Improved whale optimization algorithms based on inertia weights and theirs applications

Hongping Hu, Yanping Bai, and Ting Xu

***Abstract*—**Whale optimization algorithm (WOA), which mimics the social behavior of humpback whales, was proposed by Seyedali Mirjalili and Andrew Lewis in 2016.This paper introduces the inertia weights to WOA to obtain the improved whale optimization algorithms(IWOAs). IWOAs are tested with 27 mathematical benchmark functions and are applied to predict daily air quality index(AQI) of Taiyuan.The results show that IWOAs with inertia weights are superior to WOA,FOA,ABC,and PSO on the minimum of benchmark functions and are very competitive for prediction compared with WOA and PSO.

***Keywords*—**air quality index prediction, benchmark function, improved whale optimization algorithm, inertia weight

I. Introduction

There are more and more meta-heuristic optimization algorithms which are used extensively in science, engineering and business because they: (i) have a few parameters; (ii) do not require gradient information; (iii) can bypass local optima; (iv) can be utilized to solve the practical problems.

The fruit fly optimization algorithm(FOA) first proposed by Pan [1] in 2012, who provided an easy and powerful approach to handle the complex optimization problems,simulates the intelligent foraging behavior of fruit flies or vinegar flies in finding food. Fruit flies live in the temperate and tropical climate zones. They have very sensitive osphresis and vision organs which are superior to other species. Therefore, FOA is composed of sensitive osphresis and vision part. Fruit flies mainly use osphresis and vision to find food and can collect different kinds of airborne smells, even when the food source is 40 km away. Fruit flies use osphresis to search for food along the scent concentration path, and then use visual flight to the group gathering place or the food source.Since then, more and more researchers improve FOA and apply FOA to different regions[2-4].

As a relatively new optimization method inspired by swarm intelligence, artificial bee colony algorithm(ABC) proposed by Karaboga [5] in 2005 imitates the foraging behavior of honeybees, which consists of three kinds of honey bees:employed bees, onlooker bees and scout bees.In ABC, the number employed bees equal to the number of onlooker bees, and also equal to the number of food sources. A food source position represents a possible solution to the problem that is to be optimized and the nectar of a food source corresponds to the quality of the solution represented by the food source. During each cycle, the employed and onlooker bees are moving toward the food sources, thus calculating the nectar amounts and determining the scout bee and then moving them randomly onto the possible food sources. If the solution does not improve by a predetermined number of trials, the food source is abandoned and the corresponding employed bee is converted to the scout bee. Since 2005, researchers devote themselves to the search methods and applications of ABC[6-10].

The particle swarm optimization algorithm(PSO)was first proposed by Kennedy and Eberhart (1995)[11], which was used to simulate the group behavior. In PSO, the swarm changes its direction during its movement and therefore there are velocity update and position update. The swarm in PSO contains a lot of candidate solutions,which are treated as birds and are also called particles. Initially, these particles have the random direction and velocity. Then each particle changes its own position and velocity based on the experiences of itself and its neighbors. Finally, by the fitness values of each particle and iterations,the global solution for the overall swarm is obtained. In PSO, many researchers introduce “inertia weight” and propose many dynamic variations of PSO based on the inertia weight.Different inertia weight strategies imply different incremental changes in pursuit of a better solution [12-19].

Besides the above three swarm intelligence algorithms, there are other swarm intelligence algorithms such as the ant colony optimization(ACO)[20-21], genetic algorithm(GA) [22-23] that simulates the Darwinian evolution, Evolution Strategy(ES) [24-26], and differential evolution algorithm(DE) [27-28].

In 2016, Seyedali Mirjalili and Andrew Lewis first propose a new meta-heuristic optimization algorithm(namely, Whale Optimization Algorithm,WOA) mimicking the hunting behavior of humpback whales[29].

Fig.1[29]shows the special hunting method of the humpback whales. Humpback whales prefer to hunt school of krill or small fishes close to the surface,whose foraging is done by creating distinctive bubbles along a circle or ‘9’-shaped

This research is supported by Nation Science Foundation of China 61275120.

Hongping Hu is with School of College, North University of China,Taiyuan,Shanxi,030051 China (corresponding author to provide phone: +8603513942729; fax: +8603513942729; e-mail: hhp92@ 163.com).

Yanping Bai is with with School of College, North University of China,Taiyuan,Shanxi,030051 China (e-mail: baiyp666@163.com).

Ting Xu is with with School of College, North University of China,Taiyuan,Shanxi,030051 China (e-mail: xuting@nuc.edu.cn).

path that can only be observed in humpback whales as shown in Fig. 1 .

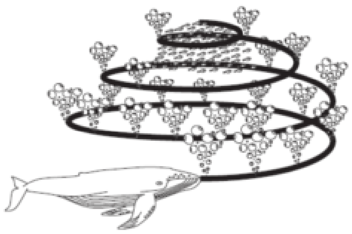


Fig.1 Bubble-net feeding behavior of humpback whales

In this paper, we introduce different inertia weights into whale optimization algorithm (IWOA) to get the better of benchmark functions and apply for AQI prediction of Taiyuan.

The structure of the rest of the paper is as follows. In Section II, basic whale optimization algorithm is described. In Section III, the inertia weight is introduced into WOA and improved whale optimization algorithm(IWOA) is proposed. In section IV, 27 benchmark functions are introduced and IWOA,WOA, FOA,ABC,and PSO are compared. In section V, we apply IWOA,WOA,and PSO for AQI prediction of Taiyuan. Section VI summarizes the main findings of this study and suggests directions for future research.

II. Basic Whale Optimization Algorithm

In this section, we describe the mathematical modal of the basic whale optimization algorithm in [29].

A. Encircling Prey

Humpback whales can recognize the location of prey and then encircle them. For the unknown position of the optimal design in the search space, the current best candidate solution is the target prey or is close to the optimum in WOA. Once the best search agent is defined, the other search agents will hence try to update their positions towards the best search agent. The updated method is represented by the following equations:

$$D = | C.X^*(t) - X(t) | \tag{1}$$

$$X(t+1) = X^*(t) - A.D \tag{2}$$

where the meanings of $t, A, C, X^*, X, |\ |$ and . are shown in Table 1.

Table 1. Meanings of $t, A, C, X^*, X, |\ |$ and .

Symbol	Meaning
t	the current iteration
A	coefficient vectors
C	coefficient vectors
X^*	the position vector of the best solution obtained so far
X	the position vector
$\|\ \|$	the absolute value
.	an element-by-element multiplication

The vectors A and C are calculated in the following:

$$A = 2ar - a \tag{3}$$

$$C = 2r \tag{4}$$

where a is linearly decreased from 2 to 0 over the course of iterations (in both exploration and exploitation phases) and r is a random vector in [0,1].

B. Bubble-net Attacking Method (Exploitation Phase)

Two improved approaches are designed as follows for mathematically simulating the bubble-net behavior of humpback whales:

One is Shrinking encircling mechanism obtained by decreasing the value of a in the (3). Note that A is a random value in the interval $[-a, a]$ where a is decreased from 2 to 0 during iterations. Setting random values for A in $[-1,1]$, we can define the new position of a search agent anywhere in between the original position of the agent and the position of the current best agent.

The other is spiral updating position created between the position of whale and prey to mimic the helix-shaped movement of humpback whales as follows:

$$X(t+1) = D'.e^{bl}.\cos(2\pi l) + X^*(t) \tag{5}$$

where $D' = | X^*(t) - X(t) |$ is the distance of the ith whale to the prey (best solution obtained), b is a constant connected with the shape of the logarithmic spiral, l is a random number in $[-1,1]$, and . is an element-by-element multiplication.

Humpback whales swim around the prey within a shrinking circle and along a spiral-shaped path simultaneously. So we assume that there is a chance of a probability about 50% to choose between either the shrinking encircling mechanism or the spiral updating position of whales during optimization. The mathematical model is shown as (6):

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

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

C. Search For Prey (Exploration Phase)

The humpback whales search for prey randomly except for the bubble-net method. Similar to the approach based on the variation of the A vector, humpback whales search randomly according to the position of each other. Therefore, A with the random values greater than 1 or less than −1 is utilized to make search agent move far away from a reference whale. Different from the exploitation phase, we update the position of a search agent in the exploration phase when a randomly chosen search agent is in place of the best search agent found so far. The mechanism and $|A| > 1$ focus on exploration and allow the WOA algorithm to perform a global search. The mathematical model is as shown in the following:

$$D = | C.X_{rand} - X | \tag{7}$$

$$X(t+1) = X_{rand} - A.D \tag{8}$$

where X_{rand} is a random position vector (a random whale) chosen from the current population.

In WOA algorithm, a set of random solutions are taken. The a parameter is decreased from 2 to 0 providing both exploration and exploitation. At each iteration, search agents gradually update their positions using either a randomly chosen search agent or the best solution obtained so far. If $|A| > 1$, a random search agent is chosen, otherwise the best solution is selected for updating the position of the search agents. According to p, WOA is able to switch between either a spiral or circular movement. Then the WOA is terminated according to a termination criterion.

The concrete steps of the WOA are the following:

Step1. Initialize the whales population $X_i (i = 1,2,\cdots,n)$ and Maxgen(maximum number of iterations).Let $t = 1$.

Step2. Calculate the fitness of $X_i (i = 1,2,\cdots,n)$,and find the best search solution X^*.

Step3. Repeat the following:

For every $X_i (i = 1,2,\cdots,n)$,update a, A, C, l, p .

If $p < 0.5$,then if $|A| < 1$,update the position of the current search agent by the (1) and if $|A| \geq 1$,select a random search solution X_{rand} and update the position of the current search agent by the (8).

If $p \geq 0.5$,update the position of the current search by the (5).

Check if any search agent goes beyond the search and amend it.Calculate the fitness of $X_i (i = 1,2,\cdots,n)$,and if there is a better solution,find the best search solution X^*.

Let $t = t + 1$.

Until t reaches Maxgen iteration, the algorithm is finished.

Step4. Return the best optimization solution X^* and the best optimization value of fitness values.

