Research on an Improved DV-HOP Localization Algorithm Based on PSODE in WSN

Dan Zhang[1,2], Zhiyi Fang[1], Lin Chen[1], Hongyu Sun[1,3], and Yue Wang[1]
[1] College of Computer Science and Technology Jilin University, Changchun, 130012, P.R. China
[2] Department of Computer Science and Information Technology, Daqing Normal University, Daqing, 163712, P.R.China
3 Department of Computer Science and Electrical Engineering, University of Maryland Baltimore County, Baltimore, MD, 21250, USA
Email: qhzd2005@163.com; fangzy@jlu.edu.cn; cldreaming@163.com; shy12@mails.jlu.edu.cn; qingtaow@qq.com

***Abstract*—** DV-Hop localization algorithm is a range free localization algorithm in WSN (Wireless Sensor Network). In recent years, the research on range-free localization algorithm is one of the hot issue in WSN. DV-Hop algorithm is a simple, convenient operation, high efficiency and low energy consumption. But the the localization accuracy is low. In this paper, we study the traditional DV-Hop localization algorithm and propose an improved DV-Hop localization algorithm. In order to make the result of the improved DV-Hop algorithm more accurate, we propose an improved method. After the unknown nodes calculated their own estimated coordinate values by using the improved DV-hop algorithm, the PSODE algorithm is applied to the values to improve the localization accuracy. Simulation results show that this method can improve the localization accuracy of nodes obviously in WSN.

***Index Terms*—**DV-Hop, location algorithm, PSO, DE, PSODE, WSN

I. Introduction

Nowadays, more and more people come into contact with wireless sensor network. In WSN, some nodes scatter in the area, they should have the ability of knowing their own position and being able to find the exact location of the node. In WSN, the observing point or technical staff can decide how to respond to the situation only when they get the exact location information and gather the data from the node. If there is no location information, the obtained data would have no meaning [1]. GPS(Global Positioning System) has developed very well. It has the advantages of locating accurate and efficient. But GPS localization requires facilities to support it. So it has high costs. GPS system applies only in unobstructed and open environment [2]. Wireless sensor network requires very little for the environment. The positioning algorithm not requires so many facilities so that it will not cost too much. Therefore, the study of localization algorithm in WSN has great significance. So for the localization algorithm, the previous researchers and scholars have made a great contribution. They have given a variety of localization algorithms. When using or improving the algorithm, people also need to pay attention to the limitation of sensor node, such as the limitation of node energy, the random distribution of node and the fickle node environment. However, each localization algorithm has its own advantages and disadvantages. We should not only try to improve the localization accuracy, to reduce the time complexity and space complexity, but also try to extend the service life of the sensor and solve the energy saving problem and so on at the same time. Therefore, how to maximize use of advantages and reduce the disadvantages is the ultimate goal of studying the localization algorithms [3].

In wireless sensor network, the localization algorithm commonly divided into two categories: range-based algorithm [4] and range-free algorithm [5], [6]. This classification is based on if the localization of the unknown nodes need hardware of sensor node and whether need to estimate and measure the distance of neighbor nodes. Range-based algorithm has higher localization accuracy compared with range-free algorithm, and it usually includes RSSI algorithm [7], [8] (based on the intensity of the arriving signal), AOA algorithm (Angle of Arrival), TDOA algorithm (Time Difference of Arrival), TOA algorithm [9], [10] (Time of Arrival) and so on. Although the range-based algorithm can get more accurate positioning results, and can use the hardware equipment to remove nodes which have a larger error, it has increased the capital requirements, these algorithms is not suitable for application in low-cost projects. Range-free localization such as DV-Hop algorithm [11], Approximate Point-in-Triangulation test algorithm (APIT) [12], Amorphous algorithm and Centroid algorithm [13].

In this paper, we will focus on the DV-Hop algorithm and an improved DV-Hop algorithm is proposed.

The remainder of this paper is structured as follows. In Section 2 we introduce the related work about DV-Hop algorithm. In Section 3 we describe the an new improved DV-Hop algorithm added PSODE. In Section 4 we evaluate our proposed algorithm and compared simulation results. In Section 5 we present the conclusions of this work and the next step work in future.

