Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2016, Article ID 1892893, 9 pages
http://dx.doi.org/10.1155/2016/1892893

Research Article

A Least Square-Based Self-Adaptive Localization Method for Wireless Sensor Networks

Baoguo Yu,[1,2] Yao Wang,[1,2] Chenglong He,[1,2] Xiaozhen Yan,[1,3] and Qinghua Luo[1,3]

[1] *State Key Laboratory of Satellite Navigation System and Equipment Technology, Shijiazhuang, China*
[2] *The 54th Research Institute of China Electronic Technology Corporation, Shijiazhuang, China*
[3] *School of Information and Electrical Engineering, Harbin Institute of Technology at Weihai, Weihai, China*

Correspondence should be addressed to Yao Wang; wy_hrbeu@163.com

Received 9 March 2016; Revised 14 June 2016; Accepted 25 October 2016

Academic Editor: Tamas Kalmar-Nagy

In the wireless sensor network (WSN) localization methods based on Received Signal Strength Indicator (RSSI), it is usually required to determine the parameters of the radio signal propagation model before estimating the distance between the anchor node and an unknown node with reference to their communication RSSI value. And finally we use a localization algorithm to estimate the location of the unknown node. However, this localization method, though high in localization accuracy, has weaknesses such as complex working procedure and poor system versatility. Concerning these defects, a self-adaptive WSN localization method based on least square is proposed, which uses the least square criterion to estimate the parameters of radio signal propagation model, which positively reduces the computation amount in the estimation process. The experimental results show that the proposed self-adaptive localization method outputs a high processing efficiency while satisfying the high localization accuracy requirement. Conclusively, the proposed method is of definite practical value.

1. Introduction

Generally, two steps are needed for the wireless sensor networks (WSN) localization algorithm to estimate the location of an unknown node based on the Received Signal Strength Indicator (RSSI) [1]. Step 1 is to try to determine the propagation parameters of the radio signal communication model with a fitting technique by measuring the mapping relation between RSSI and the distance d. Step 2 is to estimate the distance between the unknown node and each anchor node with reference to the communication RSSI value between them and obtain the estimated position value of the unknown node with reference to the coordinates of those anchor nodes. Therefore, this localization method requires a preliminary test for the environment [2] so as to determine the propagation parameters of the model. The lighted localization method Bounding-box (B-box for simplicity) [3] and its improved version [4–7], such as weighted B-box, 3-point centroid B-box, and 3-point weighted 3-point centroid B-box, were proposed to conduct localization of the unknown node.

However, the preliminary environmental test is a very complicated process that requires large amounts of experimental works. Besides, the preliminary test must be taken in a fixed localization environment. In case of any changes in the communication environment, the parameters in the radio signal propagation model would change along location information which plays an important role in location-based service application system, leading to error increment to the distance estimated upon RSSI value or the original model becoming not applicable any more. The factors affecting the environment usually include temperature, humidity, interference, and non-line-of-sight (NLOS) [8–10], which are subject to change along with the environment and time. Thus, the localization system that can keep high accuracy in a dynamic environment will be more promising. To satisfy this requirement, a maximum likelihood-based self-adaptive localization algorithm is proposed, which does not require a preliminary test for the environment in a dynamic environment but needs considerable computation amount. A distributed self-adaptive localization algorithm is proposed in the reference document [4], which, however, only investigates the localization issue when the communications range is fixed and poor in versatility. Hu and Evans proposed a Monte Carlo localization (MCL) method for mobile sensor node, and its computational time complexity is $O(Mkm)$, where M is the number of Monte Carlo sample points, m is the number of localization points, and k is the number of anchor nodes [11]. Min et al. proposed an improved version of MCL, which is called Monte Carlo localization algorithm based on anchor node selection (MCLAS) [12], and the computational time complexity is also $O(Mkm)$. To locate the mobile node, Shan et al. proposed self-adaptive localization algorithm based on Monte Carlo and gray prediction model (GPLA); however, the complexity is high [13]. For the pose-tracking problem in a dynamic and highly occluded environment, literature [14] proposed a self-adaptive tracking algorithm for mobile robots. And maximum likelihood-based self-adaptive localization algorithm is proposed for dynamic localization, of which complexity is $O(knm)$, where k is the number of anchor nodes, n is the number of iterations, and m is the number of localization points.

In consideration of this, an analysis was given to the working process of self-adaptive localization algorithm. It was found that the propagation parameters in the radio signal propagation model became linear after the model was taken from the logarithm, and the computational amount could be reduced significantly using the least square method. Hence, this paper proposes the least square-based self-adaptive WSN localization method.

2. A Least Square-Based Self-Adaptive Localization Method

In this section, the system flow of the least square-based self-adaptive localization method is presented before every part is elaborated.

2.1. System Flow of Least Square-Based Self-Adaptive Localization Method. In the proposed self-adaptive localization algorithm, initialization is given firstly, including parameter initialization and representation of RSSI in probability density, before self-adaptive localization is performed iteratively. It is divided into two steps. The first step is to estimate the location of the unknown node and the second step is to estimate the parameter in the propagation model. Graphically, the workflow of the proposed least square-based self-adaptive localization method is illustrated in Figure 1.