III. Improved Whale Optimization Algorithm

In WOA, the updated solution is mostly depended on the the current best candidate solution. Similar to PSO algorithm, an inertia weight $\omega \in [0,1]$ is introduced into WOA to obtain the improved whale optimization algorithm (IWOA).

In Encircling prey, the updated method is represented by the following equations:

$$D = | C.\omega X^*(t) - X(t) | \tag{9}$$

$$X(t+1) = \omega X^*(t) - A.D \tag{10}$$

where the meanings of $t, A, C, X^*, X, |\ |$ and . are shown in table 1.

In exploitation phase, a spiral equation created between the position of whale and prey to mimic the helix-shaped movement of humpback whales is as follows:

$$X(t+1) = D'.e^{bl}.\cos(2\pi l) + \omega X^*(t) \tag{11}$$

where $D' = | \omega X^*(t) - X(t) |$ and indicates the distance of the ith whale to the prey (best solution obtained so far), b is a constant for defining the shape of the logarithmic spiral, l is a random number in [$-1,1$], and . is an element-by-element multiplication.

Similar to WOA,we assume that there is a chance of a probability about 50% to choose between either the shrinking encircling mechanism or the spiral updating position of whales with inertia weight during optimization. The mathematical model is as follows:

$$X(t+1) = \begin{cases} \omega X^*(t) - A.D & if\ p < 0.5 \\ D'.e^{bl}.\cos(2\pi l) + \omega X^{*(t)} & if\ p \geq 0.5 \end{cases} \tag{12}$$

where p is a random number in $[0,1]$. In addition to the bubble-net method, the humpback whales search for prey randomly.

In search for prey (exploration phase),the same approach based on the variation of the A vector can be utilized to search for prey (exploration). The mathematical model is as follows:

$$D = | wC.X_{rand} - X | \tag{13}$$

$$X(t+1) = wX_{rand} - A.D \tag{14}$$

where X_{rand} is a random position vector (a random whale) chosen from the current population.

The concrete steps of the IWOA are the following:

Step1. Initialize the whales population $X_i (i = 1,2,\cdots,n)$ and Maxgen(maximum number of iterations).Let $t = 1$.

Step2. Calculate the fitness of $X_i (i = 1,2,\cdots,n)$,and find the best search solution X^*.

Step3. Repeat the following:

For every $X_i (i = 1,2,\cdots,n)$,update a, A, C, l, p .

If $p < 0.5$,then if $|A| < 1$,update the position of the current search agent by the (9) and if $|A| \geq 1$,select a random search solution X_{rand} and update the position of the current search agent by the (14).

If $p \geq 0.5$,update the position of the current search by the (11).

Check if any search agent goes beyond the search and amend it.Calculate the fitness of $X_i (i = 1,2,\cdots,n)$,and if there is a better solution,find the best search solution X^*.

Let $t = t + 1$.

Until t reaches Maxgen, the algorithm is finished.

Step4. Return the best optimization solution X^* and the best optimization value of fitness values.

There are the formulas of inertia weight ω in PSO algorithms in the following:

$$\omega(t) = \omega_{initial} - (\omega_{initial} - \omega_{final}) \frac{t}{T}, \tag{15}$$

$$\omega_i(t) = \frac{1 - \frac{t}{T}}{1 + s * \frac{t}{T}}, \tag{16}$$

$$\omega(t) = \omega_{terinal} + (\omega_{initial} - \omega_{final}) e^{-\frac{ct}{T}}, \tag{17}$$

where t is the number of current iterative steps, T is the maximum number of iterative steps allowed to continue, $\omega_{initial}$ is the initial inertia weight, ω_{final} is the final inertia weight, s is a constant larger than -1 and c is controlling parameter to control the convergence rate of the inertia weight, $c > 0$. Equation(15) is introduced by Shi and Eberhart[12] who introduce a Linear Decreasing Inertia Weight(LDIW) strategy in 1998, (16) is introduced by Lei et al. [14]who propose a Sugeno function as inertia weight(SFIW) method in which the inertia weight is neither set to a constant value nor set as linearly decreasing time-varying function, and (17) is introduced by Lu,Hu and Bai[17] who propose an Exponential Decreasing Inertia Weight (EDIW) strategy.

Thus four kinds of IWOAs are obtained as follows:

(1) IWOA with constant inertia weight(IWOA-CIW),

(2) IWOA with dynamic inertia weight shown (15) (IWOA-LDIW),
(3) IWOA with dynamic inertia weight shown (16) (IWOA-SFIW),
(4) IWOA with dynamic inertia weight show (17) (IWOA-EDIW).

IV. Numerical Simulations

A. Benchmark Functions

In order to test the performance of the IWOA, 27 benchmark functions commonly used in the literature [2-3,29] are taken, which consist of 18 unimodal functions and 8 multimodal functions. 27 benchmark functions with *n*-dimension are concrete in the following where $f_6 = f_4 + f_5$ and $f_{12} = f_{10} + f_{11}$.

(1) $f_1 = \sum_{i=1}^{n} x_i^2$,where $-100 \le x_i \le 100$. The minimum value is 0.

(2) $f_2 = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i|$,where $-10 \le x_i \le 10$. The minimum value is 0.

(3) $f_3 = \max_i\{|x_i|, 1 \le i \le n\}$, where $-100 \le x_i \le 100$. The minimum value is 0.

(4) $f_4 = \sum_{i=1}^{n-1} 100(x_{i+1} - x_i^2)^2$,where $-30 \le x_i \le 30$. The minimum value is 0.

(5) $f_5 = \sum_{i=1}^{n-1} (x_i - 1)^2$,where $-30 \le x_i \le 30$. The minimum value is 0.

(6) $f_6 = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$,where $-30 \le x_i \le 30$. The minimum value is 0.

(7) $f_7 = \sum_{i=1}^{n} (\lfloor x_i + 0.5 \rfloor)^2$,where $-100 \le x_i \le 100$. The minimum value is 0.

(8) $f_8 = \sum_{i=1}^{n} i x_i^4 + rand()$,where $-1.28 \le x_i \le 1.28$. The minimum value is 0.

(9) $f_9 = \sum_{i=2}^{n} i x_i^2$,where $-5.12 \le x_i \le 5.12$. The minimum value is 0.

(10) $f_{10} = \sum_{i=2}^{n} i(2x_i^2 - x_{i-1})^2$,where $-10 \le x_i \le 10$. The minimum value is 0.

(11) $f_{11} = (x_1 - 1)^2$,where $-10 \le x_i \le 10$. The minimum value is 0.

(12) $f_{12} = \sum_{i=2}^{n} i(2x_i^2 - x_{i-1})^2 + (x_1 - 1)^2$,where $-10 \le x_i \le 10$. The minimum value is 0.

(13) $f_{13} = -\exp(-0.5\sum_{i=1}^{n} x_i^2)$,where $-1 \le x_i \le 1$. The minimum value is -1.

(14) $f_{14} = \sum_{i=1}^{n} (10^6)^{\frac{i-1}{n-1}} x_i^2$,where $-100 \le x_i \le 100$. The minimum value is 0.

(15) $f_{15} = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j\right)^2$,where $-100 \le x_i \le 100$. The minimum value is 0.

(16) $f_{16} = \sum_{i=1}^{n} |x_i|^{i+1}$,where $-1 \le x_i \le 1$. The minimum value is 0.

(17) $f_{17} = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$, where $-32 \le x_i \le 32$. The minimum value is 0.

(18) $f_{18} = \sum_{i=1}^{n} |x_i \sin(x_i) + 0.1x_i|$,where $-10 \le x_i \le 10$. The minimum value is 0.

(19) $f_{19} = f_s(x_1, x_2) + f_s(x_2, x_3) + \cdots + f_s(x_n, x_1)$, where $f_s(x, y) = 0.5 + \frac{\sin^2(\sqrt{x^2 + y^2}) - 0.5}{(1 + 0.001(x^2 + y^2))^2}$, $-100 \le x_i \le 100$. The minimum value is 0.

(20) $f_{20} = f_{10}(x_1, x_2) + \cdots + f_{10}(x_{n-1}, x_n) + f_{10}(x_n, x_1)$ where $f_{10}(x, y) = (x^2 + y^2)^{0.25}[\sin^2(50(x^2 + y^2)^{0.1}) + 1]$, $-100 \le x_i \le 100$. The minimum value is 0.

(21) $f_{21} = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$,where $-600 \le x_i \le 600$. The minimum value is 0.

(22) $f_{22} = -\sum_{i=1}^{n-1}\left(\exp\left(-\frac{x_i^2 + x_{i+1}^2 + 0.5x_i x_{i+1}}{8}\right)\cos\left(4\sqrt{x_i^2 + x_{i+1}^2 + 0.5x_i x_{i+1}}\right)\right)$, where $-5 \le x_i \le 5$. The minimum value is $1-n$.

(23) $f_{23} = \sum_{i=1}^{n-1}\left(0.5 + \frac{\sin^2(\sqrt{100x_i^2 + x_{i+1}^2}) - 0.5}{1 + 0.001(x_i^2 - 2x_i x_{i+1} + x_{i+1}^2)^2}\right)^2$,where $-100 \le x_i \le 100$. The minimum value is 0.

(24) $f_{24} = \sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi x_i) + 10)$,where $-5.12 \le x_i \le 5.12$. The minimum value is 0.

(25) $f_{25} = 1 - \cos\left(2\pi\sqrt{\sum_{i=1}^{n} x_i^2}\right) + 0.1\sqrt{\sum_{i=1}^{n} x_i^2}$,where $-100 \le x_i \le 100$. The minimum value is 0.

(26) $f_{26} = \sum_{i=1}^{n} i x_i^2$,where $-10 \le x_i \le 10$. The minimum value is 0.

(27) $f_{27} = -\sum_{i=1}^{n} x_i \sin\sqrt{|x_i|}$,where $-500 \le x_i \le 500$. The minimum value of f_{22} is -418.9829*5.

2D representations of the above 27 benchmark mathematical functions with $n = 2$ are shown in Fig.2-Fig.4.

