Hindawi
Journal of Mathematics
Volume 2023, Article ID 8810401, 13 pages
https://doi.org/10.1155/2023/8810401

Research Article

A Novel Fruit Fly Optimization Algorithm with Evolution Strategy for Magnetotelluric Data Inversion

Bin Yin,[1] **Jie Yang**,[2] **and Yue Li**[2]

[1]*Jiangxi University of Science and Technology, Nanchang, China*
[2]*Hubei Communications Planning and Design Institute Co., Ltd., Wuhan, China*

Correspondence should be addressed to Bin Yin; yinoobin123@163.com

Received 6 July 2023; Revised 29 August 2023; Accepted 4 September 2023; Published 22 September 2023

Academic Editor: Niansheng Tang

As a novel metaheuristic algorithm, fruit fly optimization algorithm (FOA) can effectively deal with the inversion problem of one-dimensional magnetotelluric data. However, FOA still has the disadvantage of premature convergence and falling into local extreme value. Therefore, based on standard FOA, we improve the FOA algorithm by introducing evolutionary strategies. Firstly, crossover and mutation strategies are introduced to improve the updating process of FOA population individuals. Secondly, by improving the variation scale factor, the global search and local search capabilities of the algorithm are balanced, and these improvements can accelerate the algorithm convergence. The improved algorithm is compared with other algorithms. After the benchmark function test, the improved algorithm has better optimization ability. Finally, the MT theoretical model and field data are used to test that the evolutionary strategy can effectively improve the convergence speed of the algorithm, and the inversion accuracy of the new algorithm is greatly improved.

1. Introduction

The magnetotelluric (MT) technology is an active electromagnetic exploration method using the natural electromagnetic field as the field source [1, 2]. The MT method can be applied to explore the electrical structure of the earth's interior according to the theory that electromagnetic waves have different frequenciesat different skin depths. Because the MT method has the characteristics of large detection depth (from the surface to thousands of kilometers underground), is not affected by the shielding of high resistance layer, and is sensitive to the low resistance layer, it has become one of the main methods to understand the deep electrical structure of the Earth. It is widely used in solid mineral exploration, oil and gas exploration, and other fields [3]. The inversion of MT data is to infer the underground geoelectric structure from the observation data. The inversion process is essentially an optimization problem, and an optimization algorithm is needed to realize it.

Inversion methods can be divided into linear inversion methods and nonlinear inversion methods. The early MT inversion methods are based on linear inversion theory, but the inversion problem of MT inversion is nonlinear in nature. The linear inversion method needs to linearize the nonlinear inversion problem and then use the optimization method to find the extreme value. The linearization process will inevitably produce errors, so the linear inversion algorithm depends on the initial model and is easy to fall into the local minimum. Compared with the linearization method, the nonlinear inversion method has advantages. The nonlinear inversion method treats the inversion problem as a direct solution of the nonlinear problem without systematic error and involves no matrix inversion calculation, thus reducing the computational complexity and improving the inversion accuracy. Although the dependence of nonlinear inversion algorithm on the initial model is greatly weakened, the search space increases geometrically with the increase of model parameters. Therefore, the development of efficient nonlinear inversion algorithm has important theoretical and practical significance.

Since the 1980s, with the development of computer technology, many classical optimization algorithms have

been proposed and applied to engineering calculation [4, 5], economics [6], optimization and scheduling [7–12], resource exploration, and many other fields. Especially in the 21st century, metaheuristic algorithms (also named intelligent optimization algorithms) have quickly become a research hotspot of optimization algorithms, and many excellent intelligent optimization algorithms have been proposed. According to the principle of the algorithm, it can be mainly divided into the following categories: (1) algorithms based on biological evolution theory: e.g., genetic algorithm (GA) [13–15] and differential evolution (DE) [16–19]; (2) algorithms based on swarm intelligence technology such as particle swarm optimization (PSO) [20–27], ant colony optimization (ACO) [28–30], artificial fish swarm algorithm (AFSA) [31, 32], and fruit fly optimization algorithm (FOA); and (3) algorithms based on physical phenomena such as simulated annealing (SA) [33–35] and water wave optimization algorithm (WWO) [8].

Fruit fly optimization algorithm is a novel swarm intelligence optimization algorithm proposed by Pan [6]. The algorithm is based on the foraging behavior of fruit flies, using their sense of smell and vision to sense their surroundings and food. The algorithm is easy to understand, easy to implement, and computationally efficient. Therefore, FOA is widely used in many fields such as in solving power load forecasting model [36], GRNN parameter optimization [37], joint replenishment problems [38], electricity consumption forecasting [5], multidimensional knapsack problem [39], structural optimization of transmission line tower [40], and energy consumption prediction [41].

Although FOA algorithm is widely used, it still has some problems such as insufficient global optimization ability, low precision, and slow convergence speed. Therefore, many scholars have improved FOA algorithm in their own ways. Pan et al. [42] brought a new control parameter and an effective solution generating method to improve the effectiveness of the FOA algorithm for solving continuous function optimization problems. Yuan et al. [43] employed multiswarm behavior to significantly improve the performance of FOA, named multiswarm fruit fly optimization algorithm (MFOA), and several subswarms moving independently in the search space with the aim of simultaneously exploring the global optimum at the same time and the local behavior between subswarms were also considered. Marko et al. [44] improved the standard FOA by introducing a novel parameter integrated with chaos. To improve the convergence performance of FOA, a normal cloud model-based FOA (CMFOA) was proposed by Wu et al. [45]. Lv et al. [46] proposed an improved FOA based on hybrid location information exchange mechanism (HFOA) aiming at improving the swarm diversity in a more efficient way and balancing the global search and local search abilities. Du et al. [47] proposed an improved FOA based on linear diminishing step and logistic chaos mapping (named DSLC-FOA) for solving constrained structural engineering design optimization problems. To solve both continuous function optimization and clustering parameter problems, Han et al. [48] proposed a novel FOA with trend search and coevolution (CEFOA); trend search was applied to enhance the local searching capability of fruit fly swarm, and coevolution mechanism was employed to avoid the premature convergence and improve the ability of global searching. Hu et al. [41] used the normal distribution function to improve the search mode of the FOA, named normal distribution fruit fly optimization algorithm (NFOA). It enhances search accuracy in the central area and effectively expands the search scope. Experimental results show that the accuracy and stability of the algorithm were improved.

The structure of this paper is as follows. Firstly, the basic FOA algorithm is introduced, then the improved strategy based on DE algorithm is proposed, and the test function is used to test the improved algorithm. The test results are compared with the results of other algorithms, and the results show that the optimization ability of the improved FOA algorithm is better than the above algorithms. Finally, the algorithm is applied to the MT data inversion problem.

2. The Basic Theory of the Fruit Fly Optimization Algorithm

The FOA is a new kind of global optimization algorithm that simulates the foraging process of a fruit fly. According to the process of a fruit fly searching for food, shown in Figure 1, the basic idea of the FOA can be found as follows. (1) Olfaction search stage: the fruit fly has well-developed olfaction. First, it uses its olfaction to detect various odors in the surrounding environment. (2) Visual positioning stage: the fruit fly flies to the vicinity of the food through the olfactory positioning within its visible range, accurately locates the food position through its vision, and flies to the food. Therefore, the FOA can be summarized into the following steps.

