WILEY | Hindawi

Research Article

A Node Location Method in Wireless Sensor Networks Based on a Hybrid Optimization Algorithm

Jeng-Shyang Pan [1,2] **Fang Fan,** [1,3] **Shu-Chuan Chu** [1] **Zhigang Du,** [1] **and Huiqi Zhao** [3]

[1]*College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao, 266590 Shandong, China*
[2]*Department of Information Management, Chaoyang University of Technology, Taichung, Taiwan*
[3]*College of Intelligent Equipment, Shandong University of Science and Technology, Taian, 271019 Shandong, China*

Correspondence should be addressed to Shu-Chuan Chu; scchu0803@gmail.com

Wireless sensor networks (WSN) have gradually integrated into the concept of the Internet of Things (IoT) and become one of the key technologies. This paper studies the optimization algorithm in the field of artificial intelligence (AI) and effectively solves the problem of node location in WSN. Specifically, we propose a hybrid algorithm WOA-QT based on the whale optimization (WOA) and the quasi-affine transformation evolutionary (QUATRE) algorithm. It skillfully combines the strengths of the two algorithms, not only retaining the WOA's distinctive framework advantages but also having QUATRE's excellent coevolution ability. In order to further save optimization time, an auxiliary strategy for dynamically shrinking the search space (DSS) is introduced in the algorithm. To ensure the fairness of the evaluation, this paper selects 30 different types of benchmark functions and conducts experiments from multiple angles. The experiment results demonstrate that the optimization quality and efficiency of WOA-QT are very prominent. We use the proposed algorithm to optimize the weighted centroid location (WCL) algorithm based on received signal strength indication (RSSI) and obtain satisfactory positioning accuracy. This reflects the high value of the algorithm in practical applications.

1. Introduction

With the development of the Internet of Things (IoT) to this day, it can be said that the basic task originally conceived has been completed, namely, the interconnection of human, machine, and things [1]. However, people are no longer satisfied with the simple networking and put forward higher requirements. As the rapid expansion of scale, IoT has produced an alarming amount of real-time monitoring data [2]. There is no doubt that scientific management of these data requires a "Super Brain", and intelligence has become an inevitable choice for the evolution and upgrade. Artificial intelligence (AI) is fully qualified for this role. The history of AI is much longer than that of IoT, but it has been plagued by the difficulty of technical development and the lack of kinetic energy for achievement transformation. In recent years, AI has made breakthrough progress at the technical level, which makes people full of confidence and hope for its future. How

to guide AI from the laboratory into various real application scenarios is a challenging task. The booming IoT has provided a broad stage for AI to fully release its potentia, and the naturally existing data sources continuously provide power for it. Relying on the mature platform built by IoT, AI can meet the most urgent demand for technology landing. Although IoT already has a considerable volume in terms of industrial chain and scale, many key technologies still need to be polished and improved. IoT has been thinking for a long time to seek considerable progress and sustainable development, and the empowerment of AI just opens an important window of opportunity for its development. Therefore, the term AIoT (Artificial Intelligence of Things) came into being, which can be understood as the integration of AI and IoT.

AI covers a wide range of technologies, in which the optimization algorithm has received extensive attention from researchers. Among many optimization algorithms, the metaheuristic algorithm is the most active. Some of them

are inspired by the evolution phenomenon in nature, represented by genetic algorithm (GA) [3] and differential evolution algorithm (DE) [4]. Some algorithms are inspired by biological collective behavior, including particle swarm optimization (PSO) [5], artificial bee colony algorithm (ABC) [6, 7], Cat swarm optimization (CSO) [8, 9], WOA [10, 11], and symbiotic organism search algorithm [12, 13]. Some are related to physical or mathematical rules, such as black hole (BH) [14, 15] and QUATRE [16, 17]. Unfortunately, there is no one algorithm capable of solving all optimization problems according to the NFL theorem [18]. This has stimulated researchers' endless enthusiasm, either proposing new algorithms and technologies, or improving existing ones [19, 20], or combining different ones [21]. It is found that the hybrid strategy of the algorithm is worth trying in many cases. This article mixes two algorithms, WOA and QUATRE, and introduces the technology of dynamically shrinking the search space. In the test of the benchmark function, it is found that the hybrid algorithm has made great progress.

As the key core technology of IoT, WSN has been extended from military applications originally to a large number of civil fields, such as environmental and ecological protection, medical and health monitoring, agricultural planting, traffic management, and logistics management [22]. The main research contents of WSN include network topology control and network protocol, node deployment [23, 24], network security [25, 26], location technology, and data fusion and data management. The development of WSN also faces many challenges. For example, how to use the limited communication ability to complete the transmission of sensing data needs further research. And how to maximize the service life of the network through low-power design is also an urgent problem to be solved. This paper introduces the optimization algorithm into the node location problem of WSN and achieves satisfactory location accuracy.

The rest of this article is as follows: the second part is the related research work, mainly for the brief introduction of WOA and QUATRE; the third part discusses the combination and improvement strategy of the two algorithms in detail; the fourth part is the statistics and analysis of experimental data based on benchmark functions; the fifth part is to use the algorithm to realize the location in WSN; and the last part is the summary and outlook.

2. Related Works

2.1. Whale Optimization Algorithm (WOA). As a metaheuristic optimization algorithm based on swarm intelligence, WOA is exquisitely designed and full of characteristics. It simulates the hunting behavior of humpback whale groups. Inspired by this hunting method, WOA designed three models: searching for prey, encircling prey, and bubble-net attacking. First, initialize the position of each whale: $\vec{X} = (x_1, x_2, \cdots, x_n)$, which represents an n-dimensional candidate solution. Through the cooperation of two stages of exploration and exploitation, the best position is gradually found (i.e., the optimal solution).

Three vectors appear in the algorithm: $\vec{A}$, $\vec{C}$ and $\vec{D}$. $\vec{A}$ and $\vec{C}$ are defined as follows:

$$\vec{A} = \vec{a} \bullet \left(2\vec{r} - 1\right), \ \vec{C} = 2\vec{r} \tag{1}$$

where $\vec{r}$ is a random vector between [0, 1]. As the iteration progresses, $\vec{a}$ decreases linearly from 2 to 0. Therefore, $\vec{A} \in [-2, 2]$, $\vec{C} \in [0, 2]$. There is a random number prob $\in [0, 1]$, which is used to indicate the selection probability of the updated mode. The random vector $\vec{A}$ and the random number prob jointly control the conversion between exploration and exploitation. Vector $\vec{D}$ has different representations at different stages. The operation symbol ($\bullet$) means multiplying item by item, and $|\bullet|$ means taking the absolute value.

(1) Exploration: searching mode

WOA's exploration phase mimics the process of whales searching for prey in the ocean. Use a randomly selected search agent to update the solution. To achieve the goal of global search, the search agent is forced to leave the reference whale. The position update formula for exploration is as follows:

$$\begin{cases} \vec{D} = \left|\vec{C} \bullet \vec{X}_{\mathrm{rand}} - \vec{X}(t)\right|, \\ \vec{X}(t+1) = \left|\vec{X}_{\mathrm{rand}} - \vec{A} \bullet \vec{D}\right|. \end{cases} \tag{2}$$

$\vec{X}(t)$ is the solution vector at the t – th iteration. $\vec{X}_{\mathrm{rand}}$ represents a random solution in the current population. When prob < 0.5 and $|\vec{A}| \geq 1$, Equation (2) will be used to update the solution vector.

(2) Exploitation

The exploitation phase of WOA includes two modes: encircling mode and bubble-net attacking mode.

2.1.1. Encircling Mode. Once humpback whales find their prey, they surround them. Since the best position is unknown, it is assumed that the current best solution is the destination or the approximate best position. Other search agents will update their position with reference to the current optimal, as shown in Eq.(3):

$$\begin{cases} \vec{D} = \left|\vec{C} \bullet \overrightarrow{X^*}(t) - \vec{X}(t)\right| \\ \vec{X}(t+1) = \overrightarrow{X^*}(t) - \vec{A} \bullet \vec{D} \end{cases} \tag{3}$$

$\overrightarrow{X^*}(t)$ denotes the current optimal. When $prob < 0.5$ and $|\vec{A}| < 1$, the solution will be updated by the above formula.

