electronics

Article

An Improved Whale Optimizer with Multiple Strategies for Intelligent Prediction of Talent Stability

Hong Li [1], Sicheng Ke [1], Xili Rao [1], Caisi Li [1], Danyan Chen [1], Fangjun Kuang [2,*], Huiling Chen [3,*], Guoxi Liang [4,*] and Lei Liu [5]

[1] Wenzhou Vocational College of Science and Technology, Wenzhou 325006, China
[2] School of Information Engineering, Wenzhou Business College, Wenzhou 325035, China
[3] College of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou 325035, China
[4] Department of Artificial Intelligence, Wenzhou Polytechnic, Wenzhou 325035, China
[5] College of Computer Science, Sichuan University, Chengdu 610065, China
* Correspondence: kfj@wzbc.edu.cn (F.K.); chenhuiling.jlu@gmail.com (H.C.); guoxiliang2017@gmail.com (G.L.)

Abstract: Talent resources are a primary resource and an important driving force for economic and social development. At present, researchers have conducted studies on talent introduction, but there is a paucity of research work on the stability of talent introduction. This paper presents the first study on talent stability in higher education, aiming to design an intelligent prediction model for talent stability in higher education using a kernel extreme learning machine (KELM) and proposing a differential evolution crisscross whale optimization algorithm (DECCWOA) for optimizing the model parameters. By introducing the crossover operator, the exchange of information regarding individuals is facilitated and the problem of dimensional lag is improved. Differential evolution operation is performed in a certain period of time to perturb the population by using the differences in individuals to ensure the diversity of the population. Furthermore, 35 benchmark functions of 23 baseline functions and CEC2014 were selected for comparison experiments in order to demonstrate the optimization performance of the DECCWOA. It is shown that the DECCWOA can achieve high accuracy and fast convergence in solving both unimodal and multimodal functions. In addition, the DECCWOA is combined with KELM and feature selection (DECCWOA-KELM-FS) to achieve efficient talent stability intelligence prediction for universities or colleges in Wenzhou. The results show that the performance of the proposed model outperforms other comparative algorithms. This study proposes a DECCWOA optimizer and constructs an intelligent prediction of talent stability system. The designed system can be used as a reliable method of predicting talent mobility in higher education.

Keywords: swarm intelligence; whale optimization algorithm; extreme learning machine; talent stability prediction; machine learning

Citation: Li, H.; Ke, S.; Rao, X.; Li, C.; Chen, D.; Kuang, F.; Chen, H.; Liang, G.; Liu, L. An Improved Whale Optimizer with Multiple Strategies for Intelligent Prediction of Talent Stability. *Electronics* **2022**, *11*, 4224. https://doi.org/10.3390/electronics11244224

Academic Editor: Maciej Ławryńczuk

Received: 18 November 2022
Accepted: 15 December 2022
Published: 18 December 2022

1. Introduction

Talent resources are the core resources on which universities rely for survival and development. A reasonable flow of talent can stimulate the vitality of the organization, improve the quality of talent, form a virtuous cycle and promote the complementary advantages of talent resources among universities. However, the "war for talents" against the background of "double tops" has led to the disorderly and utilitarian flow of talents in colleges and universities, an increase in the introduction to talents, a continuous increase in local competition, an accelerated frequency of talent flow and a structural imbalance of talent flow among colleges and universities in the region. This has engendered many negative effects on the development of universities. Therefore, a reasonable forecast in stable trends of university talent is crucial to the survival and development of universities. However, traditional methods have some limitations on predicting the stability of talent.

It is a new trend to use artificial intelligence algorithms to achieve accurate predictions of talent stability. There have been few studies that have used the artificial intelligence tools to solve the prediction issue of talent stability, so we have summarized some related works which used artificial intelligence tools to tackle the prediction problems for students; this is shown in Table 1.

Table 1. The latest research status of prediction issues for students.

Authors	Methods	Overview
Yang et al. [1]	The theory of planned behavior	They suggested that attitudes, subjective norms, perceived behavior, gender and parental experience have a significant impact on students' entrepreneurial intentions.
Gonzalez-Serrano et al. [2]	Questionnaire method	They demonstrated that attitudes and perceived behaviors were statistically significant.
Gorgievski et al. [3]	Values theory and planned behavior theory	They found a strong link between personal values and entrepreneurial career intentions.
Nawaz et al. [4]	Partial least squares structural equation modeling (PLS-SEM)	They found emotional intelligence, entrepreneurial self-efficacy and self-regulation also directly affect college students' entrepreneurial intentions.
Yang et al. [5]	Decision tree	They extracted four key attributes that affect students' intentions to start a career.
Djordjevic et al. [6]	Data analysis approach	They predicted the entrepreneurial intentions of youth in Serbia based on demographic characteristics, social environment, attitudes, awareness of incentives and environmental assessment.
Wei et al. [7]	Kernel extreme learning machine	They provided a reasonable reference for the formulation of talent training programs and guidance for the entrepreneurial intention of students.
Bhagavan et al. [8]	Data mining tools and methods	They predicted the current performance of students through their early performance and awareness, and identified students' expected abilities.
Huang et al. [9]	Artificial intelligence algorithms and fuzzy logic models	They designed a diversified employment recommendation system, combined with students' personal interests, and provided employment plans.
Li et al. [10]	The cluster analysis technology model	They achieved accurate predictions of the employment situation of graduates.

The swarm intelligence algorithm (SIA) is a crucial optimization method by which to predict traditional talent stability. SIA is derived from natural phenomena or group behaviors, etc., such as group predation and physical phenomena. Optimization principles exist within these phenomena. As a kind of SIA, the whale optimization algorithm (WOA) [11] has a clear algorithm structure and good performance, which was proposed in 2016. It was designed by simulating the hunting behavior of whales. During foraging, the whales use bubbles as tools to surround their prey. Furthermore, the algorithm has been used in many natural science fields, such as shop scheduling problems [12,13] and engineering design problems. Navarro et al. [14] proposed a version of the WOA with the K-means mechanism to explore the algorithm's search space. The proposed model was effective against resolving complex optimization issues. Abbas et al. [15] proposed a combination of the technique of an extremely randomized tree with the WOA for the detection and prediction of medical diseases. Abd et al. [16] introduced a novel WOA version application for multilevel threshold image segmentation. Abdel-Basset et al. [17] presented a new WOA version based on local search mechanisms to optimize a scheduling problem with the multimedia data objects field. Qiao et al. [18] presented a novel version of the WOA, which combined the worst individual disturbance and the neighborhood mutation search strategy for solving engineering design problems. Peng et al. [19] introduced an enhanced WOA, which combined the information-sharing search strategy and the Nelder-Mead simplex strategy, to evaluate the parameters of solar cells and photovoltaic modules. Abderazek et al. [20] presented the WOA and a moth-flame optimizer for optimizing spur gear design.

For the high-quality training of talent, in addition to focusing on the employment and entrepreneurship of university students, the stability of talents is also an important foundation for social and economic development. Employment stability reflects psychological satisfaction with practitioners regarding the employment unit, employment environment, remuneration package and career development. In the past five years, the average turnover rate of several colleges and universities in Wenzhou was 28.1%. An appropriate turnover rate is conducive to the "catfish effect" in enterprises and institutions, and stimulates the vitality and competitiveness of the organization; however, an excessive turnover rate has a negative impact on the human resource costs and economic efficiency of universities, as well as their social reputation and the quality development of the economy and society.

Big data has a wide scope of application in the field of talent mobility management. Through the effective mining of big data onto talent flows in a university, the stability of talent employment is analyzed, and the correlation hypothesis is verified by integrating an intelligent optimization algorithm, neural network, support vector machine and other machine learning methods; an intelligent prediction model is then constructed. At the same time, key factors affecting the stability of talent employment are mined, and the key influencing factors are analyzed in depth to explore the main features affecting the stability of talent employment and to provide reference for government decision-making and policy formulation. The main contributions are shown as bellow:

(1) A multi-strategy hybrid modified whale optimization algorithm is proposed.
(2) Introducing the crossover operator to facilitate the exchange of information and improve the problem of dimensional lag.
(3) DECCWOA is verified on the 35 benchmark functions to demonstrate the optimization performance.
(4) DECCWOA is combined with KELM and feature selection to achieve efficient talent stability intelligence prediction.
(5) Results show the proposed methods surpass other reported approaches.

The remainder of this paper is structured as follows. Section 2 reviews the whale optimization algorithm. Section 3 provides a comprehensive description of the proposed method. The proposed method is verified and applied using benchmark function experiments and feature selection experiments in Section 4. The conclusion and future work are outlined in Section 5.

2. Relate Work

In recent years, swarm intelligence optimization algorithms have emerged, such as the Runge Kutta optimizer (RUN) [21], the slime mold algorithm (SMA) [22], the Harris hawks optimization (HHO) [23], the hunger games search (HGS) [24], the weighted mean of vectors (INFO) [25], and the colony predation algorithm (CPA) [26]. Moreover, they have achieved very good results in many fields, such as feature selection [27,28], image segmentation [29,30], bankruptcy prediction [31,32], plant disease recognition [33], medical diagnosis [34,35], the economic emission dispatch problem [36], robust optimization [37,38], expensive optimization problems [39,40], the multi-objective problem [41,42], scheduling problems [43–45], optimization of a machine learning model [46], gate resource allocation [47,48], solar cell parameter identification [49] and fault diagnosis [50]. In addition to the above, the whale optimization algorithm (WOA) [11] is an optimization algorithm simulating the behaviors of whales rounding up their prey. During feeding, whales surround their prey in groups and move in a whirling motion, releasing bubbles in the process, and thus, closing in on their prey. In the WOA, the feeding process of whales can be divided into two behaviors, including encircling prey and forming bubble nets. During each generation of swimming, the whale population will randomly choose between these two behaviors to hunt. In d-dimensional space, suppose that the position of each individual in the whale population is expressed as $X = (x1,\ x2,\ \ldots,\ xD)$.

Agrawal et al. [51] proposed an improved WOA and applied it to the field of feature selection [52]. Bahiraei et al. [53] proposed a novel perceptron neural network, which

combined the WOA and other algorithms, and was applied to the field of polymer materials. Qi et al. [54] introduced a new WOA with a directional crossover strategy, directional mutation strategy, and levy initialization strategy. The potential for using the suggested approach to address engineering issues is very high. Bui et al. [55] proposed a neural-network-model-based WOA, which also integrated a dragonfly optimizer and an ant colony optimizer, and was applied to the construction field. Butti et al. [56] presented an effective version of the WOA to optimize the stability of power systems. Cao et al. [57] also proposed a new WOA to improve the efficiency of the proton exchange of membrane fuel cells. Cercevik et al. [58] presented an optimization model, combined with the WOA and others, to improve the parameters of seismic isolated structures. Zhao et al. [59] presented a susceptible-exposed-infected-quarantined (hospital or home)-recovered model based on the WOA and human intervention strategies to simulate and predict recent outbreak transmission trends and peaks in Changchun. A brand-new hybrid optimizer was developed by Fan et al. [60] to solve large-scale, complex practical situations. The proposed hybrid optimization algorithm combined a fruit flew optimizer with the WOA. Raj et al. [61] proposed the application of the WOA as a solution to reactive power planning with flexible transmission systems. Guo et al. [62] proposed an improved WOA with two strategies to improve the exploration and exploitation abilities of the WOA, including the random hopping update mechanism and random control parameter mechanism. To improve the algorithm's convergence rate and accuracy, a new version of the WOA was presented by Jiang et al. [63] to apply constraints to engineering tasks.

Although the WOA has obtained good results in many fields, the algorithm easily falls into the local optimum in the face of complex problems. Therefore, many excellent improvement algorithms have been proposed. For example, Hussien et al. [29] proposed a novel version of the whale optimizer with the gaussian walk mechanism and the virus colony search strategy to improve convergence accuracy. To solve the WOA's susceptibility to falling into the local optimum with slow convergence speeds, an improved WOA with a communication strategy and the biogeography-based model was proposed by Tu et al. [64]. Wang et al. [65] presented a novel-based elite mechanism WOA, with a spiral motion strategy to improve the original algorithm. Ye et al. [49] introduced an enhanced WOA version of the levy flight strategy and search mechanism to improve the algorithm's balance. Abd et al. [66] presented an innovative method to enhance the WOA, including the differential evolution exploration strategy. Abdel-Basset et al. [67] introduced an enhanced whale optimizer, which was combined with a slime mold optimizer to improve the performance of the algorithm. To enhance the WOA's search ability and diversity, a novel version of the WOA with an information exchange mechanism was proposed by Chai et al. [68]. Heidari et al. [69] presented a whale optimizer with two strategies, including an associative learning method and a hill-climbing algorithm. Jin et al. [70] proposed a dual operation mechanism based on the WOA to solve the slow convergence speed problem. Therefore, the WOA is an effective optimizer by which to improve the performance of traditional talent stability prediction.

3. Materials and Methods

This section will improve the problems existing in the traditional whale optimization algorithm, so as to propose a new version of the algorithm. During the process of the whale population continuously approaching the optimal position, the population appears in an aggregation state, which is the main reason for the algorithm falling into the local optimal. Based on this, the DE operation is performed on the whale population during a certain period, and the whale population is disturbed by the differential information of multiple individuals, so as to ensure the diversity of the population. By introducing the idea of a crisscross optimization algorithm, a vertical crossover is performed in dimensions to improve dimensional stagnation as iterations progress, and horizontal crossover is performed between individuals to fully facilitate the exchange of information between individuals, allowing the problem space to be fully searched, effectively improving the

search capability of the algorithm. Overall, the proposed algorithm is named as the DE-based crisscross whale algorithm (DECCWOA).

3.1. Whale Optimization Algorithm

3.1.1. Encircling Prey

In the process of encircling the prey, each individual will choose the position closest to the prey in the group, that is, the global optimal solution, or will randomly select a whale and approach it. The equation for updating the position of the whale is shown in Equation (1).

$$X_i^{t+1} = X_{best}^t - A\left| C \times X_q^t - X_{it}^t \right| \tag{1}$$

where X_q^t is X_{best}^t when the whale swims toward the optimal whale position, and X_{rand}^t when the whale swims toward the random whale position. A is a random number with a uniform distribution between $(-a,\ a)$, and the initial value of a is 2, which linearly decreases to 0 with the number of iterations. C is a random number that satisfies the uniform distribution, and its value is between (0, 2). The choice of whether the whale individual swims toward the optimal whale or random position is up to the value of A. When $|A < 1|$, the whale decides to swim toward the optimal individual; otherwise, the whales will select a random location in the population and approach it.

3.1.2. Forming Bubble Nets

Whales release bubbles while hunting, thus forming a spiraling, blistering net to repel the prey. If bubble feeding is chosen, the whale first calculates the distance between itself and the best whale, then swims upwards in a spiral and spits out bubbles of varying sizes to feed on the fish and prawns. At this point, the position of the whale is updated by the equation shown in Equation (2).

$$X_i^{t+1} = \left|X_{best}^t - X_i^t\right| \times e^{bl} \times cos(2\pi l) + X_{best}^t \tag{2}$$

where b is a constant, and l is a random number between $[-1, 1]$, meeting a uniform distribution.

3.2. Differential Evolution Algorithm (DE)

The differential evolution algorithm (DE) [71] was proposed in 1997 based on the idea of evolutionary algorithms, such as genetic algorithms, which are essentially multi-objective optimization algorithms that can be used to solve the overall optimal solution in a multi-dimensional space. The DE is the same as other genetic algorithms in that the main process consists of three steps: mutation, crossover and selection. However, the variance vector of the differential DE is generated from the parent differential vector and is crossed with the parent individual vector to generate a new individual vector, which is directly selected with its parent individual. Suppose the position vector of the i-th individual in the population is X_i.

3.2.1. Crossover Operations

The basic variance vector is generated by Equation (3), and $r_1 \neq r_2 \neq r_3$. Therefore, in the DE algorithm, the population must be greater than 3. F is the crossover operator, with a value usually between [0, 2], which controls the amplification of the deviation vector. Commonly, the difference between the two vectors is multiplied by the crossover operator and added to the third vector to generate a new mutation vector.

$$X_i = X_{r_1} + F \times (X_{r_2} - X_{r_3}) \tag{3}$$

In this article, in order to allow for faster convergence of the population algorithm while maintaining population diversity, we attempt to calculate the difference between the position of the current population and the optimal population position (X_{best}), on the basis

of which a new variant population is generated. Therefore, Equation (3) is rewritten as shown in Equation (4).

$$V_i = X_i + F \times (X_{best} - X_i) \tag{4}$$

3.2.2. Mutation Operations

To increase the diversity of the interference vectors, crossover operations are introduced. Equation (5) presents the principle of the crossover operation.

$$U_{i,j} = \begin{cases} V_{ji} & if\ randb(j) \le CR\ or\ j = rnbr(i) \\ X_{i,j} & if\ randb(j) > CR\ or\ j \ne rnbr(i) \end{cases},\ i = 1,\ 2,\ \ldots,\ NP;\ j = 1,2,\ \ldots,\ D \tag{5}$$

$randb(j)$ denotes the generation of the j-th estimate of a random number between [0, 1] and $rnbr$ denotes a randomly chosen sequence. CR is the crossover operator. In simple terms, if the randomly generated $randb(j)$ is less than CR or $j = r$, then the variant population is placed in the selection population; if not, the original population is placed in the selection population.

3.2.3. Selection Operation

In order to decide whether the vectors in the selection population can become part of the next generation, the newly generated position vectors are compared with the current target vectors, and, if it appears that the objective function is further optimized or the original state is maintained, then, the newly generated individuals will appear in the next generation. The selection operation is defined as shown in Equation (6).

$$X_i = \begin{cases} U_i & if\ f(U_i) < f(X_i) \\ X_i & if\ f(U_i) \ge f(X_i) \end{cases} \tag{6}$$

The DE is a simple and easy-to-implement algorithm that mainly performs genetic operations by means of differential variation operators. The algorithm has shown good robustness and efficiency in solving most optimization problems [72–75]. Furthermore, the algorithm is intrinsically parallel and can coordinate searches, so that the DE has a faster convergence rate for the same requirement.

3.3. Crisscross Optimization Algorithm

The crisscross optimization algorithm (CSO) [76] is a new population-based stochastic search algorithm that performs both horizontal and vertical crossover in each generation during each iteration, thus allowing certain dimensions of the population that are trapped in a pseudo-optimal a chance to jump out. The new individuals obtained after each crossover need to go through competition, and only the individuals better than the parent generation will be retained for the next iteration.

3.3.1. Horizontal Crossover Operator

A horizontal crossover operation is similar to crossover operations in genetic algorithms, a kind of arithmetic crossover between the same dimension of two different individual particles in a population. Assuming a horizontal crossover in the d-th dimension for the i-th and j-th parent individual particles, the formula for generating offspring is shown in Equations (7) and (8).

$$MS_{hc}(i,d) = r_1 \times X(i,\ d) + (1 - r_1) \times X(j,\ d) + c_1 \times (X(i,\ d) - X(j,\ d)) \tag{7}$$

$$MS_{hc}(j,d) = r_1 \times X(j,\ d) + (1 - r_1) \times X(i,\ d) + c_1 \times (X(j,\ d) - X(i,\ d)) \tag{8}$$

where r_1 and r_2 are random numbers between [0, 1], and, c_1 and c_2 are random numbers between $[-1, 1]$. $X(i,\ d)$ and $X(j,\ d)$ represent the d-th dimension of the i-th and j-th individuals in the population, respectively. $MS_{hc}(i,d)$ and $MS_{hc}(j,d)$ are the d-th dimension of the offspring generated by $X(i,\ d)$ and $X(j,\ d)$ via horizontal crossover, respectively. From a so-

ciological point of view, $r_1 \times X(i,\ d)$ is the memory term of particle $X(i)$. $(1 - r_1) \times X(j,\ d)$ is the group cognitive term of particles $X(i)$ and $X(j)$, representing the interaction between different particles. c_1 is the learning factor, $c_1 \times (X(i,\ d) - X(j,\ d))$ can effectively enlarge the search interval and search for optimization at the edge. The schematic diagram of the horizontal crossover operation is shown in Figure 1.

