

## Positioning of Wireless Sensor Network under Emergency Communication Environment

Ruilin Yuan

Hebi Polytechnic, Hebi 458030, China

Corresponding Author Email: [3067349@163.com](mailto:3067349@163.com)



<https://doi.org/10.18280/i2m.190404>

### ABSTRACT

**Received:** 8 April 2020

**Accepted:** 29 June 2020

#### Keywords:

*wireless sensor, emergency communication, multi-hop positioning, Kalman filter, distributed computing*

With the development of microelectromechanical system (MEMS), embedded system, and wireless communication, it is now feasible to implement and deploy wireless sensor network (WSN) in emergency communication environment. However, the positioning accuracy of WSN nodes needs to be further improved. To solve the problem, this paper improves the initial value calculation method of multi-hop positioning algorithms, which are suitable for emergency communication environment, and puts forward a WSN node positioning algorithm that narrows the initial values of Kalman filter. By narrowing the initial value range of Kalman filter, the specially deployed sensors could accurately derive its position from the known positions of anchor nodes. To prevent error accumulation in the network, distributed computing was performed to solve the global nonlinear optimization problem, and calculate the position of the nodes. Simulation results show that the proposed algorithm can improve the WSN positioning accuracy under emergency communication environment, while greatly saving computing and communication costs. The research further improves the practicability and efficiency of multi-hop positioning algorithms in emergency communication environment.

## 1. INTRODUCTION

In recent years, wireless sensor networks (WSNs) have been widely adopted in various fields, namely, military reconnaissance, traffic monitoring, and emergency communications [1], owing to its advantages in information acquisition and processing.

Node positioning algorithm is a hotspot in WSN research. Depending on the application scenarios, many WSN positioning methods have been developed with varied positioning principles [2]. These methods can be classified based on the characteristic parameters (e.g. distance or angle between nodes [3-6], or categorized by ranging methods [7-9]. Hightower and Boriello [10] explored deep into node positioning systems, and summed up many node positioning algorithms and systems.

Similar node positioning methods exist in many other application fields [11, 12]. For instance, the positioning problem in the field of mobile robots bears high resemblance with that in WSN. However, there is a marked difference between mobile robot positioning system and conventional WSN positioning system under emergency communication: the former relies on a mileage measurement tool to estimate the initial position of the robot, while the latter does not have such an instrument. Under emergency communication, special consideration should be given to such properties of the WSN as scalability, communication efficiency, and power consumption, which might not need to be considered under other scenarios.

In actual applications of emergency communication, it is necessary to process the error in measured values to improve the accuracy of node positioning, if the propagation of wireless signal is affected by environments like tunnels, dense forests,

and cities. Kalman filter algorithm is the most widely used method to process the measuring error [13]. This algorithm has been constantly improved, creating advanced algorithms like extended Kalman filter algorithm [14] and H-infinite filter [15].

In addition, Nicolescu and Nath [16] proposed an ad hoc positioning method for WSN nodes. In this method, the position information of anchor nodes is propagated in the network. Once an anchor node is aware of the position of another anchor node, it will derive the mean hop length in its neighborhood from this information, and feedback the result to the network. In the meantime, each node whose position is unknown will record the shortest hop length to each anchor node, and multiply it with the mean hop length to obtain its approximate distance to each anchor node. In this way, each node could preliminarily estimate its position through multiple queries.

For better estimation of node position, Rabaey and Langendoen [17] improved the positioning accuracy through further refinement, e.g. computing refined node positions by least squares (LS) method. Simulation results show that their algorithms are independent of ranging technology, and capable of accurate positioning within one-third of the communication range.

In most application scenarios, WSN nodes are deployed in a specific way. However, with specific deployments, the position of each sensor cannot be predicted accurately or planned in advance. This calls for accurate positioning of sensors under indirect line-of-sight (LOS), using multi-hop positions and ranging data. To this end, many node positioning algorithms [18] have been developed based on hop count, in which the target node is positioned based on the hop count and distance between it and the anchor node. Among them, the

most common algorithms are distance vector (DV) hop and its improved versions [19, 20]: each node measures its distance to a neighbor with its ranging sensor, and shares the measured value and its position with the anchor node, which is aware of its own position.

In addition, Savvides et al. [21] designed an ad hoc localization system (AHLoS) to solve sensor positions in the WSN through the iteration of multiple associations. In collaboration with multiple extraction algorithms, the AHLoS overcomes problems like the sensitivity to anchor node density and the error propagation in large networks.