2.1.2. Bubble-Net Attacking Mode. Bubble-net attacking is the most distinctive hunting method of humpback whales. In

addition to swimming around prey, whales also move along a spiral path. WOA creates a spiral equation to simulate through the distance, as shown in the following formula:

$$\begin{cases} \overrightarrow{D'} = \left| \overrightarrow{X^*}(t) - \overrightarrow{X}(t) \right|, \\ \overrightarrow{X}(t+1) = \overrightarrow{D'} \bullet e^{bl} \bullet \cos(2\pi l) + \overrightarrow{X^*}(t). \end{cases} \quad (4)$$

Here, the distance between $\overrightarrow{X}(t)$ and $\overrightarrow{X^*}(t)$ is recorded as $\overrightarrow{D'}$, which can be understood as the distance between the whale and its prey. The constant b is used to define the shape of the logarithmic spiral. l is a random number between $[0, 1]$. When prob ≥ 0.5, Equation (4) will be used to generate a new solution.

2.2. QUasi-Affine Transformation Evolutionary (QUATRE) Algorithm. QUATRE is a coevolutionary algorithm based on quasi-affine transformation. As a new structure of evolutionary calculation, it can be used to solve the problem of distributed bias in the DE algorithm. The evolution formula is similar to the affine transformation, as shown below:

$$X \leftarrow M \bigotimes X + \bar{M} \otimes B. \quad (5)$$

Suppose there are ND-dimensional solution vectors X_i, which constitute a population matrix: $X = (X_1, X_2, \cdots, X_i, \cdots, X_N)^T$. There are two matrices: the donor matrix B and the coevolutionary matrix M. The operator $\bigotimes$ represents the bitwise multiplication of matrix elements. M and $\bar{M}$ are mutually inverse matrices, and and they are both binary matrices (matrix elements only take 0 or 1).

2.2.1. The Coevolutionary Matrix M. The QUTARE algorithm considers that all search agents in the population have the same status. By the cooperative structure, the search agents cooperate with each other and share information. The initialized matrix M_{initial} is constructed, and then, through a series of transformations, the coevolutionary matrix M is obtained.

There are three methods for constructing M_{initial}. The specific choice depends on the size relationship between the population size ps and the search space dimension D. If ps $= D$, M_{initial} is a D-dimensional binary lower triangular matrix, as shown in Equation (6). When ps $> D$, M_{initial} needs to be expanded. If ps $= p \bullet D$, M_{initial} is a matrix of pD-dimensional binary lower triangular matrices stacked vertically. If ps $= p \bullet D + q$, the first $p \bullet D$ rows are handled according to the previous situation, and the rest take the first q lines in Equation (6), as shown in Equation (7). Next, M can be obtained after two steps of transformation of M_{initial}. First, the elements of each row in M_{initial} are randomly arranged. Then, the elements of each row vector remain unchanged while their positions

in the matrix are randomly arranged. So we can get M, see

$$M_{\text{initial}} = \begin{bmatrix} 1 & & & \\ 1 & 1 & & \\ & & \cdots & \\ 1 & 1 & \cdots & 1 \end{bmatrix} \sim \begin{bmatrix} 1 & & 1 & \\ 1 & 1 & \cdots & \\ & & \cdots & \\ & 1 & \cdots & 1 \end{bmatrix} = M. \quad (6)$$

$$M_{\text{initial}} = \begin{bmatrix} 1 & & & \\ 1 & 1 & & \\ & & \cdots & \\ 1 & 1 & \cdots & 1 \\ 1 & & & \\ 1 & 1 & & \\ & & \cdots & \\ 1 & 1 & \cdots & 1 \\ 1 & & & \\ 1 & 1 & & \\ \cdots & & & \end{bmatrix} \sim \begin{bmatrix} 1 & & & \\ 1 & \cdots & & 1 \\ & 1 & \cdots & \\ 1 & & \cdots & 1 \\ 1 & & & \\ 1 & \cdots & 1 & \\ & & \cdots & \\ 1 & 1 & \cdots & 1 \\ 1 & 1 & & \\ 1 & & & 1 \\ & 1 & & \end{bmatrix} = M. \quad (7)$$

2.2.2. The Donor Matrix B. The donor matrix B dominates the way of evolution. Similar to the different mutation strategies in DE, there are many ways to generate B. The mode used here is called QUATRE/best/1, as shown in

$$B_{i,G} = X_{g\text{best},G} + F \bullet (X_{r1,G} - X_{r2,G}). \quad (8)$$

Assume that the population matrix of the Gth generation is $X_G = (X_{1,G}, X_{2,G}, \cdots, X_{i,G}, \cdots, X_{N,G})^T$, which consists of ND-dimensional candidate solutions $X_{i,G}$. Each $X_{i,G}$ has an evolution guide vector $B_{i,G}$ corresponding to it. $X_{g\text{best},G}$ represents the best individual in the population. X_{r1} and X_{r2} are generated by randomly arranging the row vectors of X_G. $F \in (0, 1]$ is the scaling factor.

3. The Hybrid Algorithms Based on WOA and QUATRE (WOA-QT)

The metaheuristic algorithm has two basic elements: exploration and exploitation. The task of exploration is to trial different solutions in the search space and try to find the global optimal. The task of the exploitation is to search in the local scope and fully tap the optimization opportunities brought by the current optimal. Ideally, if the two stages can cooperate with each other and realize smooth transition, the local optimum can be avoided and the optimal solution can be found finally [27].

WOA has proved to be superior to some advanced optimization algorithms, for example, PSO, DE, GSA, and fast evolutionary programing (FEP) [28]. It provides multiple

update modes and strives to balance the relationship between exploration and exploitation. But inevitably, it still faces many problems, such as how to further improve the exploration ability, release the exploitation potential, and avoid local optimization. Local optimal stagnation and slow convergence are its two main disadvantages. In order to improve the ability to escape from the local optimum, this paper introduces the QUATRE algorithm into WOA and use the strategy of dynamically shrinking the search space to improve time efficiency. In this part, we will analyze the causes of the problems in detail and explain the relevant countermeasures.

3.1. The Combination of WOA and QUATRE.

Although WOA has two different exploitation models, both of them are highly dependent on the current optimum. When the problem is a multimodal function or a complex function, there is a risk of falling into a local optimum. How to deal with this situation requires careful consideration.

QUATRE is a coevolution algorithm, which uses matrix mode calculation and concise parameters. From the point of view of statistics and probability theory, the moving way of individuals is more reasonable in the process of evolution. In addition, the coevolutionary matrix in the algorithm can perceive the structure of objective function more comprehensively, which makes the search of solution space more effective.

Based on the above considerations, this paper proposes a hybrid algorithm based on WOA and QUATRE. The specific idea is to embed QUATRE in the exploitation stage of WOA. The current optimal solution $\overrightarrow{X^*}(t)$ plays an absolute dominant role in both the encircling mode and the bubble-net attacking mode. The QUATRE algorithm can be used to preprocess the current optimal, and the effect is equivalent to perturb the current optimal solution in accordance with statistics and probability theory to make up for the lack of randomness and diversity. The fusion method of the two algorithms in the encircling mode is as follows:

$$\begin{cases} \overrightarrow{X_Q}(t) = M \bigotimes \overrightarrow{X^*}(t) + \bar{M} \otimes B. \\ \vec{D} = \left| \vec{C} \bullet \overrightarrow{X_Q}(t) - \vec{X}(t) \right|. \\ \vec{X}(t+1) = \overrightarrow{X_Q} - \vec{A} \bullet \vec{D}. \end{cases} \quad (9)$$

Here, the current optimal solution $\overrightarrow{X^*}(t)$ is processed by the QUATRE to get $\overrightarrow{X_Q}(t)$, which will replace $\overrightarrow{X^*}(t)$ to guide the position update of the encircling mode in the exploitation stage.