Manuscript received June 15, 2015; revised September 8, 2015.
This work was supported by the Youth Foundation of Daqing Normal University under Grant No. 12ZR22.
Corresponding author email: qhzd2005@163.com.
doi:10.12720/jcm.10.9.728-733

II. Related Work

A. DV-Hop Algorithm

DV-Hop algorithm is one of the range-free localization algorithm[14]-[16]. The algorithm flow is as follows:

Step1: Calculate the minimum hop count between the unknown nodes and each anchor nodes.

The existing nodes in the network broadcast all their information to other nodes by flooding algorithm. All of the transmitted information is hop account (initialized to zero) and coordinate information. The receiving nodes record the minimum hop count to every anchor nodes, and add 1 to the hop count. The new hop information will be forwarded to the neighbor nodes.

Step2: Calculate average hop distance between the unknown nodes and anchor nodes.

According to the hop count recorded in the first stage, every anchor node compute the average hop distance by using the formula (1).

$$\text{HopSize}_i = \frac{\sum_{i \neq j} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{i \neq j} hi_j} \quad (1)$$

$HopSize_i$ refers to the average hop distance of anchor node i ; (x_i, y_i) and (x_j, y_j) refer to the coordinates of the anchor nodes i and j ; h_{ij} refers to the hop count between i and j . Anchor nodes broadcast calculated average hop distance group with a lifetime field to the network by using flooding algorithm. The unknown node only records average hop distance that comes from the nearest anchor node, and forwards it to the neighbor nodes. After obtaining average hop distance, unknown nodes compute the distance to each anchor node distance according to the hop count obtained in the first stage.

Step3: Calculate their own position by using the trilateration method or maximum likelihood estimation method.

The unknown nodes use the trilateration method or maximum likelihood estimation method to calculate their own position, according the distance of each anchor node recorded in step 2.

B. An Improved DV-Hop Algorithm

Distance vector algorithm uses average hop distance to calculate the actual distance. It's drawback is that it use hop distance to substitute straight distance, there are some errors. In DV-Hop algorithm, the unknown node receives the average hop distance that comes from the nearest anchor nodes, but the average hop distance that comes from the nearest anchor nodes may not have the minimum hop counts. Thus through our experimental group we propose an improved DV-Hop algorithm. Experimental results show that this method can improve the localization accuracy of unknown nodes in WSN. It can be described as follows:

- Calculate the minimum hop count between the unknown nodes and each anchor nodes.
- Calculate average hop distance between the unknown nodes and anchor nodes. When the unknown node uses the anchor nodes around it to calculate the average hop distance, it selects the anchor nodes in one hop distance around it and calculate their average hop distance value as $PJhopSize$.
- Multiplying the $PJhopSize$ and minimum hop count, the result is the estimated distance between anchor nodes. Then it uses the differences between the actual distance and estimated distance divided by the actual distance, naming the result as $Subpercent$.
- Use $Subpercent$ to correct $PJhopSize$ by using the Formula $Dhop = PJhopSize \times (1 + Subpercent)$ and we can get a new value as $Dhop$, take the average of $PJhopSize$ and $Dhop$, name the average as $PDhop$, and it is the hop distance that the unknown nodes saved.
- Calculate their own position by using the trilateration method or maximum likelihood estimation method.

III. An New Improved DV-Hop Algorithm Added PSODE

The new improved DV-Hop algorithm is based on PSO (particle swarm optimization) and DE (differential evolution) algorithm.

A. Particle Swarm Optimization Algorithm (PSO)

Assuming that the target space is multidimensional, set it as D dimension, there is a group that contains m particles. The location of the particle numbered i is represented by vector notation as a set of vectors like x_{i1} , x_{i2} , x_{i3} , x_{i1} , ..., x_{iD} . The speed is represented as V_{i1} , V_{i2} , ..., V_{iD} . In the current location that the particle numbered i is found, the best position is represented as $P_i = (p_{i1}, p_{i2}, ..., p_{gD})$. Finally, the best location that the particle individual finds in all groups is represented as $P_g = (p_{g1}, p_{g2}, ..., p_{gD})$. Particle update formula is as follows:

$$V_{id}(t+1) = V_{id}(t) + c_1 r_1 (P_{id} - X_{id}(t)) + c_2 r_2 (P_{gd} - X_{id}(t)) \quad (2)$$

$$X_{id}(t+1) = X_{id}(t) + V_{id}(t+1) \quad (3)$$

where $i = 1, 2, ..., m$, $d = 12, ..., D$, c_1 and c_2 represent a non-negative acceleration constant, r_1 and r_2 are random numbers and uniformly distributed in [0,1], $X_{id}(t)$ is the current location of the particle numbered i , P_{id} is the best position that the particle numbered i currently searched, P_{gd} is the optimal position that can be found in all the particle swarm, V_{id} is the current speed of the particle numbered i , $V_{id} \in [V_{\min}, V_{\max}]$, $V_{\min}$ and $V_{\max}$ represents the upper limit and lower limit and of the particle velocity [17], [18].

