

Research Article

Mobile Industrial Robots Localization Algorithm Based on Improved Multidimensional Scale and Received Signal Strength Indication

Jian Jiao 

Chongqing Water Resources and Electeic Engineering College, Chongqing 402160, China

Correspondence should be addressed to Jian Jiao; jiaojian@st.btbu.edu.cn

Received 1 November 2021; Revised 17 November 2021; Accepted 30 November 2021; Published 15 December 2021

Academic Editor: Qiangyi Li

Copyright © 2021 Jian Jiao. 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.

Aiming at the problem of large error in the location algorithm based on MDS-MAP when the distance between mobile industrial robots is not measurable, a mobile industrial robot location algorithm based on improved MDS-MAP is proposed. Experimental simulation shows that the algorithm can achieve good positioning effect. When the distance between mobile industrial robots is measurable, the positioning algorithm based on RSSI achieves good positioning effect. Therefore, this paper discusses the influence of different anchor robot selection methods on the positioning accuracy of RSSI positioning algorithm. The experimental simulation shows that when the selection method of anchoring robot is that the unknown robot with adjacent anchoring robot uses the original anchoring robot for positioning and the unknown robot without anchoring robot uses the adjacent positioning robot as the anchoring robot for positioning, its positioning effect is the best, and it can still achieve good positioning effect when there are few anchoring robots.

1. Introduction

Industrial robot positioning refers to industrial robot determining its own position information through self-organizing network. It includes the position of the robot itself and the position of the external robot entering the network. Location technology is not only one of the research hotspots of wireless sensor networks but also an important supporting technology. It has wide application value in social life. At present, positioning technology is widely used in geographic navigation, target tracking, and other fields [1].

1.1. Ranging or Angle Positioning Algorithm and Nonranging Positioning Algorithm. The classification method is based on whether the robot has ranging function. In the distance based localization method, unknown robots in wireless sensor networks need to measure the distance (or angle) between them and each anchored robot. According to different ranging or angle technologies, such positioning

algorithms include positioning algorithm based on received signal strength indication (RSSI), positioning algorithm based on signal time of arrival (TOA), positioning algorithm based on signal time difference on arrival (TDOA), and positioning algorithm based on signal angle of arrival (AOA). This localization algorithm usually requires high-density anchoring robots in the network or the powerful computing and communication capabilities of anchoring robots. In addition to RSSI ranging technology, the other three positioning algorithms also need additional software and hardware to support their ranging and positioning [2].

The nonranging distance free localization algorithm does not require the sensor robot to have the ranging function and only uses the connectivity of the network to estimate its own position. At present, the commonly used ranging free localization algorithms can be divided into two categories: One is to estimate the distance between the unknown robot and the anchored robot and then use the trilateral measurement method or maximum likelihood estimation method for localization. The other is to determine the region

containing the unknown robot by anchoring the robot and adjacent robots and then calculate the centroid of the region as the coordinates of the unknown robot according to the centroid algorithm. Although the ranging free positioning accuracy is low, it can meet the positioning accuracy requirements of many applications. Typical range free localization algorithms include centroid localization, DV hop, amorphous, APIT, convex programming, and MDS-MAP algorithm [3].

1.2. Centralized and Distributed. Centralized location algorithm is a method to calculate the robot position through centralized calculation. Centralized computing needs to transmit the required information to a special central robot, where the robot positioning calculation is carried out. The advantages of this algorithm are that it can plan the network globally, the amounts of calculation and storage are hardly limited by the hardware conditions of the robot, and the positioning accuracy is high. The disadvantage is that it needs the technical support of the central computer, and the topology of the system is not good. When the number and position in the network change, it is necessary to recalculate the position of all robots, which will lead to the premature consumption of robot energy and the shortening of network life cycle. The distributed localization algorithm mainly depends on the information exchange and coordination between robots. After the anchor robot is set, the unknown robot close to the anchor robot can be positioned by the adjacent anchor robot, and the unknown robot far away from the anchor robot can be positioned by the adjacent positioning unknown robot as the anchor robot. In other words, each robot in the distributed positioning mode locates itself through self-calculation. This positioning method can balance the energy consumption of the network and minimize the requirements of the positioning algorithm for the number of anchored robots [4].

1.3. Absolute Positioning and Relative Positioning. The main difference between absolute positioning and relative positioning lies in different reference points. The former requires that the positioning algorithm can find the standard coordinate positions of unknown robots or foreign targets in the same reference object, while the relative positioning usually establishes the relative coordinate positions of the whole network with some robots in the network as references. In the process of the latter algorithm, the reference points are multiple sensor robots whose initial positions are known, and the reference points of each robot are different, so as to calculate the relative coordinates of the whole network robot. Compared with the two, absolute positioning can provide unique positioning results for the network, and the positioning accuracy is less affected by robot movement, so it has been widely used in many fields. However, in practical work, such as geographic cabling, as long as the relative position between robots is known, there is no need to know the absolute position of robots. Under certain conditions, the absolute positioning of network positioning can be transformed into relative positioning. Most location

algorithms can achieve absolute location. Typical relative positioning algorithms and systems include Spa (self-positioning algorithm) and LPS (local positioning system). In addition, the MDS-MAP positioning algorithm can also realize absolute positioning and relative positioning according to the actual situation [5].

