Node Localization of Wireless Sensor Networks Based on Hybrid Bat-Quasi-Newton Algorithm

http://dx.doi.org/10.3991/ijoe.v11i6.5110

Shunyuan Sun, Baoguo Xu
Jiangnan University, Wuxi, China

***Abstract*—Concerning the problem that the least square method in the third stage of DV-Hop algorithm has low positioning accuracy, a localization algorithm was proposed which is the fusion of hybrid bat-quasi-Newton algorithm and DV-Hop algorithm. First of all, the Bat Algorithm (BA) was improved from two aspects: firstly, the random vector β was adjusted adaptively according to bats fitness so that the pulse frequency had the adaptive ability. Secondly, bats were guided to move by the average position of all the best individuals before the current iteration so that the speed had variable performance; Then in the third stage of DV-Hop algorithm the improved bat algorithm was used to estimate node location and then quasi-Newton algorithm was used to continue searching for the node location from the estimated location as the initial searching point. The simulation results show that, compared with the traditional DV-Hop algorithm and the improved algorithm of DV-Hop based on bat algorithm (BADV-Hop). Positioning precision of the proposed algorithm increases about 16.5% and 5. 18%, and the algorithm has better stability, it is suitable for high positioning precision and stability situation.**

***Index Terms*—Bat Algorithm (BA), DV-Hop algorithm, quasi-Newton algorithm, Wireless Sensor Networks (WSN).**

I. INTRODUCTION

Wireless Sensor Network (WSN) is a special kind of self-organization (Ad hoc) network [1], which is widely used in military, industrial, transportation, environmental protection and other fields [2]. The localization of sensor nodes in wireless sensor networks is not only the basis of the research of wireless sensor networks [3], but also an essential part. Therefore, the node localization technology is an important research topic of the current wireless sensor network.

According to the different positioning mechanism, the wireless sensor network localization algorithm can be divided into range-based and range-free localization algorithm. range-free localization algorithm includes centroid algorithm [4], DV-Hop algorithm [5], Approximate Point-In-Ttriangulation test algorithm (APIT) [6] and Amorphous algorithm [7]. Due to its advantages of the hardware requirements, network deployment cost, energy consumption and other aspects, range-free localization algorithm is more suitable for Wireless Sensor Networks [8]. The Vector-Hop Distance (DV-Hop) algorithm, which transforms distance measurement between nodes to the product of hop count and average hop distance, is one of the most widely studied algorithms [9].

For the large error problem of the least square method in DV-Hop algorithm, many scholars put forward the improved DV-Hop algorithms based on intelligent algorithms. For example, Document [10] introduced the Particle Swarm Optimization algorithm(PSO) into DV-Hop to improve the estimation of position; Document [11] proposed the Artificial Bee Colony algorithm into DV-Hop algorithm; Document [12] proposed the hybrid GA and DV-Hop algorithm, which is composed by Genetic Algorithm (GA) and the simplex DV-Hop algorithm; Document [13] proposed the genetic Simulated Annealing algorithm as the post optimization of DV-Hop algorithm; Document [14] proposed improved DV-Hop algorithm based on Bat Algorithm (BA). BA is a kind of bionic intelligence algorithm which is raised in recent years, compared with other algorithms, it has a great advantage in the iterative optimization and has less parameters to adjust [15], as a result, it received wide attention. Document [14] combined BA and DV-Hop algorithm together, although it achieved some results, but they are still not ideal. In this paper, hybrid bat-quasi-Newton algorithm is proposed. The Bat algorithm is improved respectively in the pulse frequency and speed, and then the quasi Newton algorithm is used to improve the algorithm. The simulation results show that the positioning accuracy of the hybrid algorithm is significantly improved, and the stability is better.