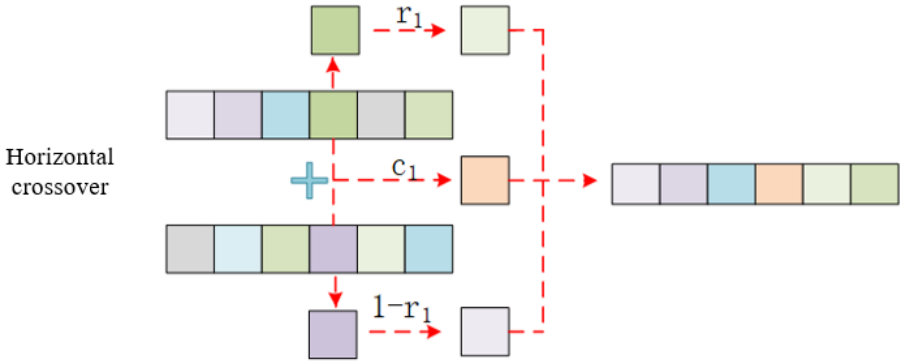

Figure 1. Schematic of horizontal crossover.

3.3.2. Vertical Crossover Operator

A vertical crossover is an arithmetic crossover between two different dimensions of a particle in a population. Since different dimensional elements have different ranges of values, the two dimensions need to be normalized before crossover. Furthermore, in order to allow the dimension that has stalled in the local optimum to jump out of the local optimum without destroying the information of the other dimension, only one child particle is generated for each vertical crossover operation, and only one of the dimensions is updated. The vertical crossover operation is defined by Equation (9).

$$MS_{vc}(i,\ d_1) = r \times X(i,\ d_1) + (1 - r) \times X(i,\ d_2),\ i \in\ N(1,\ M),\ d_1\ d_2\ \in\ N(1,\ D) \quad (9)$$

where r is a random number between [0, 1]. $MS_{vc}(i,\ d_1)$ is the d_1-th dimension of the offspring produced by the d_1-th and d_2-th dimensions of individual $X(i)$ by vertical crossover. The new individual contains not only the information of the d_1-th dimension of the parent particle, but also the information of the d_2-th dimension with a certain probability, and the information of the d_2-th dimension will not be destroyed during the crossover. A schematic diagram of the vertical crossover is shown in Figure 2.

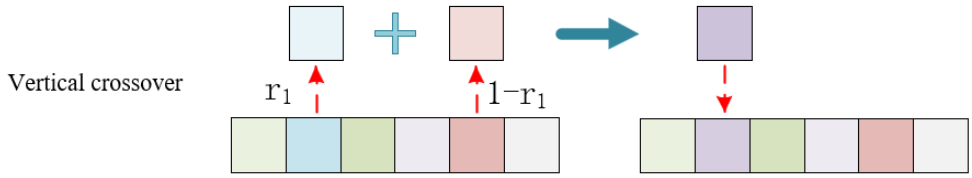

Figure 2. Schematic of vertical crossover.

3.4. Framework of Proposed DECCWOA

The whale algorithm, the crossover and mutation operations in the DE and the crisscross operators together form the overall framework of the DECCWOA. We consider a positive population renewal to be complete when a location closer to a food source is found in one iteration. When the entire whale population has completed S positive updates, we consider the population to have been concentrated and to be losing population diversity. In one iteration, after the whales have completed one location update, it is determined whether the population has completed S positive updates, and, if so, the crossover and mutation operations of the DE algorithm are performed, resulting in a perturbation of the whale population, further ensuring population quality. Moreover, vertical crossover is

performed in dimensions to improve dimensional stagnation as iterations progress, and, when the entire population has completed one location update, horizontal crossover is performed between individuals to fully facilitate the exchange of information between individuals, allowing the problem space to be fully searched, effectively improving the search capability of the algorithm. The pseudo-code of the DECCWOA can be seen in Algorithm 1, and a flow chart of the overall DECCWOA framework is shown in Figure 3.

Algorithm 1: The pseudo-code of the DECCWOA

Input: Number of populations N, maximum number of iterations T, , objective function $fobj$;
Output: Optimum whale position X_{best};

```
Initialize the whale population positions X;
Calculate fitness values for all individuals in the whale population and sort them;
Set the position of the individual with the smallest fitness value f_best to X_best;
Set s = 0;
while (t < T)
for each agent
    Update a, A, C l, S and p;
    if p < 0.5
      if |A| < 1 && s < S
          Update the position of agent using Equation (1), and set X_q as X_best;
      elseif |A| ≥ 1 && s < S
          Select a random search agent as X_rand;
          Update the position of agent using Equation (1), and set X_q as X_rand;
      elseif s ≥ S
          Performing crossover and mutation operations in DE;
      end if
    else
      Update the position of agent using Equation (2);
    end if
    Perform vertical crossover operator using Equation (9);
  end for
  Perform horizontal crossover operation using Equations (7) and (8);
  Calculate fitness values for all individuals in the whale population and sort them;
  Set the position of the individual with the smallest fitness value g_best to Xg_best;
  if g_best < f_best
   Updates X_best and f_best;
  s = s + 1;
  end if
  t = t + 1;
end while
Return X_best
```

In the basic whale algorithm, only each individual in the population is updated according to the corresponding situation in each iteration, excluding other complex operations. Therefore, the time complexity of the algorithm is only related to the maximum number of iterations T and the population size N; that is, the time complexity of the whale algorithm is $O(T * N)$. When executing the vertical crossover algorithm, the time complexity of the vertical crossover is O(D); a vertical crossover is performed at the end of each individual update as the vertical crossover occurs in dimension D. When the horizontal crossover is executed after the whole population has been updated, the time complexity of the horizontal crossover is $O(N * D)$ depending on the size of the individuals and the dimension of the problem, as the horizontal crossover is performed by communicating between individuals and updating the dimensional information in turn. In DE, a crossover is performed, and the mutation and selection operations are only related to dimensions, so the time complexity of an iteration is $O(D)$. In this work, only when the position of the population is updated every time and a certain period is met, we carry out an operation of crossover, mutation

and selection for the population. Therefore, the operation of theoretically introducing the DE does not add a high time cost to the algorithm. In summary, the time complexity of the proposed algorithm DECCWOA is $O(T * (O(N * D) + O(N)))$.

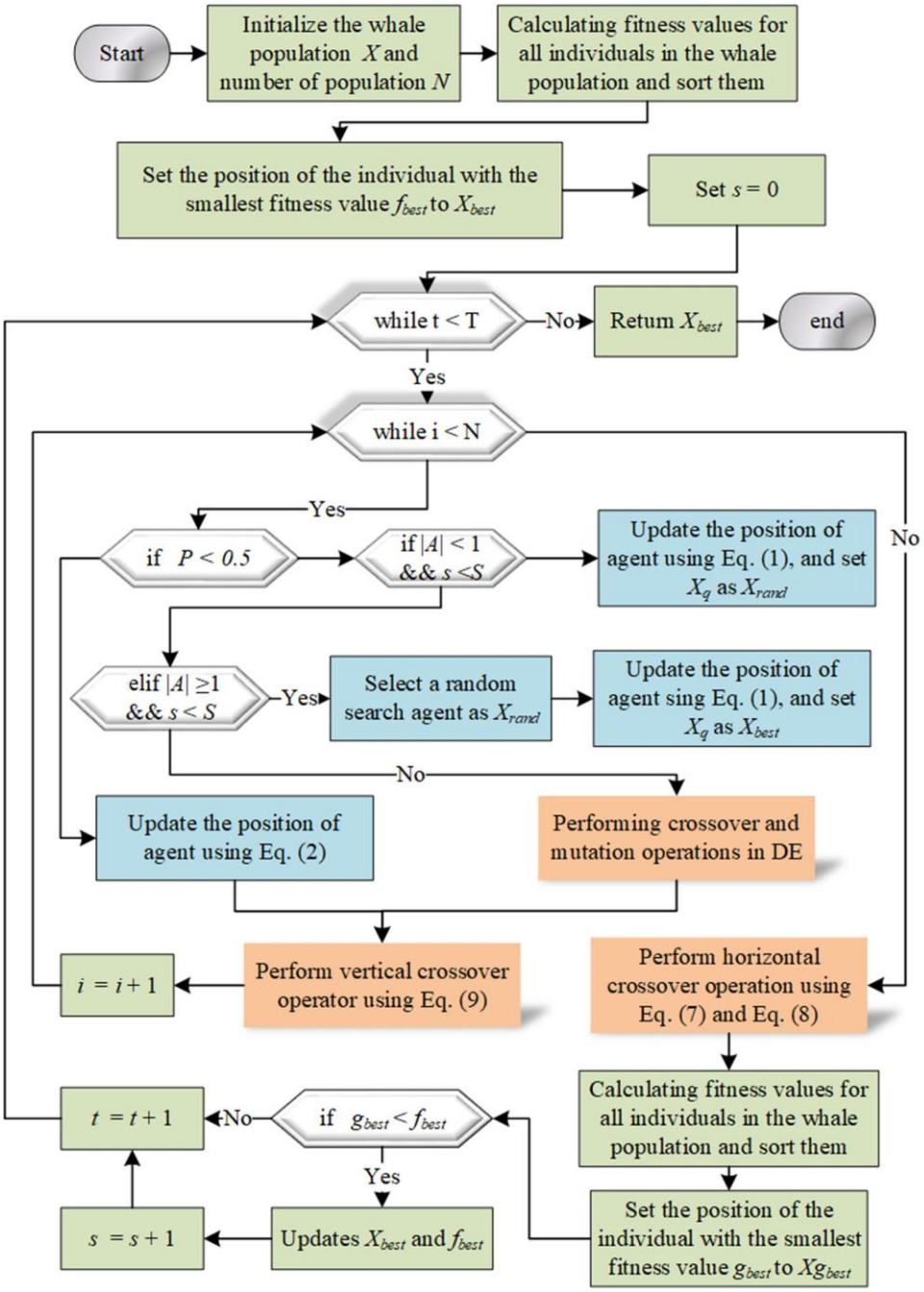

Figure 3. Flow chart of the DECCWOA framework.

4. Experimental Results

This section presents a quantitative analysis of the introduced DE and CSO mechanisms and presents the experimental results comparing the proposed algorithm, DECCWOA, with other improved WOA algorithms and improved swarm intelligence algorithms that have better performance on 35 benchmark functions. Furthermore, to show that the proposed algorithm is still valid for practical applications, the DECCWOA is applied to the intelligent prediction of talent stability in universities. All experiments were carried

out on a Windows Server 2012 R2 operating system with Intel(R) Xeon(R) Silver 4110 CPU (2.10 GHz) and 32.GB RAM. All algorithms were coded and run on MATLAB 2014b.

To ensure fairness of the experiment, all algorithms were executed in the same environment. For all algorithms, the population size was set to 30, the maximum number of function evaluations was set to 300,000 and, to avoid the effect of randomness on the results, each algorithm was individually executed 30 times on each benchmark function. *avg* and *std* reflect the average ability and stability of each algorithm after 30 independent experiments. To allow a more visual presentation of the average performance of all the algorithms, the Freidman test is used to evaluate the experimental results of all algorithms on the benchmark function and the final ranking is recorded.

4.1. Experimental Results of the DECCWOA on Benchmark Functions

The DECCWOA and its related comparison algorithm conducted comparison experiments on 35 benchmark functions selected from 23 benchmark functions and CEC2014. In detail, Table A1 of the Appendix A shows a summary of the 35 test functions, which can be divided into three categories, including unimodal functions, multimodal functions and hybrid functions.

4.1.1. Parameter Sensitivity Analysis

Not every dimension of an individual is selected for crossover in a vertical crossover operation. In the vertical crossover operation, there is a key parameter p_2. When the random probability is less than p_2, the crossover operation is performed in the corresponding dimension of the individual, as shown in Equation (9). Otherwise, the operation is considered not to be performed in that dimension. The possible values of p_2 are 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1.0. In order to visually present the impact of p_2 on the optimization capabilities of the DECCWOA, we conducted comparative experiments using different versions. The names corresponding to the different algorithm versions are shown in Table 2.

Table 2. Names of different algorithm versions when p_2 is different.

Algorithm	DEC CWOA1	DEC CWOA2	DEC CWOA3	DEC CWOA4	DEC CWOA5	DEC CWOA6	DEC CWOA7	DEC CWOA8	DEC CWOA9	DEC CWOA10
p_2	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0

Different values of p_2 have a direct impact on the optimization of the DECCWOA. Table 3 shows the results of the DECCWOA2 and Table A2 in Appendix A shows the detailed results when p_2 is taken to different values. The rankings generated by the Friedman test show that when the value of p_2 is too large, the less effective the average optimization is. The significance of introducing a longitudinal crossover operator is to help the population change dimensional stagnation. That is because when p_2 takes a larger value, it means that each dimension of the individual changes with a high probability. This not only changes the dimension of stagnation, but also the dimension of having good performance along with it. Notably, when p_2 is taken as 0.1 versus 0.2, the performance is similar for 28 of the 35 benchmark functions, but, for overall performance, the DECCWOA2 is slightly better. This is because, when the value of p_2 is too small, only a few dimensions are adjusted after the individual enters the vertical crossover operator, which does not have the problem of falling into local optima due to dimensional stagnation being significantly improved, especially in solving multimodal functions and hybrid functions. Therefore, in the course of the next experiments, p_2 was set to 0.2.

Table 3. Experimental results for analysis of different versions.

Algorithms	Overall	
	+/−/=	Rank
DECCWOA1	~	2
DECCWOA2	4/3/28	1
DECCWOA3	14/3/18	6
DECCWOA4	12/2/21	5
DECCWOA5	14/4/17	3
DECCWOA6	14/5/16	4
DECCWOA7	14/4/17	7
DECCWOA8	15/4/16	8
DECCWOA9	17/2/16	9
DECCWOA10	16/2/17	10

4.1.2. Comparison of Mechanisms

In order to verify the effectiveness of the introduced mechanism in improving the optimization capabilities of the WOA, ablation studies on the integrated DE and CSO were conducted. Table 4 presents the comparison results for the introduced mechanisms. The detailed results can be found in Table A3 of Appendix A. Notably, on most of the benchmark functions, the DECCWOA has the best optimization capability by performing the Friedman test on 30 times randomized trials. Furthermore, the CCWOA with the introduction of CSO outperforms the WOA on more than 90% of the benchmark functions. However, for the DEWOA, which introduces the DE into the WOA, although the overall results are not significantly improved, a comparison of the optimization performance of CCWOA and DECCWOA shows that the combination of CSO with the DE makes the WOA more optimizable.

Table 4. Comparison results for the introduced mechanisms.

Algorithms	Overall	
	+/−/=	Rank
DECCWOA	~	1
DEWOA	31/1/3	4
CCWOA	18/4/13	2
WOA	25/1/9	3

Convergence curves of the comparison results for the introduced mechanisms are shown in Figure 4. Among them, the CCWOA excels in both optimization accuracy and convergence speed on F4, F6 (from 23 benchmark functions) and unimodal functions of F14 (from CEC2014), F12, F13 (from 23 benchmark functions), multimodal functions of F18, F19, F23, F24, F29 (from CEC2014) and hybrid functions of F30 and F32. In particular, CCWOA also has stronger search ability in multimodal functions and hybrid functions. This shows that CSO effectively improves the problem that the basic WOA is prone to falling into the local optimum. It is also worth noting that the introduction of the DE did not give the desired results on most of the benchmark functions. However, when acting together with a CSO on the WOA, the convergence speed and optimization accuracy of the DECCWOA are significantly improved. Especially in F4, F6 and F13, it is obvious that the DECCWOA has better performance than the CCWOA. This is because we perform the DE crossover and mutation operations over a period of time in order to take advantage of differences between individuals to disturb the population, but do not perform the rounding up of prey in the basic WOA at this time, thus slowing the efficiency of the whale population towards the food source. However, when CSO is applied to the whole population, not only is the information between individuals utilized, but also the information in the spatial dimension

is considered. Combined with the periodic perturbation of the DE, the whale population can search the whole problem space more efficiently.

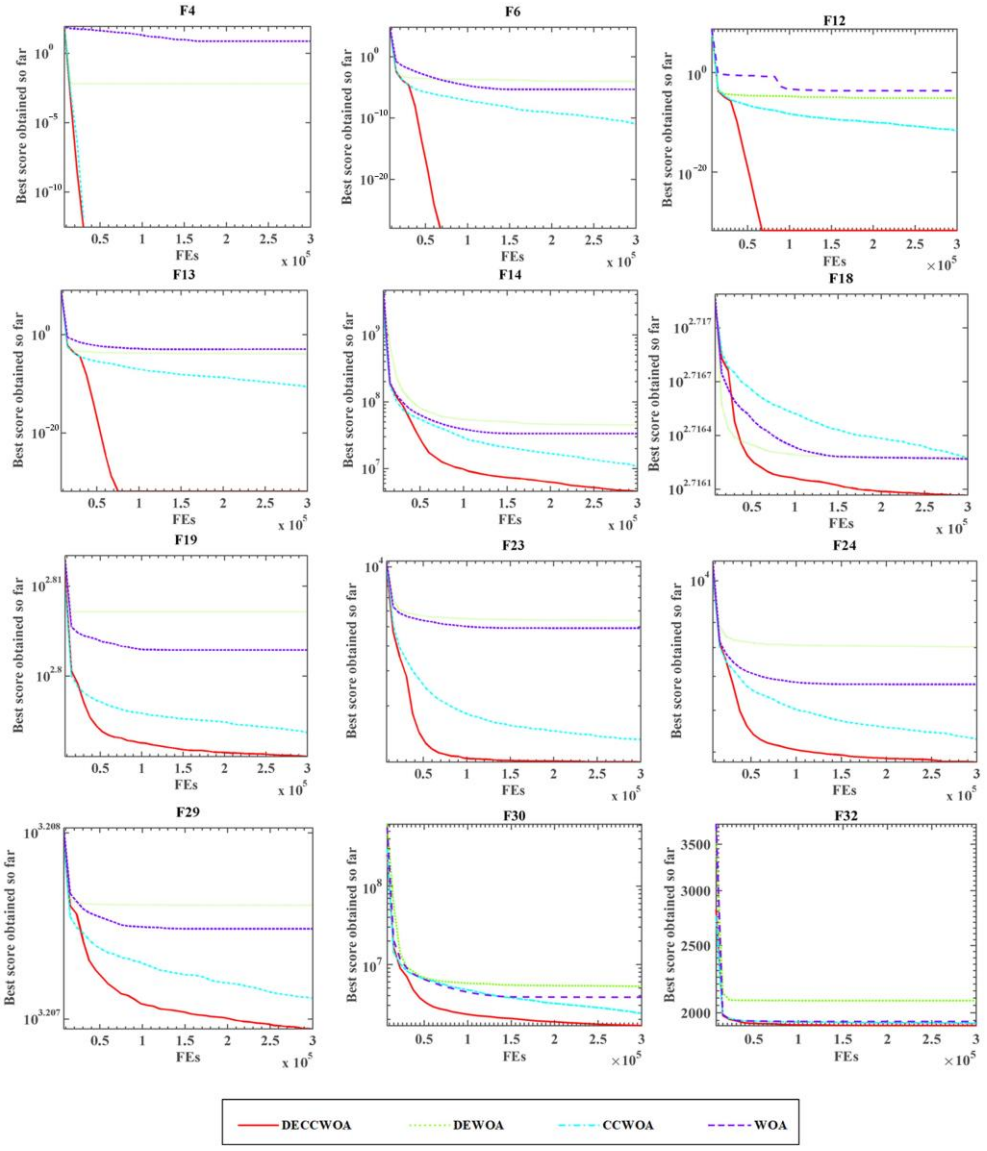

Figure 4. Convergence curves of the comparison results for the introduced mechanisms.