1.4. Coarse and Fine. The coarse-grained and fine-grained classification of location algorithm is based on information granularity. Coarse-grained localization algorithm is a localization algorithm that measures the distance between robots according to the proximity to anchored robots; the fine-grained localization algorithm measures the distance between it and the anchored robot according to the signal strength or information transmission time, signal transmission model, and so forth. Generally speaking, most ranging-based localization algorithms are fine-grained localization algorithms, such as RSSI-based localization algorithms; the nonranging location algorithm is generally coarse-grained location algorithm, such as DV hop location algorithm. Its principle is to use various physical principles to determine the position of unknown robot [6].

2. Related Work

2.1. Ranging Method. In the research of wireless sensor network location technology, it is necessary to measure the distance between robots. Common ranging methods include RSSI, TOA, TDOA, and AOA.

2.1.1. RSSI (Received Signal Strength Indicator). RSSI positioning algorithm is a technology that measures the received power of the receiving robot, calculates the propagation loss, and then converts the propagation loss into distance using theoretical or empirical signal propagation model. This technology mainly uses RF signals. The power calculation is shown in the following formula:

$$PL(d) = PL(d_0) - 10n \lg\left(\frac{d}{d_0}\right) - X_\sigma. \quad (1)$$

In formula (1), d is the distance between the transmitter and the receiver; n is the channel attenuation index (generally 2–4); X_σ is a Gaussian random noise variable with mean value of 0 σ is the signal strength at the distance from the transmitter; $PL(d)$ is the emission intensity at distance d_0 from the transmitter. Generally, $PL(d_0)$ can be obtained from experience or defined by hardware specifications. Therefore, distance d can be obtained by solving the equation according to the signal strength $PL(d)$.

Because the sensor robot itself has the ability of wireless communication, it is a low-power and cheap ranging technology. This technology is used in many projects such as radar and spot-on. Its main error source is the modeling complexity of signal propagation model caused by environmental impact: reflection, multipath propagation, non-line-of-sight, antenna gain, and other problems will produce significantly different propagation losses at the same

distance. Moreover, the technology also needs to clear the network channel characteristics [7].

2.1.2. TOA (Time of Arrival). TOA technology calculates the distance between two robots by measuring the signal propagation time and propagation speed. The main error sources are multipath propagation, multiple access interference, and non-line-of-sight interference. The most basic positioning system using this technology is GPS. However, because the GPS system requires expensive and energy-consuming electronic equipment to accurately synchronize with the satellite clock and the hardware size, price, and power consumption of WSN robots are limited, GPS and other TOA technologies are reluctant to measure the distance of wireless sensor networks [8].

2.1.3. TDOA (Time Difference on Arrival). TDOA ranging technology determines the distance between robots according to the time difference of information arriving at robots. The transmitting robot sends two different signals: signal 1 and signal 2. The time to reach the receiving robot is T_1 and T_2 . The signal propagation rate is v_1 and v_2 . Then distance l between the two robots is calculated, as shown in the following formula:

$$l = (T_2 - T_1) \frac{v_1 v_2}{v_1 - v_2}. \quad (2)$$

This technology is widely used in the localization algorithm of wireless sensor networks. Wireless sensor networks based on TDOA technology are different from mobile station positioning and robot navigation positioning in cellular wireless networks. Generally, ultrasonic and RF transceivers are installed on the robot. The transmitting end transmits two signals at the same time. At the receiving end, the arrival time difference of two different signals (RF signal and ultrasonic signal) is recorded by using the difference between the propagation rates of ultrasonic and electromagnetic waves in the air. Based on the known signal propagation speed, time is directly converted into distance according to formula (2). At present, in the research of many positioning algorithms, ranging using TDOA has appeared in many positioning algorithms. Compared with RSSI technology, this technology has higher ranging accuracy and can usually reach the centimeter level. However, due to the limited propagation distance of ultrasonic (ultrasonic signal is 6–9 meters, so the network needs to be densely deployed) and NLOS (non-line-of-sight), although ultrasonic has been found to reduce the calculation and communication overhead, this technology is not necessarily suitable for wireless sensor networks requiring ultralow power consumption [9].

2.1.4. AOA (Angle of Arrival). One of the AOA technologies is to estimate the angle of the signal sent by the neighbor robot according to the interference of the wave. If the ultrasonic transmitter with interval of $2r$ is installed at the robot and transmits the ultrasonic with the same frequency and initial phase at the same time, the robot to be measured

will receive the interfering ultrasonic, and the included angle between the robot to be measured and the reference line is as shown in the following formula:

$$\theta = \arcsin\left(\frac{v}{2r f' T}\right) = \arcsin\left(\frac{v}{2r f' F}\right). \quad (3)$$

In formula (3), v is the ultrasonic propagation speed; T is the change period of the synthetic wave received at the time point P of signal change; f' is the change rate of ultrasonic wave, as shown in the following formula:

$$f' = \frac{(f_0 - f_1)}{t_0}. \quad (4)$$

In formula (4), f_0 is the initial frequency of the emitted ultrasonic wave; f_1 is the termination frequency; t_0 is the length of change time. Further, distance l between point P and point O can be obtained according to other known parameters [10].

This technology can be realized by antenna array or the combination of multiple receivers, which mainly provides the direction information of the robot. Similarly, the error sources of AOA positioning accuracy mainly include noise and NLOS problems. At the same time, AOA requires additional hardware and may not be used for sensor robots in terms of hardware size and power consumption [11].

