IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 22 February 2024, accepted 8 April 2024, date of publication 15 April 2024, date of current version 3 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3388497

RESEARCH ARTICLE

Range-Free Localization Scheme Based on a Reconstructed Marine Predators Algorithm With Adaptive Enhancement for Wireless Sensor Networks

DAN YU, TING YUAN, PAN LI, XINZHONG LIU, WENWU XIE, AND ZHIHE YANG
School of Information Science and Engineering, Hunan Institute of Science and Technology, Yueyang 414006, China

Corresponding authors: Xinzhong Liu (lxz_jlu@163.com) and Zhihe Yang (kjc@hnist.edu.cn)

This work was supported in part by the Natural Science Foundation of Hunan Province under Grant 2023JJ50045, Grant 2023JJ50046, Grant 2024JJ7218, and Grant 2024JJ7219; in part by the National Natural Science Foundation of China under Grant 62372070; and in part by Hunan Provincial Department of Education Project under Grant 23C0217 and Grant 22B0676.

ABSTRACT Wireless sensor networks (WSN) are widely utilized in all walks of industries, and node localization is still essential as one of the fundamental functionalities. Moreover, the marine predators algorithm (MPA), as one of the swarm intelligence optimization algorithms, has proven to possess strong search capability and high convergence speed. Therefore, for the poor optimization ability of the least squares method in the DV-Hop method, a reconstructed marine predator algorithm with adaptive enhancement (RMPA-AE) for WSN node localization is proposed in this paper. Firstly, a population diversity expression based on Euclidean distance is proposed to reconstruct the phases of the algorithm. Then, an adaptive enhancement strategy is proposed to improve the local exploitation ability of the algorithm with respect to its tendency to fall into local optimality. Subsequently, a global perturbation strategy based on the iteration number is proposed for the re-generation of the population at FADs to achieve a wide range of individual jumps. Finally, the experimental results for 26 benchmark functions demonstrate the search capability of the proposed RMPA-AE algorithm, and the superiority of the proposed algorithm compared to the state-of-the-art algorithms. The feasibility of the proposed algorithm is also demonstrated in the node localization simulation experiments.

INDEX TERMS Wireless sensor networks, marine predators algorithm, DV-hop, Euclidean distance, phase reconstruct.

I. INTRODUCTION

As the global urbanization process accelerates, the concept of smart cities has emerged [1], [2], aimed at improving urban management efficiency, enhancing the quality of city services, and promoting sustainable development through advanced information technology and data analysis. Among numerous technologies, wireless sensor node localization plays a crucial role in the construction of smart cities. This technology not only promotes data-driven urban decision-making but also significantly enhances the interconnectivity and intelligence level of urban systems.With the continuous advancement of technology, wireless sensor nodes have become an essential tool for collecting and transmitting key urban data. These sensors are capable of real-time monitoring and collecting data on the environment, traffic, energy usage, and more, across various corners of the city, providing valuable information support for urban management and decision-making. From real-time monitoring of traffic flow to continuous tracking of environmental quality, from enhancing public safety to optimizing energy management, the application of wireless sensor nodes covers

The associate editor coordinating the review of this manuscript and approving it for publication was Chenshu Wu.

every aspect of smart cities [3], [4], [5], [6], paving the way for more efficient and sustainable urban living. Therefore, obtaining accurate location information of sensor nodes becomes crucial. This is not only because location data is essential for effectively interpreting and applying the information collected by sensors, but also because location information significantly impacts optimizing the layout and maintenance of sensor networks, enhancing data accuracy and reliability.

In sensor network node localization, there are range-based and range-free methods. Range-based localization techniques typically measure the distance between sensors using parameters such as Ultra-Wideband (UWB) [7], [8], [9], Amplitude of Received Signal (ARS) [10], and Time Difference of Arrival (TDOA) [11], [12]. Although these methods offer high accuracy, they require higher costs. On the other hand, range-free localization techniques usually estimate positions using the topological relationships between neighboring nodes. The widely studied DV-Hop algorithm is one of the range-free localization methods. Yanfei et al. [13]. proposed DV-hop positioning for mobile anchor nodes in wireless sensor networks. Gui et al. [14]. introduced two improved algorithms (Checkout DV-hop and Selective 3-Anchor DV-hop) to enhance localization performance. Wang et al. [15]. created a NSGA-II-based multi-objective DV-Hop localization algorithm that implemented an enhanced constraint strategy using all beacon nodes to enhance the precision of DV-Hop localization estimations. Liu et al. [16]. suggested an enhanced DV-Hop algorithm for wireless sensor networks using neural dynamics and conducted error analysis to determine the range of distance error fluctuations between unknown nodes and anchor nodes. Gui et al. [17] proposed a DV-Hop algorithm based on centralized connectivity to optimize the accuracy of DV-Hop positioning (CCDV-Hop), and established an optimization problem constrained by the actual connectivity between any two nodes, which can make the positioning results conform to the actual connectivity.

These algorithms have achieved certain success in terms of localization accuracy. The DV-Hop algorithm, which estimates the distance of each hop through the average hop distance, is more suitable for isotropic networks and performs poorly in anisotropic networks. To address such issues, scholars have improved DV-Hop using intelligent optimization algorithms to achieve high-precision positioning. Intelligent optimization algorithms are developed inspired by biological, behavioral, or processes in nature, such as the Marine Predators Algorithm (MPA) [18], [19], Cuckoo Search algorithm (CS) [20], [21], [22], Flower Pollination Algorithm (FPA) [23], Particle Swarm Optimization algorithm (PSO) [24], [25], etc. These algorithms can effectively find optimal solutions through evolutionary processes or social behavior and are applied in various fields. Surya and Ravi [26]. proposed a wireless sensor health monitoring system based on MPSO to identify faulty sensor nodes, thereby enhancing sensor reliability. Yang et al. [27]. presented an enhanced DV-HOP model, known as PSO-DV-Hop, which incorporates the Particle Swarm Optimization algorithm for improving smart campus safety. This enhancement aims to enhance the safety performance and efficiency of the smart campus. Song et al. [28]. suggested an enhanced positioning algorithm referred to as MGDV-Hop, utilizing a hybrid chaotic strategy to expand positioning coverage while simultaneously reducing and balancing energy consumption.

As one of the algorithms with good optimization effect, the MPA algorithm has attracted much attention from scholars. Abd Elminaam et al. [29]. proposed a hybrid algorithm for MPA and k-nearest neighbors (MPA-KNN) for evaluating selected features ranging in size from small to large on a medical dataset. Chen et al. [30]. proposed a new and improved Q-learning-based MPA Hybridization Algorithm (QMPA) to use reinforcement learning for choosing the optimal location update strategy for search agents in various iterative stages and states. Fan et al. [31] proposed an improved MPA and proposed a logical adversarial-based learning mechanism (LOBL) to improve population diversity and generate more accurate solutions. Zhong et al. [32] proposed a newly proposed Multi-Objective Ocean Predation Algorithm (MOMPA), called the Multi-Objective Ocean Predation Algorithm (MOMPA), which introduces an external archiving component to store the currently known non-inferior Pareto optimal solution. Although these improved MPA algorithms have achieved certain results in their application, many of them employ additional optimization methods to enhance algorithm accuracy during the refinement process, and ignore the innovative research of the algorithm itself.

In this paper, a Reconstructed Marine Predators Algorithm with Adaptive Enhancement (RMPA-AE) for Nodes Location of WSN is proposed. According to the shortcomings of the MPA algorithm, such as the defects of phasing and the unreasonable allocation of policy updates, we reconstruct the MPA algorithm and adjust the update strategy. We reconstruct the renewal phase of the population through population diversity, avoiding the iteration division that limits the search ability of Brownian motion and Lévy flight. At the same time, in phase 2, we removed the original population division into the early and later phases and reconstructed it based on the iterative cycle. The advantage of the long and short steps of Lévy flight was brought into play. Second, we have adjusted the update strategy for phase 2 and 3 of the traditional MPA. In phase 2, a kind of Lévy flight based on the number of iterations is proposed, and in phase 3, the position is updated based on the current position and the individual best solution's position, and the position is updated with that of another random individual to increase the algorithm's chances of converging to the global optimum. Finally, when considering the effects of eddy formation and fish aggregation device effects on the behavior and activity patterns of marine predators, we suggest updating the current individual's position based on the difference between the

positions of two randomly selected individuals within the current population, instead of the MPA randomly generated individual positions as the updated positions. The problem of lack of clear guidance on the direction of the search is avoided. The contributions made in this article are as follows:

- We propose a method to reconstruct the phases of the algorithm based on population diversity, which gives full play to the search ability of Brownian motion and Lévy flight. At the same time, phase 2 is also divided into early and later phase through the iterative cycle, making full use of the long and short step characteristics of the Lévy strategy.
- For the update strategy in phases 2 and 3, we propose the adaptive enhancement strategy to update the population position.
- We propose to use the differences between random individuals to guide the FADs effect to renew the population.

We employ 26 benchmark functions to evaluate the algorithm and validate its positioning accuracy through simulation experiments. Extensive experiments demonstrate that the proposed algorithm exhibits both high positioning accuracy and robustness. In the section II, the DV-Hop algorithm is introduced. We describe the traditional MPA in Section IV. Section V introduces the proposed algorithm. Section VI demonstrates the benchmark function test experiment and simulation experiment. Finally, conclusion is presented in Section VII.

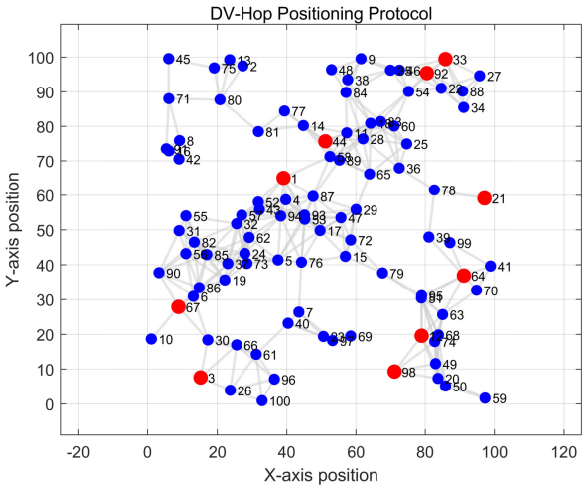

FIGURE 1. DV-Hop positioning protocol.

II. POSITIONING PROTOCOLS

During the experiment, information about other nodes is usually obtained by exchanging data frames and using specific positioning protocols. Nodes broadcast data frames containing their own information. Neighboring nodes receive these broadcast messages, thereby obtaining information about the sending nodes. The DV-Hop positioning protocol is used to process the exchanged information. In this positioning protocol, each node calculates the minimum number of hops to the anchor nodes, and the anchor nodes exchange information by estimating the average distance per hop. Non-anchor nodes can then use this information to estimate their distance to each anchor node and calculate their position based on this information. The experimental part is completed based on the above protocols.

As shown in Fig.1, the network diagram of the DV-Hop positioning protocol is drawn. For each pair of nodes, if the distance between them is less than or equal to the communication radius, a line is drawn to connect them, indicating that the two nodes can communicate directly. These connections are indicated by light gray lines to demonstrate the connectivity of the network.

The DV-Hop positioning algorithm stands out as the most commonly employed localization technique within the realm of Wireless Ad-hoc Network positioning systems. The positioning procedure is independent of the ranging technique, leveraging multi-hop beacon node data for node positioning, offering extensive coverage. The algorithm can be delineated into three distinct stages.

