International Journal of Computational Intelligence Systems (2023) 16:102
https://doi.org/10.1007/s44196-023-00279-6

RESEARCH ARTICLE

Artificial Ecosystem-Based Optimization with Dwarf Mongoose Optimization for Feature Selection and Global Optimization Problems

Ibrahim Al-Shourbaji[1,2] · Pramod Kachare[3] · Sajid Fadlelseed[2] · Abdoh Jabbari[1] · Abdelazim G. Hussien[4,5] · Faisal Al-Saqqar[6] · Laith Abualigah[6,7,8,9,10,11] · Abdalla Alameen[12]

Received: 28 March 2023 / Accepted: 29 May 2023

Abstract

Meta-Heuristic (MH) algorithms have recently proven successful in a broad range of applications because of their strong capabilities in picking the optimal features and removing redundant and irrelevant features. Artificial Ecosystem-based Optimization (AEO) shows extraordinary ability in the exploration stage and poor exploitation because of its stochastic nature. Dwarf Mongoose Optimization Algorithm (DMOA) is a recent MH algorithm showing a high exploitation capability. This paper proposes AEO-DMOA Feature Selection (FS) by integrating AEO and DMOA to develop an efficient FS algorithm with a better equilibrium between exploration and exploitation. The performance of the AEO-DMOA is investigated on seven datasets from different domains and a collection of twenty-eight global optimization functions, eighteen CEC2017, and ten CEC2019 benchmark functions. Comparative study and statistical analysis demonstrate that AEO-DMOA gives competitive results and is statistically significant compared to other popular MH approaches. The benchmark function results also indicate enhanced performance in high-dimensional search space.

Keywords Feature selection · Machine learning · Metaheuristic algorithms · Artificial ecosystem-based optimization · Dwarf mongoose optimization

Abbreviations

AEO	Artificial ecosystem-based optimization
ACO	Ant colony optimization
ALO	Ant lion optimization
BS	Bird swarms
CF	Consumption factor
DMOA	Dwarf mongoose optimization algorithm
FS	Feature selection
FV	Fitness value
GA	Genetic algorithm
GTO	Gorilla troops optimizer
GWO	Gray wolf optimizer
HHO	Harris hawks optimization
KNN	K-nearest neighbor

✉ Abdelazim G. Hussien
abdelazim.hussien@liu.se

1 Department of Computer and Network Engineering, Jazan University, Jazan 45142, Saudi Arabia

2 Department of Computer Science, University of Hertfordshire, Hatfield AL10 9AB, UK

3 Department of Electronics & Telecomm, Ramrao Adik Institute of Technology, EngineeringNerul, Navi Mumbai, Maharashtra 400706, India

4 Department of Computer and Information Science, Linköping University, Linköping, Sweden

5 Faculty of Science, Fayoum University, Faiyum, Egypt

6 Computer Science Department, Prince Hussein Bin Abdullah Faculty for Information Technology, Al al-Bayt University, Mafraq 25113, Jordan

7 School of Engineering and Technology, Sunway University Malaysia, 27500 Petaling Jaya, Malaysia

8 Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman 19328, Jordan

9 Faculty of Information Technology, Middle East University, Amman 11831, Jordan

10 Applied Science Research Center, Applied Science Private University, Amman 11931, Jordan

11 School of Computer Sciences, Universiti Sains Malaysia, 11800 Pulau Pinang, Malaysia

12 Department of Computer Science, College of Arts and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia

MH	Meta-heuristic
MVO	Multi-verse optimizer
OFS	Optimal feature subset
PSO	Particle swarm optimization
RSA	Reptile search algorithm
SA	Simulated annealing
SCA	Sine cosine algorithm
SD	Standard deviation
SO	Snake optimizer
SSA	Salp swarm algorithm
WOA	Whale optimization algorithm

1 Introduction

Due to the exponential development in the amount of data being processed and stored by information systems, it is harder to retrieve relevant information. However, these stored data may include many attributes that may not be important or irrelevant. FS methods aim to pick an Optimal Feature Subset (OFS), which reduces overfitting and the computational time of machine learning models by eliminating redundant features while maintaining high classification performance [1–3]. Finding OFS in a broad search space is considered a multi-objective issue, i.e., minimizing the number of selected features and maximizing accuracy [4–7].