Drawing on the above research results, this paper attempts to solve the limited accuracy and high computing cost of WSN node positioning under emergency communication environment. Inspired by the strategies of multi-hop positioning algorithms and initial value optimization, the authors put forward an improved multi-hop collaborative positioning algorithm for WSN nodes. The accuracy of multi-hop node positioning was improved by narrowing the range of initial values of Kalman filter. The proposed algorithm was proved valid through case analysis.

The remainder of this paper is organized as follows: Section 2 presents the improved multi-hop collaborative positioning algorithm for WSN nodes based on initial value optimization, which overcomes the defects of existing algorithms in emergency communication environment; Section 3 verifies the proposed algorithm through case analysis, and illustrates the effectiveness of the algorithm through contrastive analysis on accuracy, computing cost, and energy consumption; Section 4 puts forward the conclusions of this research.

## 2. METHODOLOGY

### 2.1 Kalman filter

Invented in the 1960s, the Kalman filter mainly estimates the state of the current moment with the minimum mean square error (MMSE) through recursion of the estimate of the previous moment and the observation of the current moment. A typical Kalman filter can be implemented in two steps: estimation and verification. The estimation calculates the a priori value of the current state by the time update equation:

$$\hat{X}_k^- = A\hat{X}_{k-1} + B\hat{U}_{k-1} \quad (1)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (2)$$

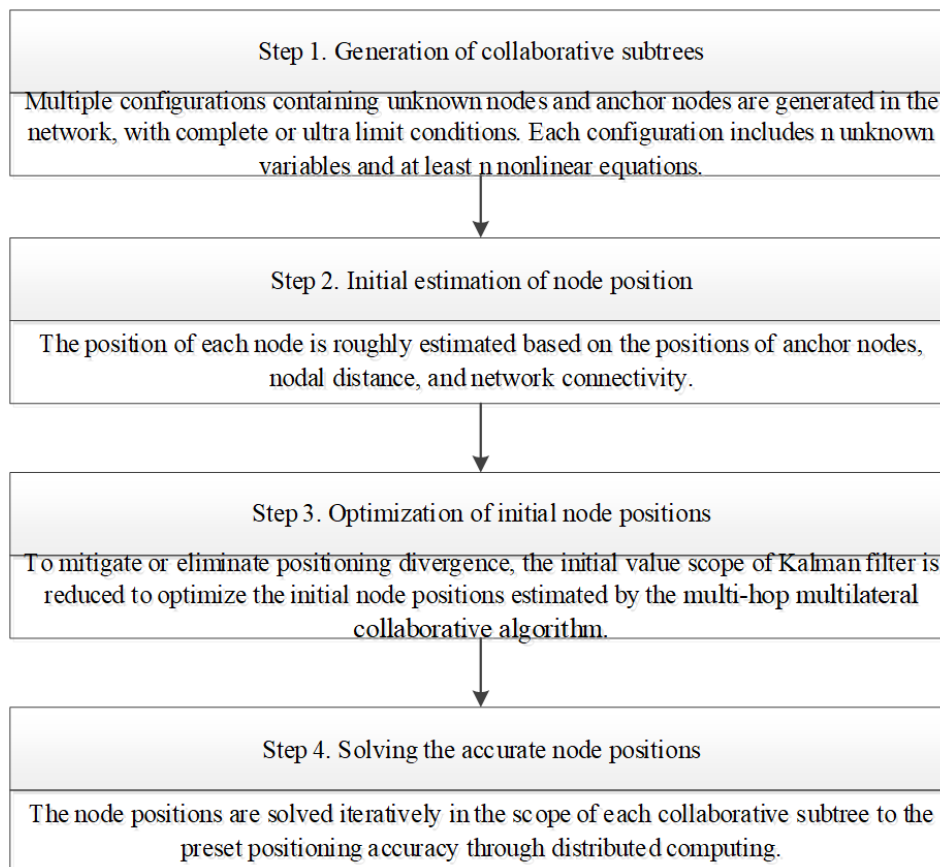
The verification calculates the posterior estimation of the improved current state by the measurement update equation:

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (3)$$

$$\hat{X}_k = \hat{X}_k^- + K(Z_k - H\hat{X}_k^-) \quad (4)$$

$$P_k = (I - K_k H) P_k^- \quad (5)$$

### 2.2 Improved multi-hop collaborative positioning algorithm



**Figure 1.** The workflow of the improved algorithm

Sawides et al. [22] proposed a multi-hop multilateral collaborative algorithm based on the AHLoS algorithm. Their algorithm iteratively refines the positioning results with Kalman filter, curbs the accumulation of error to a certain extent, and improves the accuracy of node positioning. Moreover, the sufficient conditions are provided for judging whether a node could join the multilateral collaboration. However, the algorithm faces a certain degree of positioning divergence, and requires strong computing power to complete the complex calculations.

