The Application Effect of Improved CS-RBF Neural Network in Industrial Internet of Things Node Localization

Tingting Liu[1*], Zhenhan Zhang[2]
School of Electronic Information Engineering, Henan Polytechnic Institute, Nanyang 473000, China
E-mail: liuting2023hn@163.com

Keywords: radial basis function neural network, cuckoo algorithm, internet of things, node prediction

Received: April 10, 2024

Node localization technology can help industrial Internet of Things systems control production processes more accurately, monitor product quality in real time, and handle problems timely, thereby improving product quality. It is significant in improving the efficiency and safety of industrial production. However, existing node localization technologies have low accuracy in node localization in complex environments, which cannot effectively ensure localization effectiveness and directly affect device monitoring in industrial Internet of Things. To address this issue, a node localization method is constructed based on the radial basis function neural network function. Then the cuckoo algorithm is applied to optimize it. From the results, the improved CS-RBF model had a smaller change in absolute error value, with a minimum value of -9.3 and an average value of 0.8. When the proportion of beacon nodes was 35%, the average localization errors of DE, APTT, and DV Hop were 0.27, 0.23, and 0.22, respectively. The node localization error of CS-RBF was 0.18. This indicates that the designed node localization method can effectively achieve precise localization of the required nodes, thereby effectively scheduling and managing equipment, reducing equipment waiting time, and improving production efficiency.

Povzetek: Predstavljen je izboljšan nevronski mrežni model za lokalizacijo vozlišč v industrijskem internetu stvari (IIoT). Metoda združuje radialno bazično funkcijo (RBF) z optimizacijo s pomočjo kukavičjega algoritma, kar izboljšuje točnost lokalizacije vozlišč v kompleksnih okoljih. Rezultati kažejo, da je napaka pri lokalizaciji vozlišč pri uporabi CS-RBF metode bistveno manjša, s čemer je omogočeno učinkovito upravljanje opreme, zmanjševanje čakalnega časa naprav ter učinkovitosti proizvodnje.

1 Introduction

The Industrial Internet of Things (IIoT) continuously integrates various collection and control sensors or controllers with sensing and monitoring capabilities, as well as mobile communication, and other technologies into industrial production processes. This improves manufacturing efficiency, enhance product quality, and reduce cost, ultimately achieving the transformation and upgrading of traditional industries [1-2]. In the implementation of IIoT, Wireless Sensor Networks (WSN) are composed of rich micro sensor nodes deployed in monitoring areas. Therefore, for WSN, data collection, processing, and transmission without node information is meaningless. Common methods such as target monitoring and tracking, intelligent transportation, and modern logistics management all provide their own location information through nodes [3]. Node localization technology is one of the key technologies for implementing IIoT applications, which plays an important role in improving the performance and operational efficiency of IIoT systems. Node localization technology can be used to determine the position of terminal devices, thereby providing users with accurate geographic location information and monitoring devices and resources in the IIoT [4-5]. With the continuous development of WSNs, reliable and efficient node localization technology is an urgent problem to be solved. In the current node localization technology, the localization accuracy needs to be improved, especially in complex environments, where problems such as occlusion and wireless signal attenuation seriously affect the localization accuracy. Therefore, a node localization technology based on the Radial Basis Function Neural Network (RBFNN) is constructed for the IIoT. In response to the shortcomings of RBFNN, the Cuckoo Search (CS) is applied to optimize it. An Cuckoo Search-Radical Basis Function (CS-RBF) is constructed. This study innovatively combines RBF network with CS algorithm to design a localization method for unknown nodes in the IIoT. In response to the shortcomings of the CS method in the search, its initial population is optimized to obtain better optimization results.

The study consists of four parts. The first part summarizes the current research status of RBF, CS, and node localization technology both domestically and internationally. The second part constructs an IIoT node localization method based on CS-RBF. The third part validates the performance. The fourth part summarizes the research content and points out future research directions.

2 Related works

RBFNN is an artificial neural network that uses RBF as activation functions. It can effectively overcome the characteristics of local minima, simple training, and fast convergence speed. It is widely used in fields such as time series prediction, function approximation, classification, and system control. Yang et al. used a fuzzy RBFNN to construct a multi-objective classification algorithm. This method considered three types of clustering techniques to effectively generate membership degrees. The proposed method had significant superiority, and the calculation accuracy reached 92.02%, but its computational complexity was relatively high [6]. Tian et al. developed a new data-driven model based on RBFNN. Firstly, the regularization was used to overcome the structural risks and over fitting problems, improving its complex data processing ability. Meanwhile, the improved whale optimization was applied to optimize the RBF kernel. The results indicated that the proposed model had high accuracy, verifying the feasibility and effectiveness. Its accuracy was 0.86, but the large structure of this method led to a significant increase in computational complexity [7]. Wang and Chen explored the predictive ability of quantitative structure-activity relationship models and joint optimization methods based on RBFNN to study the acute toxicity mechanism of Pimegales promelas. Then a potential acute toxicity prediction model was developed. The robustness and external prediction ability were also better than existing methods. This model demonstrated good performance, R2=0.91, and ROC=0.86. However, this method had relatively weak detection ability for outliers [8]. Pazouki et al. proposed a hybrid model that combined RBFNN and Firefly Optimization Algorithm (FOA) to predict concrete compressive strength. The input parameters included the age of the sample and the dosage of cement, fine aggregate, and high-efficiency water reducing agent. The output was the concrete compressive strength. The results indicated that the proposed model estimated compressive strength with better accuracy. The accuracy of this method reached 94.57%, but the process of parameter optimization was relatively complex [9].

CS is an emerging heuristic optimization algorithm that has been extensively studied in fields such as path planning, wireless sensor layout, big data optimization, and large-scale optimization. Shadkam and Bijari proposed an improved CS that can find the optimal solution, which had good performance. The improved method was superior to the original algorithm, and its efficiency was 82%. However, the data processing effect of this method was insufficient [10]. Ma et al. improved the optimization scheduling scheme of micro-grids using the CS. To analyze the impact of multi-objective economic environment scheduling in micro-grids, an adaptive taxation flight strategy was proposed. It improved the accuracy, stability, and scheduling convergence speed. The AUC of this method reached 0.85 and the accuracy was 84%. However, when the data size was large, the operational efficiency of this method was limited [11]. Zhang et al. proposed an improved CS to solve the soil structure parameter estimation. This algorithm extended the search range of soil model parameters. The results showed that the proposed algorithm performed well in minimizing errors. Its accuracy and recall were 086 and 0.88, respectively. However, the computational complexity of this method was relatively high [12]. Oruc et al. created a new fuel flow model for the descent phase of flight using the CS. A new fuel flow model for the B737-80 aircraft model was developed. The created model could accurately predict fuel flow based on the altitude and true airspeed. This model was useful in air traffic management decision support systems. Compared with existing methods, its prediction accuracy was increased by 8.59%, but this method still had scalability issues [13].

The research on IIoT nodes is of great significance in improving production efficiency, enhancing equipment reliability, enhancing safety performance, promoting intelligent transformation, and industrial upgrading. Li proposed a source node location privacy protection algorithm based on virtual routing in the IIoT. It effectively resisted attacks from strong visual attackers and strengthened the privacy protection. Simulation experiments showed that this algorithm had good privacy protection effects, which was superior to several comparative algorithms. The accuracy of this method was 88.97%, but it was prone to imbalance when selecting nodes [14]. Kassab et al. discussed the information centric form of wireless access in a multi cell wireless access network model. The optimal detector based on the model was introduced. The results showed that the access method designed in the study had better efficiency, which was 18.06% higher than existing methods. However, this method had cold start problem [15]. Sah et al. used heuristic algorithms to schedule the incoming traffic of backbone nodes. It provided a unique interference example for successfully transmitting control and data packets. The accuracy of data transmission was 91.32%, but this method consumed a high amount of energy during the calculation process [16]. The relevant work has been summarized, and the specific content is shown in Table 1.

