An Adaptive Algorithm for Charger Deployment Optimization in Wireless Rechargeable Sensor Networks

Ji-Hau Liao, Chi-Ming Hong, and Jehn-Ruey Jiang

Department of Computer Science and Information
National Central University
Jhongli City, Taiwan

Abstract. Wireless chargers are used to refill sensors' power supply in a wireless rechargeable sensor network (WRSN) so that the WRSN can operate sustainably. Since wireless chargers are costly, the problem about how to deploy as few as possible chargers to make a WRSN sustainable is important. This paper proposes a greedy algorithm, named adaptive pair based greedy cone selection (APB-GCS), to consider the Friis propagation model for solving the problem under the assumption that chargers are equipped with directional antennas and can be deployed on grid points at a fixed height and that the sensors are deployed on the floor or object surfaces. According to simulation results, the APB-GCS algorithm outperforms others in terms of the number of deployed chargers with moderate computation complexity.

Keywords: Friis propagation model, wireless rechargeable sensor networks, charger deployment, sustainability, greedy algorithms, directional antennas

1 INTRODUCTION

A *wireless rechargeable sensor network (WRSN)* consists of lots of *sensors* and at least one *sink*, where the sensors can sense the environmental information, such as temperature, humidity, and atmospheric pressure, and forward the sensed data to the sink via multi-hop wireless communications. Sensors can also harvest energy from different sources, such as solar power, air flow, and radio frequency (RF) signals, and convert it to direct current (DC) to refill their battery (as shown in Fig. 1) to make the WRSN sustainable. There are some recent studies related to WRSNs in the literature [1-3].

Many energy harvesting technologies [4-9] exit and they can be divided into two classes: (1) intentional energy harvesting and (2) ambient energy harvesting. The former deploys specific devices, called *chargers*, to emit energy to *harvesters* attached to sensor nodes. The latter mounts sensor nodes onto the energy harvesting device, such as a solar panel, to harvest energy from ambient environment. The latter is more difficult to control since it is easily affected by environmental factors. Therefore, this paper focuses on WRSNs using the intentional energy harvesting technology.

adfa, p. 1, 2011.
© Springer-Verlag Berlin Heidelberg 2011

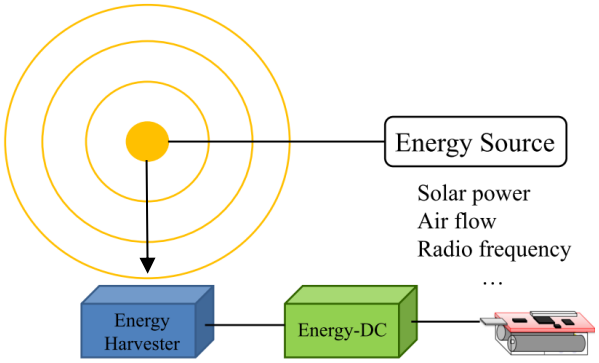

Fig. 1. Energy Harvesting in a WRSN

The chargers are expensive and their deployment is a time- and cost-consuming task. This motivates us to study the *wireless charger deployment optimization (WCDO)* problem [3] about how to deploy as few as possible chargers in a WRSN to cover all sensors (or nodes) to make the WRSN sustainable. We consider a WRSN with wireless chargers equipped with directional antennas, assume chargers are deployed on grid points at a fixed height, model the charging space of a charger as a cone, and propose an adaptive method to solve the WCDO problem. The proposed method is a greedy algorithm called the *adaptive pair based greedy cone selection (APB-GCS)*. We conduct simulation experiments for APB-GCS and compare it with two related algorithms proposed in [3], namely the *node based greedy cone selection (NB-GCS)* and the *pair based greedy cone selection (PB-GCS)*, to show the advantages of APB-GCS.

The rest of this paper is organized as follows. The WCDO problem is described in Section 2. In Section 3, the proposed APB-GCS algorithm is presented. The simulation results and comparisons are described in Section 4. Finally, Section 5 concludes the paper.