3. Introduction to Our Robots Location Algorithm

According to whether the distance between robots needs to be measured in the positioning process of the system, the positioning technology is divided into ranging-based positioning algorithm and non-ranging-based positioning algorithm. The former needs to measure the signal reception angle information between the unknown robot and the anchor robot to calculate the position of the unknown robot; the latter depends on the connectivity of the network. At present, the commonly used ranging technologies include RSSI, TOA, RTOF, TDOA, and AOA.

The ranging-based localization algorithm is usually three-step localization: the first step is to measure the distance or angle from the unknown robot to the beacon robot; the second step is to calculate the coordinates of unknown robots by trilateral measurement, triangulation, or maximum likelihood estimation; and the third step is to correct the coordinates of the unknown robot by using the distance from the unknown robot to the neighbor robot [12].

3.1. MDS-MAP Algorithm. MDS-MAP location algorithm is a classical location algorithm based on MDS (multidimensional scaling) technology. The whole algorithm process can be divided into the following:

- Step 1.* Calculate the shortest distance between robots in all detection areas to build the distance matrix of MDS
- Step 2.* According to the distance matrix, the robot uses the obtained shortest path data to calculate the relative coordinate map

Step 3. The coordinate information of the known anchor robot is used to convert the relative coordinates of the unknown robot into absolute coordinates [13]

3.2. DV Hop Algorithm. DV hop algorithm is the most widely used of the six positioning algorithms contained in ad hoc positioning system (APS). APS is a series of distributed positioning algorithms proposed by Rutgers University based on distance vector routing and GPS positioning technology, including DV hop, DV distance, DV Euclidean, DV coordinate, DV bearing, and DV red. DV hop algorithm is not based on ranging but uses multihop anchor robot information to realize robot location [14].

Firstly, the algorithm uses the typical distance vector exchange protocol to count the hop distance between all robots and anchor robots according to the information transmission between robots. Then, the anchor robot averages the distance per hop through the technical network and broadcasts the hop correction value, and the robot estimates the Euclidean distance to the nearby anchor robot according to the hop number and hop distance. Finally, the location of unknown robots is determined by trilateral measurement or multilateral measurement [15].

3.3. APIT Positioning Algorithm. APIT algorithm (approximate point in triangulation test) was proposed by T. He, C. Huang, and B. M. Blum of the University of Virginia based on APIT theory. Test according to different combinations of anchor robots until the combination with the best accuracy is obtained in all combinations. All triangles will overlap a certain area, and the unknown robots must be in the overlapping area [16].

APIT algorithm has good performance under the condition of irregular propagation signal mode and random deployment. However, the algorithm needs to build enough APIT triangles to meet the requirements of positioning accuracy. Therefore, its accuracy depends on the number of reference robots that can be communicated [17].

3.4. Centroid Location Algorithm. In the centroid algorithm [18], according to the connectivity of the network, the unknown robot determines the geometry of the surrounding anchor robots and then directly solves the polygon composed of the anchor robots to solve the centroid and determine the coordinate position of the unknown robot.

If the vector of the fixed-point position of the polygon composed of anchor robots is $p_i = (x_i, y_i)^T$, the centroid calculation formula is as follows:

$$(X_{\text{est}}, Y_{\text{est}}) = \left(\frac{1}{n} \sum_{i=1}^n X_i, \frac{1}{n} \sum_{i=1}^n Y_i \right). \quad (5)$$

3.5. Amorphous Location Algorithm. Similar to DV hop algorithm, amorphous algorithm represents the shortest distance between two points according to the hop distance between robots. The process of amorphous location

algorithm can be divided into three stages. The first stage is calculating the minimum hops between the unknown robot and each anchor robot. The second stage is taking the communication radius as the average distance per hop, so as to calculate the distance from the unknown robot to the anchor robot. In the third stage, the location of unknown robots is calculated by trilateral measurement method or maximum likelihood estimation method [19].

In the traditional amorphous algorithm, the positioning accuracy error is often large when the communication radius is used as the average distance per hop. Therefore, the specific stages of the algorithm are as follows.

3.5.1. Calculating the Minimum Number of Hops. By broadcasting packet information, the anchor robot enables the unknown robot to obtain the location coordinates of each anchor robot and calculate the minimum number of hops between itself and each anchor robot. In the traditional amorphous algorithm, the minimum number of hops is an integer, and the test shows that about 0.5 average hops are increased. Therefore, the unknown robot can use formula (6) to calculate the local hops instead of the hops between it and each anchor robot:

$$S(i, k) = \frac{\sum_{j \in \text{nbrs}(i)} h(j, k) + h(i, k)}{|\text{nbrs}(i)| + 1}. \quad (6)$$

In formula (6), $\text{nbrs}(i)$ is the set of adjacent neighbor robots of the unknown robot, $|\text{nbrs}(i)|$ is the number of adjacent robots of robot i , $h(i, k)$ represents the number of hops between robot i and the anchor robot k , and $h(j, k)$ represents the number of hops between the neighbor robot j and the anchor robot k [20].

The algorithm needs to be carried out when the average connectivity of the network is known, and the average distance per hop is required to be calculated before network deployment, which leads to poor network scalability and high density of anchor robots [21].

3.5.2. Calculating the Average Distance per Hop. In the case of random deployment of robots, the minimum hops between the unknown robot and each anchor robot multiplied by the average distance per hop are often greater than the Euclidean distance between two points, and if the average distance per hop is set as the communication radius of the robot, the result is greater than the distance per hop. The average distance per hop is calculated as shown in the following formula:

$$\text{HopSize} = r \left(1 + e^{-n_{\text{local}}} - \int_{-1}^1 e^{-(n_{\text{local}}/\pi)ar \cos t - t\sqrt{1-t^2}} dt \right), \quad (7)$$

$$\text{Dis}(i, j) = \text{Hopsize}(i) \times S(i, j), \quad (8)$$

where r is the robot communication radius, n_{local} is the average connectivity of the network, and t is the average number of adjacent robots of the network [22].