To further improve the accuracy and efficiency of the algorithm, this paper proposes an improved multi-hop collaborative positioning algorithm that narrows the initial value range of the Kalman filter. The narrowing weakens the positioning divergence, and improves the effects of subsequent filtering and positioning. The improved algorithm consists of four steps (Figure 1).

**Step 1. Generation of collaborative subtrees**

The collaborative subtree is composed of several unknown nodes and anchor nodes. Each unknown node has a unique position. The positions of unknown nodes are solved by the equation set of each collaborative subtree, including  $n$  unknown variables and at least  $n$  equations. Before solving these equations, the uniqueness of node position must be confirmed to prevent incorrect positioning. The subtrees of different structures need to meet different structural requirements for the uniqueness of the position.

**(1) Requirements of single-hop multilateral collaboration**

In single-hop setting, the basic requirement for an unknown node to have a unique position on a two-dimensional (2D) plane is that it must fall within the range of three noncollinear anchor nodes. There are more than one potential positions, if the anchor nodes are on the same straight line, and if the node configuration is symmetric.

**(2) Requirements of multi-hop multilateral collaboration**

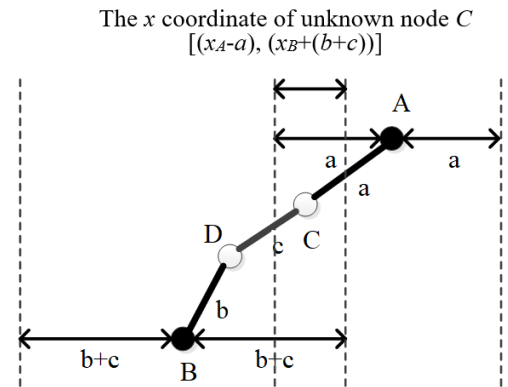
A set of similar criteria are adopted to judge whether a node within  $n$  hops has a unique position. Starting from an unknown node, it is tested whether the node has at least three neighbors with temporarily unique positions. If the node has three neighbors that do not know whether its position is unique, recursive calls will be conducted at each neighbor to determine whether its position is unique. To ensure that each node has at least one link, every node as an independent reference is marked as “used”. This prevents other nodes from reusing the node as an independent reference in subsequent recursive calls. In each step, each unknown node must use at least one reference point, which is not collinear with other reference points.

**Step 2. Initial estimation of node position**

The initial estimation of node position is to obtain the  $x$  and  $y$  coordinates of each unknown node by ranging. Figure 2 explains how to derive the range of  $x$  coordinate of unknown node  $C$  from the coordinates of two anchor nodes  $A$  and  $B$ . Let  $a$  be the distance between an unknown node and anchor node  $A$ . Then, the  $x$  coordinate of unknown node  $C$  falls between  $x_A - a$  and  $x_A + a$ . Similarly, the shortest path length of the  $x$  coordinate of unknown node  $C$  is  $b + c$ , because anchor node  $B$  is two hops away from unknown node  $C$ . Therefore, the  $x$  coordinate of unknown node  $C$  relative to anchor node  $B$  is bounded by  $x_B - (b + c)$  and  $x_B + (b + c)$ . On this basis, the  $x$  coordinate of unknown node  $C$  relative to anchor nodes  $A$  and  $B$  is bounded by  $x_B + (b + c)$  and  $x_A - a$ , respectively. This operation needs to be performed from the leftmost and

rightmost of each anchor node to obtain the  $x$  coordinate of each unknown node.