(1): Determine the minimum hop count

Beacon nodes transmit packets containing their location information to neighboring nodes. The receiving node then stores and shares the minimum hop count information to neighboring nodes, allowing all network nodes to have records of their minimum hop counts to each beacon node.

(2): Calculate the actual hopping distance

Each beacon node calculates the estimated average physical distance for each hop using its position information and the hop count recorded during phase 1.

$$\text{Hopsize}_i = \frac{\sum_{j\neq i}\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{j\neq i} h_j}, \tag{1}$$

where (x_i, y_i) and (x_j, y_j) are the coordinates of beacon nodes i and j, and h_j is the number of hops between beacon nodes i and j. Then,the beacon node broadcasts the computed average hop distance via packets with lifetime indicators, and upon receiving this average hop distance, the unknown node computes the distances between each beacon node using the recorded hop counts.

(3): Calculate position

The unknown node uses the jump distance recorded in the second stage to each beacon node, and uses the maximum likelihood estimation method to calculate its own coordinates. n nodes such as 1, 2, and 3 are the coordinates of the anchor node, and the distance from them to the unknown node D determines the coordinates of node D.

$$\begin{cases} x_1^2 - x_2^2 - 2(x_1 - x_n)x + y_1^2 - y_n^2 \\ -2(y_1 - y_n)y = d_1^2 - d_n^2, \\ \quad \cdots \\ x_{n-1}^2 - x_2^2 - 2(x_{n-1} - x_n)x + y_{n-1}^2 \\ -y_n^2 - 2(y_{n-1} - y_n)y = d_{n-1}^2 - d_n^2, \end{cases} \tag{2}$$

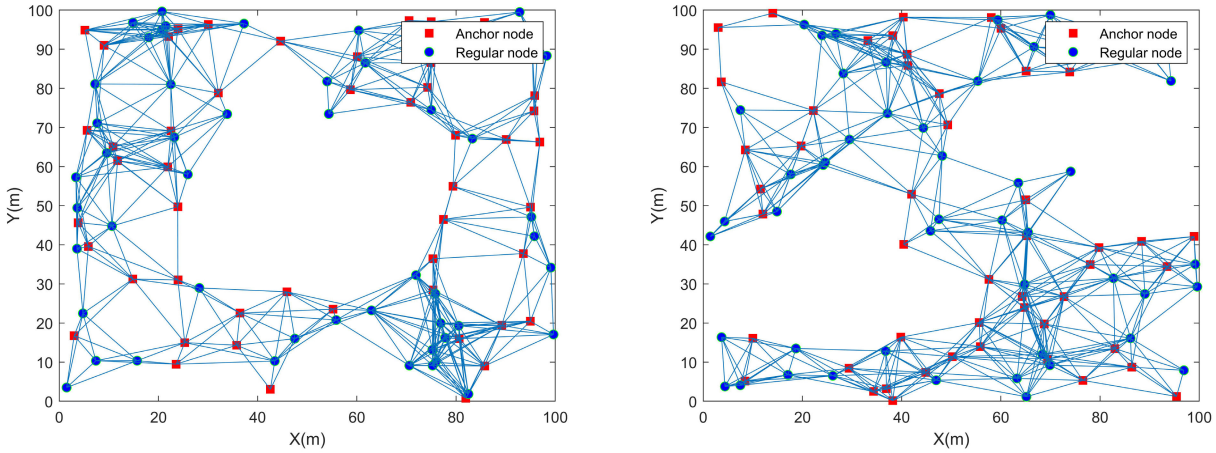

FIGURE 2. Node connectivity for O-type and S-type network structures, R=20m.

Further conversions:

$$A = -2\begin{bmatrix} x_1 - x_n & y_1 - y_n \\ x_2 - x_n & y_2 - y_n \\ \cdots & \cdots \\ x_{n-1} - x_n & y_{n-1} - y_n \end{bmatrix}, \tag{3}$$

$$B = \begin{bmatrix} d_1^2 - d_n^2 - x_1^2 + x_n^2 - y_2^2 + y_n^2 \\ d_2^2 - d_n^2 - x_2^2 + x_n^2 - y_2^2 + y_n^2 \\ \cdots \\ d_{n-1}^2 - d_n^2 - x_{n-1}^2 + x_n^2 - y_{n-1}^2 + y_n^2 \end{bmatrix},$$

$$X = \begin{bmatrix} x \\ y \end{bmatrix}. \tag{4}$$

Using the standard least squares method, the coordinates of the unknown node can be derived as:

$$\hat{X} = (A^T A)^{-1} A^T B. \tag{5}$$

III. NODE CONNECTIVITY

In the context of node localization in Wireless Sensor Networks (WSN), the connectivity of the network is crucial to ensure that all nodes can effectively communicate and perform localization computations. Particularly, when discussing the impact of network topology shapes on connectivity, the "O" and "S" type layouts are two specific topological structures that have unique considerations in design and application. As illustrated in fig. 2, having only one connected component in these two types of network layouts means that despite the different physical layouts and shapes of the network, the entire network can still maintain full connectivity. Any two nodes in the network are connected by at least one path, allowing information to flow freely between any two points.

In "O" and "S" type layouts, which may be used to surround a specific area or object for monitoring and data collection, the physical shape might result in greater distances between nodes at the edges of the network. However, as long as each node is within its communication range of its neighboring nodes, the entire network can remain connected.

In networks with nodes randomly generated by the DV-Hop method, despite the significant differences in physical layout, the communication links between nodes are effectively established. This aspect is crucial for ensuring the functionality of the network, whether for node localization, data collection, or other applications.

IV. MARINE PREDATORS ALGORITHM

Marine Predators Algorithm is a novel population-based swarm intelligence optimization algorithm proposed by Afshin Faramarzi inspired by the predation process of marine predators. Based on the predation process of marine animals, the optimization process of the algorithm is divided into three distinct stages., and different methods are used to update the population, as follows:

Phase 1: When the current iteration count t is below the maximum allowable iterations T ($t < T/3$), the position is updated according to Eq. (6):

$$\begin{aligned} \boldsymbol{s} &= \mu_{\boldsymbol{B}} \otimes (\boldsymbol{E_i} - \mu_{\boldsymbol{B}} \otimes \boldsymbol{X_i}), i = 1, \ldots n, \\ \boldsymbol{X_i} &= \boldsymbol{E_i} + P \cdot \boldsymbol{R} \otimes \boldsymbol{s}, \end{aligned} \tag{6}$$

where μ_B and $\boldsymbol{R}$ are the random number vectors of Brownian motion and the random number vectors in [0,1], respectively. $\boldsymbol{E_i}$ is the $n \times d$ predator matrix, n is the number of search agents, and d is the dimension. P is a constant of 0.5. The model of the prey's movement is represented by. At this time, a higher speed of movement allows for a high ability to explore.

Phase 2: When $\frac{T}{3} < t < \frac{2T}{3}$, it is divided into two phases according to the cycle:

$$\boldsymbol{X}_i = \begin{cases} \boldsymbol{X_i} + P \cdot \boldsymbol{R} \otimes \boldsymbol{s}, \boldsymbol{s} = \mu_{\boldsymbol{L}} \otimes (\boldsymbol{E_i} - \mu_{\boldsymbol{L}} \otimes \boldsymbol{X_i}), & \frac{T}{3} < t \leq \frac{n}{2}, \\ \boldsymbol{E_i} + P \cdot C \otimes \boldsymbol{s}, \boldsymbol{s} = \mu_{\boldsymbol{B}} \otimes (\mu_{\boldsymbol{B}} \otimes \boldsymbol{E_i} - \boldsymbol{X_i}), & \frac{n}{2} < t < \frac{2T}{3}, \end{cases} \tag{7}$$

where μ_L is the random number vector of the Lévy flight. The model that indicates the movement of the predator in Brownian motion is $\mu_B \otimes \boldsymbol{X_p^i}$, $C = (1 - t/T)^{2t/T}$ an adaptive parameter for controlling the predator's movement step size, and n is the population number.

Phase 3: When $t > \frac{2T}{3}$, the best predatory strategy for predators is to use Lévy flight as the main movement mode:

$$\begin{aligned} \boldsymbol{s} &= \mu_{\boldsymbol{L}} \otimes (\mu_{\boldsymbol{L}} \otimes \boldsymbol{E_i} - \boldsymbol{X_i}), i = 1, \ldots n, \\ \boldsymbol{X_i} &= \boldsymbol{X_i} + P \cdot C \otimes \boldsymbol{s}, \end{aligned} \tag{8}$$

where $\mu_{\boldsymbol{L}} \otimes \boldsymbol{X_i}$ symbolizes the model of the predator moving in Lévy.

The effects of eddy formation and fish aggregation effects on the behavior and activity patterns of marine predators are also considered. Model such effects as follows Eq. (9):

$$\boldsymbol{X_i} = \begin{cases} \boldsymbol{X_i} + C\left[\boldsymbol{X_{\min}} + \boldsymbol{R} \otimes (\boldsymbol{X_{\max}} - \boldsymbol{X_{\min}})\right] \otimes \boldsymbol{U}, & r \leq \text{FADs}, \\ \boldsymbol{X_i} + [\text{FADs}(1 - r) + r]\,(\boldsymbol{X_{r_1}} - \boldsymbol{X_{r_2}}), & r > \text{FADs}, \end{cases} \tag{9}$$

where FADs is the probability of the effect of eddy formation and fish aggregation on predators, set to 0.2. $\boldsymbol{U}$ is a randomly generated vector in [0, 1], taking 0 when the element value is less than 0.2 and 1 vice versa. r is a uniformly distributed random number within the range [0, 1]. $\boldsymbol{X_{\min}}$ and $\boldsymbol{X_{\max}}$ are vectors contain the lower and upper bounds of each dimension.

V. PROPOSED RMPA-AE METHOD

A. POPULATION DIVERSITY

In swarm intelligence optimization algorithms, population diversity represents the degree of difference between individuals in the population. It is one of the criteria for evaluating the quality of an algorithm and also one of the ways to express individual positional changes during the iteration process. This article evaluates population diversity in the search space by measuring the deviation of each individual from the population average across all dimensions.

First, calculate the average value of the population in each dimension, as shown in Eq. (10):

$$\text{mean}_\text{j} = \frac{1}{n} \sum_{i=1}^{n} x_{ij}, \tag{10}$$

where x_{ij} represents the value of the i-th individual in the population. in the j-th dimension, and meanj is the average of the j-dimension.

Secondly, the absolute difference between the value of each individual in the population and the mean value of the corresponding dimension is calculated, and normalized deviation maximum and minimum, as shown in Eq. (11) and Eq. (12):

$$d_i = \sum_{j=1}^{n} \left| x_{ij} - \text{mean}_\text{j} \right|, \tag{11}$$

$$\begin{cases} d_{\max} = \max(\{d_i\}_{i=1}^{n}), \\ d_{\min} = \min(\{d_i\}_{i=1}^{n}). \end{cases} \tag{12}$$

where d_i is the sum of dimensional differences of the i-th individual.

Finally, normalize the diversity score as shown in Eq. (13):

$$p_i = \frac{d_i - d_{\min}}{d_{\max} - d_{\min}}. \tag{13}$$

B. PHASE RECONSTRUCTED

The traditional MPA algorithm divides the process into three phases based on the number of iterations, with 1/3 and 2/3 of the iterations serving as the boundaries. When the number of iterations is less than 1/3, MPA relies solely on Brownian motion to update the population. Brownian motion, which primarily involves small-scale perturbations, can lead to the population becoming overly concentrated in a specific region of the search space during the iterative process, making it challenging for the algorithm to break free from local optima. Although as the number of iterations increases, the MPA algorithm updates the population by leveraging the advantages of combining long and short steps in Lévy flight, it neglects the capability of Brownian motion for local perturbations, further limiting the expression of the algorithm's search ability. This means that dividing phases based on the number of iterations restricts the search abilities of both Brownian motion and Lévy flight. Therefore, we propose a phase reconstruction strategy based on population diversity, and reconstruct the algorithm into three phases with $p/3$ and $2p/3$ as boundaries, as shown in algorithm 1.