Table 1: Literature summary table

Author(s)	Key methods and algorithms	Performance metrics	Limitations.
Yang et al. [6]	A multi-objective classification method based on fuzzy RBFNN	The calculation accuracy has reached 92.02%.	Its computational complexity is relatively high
Tian et al. [7]	A RBF-based data-driven model	Its accuracy is 0.86.	The structure of this method is relatively large, resulting in a significant increase in computational complexity.
Wang and Chen [8]	Quantitative structure-activity relationship model based on RBFNN	R2=0.91, and ROC=0.86	This method has relatively weak detection ability for outliers.
Pazouki et al. [9]	A hybrid prediction model based on RBF and FOA for concrete strength prediction	The accuracy reaches 94.57%.	The process of parameter optimization is relatively complex.
Shadkam and Bijari [10]	Determine the optimal solution based on improved CS algorithm	Its efficiency is 82%.	The data processing effect of this method is insufficient.
Ma et al. [11]	Optimization scheduling method for microgrids based on CS algorithm	AUC reaches 0.85, Accuracy is 84%.	When the data size is large, the efficiency of this method is limited.
Zhang et al. [12]	A parameter estimation method based on improved CS soil results	The accuracy and recall rates are 086 and 0.88, respectively.	This method has a high computational complexity.
Oruc et al. [13]	Aircraft fuel estimation model based on CS algorithm	Its prediction accuracy has increased by 8.59%	There are still scalability issues with this method.
Li [14]	A source node location privacy protection algorithm based on virtual routing in the Internet of Things environment	The accuracy of this method is 88.97%.	This method is prone to imbalance when selecting nodes.
Kassab et al. [15]	A wireless access method centered on edge and cloud detection information	It is 18.06% higher than existing methods.	This method has a cold start issue.
Sah et al. [16]	A backbone node incoming traffic scheduling method based on heuristic algorithm	The data transmission accuracy is 91.32%	This method consumes a high amount of energy during the calculation process.

In summary, node localization technology has received more research, but existing node localization mostly determines the position of known nodes. There is relatively insufficient research on predictive localization for unknown nodes. At the same time, in response to the high computational consumption and insufficient output processing ability in existing research, this study aims to leverage the advantages of RBFNN in data prediction to construct corresponding node prediction and localization techniques. Then it is optimized using CS algorithm. It is expected that this method can calculate unknown node positions, reduce computational complexity, and better achieve node localization.

3 Construction of node localization model based on improved CS-RBF

With the large-scale application of IIoT, sensor networks have gained favor in various fields. Sensors with location information are crucial for specific data acquisition and analysis. Sensor nodes have high randomness during deployment, making it difficult to predict the node position, obtain location information with greater difficulty, and incur higher cost. Based on the distance calculation between nodes, an improved RBFNN is used to construct node localization technology in the IIoT.

3.1 Calculation of node historical information

As a key component of the IoT, WSNs are widely used for information collection, real-time monitoring, etc. Sensors achieve information collection, data transmission, and other functions through information sharing, which are the main equipment for obtaining data information in the IoT [17]. Sensor networks typically collect information from sensor nodes and transmit it to the control center, as shown in Figure 1.

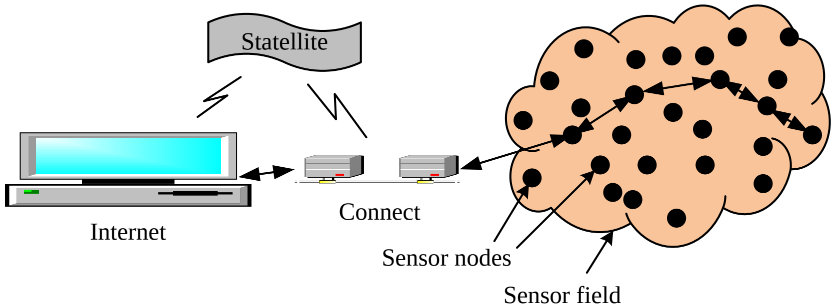


Figure 1: Basic structure of sensor network

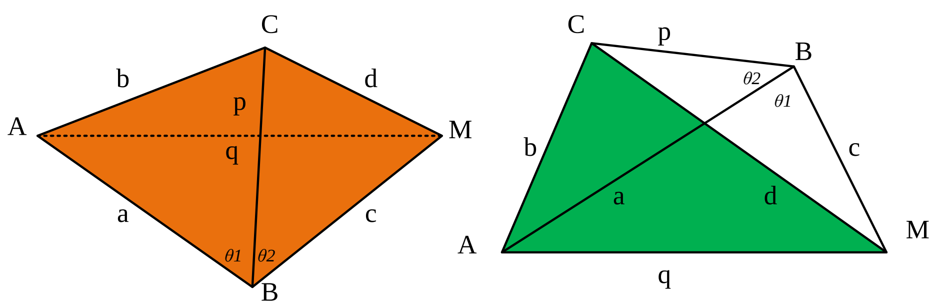
In the WSN, the location of sensor nodes is the key to achieving information collection and transmission. The information data without node location information is meaningless. Due to the influence of practical application environments, the position of sensor nodes is unknown or constantly changing. Therefore, constructing corresponding node localization techniques to locate the node positions in sensors is of great significance. Before locating nodes, firstly, the distance between nodes needs to be calculated to obtain the relative coordinates of each node. Taking the distance calculation between nodes A and M as an example, nodes A and M are on both sides of BC. Nodes B and C are neighboring nodes that can communicate directly. Therefore, the Euclidean distance between neighboring nodes can be calculated, as shown in formula (1).

$$\begin{cases} Distance_{AC} = b \\ Distance_{CM} = d \\ Distance_{AM} = q \\ Distance_{CB} = p \\ Distance_{BM} = c \end{cases} \tag{1}$$

According to formula (1), the distance calculation method between nodes A and D is shown in formula (2).

$$\begin{cases} b^2 = a^2 + p^2 - 2ap \times \cos\theta_1 \\ b^2 = c^2 + p^2 - 2cp \times \cos\theta_2 \\ q^2 = a^2 + c^2 - 2ac \times \cos(\theta_1 + \theta_2) \end{cases} \tag{2}$$

The positions of nodes A and M are shown in Figure 2.



(a) Node AM on both sides of BC (b) Node AM on the same side

Figure 2: Schematic diagram of the positions of node A and node M

After solving formula (2), the distance solutions between A and D are obtained. Based on the interference conditions between sensor nodes and practical application requirements, the corresponding interference solutions are eliminated to obtain the distance between nodes A and D. The above situation is the distance between nodes calculated with the help of auxiliary nodes. However, in practical applications, there are significant errors between the calculated node distances due to hardware conditions and external interference factors. Therefore, the study uses an error function to correct the errors in this calculation [18]. Firstly, the error function is defined, as shown in formula (3).

$$E(u)=\sum_{i=1}^{m}(\min(|u-q_i|),|u-q_i^{'}|))^2 \quad (3)$$

In formula (3), m represents the number of error estimates. q represents the distance between nodes A and D. Then, the estimated value of m is solved using the weighting method. Generally speaking, a higher density between nodes leads to higher accuracy in distance calculation. Therefore, solutions with higher node density are assigned larger weights. The node density is represented as C. The weighted solution is shown in formula (4).

$$q=\frac{\sum_{i=1}^{m}q_i(C_{ui}+C_{vi})/(m-1)}{\sum_{i=1}^{m}C_i} \quad (4)$$

In formula (4), C_{ui} and C_{vi} respectively represent the node connectivity of the two auxiliary nodes of the quadrilateral formed by node D. The practical application of nodes is complex and diverse. Not all nodes can form geometric relationships with their neighboring nodes during the calculation process. Therefore, for nodes that do not satisfy geometric relationships, other nodes that can satisfy quadrilateral relationships are used to supplement the quadrilateral and further estimate the node distance. If nodes A and M cannot form a quadrilateral relationship, the distance between the two is shown in formula (5).

$$Distance_{AM}=\mu D_s \quad (5)$$

In formula (5), μ represents the correction coefficient between the shortest path distance and the Euclidean distance in nodes A and M, with a value range of (0,1). The size of μ is positively correlated with the connectivity of each node that passes through the shortest path between nodes A and M. D_s represents the shortest path size between nodes A and M, as shown in formula (6).

$$D_s=d_1+d_2+...+d_z \quad (6)$$

In formula (6), d_z represents the size of each hop distance between nodes A and M. Then, the error correction method is used to correct the distance between the calculated AM to obtain a relatively accurate distance between nodes A and M.