4.1.3. Comparison with Improved WOA Versions

In order to provide a clearer picture of the results of the experiments comparing the DECCWOA with other improved WOA algorithms for 35 benchmark functions, *avg* and *std* of all functions obtained after 30 independent experiments on the corresponding benchmark functions and the average ranking results of the Friedman test on the average results are recorded in Table 5. The detailed results are shown in Table A4 of Appendix A. The composite average ranking of the DECCWOA is the highest, followed by the RDWOA and the CCMWOA with the lowest. Among them, +/−/= respectively records the number of benchmark functions that the DECCWOA is superior to, inferior to and similar to in terms of performance to other competing algorithms among the 35 test functions. For the worst performing CCMWOA, the DECCWOA outperforms it for twenty-eight benchmark functions, has the same performance on five functions, and performs slightly worse on only two functions. Moreover, compared to the RDWOA, which ranks second overall, the DECCWOA has better performance for sixteen benchmark functions, has the same optimization ability for thirteen functions and only has poor performance for six functions. This proves

that the DECCWOA has better performance than other improved WOA algorithms for most of the optimization problems, further demonstrating that the introduced CSO and DE have a positive steering effect on improving the basic WOA, such as slow convergence speed and poor accuracy guiding role.

Table 5. Comparison results for DECCWOAs with improved WOA versions.

Algorithms	Overall	
	+/−/=	Rank
DECCWOA	~	1
RDWOA	16/6/13	2
ACWOA	26/3/6	6
CCMWOA	28/2/5	9
CWOA	29/0/6	8
BMWOA	32/1/2	7
BWOA	23/3/9	4
LWOA	26/4/5	5
IWOA	24/1/10	3

In this section, the performance of the DECCWOA is compared with other improved versions of the WOA, including the RDWOA, the ACWOA, the CCMWOA [77], the CWOA [78], the BMWOA, the BWOA, the LWOA [79] and the IWOA [80]. Figure 5 shows the convergence curves of the average results obtained after 30 operations for all algorithms. On unimodal functions such as F6, it can be intuitively observed that the DECCWOA has the strongest search capability, with the RDWOA in second place, but the DECCWOA has a better performance than the RDWOA in terms of both accuracy and convergence speed. For both F12 and F13, the optimal values found by the other improved WOA algorithms are similar and more concentrated; however, the accuracy of the optimization obtained by the DECCWOA calculation is substantially improved. On F18, F19, F21, F23, F25 and F29, the DECCOWA can still search for more satisfactory optimal values compared to the other improved WOA algorithms. This demonstrates that the improvements to the WOA in this experiment are relatively more effective, and that, even when solving for multimodal functions, the DECCWOA can still jump out of the local optimum in time to obtain a high-quality optimal solution.

4.1.4. Comparison with Advanced Algorithms

Table 6 presents the comparison results for the DECCWOA with advanced algorithms. The detailed results can be found in Table A5 of Appendix A. *avg* reflects the average optimization ability of the algorithm after independently running on the benchmark function for 30 times, and *std* represents the influence of randomness on the optimization ability of the algorithm, which further reflects the stability of the algorithm to solve problems. From Table 6, the DECCWOA is superior to the IGWO on twenty functions and is inferior to the IGWO on eight functions (F3, F5, F22, F28, F30, F33, F34, F35). The DECCWOA beats the OBLGWOA on nineteen functions and loses to the OBLGWO on five functions (F3, F7, F28, F30, F33). For the CGPSO, ALCPSO and RCBA, the DECCWOA is inferior to them on nine functions, and outperforms most of the others. In detail, the DECCWOA is worse than the CGPSO at F5, F7, F8, F16, F17, F26, F30, F33 and F34, worse than the ALCPSO at F15, F16, F20, F21, F22, F28, F30, F33 and F34 and is worse than the RCBA at F14, F15, F16, F17, F20, F26, F30, F33 and F34. The DECCWOA beats the CBA on twenty-four functions, and loses to the CBA in six functions (F15, F16, F20, F30, F33, F34). The DECCWOA outperforms the OBSCA on 32 functions and only performs worse than the OBSCA on one function of F3. The DECCWOA is worse than the SCADE on F3 and F6. Based on the analysis above, the DECCWOA did not perform as well as the ALPSO, RCBA and CBA on the three unimodal functions (F14~F16) selected in CEC2014, but demonstrated competitive performance on the seven unimodal functions (F1~F7) selected from the twenty-three

benchmark functions. The DECCWOA does not perform more competitively than the other comparison algorithms in terms of hybrid functions, but the DECCWOA performs well on most of the multimodal functions.

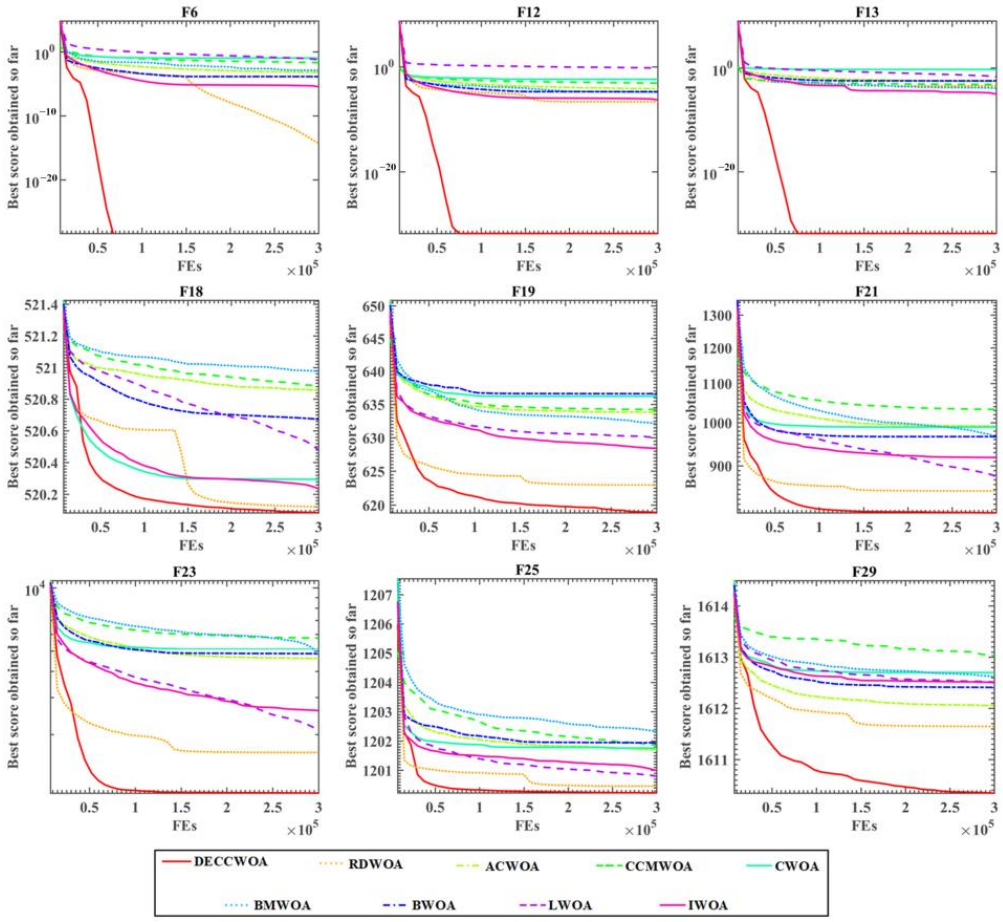

Figure 5. Convergence curves of comparison with improved WOA versions.

Table 6. Comparison results for the DECCWOA and advanced algorithms.

Algorithms	Overall	
	+/−/=	Rank
DECCWOA	~	1
IGWO	20/8/7	2
OBLGWO	19/5/11	5
CGPSO	23/9/3	4
ALPSO	19/9/7	3
RCBA	23/9/3	6
CBA	24/6/5	7
OBSCA	32/1/2	9
SCADE	28/2/5	7

In order to verify the effectiveness of the proposed DECCWOA compared to other advanced algorithms, comparison experiments were carried out. Among them, an enhanced GWO with a new hierarchical structure (IGWO) [81], boosted GWO (OBLGWO) [82], cluster guide PSO (CGPSO) [83], hybridizing sine cosine algorithm with differential evolution (SCADE) [84], particle swarm optimization with an aging leader and challengers (ALPSO) [85], hybrid bat algorithm (RCBA) [86], chaotic BA (CBA) [87] and opposition-based SCA (OBSCA) [88] were selected as the comparison algorithms. Convergence curves

for comparison with the advanced algorithms are displayed in Figure 6. In particular, for unimodal functions, the DECCWOA has the same search capability as the IGWO, OBLGWO, CGPSO and SCADE in F1. For F6, the DECCWOA has the strongest optimization capability and, as can be seen in Figure 6, the DECCWOA maintains a satisfactory convergence rate for F6. On the multimodal functions, such as F12, F13, F21, F23, F24 and F29, the DECCWOA also shows strong optimization ability. Compared with the classic ALPSO, the optimization performance of the DECCWOA is not inferior, and it can even converge to a better solution at a faster convergence rate. When solving a hybrid optimization problem, such as F31, although the IGWO can still obtain better solutions in the late iteration, its convergence speed is slow and the search ability is poor in the early iteration. The OBLGWOA, CGPAO, SCADE and OBSCA are unsatisfactory in terms of their optimization ability and convergence speeds during the entire iterative process, while the ALPSO, RCBA and CBA are relatively better; however, the DECCWOA showed better optimization than them.

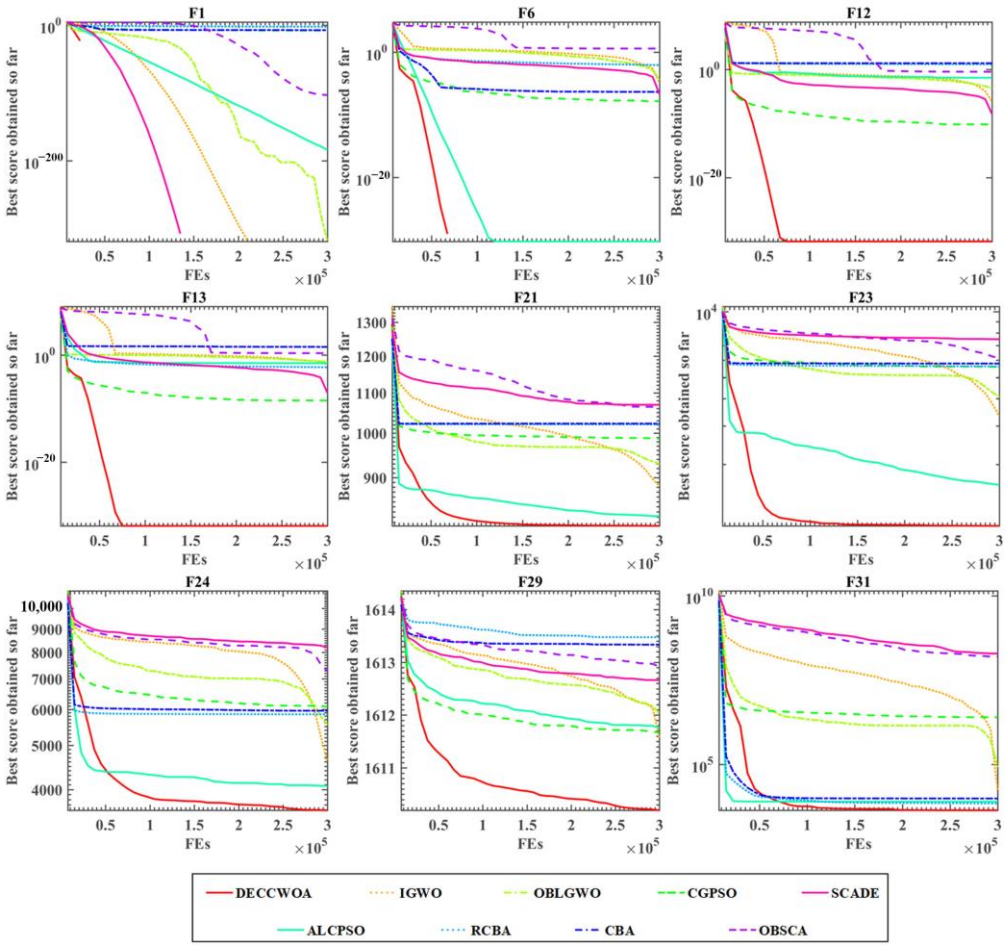

Figure 6. Convergence curves of the DECCWOA and advanced algorithms.

4.2. Experiments on Application of the DECCWOA in Predicting Talent Stability in Higher Education

4.2.1. Description of the Selected Data

The subjects studied in this paper were 69 talented individuals who left several colleges and universities in Wenzhou from 1 January 2015, accounting for 11.5% of the official staff. The following characteristics were examined: subject gender, political status, professional attributes, age, type of place of origin, category of talents above the municipal level, nature of the previous unit, type of location of college and university, year of employment at college and university, type of position at college and university, professional relevance of employment at college and university, annual salary level at college and university,

current employment unit, time of introduction of current employment unit, nature of current employment unit and type of location of current employment unit. The indicators, as presented in Table A6 of the Appendix A, were mined and analyzed to explore the importance and interconnectedness of each indicator, and to build an intelligent prediction model based on these indicators. Moreover, the following indicators are bolded as important indicators.

4.2.2. Experimental Results

The proposed DECCWOA was combined with the KELM and the feature selection (DECCWOA-KELM-FS) method to solve the classification problem of employment intention of talent. The experimental results are shown in Tables 7 and 8. The DECCWOA-KELM-FS's results on the ACC, Sensitivity, Specificity and MCC indicators are 95.87%, 94.96%, 96.59% and 91.64%, respectively. The classification results are all superior to other comparison algorithms, including the DECCWOA-KELM, DECCWOA-KELM, WOA-KELM, ANN, RF and SVM. Furthermore, the stability results of the ten experimental results of the proposed model are also superior. The std metrics results of the ACC, Sensitivity, Specificity and MCC indicators are 3.19×10^{-2}, 6.85×10^{-2}, 4.25×10^{-2} and 6.66×10^{-2}. Obviously, the stability of the proposed algorithm is better than that of most comparison algorithms. Therefore, by combining the DECCWOA with the KELM and FS, the talent stability prediction of Wenzhou Vocational College is effectively realized. To further visualize the results, Figure 7 shows a comparison of results between the proposed algorithm and the other five methods, including the average results and standard deviations of the five indicators. Similarly, the average performance and stability of the DECCWOA-KELM-FS in each index are better than most reported algorithms.

Table 7. Four avg metrics results of the proposed model and other models.

Models	ACC	Sensitivity	Specificity	MCC
DECCWOA-KELM-FS	95.87%	94.64%	96.59%	91.64%
DECCWOA-KELM	92.57%	94.05%	92.50%	86.27%
WOA-KELM	90.32%	89.39%	91.05%	81.10%
ANN	88.96%	87.58%	90.51%	77.16%
RF	92.67%	92.85%	91.01%	85.33%
SVM	89.30%	91.92%	86.96%	79.75%

Table 8. Four std metrics results of the proposed model and other models.

Models	ACC	Sensitivity	Specificity	MCC
DECCWOA-KELM-FS	3.19×10^{-2}	6.85×10^{-2}	4.25×10^{-2}	6.66×10^{-2}
DECCWOA-KELM	5.60×10^{-2}	8.31×10^{-2}	8.93×10^{-2}	1.01×10^{-1}
WOA-KELM	4.33×10^{-2}	9.06×10^{-2}	1.02×10^{-1}	8.80×10^{-2}
ANN	4.16×10^{-2}	5.61×10^{-2}	5.91×10^{-2}	9.50×10^{-2}
RF	4.17×10^{-2}	1.08×10^{-1}	6.74×10^{-2}	8.59×10^{-2}
SVM	6.72×10^{-2}	1.12×10^{-1}	1.07×10^{-1}	1.18×10^{-1}

Figure 8 shows the feature selection results of the proposed model. As can be seen, F7 (city-level and above talent categories) and F22 (professional and technical position at the time of leaving) are both screened the most, eight times. It shows that the two key factors affecting the stability of university talents are F7 and F22, which provides some guiding significance of the flow of highly educated talents. Based on the fact that the proposed method has such excellent performance, it can also be applied in many other fields in the future, such as information retrieval services [89,90], named entity recognition [91], road network planning [92], colorectal polyp region extraction [93], image denoising [94], image segmentation [95–97] and power flow optimization [98].

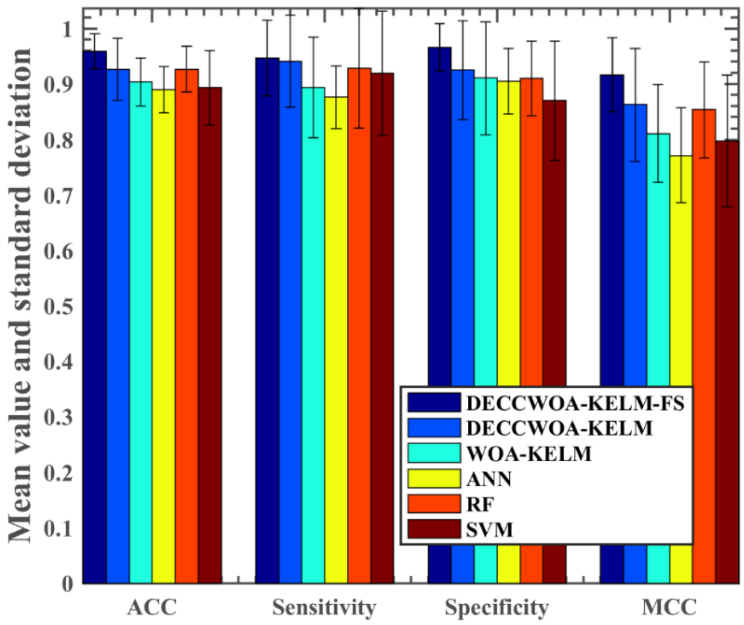

Figure 7. Mean value and standard deviation of four metrics for the DECCWOA and others methods.

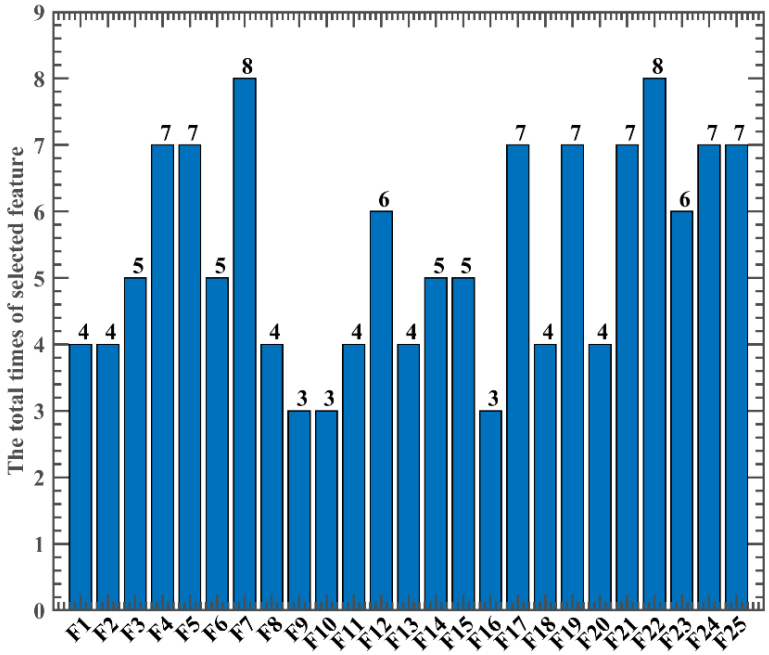

Figure 8. Feature selection results of the proposed model.