B. Differential Evolution Algorithm (DE)

To every $\min F(x)$ minimization problem, DE algorithm is based on the populations X_i^t, these populations exist N candidate solutions, $i = 1, 2, ..., N$, i is the population, t is the current generation. In the mutation operation, each random vector are obtained by the equation (4), r_1, r_2, r_3 is random numbers that ranges from 1 to N, F represents a weighting factor that ranges from 0 to 2.

$$v_i^t = x_{r1}^t + F(x_{r2}^t - x_{r3}^t) \tag{4}$$

In the hybrid operation, we use the two vectors to obtain the new population $x_i^{t'} = [x_{i1}', x_{i2}', ..., x_{iD}']$, including the random vector $v_i^t = [v_{i1}, v_{i2}, ..., v_{iD}]$ and target vector $x_i^t = [x_{i1}, x_{i2}, ..., x_{iD}]$.

$$x_{ij}' = \begin{cases} v_{ij}, if\ randb(j) \le C, R\ or\ j = randr(i) \\ x_{ij}, if\ randb(j) \ge CR\ or\ j \ne randr(i) \end{cases} \tag{5}$$

where $randb(j) \ge [0.1]$, $j \in [1, D]$, $randb(j)$ represents the $j-th$ value in the generated random numbers. $CR \in [0.1]$, CR represents the probability of mutation. $randr(i) \in [1, 2, ..., D]$, $randr(i)$ gets the index randomly. It's function is to ensure $x_i^{t'}$ can get not less than one parameters from v_i^t.

Select operation uses the greedy strategy:

$$x_i^{t+1} = \begin{cases} x_i^{t'}, if\ \varphi(x_i^{t'}) \prec \varphi(x_i^t) \\ x_i^t,\ \text{otherwise} \end{cases} \tag{6}$$

where $\varphi(x)$ represents the fitness function.

C. Particle Swarm Optimization Based on Differential Evolution (PSODE)

PSO algorithm and DE algorithm both have inspired ideas. Intelligent algorithms are generally associated with populations and they have no exception. The obvious difference between the two is the way to generate new individual. In the PSO algorithm, the way to get the best position of the current particle is to find it in the overall population. After finding the best position in the overall population, the particle compares it with the best position it gets now. In DE algorithm, we must first obtain an intermediate population to get the t next generation. If there is individual particle going into local optimum in the PSO population, the particles will take the information that the best particle in DE population finds into consideration to get the position for the next time. The essence of the PSODE algorithm is as follows:

Every time in the process of evolution, by using the exchange mechanism that the algorithm configures to exchange information, we can avoid the value of each population going into local optimum. The particle that generate by using differential operation can guide the evolution mode of the particle. If there is an individual in PSO or DE population that stopped in case of reaching the iteration upper limit, the algorithm executes as formula (7).

IF $(x_i^t = x_i^{t+1} = x_i^{t+2} = ... = x_i^{t+p}) \& (F(x_i^{t+P}) \ne F)$

$$x_i^{t+p+1} = x_{\min} + rand(0,1) \times (x_{\max} - x_{\min}) \tag{7}$$

End

The Minimum result that obtained by fitness function is represented by F. The iteration upper limit that the Algorithm stopped is represented as p, $(x_{\max} - x_{\min})$ is the allowed search border [16].