Step 1. Initialization: it includes the fruit fly swarm size, swarm location, and maximum number of iterations.

$$\begin{aligned} &\text{Init}X\text{\_axis},\\ &\text{Init}Y\text{\_axis}. \end{aligned} \tag{1}$$

Step 2. Provide a random direction and distance of individual fruit fly foraging. i stands for the ith fruit fly.

$$\begin{aligned} X_i &= X\text{\_axis} + \text{RandomValue},\\ Y_i &= Y\text{\_axis} + \text{RandomValue}. \end{aligned} \tag{2}$$

Step 3. Since the location of the food is unknown, first calculate the distance Dist between the individual fruit fly and the initial position and then calculate the smell concentration judgment value S according to Dist (Dist_i and S_i represent the distance between the ith fruit fly and the initial position and the smell concentration judgment value, respectively).

$$\begin{aligned} \text{Dist}_i &= \sqrt{X_i^2 + Y_i^2},\\ S_i &= \frac{1}{\text{Dist}_i}. \end{aligned} \tag{3}$$

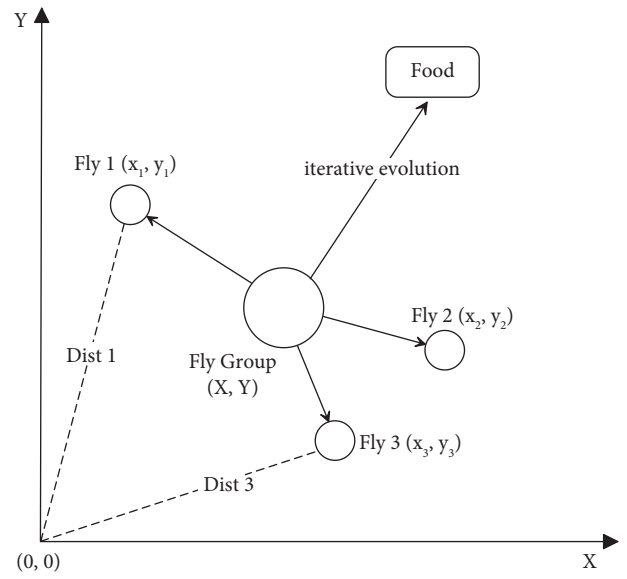

Figure 1: The interactive food search process of fruit fly swarm.

Step 4. Substitute the taste concentration judgment value S into the fitness function to calculate the taste concentration value of the individual fruit fly.

$$\text{Smell}_i = \text{Function}(S_i). \tag{4}$$

Step 5. Find the fruit fly with the best flavor concentration in the fruit fly swarm:

$$[\text{bestSmell bestIndex}] = \min(\text{Smell}). \tag{5}$$

Step 6. The optimal smell concentration value is saved along with the corresponding x, y coordinates, and the fruit fly flies to that location using its vision.

$$\begin{aligned} \text{Smellbest} &= \text{bestSmell}, \\ X\_\text{axis} &= X(\text{bestIndex}), \\ Y\_\text{axis} &= Y(\text{bestIndex}). \end{aligned} \tag{6}$$

Step 7. Execute the optimization repeatedly from Steps 2 to 5, each time, judge if the smell concentration is superior to the previous one, and if so, implement Step 6.

3. The Improvement of Fruit Fly Optimization Algorithm

The FOA has a wide range of applications in the field of engineering due to its advantages of fewer control parameters and no constraints. However, the evolutionary process of the FOA learns only from the best fruit fly in the evolutionary iteration process of the entire fruit fly swarm. When the optimal individual for this evolution is found, all individuals will gather towards the optimal individual and will search randomly within this small area. If the optimal individual of this evolution is only a local extremum, it will cause the algorithm to fall into the local optimum, from which it cannot leave, which makes the algorithm converge prematurely. Therefore, the algorithm needs to be improved.

The differential evolution (DE) algorithm is also a kind of swarm intelligent optimization algorithm [16, 49, 50]. Its core idea is to use the difference between different individuals in the population to carry out evolutionary optimization through crossover and mutation. Compared with the traditional genetic algorithm, it has the advantages of fast convergence, fewer control variables, being easy to understand, and programming implementation, so it has a wide range of applications [19, 51, 52].

The performance of DE algorithm is greatly affected by the control parameters of the algorithm. DE algorithm mainly has three control parameters: population size, variation scale factor, and crossover probability. Population size is always an important control parameter of evolutionary algorithms, which directly affects the population diversity of algorithms. If the population size is large enough, the population diversity performance is guaranteed, the optimization ability of the algorithm is enhanced, and the probability of obtaining the optimal solution is increased. However, the increase of population is accompanied by the decrease of computational efficiency. So, there is a trade-off between population diversity and computational efficiency.

Variation scale factor F and crossover probability are important control parameters of DE algorithm. Similar to the search step size, the size of the scale factor directly affects the optimization ability of the DE algorithm, and the global search ability and local search ability of the algorithm can be balanced by adjustment. The scale factor of the standard DE algorithm is fixed in the optimization process, and different scholars have proposed appropriate thresholds for some parameters according to the actual situation of the problem to be solved. Considering that most parameter settings depend on the problem, the artificial adjustment is extremely time-consuming. At present, the adaptive parameter adjustment method has become a hotspot in the research of DE algorithm. Many improvements to DE algorithms are based on adaptive scaling factors. In the early stage of algorithm evolution, appropriately setting a large scale factor will help the algorithm quickly search the near optimal solution in the early stage and reduce the scale factor in the later stage, so that the algorithm can quickly find the optimal solution.

In order to enhance the optimization ability of FOA, this study introduces the crossover and mutation process from the DE algorithm into the FOA and proposes a novel FOA algorithm based on the DE algorithm.

In the standard FOA, each search step size of a fruit fly is a random value under a fixed step size, which leads to the poor ability of the algorithm to jump out of the local extreme values. Therefore, we replaced the standard FOA with a mutation operation. After comparing several mutation strategies, we chose the "DE/best/2" mutation strategy [53]. After the mutation operation, the obtained difference vector and the optimal individual of the parent generation were crossed again to obtain the final search step size. In the DE algorithm, in order to improve the optimization ability of the

algorithm, the method of gradually decreasing the scale factor was adopted. The improvement strategy was as follows.

3.1. Search Step Size Improvement Strategy. The search step size of the standard fruit fly optimization algorithm is fixed, that is, the individual search step size is randomly generated under a given step size value, the mutation operation of the differential evolution algorithm was introduced, and the step size was updated based on the difference of the population individuals, enhancing the algorithm's optimization capabilities.

$$\begin{cases} U_{x,j} = X\_\text{axis} + F \times \left(X_{r_1,j} - X_{r_2,j}\right) + F \times \left(X_{r_3,j} - X_{r_4,j}\right), \\ U_{y,j} = Y\_\text{axis} + F \times \left(Y_{r_1,j} - Y_{r_2,j}\right) + F \times \left(Y_{r_3,j} - Y_{r_4,j}\right), \end{cases} \tag{7}$$

where r_1, r_2, r_3, r_4 are random numbers between the intervals [1, NP] and NP represents the swarm size. In order to further increase the diversity of the fruit fly swarm, we continue to use the crossover operation to update the mutation operation.

$$\text{RandomValue}_x = \begin{cases} U_{x,j}, & \text{if rand} \le \text{CR or j} = \text{randn (j)}, \\ X\_\text{axis}, & \text{otherwise}, \end{cases} \tag{8}$$

$$\text{RandomValue}_y = \begin{cases} U_{y,j}, & \text{if rand} \le \text{CR or j} = \text{randn (j)}, \\ Y\_\text{axis}, & \text{otherwise}, \end{cases} \tag{9}$$

where rand is a random number uniformly distributed in the interval [0, 1], CR is the crossover probability, randn (j) is a random integer between [1, N], and N represents the dimension of the unknown quantity.