2 THE WIRELESS CHARGER DEPLOYMENT OPTIMIZATION (WCDO) PROBLEM

The paper [3] introduced the WCDO problem. Sensor nodes in the WRSN are assumed to be deployed in a cuboid with the length L, width W and height H; they can be located on the ground or object surfaces. On the other hand, the wireless chargers equipped with directional antennas are assumed to be deployed on grid points on the grid with height H, where the side length of the grid is G and each grid point allows the deployment of several wireless chargers. All sensor nodes are homogeneous and all chargers are also homogeneous. Fig. 2 shows a scenario of the WRSN schematically.

The effective charging space of wireless chargers is assumed to be a cone, called a *charger cone*. As shown in Fig. 3, every charger cone is characterized by an apex o, a normal vector $\vec{N}$ whose direction is parallel to the symmetrical axis of the cone, an effective charging distance R, and an effective charging angle threshold θ (i.e., the acute angle between the cone lateral surface and the cone symmetrical axis). When a sensor node is within the charger cone of a charger, we assume the sensor node can be charged effectively by the charger; otherwise, the sensor node cannot be charged effectively. The point X in Fig. 3 is an *extreme point* within the charger cone; it is on the inner side of the charger cone surface and its distance to the cone apex is R.

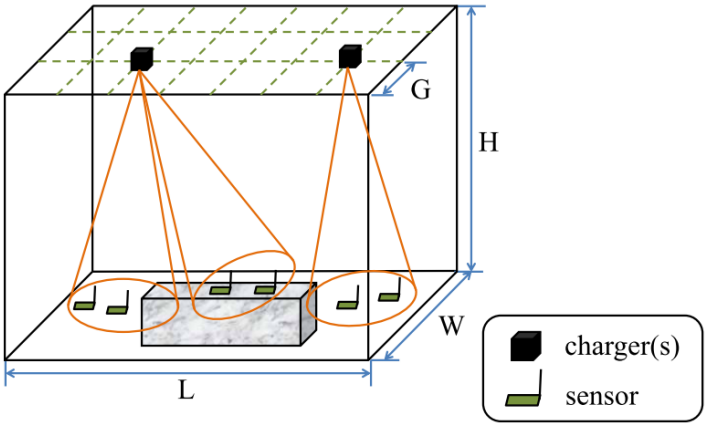

Fig. 2. The scenario of the WRSN

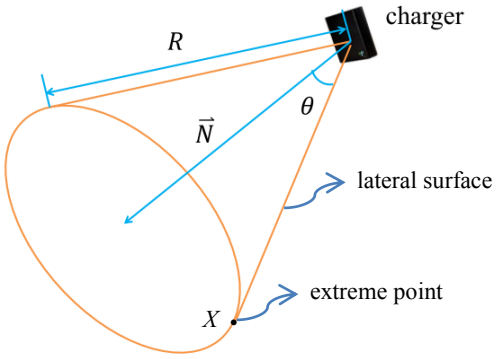

Fig. 3. A charger cone and its parameters

The WCDO problem deals with how to deploy as few as possible chargers in a WRSN to cover all sensors with different energy requirements to make the WRSN sustainable. The WCDO problem can be solved by reducing it to the NP-hard set covering (SC) problem, which is to identify the smallest number of Q's subsets whose

union is U, where U is a given universal set and Q is a collection of subsets of U. It is believed the WCDO problem is also NP-hard. Nevertheless, the NP-hardness of the WCDO problem has not yet proven.

The paper [3] proposed two greedy algorithms, called the node based greedy cone selection (NB-GCS) and the pair based greedy cone selection (PB-GCS), to solve the WCDO problem by characterizing a charger as a cone. The main difference between the two algorithms is on the methods to generate the candidate cones to be selected for covering all the sensors. NB-GCS modifies all the candidate cones' symmetrical axis vector via unit vector addition with the purpose of covering more sensors with a candidate cone. On the other hand, PB-GCS directly generates candidate cones according to three cases for each pair of sensors within the charging space associated with a grid point.

Both NB-GCS and PB-GCS use the same greedy strategy to select cones from the candidate cones, as described below. Every sensor estimates the number of chargers needed to fulfill its energy requirement according to the energy charging rate of the extreme point shown in Fig. 3. When the energy requirement of a sensor is met, it is marked; otherwise, it is unmarked. Both algorithms greedily choose the candidate cone covering the most unmarked sensors to keep the selected (or deployed) chargers as few as possible.