In this section,we compare the proposed IWOAs with basic WOA,basic ABC algorithm,basic FOA, and the basic PSO based on 27 benchmark functions. For all the algorithms, a population size and maximum iteration number equal to 30 and 500,respectively,have been utilized. We run 30 replications for these 27 benchmark functions.

B. IWOA-CIW vs. WOA

In IWOA-CIW experiments, the mean values and standard deviations(std) are obtained with the increase of ω varying step length 0.1 from 0 to 1. When $\omega = 1$, IWOA becomes WOA.When $\omega = 0$, updated solution does not depends on the current best solution. Table 2-table 4 show that the mean and the standard deviation of IWOA based on the increase of ω varying step length 0.1 from 0 to 1.

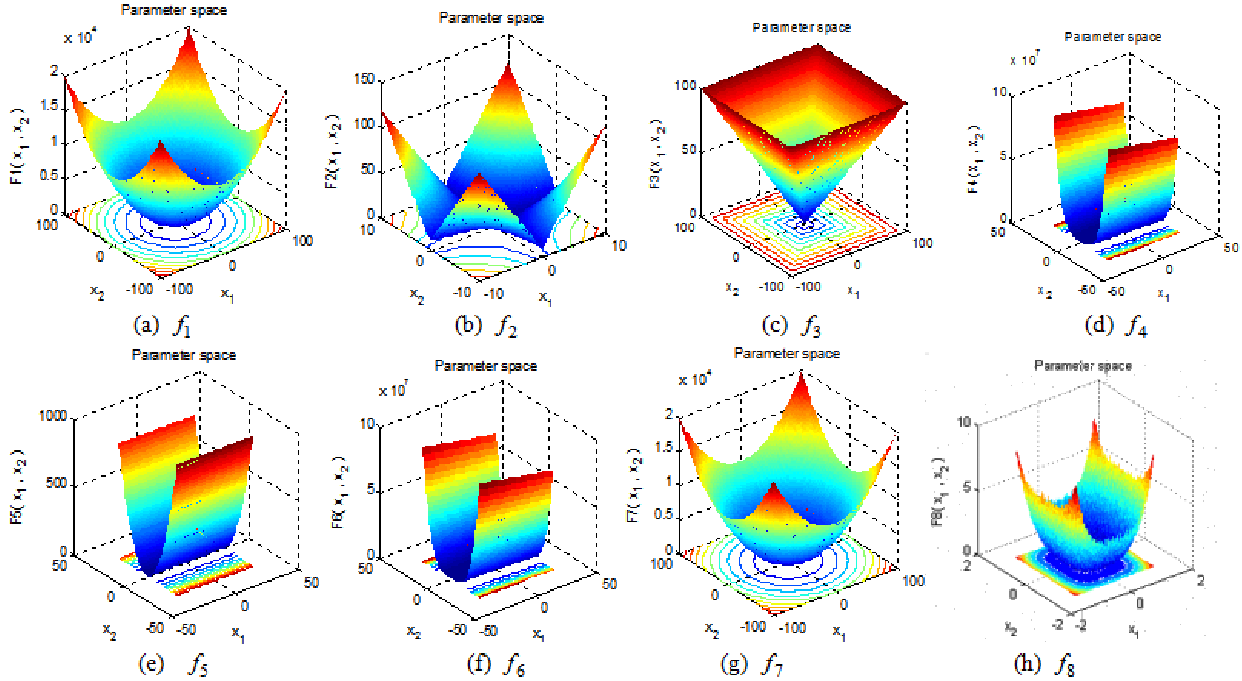


Fig.2 2D representations of functions $f_1 - f_8$

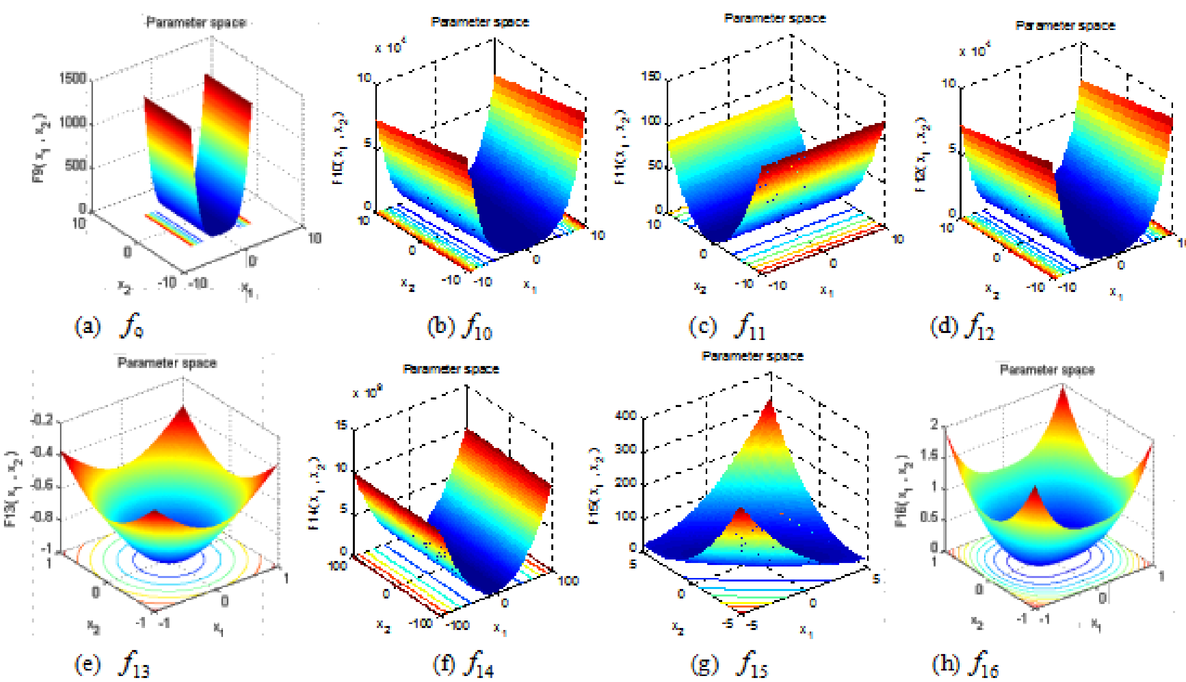
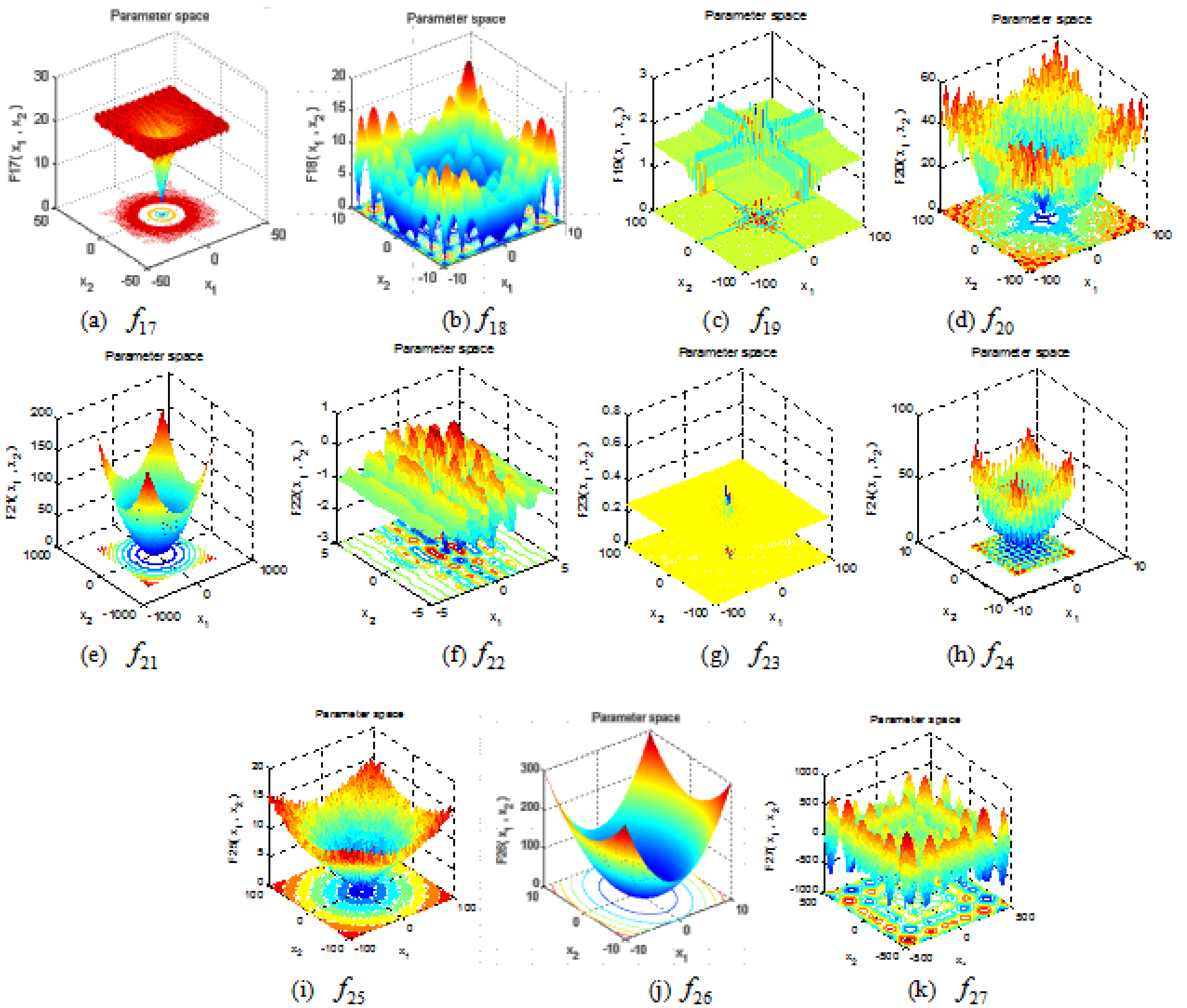


Fig.3 2D representations of functions $f_9 - f_{16}$

From Table 2-Table 4, it is shown that IWOA-CIW is superior to WOA. The values of all the functions $f_1 - f_4, f_7, f_9, f_{10}, f_{13} - f_{26}$ are increasing with the inertia weight's increase. But the value of the function f_{11} is decreasing with the inertia weight's increase. And functions f_5, f_6, f_{27} cannot trend to the minimum values. Although functions f_4, f_{10} can reach the minimum values, function f_5, f_{11} cannot reach the minimum value, so $f_6 = f_4 + f_5$, $f_{12} = f_{10} + f_{11}$ cannot reach the minimum value and have a greater bias.