Nature-inspired methods are inspired by natural phenomena and mostly MH optimization for FS problems. These methods' inspiration sources are broken down into three types [8]: swarm-based algorithms, evolutionary-based algorithms, and physics-based algorithms. These methods all share two principles, exploration and exploitation [9]. In the first principle, the algorithm tries to find new regions in the search area. In the later phase, the algorithm looks around the obtained solution from the first phase to discover the best candidate.

Some of the frequently used MH techniques for FS include Particle Swarm Optimization (PSO) [10], Multi-Verse Optimizer (MVO) [11], Whale Optimization Algorithm (WOA) [12], Salp Swarm Algorithm (SSA) [13], Genetic Algorithm (GA) [14], Gray Wolf Optimizer (GWO) [15], AEO [16], DMOA [17], Snake Optimizer [18], Fick's Law Algorithm (FLA) [19], and Jellyfish Search [20]. Moreover, MH algorithms can be combined to achieve better results for FS problems. Such examples include Simulated Annealing (SA) added to Harris Hawks Optimization (HHO) [21], Ant Lion Optimization (ALO) added to Sine Cosine Algorithm (SCA) [22], Bird Swarms (BS) added to Gorilla Troops Optimizer (GTO) [23], Reptile Search Algorithm (RSA) added to Ant Colony Optimization (ACO) [24], RSA added to Snake Optimizer (SO) [25], Evolutionary-Mean shift algorithm for dynamic multimodal function optimization [26], Group-based synchronous-asynchronous grey wolf optimizer [27] and many other [28–30]. With the advent and success of MH algorithms in solving the problem of FS, new and improved approaches are still needed to deal with this problem. MH methods solve multi-objective optimization problems by increasing classification accuracy with the smallest possible number of OFS. Several works explored and investigated MH methods to effectively search a given space to obtain the best global solutions [31–35].

The AEO method has great potential to choose OFS in several applications, such as triple diode photovoltaic [36], image segmentation [37], economic dispatch [38], and Agriculture Feeders [39]. However, it shows strong ability in the exploration stage and poor exploitation because of its stochastic nature [40]. On the other hand, DMOA is a recent MH method that shows a high capability in exploitation [17]. Therefore, an efficient FS-based approach, namely, AEO-DMOA, is presented, which merges the best strength of the AEO in exploration and DMOA in the exploitation phases toward targeting optimum solutions. The AEO is applied on half of the defined number of iterations to discover a better solution in the search space, while DMOA identifies the best candidate around the obtained optimal space in the remaining number of iterations. The contributions of this work are summarized as follows:

- AEO-DMOA, a hybrid approach, is developed to provide better performance with a better equilibrium between search space's exploration and exploitation for the problem of FS.
- AEO-DMOA is evaluated on seven UCI datasets, twenty-eight test functions, eighteen CEC2017, and ten CEC2019 test functions
- Developed AEO-DMOA applicability is compared with other competitive MH approaches.

The rest of this article is structured as follows: Sect. 2 briefly reviews AEO and DMOA, followed by a description of the interdicted AEO-DMOA in Sect. 3. Section 4 presents the experimental analysis and statistical comparison of the AEO-DMOA with other well-known MH methods on the tested datasets and studied test functions. Finally, Sect. 5 concludes this article.

2 Methods

2.1 Artificial Ecosystem-Based Optimization (AEO)

AEO is an MH method motivated by the natural ecosystem's energy flow [16]. AEO uses three operators to achieve optimal solutions, as described below.