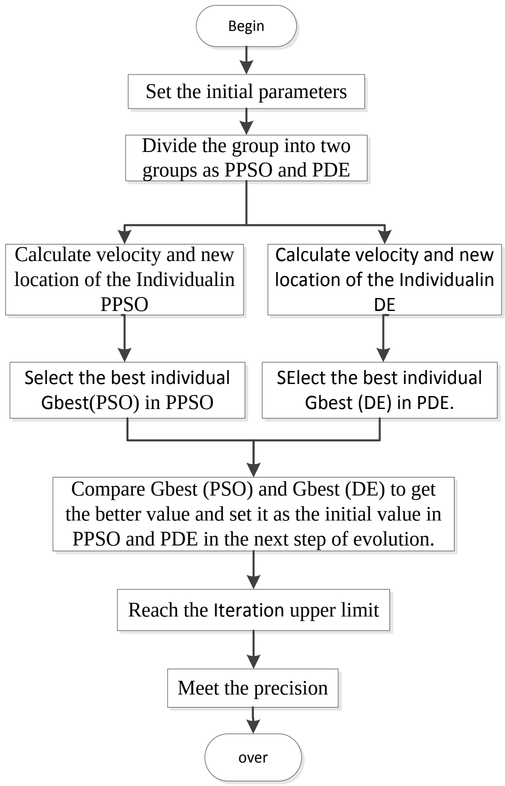


Fig. 1. The flowchart of algorithm

The flowchart of the algorithm is shown in Fig. 1. The main steps is as follows:

Step 1: Set the initial parameters: population size is M, the iteration upper limit value is $t_{\max}$, the control factor is λ, F represents the scaling factor, the solution precision is ε, the upper limit of Inertia weight value is w_{start}, the lower limit of Inertia weight value is w_{end}, the acceleration factor is set as c_1, c_2, mutation probability is set as CR.

Step 2: Divide the population into two populations as PPSO and PDE, set the initializing position in different areas.

Step 3: Calculate velocity and new location of the particle in PPSO based on formula *Step 2*, *Step 3* .

Step 4: Calculate individual velocity and the new location in PDE according to the formula *Step 4*, *Step 5*, *Step 6*.

Step 5: Select the best individual Gbest (PSO) in PPSO, select the best individual Gbest (DE) in PDE.

Step 6: Compare Gbest (PSO) and Gbest (DE) to get the better value and set it as the initial value in PPSO and PDE in the next step of evolution.

Step 7: If there is an individual that has stopped, the algorithm execute as the formula *Step 7*.

Step 8: Record the best value in all groups currently, when the algorithm reaches the iteration upper limit or meets the precision values we require. The algorithm terminates. Otherwise, proceed to *Step 3* and continue.

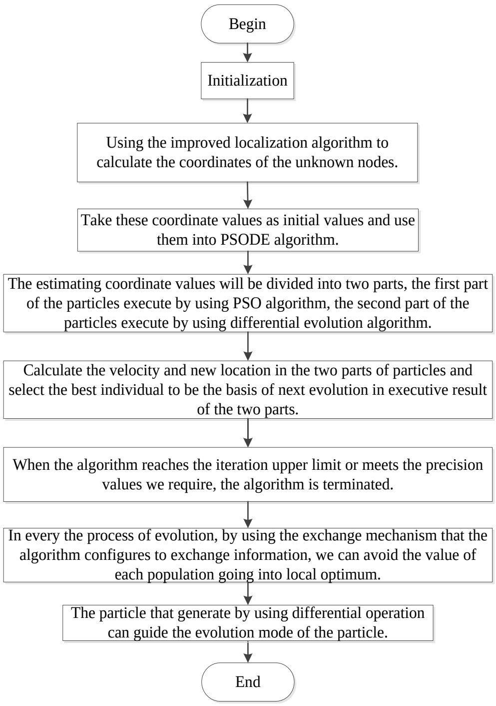


Fig. 2. The flowchart of the improved algorithm based on PSODE

D. *The Procedures of the New Algorithm*

In order to make the result of the improved DV-Hop algorithm more accurate, we propose an improved method. After the unknown nodes calculated their own estimated coordinate values by using the improved DV-hop algorithm, the PSODE algorithm is applied to the values to improve the localization accuracy.

These estimated coordinate values are divided into two parts. The first part of the particles execute by using particle swarm optimization (PSO) algorithm, the second part of the particles execute by using differential evolution (DE) algorithm. Then the velocities and the new locations of the particles in these two parts are calculated separately and the best individual is selected from the executive results of the two parts to be the initial value of the next evolution. When reaching the iteration upper limit or meeting the accuracy requirements, the algorithm terminates.