3.2 Construction of node localization model based on improved CS-RBF

Node localization prediction in WSNs is to calculate the position of unknown or target nodes based on known node information. With the progress of modern science and technology, artificial neural networks have been widely applied and developed in fields such as hospitals, industrial control, computer science, and intelligent recognition. The RBFNN has strong nonlinear approximation ability and fast training speed. Therefore, the study adopts RBFNN to construct an unknown node localization model for sensor networks. RBFNN is a three-layer feed-forward network model composed of Input Layer (IL), Hidden Layer (HL), and Output Layer (OL) [19]. When the input sample approaches the center distance of the basis function, the HL nodes are activated, producing a larger output. The weight between the IL and the HL is 1. The IL maps the output vector to the HL. The linear transformation process occurs between the HL and the OL. The final output is the weighted sum of of all neuron outputs in the HL. The activation function in RBFNN adopts Gaussian basis function, as shown in formula (7).

$$h_j=\exp\left(-\frac{\|x-c_j\|^2}{2b_j^2}\right)(j=1,2,...,m) \quad (7)$$

In formula (7), b_j represents the node width of the Gaussian basis function. x is the input variable. c_j represents the center vector of the j-th node in the network. The calculated historical node information is used as the input sample to calculate the nonlinear mapping relationship between the corresponding node positions at the fitting time point. The output of the RBFNN obtained is displayed in formula (8).

$$Y(X_n)=\sum_{i=1}^{n}\omega_i\varphi(X_n,k_i)+b_0 \quad (8)$$

In formula (8), X_n represents the set of training samples for the network. $Y(X_n)$ represents the movement position of the node. ω_i represents the weight between the i-th HL unit and the OL unit. b_0 represents the offset of the output node. $\varphi(X_n,k_i)$ represents the basis function. k_i is the center of the basis function. In this calculation process, RBFNN encounter difficulties in center selection and are prone to falling into local optima during training. When constructing RNFNN model, different parameter selection methods have different impacts on model performance. The random selection center method can effectively determine network parameters. However, when the data sample is too large, over-fitting may occur and increase computational complexity. Although orthogonal least squares method can effectively improve the generalization ability of the model, it may not be able to design an RBFNN with the optimal structure. In addition, the self-organizing selection center method can also increase the complexity of the algorithm when facing large amounts of data. With the development of current technology, intelligent optimization algorithms have provided new research directions for neural network parameter optimization. Based on this, the study adopts intelligent optimization algorithms to select the parameters of RBFNN. The intelligent optimization algorithm does not rely on gradient information during the optimization process. It only adjusts the search direction of the objective function based on the fitness function value, which has better performance. Therefore, the study adopts the CS to optimize it. The CS-RBF node localization method is constructed for the IoT, improving the performance of the RBFNN. CS is a new type of meta-heuristic optimization method that searches for the optimal solution in the solution space by simulating the breeding strategy of cuckoo birds [20]. The reproduction of CS is shown in Figure 3.

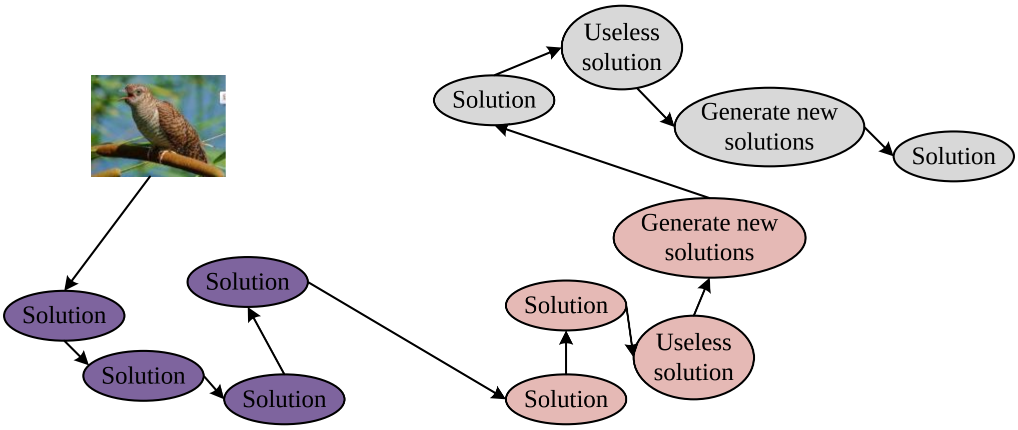


Figure 3: Schematic diagram of cuckoo bird reproduction

The CS has advantages such as fewer parameters and strong optimization ability, which is widely used in parameter optimization. During the flight, its flight trajectory y exhibits typical characteristics of Levy flight. $Levy(\lambda)$ represents the flight path, which is a typical random walk strategy. Its step size follows the Levy distribution, as displayed in formula (9).

$$Levy(s) \sim |s|^{-1-\lambda} \qquad (0 < \lambda < 2) \tag{9}$$

In formula (9), s represents the random step size of Levy flight. To generate a random step size that follows this distribution, Mantegna is used in the study, as shown in formula (10).

$$s = \mu \big/ |\upsilon|^{1/\beta} \qquad (1 \le a \le 3) \tag{10}$$

In formula (10), μ and u represent following a standard normal distribution. The position update of the CS during the optimization process is shown in formula (11).

$$C_t^{t+1} = C_i^t + \alpha \oplus Levy(\lambda) \tag{11}$$

In formula (11), C_i^{t+1} refers to the position of the i-th individual in the t-th generation. α is the step size scaling factor used to control the search range of the step size, as shown in formula (12).

$$\alpha = \alpha_0 \times (C_i^t - C_{best}) \tag{12}$$

In formula (12), C_i^{t+1} represents the current optimal position. The standard CS algorithm has relatively weak search ability and search speed during operation. Firstly, the initial population of the CS algorithm is optimized. The initial population of the standard CS algorithm is randomly generated in the solution space, requiring more time to search for the optimal solution during optimization calculations. Therefore, the initial population optimization is to make the initial population randomly and uniformly distributed in the solution space. The parameter values that need to be optimized are divided into small intervals. Individuals are randomly generated in each interval. The sum of all individuals is the total population. Through this method, the solution can be uniformly distributed. In the CS algorithm, if the population is N, and the solution space of a parameter is $[W, R]$, then the solution space of the initialized population is shown in formula (13).

$$S_p = \left[W + \frac{R-W}{N} i, W + \frac{R-W}{N}(i+1) \right] \tag{13}$$

After optimizing the initial population, the study adopts an elimination mechanism to optimize the CS algorithm, with the aim of increasing population diversity. Specifically, after completing the local search, all bird nests in the solution space are sorted according to their fitness size. Then S bird nests with the worst fitness are eliminated. A corresponding number of bird nests are generated at the current optimal bird nest position to update the position, as shown in formula (14).

$$C_i^t = C_{best} + randn\left(\frac{t_{\max} - t}{t_{\max}}\right)^{\theta} \tag{14}$$

In formula (14), $C_t^{t+1} = C_i^t + \alpha \oplus Levy(\lambda)$ refers to a normal distribution with a mean of 0 and a variance of 1, with the aim of keeping the newly generated position near the optimal nest position. θ represents the regulatory factor. The improved CS algorithm is shown in Figure 4.