Algorithm 1 Phase Reconstructed Algorithm

```
if Current population diversity p_i < p/3 then
    Updating the population's position during phase 1.
else if p/3 < p_i < 2 × p/3 then
    if t < T/2 then
        Updating the population's position in the early stage
        of phase 2.
    else if t > T/2 then
        Updating the population's position in the later stage
        of phase 2.
    end if
else if p_i > 2 × p/3 then
    Updating the population's position during phase 3.
end if
Populations are updated according to FADs.
```

In phase 2 of the traditional MPA algorithm, the population is divided into two parts, each updated based on Eq. (7). However, although this phase combines the advantages of

Brownian motion and Lévy flight, relying more on Lévy flight in the early iterations can help increase population diversity, while utilizing Brownian motion in the later iterations can improve the algorithm's ability for local exploitation. Therefore, in phase 2, we used the number of iterations as the update criterion, and reconstructed phase 2 into early and late phases with $T/2$ as the boundary. Populations are updated based on Eq. (14), respectively.

C. ADAPTIVE ENHANCEMENT STRATEGY

In phase 2 of the traditional MPA algorithm, the population is split into two segments, and the individuals rely on Brownian motion and Lévy flight to update their positions, but ignore the role of the number of iterations as the search process deepens. Therefore, a Lévy flight based on the number of iterations is proposed to meet the requirements of combining the Lévy flight step length with the evolutionary situation, as shown in Eq. (14):

$$\boldsymbol{X_i} = \begin{cases} \boldsymbol{X_i} + a \cdot \boldsymbol{R} \otimes \boldsymbol{s}, \boldsymbol{s} = C_1 \cdot \mu_{\boldsymbol{L}} \otimes (\boldsymbol{E_i} - \boldsymbol{X_i}), & \frac{pd}{3} < t \leq \frac{T}{2}, \\ \boldsymbol{E_i} + a \cdot C \otimes \boldsymbol{s}, \boldsymbol{s} = \mu_{\boldsymbol{B}} \otimes (\mu_{\boldsymbol{B}} \otimes \boldsymbol{E_i} - \boldsymbol{X_i}) & \frac{T}{2} < t < \frac{2pd}{3}, \end{cases} \tag{14}$$

where $C_1 = 2 \cdot b \cdot (1 - \frac{t}{T})$ represents the step size parameter that is randomly updated according to the number of iterations, b is a random value, a is a constant of 0.5, and pd is a population diversity.

In phase 3 of the traditional MPA algorithm, the position is updated only through the information difference between the current optimal solution and the current individual, while ignoring the position information of the global population, which will affect the possibility of the algorithm to obtain better individuals to a certain extent. In this paper, we want to set the new position of an individual as a function of its current position, the position of the optimal solution and the position of another random individual. Therefore, a new way of updating individuals is proposed, as shown in Eq. (15):

$$\boldsymbol{X_i} = r_1 \cdot (\boldsymbol{E_i} - r_2 \cdot \boldsymbol{X_i}) + C \cdot \mu_{\boldsymbol{L}} \otimes (\boldsymbol{X_j} - \boldsymbol{X_i}), \tag{15}$$

where r_1 and r_2 are two numbers that are uniformly randomly generated in the interval [0, 1]. $\boldsymbol{X_j}$ is the index position of an individual in a randomly selected population. The introduction of random individual positions helps to enhance the population diversity and randomness, which can adapt to complex and changing optimization problems, to prevent premature convergence to local optima and increase the likelihood of discovering the global optimum.

D. MODIFIED FADS

When considering the influence of eddy formation and fish aggregation device effect on the behavior and activity patterns of marine predators, the use of randomly generated individuals to replace the original individuals in the traditional MPA can enhance the diversity within the search space and partially alleviate the issue of local optimization, but the randomly generated individuals may lack clear guidance on the search direction and will not systematically point to the better solution area. Therefore, we modify the current individual's position using the disparity between the positions of two randomly chosen individuals within the current population, as shown in Eq. (16):

$$\begin{aligned} \boldsymbol{S} &= \boldsymbol{R} \otimes (\boldsymbol{X_{max}} - \boldsymbol{X_{min}}) \cdot e^{-C_2 \cdot \frac{t}{T}}, \\ \boldsymbol{X_i} &= \boldsymbol{X_i} + b \cdot (\boldsymbol{X_{r_1}} - \boldsymbol{X_{r_2}}) + \boldsymbol{S} \otimes \boldsymbol{U}, \end{aligned} \tag{16}$$

where $b \in [0, 1]$ is a random number. $\boldsymbol{X_{r_1}}$ and $\boldsymbol{X_{r_2}}$ are the positions of two random individuals in the current population. $C_2 = 2 \cdot (0.1 - 0.05 \cdot (t/T))$ is the weight factor that varies with the number of iterations. $\boldsymbol{U}$ is a randomly generated vector in [0, 1], and this update mechanism allows individuals to learn from information from other individuals, and even randomly selected individuals may provide valuable search directions. Such interactions reflect the core idea of swarm intelligence, which is to improve the performance of the entire group through the sharing of information between individuals.

E. ALGORITHM WORKFLOW

According to the proposed RMPA-AE strategy. We have reconstructed the MPA algorithm and adjusted the update strategy. The basic flow of the algorithm is shown in Algorithm 2. We give the maximum iteration limit of the algorithm and the size of the population, and finally obtain the best estimated position of each position through the proposed algorithm in the process of iteration.

TABLE 1. Algorithms parameter configurations.

Algorithms	Parameter	Value
CS	Population size N_p, Nest replacement probability P_a, Step control parameter α, Max objective function calls FE	N_p = 50, P_a = 0.25, α= 0.01, FE=1000
CLPSO	Inertia weight bounds, Learning factor c Total learning memory pool size m, Max objective function calls FE	0.4-0.9, c = 1.49445, m = 7, FE=1000
MPSO	The inertia weight parameter w, The individual learning factor c_1, The social learning factor c_2, Max objective function calls FE	w=0.5, c_1=1.5, c_2=1.5, FE=1000
FPA	Pollination probability p, Step control parameter γ, Max objective function calls FE	p=0.5, γ =0.01, FE=1000
MPA, RMPA-AE	Eddy and FAD influence parameters FADs, Max objective function calls FE	FADs=0.2, FE=1000
RACS, BBPSO	Max objective function calls FE.	FE=1000.

VI. NUMERICAL RESULTS AND DISCUSSIONS

A. DATASET INFORMATION

The experiment was simulated in a two-dimensional space of 100 × 100 square meters. The dataset of node locations

TABLE 2. Multimodal and Unimodal functions test results with a dimension of 30.

