

# Study of Improved DV-Hop Localization Algorithm in Wireless Sensor Network

<http://dx.doi.org/10.3991/ijoe.v9i6.3162>

Jingxia Wang, Haisong Chen  
Shenzhen Polytechnic, Shenzhen, China

**Abstract**—An improved method is put forward in this paper to settle the larger location error of the classic DV-Hop algorithm. The average hop size is corrected using the hop count based on both the local network connectivity and the global average hop size information of the network. The invalid result is defined and corrected based on the random number in a specified area at the same time. The simulation results show that the improved algorithm gets better localization accuracy than the classic algorithm and other existing improved algorithms. Moreover, it is simple and needs small amount of calculation. So it is a practical algorithm for nodes localization in wireless sensor network.

**Index Terms**—average hop size, DV-Hop, localization algorithm, the invalid results.

## I. INTRODUCTION

Wireless sensor network (WSN) is a new information acquisition and processing technology, which integrates cutting-edge technology such as sensors, micro-electromechanical system and network. In general, WSN intends to provide information on spatio-temporal characteristics of the observed physical world [1]. The localization technology has become a key support in the application of wireless sensor networks. In most typical applications, such as environmental monitoring, target tracking, forest fire prevention, the specific location information of the sensor nodes is essential [2]. All these mentioned above make localization algorithm become one of the most important issues in WSN researches. The existing WSN self-localization algorithm can be divided into two categories: range-based and range-free localization. Range-free localization algorithm mainly uses own connectivity of the WSN. It does not require a direct measurement of the distance or angle between two nodes. But, it can estimate the interval distance using the connectivity information between two nodes, and then estimate the location of the unknown node by a variety of the localization calculation methods. In the existing typical range-free algorithms, the DV-Hop algorithm is easy to implement and has less demanding on the hardware conditions [3].

Average hop size is a key factor which will lead to the accumulation error in the DV-Hop localization algorithm. Classic DV-Hop considers the average hop size estimated by the closest beacon node which reflects only the local connectivity of the network. This paper presents an improved DV-Hop algorithm, which can reduce location error. The remainder of this paper is organized as follows. First, the classic DV-Hop and location error analysis are

discussed. The next section gives a quick survey of previous work about DV-Hop algorithm improvement. In section IV, a new improved DV-Hop algorithm is presented. In section V, simulation results are given and analyzed. Finally, some general conclusions are drawn in Section VI.

## II. DV-HOP ALGORITHM AND ANALYSIS

### A. DV-Hop Algorithm Overview

DV-Hop algorithm was proposed by Niculescu and Nath [4,5]. This location scheme consists three phases as follows.

Stage 1. Getting minimum hop count  $h_{ij}$  between all nodes and the beacon node

Beacon node broadcasts its own location information to the neighbor nodes by flooding. Each node saves the minimum hop count value and jumps forward to next neighbor node with bigger hop count increased by one, and ignores more hop count from the same beacon node. All nodes can obtain minimum hop count value to each beacon node in the network through the typical distance vector exchange protocol.

Parameter  $h_{ij}$  is used to represent hop count value between the node  $i$  and the beacon node  $j$ ,  $\{i|1 \leq i \leq n, i \in Z\}$ ,  $\{j|1 \leq j \leq b, j \in Z\}$  and  $i \neq j$ . Where  $n$  is the number of all nodes and  $b$  is the number of the beacon nodes in the network.

Stage 2. Calculating average per hop size  $AHS_i$  and the distance  $d_{ij}$  between the unknown node and the beacon node

Each beacon node estimates the average size for one hop using (1).

$$AHS_i = \frac{\sum_{j=1}^k \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{j=1}^k h_{ij}} \quad (1)$$

Where  $(x_i, y_i)$  and  $(x_j, y_j)$  are the coordinates of beacon node  $i$  and  $j$  respectively,  $h_{ij}(i \neq j)$  is hop count between beacon node  $i$  and beacon node  $j$ ,  $AHS_i$  is the average hop size for beacon node  $i$ .