3 THE PROPOSED ALGORITHM

Both NB-GCS and PB-GCS proposed in [3] have acceptable performance; however, they use the energy charging rate of the extreme point to represent the charger's energy charging rate and therefore select an immoderate number of candidate cones. This paper proposes to consider the energy charging rate based on the Friis propagation model [10] to get more accurate energy charging rate estimation for reducing the numbers of selected cones (i.e., chargers).

Friis propagation model [10] is a common model used to describe free space loss in wireless communications. Let P_{tx} and G_{tx} denote the power and the antenna gain of the transmitter, respectively, and let P_{rx} and G_{rx} denote the power and the antenna gain of the receiver, respectively. Note that by the law of conservation of energy, $P_{rx} \leq P_{tx}$. Assume that R is the distance between the transmitter and the receiver and λ is the wavelength of the RF signal. The free space loss can be described by the Friis transmission equation, as shown in Eq. (1).

$$\frac{P_{rx}}{P_{tx}} = G_{tx} G_{rx} \left(\frac{\lambda}{4\pi R}\right)^2 \tag{1}$$

Below we introduce the proposed APB-GCS algorithm for solving the WCDO problem. We first describe the notations used in the algorithm. A WRSN is given within a cuboid space with the length L, width W and height H. The WRSN consists of n sensors, the set of which is denoted by $SN = \{s_1, s_2, \ldots, s_n\}$. Chargers will be deployed on some of p grid points, the set of which is denoted by GP=$\{g_1, g_2, \ldots,$

$g_p\}$, where $p = (\lfloor \frac{L}{G} \rfloor + 1) \times (\lfloor \frac{W}{G} \rfloor + 1)$, and G is the distance between two nearby grid points. Each charger cone is characterized by the cone symmetrical axis vector, the effective charging distance R, and the effective charging angle threshold θ.

Every sensor $s_i \in SN$ knows its own position and workload; therefore it can calculate its energy consumption rate, whose unit is mW (milliWatt), as its *energy charging demand* d_i. The set of energy charging demands of all sensors is denoted as $ECD = \{d_1, d_2, \ldots, d_n\}$. The function $ECR(w, s_i, P_{tx}, G_{tx}, G_{rx}, \lambda)$ is used to estimate the *energy charging rate* between the charger w and the sensor s_i via Friis transmission equation (i.e., Eq. (1)), where the antenna gain of the charger and harvester connected with s_i (i.e., G_{tx} and G_{rx}) are given, the power of charger (i.e., P_{tx}) and its wavelength λ of the RF are known constants. Note that the energy charging rate is also of the unit mW. The distance between w and s_i can be calculated by their location information, so the energy charging rate P_{rx} can be estimated.

Fig. 4 shows the APB-GCS algorithm. In the algorithm, $\overrightarrow{u_x}$ is the vector from the grid point g to sensor s_{a_x}, and Cone($\overrightarrow{u_x}$) represents for the cone which takes of effective charging distance R and effective charging angle threshold θ which takes g as the apex, and takes $\overrightarrow{u_x}$ as the vector of the symmetrical axis. Furthermore, $|Cone(\overrightarrow{u_x})|$ represents the number of sensors in SN covered by Cone($\overrightarrow{u_x}$); $\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y})$ represents the angle between the pair of vectors $\overrightarrow{u_x}$ and $\overrightarrow{u_y}$.