Function		Algorithm								Function		Algorithm							
		BBPSO	CLPSO	CS	FPA	MPA	MPSO	RACS	RMPA-AE			BBPSO	CLPSO	CS	FPA	MPA	MPSO	RACS	RMPA-AE
f_1	Min	2.77E-19	2.21E+03	4.41E-03	6.34E+00	2.15E-52	8.70E-73	1.94E-15	**0.00E+00**	f_{14}	Min	2.69E+01	1.05E+02	5.86E+01	5.77E+01	**0.00E+00**	**0.00E+00**	1.11E-09	**0.00E+00**
	Mean	7.03E-16	3.62E+03	8.71E-03	1.95E+01	1.04E-49	1.76E-61	7.86E-15	**0.00E+00**		Mean	7.87E+01	1.34E+02	8.27E+01	8.87E+01	**0.00E+00**	1.04E+01	1.17E-08	**0.00E+00**
	Std	1.61E-15	1.06E+03	2.98E-03	8.22E+00	2.09E-49	9.63E-61	5.29E-15	**0.00E+00**		Std	2.72E+01	1.47E+01	1.15E+01	1.25E+01	0.00E+00	1.28E+01	1.22E-08	**0.00E+00**
f_2	min	6.89E-14	2.12E+01	2.94E-01	3.21E+00	1.01E-29	1.02E-41	1.30E-09	**8.26E-233**	f_{15}	Min	3.18E-10	1.36E+01	2.96E+00	1.41E+00	**0.00E+00**	**0.00E+00**	2.18E-07	**0.00E+00**
	Mean	1.00E+01	3.37E+01	4.99E-01	6.34E+00	9.07E-28	3.52E-36	2.83E-09	**2.85E-213**		Mean	7.23E+00	1.55E+01	9.77E+00	3.59E+00	3.20E-15	4.81E-01	3.73E-01	**0.00E+00**
	Std	1.14E+01	6.61E+00	1.45E-01	2.04E+00	2.00E-27	5.79E-36	8.96E-10	**0.00E+00**		Std	9.49E+00	9.49E-01	3.23E+00	9.19E-01	1.08E-15	2.63E+00	5.12E-01	**0.00E+00**
f_3	Min	2.96E+02	1.48E+04	2.74E+02	6.28E+00	2.48E-21	6.00E+01	1.42E+02	**0.00E+00**		Min	**0.00E+00**	1.45E+00	2.37E-03	4.45E-01	**0.00E+00**	**0.00E+00**	3.33E-16	**0.00E+00**
	Mean	9.49E+03	2.79E+04	4.49E+02	1.99E+01	1.00E-12	3.60E+02	1.01E+03	**0.00E+00**	f_{16}	Mean	1.14E-02	1.87E+00	1.61E-02	6.13E-01	**0.00E+00**	**0.00E+00**	2.16E-14	**0.00E+00**
	Std	5.84E+03	6.28E+03	1.06E+02	1.70E+01	4.91E-12	2.08E+02	4.66E+02	**0.00E+00**		Std	1.33E-02	2.81E-01	9.65E-03	1.12E-01	**0.00E+00**	**0.00E+00**	3.60E-14	**0.00E+00**
f_4	Min	6.55E+00	3.03E+01	1.81E+00	4.76E+00	3.37E-20	1.98E-29	6.89E-02	**1.80E-236**	f_{17}	Min	5.30E-12	3.21E+01	1.80E+01	9.72E+00	1.56E-11	4.24E-03	**1.35E-14**	3.72E-07
	Mean	1.61E+01	3.93E+01	3.10E+00	7.32E+00	2.14E-19	4.45E-25	1.16E-01	**4.31E-201**		Mean	1.09E+02	7.29E+01	3.80E+01	2.18E+01	6.18E-11	3.15E-02	**6.56E-14**	1.52E-06
	Std	7.12E+00	4.52E+00	6.10E-01	1.25E+00	2.05E-19	1.68E-24	2.93E-02	**0.00E+00**		Std	9.02E+01	3.03E+01	1.26E+01	1.06E+01	2.67E-11	5.22E-02	**4.82E-14**	1.15E-06
f_5	Min	**1.08E+01**	6.93E+05	2.87E+01	1.57E+02	2.16E+01	2.53E+01	1.99E+01	2.15E+01	f_{18}	Min	2.40E+02	2.00E+02	1.89E+00	2.78E+01	2.13E-10	8.81E-03	**7.32E-15**	1.96E-05
	Mean	6.58E+03	1.87E+06	3.70E+01	5.94E+02	2.23E+01	5.39E+01	2.38E+01	**2.22E+01**		Mean	7.97E+02	4.56E+03	8.85E+00	7.37E+01	9.23E-10	5.29E-02	**2.99E-14**	1.84E-03
	Std	2.27E+04	1.01E+06	7.32E+00	4.09E+02	5.01E-01	1.44E+02	2.88E+00	**3.71E-01**		Std	3.43E+02	1.48E+04	4.48E+00	4.54E+01	4.78E-10	4.32E-02	**2.56E-14**	3.88E-03
f_6	Min	**0.00E+00**	1.28E+02	**0.00E+00**	9.00E+00	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	f_{19}	Min	1.30E+00	1.85E+00	1.11E+00	1.31E+00	**9.00E-01**	1.01E+00	1.02E+00	**9.00E-01**
	Mean	3.33E+00	2.06E+02	**0.00E+00**	1.86E+01	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**		Mean	2.54E+00	2.35E+00	1.16E+00	1.44E+00	9.97E-01	1.48E+00	1.03E+00	**9.00E-01**
	Std	1.81E+01	3.82E+01	**0.00E+00**	5.98E+00	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**		Std	5.57E-01	2.57E-01	2.20E-02	7.53E-02	1.83E-02	4.75E-01	8.09E-03	**4.52E-16**
f_7	Min	1.22E-02	1.03E+00	1.73E-02	2.22E-02	1.46E-04	1.10E-04	6.36E-03	**5.24E-06**	f_{20}	Min	-4.19E+01	7.33E+01	-1.42E+01	-6.59E+00	-5.16E+01	-4.30E+01	**-5.03E+01**	-4.77E+01
	Mean	5.62E-01	5.00E+00	3.44E-02	5.13E-02	4.63E-04	7.62E-04	1.51E-02	**4.38E-05**		Mean	-2.83E+01	1.13E+02	-1.67E+00	1.15E+01	-4.69E+01	-3.87E+01	-4.81E+01	**-4.19E+01**
	Std	1.64E+00	2.96E+00	1.02E-02	1.86E-02	2.33E-04	3.72E-04	4.40E-03	**3.19E-05**		Std	1.52E+01	2.24E+01	7.55E+00	8.04E+00	1.77E+00	2.28E+00	**1.30E+00**	2.62E+00
f_8	Min	2.83E+01	1.25E+02	2.35E+01	8.94E-01	3.68E-11	9.60E-01	1.98E+01	**0.00E+00**	f_{21}	Min	1.78E-11	5.10E-11	1.08E-11	3.56E-12	**3.51E-12**	1.19E-11	3.55E-12	**3.51E-12**
	Mean	1.14E+02	2.92E+02	4.40E+01	4.22E+00	4.81E-09	7.84E+00	3.95E+01	**0.00E+00**		Mean	2.27E-11	2.90E-10	1.81E-11	6.43E-12	6.48E-12	3.31E-10	**3.66E-12**	4.14E-12
	Std	5.52E+01	6.35E+01	9.14E+00	2.48E+00	6.32E-09	1.10E+01	1.06E+01	**0.00E+00**		Std	2.08E-12	3.25E-10	2.91E-12	2.09E-12	2.95E-12	1.65E-09	**1.51E-13**	1.49E-12
f_9	Min	3.53E-19	3.56E+02	4.16E-04	1.01E+00	1.34E-53	1.62E-72	1.07E-16	**0.00E+00**	f_{22}	Min	3.29E-32	1.24E-12	2.79E-14	2.68E-13	1.85E-21	2.31E-13	7.58E-17	**-1.00E+00**
	Mean	5.33E+01	5.75E+02	1.14E-03	2.12E+00	1.37E-50	3.12E-65	7.96E-16	**0.00E+00**		Mean	5.48E-14	2.28E-12	4.71E-14	3.57E-13	6.46E-21	1.45E-12	4.01E-16	**-1.00E+00**
	Std	8.60E+01	2.13E+02	4.59E-04	7.88E-01	3.97E-50	1.27E-64	3.54E-16	**0.00E+00**		Std	1.01E-13	8.13E-13	1.76E-14	6.65E-14	2.67E-21	1.45E-12	1.84E-16	**0.00E+00**
f_{10}	Min	**-5.00E+00**	-2.10E+00	**-5.00E+00**	-4.79E+00	**-5.00E+00**	**-5.00E+00**	**-5.00E+00**	**-5.00E+00**	f_{23}	Min	-1.15E+03	-1.02E+03	-1.10E+03	-1.03E+03	-1.13E+03	-1.12E+03	**-1.17E+03**	-1.12E+03
	Mean	**-5.00E+00**	-7.00E-01	-4.99E+00	-4.55E+00	**-5.00E+00**	**-5.00E+00**	**-5.00E+00**	**-5.00E+00**		Mean	-1.07E+03	-9.74E+02	-1.06E+03	-1.01E+03	**-1.10E+03**	-1.05E+03	-1.17E+03	-1.04E+03
	Std	1.56E-10	6.91E-01	1.24E-03	1.26E-01	1.23E-12	**0.00E+00**	5.96E-10	2.86E-16		Std	3.55E+01	2.57E+01	1.48E+01	1.14E+01	1.60E+01	3.24E+01	**1.74E-13**	4.11E+01
f_{11}	Min	9.95E-01	0.00E+00	9.95E-01	-9.96E-01	**-1.00E+00**	9.95E-01	9.95E-01	**-1.00E+00**	f_{24}	Min	3.00E-01	8.64E+00	1.10E+00	1.90E+00	9.99E-02	9.99E-02	3.00E-01	**0.00E+00**
	Mean	9.95E-01	0.00E+00	9.95E-01	-1.15E-01	-2.08E-01	9.95E-01	9.95E-01	**-1.00E+00**		Mean	3.62E-01	1.06E+01	1.39E+00	2.47E+00	1.03E-01	9.99E-02	3.47E-01	**2.53E-157**
	Std	3.39E-16	**0.00E+00**	3.39E-16	9.04E-01	9.74E-01	3.39E-16	3.39E-16	**0.00E+00**		Std	7.10E-02	1.06E+00	1.41E-01	3.32E-01	1.83E-02	7.80E-09	5.01E-02	**7.51E-157**
f_{12}	Min	**6.67E-01**	7.35E+03	6.78E-01	2.72E+00	**6.67E-01**	**6.67E-01**	6.67E-01	**6.67E-01**	f_{25}	Min	**5.94E-02**	2.98E-01	3.06E-01	3.56E-01	1.38E-01	1.23E-01	1.27E-01	2.29E-01
	Mean	5.82E+01	1.84E+04	7.66E-01	7.40E+00	**6.67E-01**	**6.67E-01**	**6.67E-01**	**6.67E-01**		Mean	1.29E-01	5.88E-01	4.44E-01	5.24E-01	2.05E-01	1.95E-01	**1.72E-01**	3.47E-01
	Std	1.10E+02	8.17E+03	1.05E-01	4.10E+00	9.73E-10	1.38E-07	5.18E-05	**1.58E-09**		Std	2.92E-02	1.31E-01	6.69E-02	7.00E-02	3.36E-02	5.38E-02	**2.84E-02**	6.52E-02
f_{13}	Min	-3.28E+02	-1.46E+02	-3.01E+02	-2.82E+02	-3.72E+02	**-3.53E+02**	-4.19E+02	-3.24E+02	f_{26}	Min	**3.36E-12**	5.49E+08	1.27E+02	1.21E+04	3.99E-04	7.69E-04	1.04E-01	1.29E-01
	Mean	-3.00E+02	-1.14E+02	-2.88E+02	-2.69E+02	**-3.39E+02**	-3.11E+02	-4.19E+02	-2.91E+02		Mean	**5.25E-10**	1.92E+09	2.49E+02	1.99E+05	5.11E+00	2.66E+03	2.11E-01	1.91E+00
	Std	1.68E+01	**1.18E+01**	6.49E+00	6.83E+00	1.40E+01	3.06E+01	1.58E-10	2.20E+01		Std	**9.15E-10**	9.42E+08	6.36E+01	2.04E+05	1.31E+01	5.81E+03	7.23E-02	2.33E+00

TABLE 3. Multimodal and Unimodal functions test results with a dimension of 50.

