

Research Article

A Modified Whale Optimization Algorithm and Its Application in Seismic Inversion Problem

Xiaodan Liang ¹, Siwen Xu ¹, Yang Liu ², and Liling Sun ¹

¹Tiangong University, Tianjin 300387, China

²Liaoning Inspection Examination and Certification Centre, Shenyang 110000, China

Correspondence should be addressed to Liling Sun; sunliling2019@outlook.com

Received 13 October 2021; Accepted 30 December 2021; Published 24 January 2022

Academic Editor: Jianhui Lv

Copyright © 2022 Xiaodan Liang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The whale optimization algorithm (WOA) is a popular swarm intelligence algorithm which simulates the hunting behavior of humpback whales. WOA has the deficiency of easily falling into the local optimal solutions. In order to overcome the weakness of the WOA, a modified variant of WOA called OCDWOA is proposed. There are four main operators introduced into the OCDWOA to enhance the search performance of WOA. The operators include opposition-based learning method, nonlinear parameter design, density peak clustering strategy, and differential evolution. The proposed algorithm is tested on 19 optimization benchmark functions and a seismic inversion problem. OCDWOA is compared with the classical WOA and three typical variants of WOA. The results demonstrate that OCDWOA outperforms the compared algorithms in terms of obtaining the global optimal solution.

1. Introduction

In recent years, the nature-inspired algorithms have attracted lots of attention from researchers. Swarm intelligence (SI) algorithm is a main kind of nature-inspired algorithms. Swarm intelligence algorithms are designed by simulating the behaviors of animals or plants. The widely used SI algorithms include particle swarm optimization (PSO) [1], Ant colony optimization (ACO) [2], artificial bee colony algorithm (ABC) [3], grey wolf optimization (GWO) [4], wolf pack algorithm (WPA) [5], brain storm optimization [6], and whale optimization algorithm (WOA) [7].

The whale optimization algorithm (WOA) is proposed in 2016. WOA searches the global optimal solution by simulating the foraging behaviors of the whales, including encircling prey, researching prey, and attacking prey. Many studies about WOA have been presented. Generally, these studies can be classified into two categories. One is to apply applying the WOA to solve some practical issues. The other one is to improve the performance of WOA with some

strategies. A modified WOA named chaotic WOA (CWOA) [8] was proposed by Kaur and Arora in 2017. The CWOA introduces chaos theory into WOA. The main parameter of WOA is tuned by chaotic map to control exploration and exploitation. In [9], Lévy flight trajectory-based WOA (LWOA) is presented. The Lévy flight trajectory is used to increase the diversity of the population and enhance the performance of jumping out of local optimal solutions. The balanced variant of WOA (BWOA) proposed in [10] introduced Lévy flight and chaotic local search into WOA to enhance the inclusive exploratory and neighborhood-informed capacities of the conventional technique. Chen [11] proposed a modified WOA based on adaptive convergence and Lévy features (IWOA). The adaptive convergence operator is used to balance the local and global optimization ability and the Lévy flight mechanism is introduced to improve global searching ability. For solving the large-scale global optimization problems, a modified self-adaptive WOA (MWOA) [12] is proposed. A nonlinear dynamic strategy is applied to update the control parameter and

balance the exploration and exploitation abilities. The Lévy-flight strategy is utilized to enhance the search performance and a quadratic interpolation method is adopted to improve the exploitation ability of the algorithm. For solving high-dimensional function optimization problems, A WOA based on Lamarckian learning (WOALam) [13] brought in the evolutionary theory of Lamarck to enhance the local search. Trivedi et al. [14] proposed an adaptive whale optimization algorithm (AWOA) with a randomization technique. The results show that the performance of AWOA is better than the performance of the standard WOA. Sun et al. proposed a whale optimization algorithm based on quadratic interpolation (QIWOA) [15]. A modified exploration process and quadratic interpolation are used to improve the exploitation ability and the search accuracy. A reinforced variant of WOA named RDWOA [16] was proposed in 2019. A strategy of random spare or random replacement is used to enhance the convergence speed. In addition, a strategy of double adaptive weight is introduced to improve the exploration ability in the early stage and exploitative ability in the later stage. Trivedi et al. [17] proposed a hybrid PSO-WOA. The PSO operator is used in exploitation phase and the basic WOA operator is adopted to explore in uncertain environments. The experimental results reveal the effectiveness of PSO-WOA compared to standard PSO and WOA. Wei et al. [18] proposed a modified WOA based on different searching paths and perceptual disturbance (PDWOA). Several spiral curves have been tested and the results show that the equal-pitch of Archimedean spiral curve is more capable than other types of spiral curve. Mafarja and Mirjalili [19] proposed a hybrid SA-WOA algorithm. In order to enhance the exploitation of WOA, the simulated annealing (SA) algorithm is introduced into the algorithm by searching the most promising regions located by basic WOA. An improved whale optimization algorithm (IWOA) [20] is proposed in 2016. Inertia weight, as a new parameter, is introduced to adjust the best solution in each iteration. Abdel-Basset et al. [21] proposed a memetic algorithm combining the WOA with a Tabu Search for solving quadratic assignment problems.

In addition, WOA has been applied to many engineering problems. Oliva et al. [22] used an improved chaotic whale optimization algorithm to estimate the parameters of solar cells. The design problem involves the solution of the complex nonlinear and multimodal objective functions. The proposed algorithm solves the design problem with the chaotic map and automatically adapt parameters. Zhang et al. [23] proposed an enhanced whale optimization algorithm (EWOA) to optimize the flow shop construction scheme. The enhanced WOA with Lévy flight is used to improve the robustness and efficiency of the existing construction stage and zone optimization approaches. Bentouati et al. [24] proposed a new variant WOA with pattern search algorithm (PS). The results prove the effectiveness and the performance of the proposed algorithm for solving the optimal power flow problem. Abd Elaziz and Oliva [25] proposed an enhanced WOA based on opposition (OBWOA). The opposition-based learning is used to enhance the exploration of the search space.

OBWOA has the exploration abilities for different benchmark optimization functions and has been applied to estimate three different diode models of solar cells. Aljarah et al. [26] applied WOA to neural networks training. The proposed algorithm has been tested on 20 datasets. The results showed that the proposed method can obtain the best solutions and outperform the other compared algorithms on convergence speed. Medani et al. [27] proposed a modified WOA to solve the optimal reactive power dispatch problem. The results show that the proposed algorithm is more desirable than two comparative algorithms.

Prestack seismic inversion technology is an important seismic inversion technology to solve the problems of reservoir lithology and fluid prediction in oil and gas exploration and development. Physical properties obtained by general prestack inversion methods provide many detailed subsurface characteristics, but the method is not mature and limited by the high resolution, low speed, and poor stability of prestack inversion. Currently, this technology is in the research stage. Some inevitable technology problems should be solved before its large-scale applications [28, 29].

Many researches on solving the problems of prestack inversion by optimization algorithms have been studied. Xia et al. [30] and Sen [31] improved the speed and stability of prestack seismic waveform inversion by using the step inversion method. Simulated annealing (SA) and genetic algorithm (GA) have been successfully used in prestack seismic waveform inversion. Ingber and Rosen [32] conducted a comparative study on SA and GA. The experiment results proved that both algorithms could achieve global optimization and the speed of fast simulated annealing (VFSA) was the fastest. Liu et al. [33] proposed a new adaptive genetic algorithm for seismic inversion to overcome slow convergence rate and falling into local optimal solutions. Wu et al. [34] proposed a parameter inversion method based on improved differential evolution algorithm. The proposed method was effective in solving parameter inversion problems of prestack seismic data.

WOA has a good convergence rate, but the performance of WOA in finding the global optimal solution of the multimodal problems with many local optimal solutions is not ideal. In order to enhance the global search ability and solve the prestack seismic waveform inversion, a modified whale optimization algorithm named OCDWOA is proposed in this paper. There are four operators used in the proposed algorithm. Nonlinear parameter design is introduced to balance the ratio of exploitation phase and exploration phase. Density peak clustering strategy is adopted to decompose population into numbers of groups for enhancing the diversity of population. Opposition-based learning method and differential evolution are used to improve the search performance. Experiments on 19 optimization benchmark functions and a seismic inversion problem indicate the effectiveness of the proposed algorithm. The proposed algorithm OCDWOA represents outstandingly the experiments. The results demonstrate that the proposed algorithm OCDWOA can solve the seismic inversion problem successfully.

The main contributions are summarized as follows:

- (1) The opposition-based learning method adopted by OCDWOA could avoid the solutions trapping into local optimal, which improves the ability of global search.
- (2) The nonlinear parameter design of a in OCDWOA could achieve the global exploration in early stage and local exploitation in late stage.
- (3) Density peak clustering strategy is adopted to decompose population into numbers of groups, which enhances the diversity of population.
- (4) The proposed OCDWOA has an outstanding performance of solving seismic inversion problem, which verifies its universality for solving practical engineering problems.

The rest of this paper is organized as follows. Section 2 describes the WOA algorithm. Section 3 explains the proposed algorithm and the operators. The experimental results and comparisons are shown in Section 4. Finally, in Section 5, the conclusions are presented.

2. Whale Optimization Algorithm

WOA is a kind of population-based meta-heuristic algorithm. WOA simulates the hunting mode of humpback whales, which includes encircling prey, bubble-net attacking method, and search for prey.

2.1. Encircling Prey. WOA supposes that the optimal solution with the best fitness in current population is the prey. All the other solutions will encircle the prey and update themselves toward the optimal solution. The mathematical model of encircling behavior is as follows:

$$\vec{D} = \left| \vec{C} \cdot \vec{X}^*(t) - \vec{X}(t) \right|, \quad (1)$$

$$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A} \cdot \vec{D}, \quad (2)$$

where t indicates the number of the current iteration, X is the current solution, X^* is the best solution in the population, and A and C are the coefficient vectors calculated as follows:

$$\begin{aligned} \vec{A} &= 2\vec{a} \cdot \vec{r} - \vec{a}, \\ \vec{C} &= 2\vec{r}, \end{aligned} \quad (3)$$

where \vec{a} is linearly decreased from 2 to 0 over the iterations and \vec{r} is a random vector in $[0, 1]$.