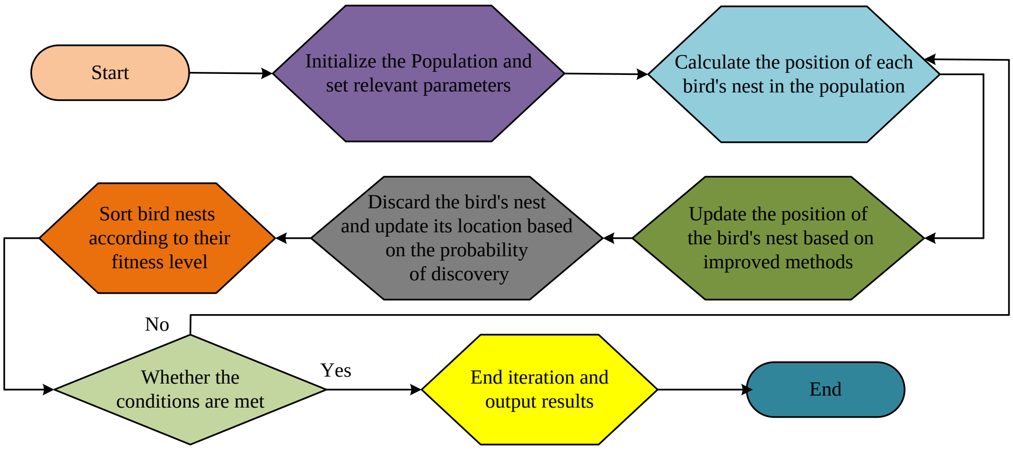


Figure 4: Improved CS algorithm process

The above optimization strategy is used to adjust the exploration and development capabilities of the CS algorithm. The algorithm can converge more quickly and effectively, with higher accuracy. In the application of the IoT, the spatial linear transformation ability of the IL to OL in the RBFNN is used to determine the mapping relationship with node positions. Next are the implementation steps. Firstly, the calculated historical node information in the IIoT is input into the RBFNN as initial data. Then the parameters are initialized to calculate the fitness function value of the CS algorithm. According to the fitness function value of the current CS and the optimal position of the bird nest, the CS algorithm is optimized. After multiple iterations, the optimal position is obtained. The specific implementation is shown in Figure 5.

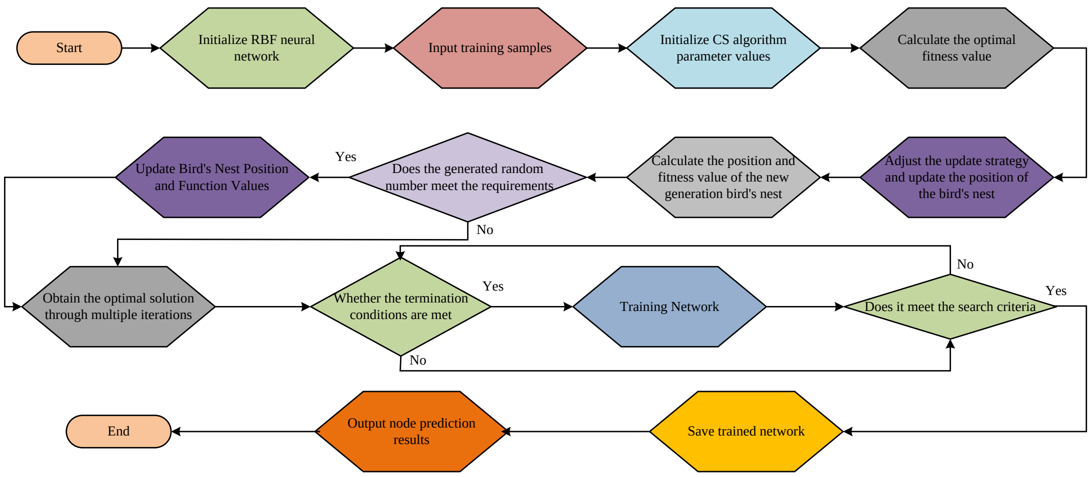


Figure 5: Implementation process of CS-RBF

In the CS-RBF, in RBFNN, the spatial linear transformation ability from IL to OL is used to determine the mapping relationship with node positions. The specific implementation process is as follows. Firstly, the CS algorithm is initialized, including the initial population, maximum iteration number, maximum discovery probability, etc. Then, the individuals in the CS algorithm are encoded. Their variance, weight, etc. are used for RBFNN training to calculate the initial fitness value of the individuals. Next, while retaining the optimal individual, the individual positions are updated to calculate the discovery probability of all individuals. Based on the calculation results, the fitness values of the individual positions before and after are compared. If the termination condition is met, the optimal parameters are obtained and assigned to RBFNN. Otherwise, the initial fitness value of the individual is recalculated and subsequent steps are repeated. Based on the above steps, the parameter optimization of RBFNN is completed. The implementation process of the entire research method is shown in Figure 6.

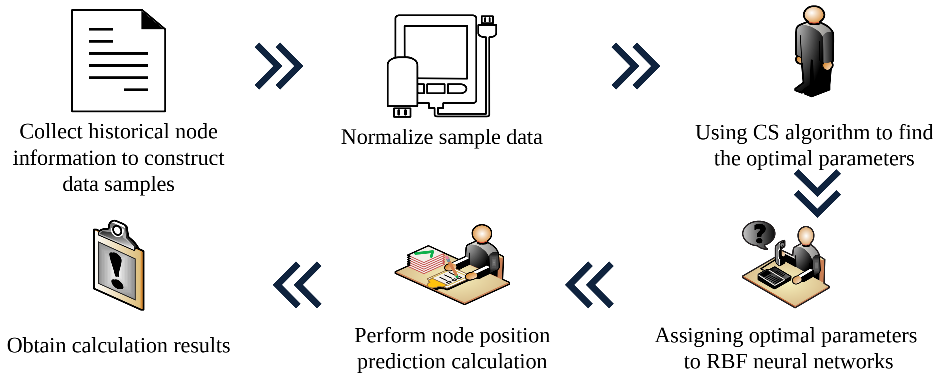


Figure 6: Implementation process of research method

In summary, in the application of the IoT, the implementation process of the research method is as follows. Firstly, historical node information in the relevant IIoT is collected to construct data samples. The collected samples are normalized to construct corresponding training and testing sets. Then the CS is initialized to optimize RBFNN. The optimal parameters obtained from the CS algorithm are assigned to the RBFNN. Finally, the CS-RBF model is used to predict and calculate the nodes.

4 Performance analysis of node localization model based on improved CS-RBF

To verify the actual performance of the proposed CS-RBF node localization method, corresponding experiments are designed to verify its application effectiveness. Firstly, the optimization performance of the CS before and after improvement is analyzed. Then the node localization performance of the CS-RBF model is verified.