3.5.3. *Calculating the Estimated Location of Unknown Robots.* Calculate the distance between the unknown robot and each anchor robot according to formulas (6) to (8), as shown in the following formula:

$$\text{Dis}(i, j) = \text{Hopsize}(i) \times S(i, j). \quad (9)$$

According to the measured distance, when the number of adjacent anchor robots of unknown robots is greater than or equal to 3 and meets the positioning requirements, the positioning calculation can be carried out according to the trilateral measurement method or maximum likelihood method [23].

3.6. *Robots Location Algorithm Based on MDS-MAP.* The location algorithm based on MDS technology is a hot issue in the research of location technology in wireless sensor networks. MDS technology can be applied to the case where the robot distance is range based or range free. At present, in the range based MDS positioning algorithm, a distance self-adjusting MDS positioning algorithm (SA-MDS) is proposed. The algorithm uses three methods to estimate the two-hop distance of robots. Under the same positioning requirements, the robots and anchor robots of SA-MDS algorithm are better than MDS. In order to speed up the calculation and reduce the computational complexity, a fast MDS map algorithm is proposed. Simulation experiments also show that the algorithm can have the same accuracy as MDS-MAP algorithm under the set error but significantly reduces the running time of the algorithm in the original MDS. On the basis of map algorithm, the improved iterative location algorithm combined with centroid algorithm is used to replace the classical MDS solution, which overcomes the disadvantage of complex calculation caused by the huge MDS matrix [24].

Multidimensional calibration is essentially a technology to transform the analysis object or degree of dissimilarity from high-dimensional space to low-dimensional space. By fitting the distance between points in low-dimensional space, the relative coordinates of each point in low-dimensional space are obtained, and its proximity is determined by determining the threat coefficient. The calculation of distance matrix in MDS algorithm can calculate the distance of each robot in the network based on measurable and unmeasurable distance, respectively [25].

In MDS technology, ξ_{ij} is assumed to represent the dissimilarity between entities i and j from the dissimilarity matrix $[\xi_{ij}]_p = [\xi_{ij}]_m * mX = [x_1, x_2, \dots, x_m]^T B = V\Lambda V^T$ in MDS technology, which is generally expressed by Euclidean distance; assume that the true Euclidean distance between entities i and j is d_{ij} . Its goal is to reconstruct the point coordinate matrix $X_{n,m}$ corresponding to these objects in multidimensional space by using the dissimilarity matrix $p = [\xi_{ij}]$ composed of Euclidean distances between entities. The number of coordinate points is n and the dimension of coordinate points is m , so as to minimize the threat coefficient; that is, the proximity is the closest, as shown in the following formula:

$$\text{STRESS} = \min \sum_{ij} [f(\xi_{ij}) - d_{ij}(X)]. \quad (10)$$

In the process of the algorithm, the dissimilarity matrix P is constructed first, and the power is recorded as $P^2 = [d_{ij}^2]_{n*m}$.

Then double $P^2(X)$ is centered. The so-called double centralization of the matrix is to successively subtract the row mean value of the row and the column mean value of the column from each element of the target matrix and then add the overall mean value of the matrix elements and multiply it by the factor -0.5 . That is, both sides of matrix $P^2(X)$ are multiplied by the center matrix J to obtain the double centered distance matrix B . The central matrix is shown in formula (11), and the matrix after double centralization is shown in formula (12):

$$J = I - e \times \frac{e^T}{n}. \quad (11)$$

In formula (11), I is the identity matrix of dimension $m * m$ and e is the identity vector of length m .

$$B = -\frac{1}{2}JP^2J. \quad (12)$$

The coordinates of robots are $X = [x_1, x_2, \dots, x_m]^T$, and the matrix can be expressed as follows:

$$B = XX^T. \quad (13)$$

Therefore, singular value decomposition B will be performed to obtain the following formula:

$$B = Q\Lambda Q^T. \quad (14)$$

In formula (14), Λ is a diagonal matrix composed of eigenvalues, Q is an orthogonal matrix composed of eigenvectors corresponding to eigenvalues, and its column vectors are eigenvectors. Therefore, the maximum m positive eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_w$ and w corresponding eigenvectors e_1, e_2, \dots, e_w can be obtained. The w eigenvalues form a w dimensional diagonal matrix Λ , and the w eigenvectors form an $n \times w$ dimensional matrix Q . The relative coordinates of all points can be obtained according to formulas (13) and (14), as in the following formula:

$$X = Q \cdot \Lambda^{1/2}. \quad (15)$$

For example, for three points in two-dimensional space, the corresponding matrix in the coordinate system is

$$X = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ x_{31} & x_{32} \end{bmatrix}. \quad (16)$$

The distance square matrix is represented by $P^2(X)$. Then it can be expressed as

$$P^2(X) = \begin{bmatrix} 0 & d_{12}^2 & d_{13}^2 \\ d_{21}^2 & 0 & d_{23}^2 \\ d_{31}^2 & d_{32}^2 & 0 \end{bmatrix}. \quad (17)$$

The square of the Euclidean distance between them is shown in formulas (12) and (13).