5. Conclusions

This paper studied the stability of higher education talent for the first time, and proposed a DECCWOA-KELM-FS model to intelligently predict the stability of higher education talent. By introducing a crossover algorithm, the information exchange between individuals was promoted and the problem of dimension stagnation was improved. The DE operation was carried out in a certain time, and the difference between individuals was used to disturb the population and ensure the diversity of the population. In order to verify the optimization performance of the DECCWOA, 35 benchmark functions were selected from 32 benchmark functions and CEC214 for comparative experiments. Experimental results showed that the DECCWOA algorithm had higher accuracy and faster convergence rates when solving unimodal and multimodal functions; although the mixture function

also had very good performance. By combining the DECCWOA with the KELM and feature selection, the stable intelligence of talent in Wenzhou colleges and universities was efficiently predicted. This method can be used as a reliable and high precision method to predict the flow of talent in colleges and universities.

Subsequent studies will further improve the generality of the proposed GLLCSA-KELM-FS and solve more complex classification problems, such as disease diagnosis and financial risk prediction.

Author Contributions: Conceptualization, G.L. and H.C.; methodology, F.K. and H.C.; software, G.L. and F.K.; validation, H.L., S.K., X.R., C.L., G.L., H.C., F.K. and L.L.; formal analysis, F.K., G.L. and L.L.; investigation, H.L., S.K., D.C. and C.L.; resources, F.K., G.L. and L.L.; data curation, F.K., G.L. and C.L.; writing—original draft preparation, H.L., S.K., X.R. and C.L.; writing—review and editing, G.L. and H.C.; visualization, G.L. and H.C.; supervision, F.K., G.L. and L.L.; project administration, F.K., G.L. and L.L.; funding acquisition, F.K., G.L. and H.C. All authors have read and agreed to the published version of the manuscript.

Funding: Zhejiang Provincial universities Major Humanities and social Science project: Innovation and Practice of Cultivating Paths for Leaders in Rural Industry Revitalization under the Background of Common Prosperity (Moderator: Li Hong), Humanities and Social Science Research Planning Fund Project of the Ministry of Education (research on risk measurement and early warning mechanism of science and technology finance based on big data analysis, 20YJA790090), Zhejiang Provincial Philosophy and Social Sciences Planning Project (Research on rumor recognition and dissemination intervention based on automated essay scoring, 23NDJC393YBM).

Data Availability Statement: The data involved in this study are all public data, which can be downloaded through public channels.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Details of the selected 35 benchmark functions.

Types	No.	Functions	Rang	f_{min}
Unimodal Functions	F1	$f_1(x) = \sum_{i=1}^{n} x_i^2$	$[-100, 100]$	0
	F2	$f_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	$[-10, 10]$	0
	F3	$f_3(x) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j \right)^2$	$[-100, 100]$	0
	F4	$f_4(x) = max_i\{\lvert x_i \rvert, 1 \le i \le n\}$	$[-100, 100]$	0
	F5	$f_5(x) = \sum_{i=1}^{n-1} \left[100 \left(x_{i+1} - x_i^2 \right)^2 + (x_i - 1)^2 \right]$	$[-30, 30]$	0
	F6	$f_6(x) = \sum_{i=1}^{n} (\lvert x_i + 0.5 \rvert)^2$	$[-100, 100]$	0
	F7	$f_7(x) = \sum_{i=1}^{n} i x_i^4 + random[0, 1]$	$[-1.28, 1.28]$	0
Multimodal Functions	F8	$f_8(x) = \sum_{i=1}^{n} -x_i \sin\left(\sqrt{\lvert x_i \rvert} \right)$	$[-500, 500]$	-418.9829×5
	F9	$f_9(x) = \sum_{i=1}^{n} \left[x_i^2 - 10 \cos(2\pi x_i) + 10 \right]$	$[-5.12, 5.12]$	0
	F10	$f_{10}(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2} \right) - \exp\left(\frac{1}{n} \sum_{i=1}^{n} \cos(2\pi x_i) \right) + 20 + e$	$[-32, 32]$	0
	F11	$f_{11}(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}} \right) + 1$	$[-600, 600]$	0
	F12	$f_{12}(x) = \frac{\pi}{n} \left\{ 10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 \left[1 + 10 \sin^2\left(\pi y_{i+1} \right) \right] + (y_n - 1)^2 \right\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_i + 1}{4}, u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	$[-50, 50]$	0
	F13	$f_{13}(x) = 0.1 \left\{ \sin^2(3\pi x_1) + \sum_{i=1}^{n} (x_i - 1)^2 \left[1 + \sin^2(3\pi x_i + 1) \right] + (x_n - 1)^2 \left[1 + \sin^2(2\pi x_n) \right] \right\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	$[-50, 50]$	0
Unimodal Functions	F14	Rotated High Conditioned Elliptic Function	$[-100, 100]$	100
	F15	Rotated Bent Cigar Function	$[-100, 100]$	200
	F16	Rotated Discus Function	$[-100, 100]$	300

Table A1. *Cont.*

Types	No.	Functions	Rang	f_{min}
Simple Multimodal Functions	F17	Shifted and Rotated Rosenbrock's Function	[−100, 100]	400
	F18	Shifted and Rotated Ackley's Function	[−100, 100]	500
	F19	Shifted and Rotated Weierstrass Function	[−100, 100]	600
	F20	Shifted and Rotated Griewank's Function	[−100, 100]	700
	F21	Shifted Rastrigin's Function	[−100, 100]	800
	F22	Shifted and Rotated Rastrigin's Function	[−100, 100]	900
	F23	Shifted Schwefel's Function	[−100, 100]	1000
	F24	Shifted and Rotated Schwefel's Function	[−100, 100]	1100
	F25	Shifted and Rotated Katsuura Function	[−100, 100]	1200
	F26	Shifted and Rotated HappyCat Function	[−100, 100]	1300
	F27	Shifted and Rotated HGBat Function	[−100, 100]	1400
	F28	Shifted and Rotated Expanded Griewank's plus Rosenbrock's Function	[−100, 100]	1500
	F29	Shifted and Rotated Expanded Scaffer's F6 Function	[−100, 100]	1600
Hybrid Function1	F30	Hybrid Function 1 (N = 3)	[−100, 100]	1700
	F31	Hybrid Function 2 (N = 3)	[−100, 100]	1800
	F32	Hybrid Function 3 (N = 4)	[−100, 100]	1900
	F33	Hybrid Function 4 (N = 4)	[−100, 100]	2000
	F34	Hybrid Function 5 (N = 5)	[−100, 100]	2100
	F35	Hybrid Function 6 (N = 5)	[−100, 100]	2200

Table A2. Experimental results for analysis of key parameter p_2.

	F1		F2		F3	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	6.0453×10^{-8}	3.3108×10^{-7}
DECCWOA2	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	4.2361×10^{-17}	2.3074×10^{-16}
DECCWOA3	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	6.1871×10^{-28}	1.4243×10^{-27}
DECCWOA4	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.8338×10^{-27}	2.4802×10^{-27}
DECCWOA5	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	5.5498×10^{-28}	1.4428×10^{-27}
DECCWOA6	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	5.9383×10^{-28}	1.6120×10^{-27}
DECCWOA7	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	9.3229×10^{-28}	1.9913×10^{-27}
DECCWOA8	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	3.4963×10^{-28}	1.1052×10^{-27}
DECCWOA9	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	5.3287×10^{-28}	1.2818×10^{-27}
DECCWOA10	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	4.2382×10^{-28}	1.3085×10^{-27}
	F4		F5		F6	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	0.0000×10^{0}	0.0000×10^{0}	2.4439×10^{1}	6.6504×10^{0}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA2	0.0000×10^{0}	0.0000×10^{0}	2.4123×10^{1}	6.5641×10^{0}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA3	0.0000×10^{0}	0.0000×10^{0}	2.6257×10^{1}	2.8910×10^{-1}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA4	0.0000×10^{0}	0.0000×10^{0}	2.5987×10^{1}	4.1389×10^{-1}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA5	0.0000×10^{0}	0.0000×10^{0}	2.6107×10^{1}	3.0119×10^{-1}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA6	0.0000×10^{0}	0.0000×10^{0}	2.5412×10^{1}	4.8062×10^{0}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA7	0.0000×10^{0}	0.0000×10^{0}	2.5432×10^{1}	4.8097×10^{0}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA8	0.0000×10^{0}	0.0000×10^{0}	2.4619×10^{1}	6.6972×10^{0}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA9	0.0000×10^{0}	0.0000×10^{0}	2.5539×10^{1}	4.8318×10^{0}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA10	0.0000×10^{0}	0.0000×10^{0}	2.6489×10^{1}	3.4054×10^{-1}	0.0000×10^{0}	0.0000×10^{0}

Table A2. *Cont.*

	F7		F8		F9	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	1.6842×10^{-4}	2.7932×10^{-4}	-1.3963×10^{4}	5.1858×10^{3}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA2	1.5645×10^{-4}	1.7877×10^{-4}	-1.2595×10^{4}	1.3910×10^{2}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA3	6.6910×10^{-5}	9.9764×10^{-5}	-1.2619×10^{4}	2.7031×10^{2}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA4	1.0031×10^{-4}	1.4658×10^{-4}	-1.2530×10^{4}	5.3042×10^{2}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA5	8.9743×10^{-5}	1.0676×10^{-4}	-1.3512×10^{4}	5.1597×10^{3}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA6	5.2519×10^{-5}	5.3764×10^{-5}	-1.2569×10^{4}	1.9404×10^{-12}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA7	6.4531×10^{-5}	6.4938×10^{-5}	-1.3485×10^{4}	2.8423×10^{3}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA8	4.0419×10^{-5}	5.3492×10^{-5}	-1.2805×10^{4}	9.0308×10^{2}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA9	6.3101×10^{-5}	7.7305×10^{-5}	-1.2569×10^{4}	2.0267×10^{-12}	0.0000×10^{0}	0.0000×10^{0}
DECCWOA10	3.8169×10^{-5}	6.1534×10^{-5}	-1.2673×10^{4}	2.7066×10^{2}	0.0000×10^{0}	0.0000×10^{0}
	F10		F11		F12	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
DECCWOA2	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
DECCWOA3	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
DECCWOA4	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
DECCWOA5	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
DECCWOA6	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
DECCWOA7	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
DECCWOA8	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
DECCWOA9	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
DECCWOA10	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
	F13		F14		F15	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	1.3498×10^{-32}	5.5674×10^{-48}	3.6810×10^{6}	2.7334×10^{6}	1.3342×10^{5}	1.9982×10^{5}
DECCWOA2	1.3498×10^{-32}	5.5674×10^{-48}	5.1469×10^{6}	3.7469×10^{6}	2.2654×10^{5}	3.0564×10^{5}
DECCWOA3	1.3498×10^{-32}	5.5674×10^{-48}	1.6507×10^{7}	1.0130×10^{7}	9.4042×10^{7}	8.6776×10^{7}
DECCWOA4	1.3498×10^{-32}	5.5674×10^{-48}	7.2801×10^{6}	5.0765×10^{6}	8.7941×10^{6}	7.2596×10^{6}
DECCWOA5	1.3498×10^{-32}	5.5674×10^{-48}	1.1619×10^{7}	7.8606×10^{6}	3.3924×10^{7}	2.5699×10^{7}
DECCWOA6	1.3498×10^{-32}	5.5674×10^{-48}	1.2238×10^{7}	8.2473×10^{6}	7.4207×10^{7}	5.2159×10^{7}
DECCWOA7	1.3498×10^{-32}	5.5674×10^{-48}	2.5408×10^{7}	1.2932×10^{7}	1.5504×10^{8}	1.0836×10^{8}
DECCWOA8	1.3498×10^{-32}	5.5674×10^{-48}	2.9064×10^{7}	1.2592×10^{7}	3.1493×10^{8}	3.1857×10^{8}
DECCWOA9	1.3498×10^{-32}	5.5674×10^{-48}	2.7983×10^{7}	1.7994×10^{7}	4.3963×10^{8}	5.9410×10^{8}
DECCWOA10	1.3498×10^{-32}	5.5674×10^{-48}	4.0140×10^{7}	2.8456×10^{7}	6.8723×10^{8}	7.6542×10^{8}
	F16		F17		F18	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	7.4819×10^{3}	4.6257×10^{3}	4.9577×10^{2}	4.4816×10^{1}	5.2004×10^{2}	4.4169×10^{-2}
DECCWOA2	5.3970×10^{3}	5.6834×10^{3}	5.2284×10^{2}	4.4110×10^{1}	5.2009×10^{2}	4.7541×10^{-2}
DECCWOA3	5.5074×10^{3}	3.3963×10^{3}	5.9759×10^{2}	5.0184×10^{1}	5.2036×10^{2}	1.4114×10^{-1}
DECCWOA4	4.7987×10^{3}	4.0122×10^{3}	5.5522×10^{2}	5.6453×10^{1}	5.2019×10^{2}	9.2319×10^{-2}
DECCWOA5	3.9879×10^{3}	2.8027×10^{3}	5.6621×10^{2}	4.6593×10^{1}	5.2029×10^{2}	1.0546×10^{-1}
DECCWOA6	4.7947×10^{3}	3.8454×10^{3}	6.0512×10^{2}	3.7665×10^{1}	5.2031×10^{2}	1.4175×10^{-1}
DECCWOA7	4.7025×10^{3}	3.4303×10^{3}	6.4612×10^{2}	5.0034×10^{1}	5.2039×10^{2}	1.4396×10^{-1}
DECCWOA8	5.7773×10^{3}	3.8347×10^{3}	6.9438×10^{2}	9.0009×10^{1}	5.2033×10^{2}	1.7149×10^{-1}
DECCWOA9	6.8194×10^{3}	3.9312×10^{3}	6.7202×10^{2}	7.1887×10^{1}	5.2035×10^{2}	1.6351×10^{-1}
DECCWOA10	7.6046×10^{3}	2.9400×10^{3}	6.9534×10^{2}	7.7239×10^{1}	5.2038×10^{2}	1.7225×10^{-1}

Table A2. *Cont.*

	F19		F20		F21	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	6.2052×10^{2}	3.2852×10^{0}	7.0040×10^{2}	2.2886×10^{-1}	8.0287×10^{2}	1.0273×10^{1}
DECCWOA2	6.1946×10^{2}	2.9290×10^{0}	7.0052×10^{2}	2.0981×10^{-1}	8.1110×10^{2}	2.0103×10^{1}
DECCWOA3	6.2495×10^{2}	2.9171×10^{0}	7.0250×10^{2}	8.3283×10^{-1}	8.7716×10^{2}	1.8311×10^{1}
DECCWOA4	6.2184×10^{2}	2.6694×10^{0}	7.0111×10^{2}	7.1368×10^{-2}	8.2441×10^{2}	7.7346×10^{0}
DECCWOA5	6.2371×10^{2}	2.6632×10^{0}	7.0164×10^{2}	4.2137×10^{-1}	8.4642×10^{2}	1.7780×10^{1}
DECCWOA6	6.2456×10^{2}	3.2157×10^{0}	7.0230×10^{2}	8.4958×10^{-1}	8.7407×10^{2}	2.4081×10^{1}
DECCWOA7	6.2621×10^{2}	3.2713×10^{0}	7.0300×10^{2}	8.7250×10^{-1}	8.9713×10^{2}	1.9802×10^{1}
DECCWOA8	6.2840×10^{2}	3.4007×10^{0}	7.0506×10^{2}	2.9144×10^{0}	9.1581×10^{2}	1.9967×10^{1}
DECCWOA9	6.2897×10^{2}	3.4733×10^{0}	7.0619×10^{2}	2.5650×10^{0}	9.2703×10^{2}	2.5100×10^{1}
DECCWOA10	6.2933×10^{2}	3.7025×10^{0}	7.0753×10^{2}	3.4438×10^{0}	9.4279×10^{2}	2.4986×10^{1}
	F22		F23		F24	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	1.0312×10^{3}	2.7380×10^{1}	1.0420×10^{3}	1.1886×10^{2}	3.9026×10^{3}	6.5371×10^{2}
DECCWOA2	1.0377×10^{3}	3.4427×10^{1}	1.0510×10^{3}	7.9342×10^{1}	4.1202×10^{3}	7.1968×10^{2}
DECCWOA3	1.0545×10^{3}	3.3397×10^{1}	1.7059×10^{3}	3.0416×10^{2}	5.3301×10^{3}	4.9187×10^{2}
DECCWOA4	1.0420×10^{3}	3.5382×10^{1}	1.1956×10^{3}	2.0676×10^{2}	4.2088×10^{3}	6.4091×10^{2}
DECCWOA5	1.0502×10^{3}	2.9746×10^{1}	1.4485×10^{3}	4.7609×10^{2}	4.9060×10^{3}	7.5372×10^{2}
DECCWOA6	1.0708×10^{3}	2.3208×10^{1}	1.6654×10^{3}	3.5966×10^{2}	5.0839×10^{3}	7.4254×10^{2}
DECCWOA7	1.0725×10^{3}	2.9966×10^{1}	2.4340×10^{3}	3.6991×10^{2}	5.5460×10^{3}	6.9622×10^{2}
DECCWOA8	1.0867×10^{3}	3.0093×10^{1}	2.8661×10^{3}	5.7401×10^{2}	5.6002×10^{3}	5.3783×10^{2}
DECCWOA9	1.0862×10^{3}	2.9502×10^{1}	3.4724×10^{3}	6.7538×10^{2}	5.6526×10^{3}	7.8496×10^{2}
DECCWOA10	1.0860×10^{3}	3.5118×10^{1}	3.7201×10^{3}	5.3438×10^{2}	5.6708×10^{3}	5.7142×10^{2}
	F25		F26		F27	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	1.2002×10^{3}	6.7099×10^{-2}	1.3005×10^{3}	1.5439×10^{-1}	1.4003×10^{3}	4.8887×10^{-2}
DECCWOA2	1.2002×10^{3}	6.0385×10^{-2}	1.3005×10^{3}	1.1841×10^{-1}	1.4003×10^{3}	5.5441×10^{-2}
DECCWOA3	1.2008×10^{3}	2.9906×10^{-1}	1.3005×10^{3}	1.1364×10^{-1}	1.4003×10^{3}	4.0852×10^{-2}
DECCWOA4	1.2004×10^{3}	1.2088×10^{-1}	1.3005×10^{3}	1.3836×10^{-1}	1.4003×10^{3}	1.9112×10^{-1}
DECCWOA5	1.2005×10^{3}	1.9004×10^{-1}	1.3005×10^{3}	1.0714×10^{-1}	1.4003×10^{3}	5.7154×10^{-2}
DECCWOA6	1.2008×10^{3}	2.5696×10^{-1}	1.3005×10^{3}	9.5896×10^{-2}	1.4003×10^{3}	1.0255×10^{-1}
DECCWOA7	1.2010×10^{3}	3.2942×10^{-1}	1.3005×10^{3}	1.1984×10^{-1}	1.4003×10^{3}	1.7872×10^{-1}
DECCWOA8	1.2011×10^{3}	3.7909×10^{-1}	1.3005×10^{3}	1.3301×10^{-1}	1.4004×10^{3}	6.0314×10^{-1}
DECCWOA9	1.2012×10^{3}	3.9281×10^{-1}	1.3006×10^{3}	1.4510×10^{-1}	1.4003×10^{3}	5.1310×10^{-2}
DECCWOA10	1.2014×10^{3}	4.0093×10^{-1}	1.3005×10^{3}	1.1885×10^{-1}	1.4004×10^{3}	1.7136×10^{-1}
	F28		F29		F30	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	1.5176×10^{3}	6.4897×10^{0}	1.6104×10^{3}	6.8683×10^{-1}	1.9247×10^{6}	1.1101×10^{6}
DECCWOA2	1.5189×10^{3}	7.4912×10^{0}	1.6104×10^{3}	7.6782×10^{-1}	1.5103×10^{6}	1.1011×10^{6}
DECCWOA3	1.5422×10^{3}	1.3549×10^{1}	1.6115×10^{3}	6.7294×10^{-1}	1.9418×10^{6}	1.2814×10^{6}
DECCWOA4	1.5266×10^{3}	6.5567×10^{0}	1.6110×10^{3}	6.9374×10^{-1}	1.7005×10^{6}	9.1883×10^{5}
DECCWOA5	1.5304×10^{3}	9.0407×10^{0}	1.6113×10^{3}	6.2976×10^{-1}	1.8482×10^{6}	1.1135×10^{6}
DECCWOA6	1.5424×10^{3}	1.2171×10^{1}	1.6116×10^{3}	5.9848×10^{-1}	1.9396×10^{6}	1.2724×10^{6}
DECCWOA7	1.6047×10^{3}	2.9073×10^{2}	1.6119×10^{3}	6.5689×10^{-1}	2.0687×10^{6}	1.2220×10^{6}
DECCWOA8	1.5684×10^{3}	3.2783×10^{1}	1.6120×10^{3}	4.3100×10^{-1}	2.2597×10^{6}	1.5545×10^{6}
DECCWOA9	1.5970×10^{3}	4.5170×10^{1}	1.6121×10^{3}	6.1504×10^{-1}	2.2349×10^{6}	1.2776×10^{6}
DECCWOA10	1.6376×10^{3}	9.5739×10^{1}	1.6119×10^{3}	7.4369×10^{-1}	2.4610×10^{6}	1.6436×10^{6}