The specific procedures can be described as follow:

- The unknown nodes use the improved DV-hop algorithm to calculate their own estimating coordinate values.
- Take these coordinate values as initial values and use them into PSODE algorithm. The estimating coordinate values will be divided into two parts, the first part of the particles execute by using PSO algorithm, the second part of the particles execute by using differential evolution algorithm.
- Calculate the velocity and new location in the two parts of particles and select the best individual to be the basis of next evolution in executive result of the two parts.
- When the algorithm reaches the iteration upper limit or meets the precision values we require, the algorithm is terminated.
- In every the process of evolution, by using the exchange mechanism that the algorithm configures to exchange information, we can avoid the value of each population going into local optimum.
- The particle that generate by using differential operation can guide the evolution mode of the particle.

The flowchart of the improved algorithm based on PSODE is shown in Fig. 2.

IV. Simulation Results

A. *The Configuration of Experimental Environment*

The experiments are carried out based on MATLAB. We use WIN7 operating system. In the 150 * 150 simulation area, 150 nodes scatter randomly, the anchor node number is 40, the communication radius is 30, The black * represents unknown nodes and red * represents anchor nodes. The distribution of Sensor nodes is shown in Fig. 3.

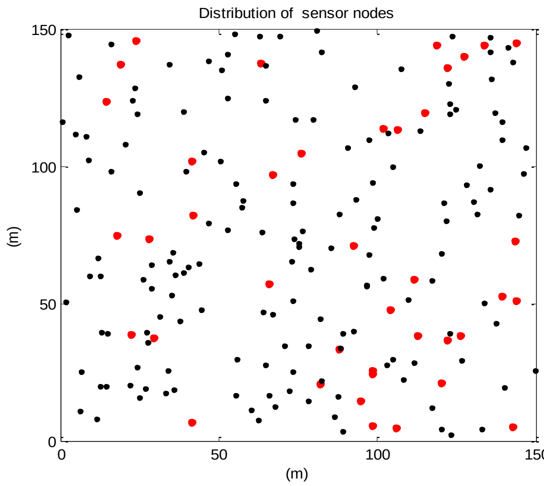


Fig. 3. The distribution of sensor nodes

B. *Compare Improved Algorithm Added PSODE with Improved Algorithm in Different Conditions*

In the paper, the localization error is tested through $e_{average}$ and $e_{nor\_average}$, the formulas are shown in (8) and (9). The localization error can be reflected from $e_{average}$ and $e_{nor\_average}$.

$$e_{average} = \frac{\sum_{i=1}^{n} \sqrt{(x_i - x_{ai})^2 + (y_i - y_{ai})^2}}{n} \tag{8}$$

$$e_{nor\_average} = \frac{\sum_{i=1}^{n} \sqrt{(x_i - x_{ai})^2 + (y_i - y_{ai})^2}}{nR} \times 100\% \quad (9)$$

- Compare the localization error in conditions of different number of anchor nodes

In the experimental area, the total nodes number is 150, the communication radius is 20, the number of anchor nodes is 5,20,35,50,65, 80, the experimental results is shown in Fig. 4.

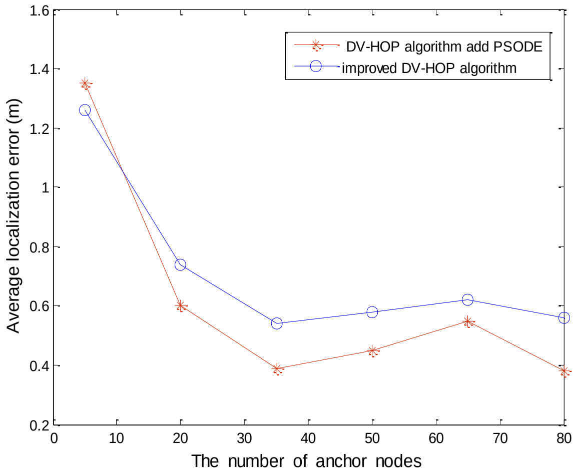


Fig. 4. localization error with different number of anchor nodes

With the change of nodes number, the two algorithms all have a decreasing trend of positioning accuracy. But the improved DV-Hop algorithm added PSODE is better than the improved DV-Hop algorithm in localization accuracy.

- Compare the localization error in conditions of different communication radius.

In the experimental area, the total nodes number is 120, the anchor nodes number is 30. The communication radius is 30,35,40,45,50,55,60. The experimental results is shown in Fig. 5.