II. DESCRIPTION OF POISITION PROBLEM

In the first and second stages of DV-Hop algorithm, the distance between the unknown node and the anchor node is obtained by the hop count and the average hop distance between nodes. Assuming that the estimated distance of the unknown node B(x,y) to the anchor node $A_1(x_1,y_1)$,$A_2(x_2,y_2)$,$A_3(x_3,y_3)$….. $A_n(x_n,y_n)$ is $d_1,d_2, d_3…. d_n$, and the ranging error is ε_1, ε_2, ε_3…. ε_n, the following equations can be got:

$$\begin{cases} d_1^2-\varepsilon_1^2 \le (x-x_1)^2+(y-y_1)^2 \le d_1^2+\varepsilon_1^2 \\ d_2^2-\varepsilon_2^2 \le (x-x_2)^2+(y-y_2)^2 \le d_2^2+\varepsilon_2^2 \\ \quad\quad\quad\quad \mathrm{M} \\ d_n^2-\varepsilon_n^2 \le (x-x_n)^2+(y-y_n)^2 \le d_n^2+\varepsilon_n^2 \end{cases} \quad (1)$$

First of all, unknown nodes coordinate (x,y) satisfies all inequalities, and if the sum of ε_1^2, ε_2^2, ε_3^2…. ε_n^2 is smaller, the location is estimated more exactly, then positioning problem is transformed into finding the minimum of nonlinear equations, that means to find coordinates (x,y) which minimize $f(x,y)$ of formula (2), where minimum $f(x,y)$ ensures the minimum error sum. So the formula (2) is defined as fitness function, which is used to evaluate the location of bats and guide the search direction of the algorithm.

$$f(x,y)=\frac{1}{M}\sum_{i=1}^{M}\left|\sqrt{(x-x_i)^2+(y-y_i)^2}-d_i\right| \quad (2)$$

where M is the number of anchor nodes.

III. IMPROVEMENT OF BAT ALGORITHM

A. Original Bat algorithm

BA is a kind of intelligent optimization algorithm proposed by *Yang* in 2010 which simulates the echo location of bat. Compared with other algorithms, it has great advantages.

Firstly, the position and speed of the bat are randomly initialized, and the position is expressed as the candidate solution. Then the optimal position of the initial population is found out by fitness function. Lastly, the position and speed of the individual is updated by adjusting the pulse frequency. The formula is as follows:

$$\boldsymbol{f}_i = f_{\min} + (f_{\max} - f_{\min})\boldsymbol{\beta} \quad (3)$$

$$\boldsymbol{v}_i^{\,t} = \boldsymbol{v}_i^{\,t-1} + (\boldsymbol{x}_i^{\,t-1} - \boldsymbol{x}_*)\boldsymbol{f}_i \quad (4)$$

$$\boldsymbol{x}_i^{\,t} = \boldsymbol{x}_i^{\,t-1} + \boldsymbol{v}_i^{\,t} \quad (5)$$

where $\boldsymbol{\beta}$ is a random vector of uniform distribution, and $\beta \in [0,1]$ is component of $\boldsymbol{\beta}$, $\boldsymbol{v}_i^t$,$\boldsymbol{x}_i^t$ are the speed and location of the bat i at the t moment, $\boldsymbol{f}_i \in [f_{\min}, f_{\max}]$ is the pulse frequency of the bat i, $\boldsymbol{x}_*$ is the optimal location for all bats.

In local search, once a solution is selected from the optimal solution set, a new local solution can be got through random walk:

$$\boldsymbol{x}_{new} = \boldsymbol{x}_{old} + \varepsilon A^t \quad (6)$$

where ε is a random number ranging in [-1,1], A^t is the average loudness of all bats at this step.

According to the bat's echo location mechanism, bat's pulse shows low rate and the loudness in the initial stage, it is beneficial to search for targets in a large range. Once the prey is found, pulse loudness becomes smaller and the rate becomes large to accurately grasp the spatial position of target. The update formula of pulse emission loudness A_i^t and rate r_i^t are:

$$A_i^t = \alpha A_i^{t-1} \quad (7)$$

$$r_i^t = r_i^0(1-e^{-\gamma(t-1)}) \quad (8)$$

where: $0<\alpha<1$, $\gamma>0$, are constants.

The update mode of bat position and speed is similar to the standard particle swarm algorithm, $\boldsymbol{f}_i$ controls the speed and range of the movement of the particles, to a certain extent, BA can be regarded as a balanced combination of the standard particle swarm algorithm and local search controlled by pulse loudness and rate [1]. It has the advantages of simplicity, robustness and easy to implement, but because of the lack of population diversity, BA has the disadvantages of easy to fall into local optimum, slow convergence speed and low search accuracy, these problems are improved from two aspects -- random vector β and the speed v_i^t in the following.

