

Research Article

Research on Mobile Node Localization Optimization Algorithm in Wireless Sensor Networks Based on Monte Carlo Method

Jian'qiang He ^{1,2}

¹Electronic Information and Electrical College of Engineering, Shangluo University, Shangluo 726000, China

²Shangluo Artificial Intelligence Research Center, Shangluo 726000, China

Correspondence should be addressed to Jian'qiang He; hejianqiang@slxy.edu.cn

Received 5 February 2022; Revised 11 August 2022; Accepted 21 September 2022; Published 20 October 2022

Academic Editor: Carles Gomez

Copyright © 2022 Jian'qiang He. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Mobile node localization is one of the key technologies in wireless sensor network applications. Aiming at the shortcomings of Monte Carlo algorithm (MCL) in mobile wireless sensor network node localization in nonideal environments, such as low accuracy and low sampling rate, in the prediction phase and filtering phase of MCL, the communication radius of the unknown node is determined according to the size of the irregularity of the node. Perform layering, assign adaptive weights to anchor nodes of different layers according to the area where they are located, and propose an adaptive improved Monte Carlo algorithm. After simulation analysis, the algorithm has an average localization error of nodes under different regularity conditions. It has dropped by about 12%, and the localization error has dropped by about 10% on average under different speed conditions. Aiming at the shortcomings of the MCL algorithm in mobile wireless sensor mobile networks such as low localization accuracy, large sample demand, and long localization time, the communication radius of the node is fuzzified to reduce the sampling area of the node, and an improved Monte Carlo localization algorithm based on fuzzy theory is proposed. The improved Monte Carlo localization algorithm, after simulation analysis, is about 50% shorter than the traditional MCL algorithm in localization time, and the localization accuracy is up to about 30% higher than the traditional MCL algorithm.

1. Introduction

In recent years, with the widespread application of wireless sensor networks (WSN), the localization technology of wireless sensor network nodes has become a hot issue of research. At present, the static WSN localization technology is relatively mature. Representative algorithms include centroid localization algorithm [1], SPA relative positioning algorithm [2], convex programming localization algorithm [3], and DV-Hop localization algorithm [4, 5]. Most of these node localization algorithms do not consider the mobility of the nodes and can no longer meet the localization requirements in the actual environment. How to design a location algorithm suitable for WSN mobile nodes has become a hot issue in research.

For the localization of mobile nodes, some WSN dynamic localization algorithms have been proposed by the academic community, such as DSL localization algorithm [6], DRL localization algorithm [7], Landscape-3D algo-

rithm [8], and MBL-MDS localization algorithm [9]. However, these algorithms generally have problems such as large amount of calculation, high hardware requirements, or more beacon nodes. Because the MCL method uses the mobility of nodes to help in positioning, it provides a new idea to solve the problem of node localization in mobile wireless sensor networks, and more and more scholars derive their own improvement schemes based on this algorithm. For example, the Monte Carlo Box Positioning Algorithm (MCB) proposed in literature [10] introduces anchor boxes and sample boxes to reduce the sampling area; the MSL* algorithm proposed in literature [11] introduces one-hop and two-hop common nodes and anchor nodes together position. Literature [12] combines Monte Carlo node localization algorithm with RSSI. Literature [13] proposes Monte Carlo node localization based on timing, which uses the order of anchor node feedback signals within one-hop to form a sampling area. Literature [14] proposed a Monte Carlo node location algorithm based on Markov

chain, which uses Markov chain to calculate the weight of particles in the process of predictive filtering. Literature [15] proposed a Monte Carlo mobile node location algorithm based on Newton interpolation. Literature [16] proposed wavelet transform to predict mobile node location algorithm, using wavelet transform method to predict node location through historical trajectory. Literature [17] combined differential evolution algorithm with Monte Carlo node localization algorithm. Literature [18] shows that the combination of particle swarm optimization and Monte Carlo node localization algorithm reduces the number of invalid samples and enhances the robustness of the algorithm. The above improved algorithms are all tested in an ideal environment. The localization accuracy of nodes is too dependent on anchor nodes, and the localization results are not ideal when the number of anchor nodes is small. High localization accuracy is based on a large amount of communication energy consumption, which does not comply with the WSN principle of low cost and low energy consumption.

Therefore, based on the Monte Carlo node localization algorithm, this paper changes the size of the filter radius in real time according to the size of the irregularity and provides the weight of the sampling particles that changes with the irregularity. The center of the sampling circle is improved from the estimated position of the original algorithm to the sampled particle position at the last moment, and this algorithm is applied to the WSN mobile node localization and strives to solve the problem of node localization in nonideal environments [19]. Through fuzzy theory, the node signal energy value is fuzzified, and the filtering conditions are accurate to make up for the shortcomings of MCL localization, such as large demand for samples, low positioning accuracy, and long positioning time.

2. Principles of Monte Carlo Algorithm

The Monte Carlo method is based on the Bayesian filtering method. The core idea of the algorithm is to use N weighted discrete samples to estimate the posterior probability density, and use importance sampling to iteratively update [20]. Including the prediction stage and the filtering stage, the basic operation steps are as follows:

- (1) *Initialize the Node Position.* According to the one-hop and two-hop anchor nodes of the position node, use the centroid method to estimate the position of the unknown node
- (2) *Prediction Stage.* The motion speed and direction of the node to be determined are unknown, only the maximum motion speed V_{\max} is known. Suppose l_{t-1} is the position of the node at the previous moment, then the position of the node at the current moment is in the circle with l_{t-1} as the center and V_{\max} as the radius, as shown in Figure 1.

Assuming that the velocity of node V satisfies uniform distribution on $[0, V_{\max}]$, $d(l_{t-1}, l_t)$ represents the Euclidean

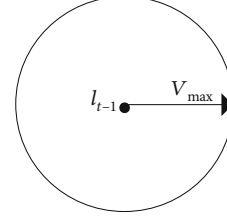


FIGURE 1: The area where the unknown node is located.

distance between nodes at adjacent times. Then the probability of node position estimation at time t is:

$$p(l_t|l_{t-1}) = \begin{cases} 1/\pi V_{\max}^2 & d(l_{t-1}, l_t) < V_{\max}, \\ 0 & d(l_{t-1}, l_t) \geq V_{\max}. \end{cases} \quad (1)$$

In the formula, $p(l_t|l_{t-1})$ is called the transition equation, which is used to represent the node position distribution at time $t-1$ to predict the position distribution at time t .

