

## Research and Simulation of Improved APIT Localization Algorithm Based On WSN

Pinglan Cai<sup>1,a</sup>, Layuan Li<sup>1,2,b</sup>, Chunlin Li<sup>1,c</sup>

<sup>1</sup>School of Computer Science and Technology, Wuhan University of Technology, Wuhan, China

<sup>2</sup>Wuhan Bioengineering Institute, Wuhan, China

<sup>a</sup>fengjiya@gmail.com, <sup>b</sup>lilayuan26@163.com, <sup>c</sup>chunlin74@aliyun.com

**Keywords:** wireless sensor network; node localization algorithm; APIT algorithm, SBL algorithm

**Abstract.** In wireless sensor networks(WSN), APIT algorithm is a good representation of localization algorithm and has been used widely. This work improves the classic APIT algorithm and does some researches on the accuracy and coverage of the improved APIT algorithm. The modified algorithm processes begin to change in the second step and have a addition in the end: At first, these nodes that were unable to locate are labeled in the second step. After all the nodes that can be test with APIT algorithm are located, these marked nodes will use SBL algorithm with these located nodes and anchor nodes within its sense range to estimate their positions. At last, the estimation coordinates are regarded as the location of the nodes. These experiment data shows that the nodes' coverage and enhances the overall localization accuracy on the whole.

### Introduction

As one of the five basic technologies, the nodes' localization algorithms of Wireless sensor networks have been paid more attention in recent years. Though APIT algorithm has a good presentation in localization algorithm, it still has some shortcomings. This article focuses on the management of nodes that don't meet the APIT test, it puts forward the SBL algorithm to handle these nodes by taking advantages of the located nodes and anchor nodes, approving that the improved algorithm could make a contribution to enhance the nodes' coverage and localization accuracy on the whole.

### APIT Algorithm

The APIT algorithm was presented by He and Huang[1] after studying the PIT(Perfect Point-In-Triangulation) test. The algorithm is made of three stages: After initializing the network, anchor nodes begin to broadcast their information, including the coordinate information and single strength. By exchanging messages between neighbor nodes, they check whether there exists some triangles that can cover it. After calculating all the triangles' overlapping area, we regard the centroid of the sharing area as the node's coordinate.

**Errors analysis:** APIT test not only depends on the neighbor nodes' density and their distribution, but also has a relationship with node's sensor range and transition radius. So two kinds of errors happen frequently in actual test: some nodes are located inside the triangle, and some of it to be outside of it. As shown in Fig.1(a), M is inside  $\triangle ABC$  actually, but when compared with node 1, node 2, node 3, the strength of signal are weaker, so it located outside the triangle. That is the InToOut Error. In Figure .1(b), we can find that M should be outside, but when compared with node 1 and node 2, the signal strength are weaker, we regard it inside the triangle. That is the OutToIn Error.

For the reasons why the two errors occur, it because the basis of APIT test has a defect. Assuming that node M is inside  $\triangle ABC$ , but when M is moved far enough, we can find that M's position is far away from A, B and C. Thus M is considered outside  $\triangle ABC$ , and this an obvious fault.

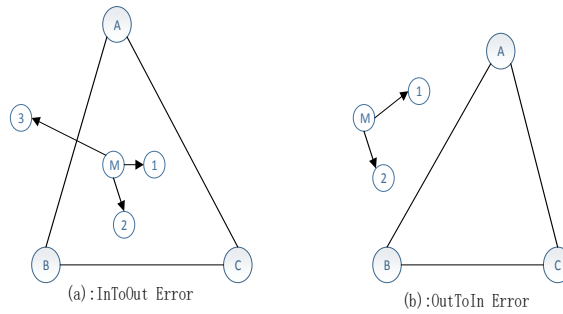


Fig. 1 InToOut Error and OutToIn Error

### Improved APIT Algorithm

The algorithm presented in this work has a complete contrast in the second stage. These nodes that can't meet APIT test are marked when calculating their positions. To deal with the situation that there doesn't exist enough anchor nodes in its sensor range, SBL (Sequence based localization) algorithm has been raised up to dispose it.

**SBL Algorithm** : Sequence based algorithm (SBL) [2] is a fresh location algorithm with higher location accuracy by using a complete new math method. Here is the location theory: first of all, the unknown node counts the anchors number and link any two of them by line. After slicing the reference area by the bisector lines of two reference nodes, the unknown node needs to be computed and find the maximum relativity strength in the sequence list and use it to estimate its coordination. As shown in the research, the unknown node is often located at the edge. So it could generate a mass of subdomains with big area. And within this condition, many location errors occur in locating process.