2.2. Bubble-Net Attacking Method. In order to simulate the behavior of bubble-net attacking, shrinking encircling mechanism and spiral updating position are designed as follows.

2.2.1. Shrinking Encircling Mechanism. The mathematical model of shrinking encircling mechanism is defined as equations (1) and (2). Because of \vec{a} is linearly decreased from

2 to 0, the value of \vec{A} is in the range of $[-1, 1]$. The current solution can update itself toward the best solution with a random step.

2.2.2. Spiral Updating Position. The new position of current solution updated with the best solution is obtained as follows:

$$\vec{D}' = \left| \vec{X}^*(t) - \vec{X}(t) \right|, \quad (4)$$

$$\vec{X}(t+1) = \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(t), \quad (5)$$

where b is constant for defining logarithmic spiral and l is randomly generated number in the range $[-1, 1]$. To balance the two models, a selection parameter p is set. The selection parameter p in WOA is 0.5; there is a probability of 50% to choose either the shrinking encircling mechanism or the spiral updating position model. The final mathematical model of bubble-net attacking is as follows:

$$\vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - \vec{A} \cdot \vec{D}, & \text{if } p < 0.5, \\ \vec{D}' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(t), & \text{if } p \geq 0.5, \end{cases} \quad (6)$$

where p is a random number in $[0, 1]$.

2.3. Search for Prey. In nature, whales will go away from each other and randomly search for a new prey. In order to simulate the search behavior, a random search operation is used in the exploration phase when $|A| \geq 1$. The current solution will update with a random solution in the current population instead of the best solution. The mathematical model is as follows:

$$\vec{D} = \left| \vec{C} \cdot \vec{X}_{\text{rand}}(t) - \vec{X}(t) \right|, \quad (7)$$

$$\vec{X}(t+1) = \vec{X}_{\text{rand}}(t) - \vec{A} \cdot \vec{D},$$

where \vec{X}_{rand} is a random solution in the current population.

The pseudocode of the WOA is shown as Algorithm 1.

3. The Proposed Approach

This section introduces a modified WOA named OCDWOA. Four strategies are introduced into the WOA. Opposition-based learning method (OBL) is used to initialize population and randomly update solutions in each iteration. Nonlinear parameter design is introduced to balance the ratio of exploitation phase and exploration phase. Density peak clustering strategy is adopted to decompose population into numbers of groups for enhancing the diversity of population. The effect of differential evolution (DE) is to exchange information between groups and improve the ability of jumping out of the local optimal solutions.

3.1. OCDWOA. The pseudocode of the OCDWOA algorithm is shown as Algorithm 2.

```

Input: Population size  $N$ 
The halting criterion
Output: The best solution  $X^*$ 
Generate the initial population  $X_i$  ( $i = 1, 2, \dots, N$ )
(1) Calculate the fitness for each solution in the population
(2) Find the best solution  $X^*$  with the best fitness
(3) while the halting criterion is not satisfied do
(4)   Update  $a$ 
(5)   for  $i = 1$  to  $N$  do
(6)     Update  $A$ ,  $C$ ,  $l$  and  $p$ 
(7)     for  $j = 1$  to  $m$  do
(8)       if  $p < 0.5$  then
(9)         if ( $|A| < 1$ ) then
(10)           $D = |C \cdot X^*(j) - X_i(j)|$ 
(11)           $X_i(j) = X^*(j) - A \cdot D$ 
(12)         else if ( $|A| \geq 1$ ) then
(13)          Select a random solution  $X_{\text{rand}}$ 
(14)           $D = |C \cdot X_{\text{rand}}(j) - X_i(j)|$ 
(15)           $X_i(j) = X_{\text{rand}}(j) - A \cdot D$ 
(16)         end if
(17)       if  $p \geq 0.5$  then
(18)           $D' = |X^*(j) - X_i(j)|$ 
(19)           $X_i(j) = D' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(j)$ 
(20)       end if
(21)     end for
(22)   Evaluate the fitness for  $X_i$ 
(23) end for
(24) Find the best solution  $X^*$ 
(25) end while

```

ALGORITHM 1: WOA.

Opposition-based learning is used to initialize the population (see lines 1–6 of Algorithm 2) and to estimate the opposition solutions (see lines 39–46 of Algorithm 2). The fitness of each solution in the population is calculated and the best solution is found (see lines 7–8 in Algorithm 2). The parameters a , CR, A , C , l , and p are updated (see lines 10 and 13 in Algorithm 2). The population is decomposed into k groups by density peak clustering algorithm (see line 11 of Algorithm 2). The mutation operator of DE is introduced into the proposed algorithm (see lines 20–23 of Algorithm 2). A solution X_i is updated by r_1 and r_2 under mutation operator, where r_1 is from the group including X_i and r_2 is from the groups excluding X_i (see lines 14–15 of Algorithm 2). With the increase of iterations, the operation probability of DE declines. The basic three operators of WOA are used in lines 25–26, 29–31, and 34–35, respectively. The encircling prey is used in lines 25–26, the search for prey is introduced in lines 29–31, and the spiral updating position is applied in lines 34–35.

3.2. Opposition-Based Learning Method. It has been tested that opposition-based learning (OBL) method proposed by Rahnamayan et al. [35] is helpful for avoiding trapping into local optimal solutions.

A solution X_i in d -dimension search space is denoted as $X_i^* = \{X_{i,1}, X_{i,2}, X_{i,3}, \dots, X_{i,d}\}$. The opposition point X_i^* of X_i can be calculated as

$$X'_{i,j} = (X_{ub,j} + X_{lb,j}) - X_{i,j}, \quad (8)$$

where $X_{ub,j}$ and $X_{lb,j}$ denote the upper and lower bound of j -th dimension.

OBL is used in the initialization step and solution update step at the end of each iteration. In the initialization step, OBL can generate a new solution on opposition side and enhance the diversity in the search space. In solution update step at the end of each iteration, OBL can generate a new solution on opposition side and choose the better solution between the new solution and the original solution. OBL can help WOA improve the search performance and jump out the local optimal solutions.

3.3. Nonlinear Parameter Design. An exponential function \vec{a} is introduced into the modified WOA. The value of coefficient vector \vec{A} controls the proportion of exploitation phase and exploration phase. The nonlinear parameter design \vec{a} is designed as follows [36]:

$$a = a_{\min} + (a_{\max} - a_{\min}) \times e^{-2t/\text{Max_iter}}, \quad (9)$$

where a_{\min} and a_{\max} denote the minimum and maximum of \vec{a} , t is the number of the current iteration, and Max_iter is the maximum iteration.

A large value of \vec{a} generates a large step size in exploitation phase, which can improve the searching speed. A

```

Input: Population size  $N$ 
          Group size  $k$ 
          The halting criterion
Output: The best solution  $X^*$ 
(1) Generate the initial population  $X_i$  ( $i = 1, 2, \dots, N$ )
(2) for  $i = 1$  to  $N$  do
(3)   if ( $\text{rand} < 0.6$ )
(4)     Generate new solution with OBL
(5)   end if
(6) end for
(7) Calculate the fitness for each solution in the population
(8) Find the best solution  $X^*$  with the best fitness
(9) while the halting criterion is not satisfied do
(10)  Update  $a$  and CR
(11)  Decompose population into  $k$  groups by density peak clustering algorithm
(12)  for  $i = 1$  to  $N$  do
(13)    Update  $A$ ,  $C$ ,  $l$  and  $p$ 
(14)     $R_1$  is the group with  $X_i$  and  $R_2$  are the groups without  $X_i$ 
(15)    Select randomly  $r_1$  from  $R_1$  and  $r_2$  from  $R_2$ 
(16)     $j_{\text{rand}} = \text{randint}(1, m)$ 
(17)    for  $j = 1$  to  $m$  do
(18)      if  $p < 0.5$  then
(19)        if ( $|A| < 1$ ) then
(20)          if  $\text{rand}(0, 1) \leq \text{CR}$  and  $j == j_{\text{rand}}$  then
(21)            Update  $F$ 
(22)            Select randomly  $r_1$  from  $R_1$  and  $r_2$  from  $R_2$ 
(23)             $U_i(j) = X^*(j) + F[X_{r_2}(j) - X_{r_1}(j)]$ 
(24)          Else
(25)             $D = |C \cdot X^*(j) - X_i(j)|$ 
(26)             $X_i(j) = X^*(j) - A \cdot D$ 
(27)          end if
(28)        elseif ( $|A| \geq 1$ ) then
(29)          Select a random solution  $X_{\text{rand}}$ 
(30)           $D = |C \cdot X_{\text{rand}}(j) - X_i(j)|$ 
(31)           $X_i(j) = X_{\text{rand}}(j) - A \cdot D$ 
(32)        end if
(33)      if  $p \geq 0.5$  then
(34)         $D' = |X^*(j) - X_i(j)|$ 
(35)         $X_i(j) = D' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(j)$ 
(36)      end if
(37)    end for
(38)  end for
(39)  for  $i = 1$  to  $N$  do
(40)    if ( $\text{rand} < 0.6$ )
(41)      Generate new solution  $O_i$  with OBL
(42)      if  $O_i$  is better than  $X_i$  then
(43)         $X_i = O_i$ 
(44)      end if
(45)    end if
(46)  end for
(47)  Find the best solution  $X^*$  with the best fitness
(48) end while

```

ALGORITHM 2: OCDWOA.

small step size with a small value of \vec{a} can be helpful for the exploration phase.