- (3) *Filter Stage.* Filter the sampled samples according to the information of the first-hop and second-hop anchor nodes received by the node, and filter out the samples that do not meet the requirements. If S is a set of one-hop anchor nodes, as shown in the shaded area in Figure 2, T is a set of two-hop anchor nodes, as shown in the shaded area in Figure 3, r is the communication radius of the anchor node, l is the sampling particle, s is a one-hop anchor node, and $d(l, s)$ is the Euclidean distance of point l , then the filter condition of the node can be expressed as

$$\text{filter}(l) = \forall s \in S, d(l, s) \leq r \wedge \forall s \in T, r < d(l, s) \leq 2r. \quad (2)$$

- (4) *Resampling Stage.* After the filtering stage is completed, the number of remaining samples may be very small. Generally, it is not possible to directly sample from the posterior distribution. It is necessary to use nonstrict filtering conditions, that is, the distance between the sampling node and the one-hop anchor node is moderately expanded to meet the filtering requirements. The conditional particle is called a sampled particle. When 50 sampled particles are obtained, the average of the position coordinates of all sampled particles is calculated as the estimated position of the node

3. Principle of Adaptive Improved Monte Carlo Algorithm

Although the MCL algorithm solves the localization problem when the node moves to a certain extent, it cannot achieve better positioning accuracy when the radio range of the node is not ideal or the node is fast. Therefore, in this paper, the communication radius of the unknown node is

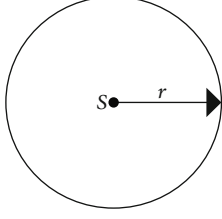


FIGURE 2: One-hop anchor node filtering area diagram.

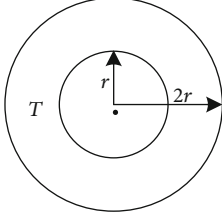


FIGURE 3: Two-hop anchor node filtering area diagram.

stratified according to the size of the irregularity, and the anchor nodes in different layers are given corresponding weights according to the region where they are located. The weights of the sampling nodes are adaptively adjusted by using an adaptive algorithm to achieve the optimal weight. The MCL algorithm is improved from the prediction stage and the filtering stage to improve the accuracy of mobile node localization algorithm in wireless sensor networks.

3.1. Prediction Stage. The MCL algorithm sampling uses importance sampling, that is, sampling from an area with high particle density. The basic idea is to choose a proposed density $q(x)$ to replace the real probability density $p(x)$ that is difficult to sample, then the MCL integration problem can be transformed into:

$$\int f(x)p(x) = \int f(x) \frac{p(x)}{q(x)} q(x) dx. \quad (3)$$

In the formula, $p(x)$ is the actual density, $q(x)$ is the estimated density instead of $p(x)$. Suppose that the shape of $q(x)$ covers the shape of $p(x)$, that is, the definition of $q(x)$ is greater than or equal to the domain of definition of $p(x)$.

Importance sampling in the MCL algorithm uses the weighting of multiple independent samples $q(x)$ to approximate

$$\hat{f} = \frac{1}{N_p} \sum_{i=1}^{N_p} w(X^{(i)}) f(X^{(i)}). \quad (4)$$

In the formula, N_p is the number of sampling particles, $w(X^{(i)}) = p(X^{(i)})/q(X^{(i)})$ is the weight, since the normalization of $p(x)$ is unknown, in order to make $\sum_{i=1}^{N_p} w(X^{(i)}) = 1$,

$$\hat{f} = \frac{(1/N_p) \sum_{i=1}^{N_p} w(X^{(i)}) f(X^{(i)})}{(1/N_p) \sum_{j=1}^{N_p} w(X^{(j)})} = \sum_{i=1}^{N_p} \tilde{w}(X^{(i)}) f(X^{(i)}). \quad (5)$$

In the formula, $\tilde{w}(X^{(i)})$ is a normalized weight whose variance is

$$\begin{aligned} \text{Var}[\hat{f}] &= \frac{1}{N_p} \text{Var}_q[f(X)w(X)] \\ &= \frac{1}{N_p} \int \left[\left(\frac{(f(X)p(x))^2}{q(x)} \right) \right] dX - \frac{1}{N_p} (E_p[f(X)])^2. \end{aligned} \quad (6)$$

In order to improve the positioning accuracy, the variance value needs to be reduced. It can be seen from formula (6) that when the shapes of $q(x)$ and $p(x)$ are more similar, the variance is smaller and the precision is higher. MCL algorithm is that the estimated position of the node at the above moment is sampled by the center. This algorithm uses the sampling point at the above moment as the center to sample, so that the shape of $q(x)$ and $p(x)$ is more similar so as to improve the positioning accuracy.

3.2. Filtering Phase. The premise of the MCL algorithm is an ideal environment, that is, the wireless ranges of unknown nodes and anchor nodes are fixed values, DOI (Degree of Irregular) = 0, It means that the radio range is the same value in different locations, different transmission directions, and different transmission times [21]. However, in the actual environment, the value of the radio range is different in all directions, that is, DOI > 0, and the radio range of the node is not an ideal circle. The schematic diagram is shown in Figure 4.

In the figure, the dotted line represents the radio range when DOI = 0, and the solid line represents the radio range when DOI = 0.015, where the communication radius is $r = 100\text{m}$.

Since DOI > 0 in the actual environment, as shown in Figure 5, point A is a jump anchor node, point B and point C are sampling points after the prediction stage, the dotted circle b is the ideal radio range, and the irregular solid line is the actual anchor node radio range. The improved algorithm modifies the filtering condition according to the influence of DOI on communication radius, which is expressed as formula (7). where S is the set of one-hop anchor nodes, T is the set of two-hop anchor nodes, r is the communication radius, l is the sampling particle, s is the anchor node, $d(l, s)$ is the Euclidean distance between point l and point s , for the sampling of one-hop anchor node particle, the distance $d(l, s)$ between the sampled particle and the anchor node is less than the sum of the radius r and r^*doi of the anchor node, as shown in the circle a in Figure 5, the circle c is the inscribed circle of the wireless range of the actual anchor node, indicating that there must be a jump anchor A collection of points within the range of node A.

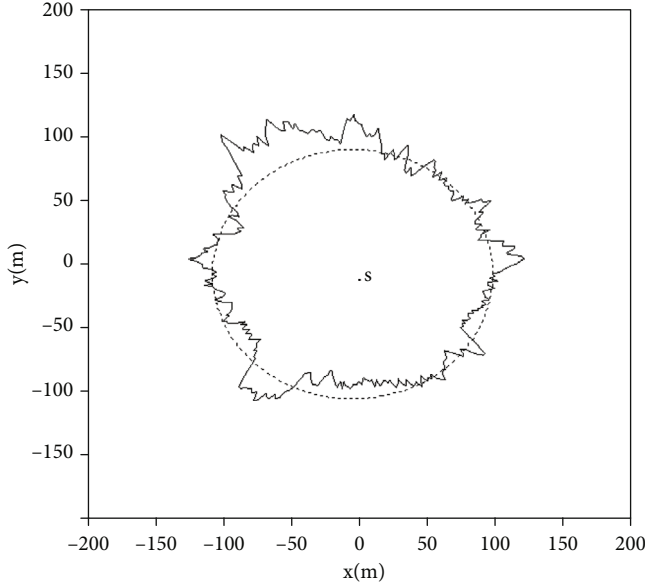


FIGURE 4: Schematic diagram of the influence of DOI value.