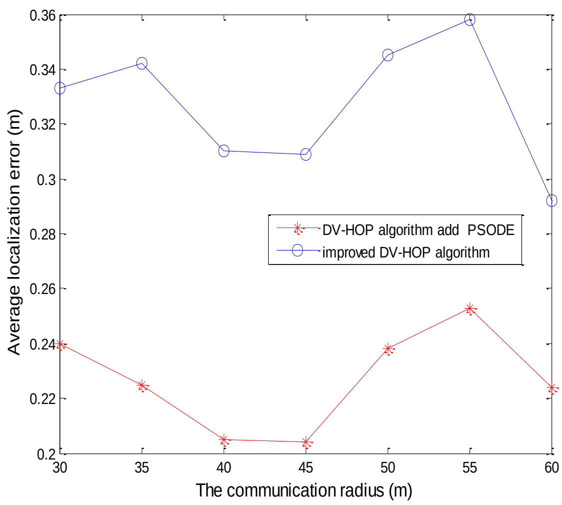


Fig. 5. localization error with different communication radius.

In the case of increasing the communication radius, the improved DV-HOP algorithm added PSODE is better than the improved DV-HOP algorithm in localization accuracy.

- Compare the localization error in conditions of different density of sensor nodes

In the experimental area, the communication radius is 40, The proportion of anchor nodes is 30%. The node number is 50, 100, 150, 200, 250, the experimental results is shown in Fig. 6.

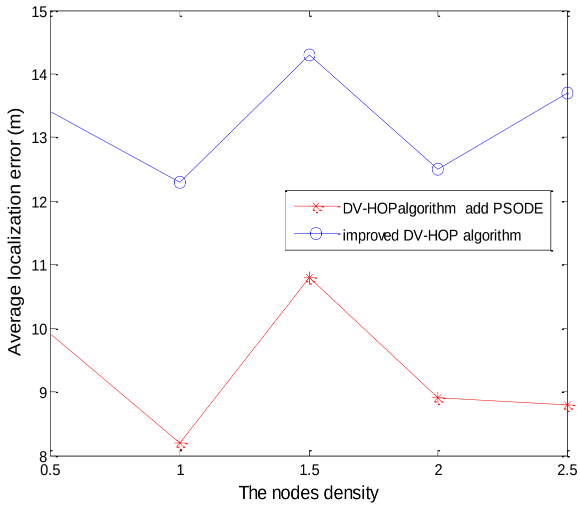


Fig. 6. localization error with different density of sensor nodes

In the case of increasing the node density, the improved DV-HOP algorithm added PSODE is better than the improved DV-HOP algorithm in localization accuracy.

V. Conclusion

In this paper, we further studied the traditional DV-Hop algorithm and an improved DV-Hop algorithm. Based on making the result of the improved DV-Hop algorithm more accurate, we propose a new method. Simulation results show that the new method has improved the positioning accuracy of the node significantly in WSN compared with the improved DV-Hop. However, the algorithm still has some deficiencies, such as the positioning problem of isolated nodes, communication overhead, etc. Therefore, about issue of node localization in WSN, we will do further research.

Acknowledgment

This work is supported by the Youth Foundation of Daqing Normal University. Grant No. 12ZR22.

References

[1] X. M. Du and Y. Chen, "Status and application of wireless sensor networks," *Beijing Technology and Business University Journal (Natural Science)*, Jan. 2008.

[2] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low cost outdoor localization for very small devices," *IEEE Personal Communications Magazine*, vol. 7, no. 5, pp. 28-34, 2000.

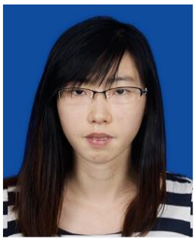
[3] J. Kennedy and R. Mendes, "Population structure and particle swarm performance," *Proceedings of the IEEE Congress on Evolutionary Computation*, 2002.