Hence, it can be concluded that as a whole the proposed IWOA-CIW significantly improves the basic WOA. And the smaller value of inertia weight is taken,the easier IWOA-CIW is to trend the minimum value of function.



(a) f_{17} (b) f_{18} (c) f_{19} (d) f_{20} (e) f_{21} (f) f_{22} (g) f_{23} (h) f_{24} (i) f_{25} (j) f_{26} (k) f_{27}

Fig.4 2D representations of functions $f_{17} - f_{27}$

Table 2 The mean and standard deviation of IWOA-CIW based on the increase of ω varying step length 0.1 from 0 to 1 from f_1 to f_9

ω		f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9
0	mean	0	0	0	0	1.1647	28.5894	0	5.9332e-05	0
	Std	0	0	0	0	0.7049	0.7024	0	5.5715e-05	0
0.1	mean	0	9.7881e-275	1.3884e-269	0	1.1461	28.2808	0	5.8764e-05	0
	Std	0	0	0	0	0.5723	0.3379	0	4.6065e-05	0
0.2	mean	0	9.1595e-218	3.0795e-221	0	1.3790	28.1812	0	5.1179e-05	0
	Std	0	0	0	0	0.7804	0.3197	0	5.0852e-05	0
0.3	mean	0	1.4647e-192	1.2851e-183	0	1.2956	28.1090	0	7.0771e-05	0
	Std	0	0	0	0	0.5638	0.3218	0	4.8943e-05	0
0.4	mean	0	2.7450e-167	1.0486e-152	0	1.0954	28.0748	0	9.2813e-05	0
	Std	0	0	5.5670e-152	0	0.4976	0.3954	0	9.7650e-05	0
0.5	mean	1.6771e-274	4.4696e-144	4.0812e-123	1.9985e-274	09552	27.9438	0	8.0091e-05	0
	Std	0	1.2140e-143	1.6887e-122	0	0.4400	0.3404	0	6.7522e-05	0
0.6	mean	1.1027e-224	3.7936e-123	3.7481e-94	1.1708e-227	0.7364	27.9135	0	8.9908e-05	0
	Std	0	1.3634e-122	1.8858e-93	0	0.2646	0.3516	0	9.2702e-05	0
0.7	mean	5.2662e-184	1.7259e-101	9.8701e-67	8.2835e-183	0.5858	27.8589	0	9.6107e-05	0
	Std	0	6.0877e-101	5.2914e-66	0	0.2038	0.3414	0	1.0087e-04	0
0.8	mean	1.7841e-142	8.6564e-83	1.3256e-46	2.1361e-144	0.4229	27.8984	0	1.2481e-04	9.1861e-263
	Std	7.1847e-142	2.4291e-82	4.1233e-46	8.9505e-144	0.1837	0.3395	0	1.1877e-04	0
0.9	mean	3.2628e-106	2.3786e-65	8.2539e-24	1.2952e-107	0.2780	27.8757	0	3.0871e-04	5.4649e-184
	Std	1.3377e-105	7.2710e-65	2.0755e-23	4.3302e-107	0.1304	0.3737	0	2.4146e-04	0
1	mean	6.5673e-74	7.9402e-51	52.2806	3.6599e-68	0.4382	28.0018	0	0.0042	1.3330e-114
	Std	3.4065e-73	4.0809e-50	30.8406	1.4110e-67	0.2928	0.4320	0	0.0038	4.2194e-114

However,the IWOA-CIW generates small gaps with the true optimal values. There is a room for the IWOA to be further improved in the future research.

C. IWOAs vs. WOA vs. ABC vs. FOA vs. PSO

By a lot of experiments, we take $\omega = 0.1$ for example in IWOA-CIW. Here, four IWOA algorithms(IWOA-CIW, IWOA-LDIW, IWOA-SFIW,and IWOA-EDIW) are compared with WOA,FOA,ABC,and PSO. Optimization results reported in Table 5-table 7 show that the IWOAs can well balance exploration and exploitation phases.

From table 5-table 7, the values of functions $f_1 - f_4$, $f_7 - f_{26}$ using IWOAs are all less than those using WOA,FOA, ABC,and PSO and trend to their minimum values by the less iteration number. But IWOAs cannot solve the minimum values of functions f_5, f_6 and f_{27}. It can be seen that IWOAs are competitive with other meta-heuristic algorithm: WOA, FOA,ABC, and PSO can hence provide very good exploitation.

D. Analysis Of Convergence Behavior

Convergence curves of four IWOAs,WOA,FOA, ABC, and PSO are compared in Fig. 5-Fig.7 for 27 benchmark functions. It can be seen that IWOAs are enough competitive.

The convergence curves of four IWOAs,WOA,FOA, ABC, and PSO are provided in Fig.5-Fig.7 to see the convergence rate of the algorithms. Here average best-so-far in each iteration over 30 runs.

Although $f_6 = f_4 + f_5$, $f_{12} = f_{10} + f_{11}$ and f_4, f_{10} have an ability to reach the minimum values within 500 iterations as shown in Fig.5-Fig.7, f_6 and f_{12} cannot trend to the minimum values owing to f_5 and f_{11} without the convergence of the minimum values.

Table 3 The mean and standard deviation of IWOA-CIW based on the increase of ω varying step length 0.1 from 0 to 1 from f_{10} to f_{18}

ω		f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}
0	mean	0	1.3375e-07	0.9853	-1	0	0	0	8.8818e-16	0
	Std	0	3.2872e-07	0.0104	0	0	0	0	0	0
0.1	mean	0	1.1370e-07	0.6672	-1	0	0	0	8.8818e-16	6.9235e-272
	Std	0	2.5164e-07	6.4777e-04	0	0	0	0	0	0
0.2	mean	0	4.9797e-08	0.6668	-1	0	0	0	8.8818e-16	6.8634e-219
	Std	0	6.9118e-08	7.3830e-05	0	0	0	0	0	0
0.3	mean	0	6.5610e-08	0.6668	-1	0	0	0	8.8818e-16	1.4805e-193
	Std	0	1.0477e-07	1.4740e-04	0	0	0	0	0	0
0.4	mean	0	4.0028e-08	0.6668	-1	0	3.8765e-282	0	8.8818e-16	1.4342e-166
	Std	0	7.0027e-08	1.3420e-04	0	0	0	0	0	0
0.5	mean	3.3193e-274	2.9708e-08	0.6668	-1	1.1189e-270	5.2376e-219	0	8.8818e-16	3.0588e-143
	Std	0	5.0772e-08	9.8997e-05	0	0	0	0	0	1.5335e-142
0.6	mean	5.5534e-228	1.6383e-08	0.6668	-1	1.8096e-223	8.7377e-169	1.2550e-304	2.1908e-15	2.7729e-122
	Std	0	3.3148e-08	1.9409e-04	0	0	0	0	1.7413e-15	7.4797e-122
0.7	mean	4.9173e-181	1.5190e-08	0.6669	-1	1.0597e-178	9.5127e-124	3.1358e-255	3.0198e-15	1.2530e-103
	Std	0	2.4319e-08	1.8176e-04	0	0	3.6297e-123	0	2.4277e-15	3.9272e-103
0.8	mean	4.9602e-143	2.8363e-09	0.6669	-1	2.0298e-141	1.3646e-81	3.5562e-205	1.7906e-15	5.8627e-85
	Std	2.7074e-142	5.0060e-09	4.4162e-04	0	6.6685e-141	4.9961e-81	0	1.7702e-15	1.7866e-84
0.9	mean	1.0288e-106	5.5892e-10	0.6674	-1	3.1224e-102	9.3366e-40	2.8814e-154	3.2567e-15	1.7677e-67
	Std	5.6347e-106	1.0350e-09	4.4162e-04	2.9156e-17	1.6742e-101	1.2615e-39	1.4912e-153	2.5265e-15	5.5500e-67
1	mean	2.9016e-75	1.2582e-15	0.6671	-1	1.3006e-69	4.6135e+04	1.2370e-107	3.8488e-15	2.3483e-49
	Std	1.0297e-74	4.4536e-15	5.9043e-04	6.1849e-17	6.8643e-69	1.2825e+04	6.5916e-107	3.3747e-15	1.2408e-48

Table 4 The mean and standard deviation of IWOA-CIW based on the increase of ω varying step length 0.1 from 0 to 1 from f_{19} to f_{27}

ω		f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}	f_{25}	f_{26}	f_{27}
0	mean	0	0	0	-29	2.7733e-32	0	0	0	-1.2531e+04
	Std	0	0	0	0	0	0	0	0	148.2557
0.1	mean	0	0	0	-29	2.7733e-32	0	0	0	-1.2509e+04
	Std	0	0	0	0	0	0	0	0	161.6693
0.2	mean	0	0	0	-29	2.7733e-32	0	0	0	-1.2545e+04
	Std	0	0	0	0	0	0	0	0	81.9206
0.3	mean	0	0	0	-29	2.7733e-32	0	0	0	-1.2551e+04
	Std	0	0	0	0	0	0	0	0	49.5361
0.4	mean	0	0	0	-29	2.7733e-32	0	7.1067e-164	0	-1.2544e+04
	Std	0	0	0	0	0	0	0	0	78.0868
0.5	mean	0	1.8510e-73	0	-29	1.8141e-14	0	0.0033	2.8529e-275	-1.2365e+04
	Std	0	3.2577e-73	0	0	9.8641e-14	0	0.0182	0	441.3631
0.6	mean	0	1.3378e-62	0	-29	6.4087e-15	0	0.0033	4.3029e-228	-1.2319e+04
	Std	0	5.2859e-62	0	0	3.2571e-14	0	0.0182	0	733.1612
0.7	mean	0	4.6115e-53	0	-29	2.3408e-13	0	0.0100	7.8627e-186	-1.2206e+04
	Std	0	1.4895e-52	0	0	7.5007e-13	0	0.0305	0	759.4011
0.8	mean	0	9.1298e-45	0	-29	1.4745e-08	0	0.0266	7.9829e-141	-1.1959e+04
	Std	0	3.1710e-44	0	0	7.0957e-08	0	0.0449	4.3724e-140	1.2588e+03
0.9	mean	0.2732	8.8715e-36	0	-29	7.7707e-08	0	0.0566	2.5342e-106	-1.0822e+04
	Std	1.4962	3.0517e-35	0	0	2.4323e-07	0	0.0503	9.6745e-106	2.0176e+03
1	mean	2.7914	2.4425e-28	3.7007e-18	-27.9343	2.1586e-07	0	0.1166	1.0666e-73	-1.0830e+04
	Std	4.0965	8.3472e-28	2.0270e-17	4.1260	1.0878e-06	0	0.0699	3.5973e-73	1.6987e+03