The parameter initialization in Figure 1 mainly concerns the parameters of radio signal propagation models α and β. Their values are arbitrary, but the convergence rate of the self-adaptive localization algorithm would be affected if their initial values are inappropriate. After the coordinates of the unknown node are estimated with the maximum likelihood estimation method, the parameters in the radio signal propagation model, $\widehat{\alpha}_n$ and $\widehat{\beta}_n$, are estimated using the least square method. And then, these two parameters are evaluated if they converge to thresholds a and b, respectively. If yes, the iteration will stop and output the estimation result;

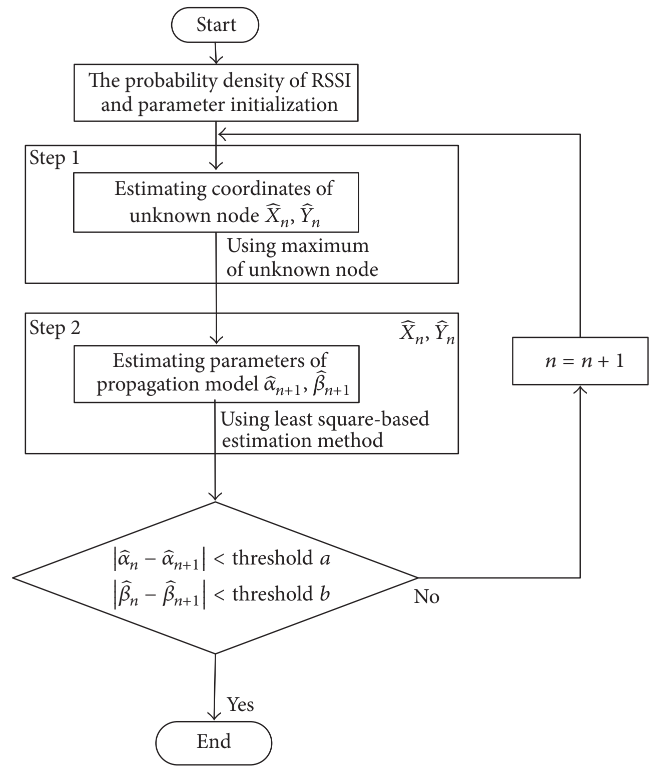


Figure 1: Workflow of least square-based self-adaptive localization method.

otherwise, the iteration will go on until the condition of convergence is satisfied.

2.2. Representation of RSSI in Probability Density. In the traditional RSSI-based localization method, a path loss model is established for mapping between RSSI and distance d, as expressed by the following equation [2–4, 9]:

$$\mathrm{RSSI}(d) = \mathrm{RSSI}(d_0) - 10 \times f \times \log(d), \tag{1}$$

where RSSI(d) represents the signal strength at the anchor node received from the localization node; RSSI(d_0) represents the signal strength at the anchor node received from the reference node; d_0 represents the distance between the reference node and the anchor node; d represents the distance between the unknown node and the anchor node; and f represents the channel attenuation index; its typical value is 2~4.

Make a transformation to (1) in the following way, that is, to convert RSSI(d) from dBm units into miliwat units. By dividing both ends of (1) with 10 and performing exponent arithmetic to both sides with base 10, we obtain

$$10^{\mathrm{RSSI}(d)/10} = 10^{\mathrm{RSSI}(d_0)/10} \cdot d^{-f}. \tag{2}$$

Now, the mean received signal strength $\overline{P}$ at the distance d may be expressed as a radio signal propagation model as follows [2]:

$$\overline{P}(d) = \alpha \cdot d^{-\beta}\,(\mathrm{mW}), \tag{3}$$

where $\overline{P}(d) = 10^{\text{RSSI}(d)/10}$, $\alpha = 10^{\text{RSSI}(d_0)/10}$, and $\beta = f$; α and β are parameters of the radio signal propagation model; specifically, α represents a constant in proportion to the received signal strength in a certain distance (typically 1 meter) and β represents the attenuation factor of path loss.

If the communication distance d_j and the parameters of the radio signal propagation model α and β are assigned, the following conditional probability density function can be established on the basis of Rayleigh distribution [2]:

$$p\left(P_j^k \mid d_j, \alpha, \beta\right) = \frac{1}{\alpha \cdot d_j^{-\beta}} \exp\left(-\frac{P_j^k}{\alpha \cdot d_j^{-\beta}}\right), \quad (4)$$

where $k = 1, \ldots, N$, $j = 1, \ldots, M$, and k represents the serial number of received signal strength; j represents the serial number of anchor node; N represents the receiving times for the overall signal strength, and M represents the total number of anchor nodes. The conditional probability density function $p(P_j^k \mid d_j, \alpha, \beta)$ represents the probability under the condition of distance d_j, radio propagation parameters α and β, and received signal strength P_j^k, where d_j represents the distance between the unknown node and the jth anchor node at the coordinates of $[x_j, y_j, z_j]$ in addition to $\overrightarrow{P} = [P_1^1, \ldots, P_M^N]$ and $\overrightarrow{d} = [d_1, \ldots, d_M]$.