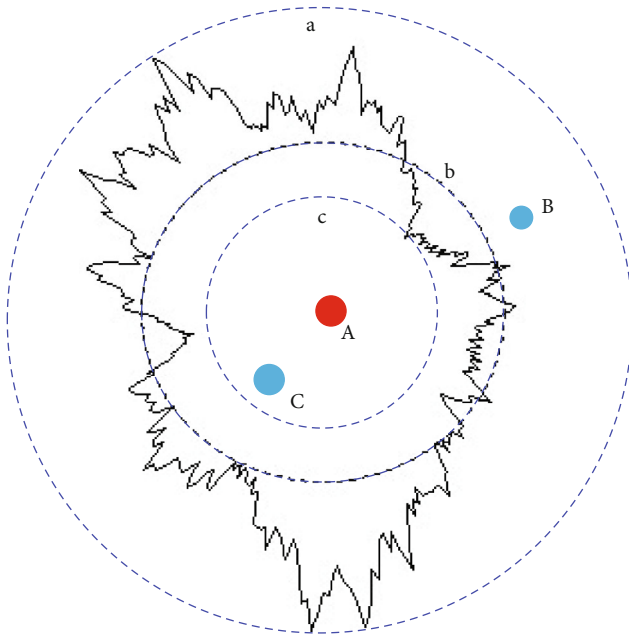


FIGURE 5: Distribution of sampling points at DOI > 0.

$$\text{filter}(l) = \forall s \in S, d(l, s) \leq r + r^* \text{doi} \wedge \forall s \in T, r - \text{doi}^* r < d(l, s) \leq 2r + \text{doi}^* r. \quad (7)$$

It can be seen from the adaptive filtering conditions that the improved MCL algorithm considers that the anchor nodes in different regions will affect the size of the positioning error. When the sampling nodes are in the sampling regions with different wireless ranges of the anchor nodes, the impact on the unknown nodes is different. In Figure 5, point C is definitely within the wireless

range of the anchor node, and the weight is set to 1; while point B is not necessarily within the range of the anchor node due to the influence of irregularity, and its impact on unknown nodes and nodes to be positioned, unlike point C. Based on this, the algorithm adjusts the sample weights of the sampled particles. Considering the change of the filtering conditions, if the weights of all particles are the same, it is not in line with the actual situation, so the corresponding weights are assigned to the sampled particles that meet different filtering conditions. When the sampling point absolutely satisfies the wireless range of the node, that is, in the circle c, the weight is set to 1. When the condition after the relaxation of the restriction is satisfied, that is, between the circle a and the circle c, the weight of the node changes with the DOI. However, since the area between the circle a and the circle c is positively correlated with the DOI of the node, in order to simplify, this algorithm defines the weight of the sampling point between the circle a and the circle c as $\text{doi}^* r/r$, then the comprehensive weight condition of the node is set as follows:

$$\text{Weight} = \begin{cases} 1 & \forall s \in S, d(l, s) \leq r - r^* \text{doi} \wedge \forall s \in T, \\ & r + \text{doi}^* r \leq d(l, s) \leq 2r - \text{doi}^* r \\ 0 & \text{else,} \\ (\text{doi}^* r/r) & \forall s \in S, r - r^* \text{doi} < d(l, s) < r + r^* \text{doi} \wedge, \\ & \forall s \in T, s - \text{doi}^* r < d(l, s) < r + \text{doi}^* r \vee, \\ & 2r - \text{doi}^* r < d(l, s) < 2r + \text{doi}^* r. \end{cases} \quad (8)$$

This algorithm takes the influence of irregularity into account, and the sampling particles in different regions have different influences on node positioning. Therefore, the adaptive filtering and adaptive weight improvement in the filtering stage just make up for the defects of the original algorithm and improve the positioning accuracy.

4. Simulation Analysis

Under the MCL-simulator, setting the WSN area size to 500*500m, the total number of nodes to 320, including 32 anchor nodes, the communication radius of the anchor node and the unknown node are both 50 m, the speed of the anchor node $V_s \in [S_{\min}, S_{\max}]$, and S_{\min} and S_{\max} represent the minimum and maximum speeds of the anchor nodes, respectively; the speeds $V \in [V_{\min}, V_{\max}]$, V_{\min} and V_{\max} of the nodes represent the minimum and maximum speeds of the nodes, respectively. 10 experiments are carried out for each algorithm, and the node moves 80 steps in each round, and finally the average value of the 80 positioning errors is obtained. The node has the ability to judge whether a node is within the communication range, but has no ranging ability. It has the ability to perceive its own speed, but does not have the ability to perceive the direction of movement, the positioning error uses the Euclidean distance between the estimated position and the actual position. It is expressed as the ratio of the communication radius. Figure 6 is the

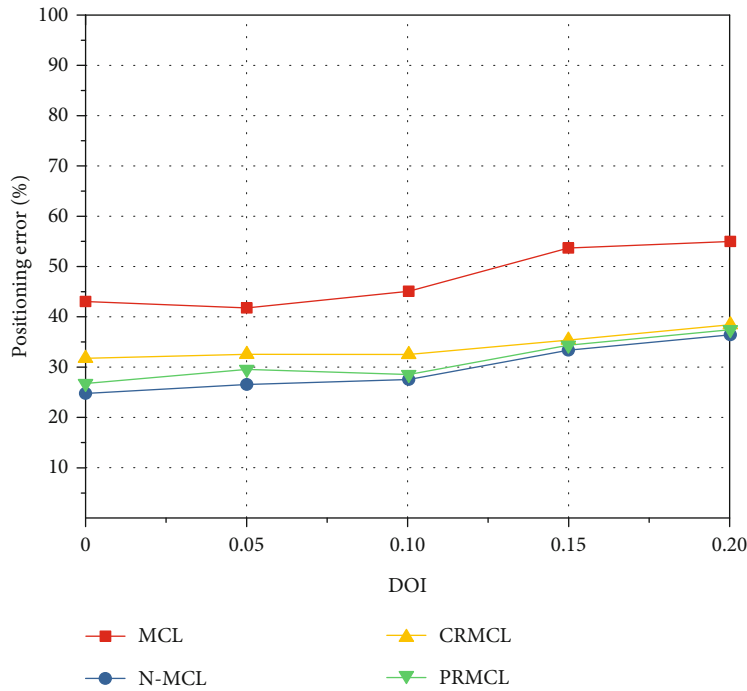


FIGURE 6: Comparison of node positioning errors under different DOIs.