$$\begin{aligned}
 d_{ij}^2(X) &= d_{ij}^2 = \sum_{a=1}^m (x_{ia} - x_{ja})^2 \\
 &= \sum_{a=1}^m (x_{ia}^2 + x_{ja}^2 - 2x_{ia}x_{ja}), \\
 P^2(X) &= \sum_{a=1}^2 x_{ia}^2 + \sum_{a=1}^2 x_{ja}^2 - 2 \sum_{a=1}^2 x_{ia}x_{ja} \\
 &= \begin{bmatrix} \sum_{a=1}^2 x_{1a}^2 \\ \sum_{a=1}^2 x_{2a}^2 \\ \sum_{a=1}^2 x_{3a}^2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\
 &+ \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} \sum_{a=1}^2 x_{1a}^2 & \sum_{a=1}^2 x_{2a}^2 & \sum_{a=1}^2 x_{3a}^2 \end{bmatrix} - 2 \sum_{a=1}^2 x_a x_a^T \\
 &= c1^T + 1c^T - 2 \sum_{a=1}^2 x_a x_a^T.
 \end{aligned} \tag{18}$$

4. Simulation Results

MDS-MAP algorithm based on multidimensional calibration applied in WSN can run in range based and range free method according to the communication information between adjacent robots. This algorithm studies the improvement and simulation of location algorithm based on range free method. The MDS-MAP algorithm can be divided into the three following steps:

Step 1. Calculate the hop distance between robots in the network.

Step 2. In the traditional MDS-MAP algorithm, the shortest path between any two points is generally calculated according to the shortest path algorithm (Dijkstra or Floyd), and then the shortest hop distance between each point is calculated. Then, taking the hops between robots as the distance matrix in MDS, in order to reduce the superposition of errors in the calculation of hops between unknown robots, the average distance per hop between robots is estimated according to the hops between anchor robots and the actual distance between anchor robots, and then the distance between robots is calculated according to the hop matrix constructed between robots to form the distance matrix. The calculation formula of average per jump of anchor robot is shown in the following formula:

$$\text{HopSize} = \frac{\sum |X_i - X_j|}{\sum \text{hop}(i, j)} \tag{19}$$

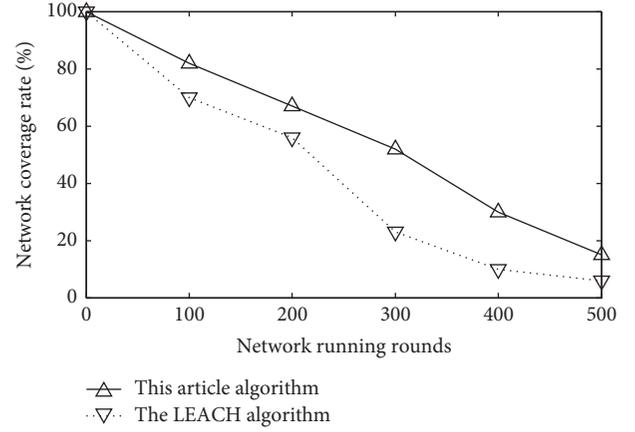


FIGURE 1: Network coverage rate comparison in this article's algorithm and the LEACH algorithm under different network running rounds (when $s = 0.05$).

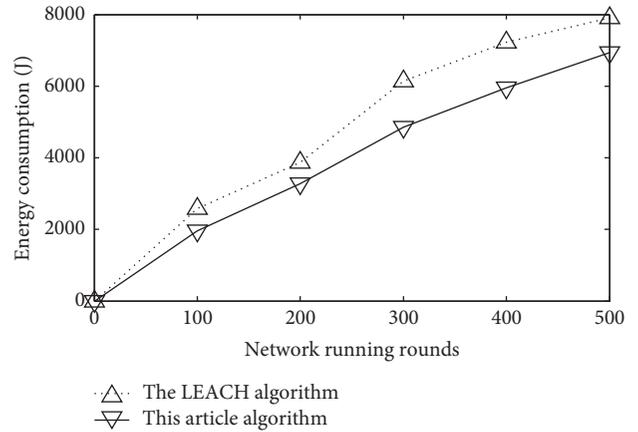


FIGURE 2: Energy consumption comparison in this article's algorithm and the LEACH algorithm under different network running rounds (when $s = 0.1$).

Step 3. From the distance matrix from the above, according to formula (7), the matrix is double-centralized to obtain matrix B .

Step 4. Singular value decomposition is performed on B to obtain the diagonal matrix composed of its eigenvalues and the orthogonal matrix composed of its eigenvectors.

Step 5. According to formula (10), find x , which is the relative coordinate of the sensor robot.

Step 6. Convert relative coordinates into absolute coordinates.

Step 7. Calculate error.

MATLAB software is used for simulation. 1000 robots are randomly deployed at $1000 \text{ m} \times 1000 \text{ m}$ rectangular plane area. The number of robots that can locate themselves $n = 1000 \times s$.

In the same network environment, running this article's algorithm and LEACH algorithm, respectively, the simulation results are shown in Figures 1 to 12.