[4] F. Yang and H. S. Shi, "An intelligent localization algorithm based on distance measuring in WSN". *Sensing Technology Journal.*, vol. 21, no. 1, pp. 135-139, 2008.
[5] T. He, C. D. Huang, and B. M. Blum, "Range-free localization schemes in large scale sensor networks," in *Proc. 9th Annual Int'1 Conf. on Mobile Computing and Networking*, 2003, pp. 81-95.
[6] Y. Wang, X. D. Wang, D. Wang, and D. P. Agrawal, "Range-free localization using expected hop progress in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 10, pp. 1540-1552, 2009.
[7] G. F. Wan, J. Zhong, and C. H. Yang, "An improved RSSI ranging and positioning algorithm," *Research on Application of Computers*, vol. 29, no. 11, pp. 4156-4158, Nov. 2012.
[8] K. Benkic, M. Malajner, P. Planinsic, and Z. Cucej, "Using RSSI value for distance estimation in wireless sensor networks based on ZigBee," in *Proc. 15th International Conference on Systems, Signals and Image Processing*, 2008, pp. 303-306.
[9] H. Y. Chen, "Research on localization technology based on ranging technology," Southwest Jiaotong University, 2006.
[10] S. Ergut, R. R. Rao, O. Dural, and Z. Sahinoglu, "Localization via TDOA in a UWB sensor network using neural networks," *Communications*, 2008, pp. 2398- 2403.
[11] D. Niculescu and B. Nath, "Ad hoc positioning system (APS)," in *Proc. IEEE Global Telecommunications Conference*, San Antonio, Tex, USA, December 2001, vol. 5, pp. 2926–2931.
[12] Y. Zhou, X. Ao, and S. X. Xia, "An improved APIT node self-localization algorithm in WSN," in *Proc. 7th World Congress on Intelligent Control and Automation*, Chongqing, China, 2012, pp. 7582-7586.
[13] X. H. Nie, Y. L. Chen, X. Y. Liu, and W. P. An, "An improved weighted centroid location algorithm," in *Proc. 9th International Symposium on Linear Drives for Industry Applications*, China, 2013 , pp. 135-140.
[14] K. Shrawan and D. K. Lobiyal, "An advanced DV-hop localization algorithm for wireless sensor networks," *Wireless Personal Communications*, vol. 71, pp. 1365-1385, July 2013.
[15] J. M. Zhang, R. Yang, and J. Li, "An enhanced DV-hop localization algorithm using RSSI," *International Journal of Future Generation Communication and Networking*, vol. 6, no. 6, pp. 91-98, 2013.
[16] B. Su and W. J. Xue, "An improved algorithm based on Dv-Hop localization for wireless sensor network. Engineering and Applications (IEA) 2012," in *Lecture Notes in Electrical Engineering 220*, Springer-Verlag London, 2013, pp. 181-190.
[17] J. J. Liang and P. N. Suganthan, "Dynamic multi-swarm particle swarm optimizer," in *Proc. IEEE Swarm Intelligence Symposium*, June 2005, pp. 124-129.
[18] N. L. Li, "A novel PSO-DE-Based hybrid algorithm for global optimization," *Lecture Notes in Computer Sciences*, vol. 52, no. 27, pp. 156-163, 2008.
[19] L. J. Tan and N. Ben, "A new hybrid global optimization algorithm based on particle swarm optimization and differential evolution algorithm," *Information and Control*, vol. 36, no. 6, pp. 708-714, 2007.

Dan Zhang was born in December 1979. She received the master degree in Software Engineering form Jilin University, Changchun, China in 2006, and now she is a Ph.D. candidate in Computer science and Technology of Jilin University, her research interest includes localization, routing of wireless sensor networks.

Zhiyi Fang received the PhD degree in computer science from Jilin University, Changchun, China, in 1998, where he is currently a professor of computer science. He was a senior visiting scholar of the University of Queensland, Australia, from 1995 to 1996, and the University of California, Santa Barbara, from 2000 to 2001. He is a member of China Software Industry Association (CSIA) and a member of Open System Committee of China Computer Federation (CCF). His research interests include distributed/parallel computing system, mobile communication, and wireless networks.

Lin. Chen was born in February, in 1989. She received bachelor of engineering in Jilin University in China. Now she is pursuing master's degree in Computer Architecture from School of Computer Science and Technology, Jilin University, Changchun, China. Her research interests are mainly on the wireless sensor networks.



Hong-Yu. Sun was born in February, in 1986. She received the master degree in Computer Science and technology form Jilin University, Changchun, China in 2012, and now she is a Ph.D. candidate in Computer science and Technology of Jilin University, her research interest include localization, routing of wireless sensor networks.

Yue Wang was admitted to Jilin University,Jilin, China, in 2013.Now he is a sophomore of software engineering in Jilin University. His research interest is the area of computer network.