The APB-GCS algorithm has two phases. The first one is the *cone generation phase* to generate candidate cones; the second one is the *cone selection phase* to select some candidate cones (for deploying chargers). In the cone generation phase, the APB-GCS algorithm first unmarks each sensor. Afterwards, for any grid point g, a sphere S is obtained with g as the center, and R as the radius. If S covers only one sensor, one candidate cone is generated. If S covers k (k>1) sensors, then candidate cones are generated based on pairs of sensors covered by S. The algorithm needs to check if every pair of sensors is close enough to be covered by a cone. It thus performs the projection of cones and vectors onto the surface of the unit sphere centered on grid point g, as shown in Fig. 5. The projection of a cone is a circle with radius r; the projection of a vector is a point. For the pair of distinct sensors s_x and s_y, let d_{xy} be the Euclidean distance between the two projection points of $\overrightarrow{gs_x}$ and $\overrightarrow{gs_y}$. By checking the relationship of d_{xy} and r, we have three cases of $\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y})$: (Case 1) ($d_{xy}< 2r$) implies $\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y}) < 2\theta$; (Case 2) ($d_{xy}$=2$r$) implies $\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y}) = 2\theta$; (Case 3) ($d_{xy}>2r$) implies $\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y}) > 2\theta$.

Fig. 6 shows the candidate cone generation of the three cases for the pair of distinct sensors s_x and s_y. (Case 1) If $\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y}) < 2\theta$, only one candidate cone Cone($\overrightarrow{u_x}$) is generated, where $\overrightarrow{u_x} = \frac{\overrightarrow{u_x}+\overrightarrow{u_y}}{\|\overrightarrow{u_x}+\overrightarrow{u_y}\|}$ is adaptively adjusted iteration by iteration for all sensors different form s_x covered by S (please refer to Fig. 7). Note that PB-GCS proposed in [3] generates 4 cones in this case. (Case 2) $\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y}) = 2\theta$, one candidate cone $Cone(\frac{\overrightarrow{u_x}+\overrightarrow{u_y}}{2})$ is generated. (Case 3) If $\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y}) > 2\theta$, two candidate cones $Cone(\overrightarrow{u_x})$ and $Cone(\overrightarrow{u_y})$ are generated.

In the cone selection phase, APB-GCS runs iteration by iteration to greedily select the candidate cone which covers most unmarked sensors. The selected cone corresponds to a practical charger to be deployed. For a selected cone w, APB-GCS updates the charging demand for every sensor s_i covered by w and marks the sensor if its charging demand d_i is fulfilled. It is noted that the charging demand update is done by $d_i = d_i - ECR(w, s_i, P_{tx}, G_{tx}, G_{rx}, \lambda)$ and the charging demand fulfillment is verified by checking if $d_i \leq 0$ or not.

Algorithm: Adaptive Pair Based Greedy Cone Selection (APB-GCS)

Input: $SN, ECD, GP, n, p, R, \theta, P_{tx}, G_{tx}, G_{rx}, \lambda$

Output: $|C^*|$

//Below is the candidate cone generation phase to generate candidate cones
Unmark s_i, $s_i \in SN$; let cone set $C=\emptyset$ and $C^*=\emptyset$
for each grid point g **do**
  S = a sphere centered at g with radius R
  if S *covers* 1 node s_{a_1} **then**
    Generate $\overrightarrow{u_1} = \overrightarrow{gs_{a_1}}$ and add $Cone(\overrightarrow{u_1})$ into C
  else if S *covers* k, $k>1$, sensors $s_{a_1}, s_{a_2}, \ldots, s_{a_k}$ **then**
    Generates $\overrightarrow{u_1} = \frac{\overrightarrow{gs_{a_1}}}{\|\overrightarrow{gs_{a_1}}\|}, \overrightarrow{u_2} = \frac{\overrightarrow{gs_{a_2}}}{\|\overrightarrow{gs_{a_2}}\|}, \ldots, \overrightarrow{u_k} = \frac{\overrightarrow{gs_{a_k}}}{\|\overrightarrow{gs_{a_k}}\|}$
    //$\overrightarrow{u_x}$ is a vector going from g to $\overrightarrow{s_{a_x}}, 1 \leq k \leq n$
   for x = 1, 2,…, k **do**
     for y = 1, 2,…, k **do**
       if $((\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y}) < 2\theta) \wedge (x \neq y))$ **then**
         if $\left(\left(s_{a_x} \in Cone(\overrightarrow{u_x} + \overrightarrow{u_y})\right) \wedge (y \leq k)\right.$
           $\wedge \left.(|Cone(\overrightarrow{u_x} + \overrightarrow{u_y})| > |Cone(\overrightarrow{u_x})|)\right)$ **then**
             $\overrightarrow{u_x} = \overrightarrow{u_x} + \overrightarrow{u_y}$; $\overrightarrow{u_x} = \frac{\overrightarrow{u_x}}{\|\overrightarrow{u_x}\|}$, as shown in Fig. 6 (i)
         else Add $Cone(\overrightarrow{u_x})$ into C
       else if $((\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y}) = 2\theta) \wedge (x \neq y))$ **then**
         Add $Cone(\frac{\overrightarrow{u_x} + \overrightarrow{u_y}}{2})$ into C, as shown in Fig. 6 (ii)
       else if $((\emptyset(\overrightarrow{u_x}, \overrightarrow{u_y}) > 2\theta) \wedge (x \neq y))$ **then**
         Add $Cone(\overrightarrow{u_x})$ and $Cone(\overrightarrow{u_y})$ into C, as shown in Fig. 6 (iii)
//Below is the cone selection phase to select some candidate cones
repeat
  Select cone w covering most unmarked sensors in SN
  Move w from C to C^*
  for every sensor s_i covered by w **do**
    $d_i = d_i - ECR(w, s_i, P_{tx}, G_{tx}, G_{rx}, \lambda)$
  if $d_i \leq 0$ **then** Mark s_i
until all nodes in SN are marked