Similarly, in the bubble-net attacking mode, the fusion of the two algorithms is also performed, as shown in the following formula:

$$\begin{cases} \overrightarrow{X_Q}(t) = M \bigotimes \overrightarrow{X^*}(t) + \bar{M} \otimes B, \\ \overrightarrow{D'} = \left| \overrightarrow{X_Q}(t) - \vec{X}(t) \right|, \\ \vec{X}(t+1) = \overrightarrow{D'} \bullet e^{bl} \bullet \cos(2\pi l) + \overrightarrow{X_Q}(t). \end{cases} \quad (10)$$

$\overrightarrow{X_Q}(t)$ replaces $\overrightarrow{X^*}(t)$ to guide the position update of the bubble-net attacking mode during the exploitation phase. In the proposed WOA-QT, Equations (9) and (10) will still have a 50% probability of being selected.

QUATRE has different mutation strategies, which emphasize different focuses of search capabilities [29, 30]. The mode QUATRE/best/1 is concise in form which only uses the current optimal solution $X_{g\text{best},G}$ and two randomly selected solutions X_{r1} and X_{r2}. $X_{g\text{best},G}$ dominates the evolution of the population, so that individuals with higher fitness values can be fully utilized to perform good local search, and the convergence speed is faster. Although the combination of the two algorithms has increased the randomness of the population, we choose QUATRE/best/1 as the mutation strategy to avoid overcorrection. Therefore, both the population diversity is ensured and the convergence speed is also taken into account. Therefore, both the population diversity is ensured and the convergence speed is also taken into account.

3.2. Strategy of Dynamically Shrinking Search Space (DSS).

The candidate solution should gradually approach the global optimal solution. The general trend is that the size of search space will decrease as the iteration progresses. However, the search space of WOA is always the size at the time of initialization. It is obviously unreasonable to roam randomly in such a fixed-size space, which may cause delay in the optimization process. In this paper, we propose a mechanism to dynamically shrink the search space, which is used in the exploration phase of WOA. As the optimization proceeds, the upper and lower limits of the selection range of the random solution will change accordingly, so that the size of the search space is gradually reduced, thereby improving search efficiency. The specific method is as follows:

First define four variables: the upper limit of the newly generated search space ub_new, the lower limit of the newly generated search space lb_new, the largest solution component of the current iteration ub_ max, and the smallest solution component of the current iteration lb_ min. The initial values of the above variables are set according to the initial size of the search space.

$$\text{ub}_{\text{new}} = \text{ub}_{\text{new}} + (\text{ub}_{\text{max}} - \text{ub}_{\text{new}}) * \left(\frac{t}{\max_{\text{iter}}} \right), \quad (11)$$

$$\text{lb}_{\text{new}} = \text{lb}_{\text{new}} + (\text{lb}_{\text{min}} - \text{lb}_{\text{new}}) * \left(\frac{t}{\max_{\text{iter}}} \right). \quad (12)$$

where t represents the current number of iterations and max _iter is the maximum number of iterations. As the iteration progresses, ub_new will decrease approximately linearly according to Equation (11), and lb_new will increase approximately linearly according to Equation (12). As a result, the exploration scope is gradually reduced, and the global optimum is constantly approached. The random solution in the exploration process is now generated from a dynamically shrinking search space defined by ub_new and

```
Generate initial population X containing N individuals Xᵢ(i = 0, 1, 2, 3,···,N).
Initialize the parameters related to the algorithm: a, r, b, prob, t, max_iter.
Initialize two matrices in QUATRE:
    the coevolutionary matrix M (Eq. (6) or Eq. (7)).
    the donor matrix B (Eq. (8)).
Initialize the parameters related to the search space size:
    ub = ub_new = ub_max.
    lb = lb_new = lb_min.
Calculate the fitness value of each solution.
X* = the best individual.
    While (t < max_iter) and (not obtain the predefined function value).
    For each solution.
    Update a, r, b, l, prob, M, B.
        If1 (prob < 0.5).
        If2 (|A| < 1).
           Update the position of the current solution using Eq. (9).
        Else if2 (|A| ≥ 1).
           Select a random solution Xrand.
           Update the position of the current solution using Eq. (2).
        End if2.
    Else if1 (prob ≥ 0.5).
           Update the position of the current solution using Eq. (10).
        End if1.
    End for.
    Check if any solution goes beyond the search space and amend it.
    Calculate the fitness value of each solution.
    Update X* if there is a better solution.
    t = t +1.
    Shrink the search space size using Eq. (11) and (12).
End while.
Return X*
```

PSEUDOCODE 1: The pseudocode of the proposed WOA-QT.

lb_new. And the contraction speed is not too fast, which can effectively avoid local optimal stagnation.

3.3. Hybrid Algorithm WOA-QT.

This section describes the optimization steps of WOA-QT. The pseudo code is shown in Pseudocode 1.

(1) Initialization: generate N individuals $X_i (i = 0, 1, 2, 3,···,N)$ randomly to form population X. Set initial values to parameters in WOA and QUATRE. Specify the maximum number of iterations max _iter and parameters related to the size of the search space (including initial upper limit ub and lower limit lb, the upper limit ub_new and lower limit lb_new of search space after each iteration, the maximum solution component ub_ max and minimum solution component lb_ max of the current iteration).

(2) Evaluation: calculate each individual's fitness value $f(X_i)$. By comparing with each other, determine the individual X^* who currently has the best fitness value.

(3) Update: choose one of the modes to update the individual's position.

(i) Exploration: when $prob < 0.5$ and $|A| \geq 1$, use Equation ((2)) to update the solution. That is to select a random solution X_{rand} in the current search space (between the upper limit ub_new and lower limit lb_new).

(ii) Exploitation

(A) When $prob < 0.5$ and $|A| < 1$, use Equation (9) to update the solution. This is the WOA encircling mode embedded with QUATRE

(B) B. When $prob \geq 0.5$, use Equation (10) to update the solution. This is the WOA bubble-net attacking mode embedded with QUATRE

(4) Reevaluation: re-evaluate the population to find the best individual X^*. And shrink the size of search space according to the Equations (11) and (12).

(5) Termination: repeat steps (3) to (4). If a predefined function value has been obtained or all iterations have been completed, record the global optimal solution X^* and its best fitness value $f(X^*)$. The optimization process ends here.

4. Experiment Results and Analysis

This part selects 30 benchmark functions for performance evaluation, as shown in Tables 1, 2, and 3. Most of the benchmark functions come from literature [31], including three categories: unimodal, multimodal, and fixed-dimension complex functions. We also separately evaluate the performance in solving low-dimensional and high-dimensional problems. The selected unimodal functions and multimodal functions are scalable functions covering low and high

TABLE 1: Details of 10 unimodal functions.

Functions	DIMENSION	RANGE	$f\_$min
$F_1(x) = \sum_{i=1}^{n} x_i^2$	30,500	[100, 100]	0
$F_2(x) = \sum_{i=1}^{n} ix_i^2$	30,500	[−10, 10]	0
$F_3(x) = \sum_{i=1}^{n} |x| + \Pi_{i+1}^{n}$	30,500	[−10, 10]	0
$F_4(x) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j^2 \right)$	30,500	[100,100]	0
$F_5(x) = \max_i \{|x_i|, i\epsilon[1, m]\}$	30,500	[100,100]	0
$F_6(x) = \sum_{i=1}^{n-1} \left[100\left(x_{x+1} - x_i^2\right) + (x_i - 1)^2 \right]$	30,500	[−30, 30]	0
$F_7(x) = \sum_{i=1}^{n-1} (x_i + 0.5)^2$	30,500	[100,100]	0
$F_8(x) = \sum_{i=1}^{n} ix^4$	30,500	[−1.28,1.28]	0
$F_9(x) = \sum_{i=1}^{n} ix^4 + \text{rand } [0, 1)$	30,500	[−1.28,1.28]	0
$F_{10}(x) = \sum_{i=1}^{n} |x_i^2|$	30,500	[−1, 1]	0

dimensions. All the experiments were compared among WOA-QT, WOA, and QUATRE. Each algorithm runs independently 40 times. The specific experimental scheme is as follows.

4.1. *Experiments for Low-Dimensional Unimodal and Multimodal Functions.* First, the low-dimensional unimodal functions and multimodal functions are tested, and the dimension is set to 30. The comparative indicators we selected include the minimum (i.e., optimal value), medium, maximum, mean, and standard deviation values of each experiment. Figure 1 shows the convergence curve of each function.