Table 5 Comparison with four IWOA,WOA,FOA,ABC, and PSO from f_1 to f_9

Algorithm		f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9
IWOA-CIW	mean	0	9.7881e-275	1.3884e-269	0	1.1678	28.3746	0	5.8764e-05	0
	Std	0	0	0	0	0.6645	0.2886	0	4.6065e-05	0
IWOA-LDIW	mean	0	1.0132e-228	1.2739e-220	0	0.6277	27.9940	0	1.1992e-04	0
	Std	0	0	0	0	0.2160	0.4094	0	7.9489e-05	0
IWOA-SFIW	mean	0	1.7721e-290	4.5192e-285	0	0.9693	28.1463	0	7.2559e-05	0
	Std	0	0	0	0	0.4565	0.3386	0	5.9930e-05	0
IWOA-EDIW	mean	0	2.3976e-211	1.4581e-208	0	0.9691	28.0572	0	1.2625e-04	0
	Std	0	0	0	0	0.5093	0.4427	0	1.2604e-04	0
WOA	mean	6.5673e-74	7.9402e-51	52.2806	3.6599e-68	0.4382	28.0018	0	0.0042	1.3330e-114
	Std	3.4065e-73	4.0809e-50	30.8406	1.4110e-67	0.2928	0.4320	0	0.0038	4.2194e-114
FOA	mean	1.0010e-08	0.0055	1.8315e-05	3.0944	25.1264	28.7094	0	0.0033	5.8874e-05
	Std	1.4142e-10	3.4600e-05	1.5755e-07	0.8696	0.4144	0.0047	0	7.9224e-04	8.0579e-07
ABC	mean	4.1071e-04	6.7956e-03	63.9439	36.9137	2.44886e-05	45.6653	1.4333	0.260689	3.07474e-05
	Std	5.9621e-04	3.1589e-03	4.83185	27.9241	3.08525e-05	26.1978	0.7739	0.0670099	6.85987e-05
PSO	mean	0.3663	3.9027	3.9458	191.9301	0.2313	293.4490	12.1667	0.0395	0.1531
	Std	0.1035	1.4724	1.8696	174.2475	0.0761	202.2484	7.4282	0.0395	0.1198

Table 6 Comparison with four IWOA,WOA,FOA,ABC,PSO ,and DE from f_{10} to f_{18}

Algorithm		f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}
IWOA-CIW	mean	0	1.1370e-07	0.6672	-1	0	0	0	8.8818e-16	6.9235e-272
	Std	0	2.5164e-07	6.4777e-04	0	0	0	0	0	0
IWOA-LDIW	mean	0	9.1917e-09	0.6670	-1	0	0	0	8.8818e-16	3.5969e-231
	Std	0	1.5441e-08	2.1378e-04	0	0	0	0	0	0
IWOA-SFIW	mean	0	3.8185e-08	0.6668	-1	0	0	0	8.8818e-16	6.8228e-288
	Std	0	5.6226e-08	1.3094e-04	0	0	0	0	0	0
IWOA-EDIW	mean	0	2.2704e-08	0.6670	-1	0	0	0	8.8818e-16	1.0821e-213
	Std	0	5.6301e-08	3.1925e-04	0	0	0	0	0	0
WOA	mean	2.9016e-75	1.2582e-15	0.6671	-1	1.3006e-69	4.6135e+04	1.2370e-107	3.8488e-15	2.3483e-49
	Std	1.0297e-74	4.4536e-15	5.9043e-04	6.1849e-17	6.8643e-69	1.2825e+04	6.5916e-107	3.3747e-15	1.2408e-48
FOA	mean	0.9978	0.2105	0.9978	-0.9999	8.7745e-04	3.1339e-06	3.3271e-06	2.2782e-04	5.4855e-04
	Std	2.9911e-05	0.1795	3.1756e-05	5.5532e-07	8.3306e-06	4.4659e-08	4.7056e-08	1.5902e-06	4.4818e-06
ABC	mean	0.89943	5.9916e-05	1.01636	-1	26.7196	17465.7	3.53512e-09	0.113683	0.0383125
	Std	0.878592	1.1370e-04	0.806525	6.5510e-08	27.1426	3031.86	5.50039e-09	0.150522	0.0285267
PSO	mean	12.9836	3.5501e-09	11.6649	-0.9475	7.7968e+05	55.3275	0	4.2481	1.9677
	Std	7.4848	6.0949e-09	5.2364	0.1360	3.1086e+05	26.7810	0	0.7559	0.8599

Table 7. Comparison with five IWOA,WOA,FOA,ABC,PSO ,and DE from f_{19} to f_{27}

Algorithm		f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}	f_{25}	f_{26}	f_{27}
IWOA-CIW	mean	0	0	0	-29	2.7733e-32	0	0	0	-1.2509e+04
	Std	0	0	0	0	0	0	0	0	161.6693
IWOA-LDIW	mean	0	0	0	-29	4.8768e-13	0	8.4296e-123	0	-1.2204e+04
	Std	0	0	0	0	2.6581e-12	0	4.6171e-122	0	890.8391
IWOA-SFIW	mean	0	0	0	-29	4.5424e-16	0	0	0	-1.2370e+04
	Std	0	0	0	0	2.0139e-15	0	0	0	426.7174
IWOA-EDIW	mean	0	0	0	-29	1.0690e-09	0	7.6683e-79	0	-1.2386e+04
	Std	0	0	0	0	4.0712e-09	0	4.2001e-78	0	489.7182
WOA	mean	2.7914	2.4425e-28	3.7007e-18	-27.9343	2.1586e-07	0	0.1166	1.0666e-73	-1.0830e+04
	Std	4.0965	8.3472e-28	2.0270e-17	4.1260	1.0878e-06	0	0.0699	3.5973e-73	1.6987e+03
FOA	mean	1.9948e-08	2.3325	1.1496e-10	-28.9999	5.9786	7.5656e-04	1.0198e-05	1.5439e-05	-1.0131
	Std	2.5651e-10	0.7373	2.6833e-12	9.7549e-07	0.1763	1.1233e-05	9.4037e-08	2.2077e-07	0.7013
ABC	mean	3.3145	58.8593	0.124501	-22.665	2.5944	4.10815	3.9224	4.32582e-05	-11548.6
	Std	0.4311	12.7812	0.0320711	0.708797	0.166755	0.610904	0.629669	4.14654e-05	172.349
PSO	mean	8.5288	107.9469	25.8416	-15.4390	0.2478	78.8370	1.3707	8.2483	-3.1569e+03
	Std	1.1389	10.0164	4.7892	1.9440	0.0021	14.3689	0.3826	3.9217	361.4954

As shown in Fig.5-Fig.7, the IWOAs shows that three different convergence behaviors when optimizing 27 benchmark functions:

(1) The convergence of IWOAs tend to be accelerated as the iteration increase,

(2) IWOAs trend to convergence within less iterations,

(3) IWOAs have the rapid convergence from the initial steps of iterations.

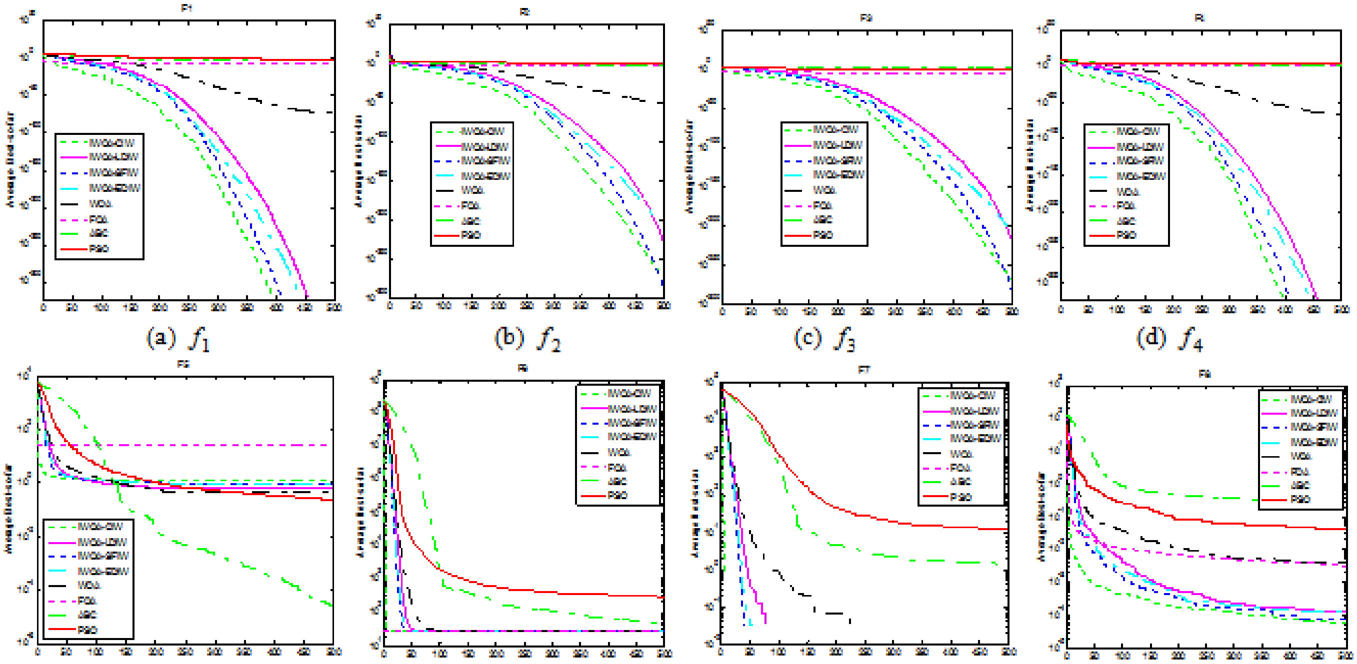
These behaviors are obvious in functions $f_1 - f_4$, $f_7 - f_{10}$, $f_{13} - f_{26}$. The results show that the IWOAs are high in solving benchmark functions.