2.1.1 Production

In this operator, the producer represents the worst individual in the population. Thus, it must be updated concerning the best individual by considering the upper and lower boundaries of the given search space so that it can guide other individuals to search other regions. The operator generates a new individual between the best individual x_{best} (based on fitness) and the randomly produced position of individuals in the search space x_{rand} by replacing the previous one. This operator can be given as,

$$x_i(t+1) = (1-\alpha)x_{best}(t) + \alpha x_{rand}(t) \tag{1}$$

$$\alpha = \left(1 - \frac{t}{T}\right) r_1 \tag{2}$$

$$x_{rand} = r_2(UB - LB) + LB \tag{3}$$

where $x_{rand}(t)$ guides the other individuals to explore search space in the subsequent iterations broadly, $x_i(t+1)$ leads the remaining individuals to exploit around intensively $x_{best}(t)$, α is a linear weight coefficient that leads the randomly positioned individual to the best individual position $x_{best}(t)$ through the pre-defined maximum number of iterations T. Two random numbers r_1 and r_2 are sampled from a uniform distribution over [0, 1]. For a given search space, UB and LB represent the upper and the lower extreme values.

2.1.2 Consumption

This operator starts after the production operator and obtains food energy by eating a producer, a random low-energy consumer, or both. A Levy flight-like random walk, called Consumption Factor (CF), is employed to enhance exploration capability, and it is defined as follows:

$$CF = \frac{1}{2}\frac{v_1}{|v_2|}, v_1, v_2 \in N(0,1) \tag{4}$$

where, $N(0,1)$ is a zero mean and unity deviation normal distribution.

Different types of consumers adopt different consumption behaviors to update their positions. These strategies include:

1. Herbivore behavior: A herbivore consumer would eat only the producer and can be formulated as:

$$x_i(t+1) = x_i(t) + CF.\left(x_i(t) - x_1(t)\right), i \in [2, \ldots P] \tag{5}$$

2. Carnivore behavior: A carnivore consumer would only eat another consumer with energy higher than itself. Mathematically, it can be modeled as follows:

$$x_i(t+1) = x_i(t) + CF.\left(x_i(t) - x_{rand \in (0,2i-1)}(t)\right), i \in [3, \ldots P] \tag{6}$$

3. Omnivore behavior: An omnivore consumer can eat a random producer or a random producer with more energy than itself. This behavior can be presented as:

$$\begin{aligned} x_i(t+1) = {} & x_i(t) + CF\left(r_2(x_i(t) - x_1(t))\right) \\ & + (1-r_2)(x_i(t) - x_{rand \in (0,2i-1)}(t)), i \in [3, \ldots P] \end{aligned} \tag{7}$$

2.1.3 Decomposition

In this final phase, the ecosystem agent dissolves. The decomposer breaks down the remains of dead individuals to provide the required growth nutrients for producers. The decomposition operator can be expressed as:

$$\begin{aligned} x_i(t+1) = {} & x_P(t) + De(e.x_P(t) - h.x_{rand \in (0,2i-1)}(t)), \\ & i \in [1, \ldots P] \end{aligned} \tag{8}$$

Where $De = 3uu \in N(0,1)$, $e = r_3.randi([1,2]) - 1$, and $h = 2r_3 - 1$where e, h, and De, are weight coefficients designed to model decomposition behavior.

2.2 Dwarf Mongoose Optimization Algorithm (DMOA)

DMOA is another MH method introduced to simulate the dwarf mongoose's prey size limitation, social organization, semi-nomadic life, and others [17]. The DMOA begins with initializing a set of random candidate populations of mongooses between the maxima (UB) and minima (LB) of the given problem. The optimization process is represented in the following phases:

2.2.1 Alpha Group

When initializing the population, each population fitness probability can be computed by:

$$\alpha = \frac{fit_i}{\sum_{i=1}^{n} fit_i} \tag{9}$$

For bs number of babysitters, the family is maintained inside a path $peep$ marked by $\alpha - bs$ number of alpha female's vocalization. The sleeping mound is initialized to α when every mongoose sleeps. In order to generate a candidate food position, DMOA employs the following:

$$X_{i+1} = x_i + phi + peep \tag{10}$$

where, phi is sampled randomly from a uniform distribution over [0, 1].

After every iteration, the sleeping mound sm can be given as:

$$sm_i = \frac{fit_{i+1} - fit_i}{max\lfloor fit_{i+1}, fit_i \rfloor} \tag{11}$$

While the Average (Avg) value of the sm can be represented as:

$$Avgsm = \frac{\sum_{i=1}^{n} sm_i}{n} \tag{12}$$

The DMOA is then moved to the next phase, named the scouting phase. In this phase, the sleeping mound or the next food source is assessed after satisfying the babysitter exchange condition.