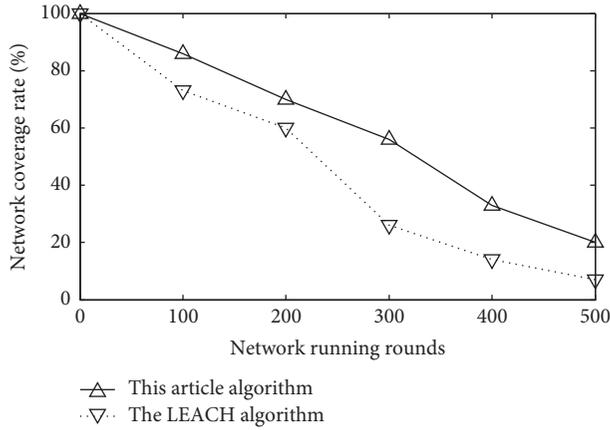


FIGURE 3: Network coverage rate comparison in this article's algorithm and the LEACH algorithm under different network running rounds.

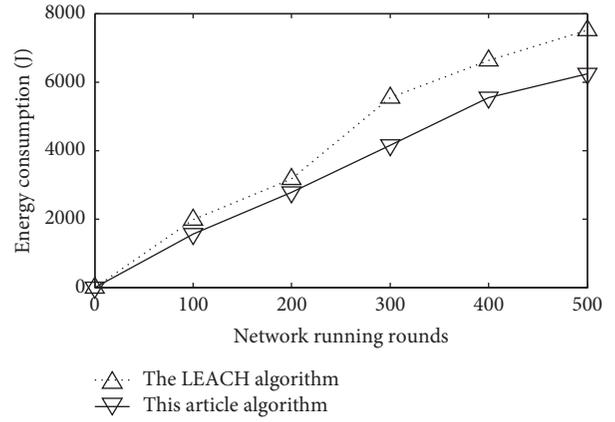


FIGURE 6: Energy consumption comparison in this article's algorithm and the LEACH algorithm under different network running rounds (when $s = 0.2$).

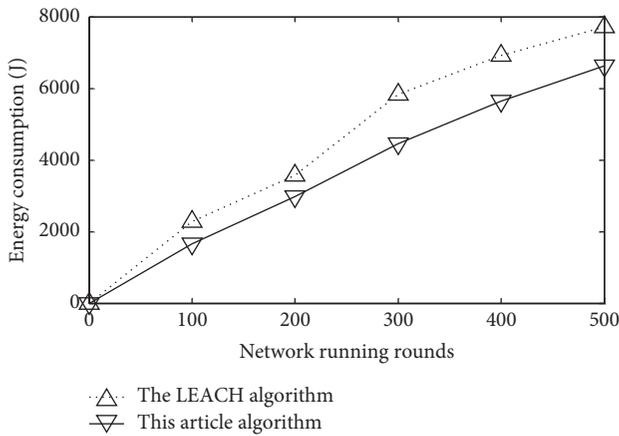


FIGURE 4: Energy consumption comparison in this article's algorithm and the LEACH algorithm under different network running rounds (when $s = 0.15$).

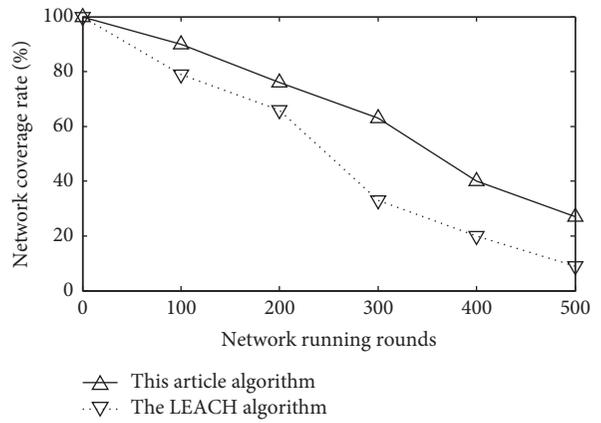


FIGURE 7: Network coverage rate comparison in this article's algorithm and the LEACH algorithm under different network running rounds.

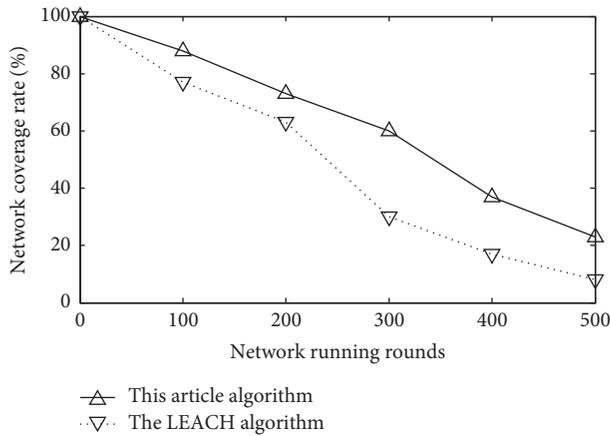


FIGURE 5: Network coverage rate comparison in this article's algorithm and the LEACH algorithm under different network running rounds.

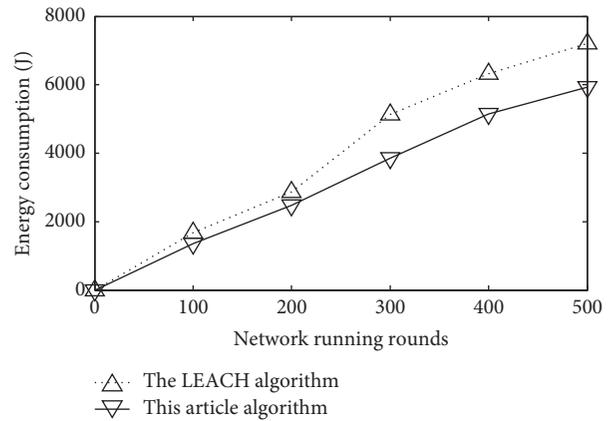


FIGURE 8: Energy consumption comparison in this article's algorithm and the LEACH algorithm under different network running rounds (when $s = 0.25$).