4.1 Performance analysis of CS-RBF model

The testing environment for the experiment is a Windows 10 system, with an Inter (R) Core (TM) i5-3230M CPU of 2.60GHz. All experiments are completed in MATLAB2016a. The input parameters for network results are as follows. The input node is 5, and the HL nodes are 20. The population size of the CS is 50, with a minimum discovery probability of 0.2, and a maximum discovery probability of 0.5. The maximum iterations are 500. Firstly, the performance of the CS method is tested. Four test functions are selected as basis functions to test the improved CS algorithm, namely J.D. Schaffer function (f1), Sphere function (f2), Schwefel's Problem function (f3), and Generalized Griewank function (f4). The dimensions of the four test functions are 2, 10, 10, and 30, and the theoretical optimal values are all 0. The search ranges are [-5, 5], [-100, 100], [-100, 100], and [-600, 600], respectively. The fitness values of the four test functions are displayed in Figure 7. In Figure 7 (a), the final convergence effect of CS and the improved SC was basically the same, but the initial value of CS convergence was relatively high. In Figure 7 (b), the improved SC converged after 40 iterations, while CS gradually converged after 100 iterations. In Figure 7 (c), the improved CS method showed over-fitting and relatively weak convergence performance. In Figure 7 (d), the improved SC gradually converged after 60 iterations. The iterations of CS exceeded the improved method. Overall, the improved CS algorithm has better convergence performance and accuracy than the standard CS algorithm. The obtained optimization results are also better than the original CS algorithm, with higher accuracy performance.

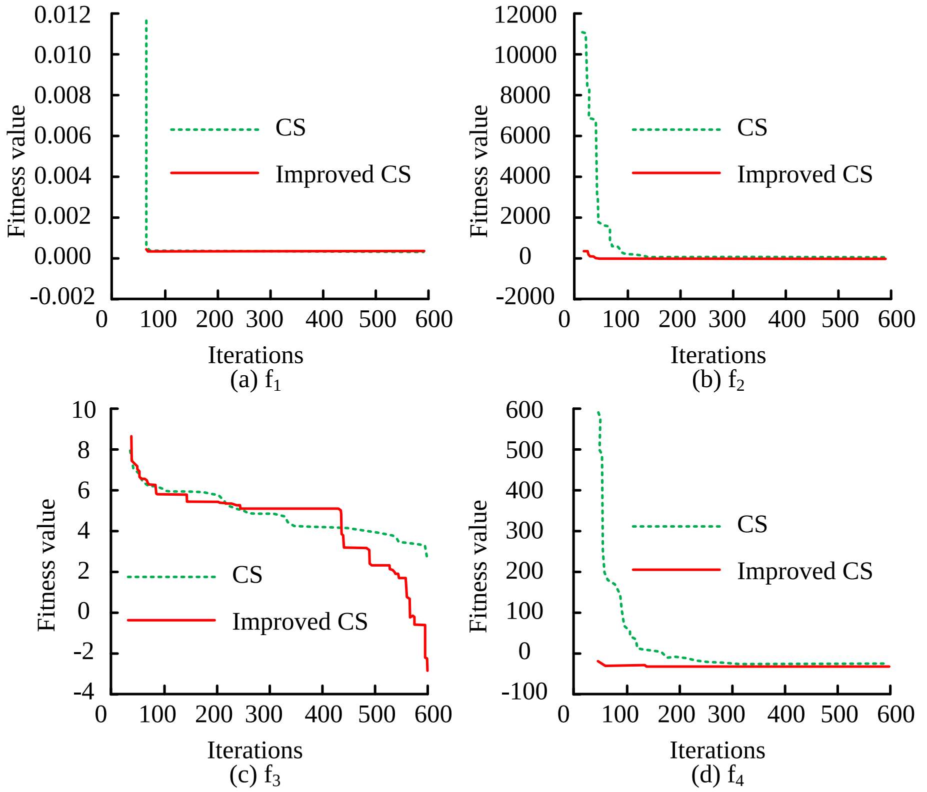


Figure 7: Performance comparison of CS algorithm before and after improvement in different test functions

To verify the optimization effect of the CS, the prediction performance of the CS-RBF before and after improvement is compared. Figure 8 displays the results. In Figure 8 (a), there was a significant difference in the potential values between the improved CS-RBF method and the RBFNN method. The improved CS algorithm used in the study had higher fitness, meeting its various changing trends. In Figure 8 (b), the absolute error value of the improved CS-RBF model changed less, with a minimum absolute error value of -9.3, and an average absolute error value of 0.8. The maximum absolute error value of the RBFNN was 17, and the average absolute error value was 2.9. The difference between the two is significant, indicating that the improved CS can effectively optimize the RBFNN.

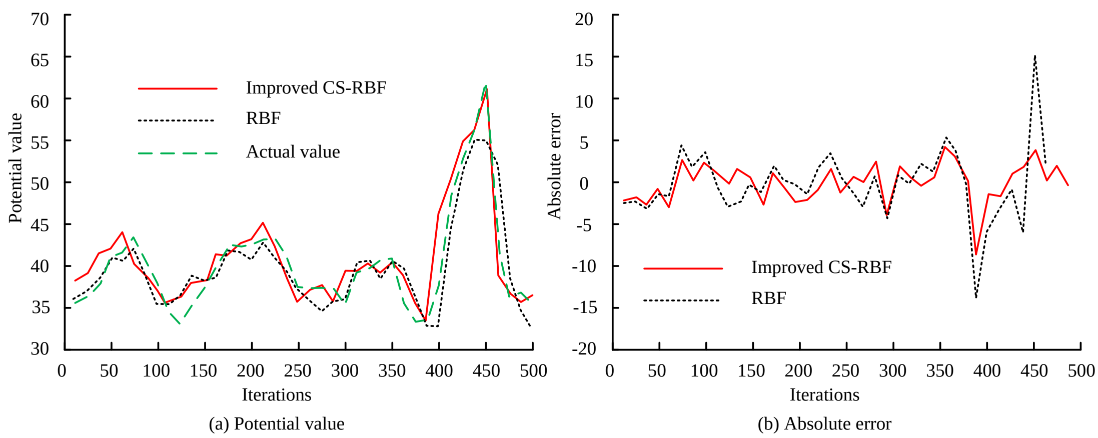


Figure 8: Comparison of predictive performance before and after RBF improvement

To verify the feasibility of the proposed method in WSN, simulation experiments are conducted to analyze it. Meanwhile, different node localization methods are compared to verify the feasibility of the designed method, including Differential Evolution (DE), Approximate Point-in-Triangulation Test (APTT), and Distance Vector Hop (DV-Hop). In the experimental area, 50 sensor nodes are randomly arranged and numbered as 1, 2, 50. Firstly, the visualization process of the nodes is presented, as shown in Figure 9.

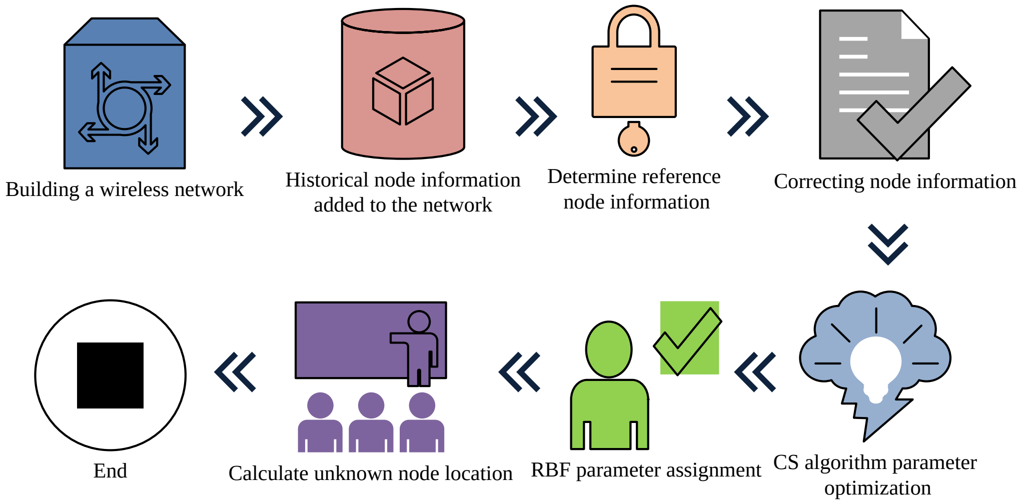


Figure 9: Visualization process of node localization in industrial Internet of Things environment

Based on the above visualization process, the performance of the research method is analyzed. Firstly, in the C-Shaped Random network, the node localization efficiency of the CS-RBF algorithm is analyzed. Figure 10 displays the results. In Figure 10, the time consumption of DE, APTT, DV Hop, and CS-RBF methods after completing sample node localization sampling was 5.6s, 4.3s, 3.2s, and 2.4s, respectively. The CS-RBF method has the lowest time cost after completing all node localization. The node localization performance is better.