Each beacon node broadcasts the calculated  $AHS$  using the data packet with the limited lifetime field. Unknown nodes only receive the first  $AHS$  message and broadcast it to neighbor nodes. This scheme ensures that most of the unknown nodes receive  $AHS$  from the closest beacon

node. For example, if the closest beacon node to unknown node  $k$  is beacon  $i$ , the unknown node  $k$  will calculate the distance to each beacon node using (2).

$$d_{kj} = AHS_i \times h_{kj} \quad (2)$$

Where  $d_{kj}$  is the distance between the unknown node  $k$  and beacon node  $j$ ,  $h_{kj}$  is hop count between the unknown node  $k$  and the beacon node  $j$ ,  $\{k|1 \leq k \leq u, k \in Z\}$ ,  $\{j|1 \leq j \leq n, j \in Z\}$ ,  $u$  is the number of the unknown nodes.

Stage 3. Calculating the unknown node estimation coordination

The unknown nodes calculate the coordinate by Maximum likelihood estimation method using the distance to each beacon estimated in Stage 2 and the known coordinates of the beacon nodes.

### B. Error Analysis

As shown in formula (2), the estimated distance  $d_{kj}$  between unknown node and beacon node is hop count  $h_{kj}$  multiplied by average hop size  $AHS_i$  in the classic DV-Hop. So, location error is usually associated with hop count  $h_{kj}$  and average hop size  $AHS_i$ . When hop count is more than one, the error will increase sharply. In general, the more hop count  $h_{kj}$  is, the greater error of the estimated distance is. At the same time, average hop size  $AHS_i$  is also a key parameter to determine the location error. It supposes that the closest beacon can provide the most accuracy  $AHS_i$  for unknown nodes in classic DV-Hop, but in reality this supposes isn't that case.  $AHS_i$  from the closest beacon reflects only the local network connectivity information, does not reflect the global network environment. In addition, DV-Hop algorithm will get a larger location error because of error accumulation and it is possible that the estimated coordinates of unknown nodes will be out of scope of the network. In fact, these out-of-scope coordinates are invalid results for the algorithm. The improved DV-Hop algorithm gives a new  $AHS$  calculation method for unknown nodes and a correction method for the invalid results.

### III. PREVIOUS RESEARCH

As a typical range-free location algorithm, DV-Hop draws a lot of attentions because it is easy to implement and less dependent on hardware. An improved algorithm is proposed based on the RSSI technology, which is selecting a group of data with the least error as the reduced equation in the least square equation set in order to improve the accuracy of positioning [6]. The average hop size AHS of unknown nodes can be amended with a variety of methods, for example, using the mean value of the whole network average hop size and the average hop size estimated by the closest anchor node [7]; using beacon nodes ratio, the whole network average hop size and the maximum average hop size [8]; and using the error analyzing between estimated distance and real distance of beacon nodes [9]. These improved methods are implemented in the second stage of DV-Hop algorithm and have less amount of calculation and larger amount of network traffic. To enhance the localization accuracy, some other algorithms and intelligent optimization algorithms can be used to accomplish the localization in stage 3, such as 2-D Hyperbolic algorithm [10], Total

Least Squares (TLS) [11], Artificial Bee Colony (ABC) algorithm [12], Particle Swarm Optimization (PSO) algorithm [13,14] and Genetic algorithm (GA) [15,16] etc. These improved algorithms greatly increased the amount of computation. The goal of this paper is to show a new improved DV-Hop algorithm with less amount calculation and communication traffic.

### IV. THE IMPROVED DV-HOP ALGORITHM

#### A. AHS Correction Method

Unknown node  $k$  uses  $AHS_i$  estimated by the closest beacon node  $i$  to calculate the distance by (2) in classic DV-Hop. But  $AHS_i$  reflects only the local connectivity information of the network, does not reflect the global network connectivity. In order to solve this problem, the network average hop size  $N\_AHS$  and max average hop size  $M\_AHS$  are defined using (3) and (4) respectively.

$$N\_AHS = \frac{\sum_i^k AHS_i}{b} \quad (3)$$

$$M\_AHS = \max(AHS_1, AHS_2, \dots, AHS_b) \quad (4)$$

$N\_AHS$  is mean value of  $AHS_i$  of all beacon nodes in the network and reflects average hop size information of entire network to a certain extent [7,8].  $M\_AHS$  is the max of  $AHS_i$  in the network. When the ratio of beacon nodes is larger,  $M\_AHS$  can be used to correct  $AHS$  [8].