And then, it is required to work out the joint probability corresponding to the total signal strength at the unknown node receiving from M anchor nodes. Let us assume that P_j^k are independent from each other; then the resulting joint conditional probability density function can be expressed as follows [2]:

$$p\left(\overrightarrow{P} \mid \overrightarrow{d}, \alpha, \beta\right) = \prod_{j=1}^{M} \prod_{k=1}^{N} p\left(P_j^k \mid d_j, \alpha, \beta\right). \quad (5)$$

If every unknown node is estimated separately, then different nodes might give different α and β values. This practice is considered unreasonable, for all these nodes are in the same environment. However, the real environment is dynamic at any time rather than the above-mentioned case, and every node on the localization platform has its distinct hardware features. In other words, every node is different from any of the others, which require separate estimation to the location information of every unknown node. Therefore, this paper will try to provide a separate solution to every unknown node.

If the joint probability density is expressed with likelihood function $\ln l(\overrightarrow{d}, \alpha, \beta)$, then the log likelihood function can be expressed as $L(\overrightarrow{d}, \alpha, \beta) = \ln l(\overrightarrow{d}, \alpha, \beta) = \ln p(\overrightarrow{P} \mid \overrightarrow{d}, \alpha, \beta)$. The maximum value of the log likelihood function may be obtained by means of alternating solution in two steps. Each step is a separate solution process for the maximum value.

2.3. Location Estimation of an Unknown Node. When the log likelihood function $L_1(\overrightarrow{d}_n, \alpha_{n-1}, \beta_{n-1})$ is solved for the maximum value, it is possible to work out the values of X and Y. More specifically, we calculate the partial derivatives of X and Y separately and reset them to zero; we will have the values of X and Y, where n represents the iteration times. This probability density function does not assume any value to parameters α and β; thus, the initial values for α_0 and β_0 in the first iteration are arbitrary [3].

$$\begin{aligned}
\left[\frac{\partial L_1\left(\overrightarrow{d}_n, \alpha_{n-1}, \beta_{n-1}\right)}{\partial X}\right]_{X=\widehat{X}} &= \left[\frac{\partial}{\partial X} \cdot \sum_{j}^{M} N \left\{\ln\left(\frac{1}{\alpha_{n-1} \cdot d_{jn}^{-\beta_{n-1}}}\right) - \frac{\sum_k^N P_j^k / N}{\alpha_{n-1} \cdot d_{jn}^{-\beta_{n-1}}}\right\}\right]_{X=\widehat{X}} = 0, \\
\left[\frac{\partial L_1\left(\overrightarrow{d}_n, \alpha_{n-1}, \beta_{n-1}\right)}{\partial Y}\right]_{Y=\widehat{Y}} &= \left[\frac{\partial}{\partial Y} \cdot \sum_{j}^{M} N \left\{\ln\left(\frac{1}{\alpha_{n-1} \cdot d_{jn}^{-\beta_{n-1}}}\right) - \frac{\sum_k^N P_j^k / N}{\alpha_{n-1} \cdot d_{jn}^{-\beta_{n-1}}}\right\}\right]_{Y=\widehat{Y}} = 0.
\end{aligned} \quad (6)$$

We work out (6) separately to get the estimated horizontal and longitudinal coordinates of the node: $\widehat{X}$ and $\widehat{Y}$.

2.4. Least Square-Based Estimation for Parameters of Radio Propagation Model. To obtain the values of P_0 and k in a log normal model with least square method, it is equivalent to minimizing the sum of square [8]; that is,

$$\min_{P_0, k} \left\{\sum_{j=1}^{N} \left(\eta_{ij}\right)^2\right\}, \quad (7)$$

where error η_{ij} is defined as follows:

$$\eta_{ij} = P_0 - k \cdot 10 \cdot \log\left(\widehat{d}_{jn}\right) - \overline{P}_{\text{dBm}}\left(d_{jn}\right), \quad (8)$$

where the relation between P_0 and α as well as that between $\overline{P}_{\text{dBm}j}$ and P_{jn}^k can be expressed as

$$\begin{aligned}
P_0 &= 10 \cdot \log(\alpha), \\
\overline{P}_{\text{dBm}j} &= 10 \cdot \log\left(\frac{1}{M} \sum_{k=1}^{M} P_{jn}^k\right).
\end{aligned} \quad (9)$$

Thereby, a linear relation is obtained between P_0 and k.

$$\text{If } S = \begin{bmatrix} 1 & -10 \cdot \log(d_{1n}) \\ 1 & -10 \cdot \log(d_{2n}) \\ \vdots & \vdots \\ 1 & -10 \cdot \log(d_{Nn}) \end{bmatrix},$$
$$w = [P_0, k]^T,$$
$$u = \left[\overline{P}_{\mathrm{dBm1}}, \overline{P}_{\mathrm{dBm2}}, \ldots, \overline{P}_{\mathrm{dBm3}}\right]^T \quad (10)$$
$$\text{then } u = Sw.$$

With least square optimization criterion, we can obtain the optimal solution shown as (11), which minimized the value of $|Sw - u|$.

$$w = \left(S^T S\right)^{-1} S^T u. \quad (11)$$

The calculated values of P_0 and k will be used for next iteration, where P_0 in dBm must be converted to the equal in dB; we have $\alpha_n = 10^{P_{0n}/10}$.