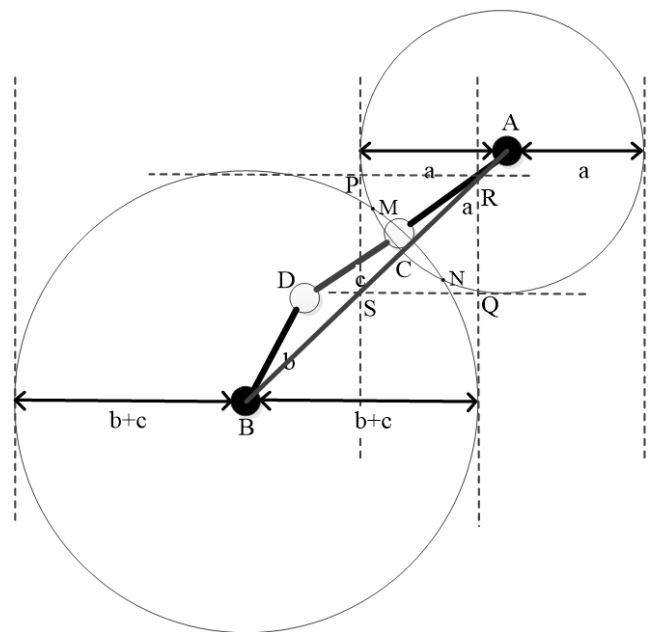
The same operation is performed to obtain the  $y$  coordinate of each unknown node. Then, the bounds of the node on the  $x$  and  $y$  coordinates are merged to obtain the bounding box of the node. To build the bounding box, the positions of all anchor nodes need to be forwarded to every unknown node along the minimum weight paths. The forwarding direction is the same as the routing of distance vector. The only difference is that the weight is measured distance rather than hop count. The initial position of each unknown node is assumed to be in the center of its bounding box. To obtain a set of good initial estimates, the anchor nodes are arranged on the perimeter of the WSN.



**Figure 2.** The sketch map of initial estimation of node position

**Step 3. Optimization of initial node positions**

In the traditional multi-hop collaborative algorithm, the accuracy of node positioning is optimized without initial estimation of node positions. The quality of initial estimations directly bears on the accuracy and efficiency of the positioning results. To avoid positioning divergence, this paper improves computing efficiency and reduces calculation cost by narrowing the range of initial estimation.



**Figure 3.** The narrowing of initial value range of Kalman filter

As shown in Figure 3, the initial value of conventional Kalman filter is obtained by solving the coordinates of the centroid PRQS of the box formed by the distances between anchor nodes and unknown node. To narrow the search range of initial value, two circles centering on the anchor node are drawn with the distances from the unknown node to the two anchor nodes as the radius, respectively. Then, the coordinates of the centroid of the intersection between the two circles, that is, those of the midpoint of the connecting line between M and N, as the initial value of Kalman filter. In this way, the initial value problem of Kalman filter becomes the geometric problem of finding the centroid coordinates of the intersection. The coordinates  $M(x_M, y_M)$  and  $N(x_N, y_N)$  of the intersection points M and N can be solved by:

$$\begin{cases} (x - x_B)^2 + (y - y_B)^2 = (b + c)^2 \\ (x - x_A)^2 + (y - y_A)^2 = a^2 \end{cases} \quad (6)$$

Then, the initial value of Kalman filter can be obtained by:

$$\begin{cases} x_C = \frac{x_M + x_N}{2} \\ y_C = \frac{y_M + y_N}{2} \end{cases} \quad (7)$$

#### Step 4. Solving the accurate node positions

Referring to the multi-hop multilateral collaborative positioning algorithm, the LS estimation is selected to solve the initial position of each unknown node accurately. There are two possible computing models for the accurate positioning: centralized computing or distributed computing.

##### (1) Centralized computing of node position

The positions of unknown nodes are estimated at one central point, using the collaborative subtree and the initial position estimation. A definite set of equations is set up according to the specific structure of the collaborative tree, and then solved by nonlinear optimization. The nonlinear optimization of a multi-hop network containing five-node subtrees can be expressed as:

$$\begin{aligned} f_{2,3} &= R_{2,3} - \sqrt{(x_2 - ex_3)^2 + (y_2 - ey_3)^2} \\ f_{3,5} &= R_{3,5} - \sqrt{(ex_3 - x_5)^2 + (ey_3 - y_5)^2} \\ f_{4,3} &= R_{4,3} - \sqrt{(ex_4 - ex_3)^2 + (ey_4 - ey_3)^2} \\ f_{4,5} &= R_{4,5} - \sqrt{(ex_4 - x_5)^2 + (ey_4 + y_5)^2} \\ f_{4,1} &= R_{4,1} - \sqrt{(ex_4 - x_1)^2 + (ey_4 - y_1)^2} \end{aligned} \quad (8)$$