Figure 2 shows the procedure of SBL algorithm. We assume that there exist many reference nodes in a particular area and cut it with the bisector lines into pieces. These subdomains could be vertex, boundary lines and geometric planes. We can build a sequence list by measuring its distance from the reference nodes. It can be easily calculated that a vertex's centroid is itself, a boundary line's centroid lies in the Middle of it, and the centroid of a geometric plane is its geometric center.

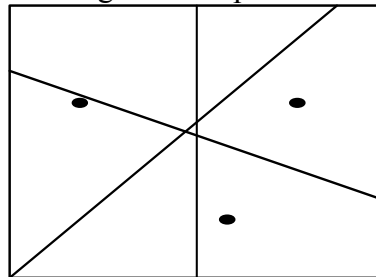


Fig. 2 Slice the area into pieces

**Combination of Two Algorithms** : In the fifth step of APIT algorithm, we take these nodes that don't meet the test conditions into other management. After all the normal nodes that can meet the test condition are located, their coordinate information are used to deal with these unknown nodes.

After computing the subdomains' sequences, we use the following stages to locate the unknown nodes:

- 1) computer all the sequences of feasible areas and store them in the sequence list;
- 2) the unknown node confirm its sequence by comparing the distances from these reference nodes;
- 3) unknown node search the sequence list and find the appropriate relativity strength;
- 4) mapping to its geometric plane and compute its centroid position, using it as the estimate coordinate.

The key point lies in the first step to compute each subdomain's sequence, the pseudo-code has been listed as the following procedure:

1 Input nodes' coordinator:  $\{(x_i, y_i) | i = 0 \leftarrow n - 1\}$

2 calculate the bisector lines of reference nodes:

$$L = \{l_i | i = 0 \leftarrow (\frac{n(n-1)}{n} - 1)\} \text{ // BisectorLines}$$

```

3 {Splane, Sline, Svertex} ← Calculate(L)
4 //calculate the vertex of bisector lines
5 for i ← 0 to (Svertex - 1)
6   Centroid[i] ← Svertex[i]
7   Sequence[i] ← GetSequence(Centroid[i])
8 end for
9 // calculate the boundary lines
10 for i ← Svertex to (Svertex + Sline - 1)
11   Centroid[i] ← GetLineCentroid(Svertex[i])
12   Sequence[i] ← GetSequence(Centroid[i])
13 end for
14 // calculate the geometric planes
15 for i ← (Svertex + Sline) to (Svertex + Sline + Splane - 1)
16   Centroid[i] ← GetPlaneCentroid(Splane[i])
17   Sequence[i] ← GetSequence(Centroid[i])
18 end for
19 return {Sequence, Centroid}
20 Output the Sequences List.
```

We use Kendall’Tau[3] to measure each relativity between the sequences in the list. Assuming there exists two orders  $\alpha = \{\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n\}$ ,  $\beta = \{\beta_1, \beta_2, \beta_3, \dots, \beta_n\}$ ,  $S_1$  is the dual subset collection of  $\alpha$ ,  $S_2$  is the dual subset collection of  $\beta$ . We calculate  $S_1$  and  $S_2$  by using the following formulas:

$$S_1 = \{(\alpha_1, \alpha_2), (\alpha_1, \alpha_3), (\alpha_1, \alpha_4), (\alpha_1, \alpha_5) \dots (\alpha_{n-1}, \alpha_n)\} \quad (1)$$

$$S_2 = \{(\beta_1, \beta_2), (\beta_1, \beta_3), (\beta_1, \beta_4), (\beta_1, \beta_5) \dots (\beta_{n-1}, \beta_n)\} \quad (2)$$

We use  $\tau$  to measure the Tau index, and it has been computed as follows[4]:

$$\tau = 1 - 2 * \frac{\text{reversions}}{n * (n - 1)} \quad (3)$$

*reversions* represents the number of contrary order in  $S_1$  and  $S_2$ ,  $n$  is the length of the sequence list. In actual computing, the relativity between sequences must be stronger than middle strength. That means it must be bigger than 0.3 when  $\tau > 0$  and not smaller than -0.3 when  $\tau < 0$ . We call these sequences with middle strength or bigger as the effective sequences, and others are useless sequences.