Fig. 4. The APB-GCS algorithm

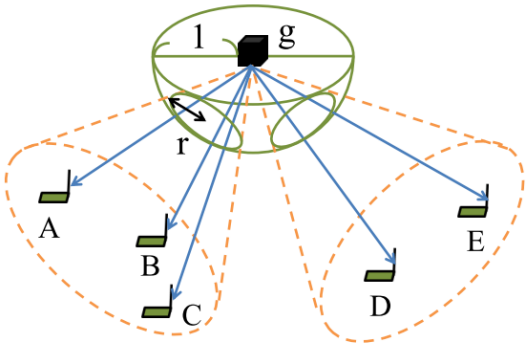

Fig. 5. The projection of cones and vectors onto the surface of the unit sphere centered at the grid point *g*

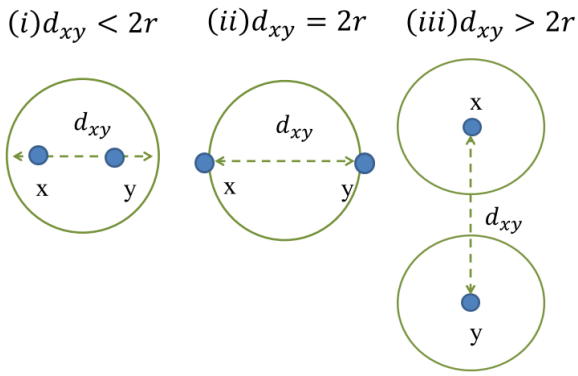

Fig. 6. Three cases for the APB-GCS algorithm to generate candidate cones

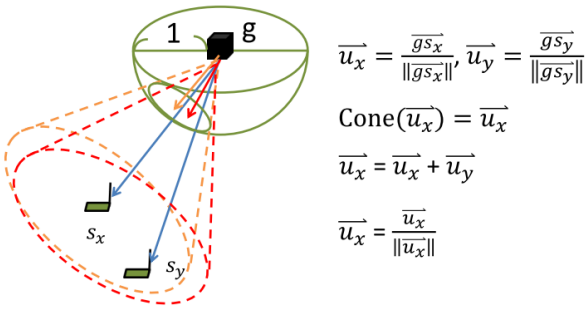

Fig. 7. Illustration of candidate cone adjustment by adding the unit vectors (the orange one is the original cone and the red one is the adjusted cone; the short vectors are unit vectors)

4 SIMULATION RESULTS

This section shows the simulation results of the proposed APB-GCS algorithm in comparison with the NB-GCS algorithm and the PB-GCS algorithm proposed in [3]. The simulator is implemented in C++ language, and the simulation parameters are shown in Table I. The sensors are deployed randomly in a 20 m by 15 m plane; the chargers are deployed on the set of grid points at height of 2.3 m, and the grid side length is 1.8 m. It is noted that 30 experiments were executed per simulation case.