3.4. Density Peak Clustering. A density-based clustering strategy named density peak clustering algorithm [37] is used in the proposed WOA. The density peak clustering algorithm

defines two important parameters for each solution i : the density ρ_i and distance δ_i . A scatter graph called decision diagram is constructed based on these two parameters. The decision diagram could decide the center of each clustering group and assign the remaining solutions to each clustering center.

The mathematical model of the density ρ_i of the i th solution used in this paper is as follows:

$$\rho_i = \sum_j e^{-(d_{ij}/d_c)^2}, \quad (10)$$

where j represents the j -th solution in current population, and $i \neq j$, d_{ij} is the Euclidean distance between i -th and j -th solutions, and d_c is a pre-set cut-off distance. A moderate value of d_c could make the average number of neighbors around 1% to 2% of the population size.

The distance parameter δ_i denotes the minimum distance among the i -th solution and the j -th solution, where the local density of j -th solution is higher than that of the i -th solution. The equation of δ_i is as follows:

$$\delta_i = \min_{j: \rho_j > \rho_i} (d_{ij}). \quad (11)$$

Figure 1(a) shows the distribution of 28 solutions in two dimensions and Figure 1(b) is the decision diagram obtained by these 28 solutions. In Figure 1(b), the horizontal axis is ρ and the vertical axis is δ . The solutions with both large values of ρ and δ are defined as the clustering centers. In Figure 2, the 1st and 10th solutions with both large density and distance will be selected as the clustering centers. The 26th, 27th, and 28th solutions have large distance and low density, and the 7th and 8th solutions have large density and low distance. Therefore, these solutions cannot be selected as the clustering centers.

According to the rules of selecting the clustering centers, a modified equation of calculating quality of solutions is designed as follows:

$$Q_i = \rho_i \cdot \delta_i. \quad (12)$$

In this paper, a modified model of cut-off distance d_c is introduced [38]. The transition distance Dis is calculated as follows:

$$\text{Dis} = 1 - e^{-d_{ij}^2/2}. \quad (13)$$

The cut-off distance d_c is calculated as follows:

$$d_c = \frac{\text{Dis}_{\max} + \text{Dis}_{\min}}{2}, \quad (14)$$

where Dis_{\max} and Dis_{\min} are the maximum and minimum of transition distances in the current iteration.

3.5. Differential Evolution. For enhancing abilities of searching and jumping out of the local optimal solutions, the DE operator is introduced in this paper. The crossover and mutation operators in differential evolution (DE) [39] are calculated as follows:

$$U_i(j) = X^*(j) + F \times [X_{r_2}(j) - X_{r_1}(j)], \quad (15)$$

where X_{r_1} and X_{r_2} are the j -th dimension of the r_1 -th and r_2 -th solutions, respectively, U is the offspring, and F is the zoom factor. CR is the crossover probability updated as an exponential function:

$$\text{CR} = 0.9 \times e^{-3t/\text{Max_iter}}. \quad (16)$$

4. Experiment Results and Discussion

4.1. Parameter Settings and Benchmark Functions. In this section, in order to analyze the performance of the proposed algorithm, 19 benchmark functions [40] F_1 – F_{19} have been used. The expressions of all the functions are given in Table 1. F_1 – F_7 are the unimodal functions with only one global optimal solution. F_8 – F_{12} are the multimodal functions with many local optimal solutions. These functions can test the performance of jumping out the local optimal solutions. F_{13} – F_{19} are the fixed-dimension multimodal functions, which can evaluate the convergence property of an algorithm.

The proposed OCDWOA is compared with the classical WOA and its typical variants, such as OWOA [41], OEWOA [42], and IWOA [43]. The parameter b is set to 1 for all the algorithms. The parameter \bar{a} decreases linearly from 2 to 0 for WOA, OWOA, and IWOA. For OEWOA, the parameter \bar{a} nonlinearly decreases from 2 to 0. For IWOA, the crossover probability CR is set to 0.9, and the zoom factor F is random within (0.2, 0.8).

For all the algorithms, population size is 30 and maximum iteration is 500. The experiment is repeated 30 times for each benchmark function.

4.2. Experiment Results. The statistics of the experimental results is demonstrated in Table 2. The best fitness, mean fitness, and standard deviation of OCDWOA and each compared algorithm are given.

For $F_6, F_9, F_{11}, F_{12}, F_{13}, F_{15}, F_{17}$, and F_{18} , the performance of OCDWOA is the absolutely best. For F_{14} , the results obtained by all the variants of WOA are comparable. For $F_1, F_2, F_3, F_4, F_5, F_{10}$, and F_{16} , OCDWOA obtains the second rank. For the simple functions F_1 and F_2 without any local optimal solution, OCDWOA is worse than OEWOA. The DE operator used in OCDWOA can enhance the search performance. The convergence speed would be reduced, though the operation probability of DE declines in exploration phase. For the nonseparable unimodal functions F_3, F_4 , and F_5 , OCDWOA is only worse than IWOA. For the multimodal function F_{16} , the result reveals the positive influence of DE.

For F_7 and F_8 , the performance of OCDWOA is similar to ones of OWOA and OEWOA. For F_{10} , OCDWOA gets the second rank. For F_8 and F_{10} , IWOA can get the optimal solution. However, IWOA fails in solving the function F_7 . The mean value and standard deviation value of IWOA are also not ideal. It shows that the DE operator is helpful for jumping out the local optimal solutions, but the stability may be impaired for the multimodal functions with easy or few local optimal solutions. In OCDWOA, the clustering operator and reducing the operation probability of DE can drop off the influence of DE. The DE operator can improve the performance of jump out the local optimal solutions. The DE operator and OBL can enhance the search performance of algorithm. Dynamical adjustment of \bar{a} can avoid premature convergence. The clustering operator enhances information exchange between solutions.

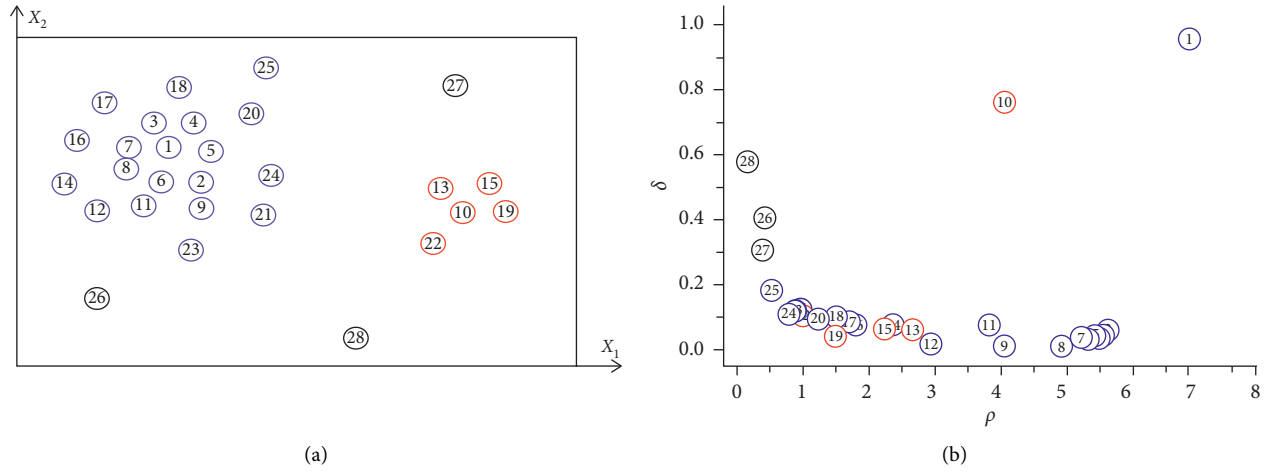


FIGURE 1: (a) Solution distribution in two dimensions. (b) Decision diagram for each sample point.