3.2. Variation Scale Factor Improvement Strategy. In the early stage of the algorithm, since the swarm position is far from the optimal solution, a larger step size can improve the global search ability of the algorithm, so that the fruit fly individual can quickly find the vicinity of the optimal solution. The step size can also improve the local search ability of the algorithm, to quickly find the optimal solution. Therefore, based on the above ideas, the variation scale factor was improved so that it gradually decreased in the iterative process of the algorithm, and the local search ability of the algorithm was increased while ensuring the diversity of the population. The scaling factor improvement formula is

$$F_{t+1} = F_t - \frac{F_0}{t_{\max}}, \tag{10}$$

where t is the current number of iterations, $t_{\max}$ is the maximum number of iterations, and F_0 is the initial scale factor, which is generally in the range of [0.4, 0.9].

3.3. The Steps of the Improved Fruit Fly Optimization Algorithm (IFOA)

Step 1. Initialization: it includes fruit fly swarm size, swarm location, maximum number of iterations, and mutation scale factor.

$$\begin{aligned} &\text{Init}X\_\text{axis}, \\ &\text{Init}Y\_\text{axis}, \end{aligned} \tag{11}$$

Step 2. Execute Steps 2 to 6 of the original FOA.

$$\begin{aligned} \text{Smellbest} &= \text{bestSmell}, \\ X\_\text{axis} &= X\,(\text{bestIndex}), \\ Y\_\text{axis} &= Y\,(\text{bestIndex}). \end{aligned} \tag{12}$$

Step 3. Perform mutation and crossover operations according to formulas (7)–(9).

Step 4. Update the positions of fruit flies using the candidate values from Step 3.

$$\begin{aligned} X_i &= X\_\text{axis} + \text{RandomValue}_x, \\ Y_i &= Y\_\text{axis} + \text{RandomValue}_y. \end{aligned} \tag{13}$$

Step 5. Execute the iterative optimization repeatedly from Steps 2 to 4, each time judging if the smell concentration is superior to the previous smell concentration until the maximum number of iterations is reached.

4. Experiments and Numerical Analysis

4.1. Benchmark Functions and Algorithm Parameter Setting. In order to test the performance of IFOA algorithm, 14 benchmark functions which are shown in Table 1 are used in this section to test the algorithm. We selected 14 classical benchmark functions for testing, among which F1–F10 are unimodal and F11–F14 are multimodal. Unimodal function is relatively simple, while multimodal function is more complex, with local extreme values and other complicated cases. The algorithms discussed in this section were coded in Matlab 2017b, and computations were conducted on a personal computer with Intel(R) Core(TM) i7-9700 CPU 3.00 GHz, 32 GB RAM, and Windows 10 operation system.

4.2. The Influence of the Two Introduced Strategies. In order to find out the effectiveness of the above two improvement strategies, we also conducted a set of comparison tests. F was fixed as a constant and the comparison algorithm was named IFOA-1. Three classic benchmark functions (from the functions in Table 1) were also used to test the three algorithms of standard FOA, IFOA, and IFOA-1 to verify the

Table 1: The classical benchmark functions (M: multimodal, U: unimodal, Dim: dimension, Range: limit of search space, and Optimum: global optimal value).

Name	Function	Type	Range	Dim	Optimum
Sphere	$f_1(x) = \sum_{i=1}^{n} x_i^2$	U	$[-100, 100]$	30	0
Schwefel's problem 2.22	$f_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	U	$[-10, 10]$	30	0
Schwefel's problem 1.2	$f_3(x) = \sum_{i=1}^{n} (\sum_{j=1}^{i} x_j)^2$	U	$[-100, 100]$	30	0
Schwefel's problem 2.21	$f_4(x) = \max_i \{\lvert x_i \rvert, 1 \le i \le n\}$	U	$[-100, 100]$	30	0
Rosenbrock	$f_5(x) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	U	$[-30, 30]$	30	0
Step	$f_6(x) = \sum_{i=1}^{n-1} (\lfloor x_i + 0.5 \rfloor)^2$	U	$[-100, 100]$	30	0
Zakharov	$f_7(x) = \sum_{i=1}^{n-1} x_i^2 + (\sum_{i=1}^{n-1} 0.5 \cdot i x_i)^2 + (\sum_{i=1}^{n-1} 0.5 \cdot i x_i)^4$	U	$[-5, 10]$	30	0
Sum of different powers	$f_8(x) = \sum_{i=1}^{n-1} \lvert x_i \rvert^{i+1}$	U	$[-1, 1]$	30	0
Sum of squares	$f_9(x) = \sum_{i=1}^{n-1} i x_i^2$	U	$[-10, 10]$	30	0
Quartic	$f_{10}(x) = \sum_{i=1}^{n-1} i x_i^4 + \text{random}[0, 1)$	U	$[-1.28, 1.28]$	30	0
Levy	$f_{11}(x) = \sin^2(\pi\omega_1) \sum_{i=1}^{n-1} (\omega_i - 1)^2 [1 + 10\sin^2(\pi\omega_i + 1)] + (\omega_n - 1)^2 [1 + \sin^2(2\pi\omega_d)]$ $\omega_i = 1 + x_i - 1/4, i = 1, \cdots, n$	M	$[-10, 10]$	30	0
Ackley	$f_{12}(x) = -20\exp(-0.2\sqrt{1/n \sum_{i=1}^{n} x_i^2}) - \exp(1/n \sum_{i=1}^{n} \cos(2\pi x_i)) + 20 + e$	M	$[-32, 32]$	30	0
Rastrigin	$f_{13}(x) = \sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi x_i) + 10)$	M	$[-5.12, 5.12]$	30	0
Griewank	$f_{14}(x) = 1/4000 \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(x_i/\sqrt{i}) + 1$	M	$[-600, 600]$	30	0

effectiveness of the improved strategies. The population is set to 100. Due to the need to gradually decrease, the initial F of IFOA is set to 1, the F of IFOA-1 is set to the constant 0.5, and the number of iterations is 100. The results of mean and standard deviation are shown in Tables 2–4.

Through the above comparative test, it is found that compared with FOA, the convergence speed of the improved algorithm is faster, which is mainly due to the crossover and mutation strategies. The improvement of F has little effect on the algorithm.

4.3. Comparison with Other Algorithms. In this section, we verify the performance of the IFOA algorithm. The above test functions have both unimodal and multimodal functions, so the global search ability and local search ability of the algorithm are considered the key to finding the optimal solution quickly. For comparison, we choose three common intelligent optimization algorithms, DE, PSO, and GWO. In order to test the performance of the algorithm in high-dimensional space, we set two schemes with parameter dimensions of 30 and 50, respectively. The parameter settings of each algorithm are shown in Table 5. The number of iterations is set to 5000 times, each algorithm runs independently for 30 times, and then the mean value and standard deviation of the optimal solution are obtained. The results are shown in Tables 6 and 7.

As shown in Table 6, when Dim = 30, for functions F1–F14, except F5 and F11, IFOA algorithm performs well, and the mean and standard deviation results are superior to other algorithms. For most test functions, IFOA can find the true optimal solution directly. For F5, the DE algorithm is superior, and the case of Dim = 50 is similar to that of Dim = 30.