where,  $R_{i,j}$  is the measured distance between two nodes; the value under the square root is the estimated distance;  $f_{i,j}$  is the residual between the measured and estimated distances. The objective function is the minimize the mean squared error (MSE) of all equations:

$$F(x_3, y_3, x_4, y_4) = \min \sum f_{i,j}^2 \quad (9)$$

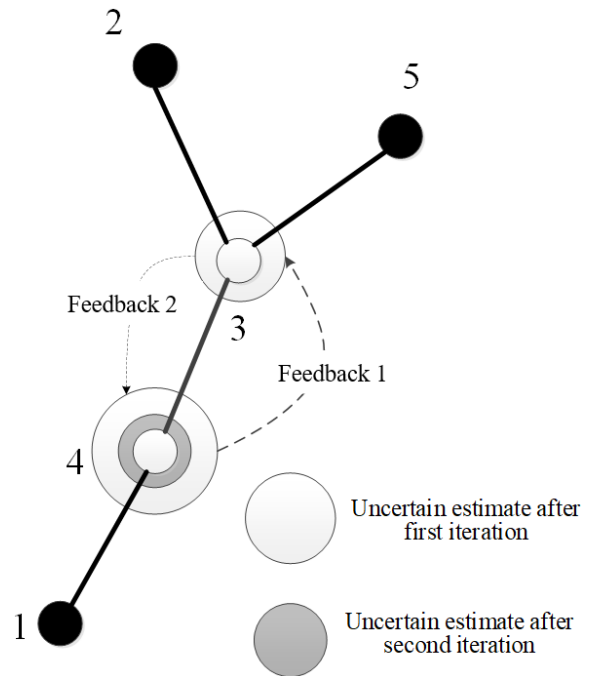
##### (2) Distributed computing of node position

Distributed computing takes place across the entire network. Each unknown node is responsible for estimating its own position through local computing and communication with neighbors. This strategy is basically the same as using a distributed Kalman filter, except for three minor differences:

First, the Kalman filtering is performed on the edge of a collaborative subtree. Each node executes single-hop multiple transforms based on its measured distance and the position information from its neighbors. Second, decentralized Kalman filter is replaced with an approximation that does not exchange covariance information. The calculation cost and energy are saved, because of the reduced communication and simplified computation. Third, calculation is driven by the ad-hoc network protocol.

The distributed computing of node position adheres to the following principles: After the first two steps, each node in the collaborative subtree has estimated its position. Since most unknown nodes are not directly connected to anchors, they take the initial estimates of their neighbors as reference points to estimate their positions. Once an unknown node calculates a new estimate, it will propagate the new estimate to its neighbors, which will update their own position estimates based on that estimate. This process will be repeated from node to node across the network until all nodes reach the preset convergence gradient.

The distributed computing process is illustrated in Figure 4. The first node 4 estimates its position with anchors 1 and 5 and node 3 as reference points. Once node 4 propagates its new estimate, node 3 will re-estimate its own position based on the new estimate, in the light of the distances to anchors 2 and 5. Finally, node 4 will make a better estimation based on the new estimate from node 3.



**Figure 4.** The process of distributed computing

If the above process goes out of control, the position of unknown node will converge to the local minimum, resulting in incorrect estimates. This is mainly attributable to the collaborative subtree with multiple unknown nodes. If two adjacent unknown nodes A and B calculate and propagate their

new estimates right after receiving the new estimate from the other party, the two nodes will update their position estimates faster than the other nodes in the subtree. However, this introduces local oscillation to the computing process. Despite the fast convergence to the final estimate, the convergence is not consistent with the global gradient, making the estimates incorrect.

To prevent the problem, the polygon on each node will be executed in a consistent order among all unknown nodes of the subtree, until the polygons of all unknown nodes on the subtree converge to the preset threshold. The sequential execution of the polygons establishes a gradient on each node relative to the global topological constraint, enabling the nodes to compute the global optimal value locally.

### 3. EXAMPLE ANALYSIS

To verify its effects of the proposed improved algorithm, a simulation environment was constructed on Matlab and NS2 for comparative analysis of positioning accuracy. Specifically, the Dynamic Source Routing protocol (DSR) and IEEE802.11 media access control (MAC) protocol was realized through NS2; the Kalman filter algorithm was implemented on Matlab, and connected with the environment of communication protocols via the complier. Under the simulation environment, the improved algorithm was compared with the multi-hop multilateral collaborative algorithm proposed by Andreas Sawides et al. (the original algorithm) under emergency communication environment in terms of positioning accuracy, calculation cost, and energy consumption.