Where the least square optimization criterion is different from maximum likelihood criterion, as in maximum likelihood criterion, the nth equation will be subtracted by the first $(n-1)$ equation, with the promise that the error in nth equation is very small, which cannot be guaranteed in real localization environment.

2.5. Computational Complexity Analysis. Computational complexity is an average measurement of calculated quantity of processing method via pseudocode. And from analysis, the computational complexity of least square-based self-adaptive localization method is $O(knm)$, which is the same as that of maximum likelihood-based self-adaptive localization algorithm, where k is the number of anchor nodes, n is the times of iteration, and m is the number of localization points.

If we equate M (the number of sample points) to n (the number of iterations), which are in Monte Carlo-based localization method and self-adaptive localization method, respectively, the computational complexity of these two localization methods is the same.

3. Performance Evaluation

In this section, the performance of the least square-based WSN self-adaptive localization algorithm will be evaluated and analyzed in comparison with that of the maximum likelihood-based self-adaptive localization algorithm [2].

3.1. Experimental Settings and Evaluating Indicators

3.1.1. Experimental Settings. The simulation environment is a 3.2 m × 3.2 m area, where 4 anchor nodes are positioned at the four vertexes and unknown nodes move horizontally and vertically at intervals of 0.8 m to establish a total of 25 fixed points, as shown in Figure 2. Since the unknown nodes match with the anchor nodes at the four vertexes, the unknown nodes at the vertexes are separately moved by 0.1 m inwards both in X direction and in Y direction. The unknown nodes and the anchor nodes at an arbitrary fixed point within the area are mutually through. In the simulation, the distance carries Gaussian noise datum $\hat{d}$. Use a log normal propagation model to convert $\hat{d}$ into the arithmetic for RSSI value.

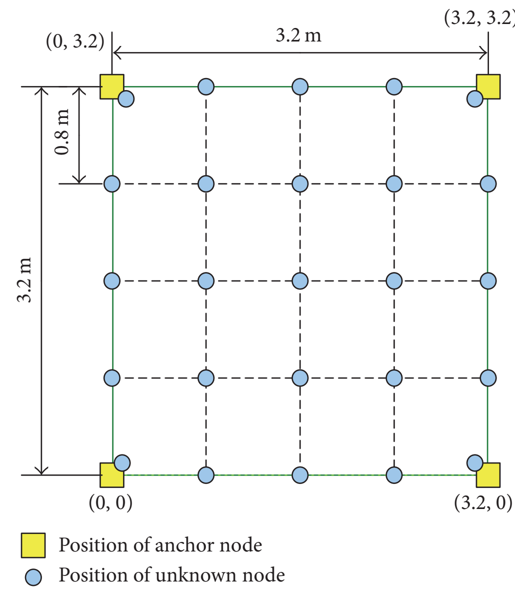


Figure 2: The localization field.

The computation platform used in the process of evaluation is parameterized as follows: CPU, i7 720QM@1.6 GHz, RAM, 4 GB, operating system, Windows XP Professional SP3, and evaluation software, Matlab 7.5.

3.1.2. Evaluating Indicators. Use the RMSE at every fixed point to evaluate the performance of the localization algorithm [4–7], as expressed in the following equation:

$$\mathrm{RMSE} = \sqrt{(\hat{x} - x)^2 + (\hat{y} - y)^2}, \quad (12)$$

where $(\hat{x}, \hat{y})$ represents the estimated coordinates of the unknown node and (x, y) represents the real anchor nodes of the same unknown node. A smaller RMSE value implies less localization error and higher localization accuracy.

3.1.3. Experiment Design. Firstly, the proposed self-adaptive localization algorithm will be evaluated in terms of localization accuracy and convergence, and a performance analysis will be given to the proposed algorithm in comparison with other algorithms; finally, different self-adaptive localization algorithms will be compared in terms of processing time.

Table 1: Localization errors using maximum likelihood-based self-adaptive localization method.

Error/m	10 times	20 times	30 times	200 times	1000 times
Variance $\sigma = d/10$	1.30 m	1.28 m	1.25 m	0.93 m	0.82 m
Variance $\sigma = 0.2$	1.33 m	1.30 m	1.28 m	1.11 m	0.98 m

Table 2: Localization errors using the least square-based self-adaptive localization method (proposed).