In order to more intuitively compare the performance of these algorithms, we sort and score them according to their optimization ability (the smaller the optimal solution, the higher the score; the same optimal solution, the same score). The results of Dim = 30 are shown in Table 8. It can be seen from the results that, except for some functions, the scores of IFOA are all above 4 points. IFOA scores 63. The final ranking results show that IFOA is an excellent optimization algorithm, which not only has global search ability and local search ability but also has good robustness and universality.

Table 2: Optimization iteration curve of Zakharov function.

Iterations	FOA	IFOA-1	IFOA
1	$1.15E+04$	$1.01E+05$	$1.81E+03$
20	$1.15E+01$	$9.18E-05$	$3.91E-25$
40	$1.49E+00$	$6.30E-12$	$1.47E-42$
60	$5.24E-01$	$3.99E-18$	$4.66E-51$
80	$2.64E-01$	$2.77E-25$	$1.55E-54$
100	$1.58E-01$	$6.57E-33$	$3.19E-57$

Table 3: Optimization iteration curve of Rastrigin function.

Iterations	FOA	IFOA-1	IFOA
1	$1.48E+01$	$3.62E+01$	$4.75E+00$
20	$3.44E-01$	$1.57E-04$	$0.00E+00$
40	$9.27E-02$	$4.59E-09$	$0.00E+00$
60	$4.14E-02$	$0.00E+00$	$0.00E+00$
80	$2.35E-02$	$0.00E+00$	$0.00E+00$
100	$1.52E-02$	$0.00E+00$	$0.00E+00$

Table 4: Optimization iteration curve of Griewank function.

Iterations	FOA	IFOA-1	IFOA
1	$3.22E-03$	$1.45E-02$	$2.48E-03$
20	$9.86E-05$	$2.09E-08$	$0.00E+00$
40	$2.86E-05$	$5.60E-14$	$0.00E+00$
60	$1.33E-05$	$0.00E+00$	$0.00E+00$
80	$7.79E-06$	$0.00E+00$	$0.00E+00$
100	$5.04E-06$	$0.00E+00$	$0.00E+00$

Table 5: Parameter setting for comparison algorithms.

Algorithm	Parameters
FOA	No special parameters
DE	$F=0.5$, CR = 0.9
PSO	$\omega=0.6, c_1=c_2=2$
GWO	a = 2-2 (g/max_iter), max_iter: maximum iteration
IFOA	The improvement strategy described above, $F_0=1$, CR = 0.9

5. Optimization of MT Data

5.1. MT Forward Modeling. Maxwell's equations are the theoretical basis of all electromagnetic methods. According to Cagniard's classical magnetotelluric theory, the field source is assumed to be a plane electromagnetic wave incident vertically on the ground, the Earth medium is a uniform horizontal layered model, and the electrical properties of each medium are uniform and isotropic. Consider a one-dimensional layered medium; assuming that the underground medium consists of n layers of horizontal layered medium, there are a total of $2n-1$ parameters.

$$m=(\rho_1,\rho_2,\ldots,\rho_n,h_1,h_2,\ldots,h_{n-1})^T, \tag{14}$$

where ρ is the conductivity, h is the depth, and $h_n=\infty$; for the 1D layered model, the apparent resistivity $\rho_a(\omega)$ and phase $\varphi(\omega)$ can be observed on the ground and the formulas are as follows:

$$\rho_a(\omega)=\frac{1}{\omega\mu}|Z(\omega)|^2,$$
$$\varphi(\omega)=\arctan\left(\frac{\operatorname{Im}Z(\omega)}{\operatorname{Re}Z(\omega)}\right), \tag{15}$$

where $\omega=2\pi/\mathrm{T}$ is the angular frequency, μ is the magnetic permeability, and $Z(\omega)$ is the wave impedance, which can be calculated by the following formula:

Table 6: The performance on benchmark functions with dimension = 30.

		FOA	DE	PSO	IFOA	GWO
F01	Avg	$3.13E-08$	$1.27E-63$	$1.76E-40$	$0.00E+00$	$0.00E+00$
	Std	$7.27E-11$	$2.92E-63$	$3.05E-40$	$0.00E+00$	$0.00E+00$
F02	Avg	$9.69E-04$	$3.09E-31$	$9.27E-27$	$0.00E+00$	$5.73E-251$
	Std	$1.07E-06$	$3.13E-31$	$1.25E-26$	$0.00E+00$	$0.00E+00$
F03	Avg	$9.86E-06$	$1.55E-11$	$3.28E+00$	$0.00E+00$	$3.96E-144$
	Std	$1.78E-08$	$3.20E-11$	$2.27E+00$	$0.00E+00$	$2.15E-143$
F04	Avg	$3.23E-05$	$2.15E-02$	$1.91E-01$	$0.00E+00$	$2.51E-111$
	Std	$3.23E-08$	$6.89E-02$	$1.06E-01$	$0.00E+00$	$5.45E-111$
F05	Avg	$2.87E+01$	$9.74E-18$	$3.52E+01$	$2.87E+01$	$2.61E+01$
	Std	$2.04E-14$	$2.58E-17$	$3.21E+01$	$7.07E-02$	$8.53E-01$
F06	Avg	$3.03E-02$	$1.42E-57$	$4.16E-34$	$0.00E+00$	$0.00E+00$
	Std	$5.95E-05$	$4.14E-57$	$1.10E-33$	$0.00E+00$	$0.00E+00$
F07	Avg	$5.64E-05$	$6.09E-15$	$8.75E-04$	$0.00E+00$	$2.37E-205$
	Std	$1.23E-07$	$7.11E-15$	$7.91E-04$	$0.00E+00$	$0.00E+00$
F08	Avg	$2.89E-04$	$8.87E-95$	$3.11E-70$	$0.00E+00$	$0.00E+00$
	Std	$1.58E-03$	$4.57E-94$	$1.70E-69$	$0.00E+00$	$0.00E+00$
F09	Avg	$4.85E-07$	$2.50E-64$	$1.53E-39$	$0.00E+00$	$0.00E+00$
	Std	$9.18E-10$	$5.11E-64$	$7.68E-39$	$0.00E+00$	$0.00E+00$
F10	Avg	$4.82E-05$	$1.71E-02$	$7.51E-03$	$2.41E-05$	$3.84E-05$
	Std	$1.57E-05$	$1.01E-02$	$2.50E-03$	$2.74E-05$	$2.15E-05$
F11	Avg	$5.02E-09$	$1.50E-32$	$1.50E-32$	$2.86E+00$	$7.77E-01$
	Std	$7.17E-09$	$1.11E-47$	$1.11E-47$	$1.43E-01$	$2.07E-01$
F12	Avg	$1.29E-04$	$3.55E-15$	$1.86E-14$	$0.00E+00$	$6.51E-15$
	Std	$1.03E-07$	$0.00E+00$	$3.92E-15$	$0.00E+00$	$1.35E-15$
F13	Avg	$6.22E-06$	$7.51E+01$	$1.66E+01$	$0.00E+00$	$0.00E+00$
	Std	$1.29E-08$	$2.74E+01$	$5.60E+00$	$0.00E+00$	$0.00E+00$
F14	Avg	$2.09E-09$	$2.47E-04$	$1.76E-02$	$0.00E+00$	$0.00E+00$
	Std	$4.09E-12$	$1.35E-03$	$1.64E-02$	$0.00E+00$	$0.00E+00$

Table 7: The performance on benchmark functions with dimension = 50.