Table A2. *Cont.*

	F31		F32		F33	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA1	5.8087×10^3	6.2358×10^3	1.9179×10^3	2.6069×10^1	1.4137×10^4	7.6746×10^3
DECCWOA2	4.8867×10^3	3.6023×10^3	1.9189×10^3	2.3237×10^1	8.3832×10^3	5.6464×10^3
DECCWOA3	4.0166×10^3	2.5970×10^3	1.9215×10^3	2.4809×10^1	5.5742×10^3	2.6752×10^3
DECCWOA4	5.2173×10^3	3.9350×10^3	1.9267×10^3	3.4926×10^1	4.9997×10^3	2.4000×10^3
DECCWOA5	4.9446×10^3	4.1973×10^3	1.9262×10^3	2.9792×10^1	4.5639×10^3	2.0732×10^3
DECCWOA6	4.5409×10^3	2.8935×10^3	1.9239×10^3	2.2044×10^1	5.0427×10^3	2.4608×10^3
DECCWOA7	4.7751×10^3	2.8190×10^3	1.9279×10^3	2.2825×10^1	4.5396×10^3	1.8457×10^3
DECCWOA8	4.6894×10^4	2.3224×10^5	1.9358×10^3	3.7797×10^1	4.8777×10^3	1.6024×10^3
DECCWOA9	7.0415×10^3	9.3191×10^3	1.9284×10^3	2.5490×10^1	4.5525×10^3	2.1545×10^3
DECCWOA10	6.2462×10^3	9.1482×10^3	1.9295×10^3	2.4829×10^1	3.7338×10^3	1.6768×10^3
Overall rank	F34		F35		overall	
	avg	*std*	*avg*	*std*	+/−/=	rank
DECCWOA1	8.6335×10^5	6.2796×10^5	2.8860×10^3	2.2542×10^2	~	2
DECCWOA2	7.0965×10^5	6.4048×10^5	2.7590×10^3	1.9713×10^2	4/3/28	1
DECCWOA3	8.1850×10^5	6.4237×10^5	2.6856×10^3	1.9951×10^2	14/3/18	6
DECCWOA4	8.3727×10^5	6.0484×10^5	2.8021×10^3	1.8728×10^2	12/2/21	5
DECCWOA5	7.2517×10^5	6.6623×10^5	2.7406×10^3	2.2485×10^2	14/4/17	3
DECCWOA6	5.9730×10^5	4.4486×10^5	2.7219×10^3	1.8356×10^2	14/5/16	4
DECCWOA7	6.4163×10^5	4.3900×10^5	2.6995×10^3	2.0571×10^2	14/4/17	7
DECCWOA8	7.0675×10^5	6.1652×10^5	2.7178×10^3	1.9634×10^2	15/4/16	8
DECCWOA9	6.4914×10^5	5.4887×10^5	2.7793×10^3	1.9480×10^2	17/2/16	9
DECCWOA10	5.9732×10^5	4.7600×10^5	2.7603×10^3	2.1609×10^2	16/2/17	10

Table A3. Comparison results for the introduced mechanisms.

	F1		F2		F3	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	0.0000×10^0	0.0000×10^0	0.0000×10^0	0.0000×10^0	4.1255×10^{-18}	2.1806×10^{-17}
DEWOA	1.2420×10^{-10}	3.9129×10^{-10}	8.5083×10^{-6}	1.4067×10^{-5}	7.8264×10^3	1.2145×10^4
CCWOA	0.0000×10^0	0.0000×10^0	0.0000×10^0	0.0000×10^0	0.0000×10^0	0.0000×10^0
WOA	0.0000×10^0	0.0000×10^0	0.0000×10^0	0.0000×10^0	3.2071×10^1	6.1783×10^1
	F4		F5		F6	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	0.0000×10^0	0.0000×10^0	2.5968×10^1	3.5258×10^{-1}	0.0000×10^0	0.0000×10^0
DEWOA	6.3178×10^{-3}	1.5936×10^{-2}	4.4319×10^{-3}	4.7769×10^{-3}	9.6180×10^{-5}	1.2811×10^{-4}
CCWOA	0.0000×10^0	0.0000×10^0	2.2483×10^1	6.1153×10^0	1.1597×10^{-11}	1.3565×10^{-11}
WOA	7.5414×10^0	1.7526×10^1	2.3562×10^1	4.4675×10^0	4.7799×10^{-6}	1.8846×10^{-6}
	F7		F8		F9	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.0328×10^{-4}	1.7676×10^{-4}	-1.2783×10^4	8.3042×10^2	0.0000×10^0	0.0000×10^0
DEWOA	3.4741×10^{-3}	5.8983×10^{-3}	-1.4406×10^4	4.9552×10^3	6.2111×10^{-10}	8.6723×10^{-10}
CCWOA	1.7804×10^{-5}	3.0937×10^{-5}	-1.2569×10^4	5.6938×10^{-7}	0.0000×10^0	0.0000×10^0
WOA	1.5818×10^{-4}	1.8724×10^{-4}	-1.2236×10^4	8.6401×10^2	0.0000×10^0	0.0000×10^0

Table A3. *Cont.*

	F10		F11		F12	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
DEWOA	4.8353×10^{-6}	1.1226×10^{-5}	6.9380×10^{-7}	3.7853×10^{-6}	7.0591×10^{-6}	9.9086×10^{-6}
CCWOA	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.6821×10^{-12}	2.1310×10^{-12}
WOA	3.6119×10^{-15}	1.7906×10^{-15}	2.7668×10^{-4}	1.5155×10^{-3}	2.1111×10^{-4}	1.1507×10^{-3}
	F13		F14		F15	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.3498×10^{-32}	5.5674×10^{-48}	4.6009×10^{6}	3.5694×10^{6}	1.5068×10^{5}	1.7046×10^{5}
DEWOA	1.3742×10^{-4}	2.1617×10^{-4}	4.5118×10^{7}	3.3352×10^{7}	2.6842×10^{9}	2.2525×10^{9}
CCWOA	2.2806×10^{-11}	2.1476×10^{-11}	1.0815×10^{7}	7.8011×10^{6}	6.3006×10^{6}	3.9908×10^{6}
WOA	1.1255×10^{-3}	3.3493×10^{-3}	3.3488×10^{7}	1.7699×10^{7}	3.8002×10^{6}	7.5915×10^{6}
	F16		F17		F18	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	6.2884×10^{3}	4.0514×10^{3}	4.9673×10^{2}	3.4575×10^{1}	5.2008×10^{2}	5.6636×10^{-2}
DEWOA	3.6134×10^{4}	3.3572×10^{4}	8.0754×10^{2}	1.5372×10^{2}	5.2033×10^{2}	2.2602×10^{-1}
CCWOA	5.8371×10^{3}	3.4122×10^{3}	5.3911×10^{2}	5.8876×10^{1}	5.2032×10^{2}	8.7776×10^{-2}
WOA	3.6143×10^{4}	2.3682×10^{4}	5.8016×10^{2}	5.5983×10^{1}	5.2032×10^{2}	1.6884×10^{-1}
	F19		F20		F21	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	6.1813×10^{2}	3.3345×10^{0}	7.0050×10^{2}	1.8285×10^{-1}	8.0336×10^{2}	5.5434×10^{0}
DEWOA	6.4146×10^{2}	2.3442×10^{0}	7.1656×10^{2}	1.1726×10^{1}	9.7436×10^{2}	2.7480×10^{1}
CCWOA	6.2191×10^{2}	3.1893×10^{0}	7.0106×10^{2}	1.1101×10^{-1}	8.2734×10^{2}	1.6890×10^{1}
WOA	6.3521×10^{2}	3.8342×10^{0}	7.0102×10^{2}	7.0807×10^{-2}	9.8601×10^{2}	3.9020×10^{1}
	F22		F23		F24	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.0398×10^{3}	4.1688×10^{1}	1.0459×10^{3}	8.8217×10^{1}	3.7945×10^{3}	5.1310×10^{2}
DEWOA	1.1215×10^{3}	3.4240×10^{1}	5.3843×10^{3}	9.7624×10^{2}	7.0285×10^{3}	1.3163×10^{3}
CCWOA	1.0486×10^{3}	3.6153×10^{1}	1.3579×10^{3}	7.6567×10^{2}	4.2859×10^{3}	5.9815×10^{2}
WOA	1.1285×10^{3}	5.7638×10^{1}	4.9219×10^{3}	6.8542×10^{2}	5.7481×10^{3}	9.5752×10^{2}
DECCWOA	1.0398×10^{3}	4.1688×10^{1}	1.0459×10^{3}	8.8217×10^{1}	3.7945×10^{3}	5.1310×10^{2}
	F25		F26		F27	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.2002×10^{3}	7.7530×10^{-2}	1.3005×10^{3}	1.0640×10^{-1}	1.4003×10^{3}	5.7690×10^{-2}
DEWOA	1.2026×10^{3}	6.7176×10^{-1}	1.3011×10^{3}	8.8278×10^{-1}	1.4032×10^{3}	6.9794×10^{0}
CCWOA	1.2007×10^{3}	1.9439×10^{-1}	1.3005×10^{3}	1.0762×10^{-1}	1.4003×10^{3}	5.6885×10^{-2}
WOA	1.2017×10^{3}	5.8032×10^{-1}	1.3005×10^{3}	1.4025×10^{-1}	1.4003×10^{3}	4.6882×10^{-2}
	F28		F29		F30	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.5215×10^{3}	7.2851×10^{0}	1.6105×10^{3}	5.6166×10^{-1}	1.6567×10^{6}	1.2704×10^{6}
DEWOA	3.9455×10^{3}	1.8290×10^{3}	1.6129×10^{3}	5.3817×10^{-1}	5.2403×10^{6}	3.8043×10^{6}
CCWOA	1.5263×10^{3}	7.8018×10^{0}	1.6111×10^{3}	5.3599×10^{-1}	2.3769×10^{6}	1.7422×10^{6}
WOA	1.5710×10^{3}	2.7076×10^{1}	1.6124×10^{3}	6.2114×10^{-1}	3.8096×10^{6}	3.0597×10^{6}

Table A3. *Cont.*

	F31		F32		F33	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	6.8401×10^{3}	6.1595×10^{3}	1.9156×10^{3}	1.7561×10^{1}	8.1086×10^{3}	4.1891×10^{3}
DEWOA	1.1805×10^{4}	1.9283×10^{4}	2.0808×10^{3}	7.9264×10^{1}	2.3551×10^{4}	1.9203×10^{4}
CCWOA	1.5061×10^{4}	2.0280×10^{4}	1.9305×10^{3}	3.7604×10^{1}	5.3268×10^{3}	3.0310×10^{3}
WOA	6.2265×10^{3}	4.2908×10^{3}	1.9415×10^{3}	3.6716×10^{1}	2.3634×10^{4}	1.4518×10^{4}
Overall rank	F34		F35		overall	
	avg	*std*	*avg*	*std*	+/−/=	rank
DECCWOA	7.8180×10^{5}	6.8802×10^{5}	2.8140×10^{3}	2.4485×10^{2}	~	1
DEWOA	1.1914×10^{6}	1.0362×10^{6}	3.3508×10^{3}	3.5303×10^{2}	31/1/3	4
CCWOA	5.8255×10^{5}	4.9559×10^{5}	2.7967×10^{3}	1.7716×10^{2}	18/4/13	2
WOA	1.3452×10^{6}	1.6988×10^{6}	3.0734×10^{3}	2.6034×10^{2}	25/1/9	3

Table A4. Comparison results for the DECCWOA with improved WOA versions.

	F1		F2		F3	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	2.7777×10^{-18}	1.2942×10^{-17}
RDWOA	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}
ACWOA	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}
CCMWOA	0.0000×10^{0}	0.0000×10^{0}	4.7501×10^{-286}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}
CWOA	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	6.9438×10^{0}	1.0312×10^{1}
BMWOA	9.0723×10^{-4}	1.2467×10^{-3}	7.9729×10^{-3}	7.3643×10^{-3}	2.4579×10^{-1}	7.2733×10^{-1}
BWOA	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}
LWOA	4.9293×10^{-2}	1.1696×10^{-2}	1.0756×10^{0}	1.8737×10^{-1}	1.8394×10^{1}	4.5449×10^{0}
IWOA	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	8.8944×10^{1}	1.3312×10^{2}
	F4		F5		F6	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	0.0000×10^{0}	0.0000×10^{0}	2.4313×10^{1}	6.6194×10^{0}	0.0000×10^{0}	0.0000×10^{0}
RDWOA	0.0000×10^{0}	0.0000×10^{0}	1.8882×10^{1}	5.1359×10^{0}	5.1469×10^{-15}	3.2926×10^{-15}
ACWOA	0.0000×10^{0}	0.0000×10^{0}	2.4274×10^{1}	4.5690×10^{0}	6.3093×10^{-4}	2.1127×10^{-4}
CCMWOA	4.3891×10^{-289}	0.0000×10^{0}	2.7607×10^{0}	7.6225×10^{0}	2.0854×10^{-2}	8.2250×10^{-3}
CWOA	8.4827×10^{0}	1.6542×10^{1}	2.5501×10^{1}	1.5480×10^{0}	1.0737×10^{-1}	1.6796×10^{-1}
BMWOA	4.4563×10^{-3}	6.7037×10^{-3}	1.2474×10^{-2}	3.0382×10^{-2}	1.2974×10^{-3}	1.8541×10^{-3}
BWOA	0.0000×10^{0}	0.0000×10^{0}	2.3788×10^{1}	6.4677×10^{0}	1.3716×10^{-4}	5.6219×10^{-5}
LWOA	3.5964×10^{-1}	9.6483×10^{-2}	4.8931×10^{1}	4.4527×10^{1}	5.8005×10^{-2}	1.4408×10^{-2}
IWOA	3.0373×10^{-4}	1.4320×10^{-3}	2.3521×10^{1}	7.0061×10^{-1}	3.5922×10^{-6}	1.7322×10^{-6}
	F7		F8		F9	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.7008×10^{-4}	2.2308×10^{-4}	-1.2569×10^{4}	2.8058×10^{-12}	0.0000×10^{0}	0.0000×10^{0}
RDWOA	2.8442×10^{-5}	3.6777×10^{-5}	-1.2521×10^{4}	1.6733×10^{2}	0.0000×10^{0}	0.0000×10^{0}
ACWOA	5.6623×10^{-6}	5.7698×10^{-6}	-1.2569×10^{4}	2.1881×10^{-3}	0.0000×10^{0}	0.0000×10^{0}
CCMWOA	1.9668×10^{-4}	1.6220×10^{-4}	-1.0928×10^{4}	9.5870×10^{2}	0.0000×10^{0}	0.0000×10^{0}
CWOA	3.1139×10^{-4}	3.9744×10^{-4}	-1.1583×10^{4}	1.6942×10^{3}	0.0000×10^{0}	0.0000×10^{0}
BMWOA	1.0619×10^{-3}	8.6629×10^{-4}	-1.2569×10^{4}	2.9396×10^{-3}	6.3549×10^{-4}	1.1849×10^{-3}
BWOA	2.5018×10^{-5}	3.0399×10^{-5}	-1.2357×10^{4}	4.2512×10^{2}	0.0000×10^{0}	0.0000×10^{0}
LWOA	1.2178×10^{-1}	4.6795×10^{-2}	-1.2382×10^{4}	4.4862×10^{2}	1.0110×10^{2}	2.6929×10^{1}
IWOA	2.6929×10^{-4}	3.2479×10^{-4}	-1.2298×10^{4}	7.5775×10^{2}	0.0000×10^{0}	0.0000×10^{0}

Table A4. *Cont.*

	F10		F11		F12	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
RDWOA	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	2.1668×10^{-7}	1.1868×10^{-6}
ACWOA	1.0066×10^{-15}	6.4863×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	6.7416×10^{-5}	1.9647×10^{-5}
CCMWOA	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	7.8157×10^{-4}	3.6365×10^{-4}
CWOA	3.0198×10^{-15}	2.0010×10^{-15}	0.0000×10^{0}	0.0000×10^{0}	4.4206×10^{-3}	6.6548×10^{-3}
BMWOA	5.1495×10^{-3}	4.7013×10^{-3}	2.1417×10^{-3}	4.0514×10^{-3}	1.5181×10^{-5}	2.4654×10^{-5}
BWOA	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.8415×10^{-5}	6.4484×10^{-6}
LWOA	6.6177×10^{-1}	6.9724×10^{-1}	1.4973×10^{-2}	1.3048×10^{-2}	5.5931×10^{-1}	1.1387×10^{0}
IWOA	2.5461×10^{-15}	2.0298×10^{-15}	1.8892×10^{-3}	1.0348×10^{-2}	5.1354×10^{-7}	1.4075×10^{-7}
	F13		F14		F15	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.3498×10^{-32}	5.5674×10^{-48}	5.3971×10^{6}	4.3864×10^{6}	1.1568×10^{5}	1.0038×10^{5}
RDWOA	3.6625×10^{-4}	2.0060×10^{-3}	1.0435×10^{7}	6.2580×10^{6}	2.2927×10^{7}	2.8097×10^{7}
ACWOA	2.5808×10^{-3}	4.6664×10^{-3}	1.4456×10^{8}	5.9824×10^{7}	7.4176×10^{9}	4.5829×10^{9}
CCMWOA	6.6551×10^{-4}	7.9798×10^{-4}	3.1814×10^{8}	1.2548×10^{8}	3.0720×10^{10}	7.7697×10^{9}
CWOA	5.3049×10^{-1}	4.3677×10^{-1}	6.6829×10^{7}	4.6037×10^{7}	2.0499×10^{9}	2.4020×10^{9}
BMWOA	1.2733×10^{-4}	2.2038×10^{-4}	1.0438×10^{8}	3.7729×10^{7}	2.8151×10^{8}	1.3106×10^{8}
BWOA	3.3556×10^{-3}	5.0182×10^{-3}	6.4289×10^{7}	2.9224×10^{7}	2.4303×10^{8}	1.3580×10^{8}
LWOA	2.1065×10^{-2}	7.2157×10^{-3}	3.7558×10^{6}	1.3953×10^{6}	5.2308×10^{5}	1.5679×10^{5}
IWOA	9.8357×10^{-6}	9.6866×10^{-6}	2.4173×10^{7}	1.2055×10^{7}	2.1211×10^{6}	3.5049×10^{6}
	F16		F17		F18	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	4.0353×10^{3}	2.8694×10^{3}	4.9768×10^{2}	4.3913×10^{1}	5.2008×10^{2}	6.1713×10^{-2}
RDWOA	6.3763×10^{3}	3.2741×10^{3}	5.3428×10^{2}	3.9426×10^{1}	5.2012×10^{2}	1.1564×10^{-1}
ACWOA	5.0442×10^{4}	6.7503×10^{3}	1.2586×10^{3}	2.9833×10^{2}	5.2085×10^{2}	1.1588×10^{-1}
CCMWOA	5.9083×10^{4}	9.2658×10^{3}	2.7400×10^{3}	1.0666×10^{3}	5.2088×10^{2}	1.7053×10^{-1}
CWOA	5.7234×10^{4}	3.7392×10^{4}	8.0857×10^{2}	2.2460×10^{2}	5.2029×10^{2}	1.3417×10^{-1}
BMWOA	5.7008×10^{4}	9.2535×10^{3}	6.7153×10^{2}	6.3683×10^{1}	5.2097×10^{2}	9.6222×10^{-2}
BWOA	3.2197×10^{4}	1.0637×10^{4}	6.9451×10^{2}	7.2749×10^{1}	5.2067×10^{2}	1.7262×10^{-1}
LWOA	1.0354×10^{3}	4.9837×10^{2}	5.0374×10^{2}	4.8236×10^{1}	5.2048×10^{2}	9.4869×10^{-2}
IWOA	1.6513×10^{4}	9.7529×10^{3}	5.7401×10^{2}	6.2791×10^{1}	5.2023×10^{2}	1.3567×10^{-1}
	F19		F20		F21	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	6.1886×10^{2}	2.8176×10^{0}	7.0051×10^{2}	2.4043×10^{-1}	8.0268×10^{2}	2.9674×10^{0}
RDWOA	6.2296×10^{2}	3.2376×10^{0}	7.0097×10^{2}	2.3043×10^{-1}	8.4662×10^{2}	1.2443×10^{1}
ACWOA	6.3380×10^{2}	2.8192×10^{0}	7.4705×10^{2}	2.6949×10^{1}	9.9173×10^{2}	2.8423×10^{1}
CCMWOA	6.3421×10^{2}	3.1895×10^{0}	9.1058×10^{2}	6.8146×10^{1}	1.0329×10^{3}	2.5834×10^{1}
CWOA	6.3619×10^{2}	2.3328×10^{0}	7.1987×10^{2}	1.9833×10^{1}	9.8919×10^{2}	3.5017×10^{1}
BMWOA	6.3209×10^{2}	3.4550×10^{0}	7.0298×10^{2}	7.9463×10^{-1}	9.6429×10^{2}	2.2995×10^{1}
BWOA	6.3659×10^{2}	2.7733×10^{0}	7.0210×10^{2}	4.7824×10^{-1}	9.6653×10^{2}	2.0133×10^{1}
LWOA	6.2990×10^{2}	3.7057×10^{0}	7.0071×10^{2}	1.0660×10^{-1}	8.7689×10^{2}	1.4956×10^{1}
IWOA	6.2837×10^{2}	3.5182×10^{0}	7.0086×10^{2}	1.8091×10^{-1}	9.1853×10^{2}	2.3903×10^{1}