Error/m	10 times	20 times	30 times	200 times	1000 times
Variance $\sigma = d/10$	0.97 m	0.94 m	0.90 m	0.29 m	0.33 m
Variance $\sigma = 0.2$	0.65 m	0.34 m	0.31 m	0.56 m	0.56 m

3.2. Analysis of Location Accuracy

3.2.1. Accuracy Comparison with Maximum Likelihood-Based Self-Adaptive Method. Use both the maximum likelihood-based self-adaptive localization algorithm [2] and the proposed least square-based self-adaptive localization algorithm to locate in the above-mentioned environment. Set the initial values for $P_0 = -40$ dB and $\beta = 3$. The threshold values for P_0 and β are 1×10^{-3} and 1×10^{-3}, respectively. The iteration times are set to 10, 20, 30, 200, and 1000, respectively, and the distribution model of the background white noise is set to variance $\sigma = d/10$ and $\sigma = 0.2$. Then, the mean localization errors for those two algorithms are tabularized in Tables 1 and 2 separately.

As shown in Tables 1 and 2, the least square-based self-adaptive localization method outputs lower localization error than the maximum likelihood-based self-adaptive localization method.

After iterating 1000 times with the maximum likelihood-based self-adaptive localization method, the localization errors at all the fixed points are shown in Figures 3(a) and 4(a); under the same conditions using the least square-based self-adaptive localization method, the localization errors at all the fixed points are shown in Figures 3(b) and 4(b).

As shown in Figures 3 and 4, the least square-based self-adaptive localization method provides less localization error than the maximum likelihood-based self-adaptive localization method. Conclusively, the least square-based self-adaptive localization method behaves higher localization performance.

3.2.2. Accuracy Comparison with Monte Carlo-Based Localization Methods. Table 3 shows the localization accuracy of different Monte Carlo-based localization methods, including MCL [4], MCLS [5], and GPLA [6], where the number of nodes k takes the value of 25, the number of anchor nodes m takes value of 4, and the number of sample points M takes value of 10, 20, 30, 200, and 1000, respectively. We also show the location accuracy of our proposed least square-based self-adaptive method in Table 3, where variance σ takes the value of 0.2.

Table 3: Localization errors of different localization methods.

Sample points	10	20	30	200	1000
MCL	1.38 m	1.21 m	1.13 m	0.93 m	0.87 m
MCLS	1.19 m	1.13 m	1.08 m	0.89 m	0.82 m
GPLA	0.98 m	0.97 m	0.95 m	0.31 m	0.30 m
Least square-based self-adaptive (proposed)	0.65 m	0.34 m	0.31 m	0.56 m	0.56 m

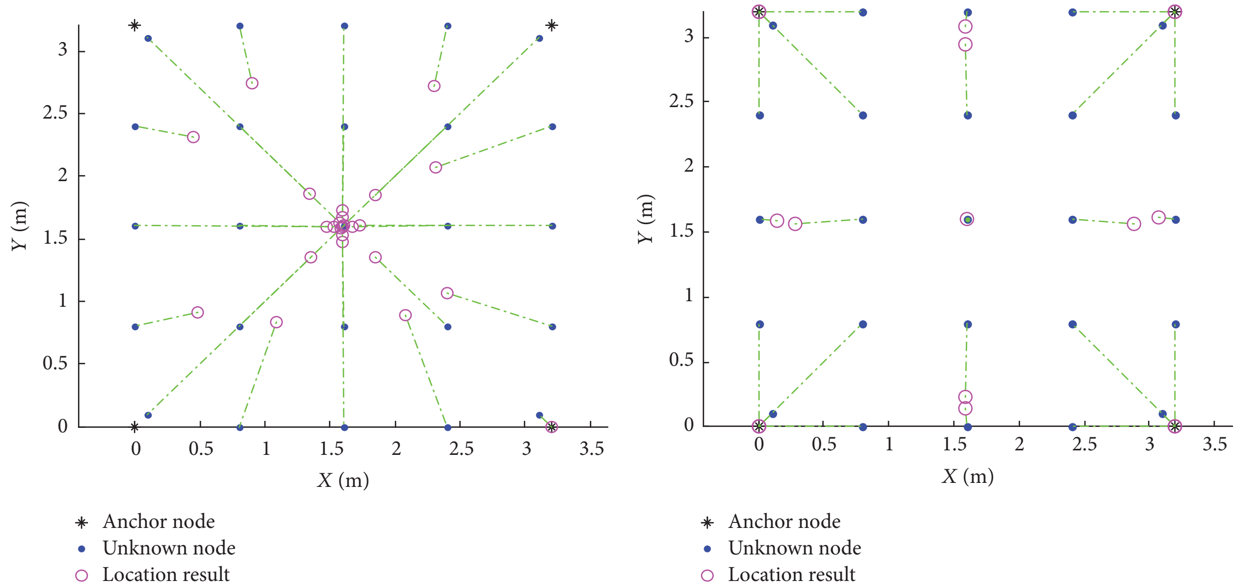
Table 3 illustrates that the localization accuracy of Monte Carlo-based localization methods is lower than that of the proposed least square self-adaptive localization method in this paper. That is mainly owing to the least square optimization criterion, which makes localization estimation value closer to the real value.