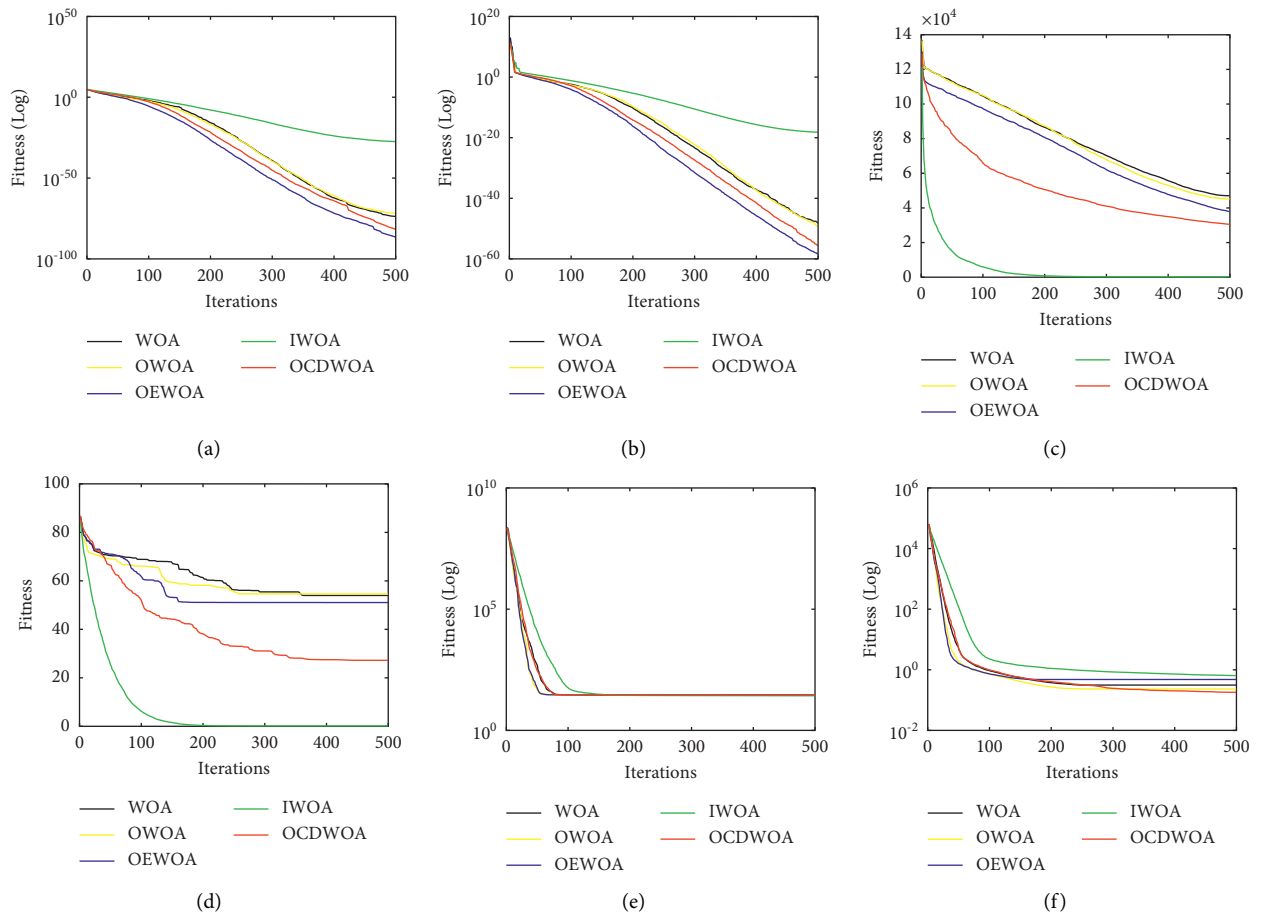


FIGURE 2: Continued.

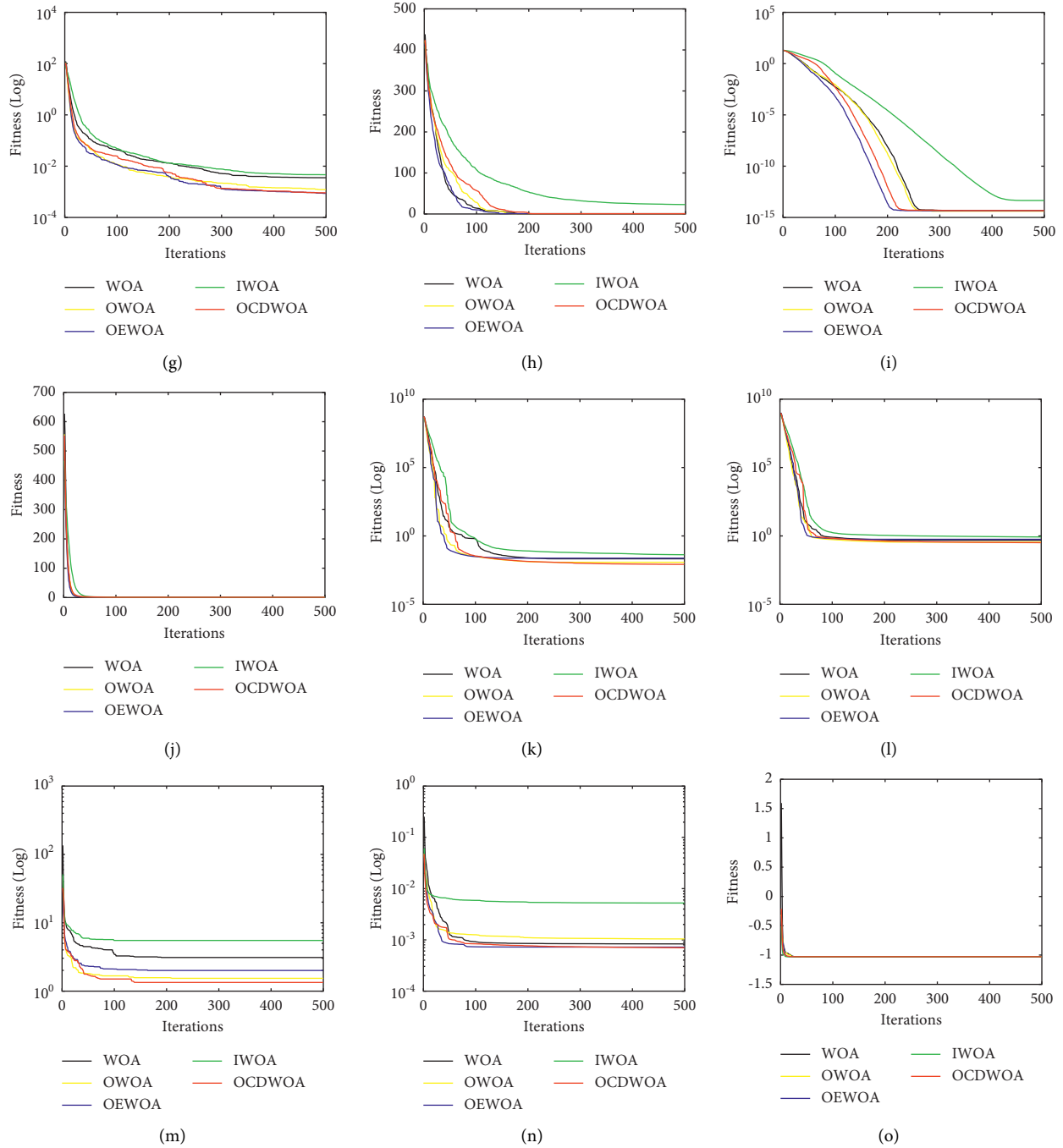


FIGURE 2: Continued.

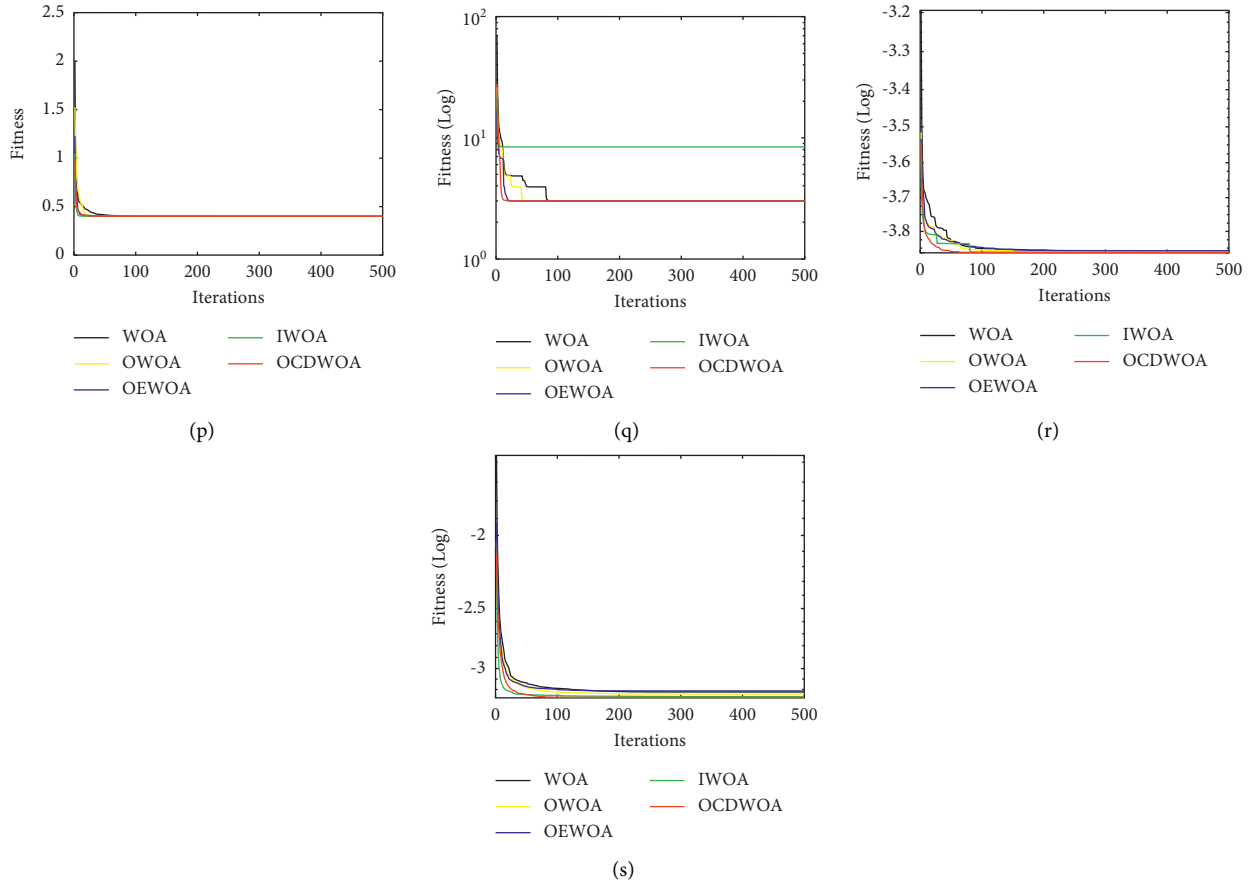


FIGURE 2: Convergence graphs of OCDWOA with respect to other algorithms. (a) F_1 . (b) F_2 . (c) F_3 . (d) F_4 . (e) F_5 . (f) F_6 . (g) F_7 . (h) F_8 . (i) F_9 . (j) F_{10} . (k) F_{11} . (l) F_{12} . (m) F_{13} . (n) F_{14} . (o) F_{15} . (p) F_{16} . (q) F_{17} . (r) F_{18} . (s) F_{19} .

The results analyzed by the Wilcoxon rank-sum test are shown in Table 3. In the Wilcoxon rank-sum test, a significance level is set to 0.05. In Table 3, “+,” “-,” and “=” indicate that the result of the proposed algorithm OCDWOA is significantly better, significantly worse, and statistically similar to that of the compared algorithm, respectively. For most test functions, the performance of OCDWOA is significantly better or statistically similar to the other variants of WOA.