In the test of 10 unimodal functions, we can see that WOA-QT has achieved the first place in the comparison of the optimal values for 9 times (F1-F6, F8-F10). Compared with the other two algorithms, WOA-Q's optimization accuracy and convergence speed have been significantly improved, showing excellent local search capabilities. In addition, WOA-QT also has an absolute advantage in the comparison of other statistical data, which shows that its performance stability is far better than WOA and QUATRE.

From the experimental results of 10 multimodal functions, we can see that WOA-QT has an absolute advantage in the ranking of all indicators. This shows that it can effectively avoid falling into the local optimal.

4.2. *Experiments for High-Dimensional Unimodal and Multimodal Functions.* This experiment continues to use the test functions from the previous section but expand the dimension to 500. The conclusion of this experiment is basically consistent with that of the last one. The WOA-QT still maintains good performance in high-dimensional situations. Please see Figure 2 for the function convergence curves.

4.3. *Experimental Results of Fixed-Dimension Complex Function.* We also tested 10 fixed-dimension complex functions. It is found that the comprehensive performance of WOA-QT is still the best of the three. Figure 3 shows the function convergence curves obtained in this experiment.

4.4. *Analysis of Experiment Results.* In this paper, different types of benchmark functions are selected for testing for the sake of the scientificity and fairness of evaluation. It also examines the algorithm's ability to solve scalable dimensional problems.

When solving the unimodal problem, WOA-QT embodies the outstanding local exploitation ability. In the face of multimodal problems, it can well avoid falling into local optimum. Both optimization accuracy and speed benefit greatly from the effective hybrid mechanism of the two algorithms. In addition, when the dimension of the problem space expands or shrinks, or when the problem is gradually complicated and the amount of calculation increases suddenly, WOA-QT still maintains reliable optimization performance, especially with a satisfactory convergence speed. The high time efficiency proves the effectiveness of the DSS strategy.

TABLE 2: Details of 10 multimodal functions.

Functions	DIMENSION	RANGE	$f\_$min
$F_{11}(x) = 1 - \cos\left(2\Pi\sqrt{\left(\sum_{i=1}^{n} x_i^2\right)}\right) + 0.1\sqrt{\sum_{i=1}^{n} x_i^2}$	30,500	$[-100,100]$	0
$F_{12}(x) = \sum_{i=1}^{n}\left[x_i^2 - 10 * \cos(2\pi x_i) + 10\right]$	30,500	$[-5.12,5.12]$	0
$F_{13}(x) = -208\exp\left(-0.2\sqrt{\dfrac{1}{m}\sum_{i}^{n} x_i^2}\right) - \exp\left(\dfrac{1}{m}\sum_{i=1}^{n}\cos(2\pi x_i) + 20 + 2.718\right)$	30,500	$[-32, 32]$	0
$F_{14}(x) = \dfrac{1}{4000} + \sum_{i=1}^{n} x_i^2 - \Pi_{i=1}^{n}\cos\left(\dfrac{x_i}{\sqrt{i}}\right) + 1$	30,500	$[-600,600]$	0
$F_{15}(x) = \dfrac{\pi}{n} * \left\{10 * \sin(\pi y_1) + \sum_{i-1}^{n-1}(y_i - 1)^2\left[1 + 10 * \sin^2(\pi y_{i+1})\right]\right\}$ $y_i = 1 + \dfrac{x_i+1}{4} * u(z_i, a, k, m) = \begin{cases} k(x_i - a), x > a \\ 0, -a < x_i < a \\ k(-x_i - a), x > a \end{cases}$	30,500	$[-50, 50]$	0
$F_{16}(x) = \sum_{i=1}^{n}\|x_i\sin(x_i) + 0.1x_i\|$	30,500	$[-100,100]$	0
$F_{17}(x) = 0.59 + \dfrac{\sin^2\left(\sqrt{\left(\sum_{i=1}^{n} x_i^2\right)}\right)^{-0.5}}{1 + 0.0001\left(\sum_{i=1}^{n} x_4^2\right)}$	30,500	$[-100,100]$	0
$F_{18}(x) = \sum_{i=1}^{n} 0.1n - \left(0.1\sum_{i=1}^{n}\cos(5\pi x_i) - \sum_{i=1}^{n}(x_i - \pi)^2\right)$	30,500	$[-10, 10]$	0
$F_{19}(x) = \sum_{i=1}^{n-1} 0.1n\left(x_i^2 + x_{x+1}^2\right)^{0.25} \times \left[1 + \sin^2\left(50\left(x_i^2 + x_{x+1}^2\right)\wedge(0.1)\right)\right]$	30,500	$[-1, 1]$	0
$F_{20}(x) = \sum_{i=1}^{n}\left(10^6\right)^{\frac{i-1}{n-1}} x_i^2$	30,500	$[-10, 10]$	0

5. Node Location in WSN

WSN, RFID, cloud computing, and wireless data communication, etc., together constitute the technical foundation of IoT. Among them, WSN is not only an important information collection subject of the perceptual layer of IoT but also a basic transmission channel of the network layer. WSN plays a vital role in how to realize the intelligent perception, transmission, processing, and application of data in IoT. There are many research topics in WSN, among which node location has always been a research hotspot. Because the monitoring data with unknown location information is usually meaningless, and the location problem is the basis and premise of most applications, the performance of the sensor node location algorithm directly affects its usability. The commonly used evaluation indicators include location accuracy, scale, node density, and power consumption. This paper uses the proposed WOA-QT algorithm to solve the node location problem in WSN. Specifically, it is to intelligentize the node location algorithm to further improve the positioning accuracy.

Although there are many node location algorithms, they can be classified into range-based and range-free. Range-based location algorithm calculates the position of unknown nodes by measuring distance or angle information. Commonly used ranging technologies include Time of Arrival (TOA), Time Difference of Arrival (TDOA), Received Signal Strength of Indicator (RSSI), and Angle of Arrival (AOA). Range-free localization algorithm only determines the node position according to the connectivity of the network and other information without knowing the distance or angle. The commonly used methods are convex programming, DV-hop, centroid location algorithm, and so on.

Different location algorithms have their own advantages and disadvantages. The location algorithm used in this paper integrates RSSI ranging and weighted centroid positioning algorithms (WCL_RSSI), which can improve positioning accuracy to a certain extent. To meet the requirements of further reducing the positioning error, this paper uses the proposed WOA-QT algorithm to optimize WCL_RSSI.

5.1. RSSI Ranging Technology. The principle of the RSSI ranging method is to convert the strength of the received wireless signal energy into the distance between the nodes, and then use it to estimate the location of the unknown node. Because no additional hardware is required, and the cost and power

TABLE 3: Details of 10 fixed-dimension complex functions.

Functions	DIMENSION	RANGE	$f\_$ min
$F_{21}(x) = \left(\sum_{i=1}^{5} i * \cos\left((i+1)x_1 + i\right)\right) * \left(\sum_{i=1}^{5} i * \cos\left((i+1)x_2 + i\right)\right)$	2	[−5.12, 5.12]	−186.7309
$F_{22}(x) = \sum_{i=1}^{11} \left[a_i - \dfrac{x_1\left(b_i^2 + b_i x_2\right)}{b_i^2 + b_i x_3 + x_4}\right]^2$	4	[−5, 5]	0.0003
$F_{23}(x) = -\dfrac{1 + \cos\left(12\sqrt{x_1^2 + x_2^2}\right)}{0.5\left(x_1^2 + x_2^2\right) + 2}$	2	[−5.12, 5.12]	−1
$F_{24}(x) = \left(x_2 - \dfrac{5.1}{4\pi^2}x_1^2 + \dfrac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \dfrac{1}{8\pi}\right)\cos x_1 + 10$	2	[−5, 5]	0.398
$F_{25}(x) = \left[1 + (x_1 + x_2 + x_3)^2 * \left(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1 x_2 + 3x_2^2\right)\right]$ $\times \left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1 x_2 + 27x_2^2\right)$	2	[−2, 2]	3
$F_{26}(x) = -0.0001\left(\left\|\sin(x_1)\sin(x_2)\exp\left(\left\|100 - \dfrac{\sqrt{x_1^2 + x_2^2}}{\pi}\right\|\right)\right\| + 1\right)^{0.1}$	2	[−10, 10]	−2.0626
$F_{27}(x) = -0.0001\left(\left\|\cos(x_1)\cos(x_2)\exp\left(\left\|100 - \dfrac{\sqrt{x_1^2 + x_2^2}}{\pi}\right\|\right)\right\| + 1\right)^{0.1}$	2	[−10, 10]	−2.0626
$F_{28}(x) = -\sum_{i=1}^{5} \left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	[0, 10]	−10.1532
$F_{29}(x) = -\sum_{i=1}^{7} \left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	[0, 10]	−10.4028
$F_{30}(x) = -\sum_{i=1}^{10} \left[(X - a_i)(X - a_i)^T + c_i\right]^{-1}$	4	[0, 10]	−10.5363