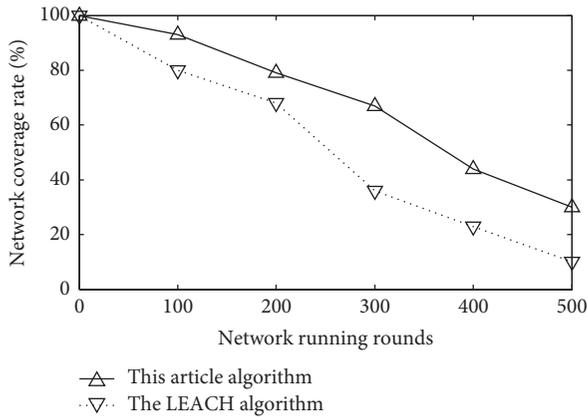


FIGURE 9: Network coverage rate comparison in this article's algorithm and the LEACH algorithm under different network running rounds.

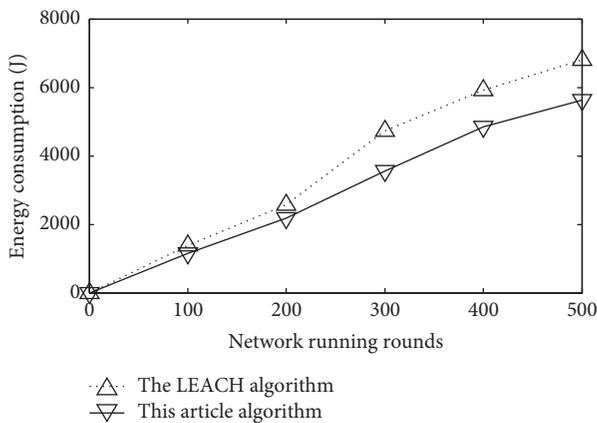


FIGURE 10: Energy consumption comparison in this article's algorithm and the LEACH algorithm under different network running rounds (when $s = 0.3$).

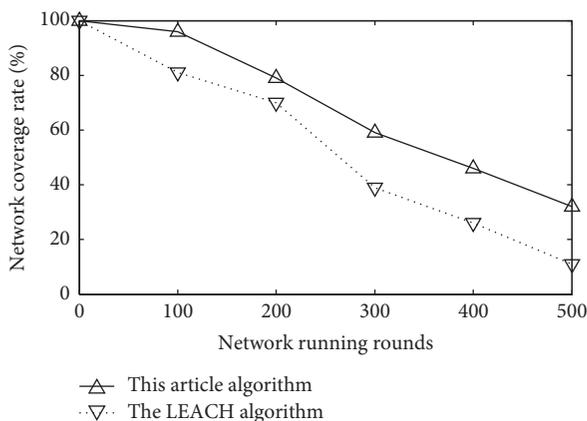


FIGURE 11: Network coverage rate comparison in this article's algorithm and the LEACH algorithm under different network running rounds.

With s increasing, the wireless sensor network coverage rate increases, and energy consumption of the robots reduces. This article's algorithm is better than the LEACH

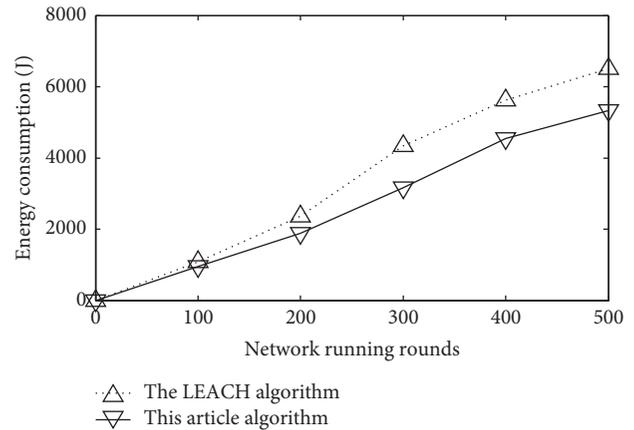


FIGURE 12: Energy consumption comparison in this article's algorithm and the LEACH algorithm under different network running rounds.

algorithm in covering effect of the network and energy consumption of the robots.

5. Conclusion

Firstly, this paper briefly summarizes the localization algorithms in sensor networks, including ranging algorithm, ranging-based localization algorithm, and non-ranging-based localization algorithm. Finally, based on the non-ranging MDS-MAP algorithm, an improved MDS-MAP algorithm is proposed. The simulation results show that the influence of communication radius on positioning accuracy, average connectivity, and average number of anchor robots is included. Aiming at the commonly used RSSI ranging methods, a positioning algorithm based on anchor robot selection is proposed. Through the selection of three kinds of anchoring robots, the simulation shows that a small number of anchoring robots can also achieve good positioning effect.

Data Availability

The labeled datasets used to support the findings of this study are available upon request to the author.

Conflicts of Interest

The author declares no conflicts of interest.

Acknowledgments

This study was sponsored by the Natural Science Foundation of Chongqing, China (no. cstc2021jcyj-msxmX1113).