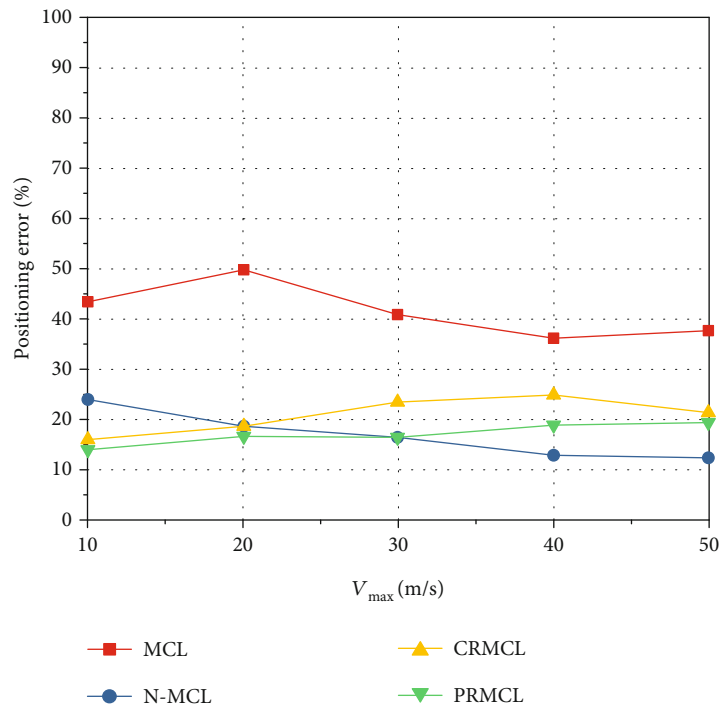


FIGURE 7: Influence on positioning error.

positioning error graph of the MCL algorithm and the adaptive MCL algorithm (N-MCL), and the CRMCL and PRMCL algorithm in literature [18] under different DOIs, where the maximum speed of the anchor node and the unknown node is 50 m/s.

It can be seen from Figure 6 that as the DOI increases, the radio propagation model of the node becomes more and more irregular, but the positioning accuracy of the new-MCL is always better than that of the MCL, and the average positioning error decreases by about 12%. With

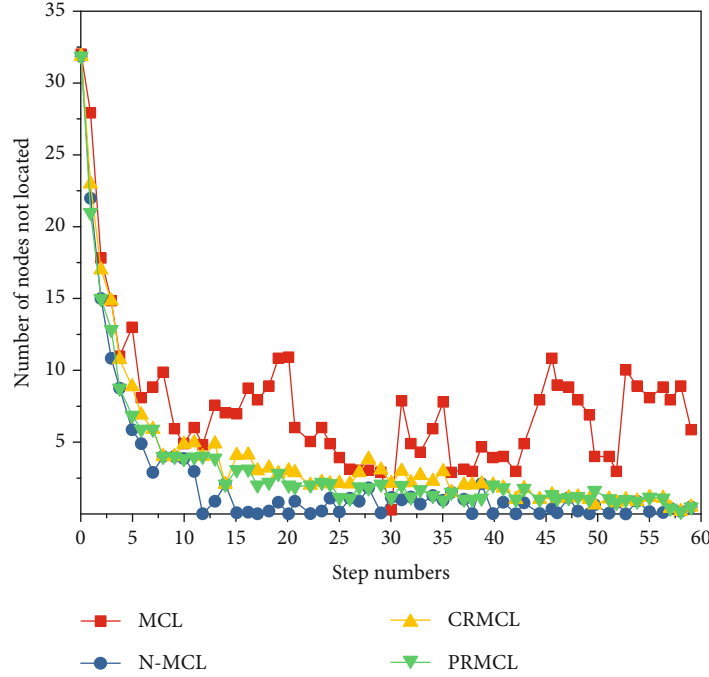


FIGURE 8: Comparison of the number of nodes that cannot be located.

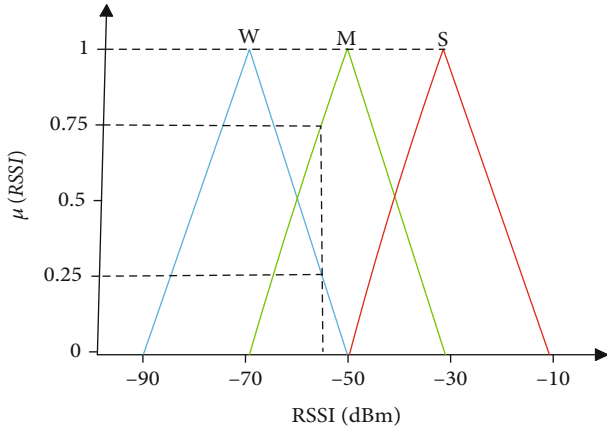


FIGURE 9: Relationship between signal strength and RSSI.

the gradual increase of DOI, the positioning errors of the four algorithms all increase to a certain extent, but the positioning error of the N-MCL algorithm is smaller than other algorithms, and the positioning accuracy of N-MCL is highest when the irregularity is 0.1.

Figure 7 shows the comparison of the average positioning error curves of new-MCL and MCL with the increase of V_{\max} . Assuming that the maximum velocity of the unknown node and the anchor node is the same, both are V_{\max} , and the actual velocity of the two is evenly distributed in interval $[0, V_{\max}]$. It can be seen from the figure that with the gradual increase of V_{\max} , the positioning error of N-MCL gradually decreases. When V_{\max} is more than 30 m/s, the positioning error of the algorithm is the lowest, and the average positioning error is reduced by about 10% compared with the MCL algorithm.

Figure 8 shows the node movement in 50 steps, and the number of nodes that cannot be positioned in each step. It can be seen from Figure 8 that the number of nodes that cannot be located by the improved algorithm is lower than that of the other three algorithms. After 46 times, the number of improved localizability is about 100%. The average number of nodes that cannot be located for 50 times is as follows: that the number of nodes that cannot be located by MCL is about 7.67, the number of nodes that cannot be located by CRMCL algorithm is about 3.26, the number of nodes that cannot be located by PRMCL algorithm is about 2.85, and the number of nodes that cannot be located by N-MCL algorithm is 2.33. The network coverage rate of the MCL algorithm is 97.36%, and the network coverage rate of the new-MCL algorithm is 99.19%. This is due to the relaxation of the filtering conditions, which increases the number of sampling particles that satisfy the conditions under the same sampling times, thereby increasing the number of locatable nodes.

5. Improved Monte Carlo Algorithm Based on Fuzzy Theory

Although the MCL algorithm has high positioning accuracy and computational efficiency for dynamic wireless sensor networks with limited node energy and low anchor node density, the value of the posterior probability density in the sample prediction area is small, and positioning requires a large number of samples. A sufficiently good positioning accuracy can be obtained. The nodes are sampled in a circle with the maximum speed as the radius each time, and the positioning time is long. Therefore, an MCL algorithm based on fuzzy theory is proposed, which makes up for the

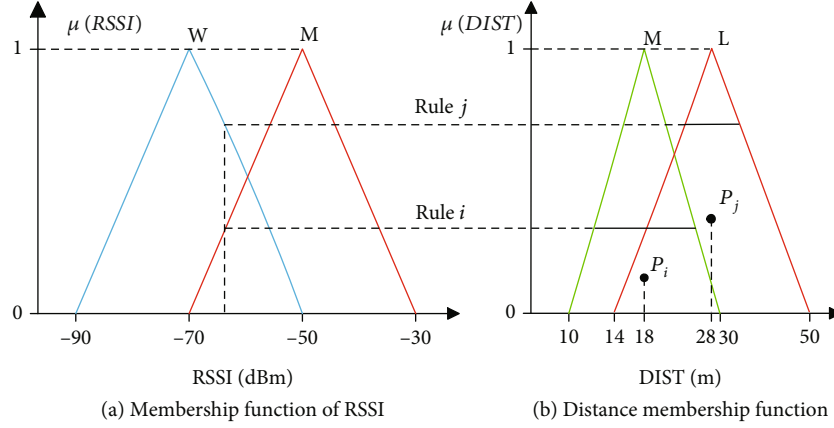


FIGURE 10: RSSI determines its area according to corresponding rules.

deficiencies of MCL by fuzzing the node signal energy value and making the filtering conditions more precise.