The convergence curves and boxplots are shown in Figures 2 and 3, respectively.

In Figure 2, for most functions, the convergence curves of OCDWOA are better or statistically similar to ones of the other variants of WOA. For F_3 and F_4 , the convergence curves of OCDWOA are stabler than those of IWOA, but the convergence rates of OCDWOA are slower than those of IWOA. In OCDWOA, the solutions selected in the crossover and mutation operators of DE are based on the clustering. The clustering operator reduces the randomness of the DE operator. The stability of OCDWOA has been improved.

In Figure 3, for most functions, the stability of OCDWOA is better or statistically similar to the stability of the other variants of WOA. Only for F_3 and F_4 , the stability of OCDWOA is worse than the stability of IWOA.

4.3. Application of Seismic Inversion Problems

4.3.1. Seismic Inversion Problem. Seismic inversion is the key technology of quantitative reservoir prediction and prestack waveform inversion [44] is a newly developed seismic inversion technique. In prestack waveform inversion technique, the complex wave propagation effect is considered and the complete waveform information is fully used. The prestack waveform inversion method assumes that each CMP/CDP point of underground medium layer structure with local one-dimensional or level uses the reflectivity method forward prestack data [45]. The prestack waveform inversion method searches the best fitting with the actual prestack seismic data of underground through genetic algorithm, simulated annealing, and other global optimization algorithms [46]. At present, there are many problems in prestack waveform inversion technology that need to be further studied and the resolution of prestack waveform inversion technology needs to be improved. The existing optimization algorithms used in prestack waveform inversion are mainly traditional intelligent optimization algorithms. These algorithms are prone to premature convergence and local optimal solution when dealing with large data volume, highly nonlinear, multiparameter, and multiextremum prestack waveform inversion problems.

TABLE 1: Description of benchmark functions.

Function	Dim	Range	f_{\min}
Unimodal functions			
$F_1 = \sum_{i=1}^n x_i^2$	30	[-100, 100]	0
$F_2 = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	30	[-10, 10]	0
$F_3 = \sum_{i=1}^n (\sum_{j=1}^i x_j^2)$	30	[-100, 100]	0
$F_4 = \max_i \{x_i\}, 1 \leq i \leq n$	30	[-100, 100]	0
$F_5 = \sum_{i=1}^n [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	[-30, 30]	0
$F_6 = \sum_{i=1}^n (x_i + 0.5)^2$	30	[-100, 100]	0
$F_7 = \sum_{i=1}^n ix_i^4 + \text{random}[0, 1]$	30	[-1.28, 1.28]	0
Multimodal functions			
$F_8 = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	[-5.12, 5.12]	0
$F_9 = -20 \exp(-0.2\sqrt{1/n} \sum_{i=1}^n x_i^2) - \exp[1/n \sum_{i=1}^n \cos(2\pi x_i)] + 20 + e$	30	[-32, 32]	0
$F_{10} = 1/4000 \sum_{i=1}^n x_i^2 + \prod_{i=1}^n \cos(x_i/\sqrt{i}) + 1$	30	[-600, 600]	0
$F_{11} = \pi/n \{10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1}) + (y_n - 1)^2]\} + \sum_{i=1}^n u(x_i, 10, 100, 4)$	30	[-50, 50]	0
$y_i = 1 + x_i + 1/4u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & a < x_i \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$			
$F_{12} = 0.1 \{ \sin^2(3\pi x_1) + \sum_{i=1}^{n-1} (x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1}) + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)]] \} + \sum_{i=1}^n u(x_i, 5, 100, 4)$	30	[-50, 50]	0
Fixed-dimension multimodal functions			
$F_{13} = (1/500 + \sum_{j=1}^{25} 1/j + \sum_{i=1}^2 (x_i - a_{ij})^6)^{-1}$	2	[-65, 65]	0
$F_{14} = \sum_{i=1}^{11} [a_i + x_i^2 (b_i^2 + b_i x_2)] b_i^2 + b_i x_3 + x_4]^2$	4	[-5, 5]	0.00030
$F_{15} = 4x_1^2 - 2.1x_1^4 + 1/3x_1^6 + x_1 x_2 - 4x_2^2 + 4x_4^2$	2	[-5, 5]	-1.0316
$F_{16} = (x_2 - 5.1/4\pi^2 x_1^2 + 5/\pi x_1 - 6)^2 + 10(1 - 1/8\pi) \cos x_1 + 10$	2	[-5, 5]	0.398
$F_{17} = [1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_2^2) - 14x_2 + 6x_1 x_2 + 3x_2^2] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1 x_2 + 27x_2^2)]$	2	[-2, 2]	3
$F_{18} = -\sum_{i=1}^4 c_i \exp[-\sum_{j=1}^3 a_{ij} (x_j - p_{ij})^2]$	3	[1, 3]	-3.86
$F_{19} = -\sum_{i=1}^4 c_i \exp[-\sum_{j=1}^6 a_{ij} (x_j - p_{ij})^2]$	6	[0, 3]	-3.12

TABLE 2: Results of OCDWOA and compared algorithms.

Function	Algorithm	Best	Mean	Std
F_1	WOA	$3.45E-84$	$1.84E-74$	$6.85E-74$
	OWOA	$3.67E-87$	$1.54E-72$	$6.15E-72$
	OEWOA	$8.10E-104$	$3.41E-87$	$1.61E-86$
	IWOA	$3.88E-31$	$4.18E-28$	$1.52E-27$
	OCDWOA	$1.16E-94$	$2.68E-82$	$8.18E-82$
F_2	WOA	$3.76E-59$	$4.29E-49$	$2.30E-48$
	OWOA	$8.72E-58$	$3.71E-50$	$9.41E-50$
	OEWOA	$8.88E-68$	$2.26E-59$	$1.15E-58$
	IWOA	$6.02E-20$	$6.36E-19$	$6.72E-19$
	OCDWOA	$1.85E-63$	$1.86E-56$	$4.92E-56$
F_3	WOA	29331.39043	46962.31228	11654.1269
	OWOA	23102.63585	45115.1214	9775.63579
	OEWOA	10896.08787	37765.42249	12462.65209
	IWOA	0.0204919	4.761001851	12.20526468
	OCDWOA	9123.146589	30378.88003	11250.945
F_4	WOA	2.845944535	53.98492009	28.64604813
	OWOA	0.221683283	54.59717151	28.7671405
	OEWOA	0.395500666	51.05649771	30.09654391
	IWOA	$2.53E-05$	$1.43E-03$	$1.46E-03$
	OCDWOA	0.091259942	27.15705548	22.72468237
F_5	WOA	27.05096579	27.98206458	0.492345074
	OWOA	27.0462713	27.8628119	0.3574981
	OEWOA	27.47961584	28.17255965	0.377986162
	IWOA	24.72608991	26.21998458	1.236007685
	OCDWOA	26.94330849	27.75236194	0.519791973
F_6	WOA	0.039352423	0.308186276	0.178197178
	OWOA	0.058900257	0.229444501	0.138575416
	OEWOA	0.094203593	0.471932521	0.211010397
	IWOA	0.132553897	0.634850188	0.363709611
	OCDWOA	0.011880743	0.177746121	0.202309313
F_7	WOA	$7.31E-05$	$3.49E-03$	$4.08E-03$
	OWOA	$9.68E-06$	$1.20E-03$	$2.15E-03$
	OEWOA	$1.78E-05$	$8.51E-04$	$9.84E-04$
	IWOA	0.001170766	0.004571304	0.003016255
	OCDWOA	$1.19E-05$	$8.98E-04$	$1.21E-03$
F_8	WOA	0	$5.68E-15$	$2.29E-14$
	OWOA	0	0	0
	OEWOA	0	0	0
	IWOA	0	22.36681071	24.91175078
	OCDWOA	0	0	0
F_9	WOA	$8.88E-16$	$4.32E-15$	$2.55E-15$
	OWOA	$8.88E-16$	$3.85E-15$	$2.48E-15$
	OEWOA	$8.88E-16$	$4.20E-15$	$2.27E-15$
	IWOA	$2.22E-14$	$4.29E-14$	$1.27E-14$
	OCDWOA	$8.88E-16$	$4.44E-15$	$2.09E-15$
F_{10}	WOA	0	0.014239143	0.055902557
	OWOA	0	0.006369305	0.034886118
	OEWOA	0	0	0
	IWOA	0	0.005601519	0.009188179
	OCDWOA	0	0.00307175	0.016824667
F_{11}	WOA	0.003303307	0.020703883	0.013561528
	OWOA	0.002886391	0.011523424	0.008134223
	OEWOA	0.008832644	0.022931786	0.008957016
	IWOA	0.01289871	0.04121751	0.023964971
	OCDWOA	0.000984744	0.008086227	0.00817336

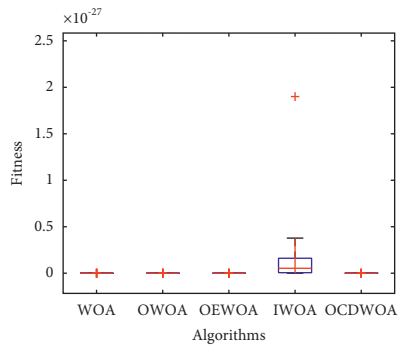
TABLE 2: Continued.