Table 1. SIMULATION SETTINGS

Item	Parameter
Sensor Node Plane	20×15 (m^2)
Number of Sensors	$50, 100, 150, 200, 250$
Effective Charging Distance	3 (m)
Angle Threshold	30°
Height of Grid Points	2.3 (m)
Separation of Grid Points	1.8 (m)
Average Power Consumption of Sensors	0.18, 0.54, 0.9, 1.26 (mW)
Number of Simulations	30 (times/case)
Power of Wireless Chargers	3 (mW)
Antenna Gain of Wireless Chargers	8 (dBi)
Antenna Gain of Harvesters	1 (dBi)
Radio Frequency	915 (MHz)

Fig. 8 shows the comparisons of NB-GCS, PB-GCS, and APB-GCS in terms of the number of the candidate cones generated for the case that the average power consumption of sensors is 0.54 mW. The number of the candidate cones generated by APB-GCS is between NB-GCS and PB-GCS. The computation complexity is proportional to the number of cone generated either in the cone generation phase or in the cone selection phase. Therefore, APB-GCS has lower computation complexity than PB-GCS and has higher computation complexity than NB-GCS.

Fig. 9 shows the comparisons of NB-GCS, PB-GCS, and APB-GCS in terms of the number of the chargers selected. Since one selected cone corresponds to a charger to be deployed, a small number of selected cones is preferred. We can observe that the proposed APB-GCS selects the smallest number of cones, which NB-GCS selects the largest number of cones. When the number of sensors is less than or equals to 150, the difference between the PB-GCS and APB-GCS is not obvious. However, when the number of sensors is greater than or equals to 200, the difference between them is significant. We may well say that APB-GCS outperforms NB-GCS and PB-GCS.

Fig. 10 shows the comparisons of the number of the cones (chargers) selected by APB-GCS for different average sensor charging demands, namely 0.18 mW, 0.54 mW, 0.9 mW and 1.26 mW. The curve of 0.l8 mW and the curve of 0.54 mW are very close. The curve of 0.9 mW goes a little higher than the two curves just mentioned. The curve of 1.26 mW goes much higher than all other curves.

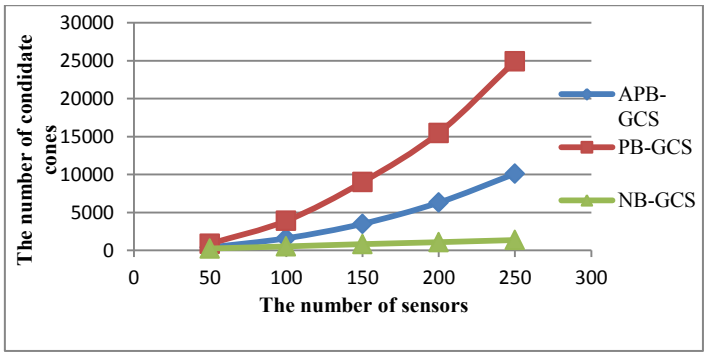

Fig. 4. The comparisons of NB-GCS, PB-GCS, and APB-GCS in terms of the number of generated candidate cones

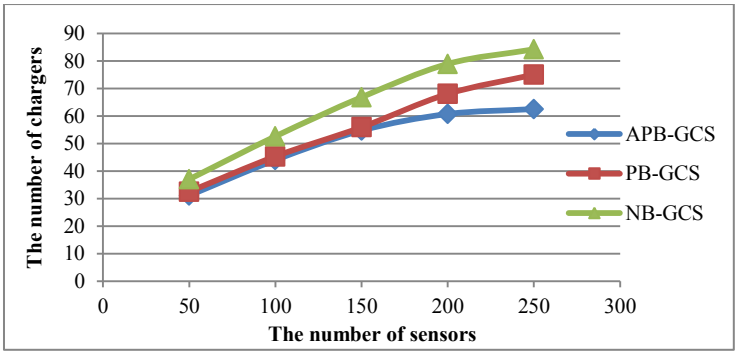

Fig. 5. The comparisons of NB-GCS, PB-GCS, and APB-GCS in terms of the number of selected cones (chargers)

