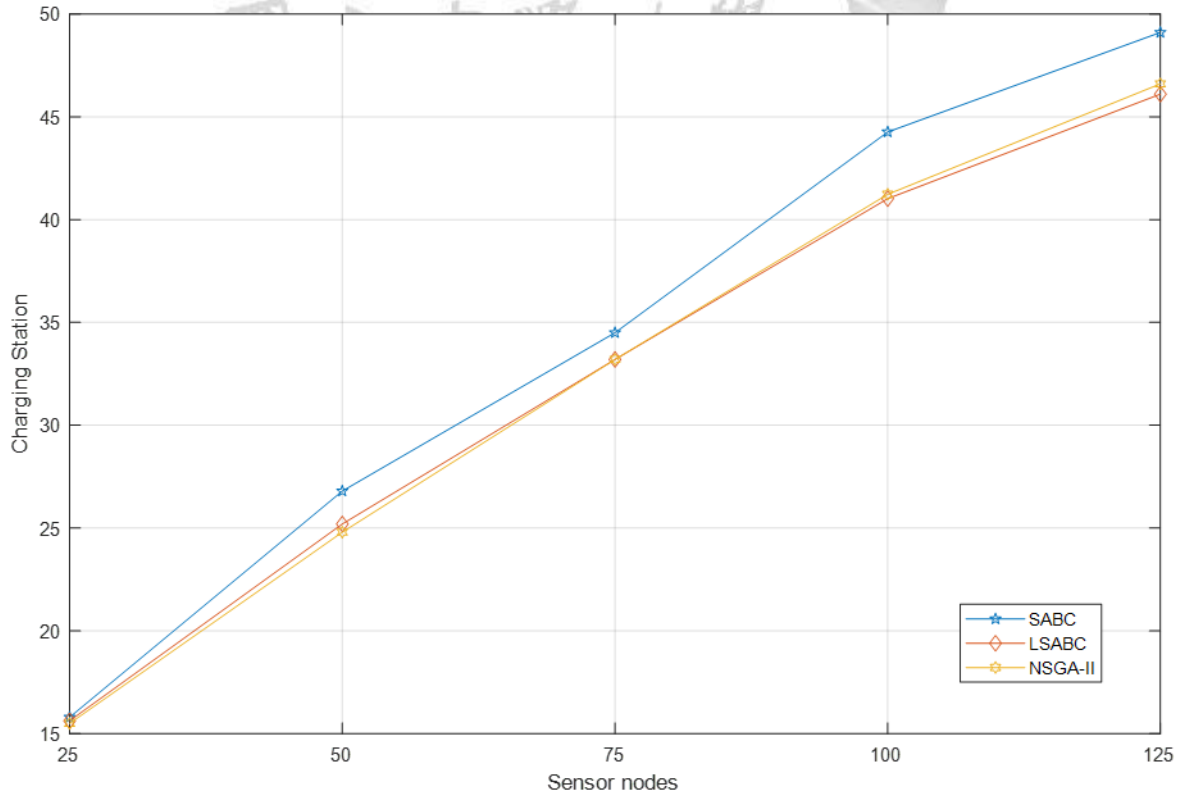


Figure 14. The number of charging station with sensor nodes increasing from 25 to 125

The Figure 15 shows the comparison of the average energy received of each sensor node. X-axis represents the number of sensor node, and Y-axis represents the average energy received of each sensor node ( $mW$ ). The simulation results revealed that NSGA-II can receive more power than LSABC and SABC, because LSABC and SABC do not take the distance between the charging station and the sensor node into account. When the distance is closer, the sensor can receive more energy.