#### (1) Comparison of positioning accuracy

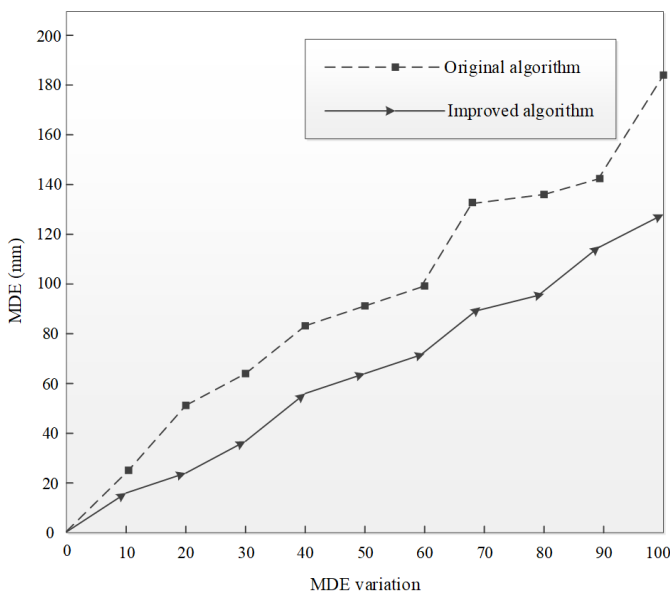


Figure 5. The comparison of positioning accuracy

Figure 5 compares the positioning accuracies of our algorithm and the original algorithm in the above simulation environment. It can be seen that the mean distance errors (MDEs) of both algorithms were on the rise, under the same scenario. But the MDE increment of the improved algorithm was relatively small than that of the original algorithm, indicating that the improved algorithm is more accurate than the original algorithm. The small MDE of the improved algorithm comes from the optimization of the initial values of

Kalman filter. Whereas the original algorithm takes the box centroid as the initial position, the improved algorithm defines the initial position of an unknown node as the centroid of the intersection between two circles with the distances between the node and two anchors as radius, respectively. The approach of the improved algorithm greatly narrows the uncertain range of the unknown node.

#### (2) Comparison of calculation cost

To verify the efficiency of the improved algorithm, the computing efficiency was measured by the number of iterations to obtain the optimal solution. During the simulation, the original and improved algorithms were separately applied to position different numbers of unknown nodes under the same number of anchor nodes. Figure 6 compares the number of iterations of the two algorithms. It can be seen that the number of iterations of both algorithms increased with the number of unknown nodes. Under the same number of unknown nodes, however, the improved algorithm needed much fewer iterations than the original algorithm. This means, under the same conditions, the improved algorithm outshines the original algorithm in computing efficiency, that is, it can complete node positioning tasks with relatively few computing resources.