3.2.3. Accuracy Comparison with Bound-Box-Based Localization Methods. To further evaluate the performance of the least square-based self-adaptive localization method, a comparison was given in terms of localization error after 1000 iteration times between the least square-based self-adaptive localization method and the original and modified Bound-box localization algorithms [4–7], as shown in Table 4.

In Bounding-box (B-box for simplicity), the centroid of the overlap region among different communication ranges is treated as location estimation value. Considering different contributions of each distance estimation value, the weighted Bounding-box was proposed to improve the location accuracy, where the weighted values are the reciprocal of distance estimation values. In the 3-point centroid Bounding-box method, three anchor nodes are selected from four anchor nodes, and we get four groups of anchor nodes and their corresponding location estimation values; then we treat the mean of the four location estimation values as the final location estimation result. And the 3-point weighted centroid Bound-box is the weighted improved version of 3-point centroid Bound-box, where the weighted value is the reciprocal of the sum of the values of three distances in each group.

As shown in Table 4, the localization performance, expressed with two variances, using the self-adaptive localization algorithm after 1000 iteration times is comparative to that using the original or modified Bounding-box localization methods. It turns out that, compared with the localization algorithms that require preliminary environment test, the proposed self-adaptive localization algorithm does not require preliminary environment test for radio propagation parameters, while the localization accuracy is close to the original and modified Bounding-box localization methods, though the processing time is longer.

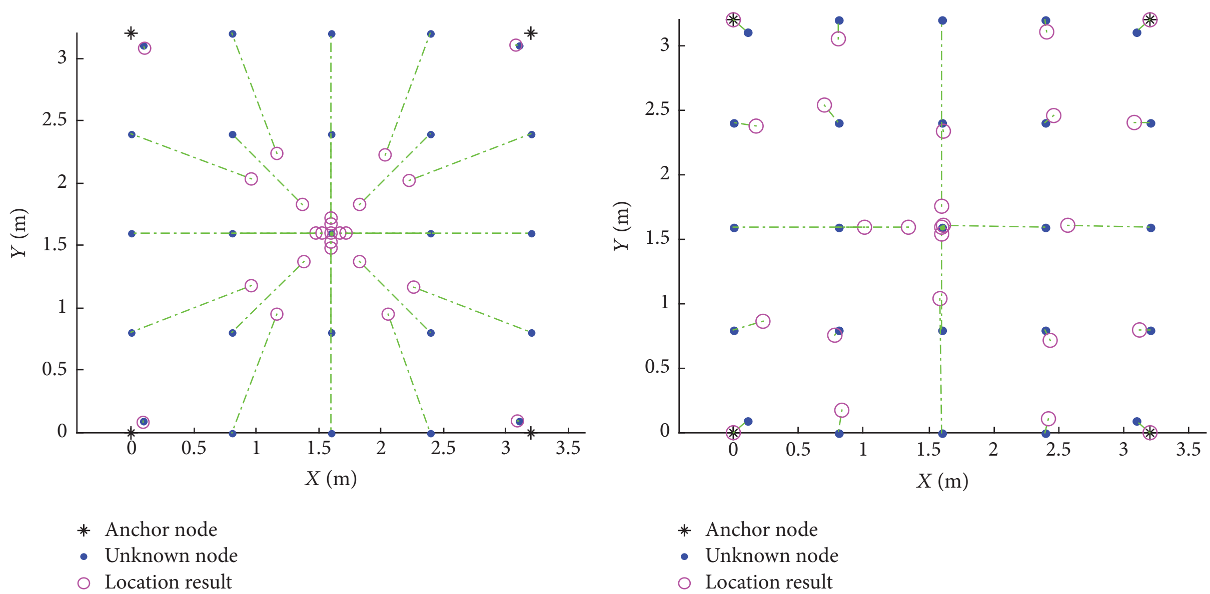
3.3. Analysis of Location Error Convergency. Location error convergence is the property that location errors of different



(a) Maximum likelihood-based self-adaptive localization

(b) Least square-based self-adaptive localization

FIGURE 3: Localization error of self-adaptive localization methods with (0, 0.2) noise.



(a) Maximum likelihood-based self-adaptive localization

(b) Least square-based self-adaptive localization

FIGURE 4: Localization error of self-adaptive localization methods with (0, $d/10$) noise.

iteration stages have the same tendency to the zero end state; that is, with the increment of iteration number, the value of location error gets smaller and smaller. We conducted an analysis of location error convergence to evaluate the convergence speed of different location methods.

After iterating 1000 times with the maximum likelihood-based self-adaptive localization method, the convergence results of localization errors at the 25 fixed points within the localization area are shown in Tables 5 and 6; with the same conditions, the localization errors convergence graphs of least square-based self-adaptive localization method are shown in Tables 5 and 6.