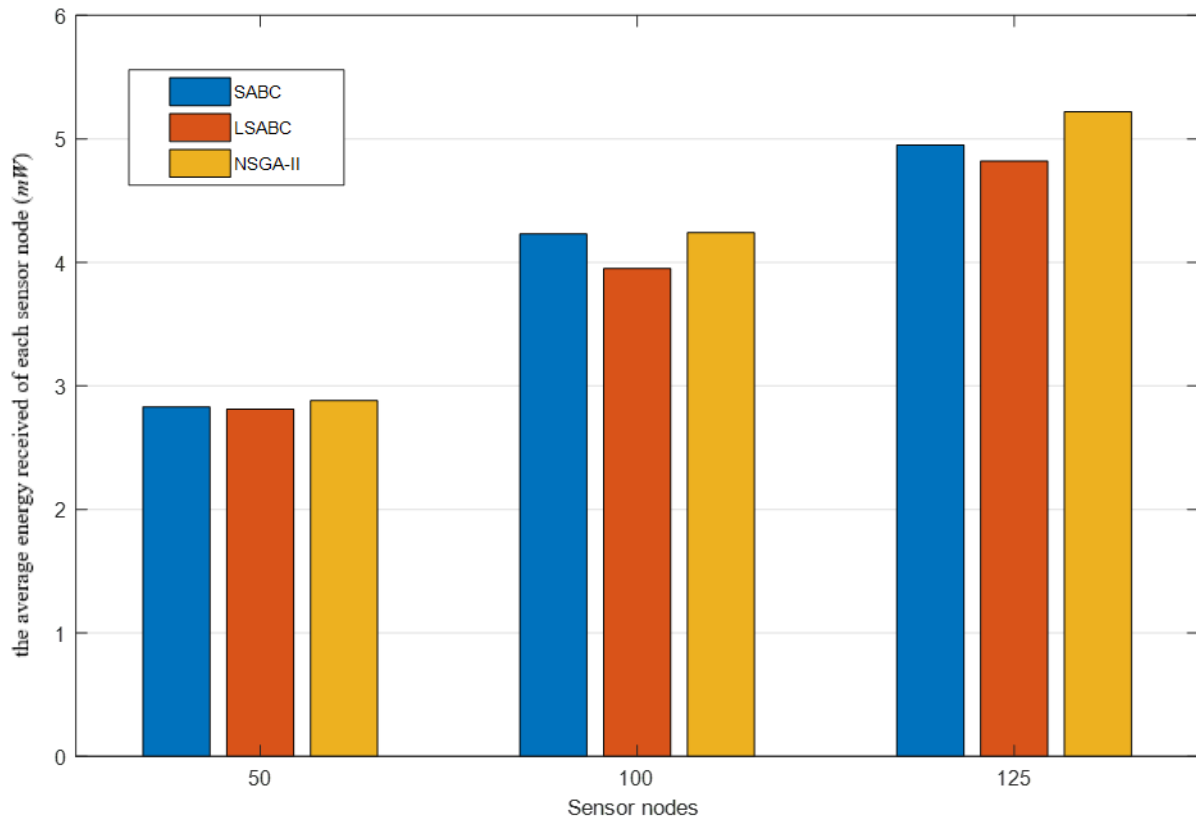
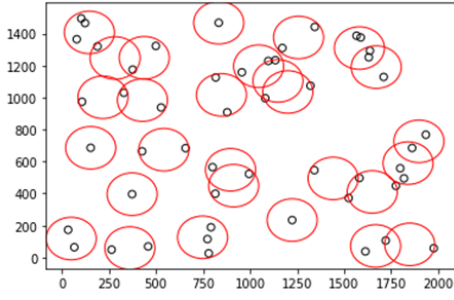


Figure 15. The average energy received of each sensor node with sensor nodes increasing from 50 to 125

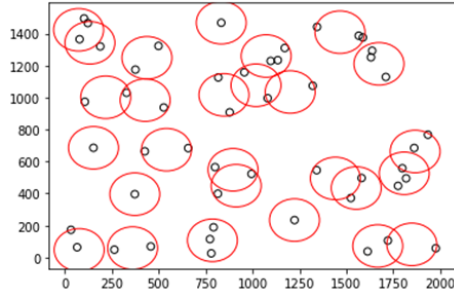
We also compare the number of charging stations required by different methods under different sensor nodes and the overall sensor nodes energy received. The details are shown in Table 2. We can be seen that under the same number of charging stations, the overall energy received by the NSGA-II method is greater than that of other methods. Figures 16-18 show where to place the charging station for different number of sensor nodes

Table 2. Charging Station vs. Charging Efficiency in different number of sensor nodes

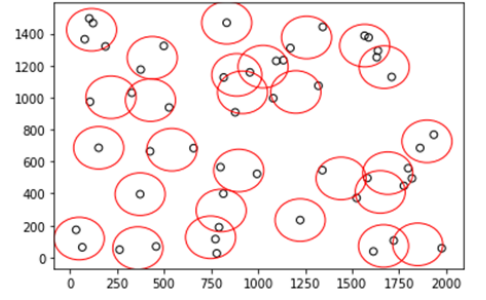
sensor nodes	SABC		LSABC		NSGA-II	
	Charging station	Energy( $mW$ )	Charging station	Energy( $mW$ )	Charging station	Energy( $mW$ )
25	15	34.886	15	34.899	15	34.907
50	28	79.272	27	76.1	27	77.94
75	34	122.513	34	119.305	33	121.703
100	42	177.8	41	162.31	40	169.83
125	45	222.94	45	217.12	45	235.25