As shown in (5), unknown nodes get a new average hop size  $AHS\_U_k$  using  $N\_AHS$  and  $AHS_i$ . Thus,  $AHS\_U_k$  contains both the network-wide average hop size estimation information, also has the local connectivity information from its closest beacon node [6].

$$AHS\_U_k = (AHS_i + N\_AHS) / 2 \quad (5)$$

And then, the correction factor  $\Delta AHS_k$  based on  $M\_AHS$  is calculated using (6).

$$\Delta AHS_k = M\_AHS - AHS\_U_k \quad (6)$$

Unknown node calculates the max of hop count between unknown node and each beacon node using (7).

$$Max\_h_k = \max(h_{m1}, h_{m2}, \dots, h_{mk}) \quad (7)$$

The hop count correction factor  $m_{kj}$  based on  $Max\_h_k$  is calculated using (8).

$$m_{kj} = h_{kj} / Max\_h_k \quad (8)$$

The average hop size  $AHS_{kj}$  of unknown node  $k$  to beacon node  $j$  is shown in

$$AHS_{kj} = AHS\_U_k + \Delta AHS_k \times m_{kj} \times e \quad (9)$$

Where  $e$  is the environment factor,  $\{e \mid 0 \leq e \leq 1, e \in \mathbb{R}\}$ .

Average hop size correction value is directly proportional to hop count, the greater the hop count is, the bigger correction value. When  $e$  equals to 0,  $AHS_{kj}$  equals to  $AHS_{U_k}$ ; when  $e$  equals to 1 and  $h_{kj}$  equals to  $Max\_h_k$ ,  $AHS_{kj}$  equals to  $M\_AHS$ .

### B. Invalid Results Correction Method

There is a specified area for a wireless sensor network generally, and some estimated coordinates of unknown node will be possibly out of the network area because of the accumulation error [9]. In fact, these out-of-scope location results are invalid for the location algorithm. As shown in Fig. 1, it is assumed that the specified network area is  $(x_{min}, y_{min})$  to  $(x_{max}, y_{max})$ , and the estimated coordinates A, B, C and D are  $(x_1^{est}, y_1^{est})$ ,  $(x_2^{est}, y_2^{est})$ ,  $(x_3^{est}, y_3^{est})$  and  $(x_4^{est}, y_4^{est})$  respectively. The estimated coordinates A is invalid result because of  $x_1^{est}$  is less than  $x_{min}$ , and B is also valid because of  $x_2^{est}$  is less than  $x_{min}$  and  $y_2^{est}$  is bigger than  $y_{max}$ . For the same reason, both C and D are invalid results, too.

At first, two parameters  $\Delta x$  and  $\Delta y$  are defined to specify the correction area for x-axis and y-axis respectively. X-axis correction area parameter  $\Delta x$  is calculated using (10) and y-axis correction area parameter  $\Delta y$  is calculated using (11).

$$\Delta x = (x_{max} - x_{min}) / 20 \quad (10)$$

$$\Delta y = (y_{max} - y_{min}) / 20 \quad (11)$$

When the estimated x-axis coordinate component  $x_i^{est}$  is less than  $x_{min}$ , a random number is generated range of  $[x_{min}, x_{min} + \Delta x]$  as the corrected x-axis coordinate component  $x_i^{corr}$ . When the estimated x-axis coordinate component  $x_i^{est}$  is bigger than  $x_{max}$ , a random number is generated range of  $[x_{max} - \Delta x, x_{max}]$  as the corrected x-axis coordinate component  $x_i^{corr}$ . The correction method is similar as above for the y-axis coordinate component of unknown node. Fig. 1 shows the possible correction coordinate a, b, c and d for the valid result A, B, C and D respectively.

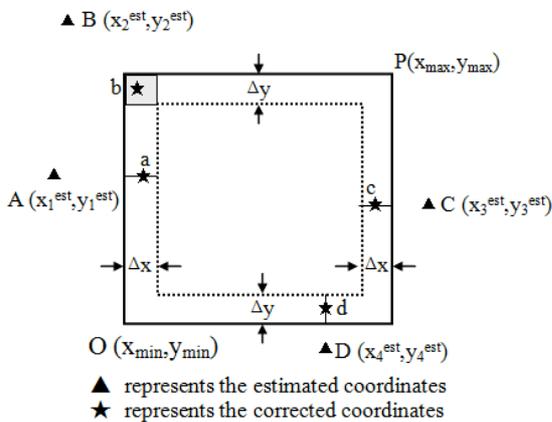


Figure 1. Invalid results correction method

### C. Invalid Results Correction Method

Similar to classic DV-Hop, the improved algorithm also includes three stages.