consumption are relatively low, the RSSI method is widely used, especially for large-scale WSN node location problems. However, environmental factors such as distance and obstacles will have a great impact on the transmission of wireless signals, resulting in severe attenuation of the signal's power strength. This paper selects the log-distance distribution model as the distance measurement model of wireless signal propagation path loss, as shown below:

$$\text{RSSI}(d) = \text{RSSI}(d_0) - 10\gamma lg\left(\frac{d}{d_0}\right) + X_\sigma. \qquad (13)$$

Assuming that the distance between the node to be located and the signal source is d, $\text{RSSI}(d)$ indicates the signal strength received by the node; d_0 is the reference distance, generally $1m$; γ is the path loss factor that reflects the influence of the environment on signal transmission. The more obstacles in the transmission path, the greater the value of γ; X_σ is a random variable, obeying normal distribution: $X_\sigma \sim N(\mu, \sigma^2)$. By Equation (13), the distance d can be obtained: $d = 10^{(RSSI(d0) - RSSI(d) + X_\sigma/10\gamma)}$.

There are K beacon nodes in the target area: $S_i(x_i, y_i)$, $i = 1, 2, \cdots, K$. $N(x, y)$ is the unknown node. If the distance between N and S_i is too far, the signal power will be greatly attenuated, which will seriously affect the location accuracy.

Generally speaking, the unknown node and the beacon node are not directly connected, and there may be several intermediate nodes between the two. Therefore, it can be considered that RSSI starts from the beacon node S_i and is finally transmitted to the unknown node N in a relay manner. The distance of the connection path between the two is recorded as D_i. If M intermediate nodes are passed, then $D_i = \sum_{l=1}^{M+1} d_r$, where d_r can be obtained by Equation (13).

In order to convert D_i to the Euclidean distance ρ_i between N and S_i, this requires knowing the approximate proportional relationship between the two distances. Assume that the connection distance between two beacon nodes S_i and S_j is $D_{i,j}$ (can be obtained by RSSI ranging), and the Euclidean distance is

$$d_{i,j} = \sqrt{\left(x_j - x_i\right)^2 + \left(y_j - y_i\right)^2}. \qquad (14)$$

We define α_i as the path scale factor of the beacon node S_i, as shown below:

$$\alpha_i = \sum_{j=1, j\neq i}^{K} \left(D_{i,j}/d_{i,j}\right). \qquad (15)$$
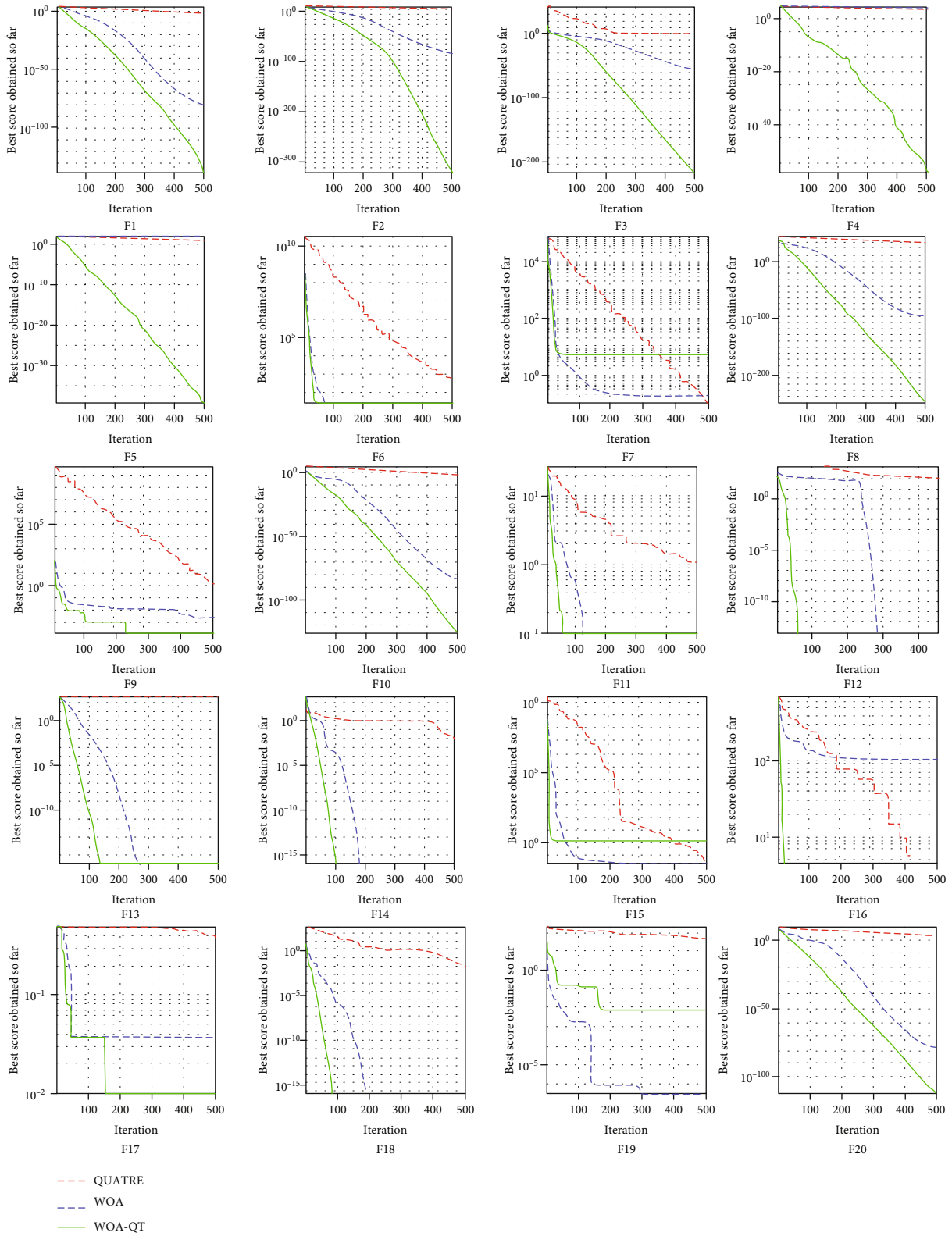
FIGURE 1: The convergence curves of 30-dimensional functions (F1-F20).