Function	Algorithm	Best	Mean	Std
F_{12}	WOA	0.091759692	0.476972639	0.2731912
	OWOA	0.071293381	0.343978384	0.176667484
	OEWOA	0.264598322	0.558788251	0.219290838
	IWOA	0.360052384	0.826186996	0.250307701
	OCDWOA	0.008972727	0.315016992	0.165937507
F_{13}	WOA	0.998003838	3.087707817	3.643013869
	OWOA	0.998003838	1.527492833	0.770156088
	OEWOA	0.998003838	1.991064676	1.244254244
	IWOA	0.998003838	5.491627348	4.859439425
	OCDWOA	0.998003838	1.328819151	0.7056337
F_{14}	WOA	0.000309354	0.000754701	0.000591433
	OWOA	0.000308046	0.000692785	0.000506369
	OEWOA	0.000308042	0.000599412	0.000433827
	IWOA	0.000307494	0.005763864	0.008959844
	OCDWOA	0.000308015	0.00068318	0.000396378
F_{15}	WOA	-1.031628453	-1.031628453	7.57E - 10
	OWOA	-1.031628453	-1.031628453	1.10E - 09
	OEWOA	-1.031628453	-1.031628453	5.29E - 10
	IWOA	-1.031628453	-1.031628453	2.29E - 10
	OCDWOA	-1.031628453	-1.031628453	3.68E - 11
F_{16}	WOA	0.397887358	0.397894412	1.60E - 05
	OWOA	0.397887358	0.397899721	2.59E - 05
	OEWOA	0.397887358	0.397916992	0.000107977
	IWOA	0.397887358	0.397887585	5.13E - 07
	OCDWOA	0.397887358	0.397888516	2.38E - 06
F_{17}	WOA	3.000000001	3.000043872	7.74E - 05
	OWOA	3.000000002	3.000044635	0.000152146
	OEWOA	3.000000001	3.000123665	0.000226823
	IWOA	3	8.400000042	20.55035867
	OCDWOA	3.000000001	3.000006355	1.60E - 05
F_{18}	WOA	-3.862688519	-3.857262065	0.007143618
	OWOA	-3.862781863	-3.861815423	0.003572799
	OEWOA	-3.862782142	-3.85726697	0.019011496
	IWOA	-3.862782147	-3.862521555	0.001426916
	OCDWOA	-3.862782147	-3.862775916	9.47E - 06
F_{19}	WOA	-3.321580972	-3.223597877	0.10220952
	OWOA	-3.321545137	-3.244143349	0.085289027
	OEWOA	-3.320833167	-3.213034803	0.091087411
	IWOA	-3.32199476	-3.270086944	0.060385879
	OCDWOA	-3.321994702	-3.278324784	0.058345799

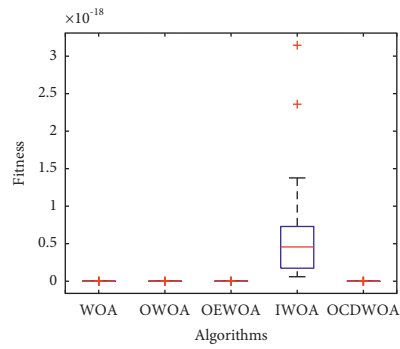
The bold values mean the best results.

TABLE 3: The Wilcoxon rank-sum test results of OCDWOA and compared algorithms.

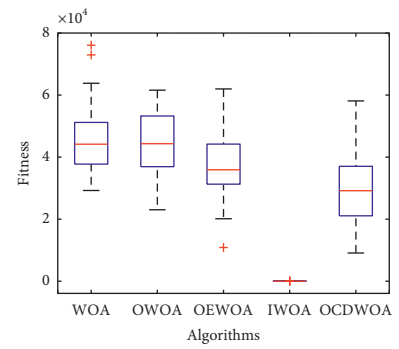
	OCDWOA-WOA	OCDWOA-OWOA	OCDWOA-OEWOA	OCDWOA-IWOA
Comparison	+/-/-	+/-/-	+/-/-	+/-/-
Result	14/5/0	10/9/0	11/6/2	10/4/5



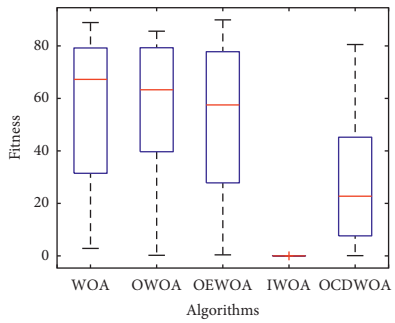
(a)



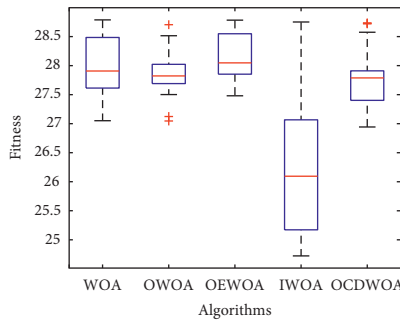
(b)



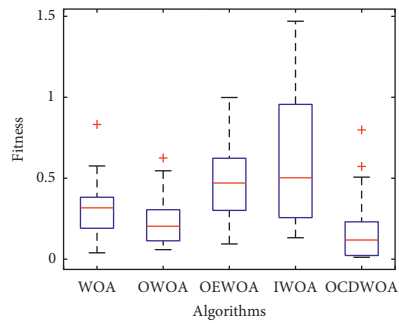
(c)



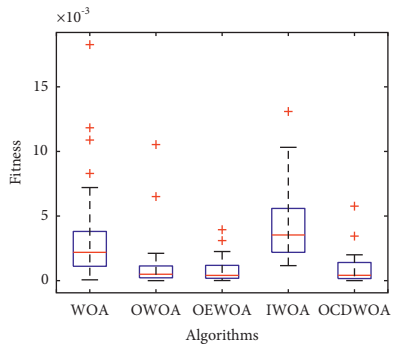
(d)



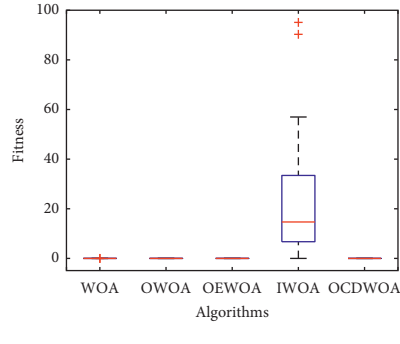
(e)



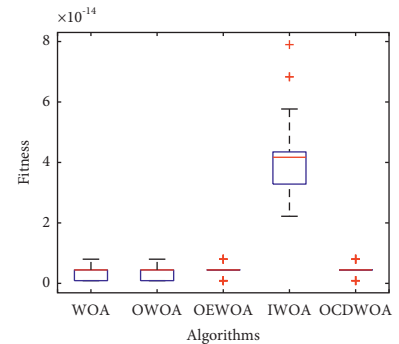
(f)



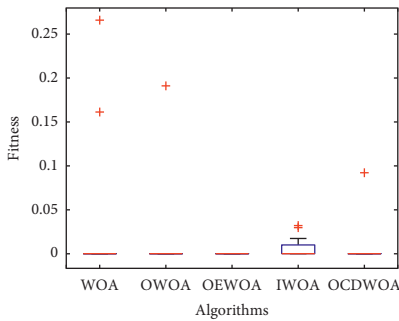
(g)



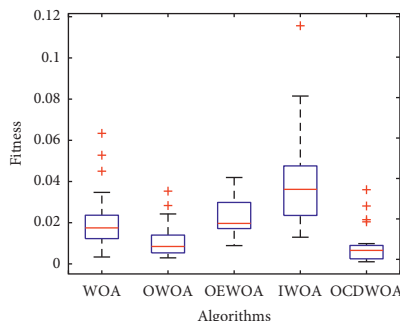
(h)



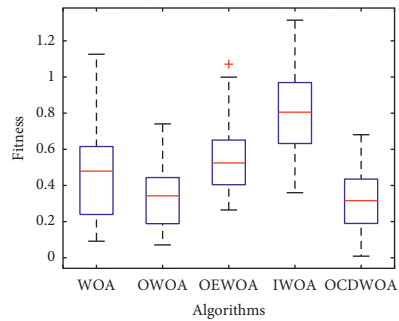
(i)



(j)



(k)



(l)

FIGURE 3: Continued.