B. Improvement of β

The bat regulates the pulse frequency by adjusting β, the bigger the β value is, the higher the frequency is, the larger the speed and the motion of the bat are, and this will be beneficial to large range global search. Meanwhile, the smaller the β value is, the lower the frequency is, the smaller the speed and the motion of the bat are, and this will be beneficial to small range local search. When the distance of the bat is far from the optimal position, it is required to use higher value of β to search the target in a large range. When the distance of the bat is close to the optimal position, the lower value of β is required to determine target in a small range. In the original algorithm, β is a random vector, which can not be adjusted adaptively to search the target by the bat position. Therefore, an adaptive updating formula is proposed to improve the accuracy and convergence rate of the algorithm.

$$\boldsymbol{\beta}_j = \beta_{\min} + (\beta_{\max} - \beta_{\min}) \times (1 - |randn|) \times \cos\left(\frac{\pi \times (fit_i - fit_{best})}{2 \times (fit_{worst} - fit_{best})}\right) \quad (9)$$

where : $\boldsymbol{\beta}_j$ is the value of the j dimension of $\boldsymbol{\beta}$, (j=1,2…d), β_{min} and β_{max} are the minimum and maximum values for the j dimension of β, fit_{worst} and fit_{best} are the best and worst current fitness value for all bats, fit_i is the fitness value of the bat i, *randn* is a random number of Gauss distribution which has mean 0 and variance 1.

C. Improvement Of Speed

BA is similar to the PSO algorithm, in which the bats move to the direction of optimal position in the later evolution, resulting in the diversity of the population and slow convergence speed becomes, and easy to fall into local optimum. Due to lack of mutation mechanism to escape from local optimum, BA's search results are often local optimum, as a result algorithm search accuracy becomes low, therefore, an improved speed update mode is proposed:

$$\boldsymbol{v}_i^{\,t} = (\boldsymbol{x}_i^{\,t-1} - \boldsymbol{x}_*^{\,t-1})\boldsymbol{f}_i + \alpha(\boldsymbol{mean}^{t-1} - \boldsymbol{x}_i^{\,t-1}) \times randn \quad (10)$$

$$\boldsymbol{mean}_j^{\,t-1} = \frac{\sum_{n=0}^{t-1} \boldsymbol{x}_*^{\,n}(j)}{t} \quad j = 1,2\ldots d \quad (11)$$

where α is a constant factor, $\boldsymbol{x}_*^{n}(j)$ is the j-dimensional value of the best current position in n-th iteration, where $\boldsymbol{x}_*^{0}(j)$ is the j-dimensional value of the optimal position in the initial population, $\boldsymbol{x}_i^{0}$ is the initial position of the bat i, $\boldsymbol{mean}_j^{t-1}$ is the j-dimension of average optimal value in t-th iteration.

Improved speed process is to use the average position of best individuals from all iterations before contemporary generation to guide the bat mobile, not just use the current best position to guide the bat, thus not only the ability to escape from local optimum, but also the diversity of the population is improved, eventually search algorithm accuracy is improved.

IV. QUASI-NEWTON ALGORITHM

Quasi-Newton algorithm is an effective method for solving nonlinear optimization problems,

It overcomes the Newton method's request's guidance and inverse shortcomings in the calculation process, this article using the variable metric method (Broyden-Fletcher-Goldfarb-Shanno, BFGS):

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \lambda_k \boldsymbol{H}_k \boldsymbol{g}_k \tag{12}$$

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_{k+1} - \frac{\boldsymbol{H}_k \boldsymbol{y}_k \boldsymbol{s}_k^T + \boldsymbol{s}_k \boldsymbol{y}_k^T \boldsymbol{H}_k}{\boldsymbol{s}_k^T \boldsymbol{y}_k} + (1 + \frac{\boldsymbol{s}_k^T \boldsymbol{H}_k \boldsymbol{y}_k}{\boldsymbol{s}_k^T \boldsymbol{y}_k}) \frac{\boldsymbol{s}_k \boldsymbol{s}_k^T}{\boldsymbol{s}_k^T \boldsymbol{y}_k} \tag{13}$$

$$\boldsymbol{s}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k \tag{14}$$

$$\boldsymbol{y}_k = \boldsymbol{g}_{k+1} - \boldsymbol{g}_k \tag{15}$$

Calculated as follows:

Step 1 Given the initial position $\boldsymbol{x}_0$, positive definite matrix $\boldsymbol{H}_0$=$\boldsymbol{I}_2$, set the maximum number of iterations G_{max}, allowable error$\varepsilon_0 \in [0,1]$, so that k=0, where $\boldsymbol{g}_k = \nabla f(\boldsymbol{x}_k)$,$f(\boldsymbol{x})$is fitness function.

Step 2 If $\|\boldsymbol{g}_k\| < \varepsilon_0$ or $k > G_{max}$, then the algorithm ends, and output estimation position $\boldsymbol{x}_k$; otherwise, compute $\boldsymbol{d}_k = -\boldsymbol{H}_k \boldsymbol{g}_k$.

Step 3 Carry out linear search along $\boldsymbol{d}_k$ direction to find out λ_k which satisfies the search condition wolf, so that $f(\boldsymbol{x}_k+\lambda_k \boldsymbol{d}_k)=\min f(\boldsymbol{x}_k+\lambda \boldsymbol{d}_k)$, and then $\boldsymbol{x}_{k+1}$ is obtained by the formula (12).

Step 4 calculate $\boldsymbol{g}_{k+1}$, use equation (13) to correct $\boldsymbol{H}_k$, then obtain $\boldsymbol{H}_{k+1}$, so that k=k+1, return to step 2.

V. HYBRID BAT-QUASI-NEWTON POISITION ALGORITHM

A. Specific Steps

Step 1 Deploy anchor nodes and unknown nodes in the monitored area randomly, then estimate the Distance between unknown node and the anchor node through the first and second phase of DV-Hop algorithm.

Step 2 Set algorithm parameters: loudness A_i, rate r_i, tolerance error ε_0. Randomly initialize speed $\boldsymbol{v}_i$ and position $\boldsymbol{x}_i$ of bat i, evaluate the merits of initial position of the bat by the fitness function, save the best location $\boldsymbol{x}_*$ and the best value $f(\boldsymbol{x}_*)$.

Step 3 Adjust the pulse frequency f_i according to formula (3) and (9), then updated speed $\boldsymbol{v}_i$ and position $\boldsymbol{x}_i$ of bat i according to the formula (4)(5)(10)(11), and conduct cross-border treatment.

Step 4 Generate a random number $rand_1$, if $rand_1 > r_i$, select a solution in optimal solution set, and generate new solutions $\boldsymbol{x}_{new}$ in its vicinity according to formula (6), followed by cross-border treatment; Otherwise, generate $\boldsymbol{x}_{new}$ in the vicinity of position $\boldsymbol{x}_i$ according to the formula (6), and conduct cross-border treatment.

Step 5 Generate a random number $rand_2$, if $rand_2 < A_i$ and $f(\boldsymbol{x}_{new}) < f(\boldsymbol{x}_*)$, accept the new solution $\boldsymbol{x}_{new}$, then update loudness A_i and rate r_i according to equation (7)(8).

Step 6 Arrange bats according to the fitness value to find and save the current optimum $\boldsymbol{x}_*$.

Step 7 If the maximum number of iterations is satisfied, output of the global best position $\boldsymbol{x}_{best}$ and proceed to step 8; otherwise, return to step 3.

Step 8 Set $\boldsymbol{x}_{best}$ as initial position, use quasi-Newton algorithm to search target position, if termination condition is reached, the algorithm ends, and output the best individual location, which is the coordinate of unknown nodes.

B. Time Complexity Analysis

Suppose the search space is 2-dimensional, the population size is n, the maximum number of iterations of improved bat algorithm, quasi-Newton algorithm, wolf search were T_{max}, G_{max}, g.

1) Time complexity of the improved bat algorithm.