		FOA	DE	PSO	IFOA	GWO
F01	Avg	$5.22E-08$	$1.44E-39$	$1.31E-17$	$0.00E+00$	$2.42E-322$
	Std	$9.64E-11$	$1.67E-39$	$2.44E-17$	$0.00E+00$	$0.00E+00$
F02	Avg	$1.62E-03$	$7.28E-20$	$2.09E-13$	$0.00E+00$	$1.16E-187$
	Std	$1.43E-06$	$5.30E-20$	$2.30E-13$	$0.00E+00$	$0.00E+00$
F03	Avg	$4.48E-05$	$1.11E+00$	$3.56E+03$	$0.00E+00$	$6.49E-85$
	Std	$8.99E-08$	$5.23E-01$	$1.04E+03$	$0.00E+00$	$3.09E-84$
F04	Avg	$3.23E-05$	$5.61E+00$	$1.50E+01$	$0.00E+00$	$2.72E-73$
	Std	$3.07E-08$	$2.44E+00$	$2.82E+00$	$0.00E+00$	$9.48E-73$
F05	Avg	$4.85E+01$	$1.63E+01$	$9.36E+01$	$4.86E+01$	$4.59E+01$
	Std	$2.68E-14$	$2.43E+00$	$5.11E+01$	$6.40E-02$	$8.79E-01$
F06	Avg	$5.11E-02$	$1.27E-33$	$1.36E-11$	$0.00E+00$	$7.58E-317$
	Std	$1.25E-04$	$1.45E-33$	$5.17E-11$	$0.00E+00$	$0.00E+00$
F07	Avg	$4.24E-04$	$1.39E-01$	$9.56E+00$	$0.00E+00$	$4.78E-123$
	Std	$8.15E-07$	$7.40E-02$	$3.74E+00$	$0.00E+00$	$2.43E-122$
F08	Avg	$1.04E-09$	$2.66E-25$	$5.26E-34$	$0.00E+00$	$0.00E+00$
	Std	$2.27E-12$	$1.42E-24$	$1.20E-33$	$0.00E+00$	$0.00E+00$
F09	Avg	$1.33E-06$	$3.07E-40$	$3.67E-18$	$0.00E+00$	$1.16E-321$
	Std	$2.50E-09$	$3.97E-40$	$7.07E-18$	$0.00E+00$	$0.00E+00$
F10	Avg	$7.10E-05$	$2.30E-02$	$3.91E-02$	$1.76E-05$	$5.41E-05$
	Std	$2.23E-05$	$1.49E-02$	$9.65E-03$	$1.59E-05$	$2.87E-05$
F11	Avg	$1.33E-08$	$1.50E-32$	$3.59E+00$	$4.68E+00$	$1.99E+00$
	Std	$1.89E-08$	$1.11E-47$	$3.86E+00$	$1.07E-01$	$3.04E-01$

Table 7: Continued.

		FOA	DE	PSO	IFOA	GWO
F12	Avg	1.29E − 04	4.74E − 15	1.21E − 09	0.00E + 00	8.64E − 15
	Std	1.09E − 07	1.70E − 15	1.27E − 09	0.00E + 00	2.41E − 15
F13	Avg	1.04E − 05	2.09E + 02	7.33E + 01	0.00E + 00	0.00E + 00
	Std	1.76E − 08	5.81E + 01	1.77E + 01	0.00E + 00	0.00E + 00
F14	Avg	2.36E − 09	0.00E + 00	6.48E − 03	0.00E + 00	0.00E + 00
	Std	4.40E − 12	0.00E + 00	9.33E − 03	0.00E + 00	0.00E + 00

Table 8: Ranking of FOA, DE, PSO, IFOA, and GWO on 14 benchmark functions with Dim = 30 according to their performance.

Function	FOA	DE	PSO	IFOA	GWO
F01	2	4	3	5	5
F02	1	3	2	5	4
F03	2	3	1	5	4
F04	3	2	1	5	4
F05	3	5	1	2	4
F06	2	4	3	5	5
F07	1	3	2	5	4
F08	2	4	3	5	5
F09	2	4	3	5	5
F10	3	1	2	5	4
F11	3	4	5	1	2
F12	1	4	2	5	3
F13	4	2	3	5	5
F14	4	3	2	5	5
SUM	33	46	33	63	59
Ranking	4	3	4	1	2

$$Z_m = Z_{0m}\frac{Z_{0m}\left(1 - e^{-2k_m h_m}\right) + Z_{m+1}\left(1 + e^{-2k_m h_m}\right)}{Z_{0m}\left(1 + e^{-2k_m h_m}\right) + Z_{m+1}\left(1 - e^{-2k_m h_m}\right)}, \qquad Z_{0m} = -\frac{i\omega\mu}{k_m}, \tag{16}$$

where Z_{0m} is the characteristic impedance of layer m, k_m is the complex propagation constant of layer m, and Z_m is the wave impedance at the top of layer m.

5.2. MT Inversion Theory. The geophysical inversion problem is to study the theory and method of using the obtained geophysical observation data to deduce the characteristics of the underground geophysical model. The purpose of the inversion problem is to find the appropriate geophysical model to fit the observation data. Therefore, geophysical inversion is essentially an optimization problem. The inversion problem can be expressed as

$$d = F(m), \tag{17}$$

where d is the N-dimensional observation data vector, m is the M-dimensional model parameter vector, and F is the model forward response function. Geophysical inversion aims to find a reasonable underground model parameter vector m to fit the observation data d. The inversion problem is an ill-posed problem, so it is necessary to obtain a stable solution through the regularization method. According to regularization theory, we adopted the Occam-like regularization [54]. Therefore, the objective function of the inversion problem is as follows:

$$\Phi = \Phi_d + \lambda\Phi_m = \left\|F(m) - d^{\text{obs}}\right\|_2 + \lambda\|\partial m\|_2, \tag{18}$$

where Φ_d is the data fitness, Φ_m is the model parameter constraint, and λ is the regularization factor, set as the trade-off between the model and data misfit to regulate the model roughness. d^{obs} are the observation data, and $F(m)$ are the predicted response data.

5.3. Numerical Results of the 1D Inverse Modeling. In order to test this new algorithm, we designed two theoretical models of layered media, and the test environment was a personal computer. The model parameters are shown in Table 9. The model parameter search interval was set to [0.5 m, 2 m], where m represents the true value of the model parameters, the fruit fly swarm size was 10, the crossover factor CR = 0.9, and the variation factor $F = 0.5$.

Model 1 was a three-layer geoelectric model. The maximum number of iterations was 1000. In order to ensure the reliability of the results, the calculation was repeated 20 times continuously, and the average value was taken as the result. The inversion results are shown in Figure 2. It can be seen from the figure that both Bostick and IFOA obtained a good result. The inversion results of the IFOA were closer to the real model than the Bostick inversion results. The low resistivity layer was well inverted, and the apparent resistivity and phase results of the prediction model fit well with the observed data.

Table 9: The parameters of model 1 and model 2.