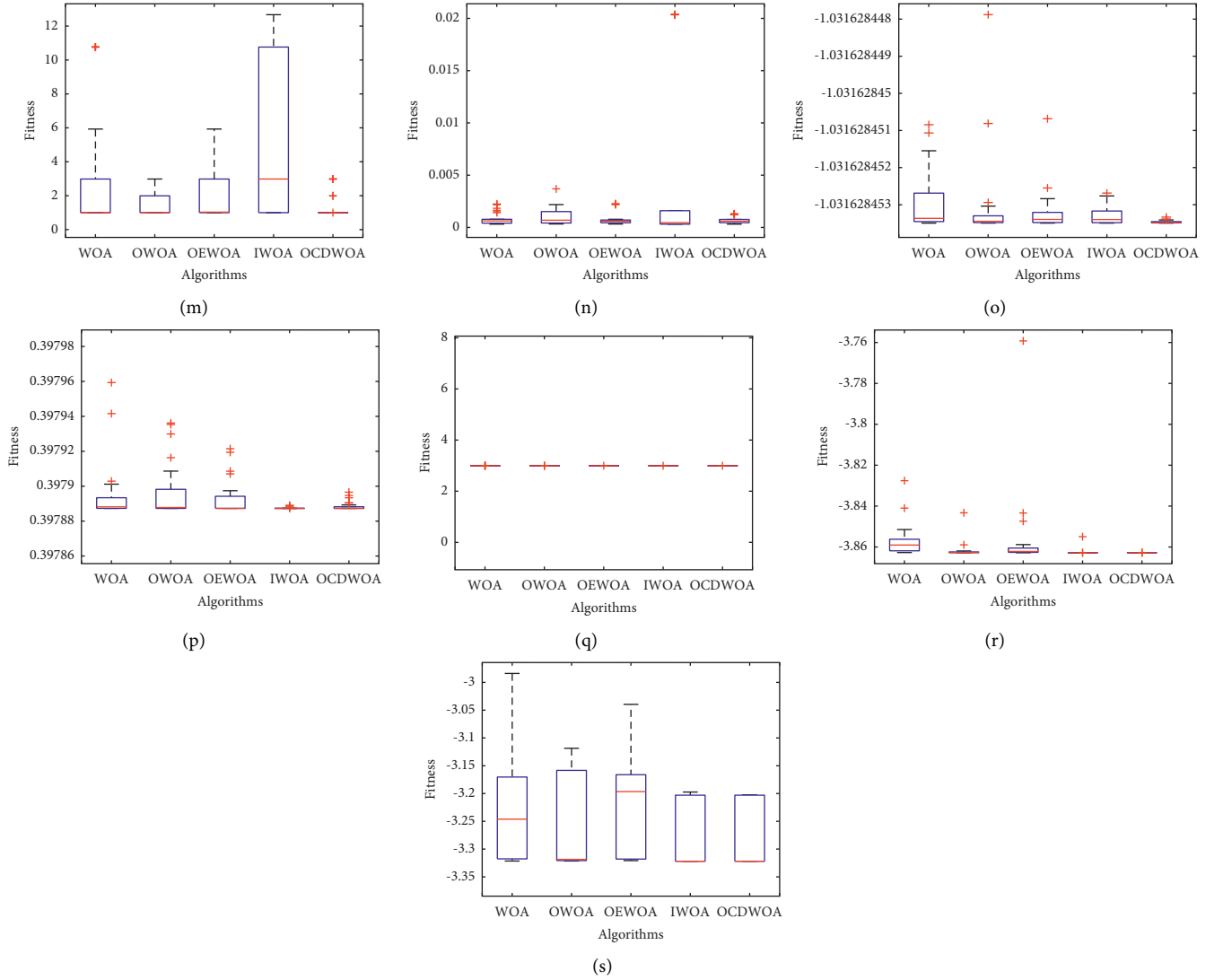


FIGURE 3: Boxplots of OCDWOA with respect to other algorithms. (a) F_1 . (b) F_2 . (c) F_3 . (d) F_4 . (e) F_5 . (f) F_6 . (g) F_7 . (h) F_8 . (i) F_9 . (j) F_{10} . (k) F_{11} . (l) F_{12} . (m) F_{13} . (n) F_{15} . (o) F_{16} . (p) F_{17} . (q) F_{18} . (r) F_{19} . (s) F_{19} .

TABLE 4: Values of the theoretical geological model.

	V_P (m·s ⁻¹)	V_S (m·s ⁻¹)	ρ (g·cm ⁻³)
Layer 1	1980	851	2
Layer 2	2185	1605	2.2
Layer 3	1983	856	2
Layer 4	2200	1622	2.2
Layer 5	2437	1510	2.3
Layer 6	2663	1380	2.4
Layer 7	1983	851	2
Layer 8	2437	1501	2.3
Layer 9	1980	856	2
Layer 10	2656	1380	2.4
Layer 11	1983	851	2

TABLE 5: Results of different algorithms.

Problem	Algorithm	Best	Mean	Std
Seismic inversion	WOA	0.007178754	0.0124127116	0.0038698430
	OWOA	0.006168262	0.0119509070	0.0053173170
	OEWOA	0.006901579	0.0112028011	0.0035417924
	IWOA	0.006136581	0.0128976065	0.0041582246
	OCDWOA	0.002662229	0.0055975119	0.0019518514

The bold values mean the best results.

In this paper, a theoretical geological model with 11 layers is established [47]. Each layer in the model contains 3 parameters: velocity of longitudinal wave (V_P), velocity of shear wave (V_S), and density (ρ). The values of model are

shown in Table 4.

In this paper, the reflection coefficients of the model will be calculated by Zoeppritz equation as follows:

$$\begin{bmatrix} R_P \\ R_S \\ T_P \\ T_S \end{bmatrix} = \begin{bmatrix} -\sin \theta_1 & -\cos \phi_1 & \sin \theta_2 & \cos \phi_2 \\ \cos \theta_1 & -\sin \phi_1 & \cos \theta_2 & -\sin \phi_2 \\ \sin 2\theta_1 & \frac{V_{P1}}{V_{S1}} \cos 2\phi_1 & \frac{\rho_2 V_{S2}^2 V_{P1}}{\rho_1 V_{S1}^2 V_{P2}} \cos 2\phi_1 & \frac{\rho_2 V_{S2} V_{P1}}{\rho_1 V_{S1}^2} \cos 2\phi_2 \\ -\cos 2\phi_1 & \frac{V_{S1}}{V_{P1}} \sin 2\phi_1 & \frac{\rho_2 V_{P2}}{\rho_1 V_{P1}} \cos 2\phi_2 & -\frac{\rho_2 V_{S2}}{\rho_1 V_{P1}} \sin 2\phi_2 \end{bmatrix}^{-1} \begin{bmatrix} \sin \theta_1 \\ \cos \theta_1 \\ \sin 2\theta_1 \\ \cos 2\phi_1 \end{bmatrix}, \quad (17)$$

where V_{P1} , V_{S1} , and ρ_1 are the parameters of the up-layer, V_{P2} , V_{S2} , and ρ_2 are the parameters of the low-layer, θ_1 and θ_2 are the incident angle and the refraction angle of the P-wave, respectively, and Φ_1 and Φ_2 are the incident angle and the refraction angle of the S-wave, respectively.

Φ_1 , θ_2 , and Φ_2 are calculated as follows, respectively:

$$\begin{aligned} \phi_1 &= \arcsin\left(\sin(\theta_1) \times \frac{V_{S1}}{V_{P1}}\right), \\ \theta_2 &= \arcsin\left(\sin(\theta_1) \times \frac{V_{S2}}{V_{P1}}\right), \\ \phi_2 &= \arcsin\left(\sin(\theta_1) \times \frac{V_{P2}}{V_{P1}}\right). \end{aligned} \quad (18)$$

Incident angles θ_1 of 50 reflection coefficients are from 1° to 50° . The 50 seismic records are obtained by convolving the reflection coefficients with the Ricker wavelet with 35 Hz as follows:

$$S = R * R_w, \quad (19)$$

where S is the seismic record, R is the reflection coefficient, and R_w is the ricker wavelet (35 Hz).

The 50 seismic records will be divided into three groups. The three groups are small incident angle group (Group 1), medium incident angle group (Group 2), and large incident

angle group (Group 3). Group 1 consists of the seismic records whose incident angles are from 1° to 15° . Group 2 consists of the seismic records whose incident angles are from 16° to 32° . Group 3 consists of the seismic records whose incident angles are from 33° to 50° . Superimposed seismic records S_1 , S_2 , and S_3 are obtained by synthesizing three groups of seismic records, respectively. The superimposed seismic records $S_1(X)$, $S_2(X)$, and $S_3(X)$ of the solution X are obtained the same way as the model.

The fitness of the solution X is expressed as follows:

$$F_{S1}(X) = \frac{1}{3} (\|S_1(X) - S_1\|_2 + \|S_2(X) - S_2\|_2 + \|S_3(X) - S_3\|_2). \quad (20)$$

4.3.2. Experiment Results. The experiment is repeated 10 times for each algorithm. Table 5 shows the best fitness, mean fitness, and standard deviation of test results on seismic inversion problem. The convergence curves and boxplots are shown in Figure 4. From the test results, the performance of OCDWOA for seismic inversion problem is the best.

In Table 5, the best fitness, mean fitness, and standard deviation of OCDWOA are all obviously smaller than those of the comparative algorithms. In Figure 4(a), the convergence rate of OCDWOA is better than those of the comparative algorithms. The convergence curve of OCDWOA is

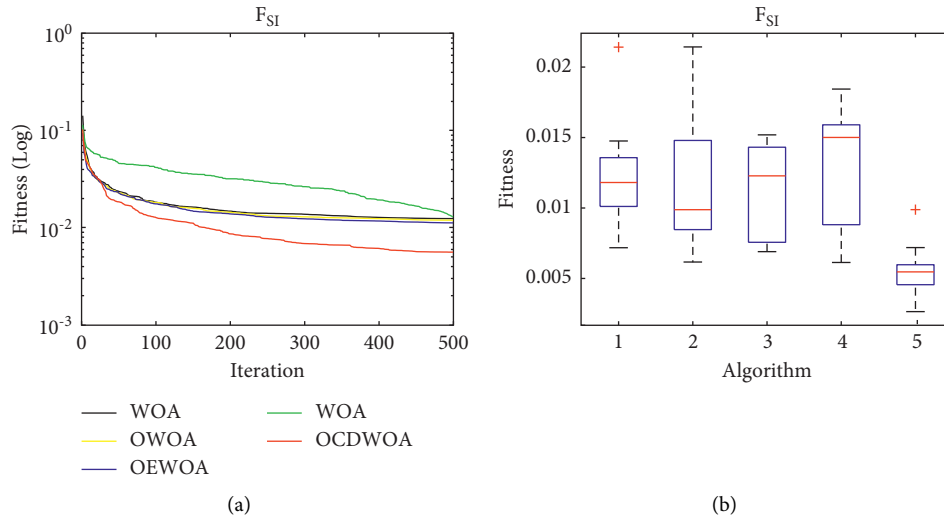


FIGURE 4: Results of OCDWOA with respect to other algorithms. (a) Convergence graph of the algorithms. (b) Boxplots of the algorithms.

stable. In Figure 4(b), the stability of OCDWOA is the best among the comparative algorithms.

5. Conclusions

In this paper, a modified WOA named OCDWOA is proposed. In order to balance the exploitation phase and exploration phase, a nonlinear parameter design is used. OBL, DE operator, and a density-based clustering strategy are introduced into the proposed algorithm to improve the performance of the search phase and the ability of jumping out of the local optimal solution. The proposed algorithm is tested on 19 optimization benchmark functions and a seismic inversion problem. The proposed algorithm is compared with WOA and the other three variants of WOA. Experimental results show that the proposed algorithm can enhance the performance of WOA and is better than other compared algorithms.