Function		Algorithm								Function		Algorithm							
		BBPSO	CLPSO	CS	FPA	MPA	MPSO	RACS	RMPA-AE			BBPSO	CLPSO	CS	FPA	MPA	MPSO	RACS	RMPA-AE
f_1	Min	1.66E-05	3.24E+03	1.10E+00	2.04E+02	8.47E-48	4.29E-64	2.77E-08	**0.00E+00**	f_{14}	Min	1.77E+02	2.62E+02	1.40E+02	1.33E+02	**0.00E+00**	**0.00E+00**	1.91E-01	**0.00E+00**
	Mean	6.67E+02	6.72E+03	2.22E+00	3.29E+02	1.07E-45	3.40E-34	6.64E-08	**0.00E+00**		Mean	2.42E+02	3.54E+02	1.79E+02	1.81E+02	**0.00E+00**	8.81E+00	1.10E+00	**0.00E+00**
	Std	2.54E+03	2.10E+03	6.55E-01	9.52E+01	2.22E-45	1.86E-33	2.98E-08	**0.00E+00**		Std	3.82E+01	3.63E+01	1.98E+01	2.26E+01	**0.00E+00**	2.69E+01	7.56E-01	**0.00E+00**
f_2	Min	1.00E+01	6.68E+01	2.91E+00	1.13E+01	5.55E-28	3.63E-37	3.50E-05	**1.15E-242**	f_{15}	Min	4.43E-03	1.51E+01	9.65E+00	2.90E+00	3.55E-15	3.55E-15	1.80E-03	**0.00E+00**
	Mean	4.60E+01	9.11E+01	5.06E+00	1.51E+01	3.35E-26	4.69E-32	4.68E-05	**1.09E-203**		Mean	1.71E+01	1.71E+01	1.44E+01	4.44E+00	3.55E-15	3.55E-15	1.04E+01	**0.00E+00**
	Std	1.99E+01	1.19E+01	1.46E+00	2.21E+00	3.33E-26	1.13E-31	7.54E-06	**0.00E+00**		Std	6.76E+00	8.03E-01	2.88E+00	1.13E+00	0.00E+00	0.00E+00	5.66E+00	**0.00E+00**
f_3	Min	2.76E+04	4.55E+04	3.34E+03	2.09E+02	1.00E-14	1.04E+03	6.09E+03	**0.00E+00**	f_{16}	Min	2.15E-07	1.80E+00	6.43E-02	1.03E+00	**0.00E+00**	**0.00E+00**	2.72E-09	**0.00E+00**
	Mean	4.64E+04	7.45E+04	6.11E+03	3.53E+02	2.03E-09	3.01E+03	1.16E+04	**0.00E+00**		Mean	2.02E-01	2.67E+00	1.43E-01	1.09E+00	**0.00E+00**	**0.00E+00**	1.94E-08	**0.00E+00**
	Std	9.84E+03	1.41E+04	1.23E+03	1.15E+02	4.11E-09	1.64E+03	2.30E+03	**0.00E+00**		Std	7.56E-01	3.66E-01	4.88E-02	3.04E-02	**0.00E+00**	**0.00E+00**	3.42E-08	**0.00E+00**
f_4	Min	8.42E+01	3.32E+01	6.67E+00	9.57E+00	1.38E-18	2.87E-27	1.67E+00	**1.03E-237**	f_{17}	Min	1.57E+02	1.06E+02	2.79E+01	1.26E+01	**1.34E-09**	1.27E-01	2.47E-07	4.70E-05
	Mean	9.12E+01	4.85E+01	9.45E+00	1.21E+01	1.61E-17	2.68E-22	2.38E+00	**6.54E-196**		Mean	4.89E+02	3.74E+03	4.78E+01	3.06E+01	2.54E-09	2.78E+00	**9.07E-07**	4.10E-04
	Std	2.74E+00	5.97E+00	1.71E+00	1.42E+00	1.18E-17	8.05E-22	6.12E-01	**0.00E+00**		Std	1.61E+02	1.63E+04	1.20E+01	8.63E+00	**7.53E-10**	5.16E+00	4.37E-07	1.48E-03
f_5	Min	4.60E+01	3.85E+06	2.06E+02	4.18E+03	4.24E+01	4.69E+01	**4.13E+01**	4.27E+01	f_{18}	Min	1.94E+03	9.39E+02	3.09E+01	6.11E+01	**1.59E-08**	1.67E-01	1.56E-07	1.39E-03
	Mean	1.87E+04	6.73E+06	3.55E+02	1.27E+04	4.36E+01	1.72E+03	5.97E+01	**4.33E+01**		Mean	3.11E+03	1.62E+05	1.02E+02	1.37E+02	3.93E-03	9.08E+00	**3.31E-07**	1.26E-02
	Std	3.63E+04	2.13E+06	1.57E+02	6.15E+03	6.90E-01	4.69E+03	2.07E+01	**4.30E-01**		Std	7.66E+02	2.08E+05	5.09E+01	4.64E+01	5.27E-03	4.23E+01	**1.29E-07**	1.97E-02
f_6	Min	**0.00E+00**	2.94E+02	1.00E+00	6.10E+01	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	f_{19}	Min	2.78E+00	4.20E+00	1.76E+00	2.63E+00	**9.00E-01**	1.06E+00	1.17E+00	**9.00E-01**
	Mean	4.39E+01	3.77E+02	3.73E+00	9.36E+01	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**		Mean	5.03E+00	6.05E+00	1.96E+00	3.13E+00	1.00E+00	2.13E+00	1.34E+00	**9.00E-01**
	Std	5.67E+01	5.85E+01	1.78E+00	1.86E+01	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**		Std	8.50E-01	7.35E-01	1.03E-01	2.11E-01	2.06E-02	8.91E-01	6.91E-02	**4.52E-16**
f_7	Min	4.82E-02	1.81E+01	6.05E-02	9.12E-02	1.44E-04	1.18E-04	1.85E-02	**1.21E-06**	f_{20}	Min	-5.85E+01	1.63E+02	1.60E+01	1.84E+01	**-8.00E+01**	-6.43E+01	-7.23E+01	-7.16E+01
	Mean	3.97E+00	5.51E+01	9.36E-02	1.62E-01	6.28E-04	7.94E-04	3.22E-02	**2.89E-05**		Mean	-1.99E+01	2.67E+02	3.40E+01	4.99E+01	-7.46E+01	-5.67E+01	**-6.85E+01**	-6.41E+01
	Std	6.37E+00	2.91E+01	2.65E-02	5.94E-02	3.36E-04	4.67E-04	7.49E-03	**2.77E-05**		Std	2.38E+01	4.06E+01	1.06E+01	9.87E+00	3.21E+00	5.15E+00	**1.86E+00**	3.44E+00
f_8	Min	5.00E+02	4.53E+02	2.25E+02	3.16E+01	3.10E-06	2.77E+01	1.79E+02	**0.00E+00**	f_{21}	Min	8.15E-20	8.19E-17	3.49E-20	1.28E-20	**1.21E-20**	1.53E-19	3.08E-20	**1.21E-20**
	Mean	6.63E+02	1.81E+06	3.04E+02	5.59E+01	4.98E-05	8.33E+01	2.82E+02	**0.00E+00**		Mean	1.15E-19	2.66E-15	7.58E-20	2.38E-20	1.76E-20	1.64E-19	3.81E-20	**1.25E-20**
	Std	9.98E+01	9.93E+06	4.18E+01	1.86E+01	3.94E-05	4.65E+01	4.38E+01	**0.00E+00**		Std	1.26E-20	3.11E-15	2.35E-20	9.80E-21	6.36E-21	4.39E-21	2.63E-21	**9.33E-22**
f_9	Min	3.92E-06	3.00E+03	2.69E-01	2.54E+01	1.44E-48	1.53E-65	5.29E-09	**0.00E+00**	f_{22}	Min	3.59E-27	5.14E-20	1.50E-21	4.41E-21	6.59E-28	4.94E-21	7.64E-22	**-1.00E+00**
	Mean	5.13E+02	4.46E+03	4.58E-01	7.10E+01	1.13E-46	1.18E-59	1.44E-08	**0.00E+00**		Mean	1.52E-20	2.76E-19	1.92E-21	5.25E-21	1.70E-27	5.10E-20	9.56E-22	**-1.00E+00**
	Std	6.57E+02	8.06E+02	1.29E-01	2.51E+01	1.82E-46	4.37E-59	7.22E-09	**0.00E+00**		Std	4.03E-20	1.86E-19	1.50E-22	5.01E-22	5.41E-28	8.91E-20	9.23E-23	**0.00E+00**
f_{10}	Min	**-5.00E+00**	2.57E+00	-4.94E+00	-3.76E+00	**-5.00E+00**	**-5.00E+00**	**-5.00E+00**	**-5.00E+00**	f_{23}	Min	-1.86E+03	-1.55E+03	-1.75E+03	-1.61E+03	-1.83E+03	-1.82E+03	**-1.96E+03**	-1.79E+03
	Mean	-4.17E+00	4.12E+00	-4.91E+00	-3.41E+00	**-5.00E+00**	**-5.00E+00**	**-5.00E+00**	**-5.00E+00**		Mean	-1.74E+03	-1.39E+03	-1.71E+03	-1.58E+03	-1.78E+03	-1.72E+03	**-1.96E+03**	-1.68E+03
	Std	1.90E+00	9.31E-01	1.52E-02	2.58E-01	1.26E-11	2.86E-16	2.91E-06	**4.95E-16**		Std	5.44E+01	5.48E+01	1.91E+01	1.23E+01	2.69E+01	5.25E+01	**2.03E-07**	5.61E+01
f_{11}	Min	9.92E-01	0.00E+00	9.92E-01	**-1.00E+00**	**-1.00E+00**	9.92E-01	9.92E-01	**-1.00E+00**	f_{24}	Min	9.00E-01	1.22E+01	2.70E+00	3.70E+00	9.99E-02	9.99E-02	6.00E-01	**0.00E+00**
	Mean	9.92E-01	0.00E+00	9.92E-01	-2.66E-01	1.39E-01	9.92E-01	9.92E-01	**-1.00E+00**		Mean	1.85E+00	1.45E+01	3.12E+00	4.64E+00	1.50E-01	9.99E-02	7.37E-01	**5.94E-146**
	Std	3.39E-16	0.00E+00	3.39E-16	8.21E-01	9.93E-01	3.39E-16	3.39E-16	**0.00E+00**		Std	2.31E+00	1.15E+00	3.56E-01	5.59E-01	5.09E-02	1.96E-08	6.27E-02	**2.81E-145**
f_{12}	Min	8.41E-01	1.17E+05	2.46E+00	6.00E+01	**6.67E-01**	**6.67E-01**	**6.67E-01**	**6.67E-01**	f_{25}	Min	**2.25E-01**	5.94E-01	5.10E-01	5.23E-01	3.20E-01	3.50E-01	2.29E-01	3.69E-01
	Mean	1.03E+04	2.31E+05	6.94E+00	1.82E+02	**6.67E-01**	**6.67E-01**	6.79E-01	**6.67E-01**		Mean	**3.33E-01**	7.87E-01	6.33E-01	6.60E-01	4.51E-01	4.48E-01	3.69E-01	5.34E-01
	Std	3.18E+04	7.75E+04	2.94E+00	9.67E+01	3.62E-09	1.92E-06	2.38E-02	**3.40E-09**		Std	5.61E-02	1.03E-01	5.87E-02	5.77E-02	6.33E-02	8.45E-02	**4.28E-02**	8.97E-02
f_{13}	Min	-3.31E+02	-1.07E+02	-2.78E+02	-2.52E+02	**-3.54E+02**	-3.07E+02	-4.19E+02	-2.88E+02	f_{26}	Min	**2.78E+01**	9.16E+08	9.57E+03	2.28E+06	7.82E-02	3.83E+02	5.40E+01	5.48E+01
	Mean	-2.91E+02	-8.92E+01	-2.62E+02	-2.38E+02	-3.22E+02	-2.64E+02	**-4.18E+02**	-2.56E+02		Mean	1.01E+03	2.23E+09	1.70E+04	8.90E+06	**4.26E+01**	1.60E+06	9.13E+01	2.20E+02
	Std	1.73E+01	1.01E+01	5.67E+00	5.61E+00	1.58E+01	2.08E+01	**1.02E+00**	1.51E+01		Std	1.13E+03	8.53E+08	6.37E+03	7.67E+06	8.49E+01	5.18E+06	**2.04E+01**	1.49E+02

is randomly generated. When specific scenarios require (e.g., verifying various heterosexual networks), node datasets are randomly generated through O-type and S-type frameworks.

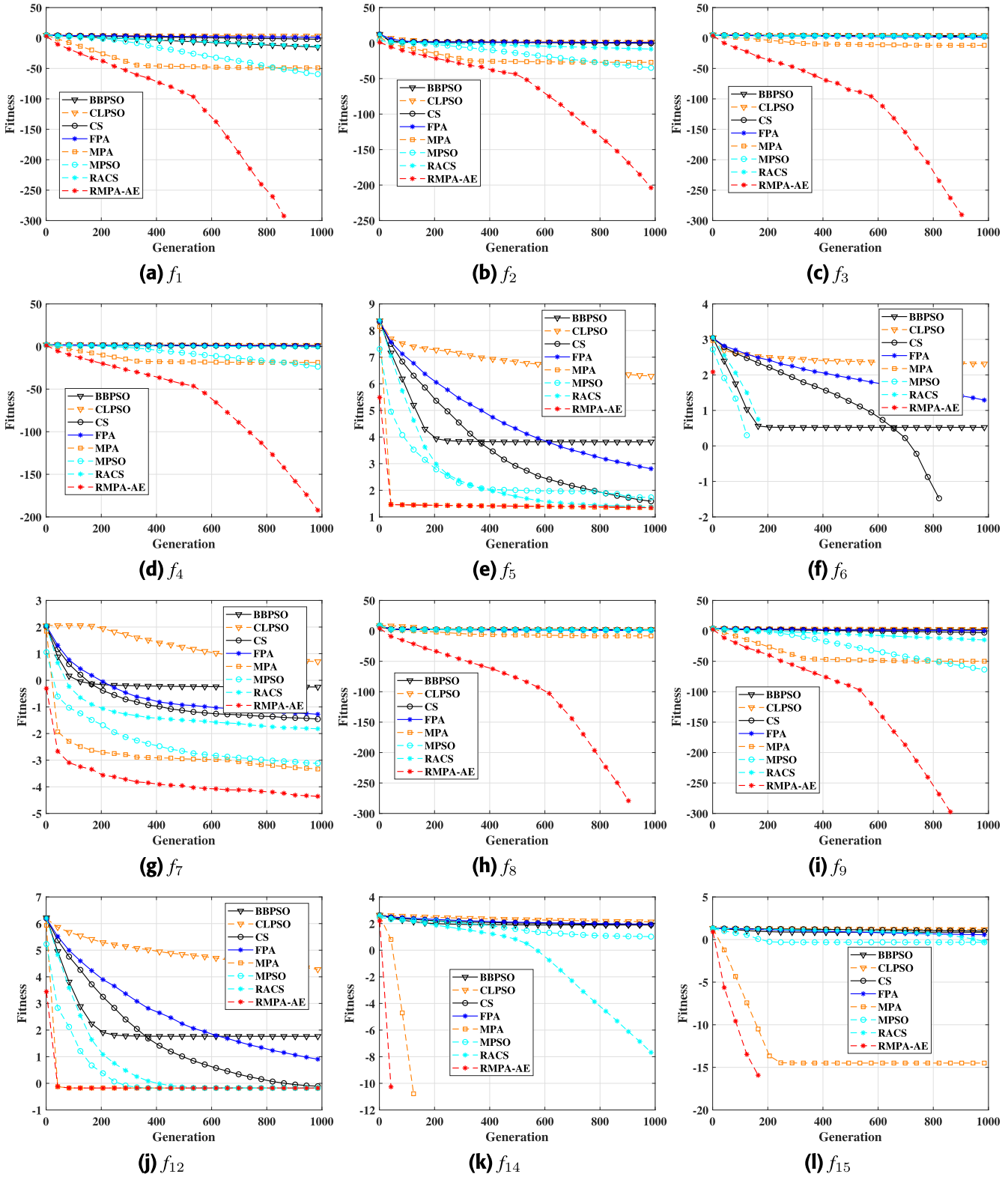

FIGURE 3. Compare the convergence of BBPSO, CLPSO, CS, FPA, MPA, MPSO, RACS and RMPA-AE algorithms, ($D = 30$).