The time complexity of updating random vector $\boldsymbol{\beta}$ and the pulse frequency $\boldsymbol{f}$ is $o(T_{max}\times n)$, of updating speed $\boldsymbol{v}$ and the position $\boldsymbol{x}$ is $o(T_{max}\times n)$, of local search is $o(T_{max}\times n)$, of accepting new solutions and updating loudness and pulse rate is $o(T_{max}\times n)$, of arranging bats and finding the current best value is $o(T_{max}\times n \log_2 n)$.

2) Quasi-Newton algorithm time complexity.

The time complexity for wolf searching is $o(G_{max}\times g\times n)$, and for other statements computation is $o(G_{max}\times n)$.

In summary, the time complexity of hybrid bat - Quasi-Newton positioning algorithm is T(n)= $o(T_{max}\times n \log_2 n)$+ $o(G_{max}\times g\times n)$+ $o(G_{max}\times n)$, where Gma and g are inappropriate for great value, because greater G_{max} and g value result in higher time complexity of hybrid algorithm. g is generally taken as 10, 20 for experiment. It can be found that linear relationship exists among G_{max}, $G_{max}\times g$ and T_{max}, so after reduction algorithm, hybrid algorithm time complexity T(n)= $o(T_{max}\times n \log_2 n)$ is in the same order of magnitude with the bat algorithm $o(T_{max}\times n \log_2 n)$.

VI. SIMULATION ANALYSIS

A. Simulation Environment and Parameters

To verify the performance of hybrid bat-Quasi Newton positioning algorithm, simulation tests of traditional DV-Hop algorithm, the BADV-Hop algorithm of document[14] and the proposed algorithm are carried out in Matlab R2010b platform. The performances of the three algorithms are compared respectively in three aspects, anchor node ratio, communication radius and the number of nodes, in the same network environment.

Use the average location error to evaluate the merits of positioning performance, the average localization error is defined as:

$$error = \frac{\sum_{k=1}^{K}\sum_{i=1}^{N}\sqrt{(x^*_i - x_i)^2 + (y^*_i - y_i)^2}}{N\times R\times K} \tag{16}$$

where (x^*_i, y^*_i) is the estimated coordinate of the unknown node i, (x_i, y_i) is the actual coordinate of the unknown node i, N is the number of unknown nodes, R is the communication radius, K is the experimental times number.

In the rectangular area 100m×100m, randomly distribute 100 sensor nodes with communication radius of

30m. In BA, set the population size as 100, the frequency range [0,100], the maximum loudness A^0=0.25, the maximum rate r^0=0.75, loudness coefficientα=0.95, the rate coefficientγ=0.05. Due to the randomness of the experiment, all the experimental results are average of 30 experiments.

B. *Anchor Node Ratio's Effect on the Positioning Accuracy*

Randomly distribute 100 sensor nodes in the square area, with communication radius 30m and the anchor node ratio of 20% to 45%. As shown in "Fig. 1", comparatively analysis the performance of the three algorithms positioning in different anchor node ratios. With the increase of the anchor node ratios, average positioning error of three algorithms gradually decreased, which is because the increase of the anchor node ratios results in increase of the number of nodes involved in targeting, then ranging error becomes smaller, thereby positioning error decreases. Compared to conventional DV-Hop algorithm and BADV-Hop algorithm, positioning accuracy of this proposed algorithm has increased by 16.6% and 5.41%.

C. *Number of Nodes' Effect on the Positioning Accuracy*

Randomly distribute 100 sensor nodes in the area, with communication radius 30m and the anchor node ratio of 30% and number of nodes from 50 to 150. Compare the performance of traditional DV-Hop algorithm, BADV-Hop algorithm and the proposed algorithm by adjusting the number of nodes, as shown in "Fig. 2". Given different number of nodes, the average positioning error of the proposed algorithm is always less than the traditional DV-Hop algorithm and BADV-Hop algorithm. Compared to these two algorithms, positioning accuracy of this proposed algorithm improves 17.85% and 5.31%.