Obviously, the distance ρ_i can be obtained with the following formula:

$$\rho_i = D_i/\alpha_i. \tag{16}$$

5.2. Weighted Centroid Location Algorithm Based on RSSI (WCL_RSSI). It is not hard to see, if only relying on RSSI ranging to locate, the accuracy is poor. Therefore, a weighted centroid location algorithm (WCL) based on
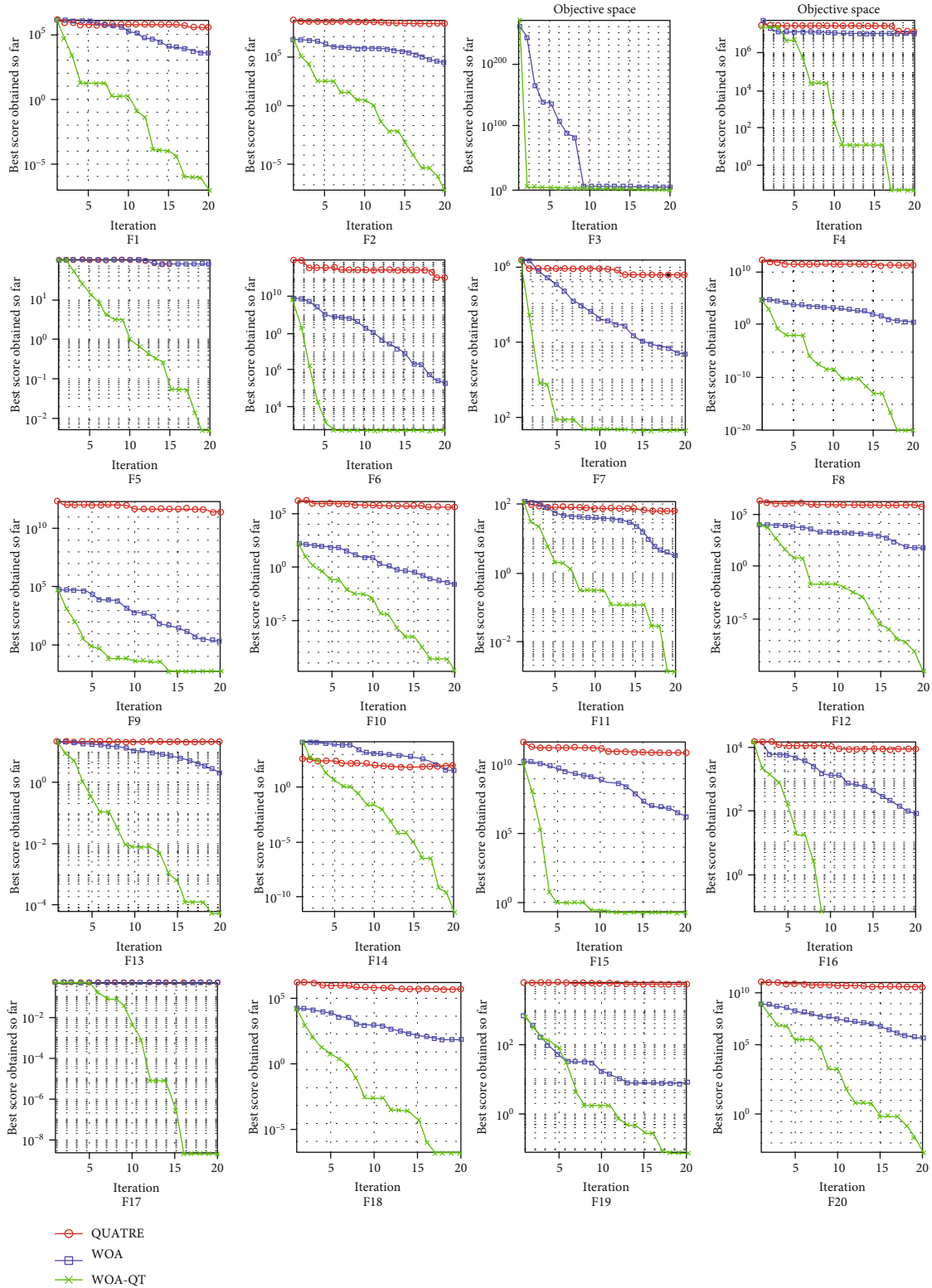
Figure 2: The convergence curves of 500-dimensional functions (F1-F20).

RSSI has emerged. The ordinary centroid location algorithm considers that all beacon nodes have equal status. In fact, different beacon nodes have different influences on the location of unknown nodes. The WCL algorithm is to reflect the degree of influence of each beacon node on the position of the centroid (i.e., the unknown node) through the weight factor. The triangle centroid positioning algorithm is used here. The pairwise distance between
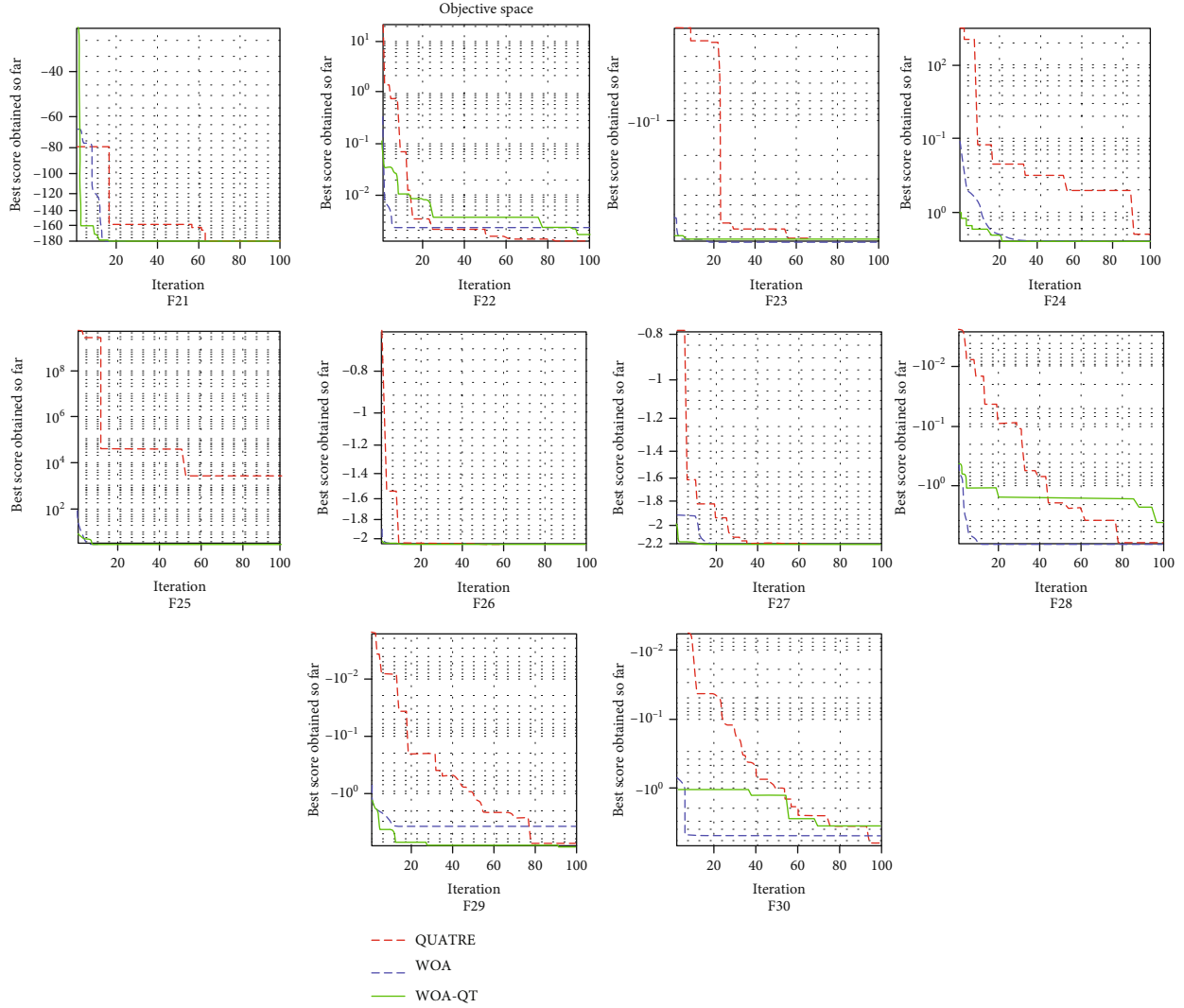
FIGURE 3: The convergence curves of fixed-dimension functions (F21-F30).

the unknown node and the three beacon nodes is recorded as (ρ_1, ρ_2, and ρ_3), which are used to construct the weight coefficient. The relevant formulas are as follows:

$$x_l = \frac{x_1/(\rho_1 + \rho_2) + x_2/(\rho_2 + \rho_3) + x_3/(\rho_1 + \rho_3)}{1/(\rho_1 + \rho_2) + 1/(\rho_2 + \rho_3) + 1/(\rho_1 + \rho_3)},$$
$$y_l = \frac{y_1/(\rho_1 + \rho_2) + y_2/(\rho_2 + \rho_3) + y_3/(\rho_1 + \rho_3)}{1/(\rho_1 + \rho_2) + 1/(\rho_2 + \rho_3) + 1/(\rho_1 + \rho_3)}. \quad (17)$$

$S_1(x_1, y_1)$, $S_2(x_2, y_2)$, and $S_3(x_3, y_3)$ represent three beacon nodes, and the coordinate of the centroid of the triangle enclosed by them is the position of unknown nodes $N_l(x_l, y_l)$.