In the first stage, beacon nodes broadcast data packet  $\{ID_i, x_i, y_i, h_i\}$  by flooding,  $ID_i$  is its own identification number,  $(x_i, y_i)$  is its own coordinate and  $h_i$  is hop counter, the initiation value of  $h_i$  equals to 0. The nodes store the smallest value of hop counter  $h_{ij}$  and jump forward to next neighbor node with hop counter increased by one.

In the second stage, beacon nodes calculate the value of average hop size  $AHS_i$  using (1) and broadcast data packet  $\{ID_i, AHS_i\}$  by flooding. All beacon nodes receive  $AHS_i$  from the other beacon nodes and calculate the network average hop size  $N\_AHS$  using (3) and the max network average hop size  $M\_AHS$  using (4), and then broadcast data packet  $\{ID_i, AHS_i, N\_AHS, M\_AHS\}$  with the limited lifetime field. Unknown nodes receive it from the nearest beacon node and calculate the correction average hop size  $AHS_{kj}$  by (5), (6), (7) and (8).

In the third stage, unknown nodes take the distance from the product of hop count and corrected average size to calculate its own estimated coordinate  $(x_i^{est}, y_i^{est})$  by Maximum likelihood estimation method. If the estimated coordinate is invalid, unknown node corrects it using (10).

## V. SIMULATION AND ANALYSIS

### A. Average Location Error Definition

This section provides a detailed analysis and simulation in the performance of the improved algorithm. Average location error (ALE) is used to measure the accuracy of the algorithm. The definition of ALE is shown in (12).

$$ALE = \frac{\sum_{i=1}^u \sqrt{(x_i - x_i^{est})^2 + (y_i - y_i^{est})^2}}{uR} \quad (12)$$

Where  $(x_i^{est}, y_i^{est})$  is the estimated coordinate of the unknown node  $i$ ,  $(x_i, y_i)$  is the real coordinate of the unknown node  $i$ ,  $u$  is the number of unknown node in the network, and  $R$  is the communication radius.

Formula (11) shows that ALE is defined as the ratio of the Euclidean distance between the estimated coordinate and the real coordinate to communication radius  $R$ . ALE can reflect the location accuracy and the stability of the localization algorithm. When the value of  $R$  is constant, the less ALE is, the more accurate the localization performance.

In this paper, Matlab7.6 is used to complete the algorithm simulations, and the nodes are randomly deployed in a two dimensional area by  $100 \times 100$  ( $x_{min}=0, x_{max}=100, y_{min}=0, y_{max}=100$ ). The value of  $R$  is 30. 200 simulations are carried out under every same network environment. The given data are average value of 200 test results.

### B. Simulation and Analysis of Environment Factor

Tests were carried out using Matlab7.6 in order to determine the influence on location error of the environment factor. Fig. 2 shows ALE comparison with different value of environment factor  $e$  when the number

of beacon nodes is constant ( $b=5$ ) and the number of nodes (including anchor nodes and unknown nodes) ranges from 75 to 200. It is found that ALE gets a lower value when  $e$  is 0.6.

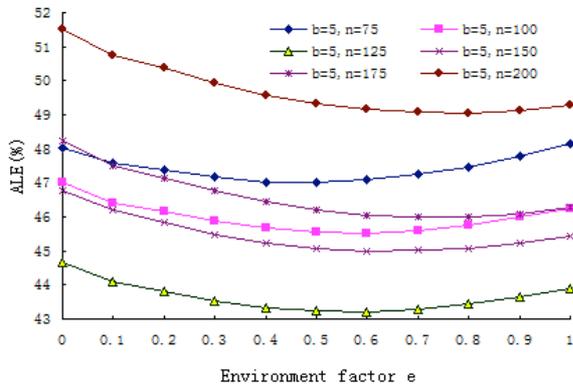


Figure 2. ALE comparison with different value of environment factor

Similarly, when the number of beacon nodes is 10 and  $e$  is 0.7 or 0.8, ALE gets a lower value too. The corresponding relationship between the number of beacon nodes and  $e$  is shown in Table I.

TABLE I.  
THE CORRESPONDING RELATIONSHIP BETWEEN THE NUMBER OF THE BEACON NODES AND ENVIRONMENT FACTOR

No.	The number of beacon nodes $b$	Environment factor $e$
1	5	0.6
2	10	0.7 or 0.8
3	15	0.9
4	20	1
5	25	1
6	30	1

As shown in Table I, environment factor  $e$  is a function of the number of beacon nodes by (12).

$$e = \begin{cases} 0.03b + 0.45 & b \leq 15 \\ 1 & b > 15 \end{cases} \quad (12)$$

### C. Simulation and Analysis of the Algorithm Performance

In order to test the performance of algorithms, simulation experiment was made for the classic DV-Hop, existing improved algorithm [8] (named as DV-Hop1) and the improved algorithm (named as DV-Hop2) in this paper.