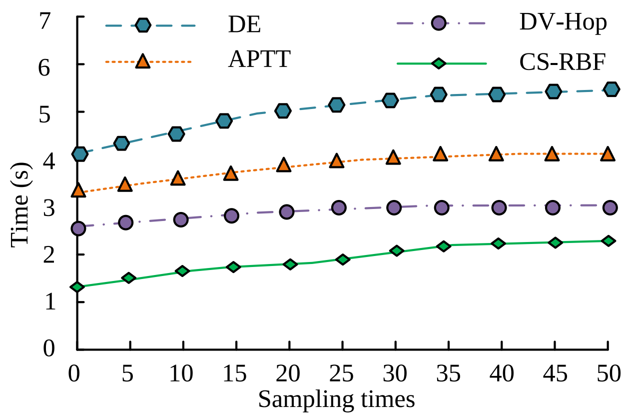


Figure 10: Comparison of node localization efficiency in the C-Shaped Random network

Next, the error performance of the designed method is analyzed. The localization errors of the four methods obtained are shown in Figure 11. Among them, Figure 11 (a) represents the average localization error rate. Figure 11 (b) represents the average error distance. In Figure 11 (a), the average localization error rate of DE, APTT, and DV Hop was 0.69, 0.52, and 0.33, respectively. The average localization error of CS-RBF in all nodes was 0.18. The node localization error of this method is significantly lower than the other three commonly used methods. Figure 11 (b) shows that the error distance difference in actual localization is relatively large. Among them, the maximum error distance of DE was 9.7m, the APTT was 8.9m, the DV-Hop was 10m, and the CS-RBF was 5.1m. Each node localization method has multiple extremes, but the proposed CS-RBF method has the smallest error distance fluctuation. The node localization results are the most stable and the localization effect is the best.

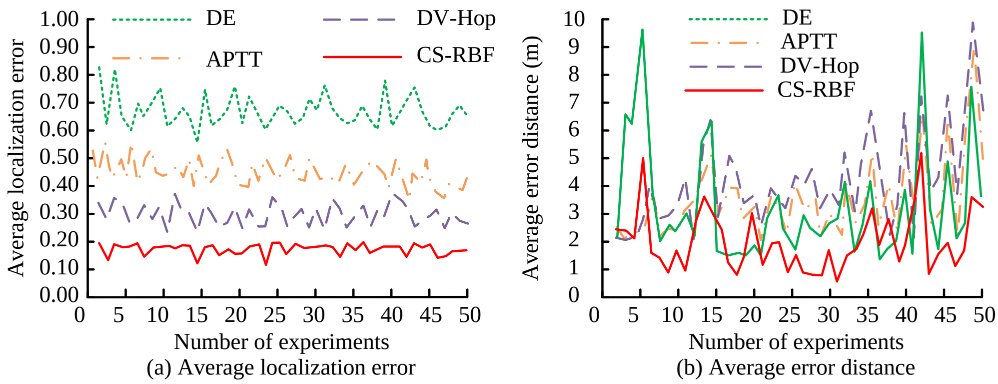


Figure 11: Comparison of average localization error and average error distance of nodes using different methods

4.2 Analysis of node localization effect for improved CS-RBF

The average localization error of different methods varies with the beacon node density, as shown in Figure 12. In Figure 12, overall, the average localization error decreased with the increase of beacon node density, indicating that the node localization effect was getting better. Among them, when the proportion of beacon nodes was 10%, the average localization error difference of the four node localization methods was significant. As the beacon node density gradually increases, the localization error difference of each method gradually narrows. When the proportion of beacon nodes was 35%, the average localization errors of DE, APTT, and DV Hop were 0.27, 0.23, and 0.22, respectively. The node localization error of CS-RBF was 0.18. In the entire change process, the node localization error of CS-RBF is consistently lower than the other three methods, indicating that this method has the best localization effect.

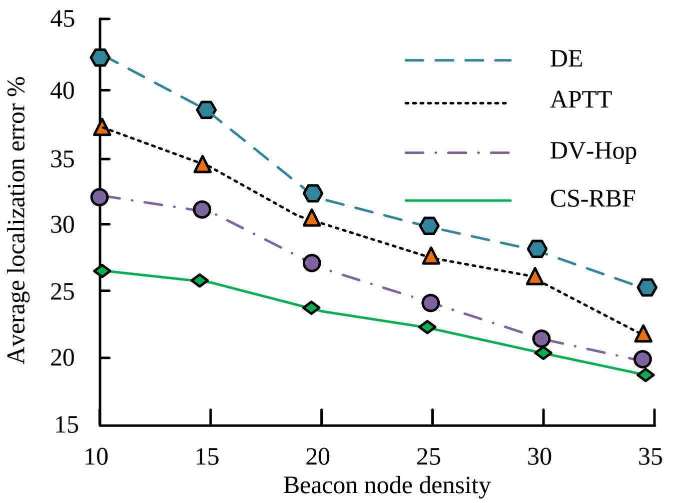
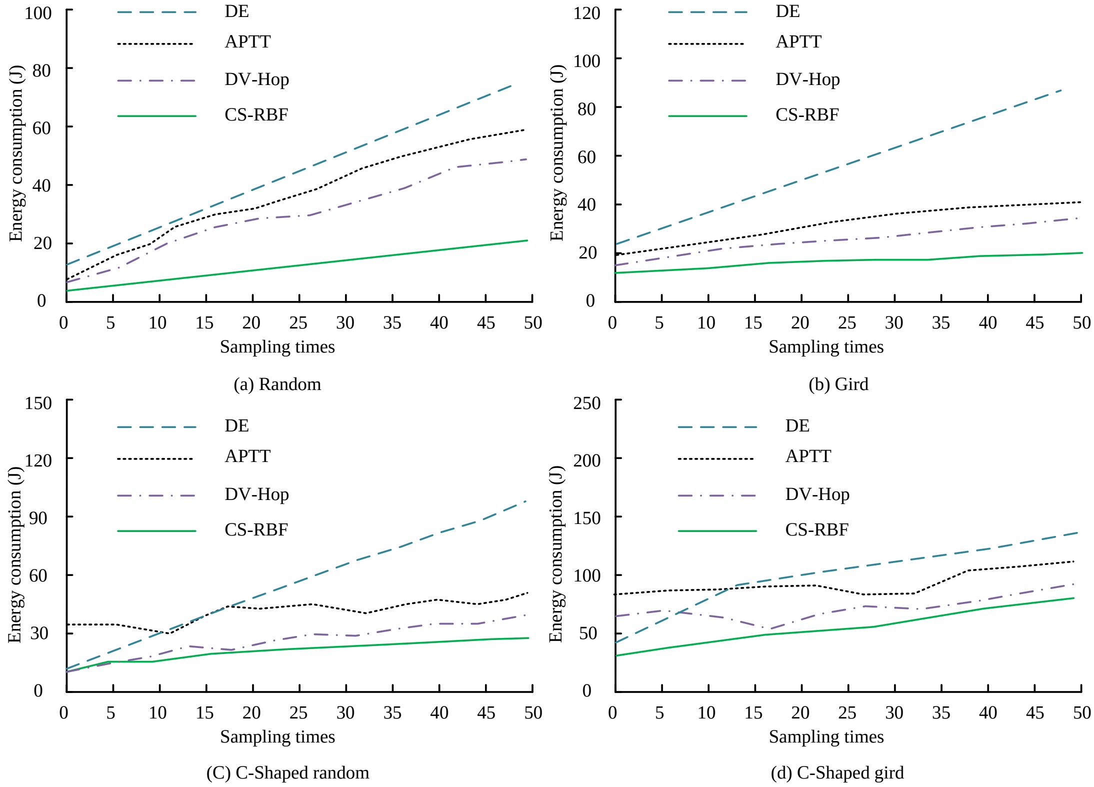