5.1. Algorithm Principle. Fuzzy theory is based on fuzzy sets and is the best way to solve problems with ambiguous boundaries. For a fuzzy set, in addition to the ordinary membership values that belong to 0 and do not belong to 1, the elements in the set are assigned to any value between 0-1 to represent the degree of membership [22]. Li et al. generalized that the characteristic function of classical sets can only take two numbers 0 and 1, and generalized it to all numbers in the closed interval of 0 and 1 [23]. And such characteristic function is called membership function $\mu(x)$, $\mu(x) \in [0, 1]$, and the set determined by $\mu(x)$ is called the fuzzy set on E . The membership function represents the degree to which A belongs to E . In this way, all intermediate values between “belongs to” and “does not belong to” can be represented.

The processing of fuzzy logic has been applied to robot localization [24], which redefines the classical set theory and makes it have nonstrict set boundaries. The so-called fuzzy sets use membership functions to describe a certain set element. Whether the feature belongs to the fuzzy set. An element can belong to one or two fuzzy sets at the same time [25]. Unlike traditional numbers that can only represent a single value, a fuzzy number can represent multiple values or an indeterminate relationship. RSSI is roughly proportional to the distance between the unknown node and the anchor node, and the triangular membership function is simple and easy to understand, so this algorithm uses the most common membership function such as formula (9):

$$\mu(x) = \begin{cases} 0 & \text{if } x < a, \\ (x-a)/(b-a) & \text{if } a \leq x \leq b, \\ (c-x)/(c-b) & \text{if } b \leq x \leq c, \\ 0 & \text{if } x > c, \end{cases} \quad (9)$$

where (a, b, c) is a triangular geometry, as shown in Figure 9, W set can be represented by $(-90, -70, -50)$ and M by $(-70, -50, -30)$, for example, a certain value $RSSI = -55dBm$ has

a membership degree of 0.25 in W and a membership degree of 0.75 in M .

The coordinate position of the node can be represented by (X, Y) , where X and Y are both fuzzy numbers. Divide the signal strength $RSSI$ into three levels: “ H ,” “ M ,” and “ L .” Suppose the unknown node S has n one-hop anchor nodes. The coordinates of node S represent $S(X, Y)$. The coordinates of the N one-hop anchor nodes are represented as $A_1(x_1, y_1)$, $A_2(x_2, y_2)$, \dots , $A_n(x_n, y_n)$, respectively. Then the coordinates of the nodes can be expressed as:

$$\begin{aligned} F_1 &= (X - x_1)^2 + (Y - y_1)^2 - D_1^2 = 0, \\ F_2 &= (X - x_2)^2 + (Y - y_2)^2 - D_2^2 = 0, \\ &\dots \\ F_n &= (X - x_n)^2 + (Y - y_n)^2 - D_n^2 = 0. \end{aligned} \quad (10)$$

Equation (10) defines a nonlinear equation to describe the distance relationship between unknown nodes and anchor nodes. Where X, Y , and $D_k (k = 1 \dots n)$ are the fuzzy numbers and $(x_k, y_k) (k = 1 \dots n)$ is the exact number, the result minimizes the squared difference of this series of equations.

Fuzzy theory makes quantitative, systematic, and hierarchical analysis of nonquantitative events [26]. This method can simplify complex problems, and this paper applies it to the wireless sensor network mobile node location algorithm. The propagation area of nodes in mobile node positioning is divided into several levels, and an orderly hierarchical model is established between the levels according to the affiliation. The judgment of certain objective facts in each layer reflects the importance of each layer in a quantitative form.

5.2. Algorithm Description

- (1) *Initialize the Node Position.* According to the one-hop and two-hop anchor nodes of the position node, use the centroid method to estimate the position of the unknown node.
- (2) *Sampling Stage.* First, establish fuzzy rules *Rule_i*: “If $RSSI$ is W , then $DIST$ is L ”; *Rule_j*: “If $RSSI$ is M , then

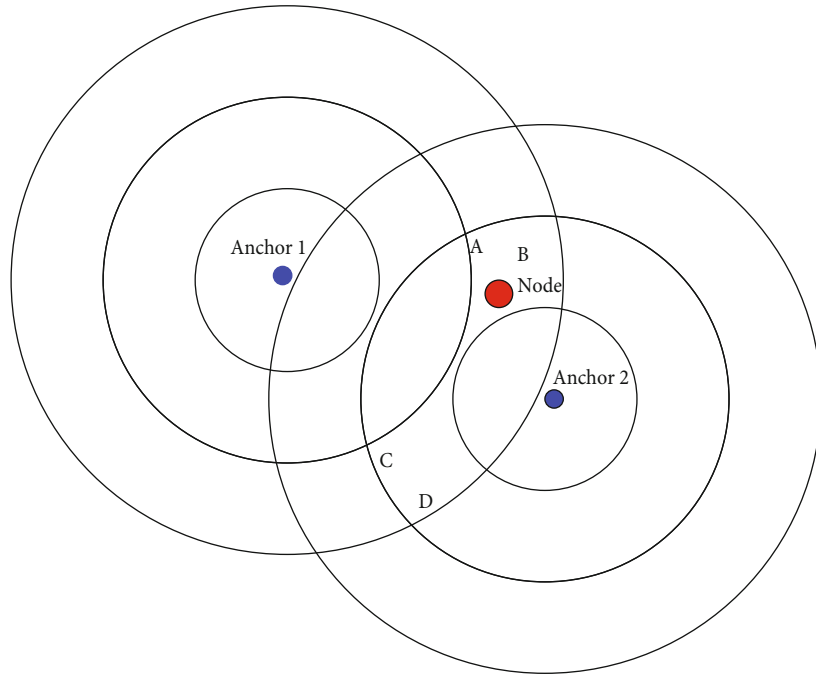


FIGURE 11: Determination of sampling range.