The possibly values of  $\tau$  are listed in Table 1 as blow[5]:

Table 1 Tau’s relativity value

Relativity	<0	>0
Strong strength	[-1.0 , -0.5]	[0.5 , 1.0]
Middle strength	[-0.5 , -0.3]	[0.3 , 0.5]
Weak strength	[-0.3 , -0.1]	[0.3 , 0.1]
No relativity	[-0.09 , -0.0]	[0.0 , 0.09]

If there are  $n$  reference nodes in the location area,  $O(n^n)$  sequences can be generated and that’s a very large number. But experiment data shows that the maximum number of feasible sequences is

$O(n^4)$  under the mathematics constraint. So word can choose and determine the useful sequences after filtering the list.

**Analysis of Improved Algorithm:** The improved algorithm focuses on making a change in dealing these nodes that can't be located. When the located neighbor nodes are added into its anchor nodes set, the occupancy of triangle made of newly added nodes  $\gamma$  can be calculated as follows:

$$\gamma = \frac{t_i}{\sum_{j=1}^n t_j} \quad (4)$$

In the formula,  $i$  is the number of these newly added normal nodes,  $t_i$  the number of triangles made of these normal nodes and the existed anchor nodes,  $n$  is the number of added nodes and anchor nodes, we call these nodes as a node set  $H$ ,  $\sum_{j=1}^n t_j$  is the number of triangles made of any three nodes chosen from set  $H$ .

It can be approved that the number of feasible sequences can't reach the magnitude of  $O(n^4)$ . The smaller  $\gamma$  is, the lesser feasible sequences in the sequences list. On one hand it magnifies the ratio of located nodes, on the other hand it increases the communications between nodes, thus increasing the energy consumption at the same time because of extra computation of unlocated nodes. To make full use of neighbor nodes, the improved algorithm makes a request to nodes' distribution. If the node density is big enough, lots of sequence list contributes to quantity of computations. On the other hand if the nodes distributes unregularly or at a special position without enough anchor nodes, they can't be located without enough coordinate information.

**Innovation Point** The improved algorithm's innovation lies in taking full advantage of these normal neighbor nodes' positions. These coordinates information will be used as the basis to calculate the positions of nodes that don't meet the conditions of APIT test, and then it uses the SBL algorithm to reduce the interferences of the invalid nodes' information. Compared with these similar methods to process the edge nodes, its diffusion range is in a better control as well as the coordinate error, thus reducing the impact on the positioning accuracy error on the whole.

## Simulation

This work makes experiments on NS2 platform. These nodes are distributed in the area with the square 1000m\*1000m. All the nodes are generated randomly. We set the number of nodes as S, the anchor ratio to K, anchor nodes has a launcher range to 200m, and normal nodes has a sensor rang to 300m. It takes no consideration of GPS error in anchor nodes and other variates are the same in all experiments.

We can obtain the result from the chart of Fig 3 that the two algorithms are constricted to 600 nodes, and the improved algorithm doesn't have an obvious increase compared with the classic one. Increasing nodes density can enlarge the neighbor nodes' number, but the probability of InToOut Error and OutToIn Error keeps rising up, and the ratio of sequence math the condition  $\tau > 0.3$  doesn't increase obviously.

It can be seen from Fig 4 that when the anchor node K rises up to 60%, the improved algorithm has a large progress in node's coverage at about 9.3%. If  $K$  is big enough to make majority of the nodes to be anchor nodes, almost all the nodes have enough neighbor nodes in APIT test and could make a big sequence list, thus reducing the probability of InToOut Error and OutToIn Error and the localization can be accurate enough.

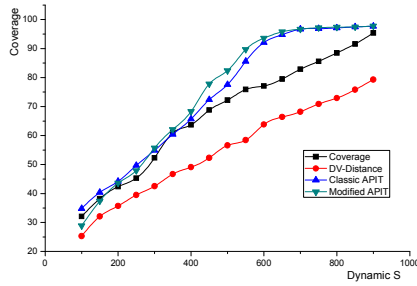


Fig 3. Dynamic node density to nodes' Coverage Coverage

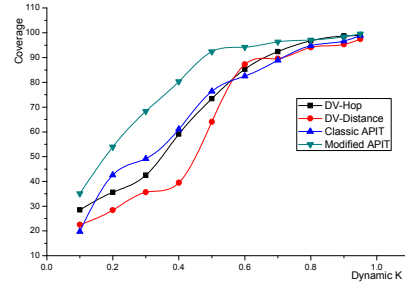


Fig 4. Dynamic anchor ratio to nodes' Coverage

Fig 5 shows that nodes' location accuracy rises up to about 4.8%. The bigger node density makes lesser nodes that can't meet the APIT conditions. More reference nodes reduce the wrong judgements in the test.