V. Apply IWOAs For AQI Prediction Of Taiyuan

According to the convergences of the four IWOAs, WOA, FOA, ABC,and PSO on 27 benchmark functions, we can see that FOA and ABC are inferior to IWOAs,WOA,and PSO. Therefore, we apply IWOAs,WOA,and PSO for prediction of Taiyuan.

In recent years, more and more people focus on the the air quality problem and find out some methods to improve the air quality and analyze the influence factors. Daily air quality index(AQI) is described by the six indicators: sulfur dioxide (SO2), nitrogen dioxide(NO2), particulate matter (PM10: particle size is less than or equal to10 microns), particulate matter(PM2.5: particle size is less than or equal to 2.5 microns), carbon monoxide(CO),and ozone(O3). Among them,SO2,NO2,and CO are all the 24-hour average density; O3 is the 8-hour moving average density. We choose 1100 sets of data from December 2 in 2013 to December 5 in 2016 as train data and 22 sets of data from December 6 in 2016 to December 27 in 2016 as test data. Fig.8 show that the actual daily AQI of Taiyuan from December 2 in 2013 to December 27 in 2016.

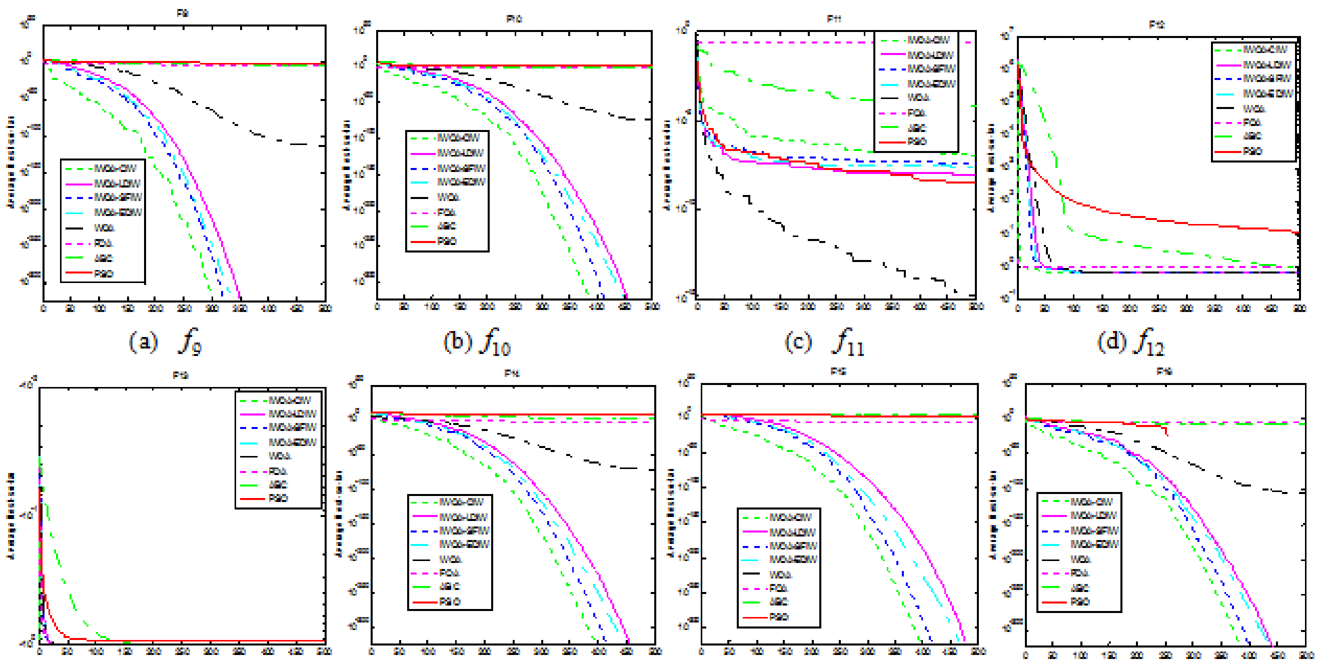
In this section, IWOAs,WOA,and PSO are used for optimizing the parameters of linear regression(LR) model for air quality index(AQI) prediction of Taiyuan.



(a) f_1 (b) f_2 (c) f_3 (d) f_4

(e) f_5 (f) f_6 (g) f_7 (h) f_8

Fig.5 The convergence curves of functions $f_1 - f_8$



(a) f_9 (b) f_{10} (c) f_{11} (d) f_{12}

(e) f_{13} (f) f_{14} (g) f_{15} (h) f_{16}

Fig.6 The convergence curves of functions $f_9 - f_{16}$

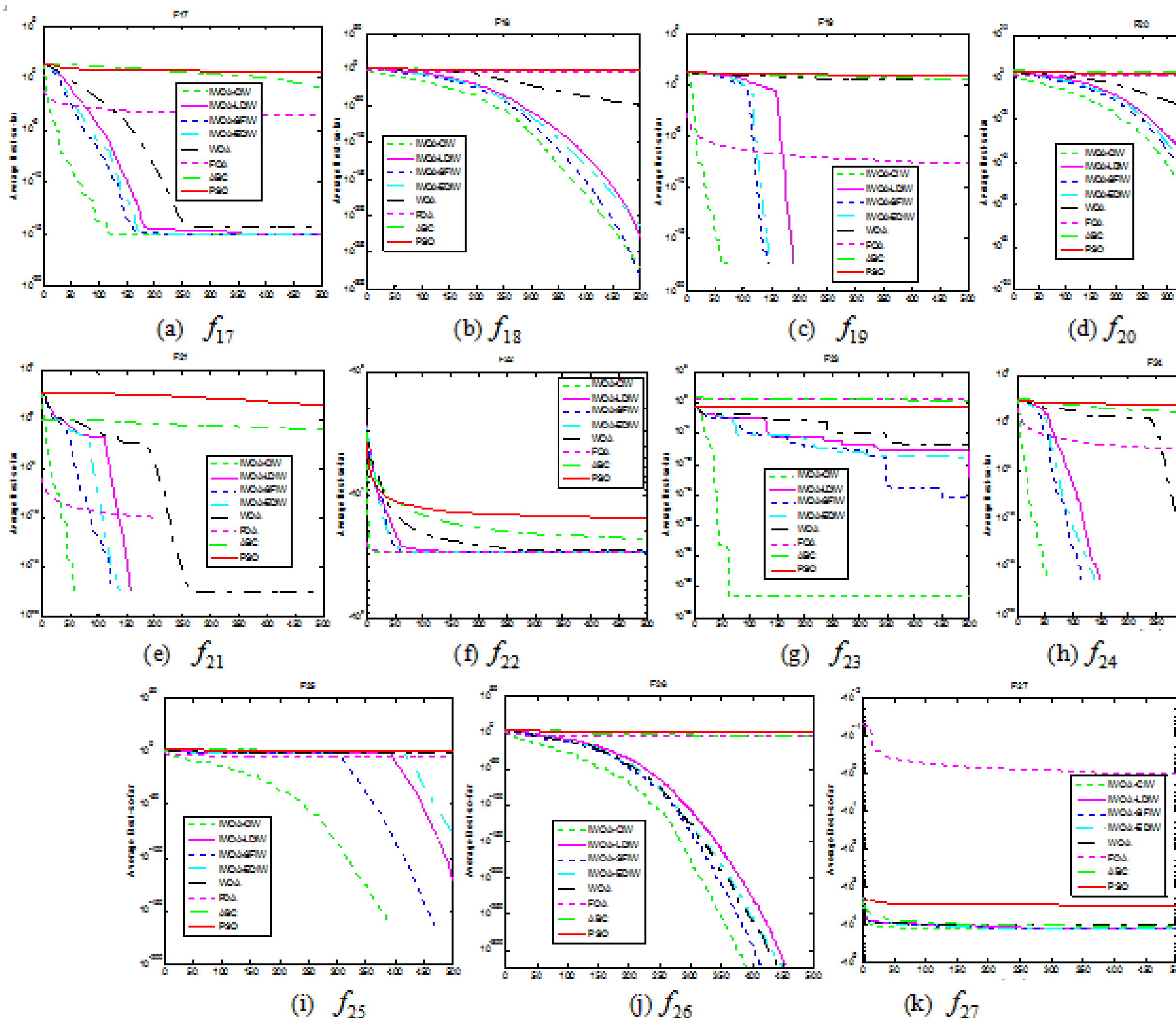


Fig.7. The convergence curves of functions $f_{17}-f_{27}$

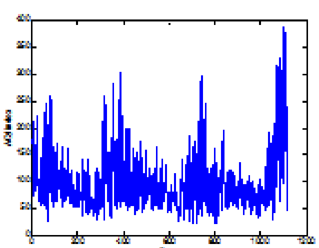


Fig.8. The actual daily air quality index(AQI) of Taiyuan from December 2 in 2013 to December 27 in 2016.