WCL_RSSI has the advantages of low computational complexity, simple implementation and high location accuracy. Moreover, the location process is less affected by the change of the transmission environment and does not require interaction between nodes [32].

5.3. Apply the Proposed WOA-QT Algorithm to WCL_RSSI. This article uses WOA-QT to improve WCL_RSSI in order to obtain the node position with higher accuracy. In the two-dimensional plane, the solution of the WOA-QT corresponds to the candidate position coordinates of the unknown node, which is denoted as $N_l(x, y)$. The Euclidean distance between $N_l(x, y)$ and the beacon node $S_i(x_i, y_i)$ is $d_o = \sqrt{(x - x_i)^2 + (y - y_i)^2}$. $N_l(x_w, y_w)$ represents the location of unknown node estimated by the WCL_RSSI. Similarly, the distance between $N_l(x_w, y_w)$ and $S_i(x_i, y_i)$ is $d_w = \sqrt{(x_w - x_i)^2 + (y_w - y_i)^2}$. Obviously, there is a deviation between d_o and d_w. It should be noted that the connection path between the unknown node and the beacon node has a great influence on the location accuracy. The greater the number of hops, the smaller the positioning error tends to be. The weight of hops needs to be considered when

TABLE 4: The average distance error of each algorism.

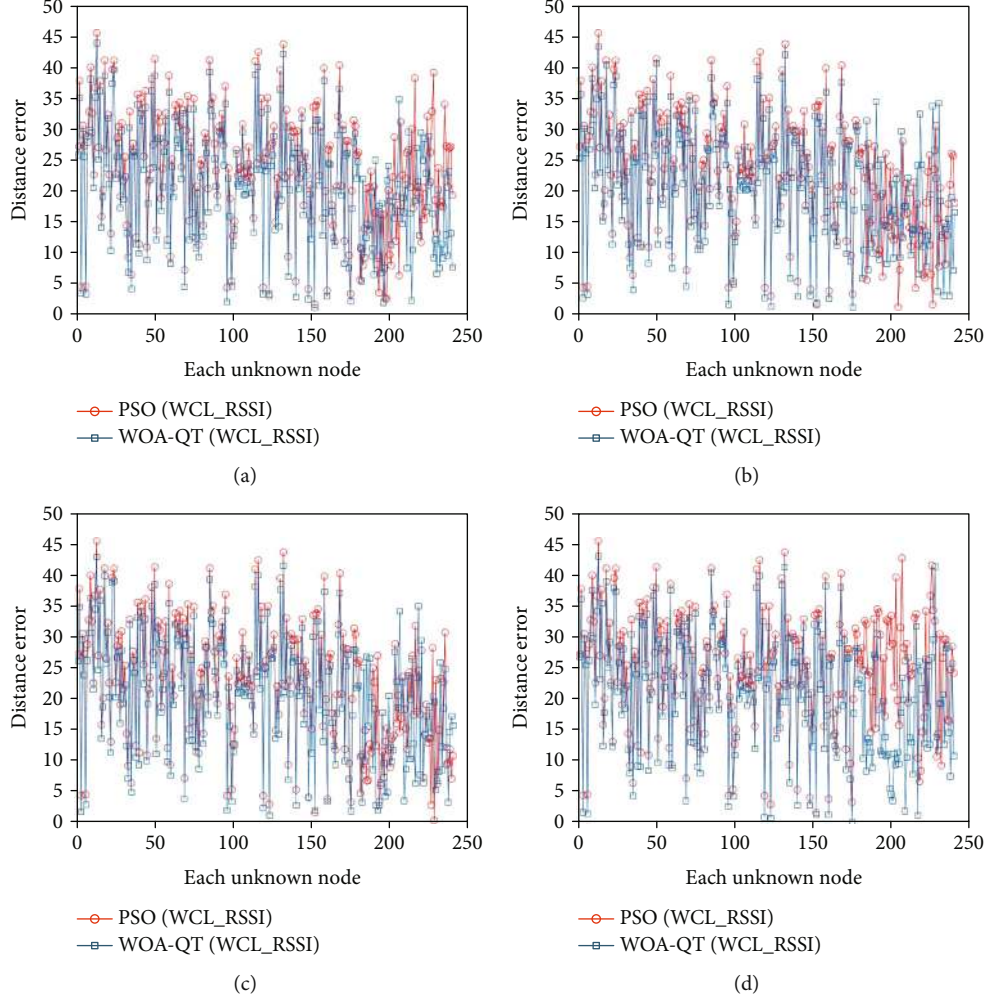| WOA-QT(WCL_RSSI)	PSO(WCL_RSSI)	WOA(WCL_RSSI)	QUATRE(WCL_RSSI)	WCL_RSSI
20.7632	23.315	22.7465	22.6332	24.9505

(a)

(b)

(c)

(d)

FIGURE 4: The comparison of location errors. (a) PSO and WOA-QT. (b) QUATRE and WOA-QT. (c) WOA and WOA-QT. (d) WCL based on RSSI and WOA-QT.

measuring the distance error. Therefore, the evaluation function $f(x, y)$ is as follows:

$$f(x, y) = \min \left(\sqrt{\frac{1}{K} \sum_{i=1}^{K} \left(\frac{d_o - d_w}{\text{hop}_i} \right)^2} \right). \qquad (18)$$

There are K beacon nodes. The hop number between the beacon node S_i and the unknown node is recorded as hop_i. Equation (18) expresses the optimization goal, that is, to minimize the root mean square error of the difference between the two distances.

5.4. Simulation Results. Assume that in a two-dimensional area of $1000\,\text{m} \times 1000\,\text{m}$, 60 beacon nodes are deployed, and the positions of 240 unknown nodes are to be deter-

mined. The communication radius of sensor node is 15 meters, and the maximum number of iterations of the algorithm is 400. Table 4 lists the average distance error between the unknown node position estimated by each algorithm and the true position. The simulation experiment involves five algorithms. Simulation results show that WCL_RSSI based on WOA-QT has the smallest error, and the positioning accuracy is improved obviously. In order to visualize the positioning error, Figure 4 marks the distance deviation between the estimated position of each unknown node and its true position.

6. Conclusion

AI and IoT are compatible and symbiotic, who are born to be the best partner. They have broken through the bottlenecks

of their respective parallel developments, enabling high-frequency resonance and coordinated development.

In this paper, the optimization algorithm in AI is used to solve the node location problem in WSN which is the key technology of IoT. There are two optimization algorithms involved: WOA and QUATRE. WOA is derived from natural wisdom, and the QUARTRE has the beauty of mathematics. The two algorithms utilize each other and work closely together to form an organic whole. The overall optimization performance is much better than that when the two are separated. The potential of local exploitation is further tapped through the mutual fusion, so that the local stagnation is effectively avoided in the process of searching for the global optimal. Although algorithm hybridization will inevitably bring more calculations, but with the cooperation of the two algorithms and the DSS strategy, the time efficiency of the algorithm has not been affected, and even better in many cases. So the increase in complexity of this algorithm is worthwhile. With the help of optimization technology, the node location accuracy of WSN has been significantly improved. It can be said that a successful AIoT case with application value is realized.

The QUATRE algorithm has many variants. When combined with other algorithms, we can consider how to choose different variants adaptively [33]. In addition, the complexity caused by algorithm hybridization can be considered to use surrogate-assisted technology to ease [34, 35]. In the field of WSN, there are many challenging problems that need to be overcome [36, 37]. It can be combined with other AI technologies such as machine learning [38–40] to promote the birth of more application scenarios of AIoT.