Model 1		Model 2	
Resistivity ($\Omega \cdot$ m)	Thickness (m)	Resistivity ($\Omega \cdot$ m)	Thickness (m)
10	500	200	200
200	2500	10	10
20		200	300
		300	

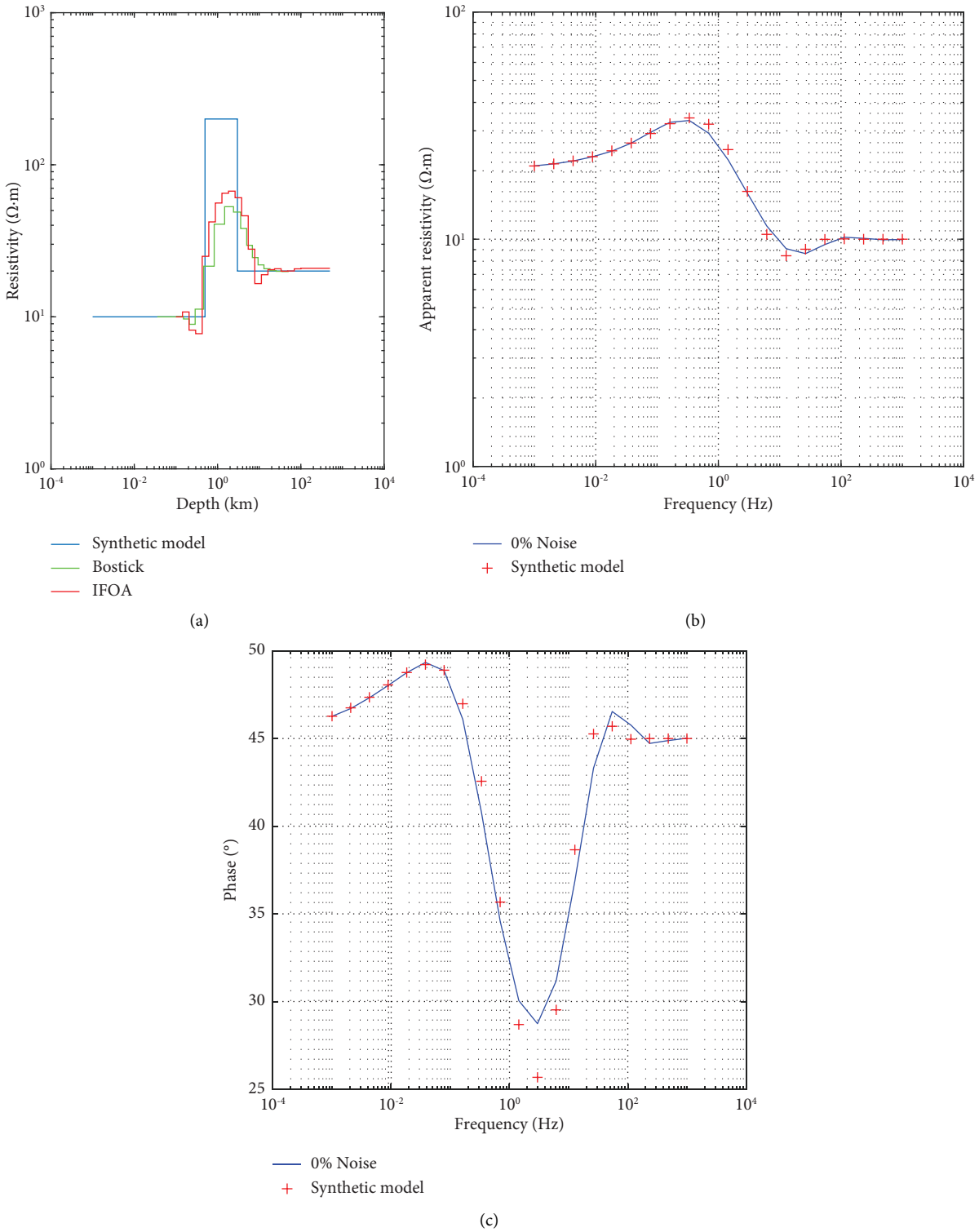

Figure 2: The three-layer geoelectric model and its MT response predicted by IFOA and Bostick inversion. (a) represents the results of the resistivity from the IFOA and Bostick inversions. (b) and (c) are the response results of the apparent resistivity and phase, respectively.

Table 10: The IFOA optimization results of the synthetic data from model 2.

Parameter	$\rho_1(\Omega \cdot m)$	$\rho_2(\Omega \cdot m)$	$\rho_3(\Omega \cdot m)$	$\rho_4(\Omega \cdot m)$	$h_1(m)$	$h_2(m)$	$h_3(m)$
Real	200	10	200	300	200	10	300
No noise	199.9411	10.5728	195.9431	299.9659	199.3871	10.4977	289.7938
10% noise	199.9834	11.6584	157.0820	299.4285	197.3866	9.3255	208.1459
20% noise	199.9976	14.3623	137.1937	299.9962	199.9884	14.0154	150.0008

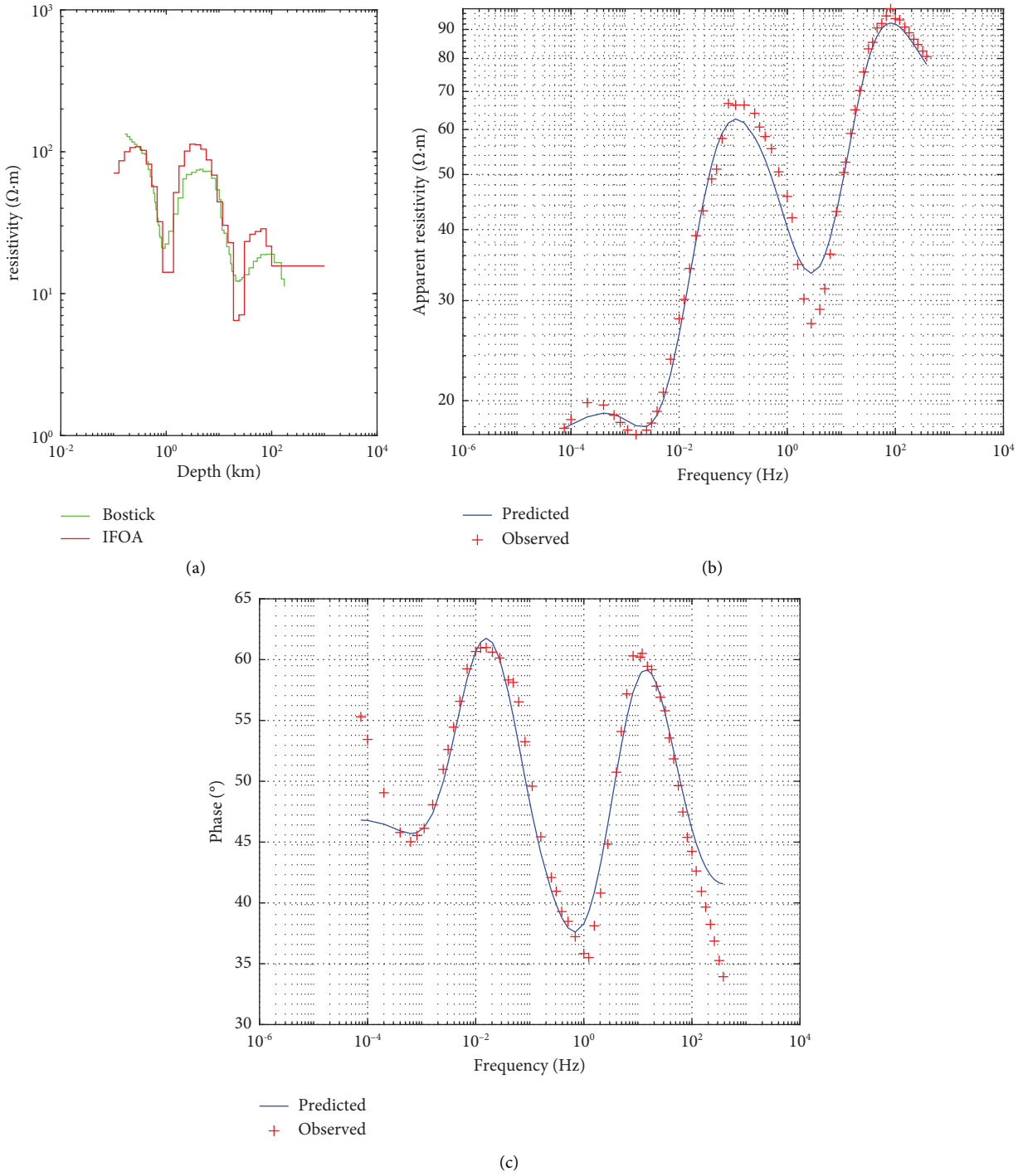

Figure 3: The IFOA optimization of the measured MT sounding. (a) represents the results of the resistivity from the IFOA and Bostick inversions. (b) and (c) are the response results of the apparent resistivity and phase, respectively.