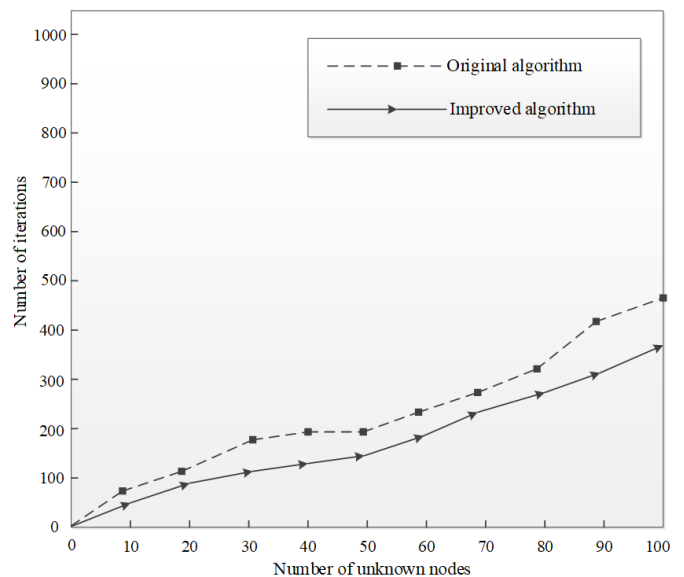


Figure 6. The comparison of calculation cost

#### (3) Comparison of energy consumption

During emergency communication, WSN nodes need to consume lots of energy. To extend the service life of the nodes, the energy consumption in calculation must be minimized. Figure 7 compares the energy consumptions of the two algorithms under the same simulation environment. It can be seen that the two algorithms differed slightly in communication cost at the same node. Overall, the improved algorithm consumed fewer energy than the original algorithm, because it reduces the number of iterations and information exchanges between nodes by reducing the range of initial values of Kalman filter. Hence, the improved algorithm is more in line with the energy-saving requirements on the WSN in emergency communication than the original algorithm.

The above comparisons fully demonstrate the effectiveness and practicality of the proposed algorithm. The superiority of our algorithm is mainly ascribed to the narrowing of the initial value range of Kalman filter, and the distributed Kalman filter. The former enables the specially deployed WSN nodes to



estimate their positions based on the known positions of anchor nodes, while the latter iteratively solves the node positions in an accurate manner. As a result, the improved algorithm can position the WSN nodes accurately in a simple process at a low calculation cost and energy cost, and adapt well to the emergency communication environment.

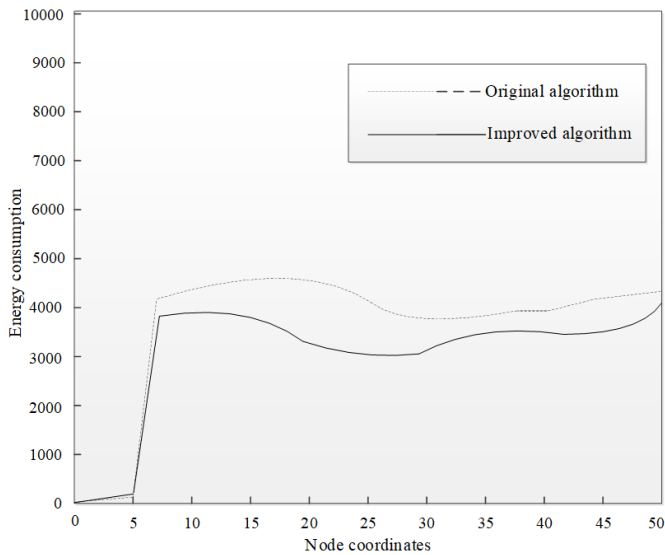


Figure 7. The comparison of energy consumption

#### 4. CONCLUSIONS

This paper mainly explores the WSN node positioning under the emergency communication environment. Considering the defects of multi-hop multilateral node positioning algorithm, the authors improved the positioning accuracy by narrowing the initial value range of Kalman filter. To demonstrate its effectiveness and practicality, the improved algorithm was compared with the original algorithm through simulation. The results show that the improved algorithm has the edge in positioning accuracy, calculation cost, and energy consumption, and greatly improves the applicability of WSN in emergency communication environment. However, the proposed algorithm does not consider the optimization of the termination condition. The future research will try to further optimize the algorithm from this angle.

#### REFERENCES

[1] Qiu, Y., Zhao, C.C., Dai, G.L., Hu, C.J. (2008). Research on localization technology for wireless sensor networks. *Computer Science*, 35(5): 47-50.

[2] Li, S., Da Xu, L., Wang, X. (2012). Compressed sensing signal and data acquisition in wireless sensor networks and internet of things. *IEEE Transactions on Industrial Informatics*, 9(4): 2177-2186. <https://doi.org/10.1109/TII.2012.2189222>

[3] Sorber, L., Van Barel, M., De Lathauwer, L. (2015). Structured data fusion. *IEEE Journal of Selected Topics in Signal Processing*, 9(4): 586-600. <https://doi.org/10.1109/JSTSP.2015.2400415>

[4] Liu, Y., Zeng, Q.A., Wang, Y.H. (2014). Data fusion in wireless sensor networks. *International Journal of Distributed Sensor Networks*, 2014: 131870.