D. *Communication Radius' Effect on the Positioning Accuracy*

Randomly distribute 100 sensor nodes in the area, wherein the number of anchor nodes is 30, with communication radius of 25 ~ 50 m, as shown in "Fig. 3". Compare the performance of three algorithms by adjusting the size of the communications radius. Compared to conventional DV-Hop algorithm and BADV-Hop algorithms, positioning accuracy of this proposed algorithm is increased by 15.07 and 4.84%.

E. *Stability and Complexity Analysis*

Randomly distribute 100 sensor nodes in the area, wherein the number of anchor nodes is 30, with communication radius of 30m. Respectively implement simulation testing on traditional DV-Hop algorithm, DV-Hop algorithm and the proposed algorithm, to obtain the mean and standard deviation of the average positioning difference of 30 experiments, and the average running time of locating each unknown node, as shown in "Tab. 1". Minimum mean and variance of the proposed algorithm indicates higher precision and better stability. Compared to BADV-Hop algorithm, the running time of the proposed increases 0.0027s<0.005s, which means the operation efficiency of the proposed algorithm is equivalent to BADV-Hop algorithm.

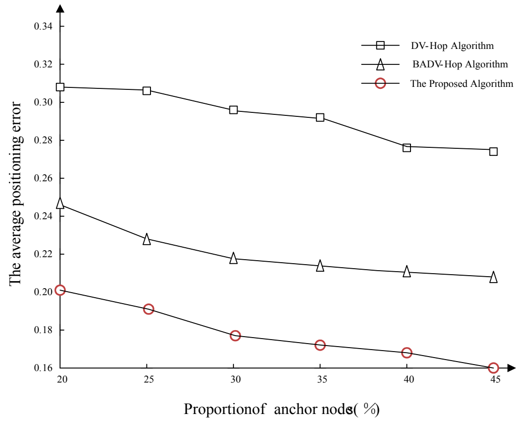

Figure 1. Anchor Node Ratio's Effect on the Positioning Accuracy

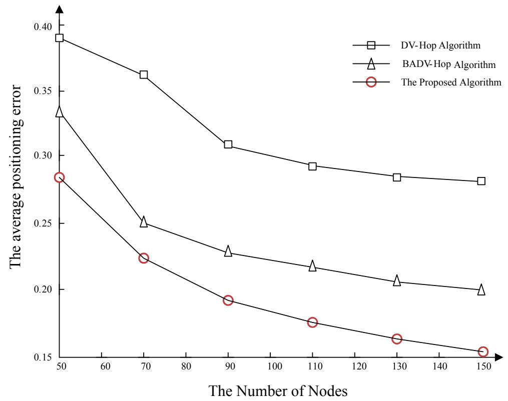

Figure 2. Number of Nodes' Effect on the Positioning Accuracy

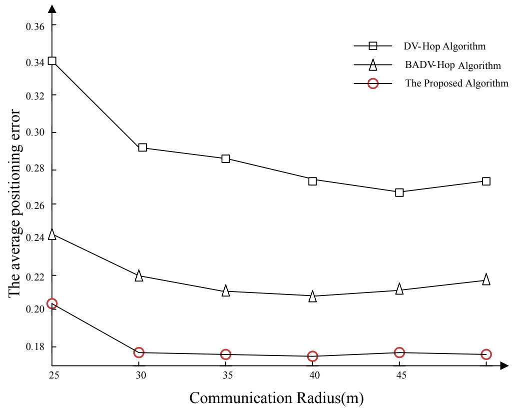

Figure 3. Communication Radius' Effect on the Positioning Accuracy

TABLE I.
STABILITY AND COMPLEXITY ANALYSIS

algorithm	mean	Standard deviation	average running time (s)
DV-Hop algorithm	0.2883	0.0333	——
BADV-Hop algorithm	0.2187	0.0185	0.4065
The proposed algorithm	0.1798	0.0167	0.4092

VII. CONCLUSION

In order to improve the performance of DV-Hop localization algorithm, a hybrid bat - Quasi-Newton positioning algorithm is proposed. Because bats algorithm is easy to fall into local optimum search and has low search accuracy, this article improves it from the pulse frequency and speed, also with the quasi-Newton algorithm applied to the third phase of DV-Hop algorithm. Simulation results show that: compared to conventional DV-Hop algorithm and BADV-Hop algorithm, this algorithm shows high positioning accuracy and good stability, but the disadvantage is the addition of certain computational complexity.