As shown in Tables 5 and 6, the least square-based self-adaptive localization method provides a higher convergence rate of localization error in comparison with the maximum likelihood-based self-adaptive localization method. Specifically, about 84% of unknown nodes completed their convergence within 1000 iteration times using the former method, while only about 52% of unknown nodes completed their convergence within 1000 iteration times using the latter

Table 4: Comparison between least square-based adaptive localization method and Bounding-box-based localization methods.

Error/m	B-box	Weighted B-box	3-point centroid B-box	3-point weighted centroid B-box	Least square-based self-adaptive (proposed)
Variance $\sigma = d/10$	0.35 m	0.28 m	0.45 m	0.45 m	0.33 m
Variance $\sigma = 0.2$	0.34 m	0.23 m	0.44 m	0.44 m	0.56 m

Table 5: Localization error convergence graphs of different location method with (0, d/10) noise.

Serial number of unknown node	Localization error convergence (iteration times)	
	Maximum likelihood-based self-adaptive	Least square-based self-adaptive (proposed)
1	150	120
2	1200	300
3	No convergence	400
4	1100	270
5	160	150
6	1000	275
7	8000	180
8	No convergence	No convergence
9	8000	370
10	1100	250
11	No convergence	8000
12	No convergence	No convergence
13	No convergence	No convergence
14	No convergence	700
15	No convergence	No convergence
16	1100	200
17	8000	280
18	No convergence	No convergence
19	8000	300
20	1200	220
21	100	100
22	1100	200
23	No convergence	480
24	1200	200
25	130	120

method. Comparatively, the proposed least square-based self-adaptive localization method is able to work at a higher convergence rate.

3.4. Analysis of Location Processing Time. As the computational complexity of Monte Carlo-based localization method and self-adaptive localization method is the same, we analyze more detail; that is, we compare the processing times.

Table 6: Localization error convergence graphs of different location method with (0, 0.2) noise.

Serial number of unknown node	Localization error convergence (iteration times)	
	Maximum likelihood-based self-adaptive	Least square-based self-adaptive (proposed)
1	15	20
2	400	30
3	No convergence	100
4	400	60
5	15	5
6	4000	40
7	8000	60
8	No convergence	100
9	8000	90
10	400	15
11	15	90
12	15	100
13	15	No convergence
14	No convergence	100
15	15	60
16	4000	50
17	6000	60
18	6000	100
19	7000	100
20	400	40
21	No convergence	5
22	8000	20
23	15	100
24	5000	120
25	15	5

Table 7: Processing time of maximum likelihood-based self-adaptive localization method.

Time/s	10 times	20 times	30 times	200 times	1000 times
Variance $\sigma = d/10$	12.75 s	18.71 s	23.15 s	205.30 s	1087.13 s
Variance $\sigma = 0.2$	11.59 s	22.97 s	28.84 s	212.64 s	1058.04 s

Table 8: Processing time of the least square-based self-adaptive localization method (proposed).

Time/s	10 times	20 times	30 times	200 times	1000 times
Variance $\sigma = d/10$	3.04 s	5.18 s	7.21 s	44.04 s	134.77 s
Variance $\sigma = 0.2$	4.17 s	6.79 s	9.43 s	28.47 s	28.64 s

3.4.1. Processing Time Comparison with Maximum Likelihood-Based Self-Adaptive Method. Tables 7 and 8 show the processing time of localization errors separately using the maximum

Table 9: Localization processing time of different localization methods.

Sample points	10	20	30	200	1000
MCL	0.9 s	1.7 s	2.5 s	17.3 s	90.3 s
MCLS	1.1 s	2.1 s	3.4 s	21.9 s	112 s
GPLA	1.3 s	2.5 s	3.9 s	25.1 s	132 s
Least square-based self-adaptive (proposed)	4.17 s	6.79 s	9.43 s	28.47 s	28.64 s

likelihood-based self-adaptive localization method and the least square-based self-adaptive localization method at the iteration times of 10, 20, 30, 200, and 1000, with different variances of background noise.

As shown in Tables 7 and 8, at the same iteration times, the processing time using the maximum likelihood-based self-adaptive localization algorithm is about 4 times longer than that using the least square-based self-adaptive localization method. Conclusively, the least square-based self-adaptive localization method can provide a higher computational efficiency, for the least square technique reduces the computational amount in the radio parameter estimation process.

3.4.2. Processing Time Comparison with Monte Carlo-Based Methods. Table 9 shows the localization processing time of different Monte Carlo-based localization methods, including MCL [4], MCLS [5], GPLA [6], and our proposed least square-based self-adaptive method, where the number of nodes k takes the value of 25, the number of anchor nodes m takes value of 4, and the number of sample points M takes value of 10, 20, 30, 200, and 1000, respectively.