B. FUNCTIONAL TESTING

In this section, we employ 26 widely recognized benchmark functions to evaluate the algorithm's performance. Functions f_1 - f_{12} are unimodal functions, and functions f_{13} - f_{26} are multimodal functions. Unimodal and multimodal functions can respectively evaluate the algorithm's abilities in terms of exploitation and exploration. We assess the performance of the algorithm by emphasizing two key aspects: search

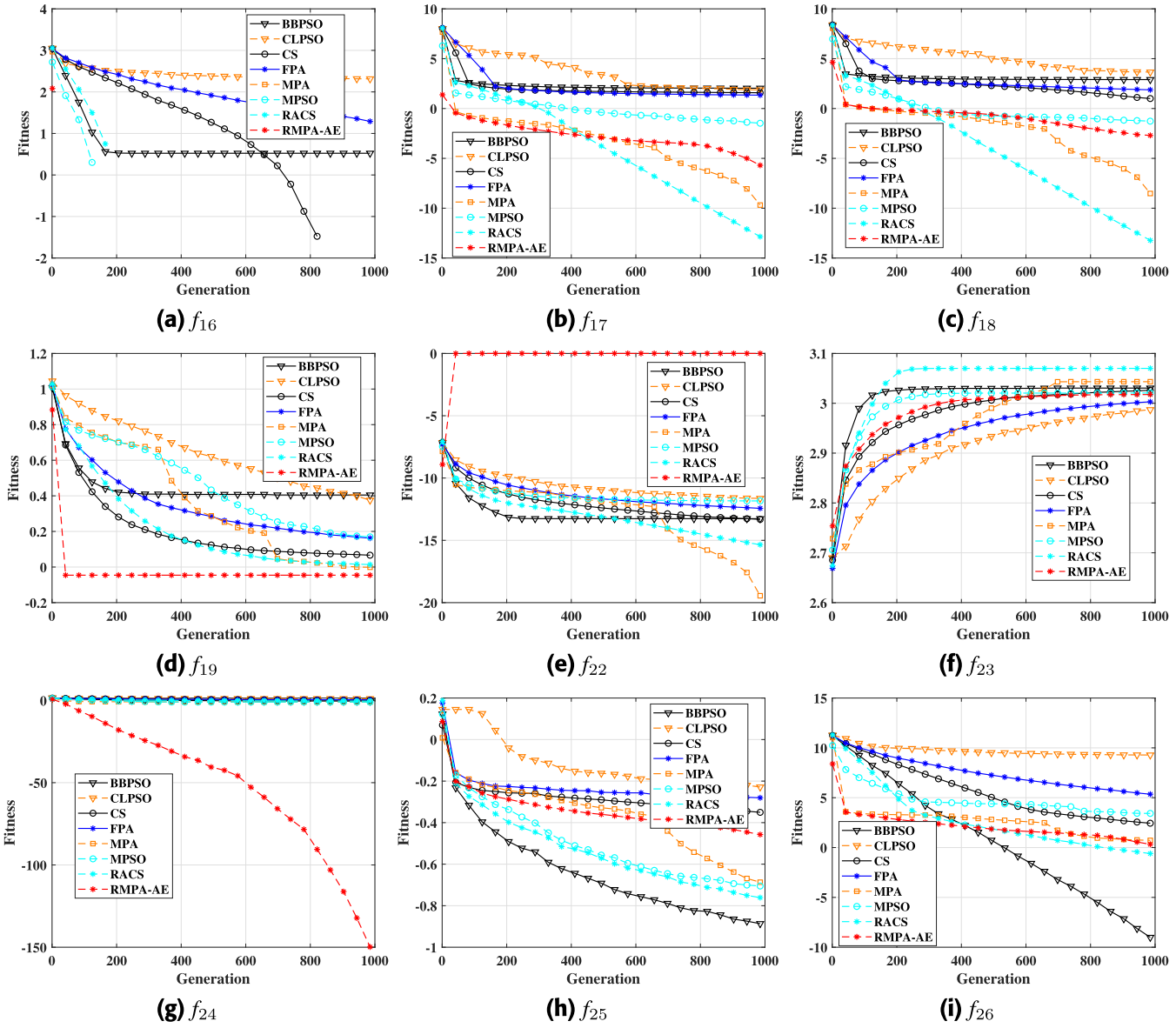

FIGURE 4. Compare the convergence of BBPSO, CLPSO, CS, FPA, MPA, MPSO, RACS and RMPA-AE algorithms, ($D = 30$).

accuracy and convergence. When testing with benchmark functions, we limit the maximum number of iterations to 1000 and run each benchmark function independently 30 times. We compare the algorithm with other algorithms that have shown good performance in recent years, such as FPA, RACS, IMPA, MPSO, MPA, CS, CLPSO, BBPSO, under the dimensions of 30 and 50. And Table 1 shows the parameter configuration of the comparison algorithm.

1) DIMENSION IS 30

In the experiment with a dimension of 30, the algorithm is iterated 1,000 times. Experiments calculate the values of the minimum, mean, and standard deviation of the algorithm on each function. We present the test results for each algorithm in 26 functions in a table, as shown in Table 2. Table 2 shows that the RMPA-AE algorithm takes 00E+00 on the functions f_1, f_3,f_6, f_8, f_9, f_{14} - f_{16}. The proposed algorithm (RMPA-AE) obtains the optimal values for the minimum, mean and standard deviation of most unimodal functions (f_1 - f_{12}). For functions that do not obtain 00E+00 values, such as f_2 and f_4, the optimal values are still obtained among the compared algorithms. This means that the RMPA-AE algorithm is highly competitive in terms of development capabilities. For the multimodal function, the RMPA-AE algorithm obtains the optimal values of six functions, which are f_{14},f_{15},f_{16},f_{19},f_{22},f_{24}, respectively. Among the contrasted algorithms, the number of functions achieving the optimal is the largest, which reflects the effectiveness of the algorithm's exploration ability and can effectively avoid falling into the local optimum.

Algorithm 2 RMPA-AE Algorithm

Input:
Maximum iteration limit T, population size n;
Output:
Output of the optimal position;
while $t \leq T$ **do**
  Parameter initialization;
  Evaluate the fitness of the initial population;
  Calculate the population diversity pd by Eq. (13);
  if Current population diversity $p_i < p/3$ **then**
    Phase 1: Update population by Eq. (6).
  else if $p/3 < p_i < 2 \times p/3$ **then**
    $t < T/2$
    Early phase 2: Update population by the previous part of Eq. (14).
    $t > T/2$
    Late phase 2: Update population by the later part of Eq. (14).
  else if $p_i > 2 \times p/3$ **then**
    Phase 3: Update population by Eq. (15).
  end if
  Populations are updated according to FADs;
end while
Output optimal result.

Fig. 3 and Fig. 4 show the convergence curves of the RMPA-AE algorithm and its comparative algorithms at a dimension of 30, after 1000 iterations. The algorithm achieves the fastest convergence speed on functions f_1 - f_9, f_{14} - f_{16}, f_{19}, and f_{24}. This indicates that the algorithm possesses an efficient search strategy and excellent optimization performance, making it particularly useful in dealing with complex problems, especially in time-sensitive or resource-limited application scenarios. Additionally, the rapid convergence also reduces the demand for computational resources.

2) DIMENSION IS 50

When the dimension is 50, we run 1000 iterations of the algorithm. Experiments also calculated the minimum, mean, and standard deviation values of the algorithm for each function. The test results are given in Table 3. From the table, it can be seen that the RMPA-AE algorithm achieves the optimal value of 00E+00 on functions f_1, f_3, f_6, f_8,f_9, f_{14} - f_{16}. The test results for unimodal functions show that the RMPA-AE algorithm achieves the optimal value on all unimodal functions, except for function f_5 where the RACS algorithm performs better. In the case of multimodal function tests, the RMPA-AE algorithm achieves optimal values in 7 out of 14 functions, specifically in functions f_{14} - f_{16}, f_{19}, f_{21}, f_{22}, and f_{24}. Among the comparative algorithms, only the MPA algorithm achieves relatively better results, obtaining optimal values in 5 functions. This emphasizes that the proposed algorithm has made effective improvements over traditional algorithms and also indicates that the algorithm has a strong exploration capability to avoid falling into local optima. Additionally, by comparing dimensions 30 and 50, it is evident that the performance of the algorithm improves with increasing dimensions.

C. SIMULATION EXPERIMENTS

In the experiment, we simulated a general network and an O-type network with a hole, and randomly distributed the nodes in a 100 m × 100 m two-dimensional region. In the same environment, the performance of the RMPA-AE algorithm is evaluated in comparison with eight optimization calculations, including DV-Hop, CS, MPA, FPA, CLPSO, BBPSO, MPSO, and RACS. The experiment considers the error comparison of each node under the same conditions. Additionally, we analyze how variations in the anchor node ratio and communication radius affect the positioning error. In the experiment, the proportion of anchor nodes is 10% - 40%. The radius size is 20 m - 30 m. All simulation experiments were averaged 100 times.

We evaluate the localization error of the proposed algorithm at the communication radius R by normalizing the root mean square error, expressed as follows Eq. (17):

$$\text{NRMSE} = \frac{\sum_{i=1}^{N} \sqrt{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2}}{N \times R}, \tag{17}$$

where (x_i, y_i) and $(\hat{x}_i, \hat{y}_i)$ denote the true and estimated coordinates of the unknown node, respectively. N indicates the number of unknown nodes.

1) ERROR AT EACH NODE

Fig. 5 compares the localization errors for 70% of the unknown nodes using the RMPA-AE algorithm against eight other optimization methods, including DV-Hop, CS, MPA, FPA, CLPSO, BBPSO, MPSO, and RACS, under the conditions of a 30% anchor node ratio and a 25 m communication radius. The figure demonstrates that the RMPA-AE algorithm provides higher positioning accuracy for the majority of unknown nodes, with positioning errors for most nodes reaching 0.1. When comparing with the traditional MPA algorithm, as shown in Fig. 5, the enhanced algorithm results in increased stability and reduced positioning error for the MPA algorithm, as evident from the observations.