<http://dx.doi.org/10.1155/2014/131870>

[5] Chair, Z., Varshney, P.K. (1986). Optimal data fusion in multiple sensor detection systems. *IEEE Transactions on Aerospace and Electronic Systems*, AES-22(1): 98-101. <https://doi.org/10.1109/TAES.1986.310699>

[6] Pascale, A., Nicoli, M., Deflorio, F., Dalla Chiara, B., Spagnolini, U. (2012). Wireless sensor networks for traffic management and road safety. *IET Intelligent Transport Systems*, 6(1): 67-77. <https://doi.org/10.1049/iet-its.2010.0129>

[7] Chen, C.M., Lin, Y.H., Lin, Y.C., Sun, H.M. (2011). RCDA: Recoverable concealed data aggregation for data integrity in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 23(4): 727-734. <https://doi.org/10.1109/TPDS.2011.219>

[8] Xie, J.R., Hu, Y.M., Liu, C.X., Liu, L. (2007). Fundamental technique for wireless sensor networks Time synchronization. *Computer Engineering and Design*, 28(1): 76-86.

[9] Pérez-Solano, J.J., Felici-Castell, S. (2015). Adaptive time window linear regression algorithm for accurate time synchronization in wireless sensor networks. *Ad Hoc Networks*, 24: 92-108. <https://doi.org/10.1016/j.adhoc.2014.08.002>

[10] Hightower, J., Borriello, G. (2001). Location systems for ubiquitous computing. *Computer*, 34(8): 57-66. <https://doi.org/10.1109/2.940014>

[11] Howard, A., Mataric, M.J., Sukhatme, G. (2001). Relaxation on a mesh: A formalism for generalized localization. *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the Next Millennium (Cat. No.01CH37180)*, Maui, HI, USA, pp. 1055-1060. <https://doi.org/10.1109/IROS.2001.976308>

[12] Roumeliotis, S.I., Bekey, G.A. (2000). Synergetic localization for groups of mobile robots. *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No.00CH37187)*, Sydney, NSW, pp. 3477-3482. <https://doi.org/10.1109/CDC.2000.912242>

[13] Musicki, D., Kaune, R., Koch, W. (2009). Mobile emitter geolocation and tracking using TDOA and FDOA measurements. *IEEE Transactions on Signal Processing*, 58(3): 1863-1874. <https://doi.org/10.1109/TSP.2009.2037075>

[14] Ho, K.C. (2012). Bias reduction for an explicit solution of source localization using TDOA. *IEEE Transactions on Signal Processing*, 60(5): 2101-2114. <https://doi.org/10.1109/TSP.2012.2187283>

[15] Yin, J., Wan, Q., Yang, S., Ho, K.C. (2015). A simple and accurate TDOA-AOA localization method using two stations. *IEEE Signal Processing Letters*, 23(1): 144-148. <https://doi.org/10.1109/LSP.2015.2505138>

[16] Niculescu, D., Nath, B. (2001). Ad hoc positioning system (APS). In *GLOBECOM'01. GLOBECOM'01. IEEE Global Telecommunications Conference (Cat. No.01CH37270)*, San Antonio, TX, pp. 2926-2931. <https://doi.org/10.1109/GLOCOM.2001.965964>

[17] Rabaey, C.S.J., Langendoen, K. (2002). Robust positioning algorithms for distributed ad-hoc wireless sensor networks. *USENIX Technical Annual Conference* pp. 317-327.

[18] Gonzalez, H., Han, J., Li, X. (2006). Mining compressed commodity workflows from massive RFID data sets. *Conference on Information and Knowledge Management*

- Arlington Virginia USA, pp. 162-171.  
<https://doi.org/10.1145/1183614.1183641>
- [19] Chan, Y.T., Ho, K.C. (1994). A simple and efficient estimator for hyperbolic location. *IEEE Transactions on Signal Processing*, 42(8): 1905-1915.  
<https://doi.org/10.1109/78.301830>
- [20] Foy, W.H. (1976). Position-location solutions by Taylor-series estimation. *IEEE Transactions on Aerospace and Electronic Systems*, AES-12(2): 187-194.  
<https://doi.org/10.1109/TAES.1976.308294>
- [21] Savvides, A., Han, C.C., Srivastava, M.B. (2001). Dynamic fine-grained localization in Ad-Hoc networks of sensors. *International Conference on Mobile Computing and Networking Rome Italy*, pp. 166-179.  
<https://doi.org/10.1145/381677.381693>
- [22] Savvides, A., Park, H., Srivastava, M.B. (2002). The bits and flops of the n-hop multilateration primitive for node localization problems. *First ACM International Workshop on Wireless Sensor Networks and Applications, Atlanta Georgia USA*, pp. 112-121.  
<https://doi.org/10.1145/570738.570755>