(a) 28 charging stations using SABC

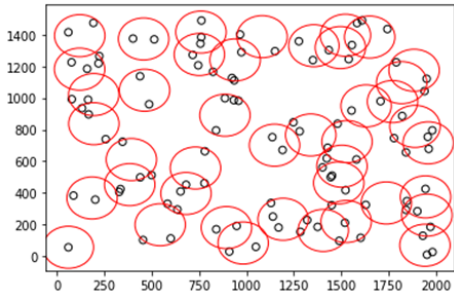


(b) 27 charging stations using LSABC

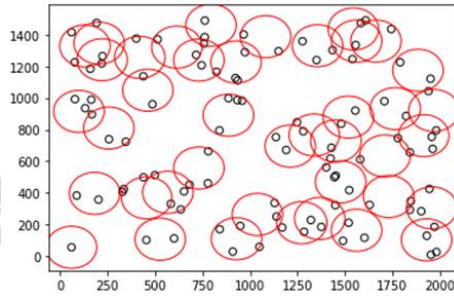


(c) 27 charging stations using NSGA-II

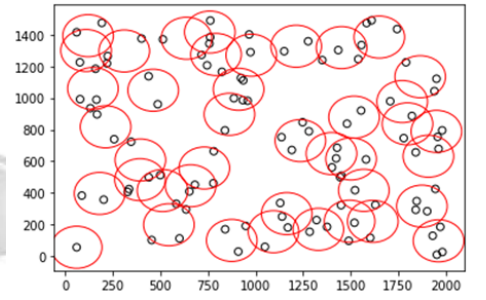
Figure 16. Charging station deployment for different methods under 50 sensor nodes



(a) 42 charging stations using SABC

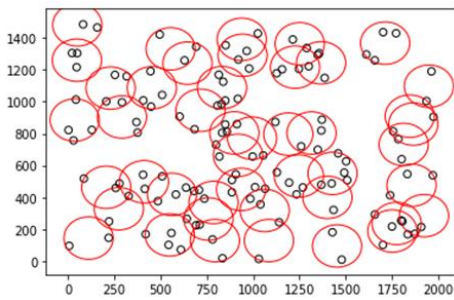


(b) 41 charging stations using LSABC

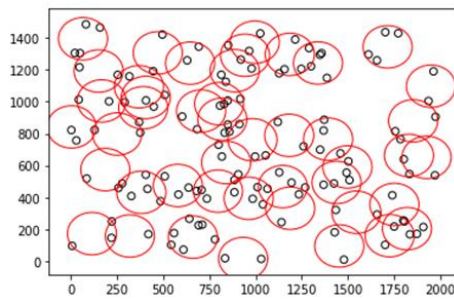


(c) 40 charging stations using NSGA-II

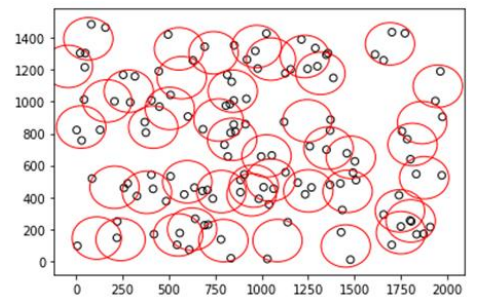
Figure 17. Charging station deployment for different methods under 100 sensor nodes



(a) 45 charging stations using SABC



(b) 45 charging stations using LSABC



(c) 45 charging stations using NSGA-II

Figure 18. Charging station deployment for different methods under 125 sensor nodes

## Chapter 6. Conclusions

This thesis proposes a new deploy strategy by taking the number of charging stations and the distance between sensor node and charging station into account simultaneously. We formulate the proposed strategy into a multi-objective problem and employ a NSGA-II to solve charging station deployment problem. We compare the proposed approach to the simulated annealing-based charging algorithm (SABC) and the layoff simulated annealing-based charging algorithm (LSABC) in terms of the number of charging stations and the overall charging power. The simulation results revealed that under the same number of charging stations, the overall charging power obtained using the proposed approach is 5% and 8% higher than that obtained using SABC and LSABC approaches. Moreover, the number of charging stations obtained using NSGA-II is 6% and 1% less than that obtained using SABC and LSABC approaches under the same number of sensor nodes, respectively.

In future work, we can consider that there will be obstacles in the real environment, and these obstacles will interfere with the charging efficiency. Therefore, these conditions can also be added to the multi-objective problem. Also, NSGA-II takes a long time to deal with multi-objective problems, and we also need to reduce the execution time of the algorithm.