2) THE IMPACT OF ANCHOR NODE RATIO

When examining the impact of the anchor node ratio on the localization error, we set the proportion of anchor nodes to 10% - 40%, and increase it at 10% intervals. The fixed radius is 25 m. The algorithm's positioning error in the O-type network with one hole and the S-type network with two holes are tested respectively. Fig. 6 represent the error results of the O-type network and the S-type network, respectively. As depicted in Fig. 6, the positioning accuracy of all algorithms improves with an increasing proportion of anchor nodes. However, the RMPA-AE algorithm achieves

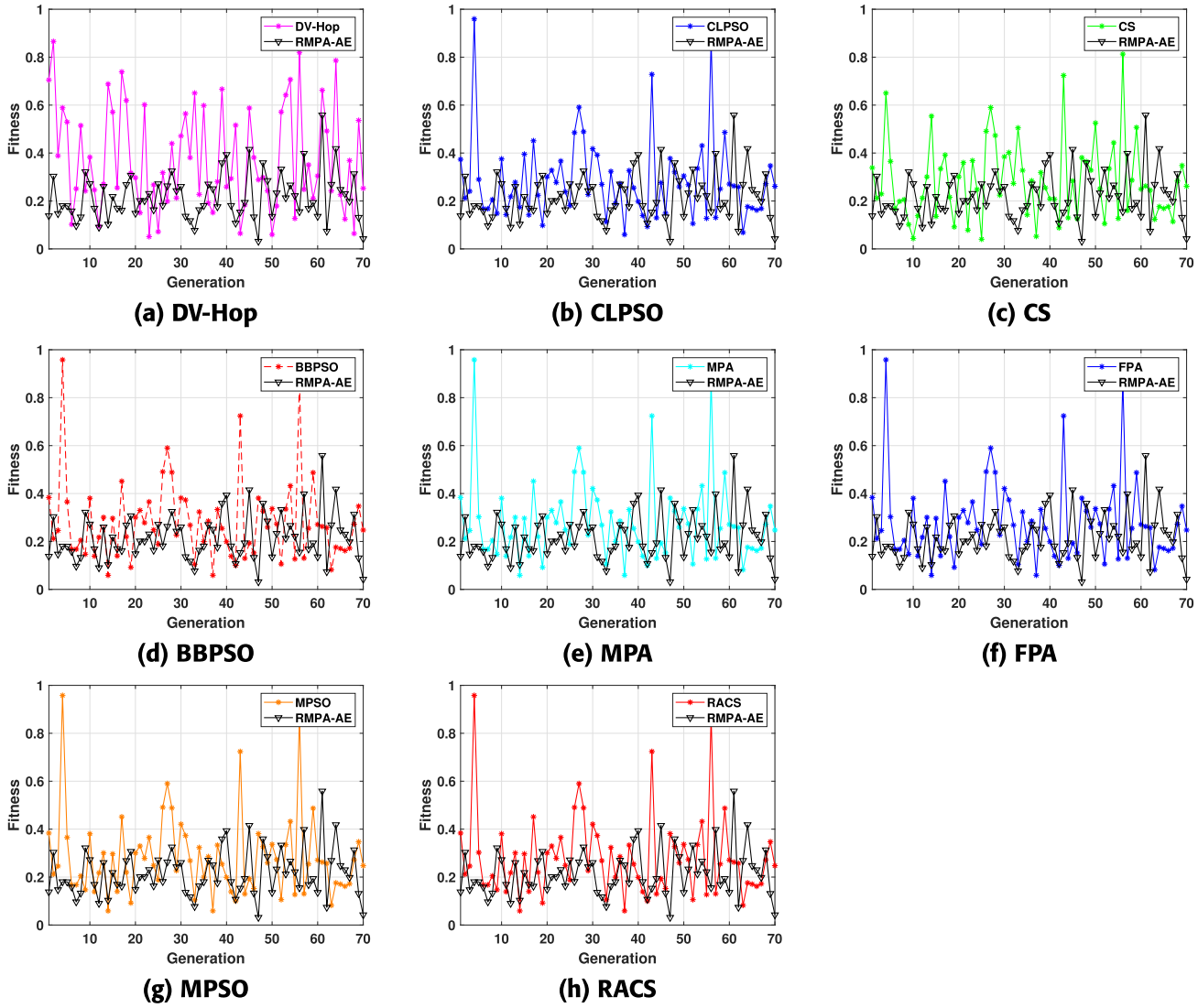

FIGURE 5. Compare the convergence of BBPSO, CLPSO, CS, FPA, MPA, MPSO, RACS and RMPA-AE algorithms, ($D = 30$).

the best positioning accuracy at any scale. Although the algorithms proposed in both types of networks achieve optimal positioning accuracy, the positioning effect of the O-type network is superior to that of the S-type network. Both O-type and S-type networks exhibit irregularities, but the degree of irregularity in a single-hole O-type network is lower than that in a double-hole S-type network. The higher the degree of irregularity, the more uneven the signal coverage or the fewer the node connections. Therefore, the algorithm faces challenges when dealing with different network structures. Under low node density conditions, this algorithm is not the best choice. However, its performance reaches optimal under high node density conditions.

3) THE IMPACT OF COMMUNICATION RADIUS

Fig. 7 shows the effect of radius on positioning error in S-shaped and O-type networks. In the experiment, we set the anchor node ratio to 30%, and the radius size is 20m 35m, which is increased at 5m intervals. In the experiment, we performed 100 experiments on each algorithm to take the mean. As illustrated in the figure, with the increase in radius, the positioning accuracy of all algorithms in the O-type network increases, while the positioning precision of all comparison algorithms in the S-type network fluctuates. However, in general, the RMPA-AE algorithm still achieves the best positioning accuracy. In the S-type network, the positioning error is unstable mainly because the two holes cause poor node connectivity to the network, so one node needs to detour to the other node, and the error will increase when estimating the distance based on the DV-Hop algorithm. Therefore, from the perspective of stability, the RMPA-AE algorithm is more suitable for O-type networks. Whether the ratio of anchor nodes increases or the radius increases, the positioning error of the algorithm in the O-type decreases.

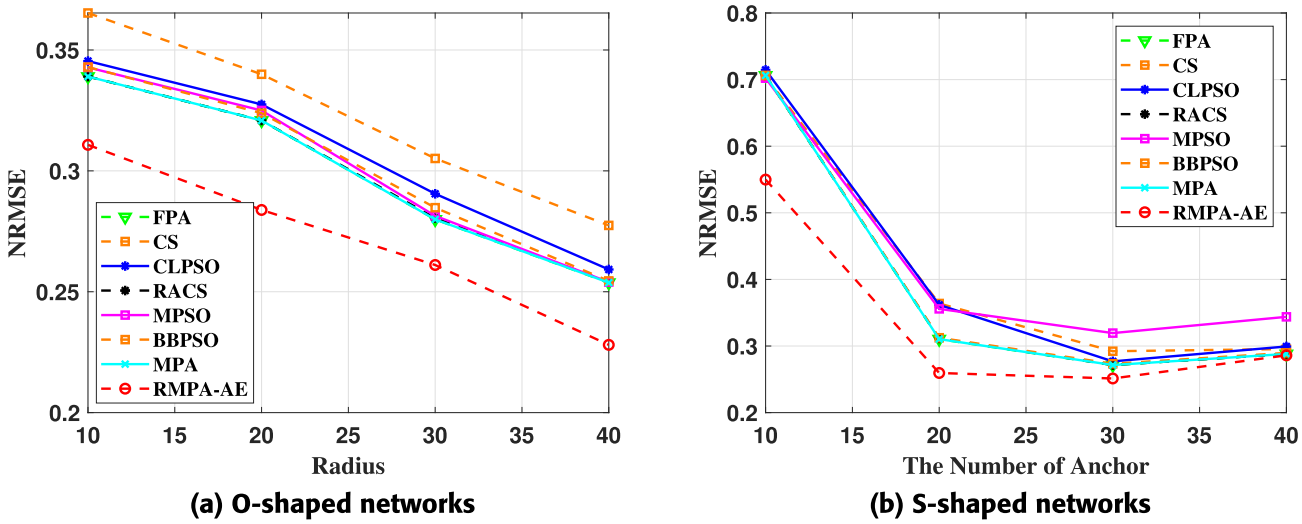

FIGURE 6. Effect of anchor node ratio in O-type and S-type networks, R=25 m.

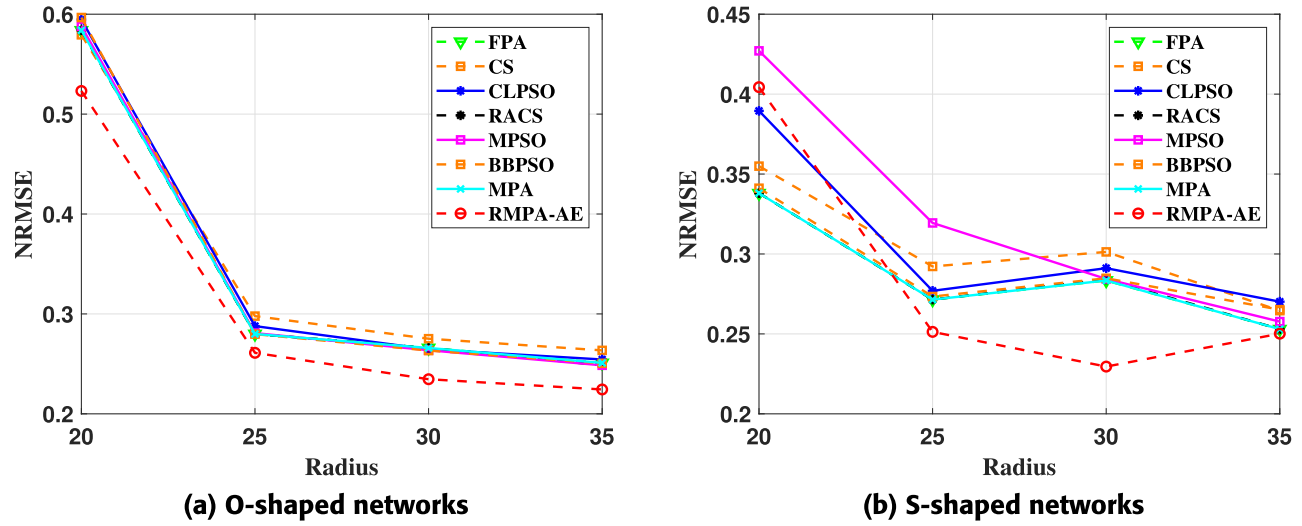

FIGURE 7. Effect of the radius in O-type and S-type networks, anchor node ratio: 30%.

From the perspective of positioning accuracy, when the radius is 30m, the localization error of the S-type network is better than that of the O-type network.

VII. CONCLUSION

This paper proposes a Reconstructed Marine Predators Algorithm with Adaptive Enhancement (RMPA-AE) strategy to improve the accuracy of wireless sensor node localization. To avoid the limitation on the searching ability of Brownian motion and Lévy flights due to the division of marine predation process by iteration count, we reconstruct the phases through population diversity. Additionally, to fully leverage the advantages of long and short steps in Lévy flights during phase 2, we restructure the updating phase into early and late stages based on the iteration cycle. Subsequently, we adjust the population updates separately for phase 2, phase 3, and the FADs phase. In phase 2, which is the exploration phase, the characteristics of Lévy flights are more pronounced. We propose a Lévy flight strategy based on iteration count to fully utilize the advantages of the Lévy strategy. In phase 3 and the FADs phase, the introduction of random individual positions guides the population update, promoting individuals to learn from other individuals' information to enhance the overall performance of the population. Benchmark function tests demonstrate that our algorithm achieves excellent convergence speed and exploration ability across different dimensions. Simulation experiments show that under various anchor node ratios, communication radius, and network distributions, the RMPA-AE algorithm outperforms other comparative algorithms in terms of localization accuracy. When applied in areas such as traffic management, environmental monitoring,

infrastructure maintenance, and emergency response systems, the precise location information significantly enhances service quality and efficiency, and augments the level of intelligence in urban management.