Fig. 3 shows the location error comparison of the classic DV-Hop algorithm, DV-Hop1 and DV-Hop2 with 100 nodes generated randomly in the specified region when the number of nodes is constant ( $n=100$ ) and the number of beacon nodes ranges from 5 to 35. When the number of nodes is constant, location error ALE of three algorithms will reduce with the increase of the number of beacon nodes. Location error ALE of DV-Hop2 decreases by 8.7% compared with DV-Hop algorithm and decreases by 2.3% compared with DV-Hop1. Moreover, the less the number of beacon nodes is, the more ALE reduces. Therefore, the performance of DV-Hop2 is better than

DV-Hop and DV-Hop1, and more suitable for the network with the less beacon numbers.

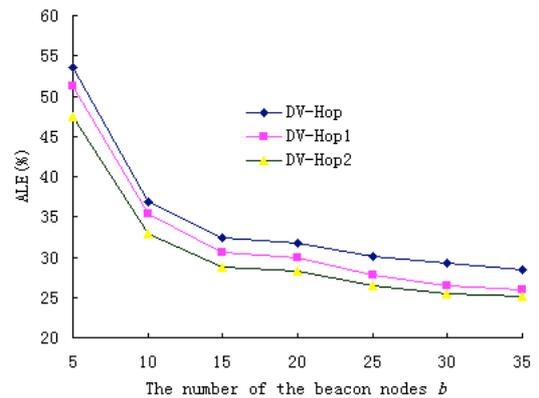


Figure 3. ALE comparison with different number of beacon nodes

Fig. 4 shows location error comparison of the DV-Hop, DV-Hop1 and DV-Hop2 when the number of beacon nodes is constant ( $b=15$ ) and the number of nodes ranges from 75 to 225. When the number of beacon nodes is constant, ALE of DV-Hop2 is lower than the other two algorithms obviously. It is noteworthy that ALE of three algorithms does not always reduce with the increase of the number of nodes. This is because the network connectivity will be improved and the location error will decrease with the increase of the number of nodes when the network area is specified. However, if there is excessive number of nodes, the network throughput and the feasibility of the algorithm will be a big impact, and the network connectivity won't be improved and the network congestion will be possible. Therefore, it isn't true that the more the number of nodes, the higher location accuracy. Fig. 4 also shows that the stability of DV-Hop2 is much better than the other two algorithms.

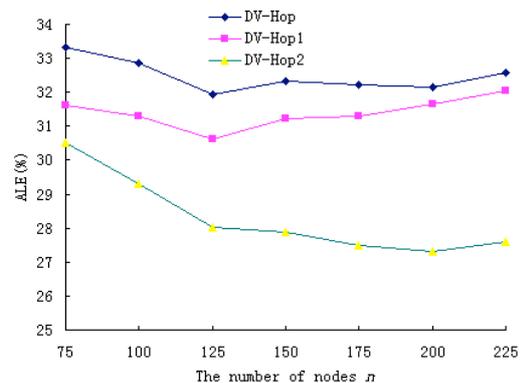


Figure 4. ALE comparison with different number of nodes

## VI. CONCLUSION

Based on the characteristics of wireless sensor network, this paper gives error analysis on the average hop size calculation method of the classic DV-Hop and presents an improved algorithm. The average hop size estimated by the nearest beacon node can reflect the local connectivity information of the network and the average hop size and the max hop size of the whole network can reflect the network-wide information. The correction method of average hop size is proposed using these three parameters based on hop count and environment factor, and a

correction algorithm for the invalid results is given based on the random number in a specified area at the same time. It is tested that environment factor is a function of the number of beacon nodes. The improved algorithm needs a little more communication traffic and calculation than the classic DV-Hop, no further communication resources and accessional localization devices. Simulations were carried out and the experimental results demonstrate that the proposed method is feasible and immensely improves the accuracy of localization.