ACKNOWLEDGMENT

The authors thank the support from the Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), the Fundamental Research Funds for the Central Universities.

REFERENCES

[1] Estrin D, Govindan R, and Heidemann J, "Next century challenges: scalable coordination in sensor network," *Proceeding of Fifeh Annual International Conference on Mobile Computing and Networks [C]*. Washington, USA, 1999. pp. 263-270. http://dx.doi.org/10.1145/313451.313556

[2] Ma Z, Sun Y, "Mei T. Survey on Wireless Sensor Network," Journal on Communications, 2004,25(4): pp. 114-124.

[3] Zhao Yan-hang, Qian Zhi-hong, Shang Xiao-hang, and CHENG Chao, "PSO localization algorithm for WSN nodes based on modifying average hop distance," *Journal on Communications*, 2013, 34(9): pp. 105-114.

[4] Hong C, Sezaki K, Ping D, et al, "An Improved DV-Hop Localization Algorithm with Reduced Node Location Error for Wireless Sensor Networks," *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, 2008, 91 (8): pp. 2232-2236.

[5] Chen H, Sezaki K, Deng P, et al, "An Improved DV-Hop Localization Algorithm for Wireless Sensor Networks," *Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference on IEEE*, 2008: pp.1557-1561.

[6] Tian S, Zhang X, Liu P, et al. "A RSSI-Based DV-Hop Algorithm for Wireless Sensor Networks[C]//*Wireless Communications," Networking and Mobile Computing, 2007. WiCom 2007. International Conference on IEEE*, 2007: pp. 2555-2558.

[7] Zhou Z, Xiao M, Liu L, et al. "An Improved DV-HOP Localization Algorithm," *Information Science and Engineering (ISISE), 2009 Second International Symposium on IEEE*, 2009: pp. 598-602. http://dx.doi.org/10.1109/isise.2009.79

[8] Yang W, Pan W. "DV-Hop localization algorithm based on RSSI ratio correction in wireless sensor network," T*ransducer and Microsystem Technologies*, 2013, 32(7): pp. 126-135.

[9] Wen J, Fan X, WU, and X. "DV-Hop localization algorithm based on the RSSI hop correction," *Chinese Journal of Sensors and Actuators*, 2014, 27(1): pp. 113-117.

[10] Zhao J, Fu Y, and MEI J. "An Improved DV-Hop Localisation Algorithm Based On Particle Swarm Optimisation," *Computer Applications and Software*, 2012, 29(12): pp. 69-76.

[11] Li M, Xiong W, and Guo L. "Improvement of DV-Hop Localization Based on Artificial Bee Colony Algorithm," *Computer Science*, 2013,40(1): pp. 33-36.

[12] Liu Y, Lv X, and Wang X. "Application of hybrid genetic algorithm in WSNs localization," *Transducer and Microsystem Technologies*, 2014, 33(2): pp. 150-153.

[13] Zhao S, Sun M, and Tang Y. "Gasa-Based Localization Alogorithm For Wireless Sensor Networks," *Computer Applications and Software*, 2009, 26(10): pp. 189-192.

[14] Zhao W, Li F, Zheng H, Liu Y, and Feng H. "Node localization method of WSNs based on BADV-Hop," *Transducer and Microsystem Technologies*, 2014, 33(10): pp. 118-124.

[15] Yang X S, Gandomi A H. "Bat algorithm: A novel approach for global engineering optimization," *Engineering Computations*, 2012, 29(5): pp. 464-483 http://dx.doi.org/10.1108/02644401211235834

AUTHORS

Shunyuan Sun and **Baoguo Xu** are with the School of Internet of Things Engineering and with the Key Laboratory of Advanced Process Control for Light Industry (Jiangnan University), Ministry of Education, Wuxi 214122, China.

This work has been suported by the the Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), the Fundamental Research Funds for the Central Universities, No. JUSRP11560 and JUSRP51510. Submitted 10 October 2015. Published as resubmitted by the authors 30 October 2015.