Our future work will pay attention to the following two points: first, we would like to research the relation between the parameters and the performance of the OCDWOA. Second, we would like to apply the OCDWOA to machine learning.

Data Availability

Some or all data, models, or codes that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

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

Acknowledgments

This work was supported by National Natural Science Foundation of China under Grants 41772123, 61772365, 61802280, and 61806143.

References

- [1] G. Venter and S. S. Jaroslaw, "Particle swarm optimization," *AIAA Journal*, vol. 41, no. 8, pp. 129–132, 2003.
- [2] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28–39, 2006.
- [3] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.
- [4] A. Sm, B. Smm, and A. Al, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
- [5] S. Wu and F.-M. Zhang, "Wolf pack algorithm for unconstrained global optimization," *Mathematical Problems in Engineering*, vol. 2014, Article ID 465082, 17 pages, 2014.
- [6] L. Ma, S. Cheng, and Y. Shi, "Enhancing learning efficiency of brain storm optimization via orthogonal learning design," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 11, pp. 6723–6742, 2021.
- [7] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.
- [8] G. Kaur and S. Arora, "Chaotic whale optimization algorithm," *Journal of Computational Design and Engineering*, vol. 5, no. 3, pp. 275–284, 2018.
- [9] Y. Zhou, Y. Ling, and Q. Luo, "Lévy flight trajectory-based whale optimization algorithm for global optimization," *IEEE Access*, vol. 5, pp. 6168–6186, 2017.
- [10] H. Chen, Y. Xu, M. Wang, and X. Zhao, "A balanced whale optimization algorithm for constrained engineering design problems," *Applied Mathematical Modelling*, vol. 71, no. JUL, pp. 45–59, 2019.
- [11] X. Chen, "Research on new adaptive whale algorithm," *IEEE Access*, vol. 8, Article ID 90165, 2020.
- [12] Y. Sun, X. Wang, Y. Chen, and Z. Liu, "A modified whale optimization algorithm for large-scale global optimization problems," *Expert Systems with Applications*, vol. 114, no. DEC, pp. 563–577, 2018.
- [13] Q. Zhang and L. Liu, "Whale optimization algorithm based on lamarckian learning for global optimization problems," *IEEE Access*, vol. 7, 2019.

- [14] I. N. Trivedi, J. Pradeep, J. Narottam, K. Arvind, and L. Dilip, "Novel adaptive whale optimization algorithm for global optimization," *Indian Journal of Science and Technology*, vol. 9, no. 38, pp. 319–326, 2016.
- [15] Y. Sun, T. Yang, and Z. Liu, "A whale optimization algorithm based on quadratic interpolation for high-dimensional global optimization problems," *Applied Soft Computing*, vol. 85, Article ID 105744, 2019.
- [16] H. Chen, C. Yang, A. A. Heidari, and X. Zhao, "An efficient double adaptive random spare reinforced whale optimization algorithm," *Expert Systems with Applications*, vol. 154, Article ID 113018, 2019.
- [17] I. N. Trivedi, P. Jangir, A. Kumar, N. Jangir, and R. Totlani, "A novel hybrid PSO-WOA algorithm for global numerical functions optimization," in *Advances in Computer and Computational Sciences*, pp. 53–60, Springer, Berlin, Germany, 2018.
- [18] S. Z. Wei, W. J. Sheng, and W. Xian, "An improved whale optimization algorithm based on different searching paths and perceptual disturbance," *Symmetry Plus*, vol. 10, no. 6, 2018.
- [19] M. M. Mafarja and S. Mirjalili, "Hybrid whale optimization algorithm with simulated annealing for feature selection," *Neurocomputing*, vol. 260, pp. 302–312, 2017.
- [20] Y. X. Ting and H. U. H. Bai, "A whale optimization algorithm with inertia weight," *WSEAS Transactions on Computers*, vol. 15, pp. 319–326, 2016.
- [21] M. Abdel-Basset, G. Manogaran, D. El-Shahat, and S. Mirjalili, "Integrating the whale algorithm with Tabu search for quadratic assignment problem: a new approach for locating hospital departments," *Applied Soft Computing*, vol. 73, pp. 530–546, 2018.
- [22] D. Oliva, M. E. Abd Elaziz, and A. E. Hassanien, "Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm," *Applied Energy*, vol. 200, pp. 141–154, 2017.
- [23] J. Zhang, D. Zhong, M. Zhao, J. Yu, and F. Lv, "An optimization model for construction stage and zone plans of rockfill dams based on the enhanced whale optimization algorithm," *Energies*, vol. 12, no. 3, 2019.
- [24] B. Bentouati, L. Chaib, and S. Chettih, "A hybrid whale algorithm and pattern search technique for optimal power flow problem," in *Proceedings of the 2016 8th international conference on modelling, identification and control (ICMIC)*, pp. 1048–1053, IEEE, Algiers, Algeria, November 2016.
- [25] M. Abd Elaziz and D. Oliva, "Parameter estimation of solar cells diode models by an improved opposition-based whale optimization algorithm," *Energy Conversion and Management*, vol. 171, no. 1083–186, pp. 1843–1859, 2018.
- [26] I. Aljarah, H. Faris, and S. Mirjalili, "Optimizing connection weights in neural networks using the whale optimization algorithm," *Soft Computing*, vol. 22, no. 1, pp. 1–15, 2018.
- [27] K. b. o. Medani, S. Sayah, and A. Bekrar, "Whale optimization algorithm based optimal reactive power dispatch: a case study of the Algerian power system," *Electric Power Systems Research*, vol. 163, pp. 696–705, 2018.
- [28] L. Ma, M. Huang, S. Yang, R. Wang, and X. Wang, "An adaptive localized decision variable analysis approach to large-scale multiobjective and many-objective optimization," *IEEE Transactions on Cybernetics*, 2021.
- [29] L. Ma, N. Li, Y. Guo et al., "Learning to optimize: reference vector reinforcement learning adaption to constrained many-objective optimization of industrial copper burdening system," *IEEE Transactions on Cybernetics*, 2021.
- [30] G. Xia, M. K. Sen, and P. L. Stoffa, "1-D elastic waveform inversion: a divide-and-conquer approach," *Geophysics*, vol. 63, no. 5, pp. 1670–1684, 1998.
- [31] M. K. Sen, "Pre-stack waveform inversion: current status and future directions," in *Proceedings of the 7th International Congress of the Brazilian Geophysical Society*, European Association of Geoscientists & Engineers, Bahia, Brazil, October 2001, <https://www.earthdoc.org/content/papers/10.3997/2214-4609-pdb.217.141>.
- [32] L. Ingber and B. Rosen, "Genetic algorithms and very fast simulated reannealing: a comparison," *Mathematical & Computer Modeling*, vol. 16, no. 11, 1992.
- [33] Liu, W. A. N. G. Sixiu, H. U. Deli, and Bin, "Pre-stack seismic waveform inversion based on adaptive genetic algorithm," *Global Geology*, vol. 22, no. 3, pp. 50–60, 2019.
- [34] Q. Wu, Z. Zhu, and X. Yan, "Research on the parameter inversion problem of prestack seismic data based on improved differential evolution algorithm," *Cluster Computing*, vol. 20, no. 4, pp. 2881–2890, 2017.
- [35] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-based differential evolution," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 64–79, 2008.
- [36] S. Mirjalili, S. M. Mirjalili, S. Saremi, and S. Mirjalili, "Whale optimization algorithm: theory, literature review, and application in designing photonic crystal filters," *Nature-Inspired Optimizers*, pp. 219–238, 2020.
- [37] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.
- [38] J. Cai, H. Wei, H. Yang, and X. Zhao, "A novel clustering algorithm based on DPC and PSO," *IEEE Access*, vol. 8, Article ID 88200, 2020.
- [39] K. V. Price, "Differential evolution," in *Handbook of Optimization*, pp. 187–214, Springer, Berlin, Heidelberg, 2013.
- [40] P. N. Suganthan, N. Hansen, J. J. Liang et al., "Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization," *Nanyang Technol. University*, Singapore, 2005.
- [41] H. S. Alamri, Y. A. Alsariera, and K. Z. Zamli, "Opposition-based whale optimization algorithm," *Advanced Science Letters*, vol. 24, no. 10, pp. 7461–7464, 2018.
- [42] R. Salgotra, U. Singh, and S. Saha, "On some improved versions of whale optimization algorithm," *Arabian Journal for Science and Engineering*, vol. 44, no. 11, pp. 9653–9691, 2019.
- [43] S. Mostafa Bozorgi and S. Yazdani, "TWOA: an improved whale optimization algorithm for optimization problems," *Journal of Computational Design and Engineering*, vol. 6, no. 3, pp. 243–259, 2019.
- [44] J. Pafeng, S. Mallick, and H. S. Sharma, "3-D prestack waveform inversion-a real data example," in *Proceedings of the 2015 SEG Annual Meeting*, OnePetro, New Orleans, LO, USA, October 2015, <https://onepetro.org/SEGAM/conferences/browse-by-conference>.
- [45] J. P. Tchuindjang, S. Mallick, and H. Sharma, "Reservoir characterization using prestack waveform inversion," in *Proceedings of the SEG Technical Program Expanded Abstracts 2016*, pp. 3597–3602, Society of Exploration Geophysicists, Dallas, TX, USA, September 2016.
- [46] J. Pafeng, S. Mallick, and H. Sharma, "Prestack waveform inversion of three-dimensional seismic data - an example from the Rock Springs Uplift, Wyoming, USA," *Geophysics*, vol. 82, no. 1, pp. B1–B12, 2017.

- [47] H. D. Huang, Y. C. Wang, F. Guo, and C. H. Liu, "High precision pre-stack inversion algorithm based on Zoeppritz equations," *OG*, vol. 48, no. 5, pp. 740–749, 2013.