Table A4. *Cont.*

	F22		F23		F24	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.0349×10^3	3.5053×10^1	1.0502×10^3	7.2638×10^1	3.9773×10^3	5.2239×10^2
RDWOA	1.0862×10^3	3.9406×10^1	1.6401×10^3	2.1508×10^2	4.8765×10^3	4.8011×10^2
ACWOA	1.1353×10^3	2.7084×10^1	4.6113×10^3	7.8646×10^2	6.0951×10^3	8.4619×10^2
CCMWOA	1.1585×10^3	2.0571×10^1	5.7676×10^3	4.6957×10^2	7.0634×10^3	8.3597×10^2
CWOA	1.1502×10^3	5.9550×10^1	5.0935×10^3	8.0866×10^2	6.4607×10^3	8.0167×10^2
BMWOA	1.1247×10^3	3.0558×10^1	4.7543×10^3	7.0975×10^2	7.1260×10^3	9.0188×10^2
BWOA	1.1033×10^3	2.3404×10^1	4.8599×10^3	7.9435×10^2	6.5557×10^3	1.0588×10^3
LWOA	1.1231×10^3	4.1332×10^1	2.1055×10^3	5.0404×10^2	5.3203×10^3	5.2375×10^2
IWOA	1.1290×10^3	5.0219×10^1	2.6021×10^3	4.6117×10^2	5.5791×10^3	7.2655×10^2
	F25		F26		F27	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.2002×10^3	5.1670×10^{-2}	1.3005×10^3	1.1585×10^{-1}	1.4003×10^3	4.7444×10^{-2}
RDWOA	1.2005×10^3	1.8157×10^{-1}	1.3005×10^3	1.0242×10^{-1}	1.4002×10^3	3.7231×10^{-2}
ACWOA	1.2017×10^3	5.3455×10^{-1}	1.3011×10^3	8.3617×10^{-1}	1.4239×10^3	1.2976×10^1
CCMWOA	1.2018×10^3	4.5129×10^{-1}	1.3041×10^3	8.3728×10^{-1}	1.4661×10^3	1.6380×10^1
CWOA	1.2018×10^3	5.2022×10^{-1}	1.3006×10^3	1.2049×10^{-1}	1.4102×10^3	1.2849×10^1
BMWOA	1.2023×10^3	4.1490×10^{-1}	1.3005×10^3	1.1904×10^{-1}	1.4003×10^3	1.0699×10^{-1}
BWOA	1.2019×10^3	4.8775×10^{-1}	1.3005×10^3	1.3303×10^{-1}	1.4003×10^3	4.2027×10^{-2}
LWOA	1.2008×10^3	3.0020×10^{-1}	1.3005×10^3	1.1102×10^{-1}	1.4003×10^3	9.9959×10^{-2}
IWOA	1.2010×10^3	2.9781×10^{-1}	1.3005×10^3	9.8276×10^{-2}	1.4003×10^3	5.2298×10^{-2}
	F28		F29		F30	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.5195×10^3	5.8976×10^0	1.6103×10^3	7.8355×10^{-1}	1.4951×10^6	1.0794×10^6
RDWOA	1.5215×10^3	7.2912×10^0	1.6116×10^3	5.9134×10^{-1}	1.1935×10^6	1.0797×10^6
ACWOA	1.8396×10^3	4.0931×10^2	1.6121×10^3	4.8464×10^{-1}	1.3124×10^7	8.0308×10^6
CCMWOA	7.0062×10^3	4.0693×10^3	1.6130×10^3	3.2081×10^{-1}	2.6816×10^7	1.9260×10^7
CWOA	1.9731×10^3	8.2350×10^2	1.6127×10^3	5.6484×10^{-1}	9.8463×10^6	8.8774×10^6
BMWOA	1.5828×10^3	3.6612×10^1	1.6126×10^3	2.1626×10^{-1}	6.9174×10^6	4.9113×10^6
BWOA	1.6258×10^3	4.8685×10^1	1.6124×10^3	4.8342×10^{-1}	7.5640×10^6	5.4938×10^6
LWOA	1.5213×10^3	4.9669×10^0	1.6125×10^3	5.6375×10^{-1}	4.9220×10^5	2.9566×10^5
IWOA	1.5506×10^3	1.6210×10^1	1.6125×10^3	5.1867×10^{-1}	2.7873×10^6	2.0454×10^6
	F31		F32		F33	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	9.2541×10^3	2.1036×10^4	1.9198×10^3	2.4744×10^1	8.8237×10^3	5.3819×10^3
RDWOA	4.8557×10^3	3.4855×10^3	1.9194×10^3	2.6942×10^1	6.7743×10^3	3.5180×10^3
ACWOA	4.4984×10^7	4.3109×10^7	2.0047×10^3	3.3162×10^1	3.7129×10^4	1.9130×10^4
CCMWOA	9.8440×10^7	1.2615×10^8	2.0824×10^3	5.0281×10^1	5.7205×10^4	2.2855×10^4
CWOA	3.9424×10^6	1.2057×10^7	2.0018×10^3	6.3131×10^1	5.7907×10^4	5.9437×10^4
BMWOA	1.1037×10^5	1.2099×10^5	1.9467×10^3	4.0178×10^1	3.3436×10^4	1.7890×10^4
BWOA	1.1889×10^5	3.5142×10^5	1.9593×10^3	3.8145×10^1	3.2143×10^4	1.6907×10^4
LWOA	1.0695×10^4	5.9845×10^3	1.9230×10^3	2.4187×10^1	3.0576×10^3	7.5160×10^2
IWOA	5.4855×10^3	4.2910×10^3	1.9348×10^3	3.5960×10^1	1.6411×10^4	1.0072×10^4

Table A4. *Cont.*

Overall rank	F34		F35		overall	
	avg	*std*	*avg*	*std*	+/−/=	rank
DECCWOA	1.0426×10^{6}	8.4796×10^{5}	2.8721×10^{3}	2.1610×10^{2}	~	1
RDWOA	4.2175×10^{5}	3.3446×10^{5}	2.7874×10^{3}	2.1665×10^{2}	16/6/13	2
ACWOA	4.2559×10^{6}	3.6243×10^{6}	3.0278×10^{3}	2.2024×10^{2}	26/3/6	6
CCMWOA	8.7004×10^{6}	5.9791×10^{6}	3.2984×10^{3}	4.4928×10^{2}	28/2/5	9
CWOA	3.1633×10^{6}	2.9730×10^{6}	3.1092×10^{3}	2.3259×10^{2}	29/0/6	8
BMWOA	1.0736×10^{6}	9.0915×10^{5}	3.0014×10^{3}	2.7107×10^{2}	32/1/2	7
BWOA	1.9551×10^{6}	1.5899×10^{6}	2.9774×10^{3}	2.8735×10^{2}	23/3/9	4
LWOA	2.0517×10^{5}	1.6837×10^{5}	2.9007×10^{3}	2.4887×10^{2}	26/4/5	5
IWOA	9.3081×10^{5}	7.7249×10^{5}	2.9329×10^{3}	1.7696×10^{2}	24/1/10	3

Table A5. Comparison results for the DECCWOA with advanced algorithms.

	F1		F2		F3	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.0221×10^{-18}	4.2975×10^{-18}
IGWO	0.0000×10^{0}	0.0000×10^{0}	3.6328×10^{-261}	0.0000×10^{0}	1.3124×10^{-86}	7.1881×10^{-86}
OBLGWO	0.0000×10^{0}	0.0000×10^{0}	3.6589×10^{-142}	2.004×10^{-141}	6.2014×10^{-293}	0.0000×10^{0}
CGPSO	2.3583×10^{-8}	7.7088×10^{-8}	3.9726×10^{-5}	2.8781×10^{-5}	6.3491×10^{-2}	5.1833×10^{-2}
ALPSO	1.1539×10^{-184}	0.0000×10^{0}	2.5959×10^{-8}	7.1555×10^{-8}	2.2102×10^{-11}	2.9723×10^{-11}
RCBA	8.9446×10^{-3}	2.9769×10^{-3}	5.8765×10^{-1}	8.4909×10^{-2}	2.1948×10^{0}	5.2552×10^{-1}
CBA	7.2954×10^{-8}	3.8213×10^{-7}	4.1161×10^{1}	1.3912×10^{2}	1.3118×10^{1}	6.5496×10^{0}
OBSCA	1.0911×10^{-103}	5.5402×10^{-103}	4.3833×10^{-91}	1.1161×10^{-90}	3.1617×10^{-24}	1.1702×10^{-23}
SCADE	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}
	F4		F5		F6	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.0221×10^{-18}	4.2975×10^{-18}	2.6083×10^{1}	3.1418×10^{-1}	0.0000×10^{0}	0.0000×10^{0}
IGWO	1.3124×10^{-86}	7.1881×10^{-86}	2.3216×10^{1}	1.8144×10^{-1}	1.2448×10^{-5}	3.5159×10^{-6}
OBLGWO	6.2014×10^{-293}	0.0000×10^{0}	2.6052×10^{1}	3.8656×10^{-1}	3.9085×10^{-5}	1.4498×10^{-5}
CGPSO	6.3491×10^{-2}	5.1833×10^{-2}	1.0747×10^{-7}	1.4040×10^{-7}	1.5149×10^{-8}	2.7356×10^{-8}
ALPSO	2.2102×10^{-11}	2.9723×10^{-11}	3.5496×10^{1}	3.2473×10^{1}	5.9288×10^{-31}	2.2626×10^{-30}
RCBA	2.1948×10^{0}	5.2552×10^{-1}	3.6041×10^{1}	4.0444×10^{1}	8.7533×10^{-3}	2.4284×10^{-3}
CBA	1.3118×10^{1}	6.5496×10^{0}	7.3423×10^{1}	1.2319×10^{2}	4.4526×10^{-7}	2.4194×10^{-6}
OBSCA	3.1617×10^{-24}	1.1702×10^{-23}	2.7647×10^{1}	3.8007×10^{-1}	3.8321×10^{0}	2.7513×10^{-1}
SCADE	0.0000×10^{0}	0.0000×10^{0}	1.5398×10^{1}	1.3017×10^{1}	1.7996×10^{-7}	1.6508×10^{-7}
	F7		F8		F9	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.1896×10^{-4}	1.4262×10^{-4}	-1.3066×10^{4}	2.6313×10^{3}	0.0000×10^{0}	0.0000×10^{0}
IGWO	2.9290×10^{-4}	2.6976×10^{-4}	-7.4319×10^{3}	6.6317×10^{2}	0.0000×10^{0}	0.0000×10^{0}
OBLGWO	2.4381×10^{-5}	2.9727×10^{-5}	-1.2561×10^{4}	4.4545×10^{1}	0.0000×10^{0}	0.0000×10^{0}
CGPSO	1.4906×10^{-5}	1.4183×10^{-5}	-3.7698×10^{4}	6.6756×10^{3}	3.0143×10^{-9}	6.2053×10^{-9}
ALPSO	7.8389×10^{-2}	3.1754×10^{-2}	-1.1531×10^{4}	2.8700×10^{2}	1.9471×10^{1}	7.9710×10^{0}
RCBA	1.1712×10^{-1}	5.5739×10^{-2}	-7.3244×10^{3}	5.4651×10^{2}	2.0111×10^{1}	4.6024×10^{0}
CBA	1.5885×10^{-1}	3.4560×10^{-1}	-7.3445×10^{3}	6.5505×10^{2}	1.2498×10^{2}	4.7753×10^{1}
OBSCA	8.1175×10^{-4}	5.3137×10^{-4}	-4.1274×10^{3}	2.4305×10^{2}	0.0000×10^{0}	0.0000×10^{0}
SCADE	2.9509×10^{-4}	2.0997×10^{-4}	-1.2569×10^{4}	1.1550×10^{-2}	0.0000×10^{0}	0.0000×10^{0}

Table A5. *Cont.*

	F10		F11		F12	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	1.5705×10^{-32}	5.5674×10^{-48}
IGWO	4.9146×10^{-15}	1.2283×10^{-15}	0.0000×10^{0}	0.0000×10^{0}	1.1169×10^{-6}	3.8305×10^{-7}
OBLGWO	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	3.8858×10^{-4}	1.1452×10^{-3}
CGPSO	1.8069×10^{-5}	1.2953×10^{-5}	4.3701×10^{-8}	7.1267×10^{-8}	6.2743×10^{-11}	1.4249×10^{-10}
ALPSO	8.0156×10^{-1}	8.3429×10^{-1}	1.6465×10^{-2}	1.5344×10^{-2}	3.0222×10^{-2}	8.1300×10^{-2}
RCBA	1.0853×10^{-1}	2.7602×10^{-2}	1.0473×10^{-2}	1.0208×10^{-2}	9.1885×10^{0}	2.8806×10^{0}
CBA	1.5880×10^{1}	2.1141×10^{0}	1.3514×10^{-2}	1.8331×10^{-2}	1.4396×10^{1}	4.7131×10^{0}
OBSCA	4.3225×10^{-15}	6.4863×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	3.8964×10^{-1}	4.5185×10^{-2}
SCADE	8.8818×10^{-16}	0.0000×10^{0}	0.0000×10^{0}	0.0000×10^{0}	5.6904×10^{-9}	4.8080×10^{-9}
	F13		F14		F15	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.3498×10^{-32}	5.5674×10^{-48}	4.2212×10^{6}	3.5710×10^{6}	1.6057×10^{5}	1.7381×10^{5}
IGWO	1.4838×10^{-2}	2.6985×10^{-2}	1.7924×10^{7}	6.8885×10^{6}	2.3989×10^{6}	1.4218×10^{6}
OBLGWO	5.7997×10^{-2}	7.7859×10^{-2}	1.7531×10^{7}	1.0402×10^{7}	1.3307×10^{7}	1.0251×10^{7}
CGPSO	3.1353×10^{-9}	9.6561×10^{-9}	9.8321×10^{6}	2.5274×10^{6}	1.5880×10^{8}	1.6828×10^{7}
ALPSO	2.8346×10^{-2}	9.8785×10^{-2}	5.7145×10^{6}	5.4783×10^{6}	2.9466×10^{3}	3.6264×10^{3}
RCBA	5.1201×10^{-3}	4.6593×10^{-3}	1.0422×10^{6}	3.8825×10^{5}	2.5206×10^{4}	1.0389×10^{4}
CBA	3.1616×10^{1}	2.6961×10^{1}	4.7886×10^{6}	1.8780×10^{6}	1.2467×10^{4}	1.0840×10^{4}
OBSCA	2.1646×10^{0}	1.1588×10^{-1}	3.9908×10^{8}	1.1211×10^{8}	2.4969×10^{10}	3.6691×10^{9}
SCADE	8.3767×10^{-8}	6.6478×10^{-8}	4.7479×10^{8}	9.8759×10^{7}	2.9269×10^{10}	4.2012×10^{9}
	F16		F17		F18	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	7.8122×10^{3}	5.4762×10^{3}	5.1647×10^{2}	4.8965×10^{1}	5.2007×10^{2}	5.1181×10^{-2}
IGWO	7.2399×10^{3}	2.0340×10^{3}	5.2142×10^{2}	3.0682×10^{1}	5.2050×10^{2}	1.4090×10^{-1}
OBLGWO	1.0074×10^{4}	3.8144×10^{3}	5.4284×10^{2}	3.5930×10^{1}	5.2095×10^{2}	5.4001×10^{-2}
CGPSO	2.3365×10^{3}	5.0383×10^{2}	4.6884×10^{2}	3.1903×10^{1}	5.2098×10^{2}	4.2868×10^{-2}
ALPSO	3.7450×10^{2}	1.3166×10^{2}	5.4221×10^{2}	5.6842×10^{1}	5.2080×10^{2}	5.8119×10^{-2}
RCBA	3.2947×10^{2}	1.2959×10^{1}	4.7036×10^{2}	3.7390×10^{1}	5.2010×10^{2}	9.5615×10^{-2}
CBA	4.7703×10^{3}	9.2720×10^{3}	4.9781×10^{2}	4.4714×10^{1}	5.2009×10^{2}	1.3697×10^{-1}
OBSCA	5.3939×10^{4}	6.9272×10^{3}	2.2087×10^{3}	5.3362×10^{2}	5.2096×10^{2}	5.8056×10^{-2}
SCADE	5.4785×10^{4}	6.7546×10^{3}	2.2800×10^{3}	4.7955×10^{2}	5.2094×10^{2}	5.3187×10^{-2}
	F19		F20		F21	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	6.1885×10^{2}	2.4155×10^{0}	7.0061×10^{2}	3.1716×10^{-1}	8.0279×10^{2}	8.0509×10^{0}
IGWO	6.1887×10^{2}	2.6487×10^{0}	7.0099×10^{2}	5.1420×10^{-2}	8.8181×10^{2}	1.6752×10^{1}
OBLGWO	6.1968×10^{2}	4.3347×10^{0}	7.0117×10^{2}	9.6122×10^{-2}	9.2963×10^{2}	3.9241×10^{1}
CGPSO	6.2402×10^{2}	2.9481×10^{0}	7.0241×10^{2}	2.0187×10^{-1}	9.8743×10^{2}	2.5413×10^{1}
ALPSO	6.1705×10^{2}	2.4439×10^{0}	7.0001×10^{2}	8.9890×10^{-3}	8.2142×10^{2}	9.5753×10^{0}
RCBA	6.3882×10^{2}	3.5284×10^{0}	7.0007×10^{2}	1.8671×10^{-2}	1.0209×10^{3}	3.9995×10^{1}
CBA	6.4033×10^{2}	2.8154×10^{0}	7.0003×10^{2}	3.4856×10^{-2}	1.0228×10^{3}	6.1207×10^{1}
OBSCA	6.3161×10^{2}	1.3106×10^{0}	9.0597×10^{2}	3.6858×10^{1}	1.0643×10^{3}	1.7737×10^{1}
SCADE	6.3355×10^{2}	2.2424×10^{0}	8.9018×10^{2}	3.3322×10^{1}	1.0695×10^{3}	1.2949×10^{1}