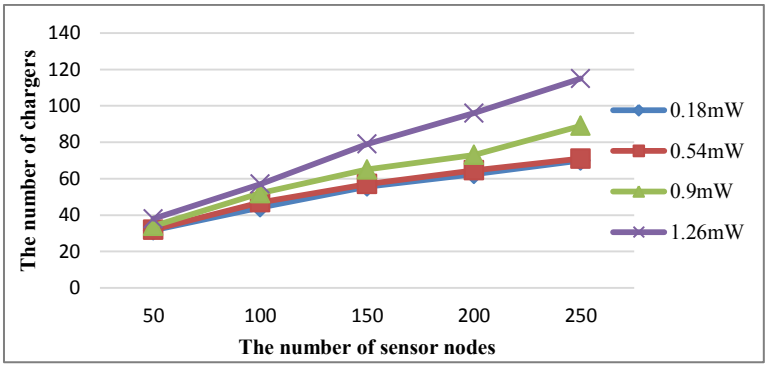

Fig. 6. The number of the cones (chargers) selected by APB-GCS for different average charging demands of sensors

5 CONCLUSION

In this paper, we study the wireless charger deployment optimization (WCDO) problem concerning with how to deploy as few as possible chargers to make a WRSN sustainable. We have proposed a greedy algorithm, called adaptive pair based greedy cone selection (APB-GCS), which uses the Friis transmission equation to estimate energy charging rate between a sensor and a charger to reduce the number of selected cones. We have simulated APB-GCS and compared the simulation results with those of tow related algorithms, namely NB-GCS and PB-GCS in terms of the number of candidate cones generated and the number of candidate cones selected. The number of candidate cones generated is related to computation complexity. We can see that the proposed APB-GCS is better than PB-GCS and is worse than NB-GCS in this aspect. The number of candidate cones selected is related to the charger deployment cost. We can see that APN-GCS outperforms the other two algorithms in this aspect.

In this paper, the chargers are restricted to be deployed on grid points of a grid, which may limit the application of the proposed algorithm. In the future, we plan to design more efficient and flexible algorithms, such as generic algorithms, without the deployment restriction.

REFERENCES

1. L. Fu, P. Cheng, Y. Gu, J. Chen, and T. He, "Minimizing Charging Delay in Wireless Rechargeable Sensor Networks," *Proc. IEEE INFOCOM*, 2013.
2. H. Dai, L. Xu, X. Wu, C. Dong, and G. Chen, "Impact of Mobility on Energy Provisioning in Wireless Rechargeable Sensor Networks," *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, 2013.
3. J.-H. Liao and J.-R. Jiang, "Wireless Charger Deployment Optimization for Wireless Rechargeable Sensor Networks," *Proc. International Conference on Ubi-Media Computing)*, 2014.
4. C. Huang, R. Zhang, and S. Cui, "Optimal Power Allocation for Outage Probability Minimization in Fading Channels with Energy Harvesting Constraints," *IEEE Transactions on Wireless Communications*, Vol. 13, No. 2, pp. 1074 – 1087, 2014.
5. Z. Bo, R. Simon, and H. Aydin, "Included in Your Digital Subscription Harvesting-Aware Energy Management for Time-Critical Wireless Sensor Networks With Joint Voltage and Modulation Scaling," *IEEE Transactions on Industrial Informatics*, Vol. 9, Issue 1, pp. 514-526, 2013.
6. M. Al-Lawati, M. Al-Busaidi, and Z. Nadir, "Included in Your Digital Subscription RF energy harvesting system design for wireless sensors," *Proc. International Multi-Conference on Systems (IMCS)*, pp. 1-4, 2012.
7. Nokia website: http://www.nokia.com/
8. Fu Da Tong website: http://www.rfidpower.com.tw/chinese/index.htm
9. PowerCast website: http://www.powercastco.com/
10. J. A. Shaw, "Radiometry and the Friis Transmission Equation," *American Journal of Physics (AJP)*, Vol. 81, pp. 33-38, 2012.