References

- [1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [2] J. Zhang, X. Hu, Z. Ning et al., "Energy-latency tradeoff for energy-aware offloading in mobile edge computing

- networks,” *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2633–2645, 2018.
- [3] Y. Zou and K. Chakrabarty, “Sensor deployment and target localization in distributed sensor networks,” *ACM Transactions on Embedded Computing Systems*, vol. 3, no. 1, pp. 61–91, 2004.
- [4] W.-H. Liao, Y. Kao, and Y.-S. Li, “A sensor deployment approach using glowworm swarm optimization algorithm in wireless sensor networks,” *Expert Systems with Applications*, vol. 38, no. 10, pp. 12180–12188, 2011.
- [5] Z. Ning, X. Hu, Z. Chen et al., “A cooperative quality-aware service access system for social internet of vehicles,” *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2506–2517, 2018.
- [6] D. Demin Wang, B. Bin Xie, and D. P. Agrawal, “Coverage and lifetime optimization of wireless sensor networks with Gaussian distribution,” *IEEE Transactions on Mobile Computing*, vol. 7, no. 12, pp. 1444–1458, 2008.
- [7] Yi Shang, R. Wheeler, and Y. Zhang, “Location from mere connectivity 201-212,” *Journal of Network and Computer Applications*, vol. 3, pp. 652–659, 2009.
- [8] Y. Shang, W. Rumi, Y. Zhang, and M. Fromherz, “Localization from connectivity in sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, no. 11, pp. 961–974, 2004.
- [9] X. Wang, S. Yuan, R. Laur, and W. Lang, “Dynamic localization based on spatial reasoning with RSSI in wireless sensor networks for transport logistics,” *Sensors and Actuators A: Physical*, vol. 171, no. 2, pp. 421–428, 2011.
- [10] B. Hu, G.-P. Gao, L.-L. He, X.-D. Cong, and J.-N. Zhao, “Bending and on-arm effects on a wearable antenna for 2.45 GHz body area network,” *IEEE Antennas and Wireless Propagation Letters*, vol. 15, pp. 378–381, 2016.
- [11] C. Long, H. Lin, C. Huang, and S. Liao, “Improved LEACH multi-hop algorithm based on residual energy and region distribution,” *Journal of Information and Computational Science*, vol. 11, no. 9, pp. 2955–2964, 2014.
- [12] H. Peng, B. Hu, Q. Shi et al., “Removal of ocular artifacts in EEG—an improved approach combining DWT and ANC for portable Applications,” *IEEE journal of biomedical and health informatics*, vol. 17, no. 3, pp. 600–607, 2013.
- [13] C. Long, X. Zhou, S. Liao, and N. Zhang, “An improved LEACH multi-hop routing protocol based on genetic algorithms for heterogeneous wireless sensor networks,” *Journal of Information and Computational Science*, vol. 11, no. 2, pp. 415–424, 2014.
- [14] H. Zhang, B. Ge, and C. Liu, “Improved simulated annealing algorithm for broadcast routing of wireless sensor network,” *Journal of Computational Information Systems*, vol. 9, no. 6, pp. 2303–2310, 2013.
- [15] J. Xu and D. Qin, “A new LEACH-based routing clustering protocol in WSN,” *Journal of Information and Computational Science*, vol. 10, no. 18, pp. 6005–6011, 2013.
- [16] H. Jing, “Coverage holes recovery algorithm based on nodes balance distance of underwater wireless sensor network,” *International Journal on Smart Sensing and Intelligent Systems*, vol. 7, no. 4, pp. 1890–1907, 2014.
- [17] X. Hu, J. Cheng, M. Zhou et al., “Emotion-Aware cognitive system in multi-channel cognitive radio ad hoc networks,” *IEEE Communications Magazine*, vol. 56, no. 4, pp. 180–187, 2018.
- [18] L. Xu, C. Li, and J. Yang, “Multi-objective strategy of multiple coverage in heterogeneous sensor networks,” *Journal of Electronics and Information Technology*, vol. 36, no. 3, pp. 692–695, 2014.
- [19] D. Xiao-yu, L. Sun, J. Guo, and C. Han, “Coverage optimization algorithm for heterogeneous WSNs,” *Journal of Electronics and Information Technology*, vol. 36, no. 3, pp. 696–702, 2014.
- [20] J. Xiong, J. Ren, L. Chen et al., “Enhancing privacy and availability for data clustering in intelligent electrical service of IoT,” *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1530–1540, 2019.
- [21] J. Xiong, R. Bi, M. Zhao, J. Guo, and Q. Yang, “Edge-assisted privacy-preserving raw data sharing framework for connected autonomous vehicles,” *IEEE Wireless Communications*, vol. 27, no. 3, pp. 24–30, 2020.
- [22] J. Xiong, X. Chen, Q. Yang, L. Chen, and Z. Yao, “A task-oriented user selection incentive mechanism in edge-aided mobile crowd sensing,” *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2347–2360, 2020.
- [23] J. Xiong, R. Ma, L. Chen et al., “A personalized privacy protection framework for mobile crowd sensing in IIoT,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4231–4241, 2020.
- [24] J. Xiong, M. Zhao, M. Z. A. Bhuiyan, L. Chen, and Y. Tian, “An AI-enabled three-party game framework for guaranteed data privacy in mobile edge crowd sensing of IoT,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 2, pp. 922–933, 2021.
- [25] Y. Tian, Z. Wang, J. Xiong, and J. Ma, “A blockchain-based secure key management scheme with trustworthiness in DWSNs,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6193–6202, 2020.