Data Availability

1. 30 well-known benchmark functions. 2. Random position location of sensor nodes by computers.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] J. Gubbi, B. Rajkumar, M. Slaven, and P. Marimuthu, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[2] A. Triantafyllou, P. Sarigiann, and T. D. Lagkas, "Network Protocols, Schemes, and Mechanisms for Internet of Things (IoT): Features, open Challenges, and Trends," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 5349894, 24 pages, 2018.

[3] K. Deb, A. Samir, P. Amrit, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[4] K. V. Price, *Differential Evolution*, Springer, Berlin, Heidelberg, 2013.

[5] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-International Conference on Neural Networks*, Perth, Australia, November 1995.

[6] D. Karaboga and C. Ozturk, "A novel clustering approach: artificial bee colony (ABC) algorithm," *Applied Soft Computing*, vol. 11, no. 1, pp. 652–657, 2011.

[7] P. W. Tsai, M. K. Khan, J.-S. Pan, and B. Y. Liao, "Interactive artificial bee colony supported passive continuous authentication system," *IEEE Systems Journal*, vol. 8, no. 2, pp. 395–405, 2014.

[8] S.-C. Chu, P. W. Tsai, and J.-S. Tsai, "Cat swarm optimization," in *Pacific Rim international conference on artificial intelligence*, Springer, Berlin, Heidelberg, August, 2006.

[9] P. W. Tsai, J.-S. Pan, S. M. Chen, and B. Y. Liao, "Enhanced parallel cat swarm optimization based on the Taguchi method," *Expert Systems with Applications*, vol. 39, no. 7, pp. 6309–6319, 2012.

[10] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, no. 5, pp. 51–67, 2016.

[11] J.-S. Pan, J.-L. Liu, and E.-J. Liu, "Rank-based whale optimization algorithm for solving parameter optimization of solar cells," *International Journal of Modeling and Optimization*, vol. 9, no. 4, pp. 209–215, 2019.

[12] A. E. Ezugwu and D. Prayogo, "Symbiotic organisms search algorithm: theory, recent advances and applications," *Expert Systems with Applications*, vol. 119, pp. 184–209, 2019.

[13] S.-C. Chu, Z. G. Du, and J.-S. Pan, "Symbiotic Organism Search Algorithm with Multi-Group Quantum-Behavior Communication Scheme Applied in Wireless Sensor Networks," *Applied Sciences*, vol. 10, no. 3, p. 930, 2020.

[14] H. Abdolreza, "Black hole: a new heuristic optimization approach for data clustering," *Information Sciences*, vol. 222, pp. 175–184, 2013.

[15] J.-S. Pan, Q. W. Chai, S.-C. Chu, and N. X. Wu, "3-D Terrain Node Coverage of Wireless Sensor Network Using Enhanced Black Hole Algorithm," *Sensors*, vol. 20, no. 8, p. 2411, 2020.

[16] Z. Y. Meng, J.-S. Pan, and H. Xu, "QUasi-Affine TRansformation evolutionary (QUATRE) algorithm: a cooperative swarm based algorithm for global optimization," *Knowledge-Based Systems*, vol. 109, pp. 104–121, 2016.

[17] Z. Y. Meng and J.-S. Pan, "QUasi-Affine TRansformation evolution with external ARchive (QUATRE-EAR): an enhanced structure for differential evolution," *Knowledge-Based Systems*, vol. 155, pp. 35–53, 2018.

[18] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997.

[19] T. T. Nguyen, J.-S. Pan, and T. K. Dao, "An improved flower pollination algorithm for optimizing layouts of nodes in wireless sensor network," *IEEE Access*, vol. 7, pp. 75985–75998, 2019.

[20] J.-S. Pan, P. Hu, and S.-C. Chu, "Novel parallel heterogeneous meta-heuristic and its communication strategies for the prediction of wind power," *PRO*, vol. 7, no. 11, pp. 845–868, 2019.

[21] P. Hu, J.-S. Pan, S.-C. Chu, Q. W. Chai, T. Liu, and Z. C. Li, "New Hybrid Algorithms for Prediction of Daily Load of Power Network," *Applied Sciences*, vol. 9, no. 21, p. 4514, 2019.

[22] J.-S. Pan, P. C. Song, S.-C. Chu, and Y. J. Peng, "Improved Compact Cuckoo Search Algorithm Applied to Location of Drone Logistics Hub," *Mathematics*, vol. 8, no. 3, p. 333, 2020.

[23] M. Rout and R. Roy, "Optimal wireless sensor network information coverage using particle swarm optimisation method," *International Journal of Electronics Letters*, vol. 5, no. 4, pp. 491–499, 2016.

[24] T.-W. Sung and C.-S. Yang, "Distributed Voronoi-based self-redeployment for coverage enhancement in a mobile directional sensor network," *International Journal of Distributed Sensor Networks*, vol. 9, no. 11, Article ID 165498, 2016.

[25] S.-C. Chu, T. K. Dao, and J.-S. Pan, "Identifying correctness data scheme for aggregating data in cluster heads of wireless sensor network based on naive Bayes classification," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, pp. 1–15, 2020.

[26] T. K. Dao, T. T. Nguyen, J.-S. Pan, Y. Qiao, and Q.-A. Lai, "Identification failure data for cluster heads aggregation in WSN based on improving classification of SVM," *IEEE Access*, vol. 8, pp. 61070–61084, 2020.

[27] P. W. Tsai and C.-W. Chen, "Review on swarm intelligence for optimization," *Computing Science and Technology International Journal*, vol. 2, no. 1, pp. 13–17, 2014.

[28] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2, pp. 82–102, 1999.

[29] S. Das, S. S. Mullick, and P. N. Suganthan, "Recent advances in differential evolution – An updated survey," *Swarm and Evolutionary Computation*, vol. 27, pp. 1–30, 2016.

[30] Q. Fan, X. F. Yan, and Y. Xue, "Prior knowledge guided differential evolution," *Soft Computing*, vol. 21, no. 22, pp. 6841–6858, 2016.

[31] S. Gupta, K. Deep, S. Mirjalili, and J. H. Kim, "A modified sine cosine algorithm with novel transition parameter and mutation operator for global optimization," *Expert Systems with Applications*, vol. 154, article 113395, 2020.

[32] J. Wang, P. Urriza, Y. X. Han, and D. Cabric, "Weighted centroid localization algorithm: theoretical analysis and distributed implementation," *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3403–3413, 2011.

[33] S. L. Wang, T. F. Ng, and F. Morsidi, "Self-adaptive ensemble based differential evolution," *International Journal of Machine Learning and Computing*, vol. 8, no. 3, pp. 286–293, 2018.

[34] C. L. Sun, Y. C. Jin, R. Cheng, J. L. Ding, and J. C. Zeng, "Surrogate-assisted cooperative swarm optimization of high-dimensional expensive problems," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 4, pp. 644–660, 2017.

[35] S. F. Qin, C. L. Sun, Y. C. Jin, and G. C. Zhang, "Bayesian approaches to surrogate-assisted evolutionary multi-objective optimization: a comparative study," in *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, Xiamen, China, December 2019.

[36] V. Sadhu, X. Y. Zhao, and D. Pompili, "Energy-efficient analog sensing for large-scale and high-density persistent wireless monitoring," *IEEE Internet of Things Journal*, vol. 7, p. 1, 2020.

[37] M. T. Lazarescu, "Design of a WSN platform for long-term environmental monitoring for IoT applications," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 3, no. 1, pp. 45–54, 2013.

[38] M. A. Alsheikh, S. W. Lin, D. Niyato, and H.-P. Tan, "Machine learning in wireless sensor networks: algorithms, strategies, and applications," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 1996–2018, 2019.

[39] K. Thangaramya, K. Kulothungan, R. Logambigai, M. Selvi, S. Ganapathy, and A. Kannan, "Energy aware cluster and Neuro-fuzzy based routing algorithm for wireless sensor networks in IoT," *Computer Networks*, vol. 151, pp. 211–223, 2019.

[40] L. F. Gou, H. H. Li, H. Zheng, H. C. Li, and X. X. Pei, "Aeroengine control system sensor fault diagnosis based on CWT and CNN," *Mathematical Problems in Engineering*, vol. 2020, Article ID 5357146, 12 pages, 2020.