Table 9 illustrates that when the number of sample points is small, that is, it is less than 1000, localization processing time of Monte Carlo-based method is lower than that of the least square-based self-adaptive method. However, when the number of sample points is large, localization processing time of Monte Carlo-based method is bigger than that of the least square-based self-adaptive method. That is because least square-based self-adaptive method has ability on error convergence, while the number of iterations has to be set in advance in Monte Carlo-based method.

To sum up, the simulation results show that the proposed least square-based self-adaptive localization algorithm has definite advantages over the maximum likelihood-based self-adaptive localization algorithm in terms of both localization accuracy and localization processing time. Considering the fact that the nodes in a wireless sensor network are still, the environment changes relatively slowly and the signal attenuation parameter varies slowly with time in the communication environment; it is believed that the proposed least square-based self-adaptive localization method is capable of satisfying the typical dynamic localization requirement.

4. Conclusion

The traditional WSN localization method requires a preliminary environmental test to determine the radio signal propagation parameters, leading to a complex localization process, highly experimental workload, and poor environment adaptability. In view of these weaknesses, this paper proposes a least square-based WSN self-adaptive localization method. Using least square technique and iteration strategy to estimate the radio parameters, this method not only reduces the computational amount in the localization process but also improves the localization accuracy. It provides methodological and technical means for the dynamic localization applications.

It is requisite for a self-adaptive localization method to be finally applied into an actual WSN localization system. For this reason, the proposed localization method is going to be demonstrated and evaluated in the true WSN localization environment.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work is funded by National Natural Science Foundation of China (61601142 and 61671174), space support technology fund projects (2014-HT-HGD5), Natural Science Foundation of Shandong Province of China (ZR2015FM027), State Key Laboratory of Geo-information Engineering (no. SKLGIE2014-M-2-4), State Key Laboratory of Satellite Navigation System and Equipment Technology, Guangxi Key Laboratory of Automatic Detecting Technology and Instruments (YQ14205 and YQ15203), and Natural Scientific Research Innovation Foundation in Harbin Institute of Technology (HIT.NSRIF.201721).

References

[1] Y. Liu, Z. Yang, X. Wang, and L. Jian, "Location, localization, and localizability," *Journal of Computer Science and Technology*, vol. 25, no. 2, pp. 274–297, 2010.

[2] Y. Nishi and T. Ohtsuki, "A distributed localization with unknown attenuation coefficient in wireless sensor networks," in *Proceedings of the 14th Asia-Pacific Conference on Communications (APCC '08)*, pp. 1–5, Tokyo, Japan, October 2008.

[3] R. Zemek, S. Hara, K. Yanagihara et al., "A joint estimation of target location and channel model parameters in an IEEE 802.15.4-based wireless sensor network," in *Proceedings of the 18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '07)*, September 2007.

[4] Y.-B. Yao, R. Zeng, and Z.-Q. Yi, "Bounding box based distributed search localization algorithm for WSN," *Journal on Communications*, vol. 33, no. 2, pp. 135–140, 2012.

[5] A. Savvides, H. Park, and M. Srivastava, "The bits and flops of the n-hop multi-alterations primitive for node localization problems," in *Proceedings of the ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '02)*, pp. 112–121, 2002.

[6] GuoHaiqi, *Research of the localization method based on ZigBee [M.S. thesis]*, South West Communication University, 2007.

[7] W. Dan, *Research on the localization method based on RSSI for wireless sensor network [M.S. thesis]*, Harbin Institute of Technology, 2011.

[8] Y. Peng, Q. Luo, D. Wang, and X. Peng, "WSNs distance estimation method with RSSI-D using interval data clustering algorithm," *Chinese Journal of Scientific Instrument*, vol. 33, no. 3, pp. 491–498, 2012.

[9] Q. Luo, Y. Peng, X. Peng, and A. El Saddik, "Uncertain data clustering-based distance estimation in wireless sensor networks," *Sensors*, vol. 14, no. 4, pp. 6584–6605, 2014.

[10] Q. Luo, Y. Peng, J. Li, and X. Peng, "RSSI-based localization through uncertain data mapping for wireless sensor networks," *IEEE Sensors Journal*, vol. 16, no. 9, pp. 3155–3162, 2016.

[11] L. Hu and D. Evans, "Localization for mobile sensor networks," in *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom '04)*, pp. 45–57, Philadelphia, Pa, USA, October 2004.

[12] L. I. Min, L. Ting, and X. U. Hua, "Localization algorithm based on anchor node select model for wireless sensor networks," *Chinese Journal of Sensors and Actuators*, vol. 24, no. 2, pp. 264–268, 2011.

[13] Z.-L. Shan, L.-H. Liu, Y.-S. Zhang, and G.-X. Huang, "A strong self-adaptivity localization algorithm based on gray prediction model for mobile nodes," *Journal of Electronics and Information Technology*, vol. 36, no. 6, pp. 1492–1497, 2014.

[14] Y. Wang, W. Chen, J. Wang, and P. Xiao, "Self-adaptive pose-tracking algorithm for mobile robots in dynamic and highly-occluded environments," *Jiqiren/Robot*, vol. 37, no. 1, pp. 112–121, 2015.