Table A5. *Cont.*

	F22		F23		F24	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.0366×10^{3}	3.6638×10^{1}	1.0472×10^{3}	5.2652×10^{1}	3.6010×10^{3}	4.5441×10^{2}
IGWO	1.0087×10^{3}	1.9673×10^{1}	3.3271×10^{3}	4.7812×10^{2}	4.6073×10^{3}	7.5057×10^{2}
OBLGWO	1.0661×10^{3}	4.0353×10^{1}	4.0667×10^{3}	1.0107×10^{3}	5.5480×10^{3}	1.0343×10^{3}
CGPSO	1.1225×10^{3}	2.5722×10^{1}	5.5982×10^{3}	5.0945×10^{2}	6.0825×10^{3}	5.5185×10^{2}
ALPSO	1.0021×10^{3}	2.9422×10^{1}	1.6121×10^{3}	3.2630×10^{2}	4.0735×10^{3}	5.3691×10^{2}
RCBA	1.1638×10^{3}	6.3597×10^{1}	5.6153×10^{3}	6.4055×10^{2}	5.8506×10^{3}	8.1461×10^{2}
CBA	1.1526×10^{3}	7.2911×10^{1}	5.7931×10^{3}	7.2017×10^{2}	5.9590×10^{3}	7.1736×10^{2}
OBSCA	1.1929×10^{3}	1.5671×10^{1}	6.1341×10^{3}	3.2876×10^{2}	7.2777×10^{3}	4.0271×10^{2}
SCADE	1.2049×10^{3}	1.8580×10^{1}	7.4883×10^{3}	2.3162×10^{2}	8.2006×10^{3}	2.9486×10^{2}
	F25		F26		F27	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.2002×10^{3}	7.1972×10^{-2}	1.3005×10^{3}	1.4836×10^{-1}	1.4003×10^{3}	4.2207×10^{-2}
IGWO	1.2008×10^{3}	3.5285×10^{-1}	1.3006×10^{3}	1.2395×10^{-1}	1.4004×10^{3}	2.5832×10^{-1}
OBLGWO	1.2023×10^{3}	6.8209×10^{-1}	1.3005×10^{3}	1.1878×10^{-1}	1.4005×10^{3}	2.3451×10^{-1}
CGPSO	1.2025×10^{3}	2.0362×10^{-1}	1.3004×10^{3}	1.0032×10^{-1}	1.4003×10^{3}	1.2353×10^{-1}
ALPSO	1.2013×10^{3}	5.4330×10^{-1}	1.3005×10^{3}	7.9368×10^{-2}	1.4006×10^{3}	2.8021×10^{-1}
RCBA	1.2006×10^{3}	3.7833×10^{-1}	1.3005×10^{3}	1.3700×10^{-1}	1.4003×10^{3}	9.7996×10^{-2}
CBA	1.2011×10^{3}	7.3537×10^{-1}	1.3005×10^{3}	1.4823×10^{-1}	1.4003×10^{3}	1.5708×10^{-1}
OBSCA	1.2023×10^{3}	4.4892×10^{-1}	1.3036×10^{3}	2.6704×10^{-1}	1.4695×10^{3}	1.1814×10^{1}
SCADE	1.2026×10^{3}	2.4794×10^{-1}	1.3039×10^{3}	2.9836×10^{-1}	1.4881×10^{3}	1.3091×10^{1}
	F28		F29		F30	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	1.5202×10^{3}	6.6341×10^{0}	1.6102×10^{3}	7.9146×10^{-1}	1.8690×10^{6}	1.1246×10^{6}
IGWO	1.5176×10^{3}	3.7863×10^{0}	1.6116×10^{3}	5.8466×10^{-1}	9.2251×10^{5}	5.5675×10^{5}
OBLGWO	1.5150×10^{3}	5.7646×10^{0}	1.6120×10^{3}	6.3233×10^{-1}	1.3955×10^{6}	1.0330×10^{6}
CGPSO	1.5176×10^{3}	1.3084×10^{0}	1.6117×10^{3}	3.2922×10^{-1}	3.3985×10^{5}	1.7015×10^{5}
ALPSO	1.5115×10^{3}	4.1448×10^{0}	1.6118×10^{3}	3.2864×10^{-1}	5.5179×10^{5}	4.6710×10^{5}
RCBA	1.5371×10^{3}	9.0984×10^{0}	1.6135×10^{3}	3.4738×10^{-1}	1.2165×10^{5}	7.5279×10^{4}
CBA	1.5589×10^{3}	1.7899×10^{1}	1.6133×10^{3}	5.2338×10^{-1}	1.8619×10^{5}	1.2340×10^{5}
OBSCA	1.4085×10^{4}	8.8848×10^{3}	1.6129×10^{3}	2.5975×10^{-1}	1.1278×10^{7}	5.7586×10^{6}
SCADE	1.8666×10^{4}	6.8024×10^{3}	1.6127×10^{3}	1.7040×10^{-1}	1.5787×10^{7}	8.3755×10^{6}
	F31		F32		F33	
	avg	*std*	*avg*	*std*	*avg*	*std*
DECCWOA	4.3650×10^{3}	3.0442×10^{3}	1.9120×10^{3}	1.2480×10^{1}	7.8097×10^{3}	4.1959×10^{3}
IGWO	1.7080×10^{4}	2.1685×10^{4}	1.9180×10^{3}	1.4497×10^{1}	3.0438×10^{3}	6.5561×10^{2}
OBLGWO	8.3234×10^{4}	1.7925×10^{5}	1.9215×10^{3}	2.3356×10^{1}	5.5656×10^{3}	2.6114×10^{3}
CGPSO	2.4871×10^{6}	7.6618×10^{5}	1.9170×10^{3}	2.9535×10^{0}	2.4762×10^{3}	1.6062×10^{2}
ALPSO	7.8595×10^{3}	7.3812×10^{3}	1.9170×10^{3}	2.0884×10^{1}	3.0087×10^{3}	4.2177×10^{2}
RCBA	6.9919×10^{3}	7.0401×10^{3}	1.9292×10^{3}	2.9262×10^{1}	2.4379×10^{3}	1.3813×10^{2}
CBA	9.7529×10^{3}	9.7686×10^{3}	1.9246×10^{3}	2.6288×10^{1}	2.9663×10^{3}	1.2059×10^{3}
OBSCA	1.5585×10^{8}	1.0228×10^{8}	2.0091×10^{3}	1.5224×10^{1}	3.1925×10^{4}	1.3827×10^{4}
SCADE	1.9277×10^{8}	9.7379×10^{7}	2.0133×10^{3}	1.3639×10^{1}	2.7694×10^{4}	1.1641×10^{4}

Table A5. *Cont.*

Overall rank	F34		F35		overall	
	avg	*std*	*avg*	*std*	+/−/=	rank
DECCWOA	5.6266×10^5	5.3759×10^5	2.8005×10^3	2.0373×10^2	~	1
IGWO	2.5893×10^5	2.2750×10^5	2.5846×10^3	1.4319×10^2	20/8/7	2
OBLGWO	5.1886×10^5	3.8300×10^5	2.6930×10^3	1.9962×10^2	19/5/11	5
CGPSO	1.2464×10^5	6.8764×10^4	2.9020×10^3	2.0995×10^2	23/9/3	4
ALPSO	1.1929×10^5	2.9474×10^5	2.7316×10^3	2.0662×10^2	19/9/7	3
RCBA	8.3440×10^4	3.8322×10^4	3.3862×10^3	3.5129×10^2	23/9/3	6
CBA	1.2013×10^5	7.8959×10^4	3.4067×10^3	2.6854×10^2	24/6/5	7
OBSCA	1.9523×10^6	9.0846×10^5	3.1622×10^3	1.4760×10^2	32/1/2	9
SCADE	2.4532×10^6	1.2033×10^6	3.1167×10^3	1.2839×10^2	28/2/5	7

Table A6. Description of each attribute for the talent stability data.

Attributes	Name	Description
F1	Sex	1 for male and 2 for female.
F2	Political affiliation	There are five categories: Communist Party members, reserve party members, democratic party members, Communist Youth League members and the masses, denoted by 1, 2, 3, 4 and 13, respectively.
F3	Professional attributes	1 indicates arts, 2 indicates science and 3 indicates less than junior college (junior college not divided into arts and science subjects)
F4	Age	Ages 25–30, 31–35, 36–40, 41–45, 46–50, 51–55 and 56–60 are indicated by 1, 2, 3, 4, 5, 6 and 7, respectively. Young and middle-aged people have a strong level of competence and a strong tendency to move because of upward mobility, life pressures, etc.
F5	Household Registration	There are three categories: in-city, in-province and out-of-province, indicated by 0, 1 and 2, respectively.
F6	Type of place of origin	There are three categories: urban, township and rural, denoted by 1, 2 and 3, respectively.
F7	City-level and above talent categories	There are categories A, B, C, D and E, denoted by 1, 2, 3, 4 and 5, respectively. 6 is for talent category F and no talent category is denoted by 10.
F8	Nature of previous unit	0 indicating pending employment, 10 indicating state institutions, 20 indicating scientific research institutions, 21 indicating higher education institutions, 22 indicating secondary and junior education institutions, 23 indicating health and medical institutions, 29 indicating other institutions, 31 indicating state-owned enterprises, 32 indicating foreign-funded enterprises, 39 indicating private enterprises, 40 indicating the army, 55 indicating rural organizations, and 99 indicating self-employment. No previous unit is denoted by 100.
F9	Wenzhou colleges and university's location type	Prefectural level cities, denoted by 2.
F10	Year of employment at Wenzhou colleges and universities	This is a measure of stability in the unit of employment. 1 is used for entry before 2000 (merger), 2 for entry from 2001–2006 (preparation), 3 for entry from 2007–2008 (de-preparation), 4 for entry from 2009–2014 (school introduction policy), 5 for entry from 2015–2017 (city introduction policy) and 6 for entry from 2018. To date (increased introduction by the school) entry is indicated by 6. (It can also be described in terms of stable years 3, 4–6, 7–10, 11+ years).
F11	Types of positions at Wenzhou colleges and universities	Teaching staff are represented by 24, PhD students and research staff by 11, professional and technical staff by 29, administrative staff by 101 and counsellors by 102.

Table A6. *Cont.*

Attributes	Name	Description
F12	Professional relevance of employment at Wenzhou colleges and universities	It is used to measure the relevance of the major studied to the job, with higher percentages indicating higher relevance.
F13	Monthly salary level for employment at Wenzhou colleges and universities: RMB	It is used to measure the average monthly salary received, with higher values indicating higher salary levels.
F14	Current employment	Current employment is indicated by 1 for Wenzhou undergraduate institutions; 2 for civil servants or institutions; 3 for undergraduate institutions (including doctoral studies); 4 for vocational institutions in other cities; 5 for vocational institutions in the city, 6 for enterprises, 7 for going abroad and 8 for pending employment.
F15	Time of introduction at current employment	Indicated by 1 for entry before 2000 (merger), 2 for entry from 2001–2006 (preparation), 3 for entry from 2007–2008 (de-preparation), 4 for entry from 2009–2014 (school introduction policy), 5 for entry from 2015–2017 (city introduction policy) and 6 for entry from 2018-present (increased school introduction).
F16	Nature of current employment	0 indicating pending employment, 10 indicating state institutions, 20 indicating scientific research institutions, 21 indicating higher education institutions, 22 indicating secondary and junior education institutions, 23 indicating health and medical institutions, 29 indicating other institutions, 31 indicating state-owned enterprises, 32 indicating foreign-funded enterprises, 39 indicating private enterprises, 40 indicating the army, 55 indicating rural organizations and 99 indicating self-employment.
F17	Type of location of current employment unit	Pending employment is represented by 0, sub-provincial and large cities by 1, prefecture-level cities by 2 and counties and villages by 3.
F18	Type of current employment	The type of position currently employed is expressed in the same way as the type of position in the previous employment unit indicated in F11. Pending employment is indicated by 0, civil servants by 10, doctoral students and researchers by 11, engineers and technicians by 13, teaching staff by 24, professional and technical staff by 29, commercial service staff and clerks by 30, military personnel by 80, administrative staff by 101 and counsellors by 102.
F19	Relevance of current employment profession	The professional relevance of current employment is expressed in the same way as the type of position in the previous employment unit indicated by F11.
F20	Monthly salary level in current employment unit: RMB	The current employment monthly salary level is expressed in the same way as the previous employment monthly salary level in F13.
F21	Salary differential	It is used to measure the change in the monthly salary of the current employment unit from that of the previous employment unit, that is, the difference between the monthly salary level of the current employment unit expressed in F21 and the monthly salary level of the previous employment unit expressed in F13, with a larger value indicating a larger increase in monthly salary.
F22	Professional and technical position at the time of leaving	Positive senior, deputy senior, intermediate, primary and none are represented by 1, 2, 3, 4, and 5, respectively.
F23	Double first-rate	1 means double first-rate, 2 means not.
F24	Highest Education	College, university and postgraduate are denoted by 0, 1 and 2, respectively. Below junior college, it is denoted by 5.
F25	Highest degrees	Tertiary, bachelor, master and doctoral degrees are denoted by 0, 1, 2 and 3, respectively. Below the tertiary level, they are denoted by 5.
F26	Change in place of employment	A variation is indicated by 1 and no variation is indicated by 0.

References

1. Yang, J. The Theory of Planned Behavior and Prediction of Entrepreneurial Intention Among Chinese Undergraduates. *Soc. Behav. Pers. Int. J.* **2013**, *41*, 367–376. [CrossRef]
2. González-Serrano, M.H.; Moreno, F.C.; Hervás, J.C. Prediction model of the entrepreneurial intentions in pre-graduated and post-graduated Sport Sciences students. *Cult. Cienc. Y Deporte* **2018**, *13*, 219–230. [CrossRef]
3. Gorgievski, M.J.; Stephan, U.; Laguna, M.; Moriano, J.A. Predicting Entrepreneurial Career Intentions: Values and the theory of planned behavior. *J. Career Assess.* **2018**, *26*, 457–475. [CrossRef] [PubMed]
4. Nawaz, T.; Khattak, B.K.; Rehman, K. New look of predicting entrepreneurial intention: A serial mediation analysis. *Dilemas Contemp. Educ. Polit. Y Valor.* **2019**, *6*, 126.
5. Yang, F. Decision Tree Algorithm Based University Graduate Employment Trend Prediction. *Informatica* **2019**, *43*. [CrossRef]
6. Djordjevic, D.; Cockalo, D.; Bogetic, S.; Bakator, M. Predicting Entrepreneurial Intentions among the Youth in Serbia with a Classification Decision Tree Model with the QUEST Algorithm. *Mathematics* **2021**, *9*, 1487. [CrossRef]
7. Wei, Y.; Lv, H.; Chen, M.; Wang, M.; Heidari, A.A.; Chen, H.; Li, C. Predicting Entrepreneurial Intention of Students: An Extreme Learning Machine with Gaussian Barebone Harris Hawks Optimizer. *IEEE Access* **2020**, *8*, 76841–76855. [CrossRef]
8. Bhagavan, K.S.; Thangakumar, J.; Subramanian, D.V. RETRACTED ARTICLE: Predictive analysis of student academic performance and employability chances using HLVQ algorithm. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *12*, 3789–3797. [CrossRef]
9. Huang, Z.; Liu, G. Prediction model of college students entrepreneurship ability based on artificial intelligence and fuzzy logic model. *J. Intell. Fuzzy Syst.* **2021**, *40*, 2541–2552. [CrossRef]
10. Li, X.; Yang, T. Forecast of the Employment Situation of College Graduates Based on the LSTM Neural Network. *Comput. Intell. Neurosci.* **2021**, *2021*, 5787355. [CrossRef]
11. Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]
12. Li, J.; Guo, L.; Li, Y.; Liu, C.; Wang, L.; Hu, H. Enhancing Whale Optimization Algorithm with Chaotic Theory for Permutation Flow Shop Scheduling Problem. *Int. J. Comput. Intell. Syst.* **2021**, *14*, 651–675. [CrossRef]
13. Luan, F.; Cai, Z.; Wu, S.; Jiang, T.; Li, F.; Yang, J. Improved Whale Algorithm for Solving the Flexible Job Shop Scheduling Problem. *Mathematics* **2019**, *7*, 384. [CrossRef]
14. Navarro, M.A.; Oliva, D.; Ramos-Michel, A.; Zaldívar, D.; Morales-Castañeda, B.; Pérez-Cisneros, M.; Valdivia, A.; Chen, H. An improved multi-population whale optimization algorithm. *Int. J. Mach. Learn. Cybern.* **2022**, *13*, 2447–2478. [CrossRef]
15. Abbas, S.; Jalil, Z.; Javed, A.R.; Batool, I.; Khan, M.Z.; Noorwali, A.; Gadekallu, T.R.; Akbar, A. BCD-WERT: A novel approach for breast cancer detection using whale optimization based efficient features and extremely randomized tree algorithm. *PeerJ Comput. Sci.* **2021**, *7*, e390. [CrossRef]
16. Elaziz, M.A.; Nabil, N.; Moghdani, R.; Ewees, A.A.; Cuevas, E.; Lu, S. Multilevel thresholding image segmentation based on improved volleyball premier league algorithm using whale optimization algorithm. *Multimed. Tools Appl.* **2021**, *80*, 12435–12468. [CrossRef]
17. Abdel-Basset, M.; El-Shahat, D.; El-Henawy, I. A modified hybrid whale optimization algorithm for the scheduling problem in multimedia data objects. *Concurr. Comput. Pr. Exp.* **2020**, *32*, e5137. [CrossRef]
18. Qiao, S.; Yu, H.; Heidari, A.A.; El-Saleh, A.A.; Cai, Z.; Xu, X.; Mafarja, M.; Chen, H. Individual disturbance and neighborhood mutation search enhanced whale optimization: Performance design for engineering problems. *J. Comput. Des. Eng.* **2022**, *9*, 1817–1851. [CrossRef]
19. Peng, L.; He, C.; Heidari, A.A.; Zhang, Q.; Chen, H.; Liang, G.; Aljehane, N.O.; Mansour, R.F. Information sharing search boosted whale optimizer with Nelder-Mead simplex for parameter estimation of photovoltaic models. *Energy Convers. Manag.* **2022**, *270*, 116246. [CrossRef]
20. Abderazek, H.; Hamza, F.; Yildiz, A.R.; Sait, S.M. Comparative investigation of the moth-flame algorithm and whale optimization algorithm for optimal spur gear design. *Mater. Test.* **2021**, *63*, 266–271. [CrossRef]
21. Ahmadianfar, I.; Asghar Heidari, A.; Gandomi, A.H.; Chu, X.; Chen, H. RUN Beyond the Metaphor: An Efficient Optimization Algorithm Based on Runge Kutta Method. *Expert Syst. Appl.* **2021**, *181*, 115079. [CrossRef]
22. Zhang, H.; Li, R.; Cai, Z.; Gu, Z.; Heidari, A.A.; Wang, M.; Chen, H.; Chen, M. Advanced orthogonal moth flame optimization with Broyden–Fletcher–Goldfarb–Shanno algorithm: Framework and real-world problems. *Expert Syst. Appl.* **2020**, *159*, 113617. [CrossRef]
23. Heidari, A.A.; Mirjalili, S.; Faris, H.; Aljarah, I.; Mafarja, M.; Chen, H. Harris hawks optimization: Algorithm and applications. *Futur. Gener. Comput. Syst.* **2019**, *97*, 849–872. [CrossRef]
24. Yang, Y.; Chen, H.; Heidari, A.A.; Gandomi, A.H. Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts. *Expert Syst. Appl.* **2021**, *177*, 114864. [CrossRef]
25. Ahmadianfar, I.; Asghar Heidari, A.; Noshadian, S.; Chen, H.; Gandomi, A.H. INFO: An Efficient Optimization Algorithm based on Weighted Mean of Vectors. *Expert Syst. Appl.* **2022**, *195*, 116516. [CrossRef]
26. Tu, J.; Chen, H.; Wang, M.; Gandomi, A.H. The Colony Predation Algorithm. *J. Bionic Eng.* **2021**, *18*, 674–710. [CrossRef]
27. Hu, J.; Gui, W.; Heidari, A.A.; Cai, Z.; Liang, G.; Chen, H.; Pan, Z. Dispersed foraging slime mould algorithm: Continuous and binary variants for global optimization and wrapper-based feature selection. *Knowl. Based Syst.* **2022**, *237*, 107761. [CrossRef]