REFERENCES

[1] P. K. Mall, V. Narayan, S. Pramanik, S. Srivastava, M. Faiz, S. Sriramulu, and M. N. Kumar, "Fuzzynet-based modelling smart traffic system in smart cities using deep learning models," in *Handbook of Research on Data-Driven Mathematical Modeling in Smart Cities*. Hershey, PA, USA: IGI Global, 2023, pp. 76–95.

[2] S. Chen, L. Zhang, Y. Tang, C. Shen, R. Kumar, K. Yu, U. Tariq, and A. K. Bashir, "Indoor temperature monitoring using wireless sensor networks: A SMAC application in smart cities," *Sustain. Cities Soc.*, vol. 61, Oct. 2020, Art. no. 102333.

[3] H. Sharma, A. Haque, and F. Blaabjerg, "Machine learning in wireless sensor networks for smart cities: A survey," *Electronics*, vol. 10, no. 9, p. 1012, Apr. 2021.

[4] A. Khalifeh, K. A. Darabkh, A. M. Khasawneh, I. Alqaisieh, M. Salameh, A. Alabdala, S. Alrubaye, A. Alassaf, S. Al-HajAli, R. Al-Wardat, N. Bartolini, G. Bongiovannim, and K. Rajendiran, "Wireless sensor networks for smart cities: Network design, implementation and performance evaluation," *Electronics*, vol. 10, no. 2, p. 218, Jan. 2021.

[5] T. M. Ghazal, M. K. Hasan, H. M. Alzoubi, M. Alshurideh, M. Ahmad, and S. S. Akbar, "Internet of Things connected wireless sensor networks for smart cities," in *The Effect of Information Technology on Business and Marketing Intelligence Systems*. Berlin, Germany: Springer, 2023, pp. 1953–1968.

[6] V. C. Pujol, P. K. Donta, A. Morichetta, I. Murturi, and S. Dustdar, "Edge intelligence—Research opportunities for distributed computing continuum systems," *IEEE Internet Comput.*, vol. 27, no. 4, pp. 53–74, Jul. 2023.

[7] A. Poulose and D. S. Han, "UWB indoor localization using deep learning LSTM networks," *Appl. Sci.*, vol. 10, no. 18, p. 6290, Sep. 2020.

[8] L. Barbieri, M. Brambilla, A. Trabattoni, S. Mervic, and M. Nicoli, "UWB localization in a smart factory: Augmentation methods and experimental assessment," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–18, 2021.

[9] Y. V. Lakshmi, P. Singh, S. Mahajan, A. Nayyar, and M. Abouhawwash, "Accurate range-free localization with hybrid DV-hop algorithms based on PSO for UWB wireless sensor networks," *Arabian J. Sci. Eng.*, vol. 49, no. 3, pp. 4157–4178, Oct. 2023.

[10] W. Wang, X. Liu, M. Li, Z. Wang, and C. Wang, "Optimizing node localization in wireless sensor networks based on received signal strength indicator," *IEEE Access*, vol. 7, pp. 73880–73889, 2019.

[11] R. Kaune, J. Hörst, and W. Koch, "Accuracy analysis for tdoa localization in sensor networks," in *Proc. 14th Int. Conf. Inf. Fusion*, Aug. 2011, pp. 1–8.

[12] M. Yang, D. R. Jackson, J. Chen, Z. Xiong, and J. T. Williams, "A TDOA localization method for nonline-of-sight scenarios," *IEEE Trans. Antennas Propag.*, vol. 67, no. 4, pp. 2666–2676, Apr. 2019.

[13] J. Yanfei, Z. Kexin, and Z. Liquan, "Improved DV-hop location algorithm based on mobile anchor node and modified hop count for wireless sensor network," *J. Electr. Comput. Eng.*, vol. 2020, pp. 1–9, Aug. 2020.

[14] L. Gui, T. Val, A. Wei, and R. Dalce, "Improvement of range-free localization technology by a novel DV-hop protocol in wireless sensor networks," *Ad Hoc Netw.*, vol. 24, pp. 55–73, Jan. 2015.

[15] P. Wang, F. Xue, H. Li, Z. Cui, and J. Chen, "A multi-objective DV-hop localization algorithm based on NSGA-II in Internet of Things," *Mathematics*, vol. 7, no. 2, p. 184, Feb. 2019.

[16] J. Liu, M. Liu, X. Du, P. S. Stanimirovi, and L. Jin, "An improved DV-hop algorithm for wireless sensor networks based on neural dynamics," *Neurocomputing*, vol. 491, pp. 172–185, Jun. 2022.

[17] L. Gui, F. Xiao, Y. Zhou, F. Shu, and T. Val, "Connectivity based DV-hop localization for Internet of Things," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 8949–8958, Aug. 2020.

[18] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, "Marine predators algorithm: A nature-inspired metaheuristic," *Ex. Syst. Appl.*, vol. 152, Aug. 2020, Art. no. 113377.

[19] M. A. Al-Betar, M. A. Awadallah, S. N. Makhadmeh, Z. A. A. Alyasseri, G. Al-Naymat, and S. Mirjalili, "Marine predators algorithm: A review," *Arch. Comput. Methods Eng.*, vol. 30, pp. 3405–3435, Jun. 2023.

[20] B. Sahu, P. K. Das, and R. Kumar, "A modified cuckoo search algorithm implemented with SCA and PSO for multi-robot cooperation and path planning," *Cognit. Syst. Res.*, vol. 79, pp. 24–42, Jun. 2023.

[21] X. Yu and W. Luo, "Reinforcement learning-based multi-strategy cuckoo search algorithm for 3D UAV path planning," *Exp. Syst. Appl.*, vol. 223, Aug. 2023, Art. no. 119910.

[22] Y. Pu, J. Song, M. Wu, X. Xu, and W. Wu, "Node location using cuckoo search algorithm with grouping and drift strategy for WSN," *Phys. Commun.*, vol. 59, Aug. 2023, Art. no. 102088.

[23] X.-S. Yang, M. Karamanoglu, and X. He, "Flower pollination algorithm: A novel approach for multiobjective optimization," *Eng. Optim.*, vol. 46, no. 9, pp. 1222–1237, Sep. 2014.

[24] T. M. Shami, A. A. El-Saleh, M. Alswaitti, Q. Al-Tashi, M. A. Summakieh, and S. Mirjalili, "Particle swarm optimization: A comprehensive survey," *IEEE Access*, vol. 10, pp. 10031–10061, 2022.

[25] V. Kanwar and A. Kumar, "DV-hop localization methods for displaced sensor nodes in wireless sensor network using PSO," *Wireless Netw.*, vol. 27, no. 1, pp. 91–102, Jan. 2021.

[26] S. Surya and R. Ravi, "MPSO-SHM: Modified PSO based structural health monitoring system for detecting the faulty sensors in WSN," *Wireless Pers. Commun.*, vol. 108, no. 1, pp. 141–157, May 2019.

[27] A. Yang, Q. Zhang, Y. Liu, and J. Zhao, "The improvement of DV-hop model and its application in the security performance of smart campus," *Mathematics*, vol. 10, no. 15, p. 2663, Jul. 2022.

[28] L. Song, L. Zhao, and J. Ye, "DV-hop node location algorithm based on GSO in wireless sensor networks," *J. Sensors*, vol. 2019, pp. 1–9, Feb. 2019.

[29] D. S. A. Elminaam, A. Nabil, S. A. Ibraheem, and E. H. Houssein, "An efficient marine predators algorithm for feature selection," *IEEE Access*, vol. 9, pp. 60136–60153, 2021.

[30] T. Chen, Y. Chen, Z. He, E. Li, C. Zhang, and Y. Huang, "A novel marine predators algorithm with adaptive update strategy," *J. Supercomput.*, vol. 79, no. 6, pp. 6612–6645, Apr. 2023.

[31] Q. Fan, H. Huang, Q. Chen, L. Yao, K. Yang, and D. Huang, "A modified self-adaptive marine predators algorithm: Framework and engineering applications," *Eng. Comput.*, vol. 38, no. 4, pp. 3269–3294, Mar. 2021.

[32] K. Zhong, G. Zhou, W. Deng, Y. Zhou, and Q. Luo, "MOMPA: Multi-objective marine predator algorithm," *Comput. Methods Appl. Mech. Eng.*, vol. 385, Nov. 2021, Art. no. 114029.

DAN YU received the B.E. degree from the National University of Defense Technology, China, in 2003, and the M.E. degree in optical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2006.

He is currently a Lecturer with the School of Information Science and Engineering, Hunan Institute of Science and Technology, China. His research interests include wireless network algorithms, the Internet of Things, the combination of artificial intelligence and wireless communication, and signal processing and communication techniques.

TING YUAN received the B.S. degree from Ma'anshan University, Anhui, China, in 2020. She is currently pursuing the master's degree with the School of Information Science and Engineering, Hunan Institute of Science and Technology, China.

Her research interests include the Internet of Things, wireless sensors, intelligent algorithms, and artificial intelligence.

Ms. Yuan received several awards and honors, including the First-Class Scholarship of Hunan Institute of Technology, the Third Prize of the National Mathematical Modeling Award, and the Honorary Title of "Pacesetter of Three Good Graduate Students."

PAN LI received the B.S. degree from Changsha University of Science and Technology, Hunan, China, in 2020. He is currently pursuing the M.S. degree with the School of Information Science and Engineering, Hunan Institute of Science and Technology, China.

His research interests include semantic communication, wireless sensing, channel estimation, and decoding algorithms.

Mr. Li received several awards and honors, including the First-Class Scholarship of Hunan Institute of Technology, the Second Prize of the National Mathematical Modeling Award, and the Honorary Title of ''Outstanding Graduate Pacesetter of the Third Session.''

XINZHONG LIU received the B.S. and M.S. degrees from Jilin University, China, in 2001 and 2005, respectively, and the Ph.D. degree from Hunan University, in 2010.

His research interests include information security, edge computing, heterogeneous parallel computing, automated test technologies, software reliability evaluation, and wireless sensors.

Dr. Liu received several awards and honors, including the Third Prize of the 2019 Hunan Provincial Technological Invention Award, the Second Prize of the 2020 Changchun First Youth Science and Technology Innovation and Entrepreneurship Competition, and the Second Prize of the National Graduate Mathematical Modeling Competition, in 2021.

WENWU XIE received the B.S., M.S., and Ph.D. degrees in communication engineering from Central China Normal University, in 2004, 2006, and 2017, respectively.

He is currently a Professor with the School of Information Science and Engineering, Hunan Institute of Science and Technology, China. His research interests include communication algorithms, the Internet of Things, evolutionary computing, heterogeneous parallel computing, wireless sensors, and channel estimation algorithms.

Dr. Xie received several awards and honors, including the Outstanding Faculty Advisor, who has supervised students to win the first and second prizes in the National Graduate Mathematical Contest in Modeling.

ZHIHE YANG received the bachelor's and master's degrees in computer technology from the Huazhong University of Science and Technology, in 2003 and 2006, respectively.

He is currently a Professor with the School of Information Science and Engineering, Hunan Institute of Science and Technology, China. His main research interests include the Internet of Things, intelligent computing and intelligent information systems, intelligent algorithms, channel estimation, and wireless communication.