Figure 12: Comparison of average localization error with changes in beacon node density

To verify the actual effectiveness, different WSNs are selected to compare the node localization performance, including a completely random network (Random), a random lattice network (Gird), a C-shaped random network (C-Shaped Random), and a C-shaped random lattice network (C-Shaped Gird). The energy consumption of different node localization methods is shown in Figure 13. In Figure 13 (a), the energy consumption of DE, APTT, DV-Hop, and CS-RBF were 75J, 62J, 51J, and 23J, respectively. In the Gird network, the energy consumption of the four methods was 87J, 46J, 39J, and 21J, respectively. In Figure 13 (c), the energy consumption of the four methods was 105J, 56J, 47J, and 38J, respectively. In Figure 13 (d), the energy consumption of the four methods was 142J, 123J, 97J, and 86J, respectively. In four different WSNs, the energy consumption of each method varies. Overall, CS-RBF has the lowest energy consumption, highest efficiency, and lowest energy cost required for IoT node localization.

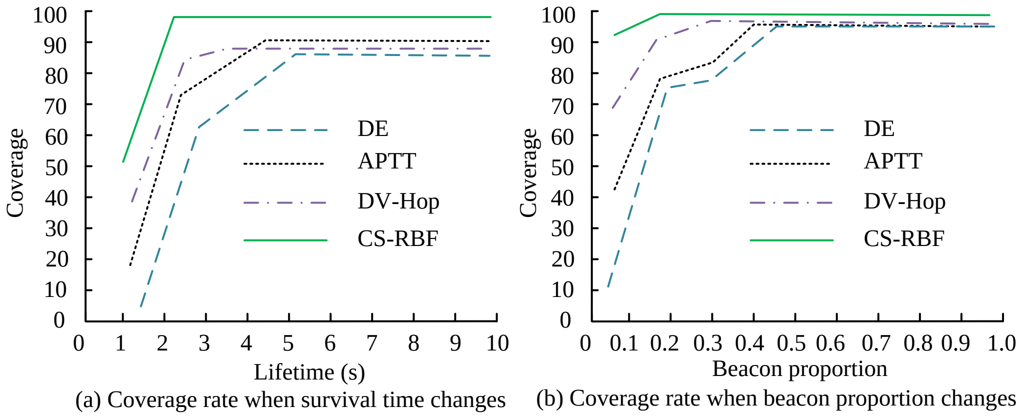


(a) Random (b) Gird

(C) C-Shaped random (d) C-Shaped gird

Figure 13: Energy consumption of different node location methods in different sensor network environments

Taking the C-Shaped random as an example, the proportion and coverage change of its beacon nodes are displayed in Figure 14. Figure 14 (a) presents the coverage change of node localization when the survival time changes. Figure 14 (b) shows the node localization coverage change when the beacon ratio changes. In Figure 14 (a), after 5s, the coverage of the four node localization methods reached a stable state. After stabilization, the coverage rates of DE, APTT, DV-Hop, and CS-RBF were 87%, 88%, 91%, and 99%, respectively. The coverage of CS-RBF method exceeded the comparison method. In Figure 14 (b), the coverage rates of DE, APTT, DV-Hop, and CS-RBF were 94%, 94%, 95%, and 100%, respectively. Under different conditions, the node coverage outperforms commonly used node localization methods, indicating that its localization accuracy is optimal.



(a) Coverage rate when survival time changes (b) Coverage rate when beacon proportion changes

Figure 14: The impact of changes in the beacon proportion and survival time on coverage using different methods

To further validate the effectiveness of the proposed method, the SOMCL method is used to collect 120 positions of 60 unknown nodes in the movement trajectory, obtain corresponding time series. Then a prediction model for node positions is constructed. The prediction results obtained by different methods are shown in Figure 15. As shown in Figure 15, in this dataset, the error range of CS-RBF was [2, -2]m, DV-Hop was [4, -6]m, APTT was [2, -5]m, and DE was [6, -7]m. Based on the error statistics of 50 sensor nodes arranged in the previous study, the designed method has better performance in different node datasets, which can adapt to different data environment changes.

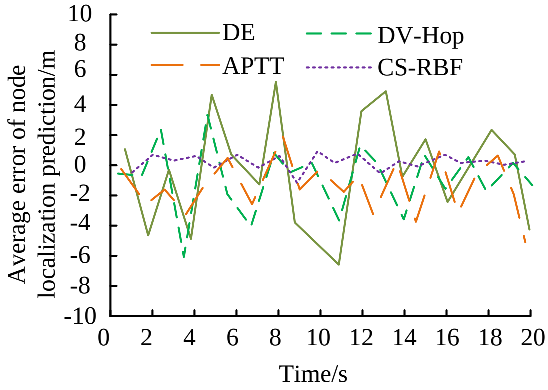


Figure 15: Comparison of error curves

The comparison between the proposed method and the APTT method with relatively better performance in the comparative methods is shown in Table 2. In Table 2, when comparing the research method with APTT, the localization error and error distance of the two methods are selected for statistical analysis. The results showed that there was a significant difference in localization error between the two methods (P<0.05). The error distances of the two methods obtained were 5.1 and 8.9, respectively, with significant statistical differences (P<0.05). There was a significant difference in time consumption between the two methods (P<0.05). The energy consumption was 23 and 62, respectively (P<0.05), and the difference was statistically significant.

Table 2: Statistical comparison of performance of different methods

Comparative indicators	Research method	APTT	t	P
Localization error	0.18	0.52	0.253	0.002
Error distance	5.1	8.9	0.349	0.001
Time consumption	2.4	4.3	0.268	0.001
Energy consumption	23	62	0.451	0.001

5 Discussion

A node prediction method based on improved RBFNN was designed for the localization of unknown nodes in IIoT. The designed method shows a smaller change in absolute error value, with a minimum absolute error value of -9.3 and an average absolute error value of 0.8. Compared with the RBFNN based on whale optimization proposed by Tian et al. [7], the research method can more effectively expand the search range, resulting in better optimal parameter values. In addition, this method has better universality and robustness. The node localization method based on CS-RBF showed that its average localization error among all nodes was 0.18. Compared with the method proposed by Yihong L I et al. [14], the research method has smaller localization error and error distances. Because the research method is based on the calculation results of historical node information for data training, it can effectively reduce the training error and improve its performance. Compared with the method proposed by Kassab R et al. [15], the research method corrects the distance error between historical nodes, reduces the influence of error interference and meaningless node information, thereby improving the possible imbalance between nodes and improving node localization accuracy. Compared with Sah D K et al. [16], the designed method has lower energy consumption in different sensor network environments, that is, the method has better operation effect and performance. Because the research method reduces the influence of unnecessary node information during the initial historical node data processing, it can not only improve node localization accuracy, but also reduce node computation and improve the operational efficiency.

Overall, the research method has better performance and operational efficiency in locating unknown node positions in IIoT. Given the current demand for node information localization in IIoT, this method has broad application prospects, such as the localization and intensity determination of various natural disasters, the localization services required by intelligent control systems, intelligent transportation systems, etc. This not only provides new ideas and technical means for the localization and prediction of future node information, but also has a positive impact on improving the overall performance of the IIoT.