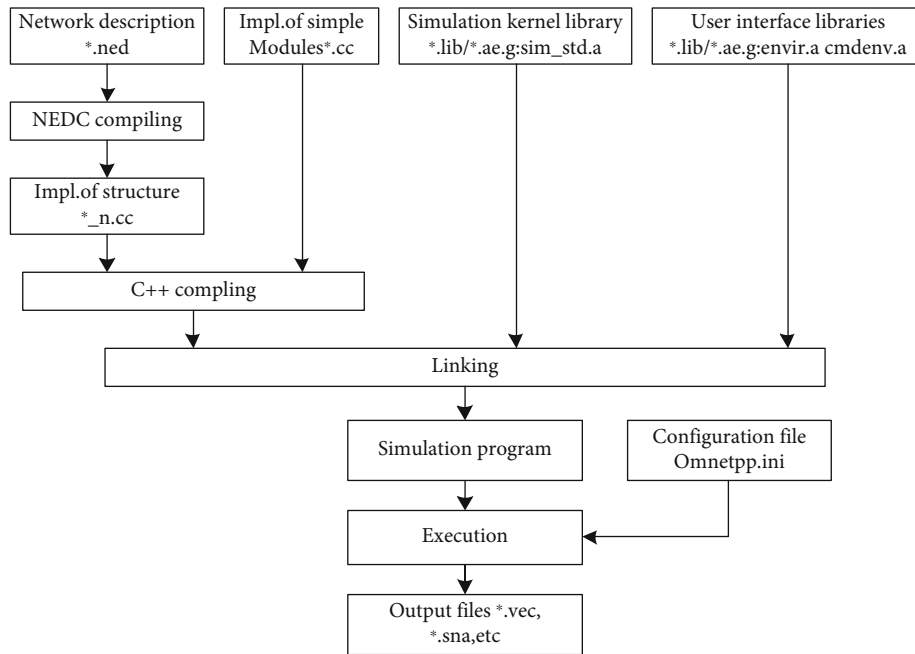


FIGURE 12: OMNET++ simulation program diagram.

DIST is M^* . When the anchor node receives the exact value of $RSSI$ sent by the unknown node, it converts it into a fuzzy number of the distance between the anchor node and the unknown node. As shown in Figure 10, when $RSSI$ is $-62dBm$, the membership function $\mu(RSSI)$, W and M have different degrees of membership, which are mapped to the fuzzy sets M and L of the distance through the corresponding fuzzy rules $Rule_i$ and $Rule_j$, respectively. Where P_i

and P_j are the centroid of the trapezoidal region formed after $RSSI$ is mapped to $DIST$.

For the value of $RSSI$, the range of its abstraction is narrowed by fuzzy rules, such as $-62dBm$ is in the ring 18 and 28 away from the anchor node $DIST$. Then compare whether 18 and 28 belong to the same fuzzy number of $DIST$, if not, take the fuzzy number with the larger membership degree. As shown in the figure, $-62dBm$ corresponds

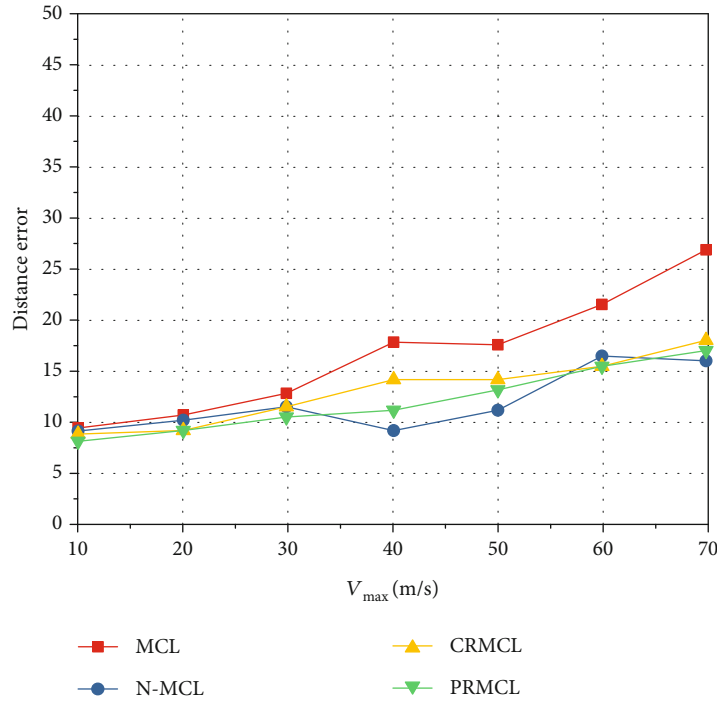


FIGURE 13: Influence of maximum speed on distance error.

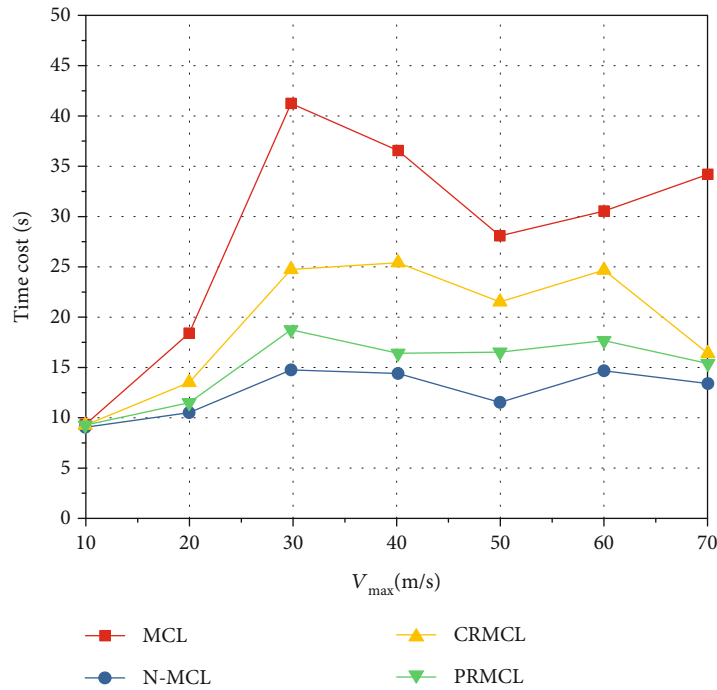
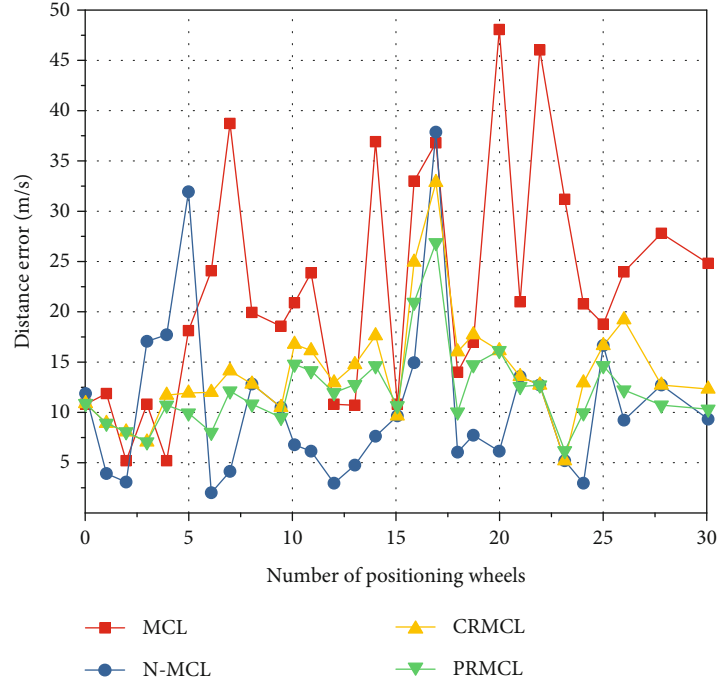


FIGURE 14: Influence of maximum speed on positioning time.