28. Liu, Y.; Heidari, A.A.; Cai, Z.; Liang, G.; Chen, H.; Pan, Z.; Alsufyani, A.; Bourouis, S. Simulated annealing-based dynamic step shuffled frog leaping algorithm: Optimal performance design and feature selection. *Neurocomputing* **2022**, *503*, 325–362. [CrossRef]
29. Hussien, A.G.; Heidari, A.A.; Ye, X.; Liang, G.; Chen, H.; Pan, Z. Boosting whale optimization with evolution strategy and Gaussian random walks: An image segmentation method. *Eng. Comput.* **2022**, 1–45. [CrossRef]
30. Yu, H.; Song, J.; Chen, C.; Heidari, A.A.; Liu, J.; Chen, H.; Zaguia, A.; Mafarja, M. Image segmentation of Leaf Spot Diseases on Maize using multi-stage Cauchy-Enabled grey wolf algorithm. *Eng. Appl. Artif. Intell.* **2022**, *109*, 104653. [CrossRef]
31. Xu, Y.; Chen, H.; Heidari, A.A.; Luo, J.; Zhang, Q.; Zhao, X.; Li, C. An efficient chaotic mutative moth-flame-inspired optimizer for global optimization tasks. *Expert Syst. Appl.* **2019**, *129*, 135–155. [CrossRef]
32. Zhang, Y.; Liu, R.; Heidari, A.A.; Wang, X.; Chen, Y.; Wang, M.; Chen, H. Towards augmented kernel extreme learning models for bankruptcy prediction: Algorithmic behavior and comprehensive analysis. *Neurocomputing* **2021**, *430*, 185–212. [CrossRef]
33. Yu, H.; Cheng, X.; Chen, C.; Heidari, A.A.; Liu, J.; Cai, Z.; Chen, H. Apple leaf disease recognition method with improved residual network. *Multimed. Tools Appl.* **2022**, *81*, 7759–7782. [CrossRef]
34. Wang, M.; Chen, H.; Yang, B.; Zhao, X.; Hu, L.; Cai, Z.; Huang, H.; Tong, C. Toward an optimal kernel extreme learning machine using a chaotic moth-flame optimization strategy with applications in medical diagnoses. *Neurocomputing* **2017**, *267*, 69–84. [CrossRef]
35. Chen, H.L.; Wang, G.; Ma, C.; Cai, Z.N.; Liu, W.B.; Wang, S.J. An efficient hybrid kernel extreme learning machine approach for early diagnosis of Parkinson's disease. *Neurocomputing* **2016**, *184*, 131–144. [CrossRef]
36. Dong, R.; Chen, H.; Heidari, A.A.; Turabieh, H.; Mafarja, M.; Wang, S. Boosted kernel search: Framework, analysis and case studies on the economic emission dispatch problem. *Knowl. Based Syst.* **2021**, *233*, 107529. [CrossRef]
37. He, Z.; Yen, G.G.; Ding, J. Knee-Based Decision Making and Visualization in Many-Objective Optimization. *IEEE Trans. Evol. Comput.* **2020**, *25*, 292–306. [CrossRef]
38. He, Z.; Yen, G.G.; Lv, J. Evolutionary Multiobjective Optimization with Robustness Enhancement. *IEEE Trans. Evol. Comput.* **2019**, *24*, 494–507. [CrossRef]
39. Wu, S.-H.; Zhan, Z.-H.; Zhang, J. SAFE: Scale-Adaptive Fitness Evaluation Method for Expensive Optimization Problems. *IEEE Trans. Evol. Comput.* **2021**, *25*, 478–491. [CrossRef]
40. Li, J.Y.; Zhan, Z.H.; Wang, C.; Jin, H.; Zhang, J. Boosting data-driven evolutionary algorithm with localized data generation. *IEEE Trans. Evol. Comput.* **2020**, *24*, 923–937. [CrossRef]
41. Deng, W.; Zhang, X.; Zhou, Y.; Liu, Y.; Zhou, X.; Chen, H.; Zhao, H. An enhanced fast non-dominated solution sorting genetic algorithm for multi-objective problems. *Inf. Sci.* **2022**, *585*, 441–453. [CrossRef]
42. Hua, Y.; Liu, Q.; Hao, K.; Jin, Y. A Survey of Evolutionary Algorithms for Multi-Objective Optimization Problems with Irregular Pareto Fronts. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 303–318. [CrossRef]
43. Han, X.; Han, Y.; Chen, Q.; Li, J.; Sang, H.; Liu, Y.; Pan, Q.; Nojima, Y. Distributed Flow Shop Scheduling with Se-quence-Dependent Setup Times Using an Improved Iterated Greedy Algorithm. *Complex Syst. Model. Simul.* **2021**, *1*, 198–217. [CrossRef]
44. Gao, D.; Wang, G.-G.; Pedrycz, W. Solving Fuzzy Job-Shop Scheduling Problem Using DE Algorithm Improved by a Selection Mechanism. *IEEE Trans. Fuzzy Syst.* **2020**, *28*, 3265–3275. [CrossRef]
45. Wang, G.-G.; Gao, D.; Pedrycz, W. Solving Multiobjective Fuzzy Job-Shop Scheduling Problem by a Hybrid Adaptive Differential Evolution Algorithm. *IEEE Trans. Ind. Informatics* **2022**, *18*, 8519–8528. [CrossRef]
46. Chen, H.L.; Yang, B.; Wang, S.J.; Wang, G.; Liu, D.Y.; Li, H.Z.; Liu, W. Towards an optimal support vector machine classifier using a parallel particle swarm optimization strategy. *Appl. Math. Comput.* **2014**, *239*, 180–197. [CrossRef]
47. Deng, W.; Xu, J.; Zhao, H.; Song, Y. A Novel Gate Resource Allocation Method Using Improved PSO-Based QEA. *IEEE Trans. Intell. Transp. Syst.* **2020**, *23*, 1737–1745. [CrossRef]
48. Deng, W.; Xu, J.; Song, Y.; Zhao, H. An effective improved co-evolution ant colony optimisation algorithm with multi-strategies and its application. *Int. J. Bio Inspir. Comput.* **2020**, *16*, 158–170. [CrossRef]
49. Ye, X.; Liu, W.; Li, H.; Wang, M.; Chi, C.; Liang, G.; Chen, H.; Huang, H. Modified Whale Optimization Algorithm for Solar Cell and PV Module Parameter Identification. *Complexity* **2021**, *2021*, 8878686. [CrossRef]
50. Yu, H.; Yuan, K.; Li, W.; Zhao, N.; Chen, W.; Huang, C.; Chen, H.; Wang, M. Improved Butterfly Optimizer-Configured Extreme Learning Machine for Fault Diagnosis. *Complexity* **2021**, *2021*, 6315010. [CrossRef]
51. Agrawal, R.; Kaur, B.; Sharma, S. Quantum based Whale Optimization Algorithm for wrapper feature selection. *Appl. Soft Comput.* **2020**, *89*, 106092. [CrossRef]
52. Bai, L.; Han, Z.; Ren, J.; Qin, X. Research on feature selection for rotating machinery based on Supervision Kernel Entropy Component Analysis with Whale Optimization Algorithm. *Appl. Soft Comput.* **2020**, *92*, 106245. [CrossRef]
53. Bahiraei, M.; Foong, L.K.; Hosseini, S.; Mazaheri, N. Predicting heat transfer rate of a ribbed triple-tube heat exchanger working with nanofluid using neural network enhanced by advanced optimization algorithms. *Powder Technol.* **2021**, *381*, 459–476. [CrossRef]
54. Qi, A.; Zhao, D.; Yu, F.; Heidari, A.A.; Chen, H.; Xiao, L. Directional mutation and crossover for immature performance of whale algorithm with application to engineering optimization. *J. Comput. Des. Eng.* **2022**, *9*, 519–563. [CrossRef]
55. Bui, D.T.; Abdullahi, M.M.; Ghareh, S.; Moayedi, H.; Nguyen, H. Fine-tuning of neural computing using whale optimization algorithm for predicting compressive strength of concrete. *Eng. Comput.* **2021**, *37*, 701–712. [CrossRef]

56. Butti, D.; Mangipudi, S.K.; Rayapudi, S. Model Order Reduction Based Power System Stabilizer Design Using Improved Whale Optimization Algorithm. *IETE J. Res.* **2021**, 1–20. [CrossRef]
57. Cao, Y.; Li, Y.; Zhang, G.; Jermsittiparsert, K.; Nasseri, M. An efficient terminal voltage control for PEMFC based on an improved version of whale optimization algorithm. *Energy Rep.* **2020**, *6*, 530–542. [CrossRef]
58. Çerçevik, A.E.; Avşar, Ö.; Hasançebi, O. Optimum design of seismic isolation systems using metaheuristic search methods. *Soil Dyn. Earthq. Eng.* **2019**, *131*, 106012. [CrossRef]
59. Zhao, S.; Song, J.; Du, X.; Liu, T.; Chen, H.; Chen, H. Intervention-Aware Epidemic Prediction by Enhanced Whale Optimization. In *International Conference on Knowledge Science, Engineering and Management*; Memmi, G., Yang, B., Kong, L., Zhang, T., Qiu, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2022; pp. 457–468.
60. Fan, Y.; Wang, P.; Heidari, A.A.; Wang, M.; Zhao, X.; Chen, H.; Li, C. Boosted hunting-based fruit fly optimization and advances in real-world problems. *Expert Syst. Appl.* **2020**, *159*, 113502. [CrossRef]
61. Raj, S.; Bhattacharyya, B. Optimal placement of TCSC and SVC for reactive power planning using Whale optimization algorithm. *Swarm Evol. Comput.* **2018**, *40*, 131–143. [CrossRef]
62. Guo, Y.; Shen, H.; Chen, L.; Liu, Y.; Kang, Z. Improved whale optimization algorithm based on random hopping update and random control parameter. *J. Intell. Fuzzy Syst.* **2021**, *40*, 363–379. [CrossRef]
63. Jiang, R.; Yang, M.; Wang, S.; Chao, T. An improved whale optimization algorithm with armed force program and strategic adjustment. *Appl. Math. Model.* **2020**, *81*, 603–623. [CrossRef]
64. Tu, J.; Chen, H.; Liu, J.; Heidari, A.A.; Zhang, X.; Wang, M.; Ruby, R.; Pham, Q.V. Evolutionary biogeography-based whale optimization methods with communication structure: Towards measuring the balance. *Knowl. Based Syst.* **2021**, *212*, 106642. [CrossRef]
65. Wang, G.; Gui, W.; Liang, G.; Zhao, X.; Wang, M.; Mafarja, M.; Turabieh, H.; Xin, J.; Chen, H.; Ma, X. Spiral Motion Enhanced Elite Whale Optimizer for Global Tasks. *Complexity* **2021**, *2021*, 1–33. [CrossRef]
66. Abd Elazim, S.M.; Ali, E.S. Optimal network restructure via improved whale optimization approach. *Int. J. Commun. Syst.* **2021**, *34*, e4617. [CrossRef]
67. Abdel-Basset, M.; Chang, V.; Mohamed, R. HSMA_WOA: A hybrid novel Slime mould algorithm with whale optimization algorithm for tackling the image segmentation problem of chest X-ray images. *Appl. Soft Comput.* **2020**, *95*, 106642. [CrossRef]
68. Chai, Q.-W.; Chu, S.-C.; Pan, J.-S.; Hu, P.; Zheng, W.-M. A parallel WOA with two communication strategies applied in DV-Hop localization method. *EURASIP J. Wirel. Commun. Netw.* **2020**, *2020*, 1–10. [CrossRef]
69. Heidari, A.A.; Aljarah, I.; Faris, H.; Chen, H.; Luo, J.; Mirjalili, S. An enhanced associative learning-based exploratory whale optimizer for global optimization. *Neural Comput. Appl.* **2020**, *32*, 5185–5211. [CrossRef]
70. Jin, Q.; Xu, Z.; Cai, W. An Improved Whale Optimization Algorithm with Random Evolution and Special Reinforcement Dual-Operation Strategy Collaboration. *Symmetry* **2021**, *13*, 238. [CrossRef]
71. Qin, A.K.; Huang, V.L.; Suganthan, P.N. Differential Evolution Algorithm with Strategy Adaptation for Global Numerical Optimization. *IEEE Trans. Evol. Comput.* **2008**, *13*, 398–417. [CrossRef]
72. Wan, X.; Zuo, X.; Zhao, X. A differential evolution algorithm combined with linear programming for solving a closed loop facility layout problem. *Appl. Soft Comput.* **2022**, 108725. [CrossRef]
73. Yuan, Y.; Cao, J.; Wang, X.; Zhang, Z.; Liu, Y. Economic-effectiveness analysis of micro-fins helically coiled tube heat exchanger and optimization based on multi-objective differential evolution algorithm. *Appl. Therm. Eng.* **2022**, *201*, 117764. [CrossRef]
74. Liu, D.; Hu, Z.; Su, Q.; Liu, M. A niching differential evolution algorithm for the large-scale combined heat and power economic dispatch problem. *Appl. Soft Comput.* **2021**, *113*, 108017. [CrossRef]
75. He, Z.; Ning, D.; Gou, Y.; Zhou, Z. Wave energy converter optimization based on differential evolution algorithm. *Energy* **2022**, *246*, 123433. [CrossRef]
76. Meng, A.-B.; Chen, Y.-C.; Yin, H.; Chen, S.-Z. Crisscross optimization algorithm and its application. *Knowl. Based Syst.* **2014**, *67*, 218–229. [CrossRef]
77. Luo, J.; Chen, H.; Heidari, A.A.; Xu, Y.; Zhang, Q.; Li, C. Multi-strategy boosted mutative whale-inspired optimization approaches. *Appl. Math. Model.* **2019**, *73*, 109–123. [CrossRef]
78. Yousri, D.; Allam, D.; Eteiba, M. Chaotic whale optimizer variants for parameters estimation of the chaotic behavior in Permanent Magnet Synchronous Motor. *Appl. Soft Comput.* **2019**, *74*, 479–503. [CrossRef]
79. Ling, Y.; Zhou, Y.; Luo, Q. Lévy Flight Trajectory-Based Whale Optimization Algorithm for Global Optimization. *IEEE Access* **2017**, *5*, 6168–6186. [CrossRef]
80. Tubishat, M.; Abushariah, M.A.M.; Idris, N.; Aljarah, I. Improved whale optimization algorithm for feature selection in Arabic sentiment analysis. *Appl. Intell.* **2018**, *49*, 1688–1707. [CrossRef]
81. Cai, Z.; Gu, J.; Luo, J.; Zhang, Q.; Chen, H.; Pan, Z.; Li, Y.; Li, C. Evolving an optimal kernel extreme learning machine by using an enhanced grey wolf optimization strategy. *Expert Syst. Appl.* **2019**, *138*, 112814. [CrossRef]
82. Heidari, A.A.; Ali Abbaspour, R.; Chen, H. Efficient boosted grey wolf optimizers for global search and kernel extreme learning machine training. *Appl. Soft Comput.* **2019**, *81*, 105521. [CrossRef]
83. Sun, T.-Y.; Liu, C.-C.; Tsai, S.-J.; Hsieh, S.-T.; Li, K.-Y. Cluster Guide Particle Swarm Optimization (CGPSO) for Underdetermined Blind Source Separation with Advanced Conditions. *IEEE Trans. Evol. Comput.* **2010**, *15*, 798–811. [CrossRef]

84. Nenavath, H.; Jatoth, R.K. Hybridizing sine cosine algorithm with differential evolution for global optimization and object tracking. *Appl. Soft Comput.* **2018**, *62*, 1019–1043. [CrossRef]
85. Singh, R.P.; Mukherjee, V.; Ghoshal, S.P. Optimal power flow by particle swarm optimization with an aging leader and challengers. *Int. J. Eng.* **2015**, *7*, 123–132. [CrossRef]
86. Liang, H.; Liu, Y.; Shen, Y.; Li, F.; Man, Y. A Hybrid Bat Algorithm for Economic Dispatch with Random Wind Power. *IEEE Trans. Power Syst.* **2018**, *33*, 5052–5061. [CrossRef]
87. Adarsh, B.R.; Raghunathan, T.; Jayabarathi, T.; Yang, X.-S. Economic dispatch using chaotic bat algorithm. *Energy* **2016**, *96*, 666–675. [CrossRef]
88. Abd Elaziz, M.; Oliva, D.; Xiong, S. An improved Opposition-Based Sine Cosine Algorithm for global optimization. *Expert Syst. Appl.* **2017**, *90*, 484–500. [CrossRef]
89. Wu, Z.; Li, R.; Xie, J.; Zhou, Z.; Guo, J.; Jiang, J.; Su, X. A user sensitive subject protection approach for book search service. *J. Assoc. Inf. Sci. Technol.* **2020**, *71*, 183–195. [CrossRef]
90. Wu, Z.; Shen, S.; Zhou, H.; Li, H.; Lu, C.; Zou, D. An effective approach for the protection of user commodity viewing privacy in e-commerce website. *Knowl. Based Syst.* **2021**, *220*, 106952. [CrossRef]
91. Yang, Z.; Chen, H.; Zhang, J.; Chang, Y. Context-aware Attentive Multilevel Feature Fusion for Named Entity Recognition. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**. [CrossRef]
92. Huang, L.; Yang, Y.; Chen, H.; Zhang, Y.; Wang, Z.; He, L. Context-aware road travel time estimation by coupled tensor decomposition based on trajectory data. *Knowl. Based Syst.* **2022**, *245*. [CrossRef]
93. Hu, K.; Zhao, L.; Feng, S.; Zhang, S.; Zhou, Q.; Gao, X.; Guo, Y. Colorectal polyp region extraction using saliency detection network with neutrosophic enhancement. *Comput. Biol. Med.* **2022**, *147*, 105760. [CrossRef] [PubMed]
94. Zhang, X.; Zheng, J.; Wang, D.; Zhao, L. Exemplar-Based Denoising: A Unified Low-Rank Recovery Framework. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *30*, 2538–2549. [CrossRef]
95. Qi, A.; Zhao, D.; Yu, F.; Heidari, A.A.; Wu, Z.; Cai, Z.; Alenezi, F.; Mansour, R.F.; Chen, H.; Chen, M. Directional mutation and crossover boosted ant colony optimization with application to COVID-19 X-ray image segmentation. *Comput. Biol. Med.* **2022**, *148*, 105810. [CrossRef]
96. Ren, L.; Zhao, D.; Zhao, X.; Chen, W.; Li, L.; Wu, T.; Liang, G.; Cai, Z.; Xu, S. Multi-level thresholding segmentation for pathological images: Optimal performance design of a new modified differential evolution. *Comput. Bio. Med.* **2022**, *148*, 105910. [CrossRef]
97. Su, H.; Zhao, D.; Elmannai, H.; Heidari, A.A.; Bourouis, S.; Wu, Z.; Cai, Z.; Gui, W.; Chen, M. Multilevel threshold image segmentation for COVID-19 chest radiography: A framework using horizontal and vertical multiverse optimization. *Comput. Biol. Med.* **2022**, *146*, 105618. [CrossRef]
98. Cao, X.; Wang, J.; Zeng, B. A Study on the Strong Duality of Second-Order Conic Relaxation of AC Optimal Power Flow in Radial Networks. *IEEE Trans. Power Syst.* **2021**, *37*, 443–455. [CrossRef]