Model 2 was a four-layer geoelectric model. In order to study the influence of the equivalence problem on the algorithm optimization capability, a four-layer (HA type) geoelectric model was designed. The parameters of this model are shown in Table 9, and it has S equivalence. The maximum number of iterations was set to 5000. To illustrate the influence of data errors on the inversion results, 10% and 20% of Gaussian random noise were added to the original model forward results, respectively. We repeated the calculation 20 times in a row and took the average value as the result. The inversion calculation results are shown in Table 10. It can be seen from the table that the IFOA inverted the model parameters well and had good robustness.

5.4. Field Data. In order to further verify the feasibility of the algorithm, the IFOA was used to invert the real MT data. The data came from the open-source data, US Array MT stations, by Oregon State University [55]. The inversion result is shown in Figure 3. It can be seen from the figure that the apparent resistivity and phase of the inversion model fit the measured data well. Two low-resistivity layers were well inverted.

6. Conclusions

As a swarm intelligence global optimization algorithm, IFOA can be fully applied to MT data inversion through the inversion of a theoretical model and field data. Through improvement, the algorithm overcame the equivalence problem of the geoelectric model and accurately restored the deep high-conductivity and low-resistance layers. This shows that the IFOA can be used for one-dimensional inversion of MT data. The IFOA avoids the defect of the linearization optimization theory needing to calculate the Jacobian matrix, which easily falls into local extreme values, and it has better global optimization ability. The inversion results of the measured data confirmed the capability of IFOA.

The following research will continue in the future. First, research on an intelligent optimization algorithm will be conducted. At present, due to the progress of science and technology and the development of computer hardware, the complete nonlinear global optimization algorithm has been a research hotspot in the field of mathematics and engineering, asking how to improve the optimization ability of the algorithm and how to combine different optimization algorithms, to learn from each other, and put forward a more suitable optimization method for geophysical inversion. These are questions for the next step. Second, the two-dimensional inversion of the fruit fly optimization algorithm will be realized. Since time is limited, this paper implements only the one-dimensional inversion of the optimization algorithm, which is not enough; the next step will continue to excavate the potential of the FOA with the study of two-dimensional magnetotelluric data inversion. Due to the data, the model parameters of 2D case, and the need to consider the parameters of the space constraints, this is a challenge for a global optimization algorithm.

Data Availability

Datasets used in experiments are public and they were retrieved from the following URL: https://ds.iris.edu/spud/emtf.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was mainly supported by the scientific research project of the Jiangxi Provincial Department of Education (no. GJJ180492).

References

[1] A. N. Tikhonov, "On determining electric characteristics of the deep layers of the Earth's crust," *Doklady Akademii Nauk SSSR*, vol. 73, no. 2, 1950.

[2] L. Cagniard, "Basic theory of the magneto-telluric method of geophysical prospecting," *Geophysics*, vol. 18, no. 3, pp. 605–635, 1953.

[3] N. Yu, E. Wang, and X. Wang, "The influence of the ailaoshan-red river shear zone on the mineralization of the beiya deposit on the southeastern margin of the Tibetan plateau revealed by a 3-D magnetotelluric survey," *Journal of Geophysical Research: Solid Earth*, vol. 127, no. 2, Article ID e2021JB022923, 2022.

[4] J. Niu, W. Zhong, and Y. Liang, "Fruit fly optimization algorithm based on differential evolution and its application on gasification process operation optimization," *Knowledge-Based Systems*, vol. 88, pp. 253–263, 2015.

[5] G. Cao and L. Wu, "Support vector regression with fruit fly optimization algorithm for seasonal electricity consumption forecasting," *Energy*, vol. 115, pp. 734–745, 2016.

[6] W.-T. Pan, "A new Fruit Fly Optimization Algorithm: taking the financial distress model as an example," *Knowledge-Based Systems*, vol. 26, pp. 69–74, 2012.

[7] J. J. Wang and L. Wang, "A cooperative memetic algorithm with feedback for the energy-aware distributed flow-shops with flexible assembly scheduling," *Computers & Industrial Engineering*, vol. 168, Article ID 108126, 2022.

[8] F. Zhao, L. Zhang, and J. Cao, "A cooperative water wave optimization algorithm with reinforcement learning for the distributed assembly no-idle flowshop scheduling problem," *Computers & Industrial Engineering*, vol. 153, Article ID 107082, 2021.

[9] F. Zhao, T. Jiang, and L. Wang, "A reinforcement learning driven cooperative meta-heuristic algorithm for energy-efficient distributed No-wait flow-shop scheduling with sequence-dependent setup time," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 7, pp. 8427–8440, 2023.

[10] F. Zhao, S. Di, and L. Wang, "A hyperheuristic with Q-learning for the multiobjective energy-efficient distributed blocking flow shop scheduling problem," *IEEE Transactions on Cybernetics*, vol. 53, no. 5, pp. 3337–3350, 2023.

[11] F. Zhao, H. Zhang, and L. Wang, "A pareto-based discrete jaya algorithm for multiobjective carbon-efficient distributed blocking flow shop scheduling problem," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 8, pp. 8588–8599, 2023.

[12] Z. Pan, D. Lei, and L. Wang, "A knowledge-based two-population optimization algorithm for distributed energy-efficient parallel machines scheduling," *IEEE Transactions on Cybernetics*, vol. 52, no. 6, pp. 5051–5063, 2022.

[13] P. L. Stoffa and M. K. Sen, "Nonlinear multiparameter optimization using genetic algorithms: inversion of plane-wave seismograms," *Geophysics*, vol. 56, no. 11, pp. 1794–1810, 1991.

[14] M. A. Pérez-Flores and A. Schultz, "Application of 2-D inversion with genetic algorithms to magnetotelluric data from geothermal areas," *Earth Planets and Space*, vol. 54, no. 5, pp. 607–616, 2002.

[15] J. Da Conceição Batista and E. E. S. Sampaio, "Magnetotelluric inversion of one- and two-dimensional synthetic data based on hybrid genetic algorithms," *Acta Geophysica*, vol. 67, no. 5, pp. 1365–1377, 2019.

[16] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[17] M. Bilal and M. Pant, "Differential Evolution: a review of more than two decades of research," *Engineering Applications of Artificial Intelligence*, vol. 90, Article ID 103479, 2020.

[18] A. P. Piotrowski, "Review of differential evolution population size," *Swarm and Evolutionary Computation*, vol. 32, pp. 1–24, 2017.

[19] A. Roy, C. P. Dubey, and M. Prasad, "Gravity inversion for heterogeneous sedimentary basin with b-spline polynomial approximation using differential evolution algorithm," *Geophysics*, vol. 86, no. 3, pp. F35–F47, 2021.