However, there are still shortcomings in the research. The method used in the study only considers its application in a two-dimensional planar environment. Meanwhile, the node position prediction is based on the analysis of node positions in static environments, without considering node information localization in dynamic environments. In addition, computer hardware devices may also cause issues such as data overflow and data stream destruction. This will also require regular inspection and maintenance of computer hardware equipment in future research.

6 Conclusion

In IIoT applications, node localization technology is the key to obtaining accurate location information. On the basis of calculating the distance between nodes, a node localization method based on RBFNN was constructed. Then, the CS was used to optimize the performance of the RBFNN node localization method. Corresponding experiments were designed to verify its performance. The CS-RBF designed in the study had better convergence times and accuracy than the standard CS algorithm in different test functions. The optimization results obtained were also better than the unimproved CS algorithm. The time consumption of DE, APTT, DV-Hop, and CS-RBF methods after completing sample node localization sampling was 5.6s, 4.3s, 3.2s, and 2.4s, respectively. That is to say, the proposed CS-RBF method had the lowest time cost after completing all node localization, with better node localization performance. When the proportion of beacon nodes was 35%, the average localization errors of DE, APTT, and DV-Hop were 0.27, 0.23, and 0.22, respectively. The node localization error of CS-RBF was 0.18. When the beacon ratio changed, the coverage rates of DE, APTT, DV-Hop, and CS-RBF were 94%, 94%, 95%, and 100%, respectively. This indicates that the proposed CS-RBF node localization method has higher localization efficiency and accuracy, which has good application effects in node localization in the IIoT. However, there are still shortcomings in the research. Although error functions are used to correct the error values when calculating the distance between nodes, the correction effect in different application scenarios is affected to a certain extent. Future research will further optimize the node calculation in different scenarios and improve the correction effect.

References

[1] Zhen Chao, and Wenting Xu. A new general maximum intensity projection technology via the hybrid of U-Net and radial basis function neural network. Journal of Digital Imaging. 34(5):1264-1278, 2021. https://doi.org/10.1007/s10278-021-00504-8.

[2] Zhuopu Huo, Min Zhou, Clive Roberts, and Hairong Dong. Cuckoo search approach for automatic train regulation under capacity limitation. Science China Information Sciences. 66(4):295-296, 2023. https://doi.org/10.1007/s11432-020-3254-0.

[3] Ren Pu, Shunli Wang, Xianpei Chen, Junhan Huang, Mingfang He, and Wen Cao. A novel cuckoo search particle filtering strategy for the remaining useful life prediction of the lithium-ion batteries in hybrid electric vehicle. International Journal of Energy Research. 46(15):21703-21715, 2022. https://doi.org/10.1002/er.8712.

[4] Baida'a Abdul Qader Khudor, Yousif Abdulwahab Kheerallah, and Jawad Alkenani. A new method based on machine learning to increase efficiency in wireless sensor networks. Informatic. 46(9):45-52, 2022. https://doi.org/10.31449/inf.v46i9.4396.

[5] Gilsoo Lee, Walid Saad, Mehdi Bennis, Cheonyong Kim, and Minchae Jung. An online framework for ephemeral edge computing in the internet of things. IEEE Transactions on Wireless Communications. 22(3):1992-2007, 2023. https://doi.org/10.1109/TWC.2022.3208096.

[6] Cheng Yang, Zheng Wang, Sung-Kwun Oh, Witold Pedrycz, and Bo Yang. Ensemble fuzzy radial basis function neural networks architecture driven with the aid of multi-optimization through clustering techniques and polynomial-based learning. Fuzzy Sets and Systems. 438(6):62-83, 2022. https://doi.org/10.1016/j.fss.2021.06.014.

[7] Ye Tian, Yan-Lin He, and Qun-Xiong Zhu. Soft sensor development using improved whale optimization and regularization-based functional link neural network. Industrial and Engineering Chemistry Research. 59(43):19361-19369, 2020. https://doi.org/10.1021/acs.iecr.0c03839.

[8] Yukun Wang, and Xuebo Chen. A joint optimization QSAR model of fathead minnow acute toxicity based on a radial basis function neural network and its consensus modeling. RSC Advances. 10(36):21292-21308, 2020. https://doi.org/10.1039/d0ra02701d.

[9] Gholamreza Pazouki, Emadaldin Mohammadi Golafshani, and Ali Behnood. Predicting the compressive strength of self-compacting concrete containing Class F fly ash using metaheuristic radial basis function neural network. Structural Concrete: Journal of the FIB. 23(2):1191-1213, 2022. https://doi.org/10.1002/suco.202000047.

[10] Elham Shadkam, and Mehdi Bijari. A novel improved cuckoo optimisation algorithm for engineering optimisation. International Journal of Artificial Intelligence and Soft Computing. 7(2):164-177, 2020. https://doi.org/10.1504/IJAISC.2020.113498.

[11] Xiaoyan Ma, Yunfei Mu, Yu Zhang, Chenxi Zang, Shurong Li, Xinyang Jiang, and Meng Cui. Multi-objective microgrid optimal dispatching based on improved bird swarm algorithm. Global Energy Interconnection. 5(2):514-167, 2022. https://doi.org/10.1016/j.gloei.2022.04.013.

[12] Dong Zhang, Lijun Zhou, Lin Huang, Xue Chen, Jinfu Zhang, and Lei Guo. Study on improved cuckoo search algorithm applied in parameters estimation of multi-layer soil: 1D and 3D. IET Science, Measurement and Technology. 14(7):14.844-852, 2020. https://doi.org/10.1049/iet-smt.2020.0059.

[13] Ridvan Oruc, Ozlem Sahin, and Tolga Baklacioglu. Fuel flow rate modeling for descent using cuckoo search algorithm: a case study for point merge system procedure at Istanbul airport. Aircraft Engineering and Aerospace Technology. 94(5):824-833, 2022. https://doi.org/10.1108/AEAT-08-2021-0246.

[14] Yihong Li. Privacy protection algorithm for source node location based on phantom routing in the internet of things environment. International Journal of Innovative Computing, Information and Control. 17(3):973-989, 2021. https://doi.org/10.24507/ijicic.17.03.973.

[15] Rahif Kassab, Osvaldo Simeone, and Petar Popovski. Information-centric grant-free access for IoT fog networks: Edge vs. cloud detection and learning. IEEE Transactions on Wireless Communications. 19(10):6347-6361, 2020. https://doi.org/10.1109/TWC.2020.3002782.

[16] Dinesh Kumar Sah, Tu N. Nguyen, Korhan Cengiz, Braulio Dumba, and Vikas Kumar. Load-balance scheduling for intelligent sensors deployment in industrial internet of things. Cluster Computing. 25(3):1415-1727, 2022. https://doi.org/10.1007/s10586-021-03316-1.

[17] Juan Zhao, Xiaolong Xu, and Wei-Ping Zhu. Adaptive delay-constrained resource allocation in mobile edge computing for Internet of Things communications networks. Computer Communications. 160(7):607-613, 2020. https://doi.org/10.1016/j.comcom.2020.06.031.

[18] Ali Hasan Aidaros Alattas, Mahmood A. Al-Shareeda, Selvakumar Manickam, and Murtaja Ali Saare. Enhancement of NTSA secure communication with One-Time Pad (OTP) in IoT. Informatic. 47(1):1-10, 2023. https://doi.org/10.31449/inf.v47i1.4463.

[19] Peter P. Groumpos. A critical historic overview of artificial intelligence: Issues, challenges, opportunities, and threats. Artificial Intelligence and Applications. 1(4):197-213, 2023. https://doi.org/10.47852/bonviewAIA3202689.

[20] Song Yang, Nan He, Fan Li, Stojan Trajanovski, Xu Chen, Yu Wang, and Xiaoming Fu. Survivable task allocation in cloud radio access networks with mobile-edge computing. IEEE Internet of Things Journal. 8(2):1095-1108, 2020. https://doi.org/10.1109/JIOT.2020.3010533.