## REFERENCES

- [1] Jian Li, Jianmin Zhang, and Liu Xiande, "A weighted DV-Hop localization scheme for wireless sensor networks," Eighth IEEE International Conference on Embedded Computing; IEEE International Conference on Scalable Computing and Communic, 2009, pp. 269-272.
- [2] Wu Yanhai, Zhang Jia, Wu Nan, and Geng Fang, "Improved localization algorithm in wireless sensor networks," 2011 Fourth International Conference on Intelligent Computation Technology and Automation, 2011, pp. 576-579. <http://dx.doi.org/10.1109/ICICTA.2011.429>
- [3] Tingting YI, Zhiyi FANG and Ruixue LI, "RMADV-Hop: An improved DV-Hop localization algorithm," 2010 Seventh International Conference on Information Technology, 2010, pp. 939-943.
- [4] Niculescu D and Nath B, "Ad-hoc positioning system (APS)," Proc. of the IEEE GLOGECOM, San Antonio, 2001, pp. 2926-2931.
- [5] Niculescu D and Nath B, "DV Based Positioning in Ad hoc Networks," Journal of Telecommunication Systems, 2003, 22(1-4), pp. 267-280. <http://dx.doi.org/10.1023/A:1023403323460>
- [6] JIN Chun, YE Cheng, and HAN Zhi-bin, etc. "DV-Hop localization algorithm improvements in wireless sensor networks," Computer Engineering and Design, Vol.34 No.2, pp. 401-404, Feb, 2013.
- [7] Zhou Yang, Qin Jiahang, and Li Wanlei, "Improved Algorithm for DV-Hop Position in Wireless Sensor Network," Microcontrollers & Embedded Systems, pp. 1-4, January 2013.
- [8] Jingxia Wang and Jiandong Tang, "Correction of average hop distance DV-Hop Localization Algorithm in Wireless Sensor Network," Digital Technology and Application, pp. 110-113, April 2012.
- [9] LIN Jin-zhao, LIU Haibo, and LI Guo-jun, "Study for improved DV-Hop localization algorithm in WSN," Application Research of Computers, Vol 26 No.4, pp. 1272-1275, April, 2009.
- [10] Chen Hong-yang, Sezaki K, and Deng Ping, et al, "An improved DV-Hop Localization algorithm for wireless sensor networks", Proc of the 3rd IEEE Conference on Industrial Electronics and Applications (ICIEA). Singapore: IEEE, 2008, pp. 1557-1561.
- [11] ZHANG Jia, WUYan-ha,i SHIFeng, and GENG Fang, "Localization algorithm based on DV-HOP for wireless sensor networks [J]. Journal of Computer Applications," Vo.1 30 No. 2. 2010, pp. 323-326.
- [12] LI Mu-dong, XIONG Wei, and Guo Long. "Improvement of DV-Hop Localization Based on Artificial Bee Colony Algorithm," Computer Science, Vol.40 No.1, pp. 33-36, Jan 2013.
- [13] CHEN Xing-zhou, LIAO Ming-hong, and LIN Jian-hua. "Improvement of node localization in wireless sensor network based on particle swam optimization," Journal of Computer Applications, Vol.30 No.7, pp. 1736-1738, July 2010. <http://dx.doi.org/10.3724/SP.J.1087.2010.01736>
- [14] Zhao Ji, Fu Yi, and Mei Juan, "An improved DV-Hop localization algorithm based on particle swarm optimization," Computer Applications and Software, Vol.29 No.12, pp. 69-76, Dec 2012.
- [15] DENG Li, "Wireless sensor network based on genetic algorithm of localization algorithms," Computer Simulation, Vol.28 No.9, pp. 161-164, 2011.
- [16] ZHANG Li-hong, "Wireless sensor network localization based on improved DV-Hop algorithms," Microelectronics & Computer, Vol.29 No.9, pp. 171-178, 2012.
- [17] Fang Wangsheng and Zeng Jing. "On range-free node localization algorithm based on quantum genetic algorithm," Computer Applications and Software, Vol.30 No.2, pp. 180-183, 2013.

## AUTHORS

**Jingxia Wang** is with the Department of electronic and communication engineering, Shenzhen Polytechnic, Shenzhen 518055, PR China (e-mail: [wjx2000@szpt.edu.cn](mailto:wjx2000@szpt.edu.cn)).

**Haisong Chen** is with the Industrial Training Centre, Shenzhen Polytechnic, Shenzhen 518055, PR China (e-mail: [hschen@szpt.edu.cn](mailto:hschen@szpt.edu.cn)).

Submitted 04 September 2013. Published as re-submitted by the authors 22 October 2013.