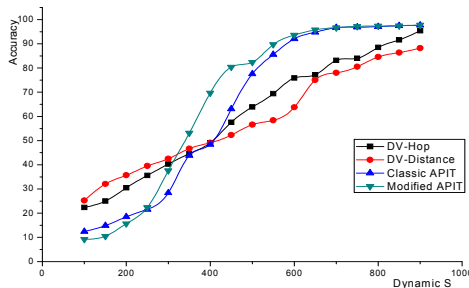


Fig 5. Dynamic node density to location Accuracy Accuracy

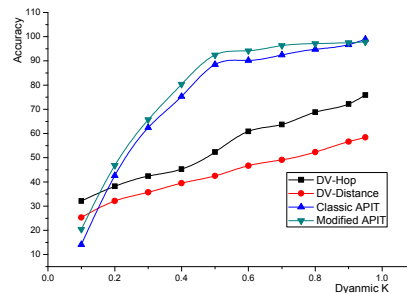


Fig 6. Dynamic anchor ratio to location Accuracy

As its shown in Fig 6, nodes coverage increases about 2.3% and location accuracy rises up about 3.9%. High anchor ratio makes more triangle could contribute to the sharing area in grid scanning, but the number of feasible areas doesn't have a big difference.

Fig 7 and Fig 8 show that the improved algorithm have a progress in nodes coverage at about 3.8% and an obviously increase in location accuracy at about 5.2%. With a bigger sensor range, each node's neighbor nodes rise up in a big step. The number of subdomain in SBL increase a lot and enhance the relativity strength of sequence list. More feasible subdomains are used and lesser nodes that can't be located to be left.

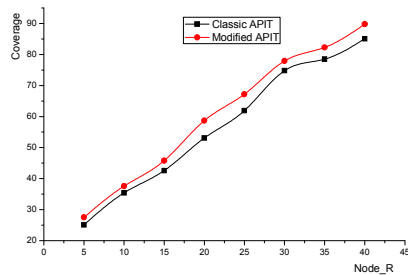


Fig 7 Dynamic sensor range to nodes' coverage Accuracy

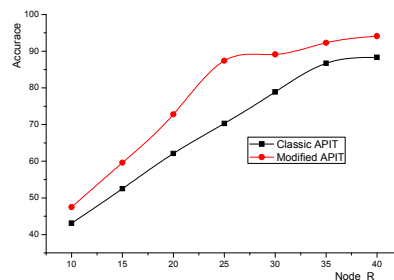


Fig 8. Dynamic sensor range to location Accuracy

## Conclusion

This work focuses on dealing with these nodes that can't be located by APIT algorithm, it's the combination of APIT and SBL algorithm: Use SBL algorithm to calculate the sequence list and locate these nodes. Through experiments built on NS2 platform, it approves that the improved algorithm can make a progress in nodes' coverage and location accuracy, making it a better location algorithm that classic APIT algorithm.

**Acknowledgements:**

This thesis is supported by National Natural Science Foundation of China(No: 61171075,61171094),Specialized Research Fund for the Doctoral Program of Higher Education of China (No: 200604 97105), Natural Science Foundation of HuBei(No: 2014CFA 050),Wuhan Foundation Research and Application(No: 2015010101010021).

**References:**

- [1] Tian He, Huang Chengdu, John A S, et. Range-Free Localization Schemes for Large Scale Sensor Networks[C]. Proceedings of the 9th Annual International Conference on Mobile Computing and Networking, MobiCom, 2003,: 81-95.
- [2] Bhaskar Krishnamachari, Kiran Yedavalli. Secure Sequence-Based Localization for Wireless Sensor Networks[J]. Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks Advances in Information Security, 2007(30):237—247.
- [3] Kendall, M. A New Measure of Rank Correlation[J]. Biometrika, 1938,30(1):81-89
- [4]I Kasmi, G Kasmi, S Sallabanda. ect. Incidence of nosocomial respiratory tract infection in pediatric intensive care unit of University Hospital Center (UHC) of Tirana [J]. International Journal of Infectious Diseases,2010,14:254-259.
- [5] DeLong E R, DeLong D M, Clarke-Pearson D L. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach[J]. Biometrics, 1988: 837-845.