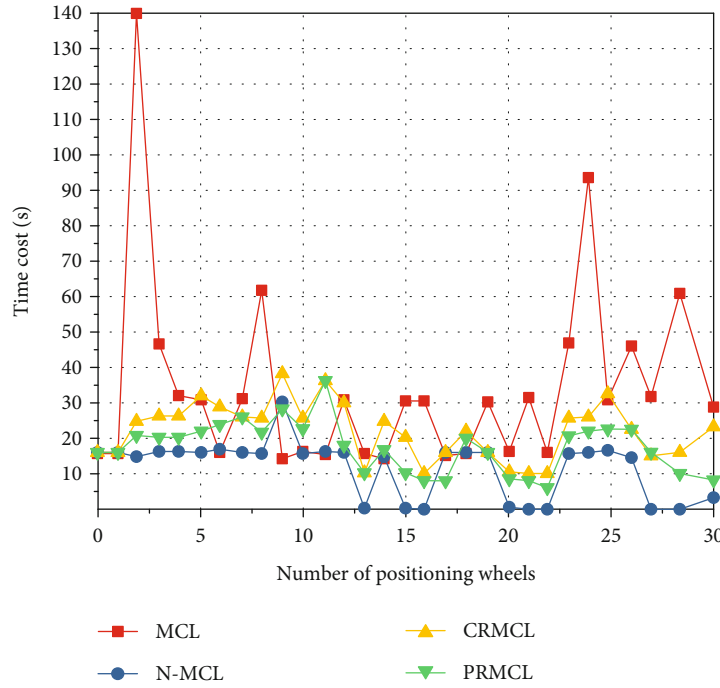
to the “L” of *DIST* in Anchor1. Then, the area composed of multiple one-hop anchor nodes is ABCD, and sampling is performed in this area, as shown in Figure 11.

(3) *Filtering Stage*. The limit condition of one-hop anchor node is to judge the distance between the node and the anchor node according to the corresponding fuzzy

number of *DIST*. For example, *DIST* with an anchor node is L , then the distance between the node and the anchor node is in $[1/3r, r]$, where r is the communication radius of the anchor node. In this way, the area where the particles are located is limited, so that the positioning time of the node is shorter, the accuracy is higher, and the algorithm is more complete.



(a) Positioning accuracy



(b) Positioning time

FIGURE 15: Simulation at time $v = 50\text{m/s}$.

5.3. Motion Model of Mobile Nodes. The mobile model is also an important factor affecting mobile node localization in WSN. In this paper, the random model of moving waypoint is used. Assume that the speed of the mobile node at time t is $v_{i(t)}$, the direction angle is $\theta_{i(t)}$, the coordinate is $(x_{i(t)}, y_{i(t)})$, and $X_{i(t)}$ is the time interval from time $t-1$ to time t . The position of the current node is estimated according to the

movement speed and state of the node at the last moment. The formula is expressed as follows:

$$\begin{cases} X_{i(t)} = X_{i(t-1)} + v_{i(t-1)} * \Delta T * \begin{bmatrix} \cos \theta_{i(i-1)} \\ \sin \theta_{i(i-1)} \end{bmatrix}, \\ v_{i(t)} = \omega v_{i(t-1)} + C_1 \text{ rand.} \end{cases} \quad (11)$$

Among them, w is the inertia weight, which can adjust the search ability of the algorithm, q is the acceleration constant, rand is the directional random motion setting of the node, and is a random number between $[0, 1]$.

The steps of the model are described as follows:

- (1) All nodes are randomly deployed in the wireless sensor network area, and then each node arbitrarily selects a point within the speed range as the destination
- (2) The node moves randomly and at a constant speed within the interval at any speed $[V_{\min}, V_{\max}]$ less than or equal to the maximum speed
- (3) After reaching the destination, the mobile node stops for T time, and then reselects the movement speed and destination, and starts to move

Repeat steps (2) and (3) until the simulation ends. The block diagram of the simulation program composed of OMNET++ is shown in Figure 12.

5.4. Simulation Results. The simulation experiment using OMNET++ (Objective Modular Network Tested in C++) and the sensor nodes are randomly placed in a square area of $500\text{m} \times 500\text{m}$, and there are no obstacles in the area. The irregularity of node communication is 0, and the communication radius of the nodes is 100 m. The maximum speed of the nodes and the anchor nodes are V_{\max} , the movement speed of the nodes and the anchor nodes are arbitrarily selected in $[0, V_{\max}]$, and the movement of the nodes adopts the Random Waypoint model. For the convenience of the experiment, the study sets a total of 20 nodes, including 18 anchor nodes and 2 nodes. The effect of the maximum speed of the node on the distance error is shown in Figure 13.

The speed of the node has a great influence on the positioning accuracy. First, the increase of the speed of the node leads to the increase of the distribution area of possible positions at the next moment. Secondly, the increase of the speed of the node also enables the node to detect more location information of the anchor node, filter out impossible positions, and assist positioning. Figure 13 shows that with the increasing of the speed of the node, the positioning error is increasing too. The distance error of the MCL algorithm, CRMCL algorithm, and PRMCL algorithm was generally increasing with the increasing of the speed; for the N-MCL algorithm, the distance error is minimized while the node speed is maximum at 40 m/s. When the speed of the node continues to increase and is greater than 50 m/s, due to the serious particle degradation problem, a sufficient number of samples cannot be collected, and the positioning accuracy decreases.

In addition to the positioning accuracy, the time required for positioning is also an important indicator to measure the quality of a positioning algorithm. Figure 14 reflects the influence of the maximum speed on the positioning time, indicating that the time spent by N-MCL on positioning is less than the other three algorithms. With the

increase of the node moving speed, the positioning time of MCL algorithm and CRMCL algorithm increases, while the PRMCL algorithm and N-MCL algorithm basically changed a little. When the maximum speed is 50 m/s, the time taken by the two algorithms reaches the minimum.

Experiments show that the N-MCL achieves the best localization effect when the speed is between 45-50 m/s. Figure 15 shows the simulation diagram of the 30-round positioning of the node when the speed is $v = 50\text{m/s}$, which is the distance error of each round of positioning and the required time. It can be seen from the figure that N-MCL algorithm is superior to MCL algorithm, CRMCL algorithm, and PRMCL algorithm in positioning accuracy and positioning time. According to statistics, the positioning time of N-MCL algorithm is shortened about 50% compared with the traditional MCL algorithm, and the positioning accuracy is improved about 30% compared with the traditional MCL algorithm.