## References

- [1] X. Liu, S. Zhao, A. Liu, N. Xiong, and A. V. Vasilakos, "Knowledge-aware proactive nodes selection approach for energy management in Internet of Things," *Future generation computer systems*, vol. 92, pp. 1142-1156, 2019.
- [2] S. Rajba, P. Raif, T. Rajba, and M. Mahmud, "Wireless sensor networks in application to patients health monitoring," in *2013 IEEE Symposium on Computational Intelligence in Healthcare and e-health (CICARE)*, 2013, pp. 94-98.
- [3] K. Huang, Q. Zhang, C. Zhou, N. Xiong, and Y. Qin, "An efficient intrusion detection approach for visual sensor networks based on traffic pattern learning," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 10, pp. 2704-2713, 2017.
- [4] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393-422, 2002.
- [5] C. Lin, N. Xiong, J. H. Park, and T. h. Kim, "Dynamic power management in new architecture of wireless sensor networks," *International Journal of communication systems*, vol. 22, no. 6, pp. 671-693, 2009.
- [6] Y. Zeng, C. J. Sreenan, N. Xiong, L. T. Yang, and J. H. Park, "Connectivity and coverage maintenance in wireless sensor networks," *The Journal of Supercomputing*, vol. 52, no. 1, pp. 23-46, 2010.
- [7] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, "Wireless Sensor Networks: recent developments and potential synergies," *Journal of SuperComputing*, [www.researchgate.net/publication/25](http://www.researchgate.net/publication/25), vol. 8165429.
- [8] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182-197, 2002.
- [9] J. D. Schaffer, "SOME EXPERIMENTS IN MACHINE LEARNING USING VECTOR EVALUATED GENETIC ALGORITHMS (ARTIFICIAL INTELLIGENCE, OPTIMIZATION, ADAPTATION, PATTERN RECOGNITION)," 1986.
- [10] T. Murata and H. Ishibuchi, "MOGA: multi-objective genetic algorithms," in

*IEEE international conference on evolutionary computation*, 1995, vol. 1, pp. 289-294.

- [11] C. M. Fonseca and P. J. Fleming, "Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization," in *Icga*, 1993, vol. 93, no. July: Citeseer, pp. 416-423.
- [12] S. Zhang, J. Wu, and S. Lu, "Collaborative mobile charging," *IEEE Transactions on Computers*, vol. 64, no. 3, pp. 654-667, 2014.
- [13] T. Liu, B. Wu, H. Wu, and J. Peng, "Low-cost collaborative mobile charging for large-scale wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 8, pp. 2213-2227, 2016.
- [14] Z. Lyu, "Periodic charging planning for a mobile WCE in wireless rechargeable sensor networks based on hybrid PSO and GA algorithm," *Applied Soft Computing*, vol. 75, pp. 388-403, 2019.
- [15] W.-J. Jian, H.-H. Cho, C.-Y. Chen, H.-C. Chao, and T. K. Shih, "Movable-charger-based planning scheme in wireless rechargeable sensor networks," in *2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2015, pp. 144-148.
- [16] W.-C. Chien, H.-H. Cho, C.-Y. Chen, H.-C. Chao, and T. K. Shih, "An efficient charger planning mechanism of WRSN using simulated annealing algorithm," in *2015 IEEE International Conference on Systems, Man, and Cybernetics*, 2015, pp. 2585-2590.
- [17] W.-C. Chien, H.-H. Cho, H.-C. Chao, and T. K. Shih, "Enhanced SA-based charging algorithm for WRSN," in *2016 International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2016, pp. 1012-1017.
- [18] Powercast Corporation, available from: <http://www.powercastco.com/>.
- [19] S. He, J. Chen, F. Jiang, D. K. Yau, G. Xing, and Y. Sun, "Energy provisioning in wireless rechargeable sensor networks," *IEEE transactions on mobile computing*, vol. 12, no. 10, pp. 1931-1942, 2012.
- [20] J.-R. Jiang, Y.-C. Chen, and T.-Y. Lin, "Particle swarm optimization for charger deployment in wireless rechargeable sensor networks," *International Journal of Parallel, Emergent and Distributed Systems*, pp. 1-16, 2018.
- [21] Y. Ma, W. Liang, and W. Xu, "Charging utility maximization in wireless

- rechargeable sensor networks by charging multiple sensors simultaneously," *IEEE/ACM Transactions on Networking*, vol. 26, no. 4, pp. 1591-1604, 2018.
- [22] W.-Y. Lai and T.-R. Hsiang, "Wireless Charging Deployment in Sensor Networks," *Sensors*, vol. 19, no. 1, p. 201, 2019.