As shown above, AQI depends on the six indicators: PM2.5, PM10, SO2, CO,NO2,O3. We decide to approximate AQI as a linear function of these six indicators' values $x_{\mathrm{PM2.5}}, x_{\mathrm{PM10}}, x_{\mathrm{SO_2}}, x_{\mathrm{CO}}, x_{\mathrm{NO_2}}, x_{\mathrm{O_3}}$:

$$\begin{aligned}\mathrm{AQI} = \theta_0 + \theta_1 x_{\mathrm{PM2.5}} + \theta_2 x_{\mathrm{PM10}} + \theta_3 x_{\mathrm{SO_2}} \\ + \theta_4 x_{\mathrm{CO}} + \theta_5 x_{\mathrm{NO_2}} + \theta_6 x_{\mathrm{O_3}}.\end{aligned} \tag{18}$$

where the θ_i 's are the parameters of linear functions.

In order to asses the performance of the above model, we take mean square error (MSE),relative mean square error(RMSE) and mean absolute percentage error(MAPE) as criteria, defined as

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2, \tag{19}$$

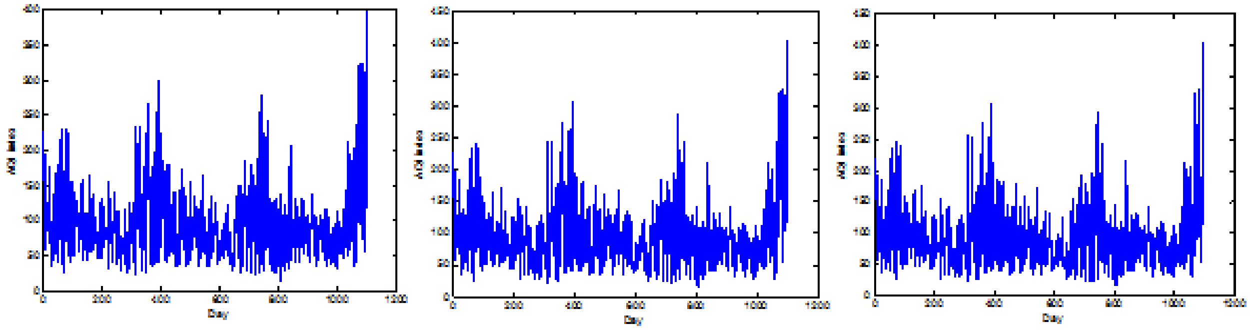
$$\mathrm{RMSE} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{y_i - \hat{y}_i}{\hat{y}_i}\right)^2, \tag{20}$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{\hat{y}_i}\right| \times 100\%, \tag{21}$$

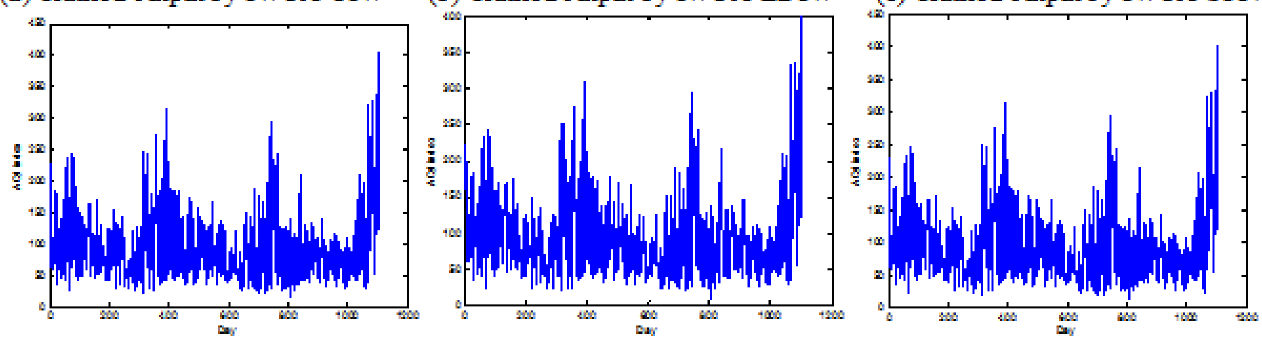
where $\hat{y}_i$ and y_i denote the actual value and the output value by (18), respectively. Fig. 9 shows the trained output curve by the IWOAs for the AQI of Taiyuan. Fig. 10 shows the plots of trained absolute errors between the trained outputs and the actual valued by the IWOA algorithms. And the values of MSE,RMSE, and MAPE(%) of trained output by the IWOAs for AQI of Taiyuan are shown in Table 8.

Based on the trained optimal parameters of (18) by the IWOA algorithms,respectively, we predict the AQI index of Taiyuan of following 22 days from December 6 in 2016 to

December 27 in 2016. Table 9 shows the predicted outputs by using the IWOA algorithms. Table 10 shows that the values of MSE,RMSE, and MAPE(%) of predicted output by the IWOA algorithms for AQI of Taiyuan. From Table 8 and Table 10, we can see that IWOAs with stable inertia weights or dynamic inertia weights are superior to WOA and PSO with respect to MSE,RMSE, and MAPE and therefore are more adaptive to predict the AQI values.

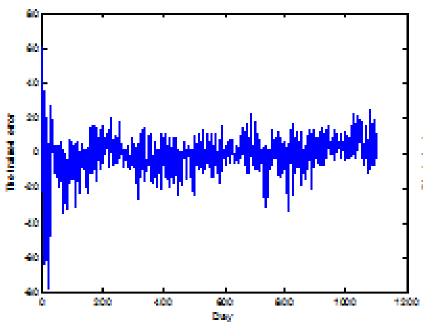


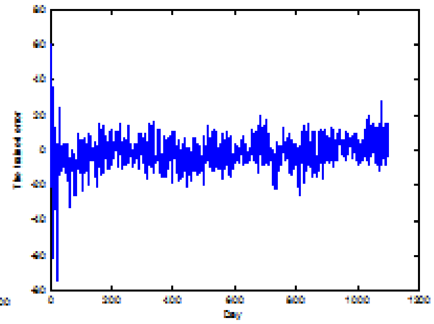
(a) Trained output by IWOA-CIW (b) Trained output by IWOA-LDIW (c) Trained output by IWOA-SFIW

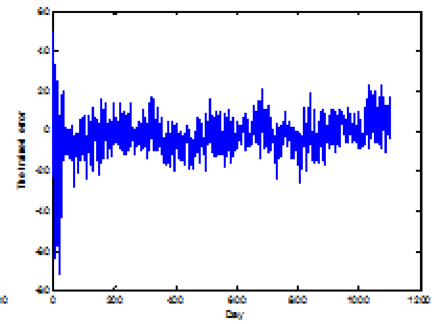


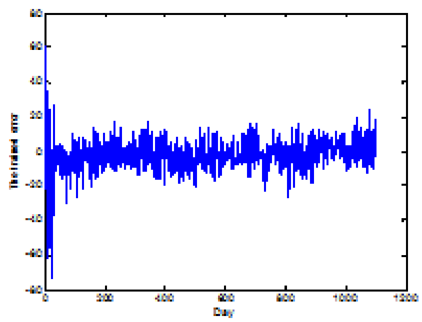
(d) Trained output by IWOA-EDIW (e) Trained output by WOA (f) Trained output by PSO

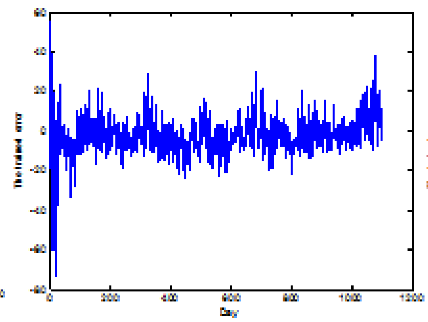
Fig. 9. The trained output curve by IWOAs,WOA, and PSO for the AQI of Taiyuan


(a) Trained error by IWOA-CIW


(b) Trained error by IWOA-LDIW


(c) Trained error by IWOA-SFIW


(d) Trained error by IWOA-EDIW


(e) Trained error by WOA

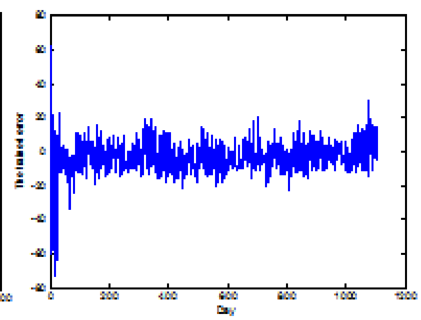

(f) Trained error by PSO

Fig. 10. Trained absolute errors between the trained outputs and the actual valued by IWOAs,WOA and PSO.

Table 8. The values of MSE,RMSE, and MAPE(%) of trained output by IWOAs, WOA, and PSO for AQI of Taiyuan

Error	IWOA-CIW	IWOA-LDIW	IWOA-SFIW	IWOA-EDIW	WOA	PSO
MSE($*\varepsilon^2$)	0.9905	0.8447	0.8234	0.8277	1.0135	0.8942
RMSE	0.0104	0.0103	0.0111	0.0110	0.0139	0.0135
MAPE(%)	7.7567	7.6325	7.8525	7.8694	8.9627	8.7750

Table 9. The predicted outputs by using IWOAs, WOA,and PSO for AQI of Taiyuan

Day	Actual	IWOA-CIW	IWOA-LDIW	IWOA-SFIW	IWOA-EDIW	WOA	PSO
1	143	157.2687	153.9617	148.1389	153.0401	149.6477	151.5613
2	133	144.0815	141.1212	137.4794	140.1434	138.4456	138.6943
3	139	142.4272	140.1033	137.1792	139.6982	136.9146	137.8110
4	94	102.2962	97.6425	96.7131	95.3261	99.0838	93.7954
5	208	209.3517	207.7187	207.3221	206.9071	209.6435	205.8379
6	193	182.2236	180.7762	183.7089	180.4548	182.9695	177.9846
7	377	365.6333	372.0338	380.3302	377.7501	368.9741	371.7747
8	156	135.1924	134.8706	139.2012	136.2761	133.2854	131.8705
9	201	178.4328	178.4492	182.2282	180.6347	174.7013	175.2570
10	196	188.2178	188.2976	190.6199	190.0495	185.1273	185.8449
11	201	198.7698	197.5373	194.9014	198.0620	193.6999	195.5587
12	248	243.8703	241.8791	240.8426	242.4487	238.6637	238.7176
13	250	245.8880	244.0664	244.1924	244.6814	241.6229	240.8397
14	229	225.3582	223.0888	224.2850	223.1567	222.4431	219.6051
15	174	167.9413	164.8724	166.3462	164.2483	164.6581	160.8428
16	173	152.9039	153.4943	157.9378	155.5422	150.8027	150.8884
17	58	56.9036	54.97952	54.6615	55.1679	50.9397	52.1687
18	117	126.4711	121.7301	118.9606	120.2806	119.1360	117.5390
19	150	150.2562	147.0683	146.4567	146.6440	144.3212	143.3140
20	240	224.2156	224.4810	226.5871	226.1980	221.6362	222.0375
21	137	131.8470	132.3340	131.6138	134.1950	126.1813	130.8541
22	46	37.05645	36.1308	35.7904	35.8148	35.1595	34.7713