6. Conclusion

This paper aims at the shortcomings of the MCL algorithm's poor positioning accuracy in nonideal environments, and the filtering stages were improved. In the prediction stage, the sampling point at the previous moment is used as the center of the circle for sampling, and the filtering stage adaptively changes the weight of the sampled particles in different regions according to different DOIs, thereby improving the positioning accuracy and network coverage of WSN nodes in nonideal environments. Finally, in view of the shortcomings of the Monte Carlo algorithm (MCL) in the mobile node positioning, the demand for samples is large, the positioning accuracy is not high, and the positioning time is long. An improved Monte Carlo mobile node positioning algorithm based on fuzzy theory is proposed. The algorithm improves the positioning accuracy and shortens the time required for positioning by introducing fuzzy rules in the sampling and filtering stages of nodes.

Data Availability

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest

The author declares that they have no conflicts of interest.

Acknowledgments

This work is supported in part by the Shaanxi Provincial Department of Education 2019 Special Scientific Research Plan Project under Grant 19JK0261, 2021 regular project of Shaanxi Provincial Sports Bureau under Grant 2021324, Special Project of Shangluo University Serving Local Scientific Research under Grant 19FK002.

References

- [1] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low-cost outdoor localization for very small devices," *IEEE Personal Communications*, vol. 7, no. 5, pp. 28–34, 2000.
- [2] S. Capkun, M. Hamdi, and J. P. Hubaux, "PS-free positioning in Mobile ad-hoc networks," *Cluster Computing*, vol. 5, no. 2, pp. 157–1167, 2002.
- [3] J. Li, X. Zhong, and C. Xu, "Review of dynamic node localization algorithm for wireless sensor networks," *Journal of North-east Dianli University*, vol. 35, no. 2, pp. 3–82, 2015.
- [4] C. Liu and L. Zhang, "Research on optimization of localization accuracy of wireless sensor network based on DV-hop algorithm," *Laser & Optoelectronics Progress*, vol. 12, no. 12, pp. 1–13, 2021.
- [5] J. Lu, M. Long, and K. Yin, "DV-HOP location algorithm based on weighted least squares optimization," *Chinese Journal of Sensors and Actuators*, vol. 33, no. 3, pp. 450–455, 2020.
- [6] X. Chen, P. Han, Q. He, and S. Tu, "A viable localization scheme for dynamic wireless sensor networks," in *First International Multi-Symposiums on Computer and Computational Science*, pp. 587–593, Hangzhou, China, 2006.
- [7] Y. L. Hsieh and K. Wang, "Efficient localization in mobile wireless sensor networks," in *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'06)*, vol. 1, pp. 292–297, Taichung, Taiwan, 2006.
- [8] L. Q. Zhang, X. B. Zhou, and Q. Cheng, "Landscape-3D: a robust localization scheme for sensor Networks over complex 3D terrains," in *31st IEEE Conference on Local Computer Networks*, pp. 239–246, Tampa, 2006.
- [9] E. Kim, S. Lee, and C. Kim, "Mobile beacon-based 3D-localization with multidimensional scaling in large sensor networks," *IEEE Communications Letters*, vol. 14, no. 7, pp. 641–649, 2010.
- [10] A. Baggio and K. Langendoen, "Monte Carlo localization for mobile wireless sensor networks," *Ad Hoc Networks*, vol. 6, no. 5, pp. 718–733, 2008.
- [11] M. Rudafshani and S. Datta, "Localization in wireless sensor networks," in *Proceedings of International Conference on Information Processing in Sensor Networks*, pp. 51–60, Piscataway, 2007.
- [12] H. Zhu, X. Zhong, and Q. Yu, "A localization algorithm for mobile wireless sensor networks," in *Proceedings of Third International Conference on Intelligent System De-sign and Engineering Applications*, pp. 81–85, Piscataway, 2013.
- [13] H. Tian, C. Li, J. Xie, and Y. Liang, "Node localization algorithm for WSN based on time sequence Monte Carlo," *Chinese Journal of Sensors and Actuators*, vol. 29, no. 11, pp. 1724–1730, 2016.
- [14] F. Zeng and X. Guo, "A Markov chain Monte Carlo node localization algorithm based on wireless sensors," *Journal of Hainan Tropical Ocean University*, vol. 26, no. 2, pp. 82–86, 2019.
- [15] J. Y. Lu and C. Wang, "A new Monte Carlo mobile node localization algorithm based on Newton interpolation," *Eurasip Journal on Wireless Communications & Networking*, vol. 2018, no. 1, pp. 1–8, 2018.
- [16] Z. Xu and Z. Shan, "A localization algorithm based on wavelet transform prediction for mobile nodes," *Chinese Journal of Sensors and Actuators*, vol. 29, no. 4, pp. 566–571, 2016.
- [17] M. Qin and R. Zhu, "A Monte Carlo localization method based on differential evolution optimization applied into economic forecasting in mobile wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 32, 9 pages, 2018.
- [18] S. Hao, Y. Yang, and Y. Dong, "Particle swarm and Monte Carlo optimized mobile localization algorithm in underwater acoustic sensor networks," *Acta Electronica Sinica*, vol. 49, no. 2, pp. 292–299, 2021.
- [19] L. Song, X. Jiang, and X. Huang, "Improved Monte Carlo localization algorithm based on temporary anchor nodes," *Huazhong University of Science & Technology (Natural Science Edition)*, vol. 49, no. 2, pp. 89–94, 2021.
- [20] G. Du, S. Zhang, and L. Qiu, "WSN node location algorithm based on motion trajectory capture and coupled orthogonal coverage," *Computer Engineering*, vol. 46, no. 3, pp. 157–162, 171, 2020.
- [21] L. Meng, H. Wang, and S. Xia, "TOA estimation method in frequency domain for acoustic ranging of WSN node," *Journal of Electronics & Information Technology*, vol. 32, no. 4, pp. 993–997, 2010.
- [22] Y. Li, "Hop-distance-corrected DV-HOP localization algorithm based on distribution of anchor nodes," *Chinese Journal of Sensors and Actuators*, vol. 34, no. 10, pp. 1379–1384, 2021.
- [23] J. Li, M. Gao, J. S. Pan, and S. C. Chu, "A parallel compact cat swarm optimization and its application in DV-hop node localization for wireless sensor network," *Wireless Networks*, vol. 27, no. 3, pp. 2081–2101, 2021.
- [24] L. Kang, M. Li, and J. Yang, "A high-precision positioning method for wireless sensor networks based on ultrasonic and TDOA," *Microcomputer Information*, vol. 26, no. 10, pp. 79–87, 2010.
- [25] P. Li, G. Chen, and W. Hu, "Research on wireless sensor network location based on chicken swarm optimization," *Chinese Journal of Sensors and Actuators*, vol. 32, no. 6, pp. 68–73, 93, 2019.
- [26] F. Ye, H. Shen, and L. Cao, "Self-adjusting sampling area MCL algorithm for Mobile WSNs," *Computer Science*, vol. 38, no. 12, pp. 49–60, 2011.