[20] R. Shaw and S. Srivastava, "Particle swarm optimization: a new tool to invert geophysical data," *Geophysics*, vol. 72, no. 2, pp. F75–F83, 2007.

[21] J. L. Fernández Martínez, E. García Gonzalo, and J. P. Fernández Álvarez, "PSO: a powerful algorithm to solve geophysical inverse problems," *Journal of Applied Geophysics*, vol. 71, no. 1, pp. 13–25, 2010.

[22] A. Santilano, A. Godio, and A. Manzella, "Particle swarm optimization for simultaneous analysis of magnetotelluric and time-domain electromagnetic data," *Geophysics*, vol. 83, no. 3, pp. E151–E159, 2018.

[23] A. Godio and A. Santilano, "On the optimization of electromagnetic geophysical data: application of the PSO algorithm," *Journal of Applied Geophysics*, vol. 148, pp. 163–174, 2018.

[24] F. Pace, A. Santilano, and A. Godio, "Particle swarm optimization of 2D magnetotelluric data," *Geophysics*, vol. 84, no. 3, pp. E125–E141, 2019.

[25] F. M. Barboza, W. E. Medeiros, and J. M. Santana, "A user-driven feedback approach for 2D direct current resistivity inversion based on particle swarm optimization," *Geophysics*, vol. 84, no. 2, pp. E105–E124, 2019.

[26] Y. A. Cui, L. Zhang, and X. Zhu, "Inversion for magnetotelluric data using the particle swarm optimization and regularized least squares," *Journal of Applied Geophysics*, vol. 181, Article ID 104156, 2020.

[27] A. Roy, C. P. Dubey, and M. Prasad, "Gravity inversion of basement relief using Particle Swarm Optimization by automated parameter selection of Fourier coefficients," *Computers & Geosciences*, vol. 156, Article ID 104875, 2021.

[28] S. Chen, S. Wang, and M. Ji, "The ant colony algorithm for the seismic impedance inversion," *Geophysical Prospecting for Petroleum*, vol. 44, pp. 551–553, 2005.

[29] S. Liu, X. Hu, and T. Liu, "A stochastic inversion method for potential field data: ant colony optimization," *Pure and Applied Geophysics*, vol. 171, no. 7, pp. 1531–1555, 2014.

[30] S. Liu, X. Hu, and T. Liu, "Ant colony optimisation inversion of surface and borehole magnetic data under lithological constraints," *Journal of Applied Geophysics*, vol. 112, pp. 115–128, 2015.

[31] Z. Hu, Y. Shi, and X. Liu, "Two-dimensional magnetotelluric parallel-constrained-inversion using artificial-fish-swarm algorithm," *Magnetochemistry*, vol. 9, no. 2, p. 34, 2023.

[32] L. M. Xiao, "An optimizing method based on autonomous animats: fish-swarm algorithm," *Systems Engineering Theory & Practice*, vol. 18, 2002.

[33] M. K. Sen and P. L. Stoffa, "Nonlinear one-dimensional seismic waveform inversion using simulated annealing," *Geophysics*, vol. 56, no. 10, pp. 1624–1638, 1991.

[34] C. Yin and G. Hodges, "Simulated annealing for airborne EM inversion," *Geophysics*, vol. 72, no. 4, pp. F189–F195, 2007.

[35] L. Yang and Z. D. Li, "Inverse analysis of rock creep model parameters based on improved simulated annealing differential evolution algorithm," *Geotechnical & Geological Engineering*, vol. 37, no. 2, pp. 639–649, 2019.

[36] H. Z. Li, S. Guo, and C. J. Li, "A hybrid annual power load forecasting model based on generalized regression neural network with fruit fly optimization algorithm," *Knowledge-Based Systems*, vol. 37, pp. 378–387, 2013.

[37] W.-T. Pan, "Using modified fruit fly optimisation algorithm to perform the function test and case studies," *Connection Science*, vol. 25, no. 2-3, pp. 151–160, 2013.

[38] L. Wang, Y. Shi, and S. Liu, "An improved fruit fly optimization algorithm and its application to joint replenishment problems," *Expert Systems with Applications*, vol. 42, no. 9, pp. 4310–4323, 2015.

[39] T. Meng and Q.-K. Pan, "An improved fruit fly optimization algorithm for solving the multidimensional knapsack problem," *Applied Soft Computing*, vol. 50, pp. 79–93, 2017.

[40] J. Cheng and T. Shi, "Structural optimization of transmission line tower based on improved fruit fly optimization algorithm," *Computers & Electrical Engineering*, vol. 103, Article ID 108320, 2022.

[41] G. Hu, Z. Xu, and G. Wang, "Forecasting energy consumption of long-distance oil products pipeline based on improved fruit fly optimization algorithm and support vector regression," *Energy*, vol. 224, Article ID 120153, 2021.

[42] Q. K. Pan, H. Y. Sang, and J. H. Duan, "An improved fruit fly optimization algorithm for continuous function optimization problems," *Knowledge-Based Systems*, vol. 62, pp. 69–83, 2014.

[43] X. Yuan, X. Dai, and J. Zhao, "On a novel multi-swarm fruit fly optimization algorithm and its application," *Applied Mathematics and Computation*, vol. 233, pp. 260–271, 2014.

[44] M. Mitić, N. Vuković, and M. Petrović, "Chaotic fruit fly optimization algorithm," *Knowledge-Based Systems*, vol. 89, pp. 446–458, 2015.

[45] L. Wu, C. Zuo, and H. Zhang, "A cloud model based fruit fly optimization algorithm," *Knowledge-Based Systems*, vol. 89, pp. 603–617, 2015.

[46] S.-X. Lv, Y.-R. Zeng, and L. Wang, "An effective fruit fly optimization algorithm with hybrid information exchange and its applications," *International Journal of Machine Learning and Cybernetics*, vol. 9, no. 10, pp. 1623–1648, 2017.

[47] T.-S. Du, X.-T. Ke, J.-G. Liao, Y. J. Shen, and Dslc-Foa, "DSLC-FOA:improved fruit fly optimization algorithm for application to structural engineering design optimization

problems," *Applied Mathematical Modelling*, vol. 55, pp. 314–339, 2018.

[48] X. Han, Q. Liu, and H. Wang, "Novel fruit fly optimization algorithm with trend search and co-evolution," *Knowledge-Based Systems*, vol. 141, pp. 1–17, 2018.

[49] R. Storn and K. Price, "Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces," *ICSI Berkeley*, vol. 21, 1995.

[50] G. Wu, X. Shen, and H. Li, "Ensemble of differential evolution variants," *Information Sciences*, vol. 423, pp. 172–186, 2018.

[51] L. Cui, G. Li, and Z. Zhu, "Adaptive multiple-elites-guided composite differential evolution algorithm with a shift mechanism," *Information Sciences*, vol. 422, pp. 122–143, 2018.

[52] Y. L. Ekinci, Ç. Balkaya, and G. Göktürkler, "Model parameter estimations from residual gravity anomalies due to simple-shaped sources using Differential Evolution Algorithm," *Journal of Applied Geophysics*, vol. 129, pp. 133–147, 2016.

[53] 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.

[54] S. C. Constable, R. L. Parker, and C. G. Constable, "Occam's inversion: a practical algorithm for generating smooth models from electromagnetic sounding data," *Geophysics*, vol. 52, no. 3, pp. 289–300, 1987.

[55] A. Kelbert, "Taking magnetotelluric data out of the drawer," *AGU Fall Meeting Abstracts*, vol. 565, pp. IN21A-01, 2019.