Table 10.The values of MSE,RMSE, and MAPE(%) of predicted output by IWOAs, WOA, and PSO for AQI of Taiyuan

Error	IWOA-CIW	IWOA-LDIW	IWOA-SFIW	IWOA-EDIW	WOA	PSO
MSE($*e^2$)	1.1802	1.0784	0.7223	0.9029	1.4044	1.4673
RMSE	0.0058	0.0055	0.0044	0.0050	0.0073	0.0073
MAPE(%)	5.8378	5.5963	4.7705	5.0582	6.6213	6.4052

VI. Conclusion

This study introduces the inertia weight to whale optimization algorithm (WOA) by the hunting behavior of humpback whales. Thus the improved whale optimization algorithm (IWOA) is obtained. According to four different inertia weights, the corresponding IWOA becomes IWOA-CIW, IWOA-LDIW, IWOA-SFIW, and IWOA-EDIW, respectively.

We conducted the proposed IWOAs on 27 mathematics benchmark functions to analyze exploration,exploitation,local optima avoidance and convergence behavior by comparison with WOA,FOA,ABC,and PSO. IWOAs were found to be enough competitive.

At the same time, we found that FOA and ABC were inferior to IWOAs,WOA,and PSO. Therefore, we only applied IWOA,WOA,and PSO for AQI prediction of Taiyuan. The results obtained from MSE,RMSE and MAPE were shown that IWOAs with inertia weights are superior to WOA and PSO and were very competitive for applications.

We also improve whale optimization algorithm and apply it for different regions.

References

[1] 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.

[2] L.H. Wu, C.L. Zuo, H.Q. Zhang,“ A cloud model based fruit fly optimization algorithm, ” *Knowledge-Based Systems*, vol.89, pp. 603-617 , 2015.

[3] Q.K. Pan, H.Y. Sang, J.H. Duan, L. Gao, “An improved fruit fly optimization algorithm for continuous function optimization problems, ”*Knowledge-Based Systems*,vol.62,pp.69-83,2014 .

[4] L. Wang, R. Liu, S. Liu, “An effective and efficient fruit fly optimization algorithm with level probability policy and its applications, ”*Knowledge-Based Systems*, vol.97, pp.158-174, 2016.

[5] D. Karaboga, “An idea based on honey bee swarm for numerical optimization,”Technical Report T R06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[6] X.N.Li,G.F.Yang,“Artificial bee colony algorithm with memory, ”*Applied Soft Computing*, vol.41,pp.362-372,2016.

[7] K.Z. Gao, P.N. Suganthan, T.J. Chua,C.S. Chong,T.X. Cai,Q.K.Pan, “A two-stage artificial bee colony algorithm scheduling flexible job-shop scheduling problem with new job insertion,” *Expert Systems with Applications*, vol.42,pp.7652-7663,2015.

[8] S.K. Nseef, S. Abdullah, A. Turky, G. Kendall, “An adaptive multi-population artificial bee colony algorithm for dynamic optimization problems, ”*Knowledge-Based Systems*, vol.104 ,pp.14-23,2016.

[9] D. Karaboga, “Artificial bee colony algorithm,” *Scholarpedia*, vol. 5,no.3,pp.6915,2010.

[10] H.T. Jadhav, P.D. Bamane, “Temperature dependent optimal power flow using g-best guided artificial bee colony algorithm,” *Electrical Power and Energy Systems*, vol.77,pp.77-90,2016.

[11] R.Eberhart, J.Kennedy, “New optimizer using particle swarm theory,” *in Proceedings of the 6th International Sympo-siumon Micro Machine and Human Science*,pp.39－43,Nagoya,Japan,October1995.

[12] Y.Shi,R.Eberhart,“Amodi□edparticleswarmoptimizer,”in Proceedings of the 1998 IEEE International Conferenceon Evolutionary Computation(ICEC’98),pp.69－73,May1998.

[13] A.Chatterjee,P.Siarry,”Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization,” *Computers and Operations Research*,vol.33,no.3,pp.859－871,2006.

[14] K.Lei,Y.Qiu,Y.He,“A new adaptive well-chosen inertia weight strategy to automatically harmonize global and local search ability in particle swarm optimization, ”in: First International Symposiumon Systems and Controlin Aerospace and stronautics, 2006. ISSCAA2006, IEEE, Harbin,p.4, 2006.

[15] H. Mashhadban, S.S. Kutanaei, M.A. Sayarinejad, “Prediction and modeling of mechanical properties in fiber reinforced self-compacting concrete using particle swarm optimization algorithm and artificial neural network, ”*Construction and Building Materials,* vol.119,pp.277-287,2016.

[16] Y.J. Zhang,Y. Zhao, X.H. Fu,, J.R. Xu, “A feature extraction method of the particle swarm optimization algorithm based on adaptive inertia weight and chaos optimization for Brillouin scattering spectra, ”*Optics Communications*,vol.376,pp.56-66,2016.

[17] J.N. Lu, H.P. Hu, Y.P. Bai, “Radial Basis Function Neural Network Based on an Improved Exponential Decreasing Inertia Weight-Particle Swarm Optimization Algorithm for AQI Prediction, ”*Abstract and Applied Analysis*,2014. doi:org/10.1155/2014 /17813.

[18] S. Chen,X. Hong,B.L. Luk, C.J. Harris,“Non-linear system identification using particle swarm optimization tuned radial basis function models,”*International Journal of Bio-Inspired Computation,* vol.1,no.4,pp.246-258,2009.

[19] D. Wu, K. Warwick,Z. Ma,et al.,“Prediction of parkinson’s disease tremor onset using a radial basis function neural network based on particle swarm optimization,”*International Journal of Neural Systems*,vol.20,no.2,pp.109-116,2010.

[20] P. Sitarz, B. Powa ka, “Modal parameters estimation using ant colony optimization algorithm, ”*Mechanical Systems and Signal Processing*,vol.76-77,pp.531-554,2016.

[21] S. Tabakhi, P. Moradi, “Relevance-redundancy feature selection based on ant colony optimization, ”*Pattern Recognition*, vol.48, pp.2798-2811, 2015.

[22] Y.W. Tun, D.M. Pedroso, A. Scheuermann, D.J. Williams, “Probabilistic reliability analysis of multiple slopes with genetic algorithms, ”*Computers and Geotechnics*,vol.77, pp.68-76,2016.

[23] E. Roghaniana, P. Pazhoheshfar, “An optimization model for reverse logistics network under stochastic environment by using genetic algorithm,” *Journal of Manufacturing Systems,* vol.33, pp.348-356, 2014.

[24] M. Hellwig,H.G. Beyer, “Mutation strength control via meta evolution strategies on the ellipsoid model,”*Theoretical Computer Science,* vol.623,pp.160-179,2016.

[25] J. Barrachina,P. Garrido, M. Fogue, F.J. Martinez, J.C. Cano,C.T. Calafate,P. Manzoni,“Reducing emergency services arrival time by using vehicular communications and Evolution Strategies,”*Expert Systems with Applications*,vol.41,pp.1206-1217,2014.

[26] A.O. Kusakci, M.Can,“An adaptive penalty based covariance matrix adaptation-evolution strategy,”*Computers & Operations Research*, vol.40,pp.2398-2417,2013.

[27] S.C. Zhou, M. Liu, H.P. Chen, X.P. Li, “An effective discrete differential evolution algorithm for scheduling uniform parallel batch processing machines with non-identical capacities and arbitrary job sizes, ”*Int. J.Production Economics*,vol.179,pp.1-11,2016.

[28] Y. Chen, W.C. Xie, X.F. Zou, “A binary differential evolution algorithm learning From explored solutions, ”*Neurocomputing*, vol.149, pp.1038－1047, 2015.

[29] S. Mirjalili, A. Lewis,“The Whale Optimization Algorithm, ”*Advances in Engineering Software*,vol.95,pp.51–67,2016.

[30] P.R. Hof, E. Van Der Gucht, "Structure of the cerebral cortex of the humpback whale, Megaptera novaeangliae(Cetacea, Mysticeti, Balaenopteridae), "*Anat Rec*,vol.290,pp.1-31,2007.

[31] W.A. Watkins, W.E. Schevill, "Aerial observation of feeding behavior in four baleen whales: Eubalaena glacialis, Balaenopter a borealis, Megapter a novaeangliae, and Balaenoptera physahis, "*J.Mammal*, vol.1, pp.55-63,1979.

Hongping Hu was born in July 1973. She is an associate professor and a master tutor in Department of Mathematics, North University of China. She received her Ph.D. degree from North University of China in 2009. Her research interests include combinatorial mathematics, artificial intelligence and image processing on applied mathematics.

Yanping Bai was born in July 1962. She is a professor and a doctoral tutor in School of Information and Communication Engineer, North University of China. She received her Ph.D. degree from North University of China in 2005. Her research interests include optimization algorithm, artificial intelligence, signal processing, image processing and MEMS reliability research.

Ting Xu was born in April 1980. She is a Ph.D. student in School of Information and Communication Engineer, North University of China. She is also a lecturer in Department of Mathematics, North University of China. Her current research interests include artificial intelligence, pattern recognition and data mining.