2.2.2 Scout Group

This phase looks at the next sleeping mound, where exploration is guaranteed as the mongooses do not go back to the previous sleeping mound. The overall performance of the Mongooses decides this movement. The motivation is that sufficiently large foraging will discover a new sm. The scout mongoose can be presented as:

$$X_{i+1} = \begin{cases} X_i - CF * phi * rand[X_i - \overrightarrow{M}] \; if Avgsm_{i+1} > Avgsm_i \\ X_i + CF * phi * rand[X_i - \overrightarrow{M}] \; otherwise \end{cases} \tag{13}$$

where, $CF = \left(1 - \frac{iter}{max_{iter}}\right)^{2\frac{iter}{max_{iter}}}$ and $\overrightarrow{M} = \sum_{i=1}^{n} \frac{X_i * sm_i}{X_i}$ where, X_i is a vector determining the movement of the mongoose to the new sm, CF is a parameter controlling the group collective-volitive movement. CF is decreased linearly with the number of iterations. The $rand$ is sampled randomly from a uniform distribution over [0, 1] and $\overrightarrow{M}$ is a vector specifying the mongoose's movement to the new sm.

2.2.3 The Babysitters

The young mongooses are cared for by supporting members of the group, the babysitters. A regular rotation of babysitters allows the daily foraging of the rest of the group by the alpha female (mother). At midday and in the evening, the alpha female returns to suckle the young. The population size decides the babysitter count. The percentage of babysitter representatives simulates this group by shrinking the population, affecting the DMOA. The previously held food source and scouting information of the replacing members of the family is reset by the exchange parameter of the babysitter. In the next iteration, the average weight of the alpha group is reduced by setting the fitness weight of the babysitters as zero. It hinders the group movement and emphasizes exploitation.

3 Proposed Method

This section presents the developed AEO-DMOA for FS. The primary objective of the developed AEO-DMOA is to split the number of iterations into two halves and apply AEO in the first half to explore the entire search area for optimizing the search area boundaries, while DMOA in the second half to exploit the AEO-optimized search area for obtaining the best solution. This sequential implementation helps both methods by alleviating the chances of being trapped in local optima and maintaining an appropriate balance between exploration and exploitation during optimization.

Firstly, the hyper-parameters of AEO and DMOA, such as the maximum number of iterations (T) and the number of candidate solutions (N), are initialized. The upper (UB) and lower (LB) boundaries of each feature dimension of the given search space are calculated. All N candidate solutions are initialized uniformly in the range [− 1, 1] as described earlier in Eq. (1). The Fitness Value (FV) is calculated for each candidate solution using K-Nearest Neighbor (KNN) classifier with the 5-nearest neighbor with a Euclidean distance measure. The candidate solution with the smallest FV is stored as the globally best solution. The FV can be calculated as follows:

$$FV = \lambda \times (1 - AC) + (1 - \lambda) \times \frac{SF_i}{M} \tag{14}$$

where, λ is a weight that controls the importance of classification performance and fraction of selected features, AC is the accuracy of the KNN, SF_i is the number of features selected by the candidate solution, and M is the dimensionality of the original dataset. The value λ ranges from 0 (no importance to classification accuracy) to 1 (no importance to feature selection) and is set as 0.99 in this work, as suggested in the literature. The SF is calculated by thresholding the current position of the candidate solution as follows:

$$SF_i = \begin{cases} 1 \; if x_i > 0.5 \\ 0 \; otherwise \end{cases} \tag{15}$$

It must be noted that the threshold of 0.5 used to select features is empirically selected. The exact value of the threshold used during the training does not affect the feature selection as MH algorithm adapts the positions of important features above the threshold. The extreme values for the threshold make it difficult for the MH algorithm to separate important features from the redundant ones. Hence, as the literature suggests, a threshold value 0.5 is used [24].

The optimization starts with the AEO method. At the first iteration, the Production phase of the AEO updates the worst candidate solution (i.e., one with the largest fitness value) with the best individual (i.e., one with the smallest fitness value) by considering UB and LB. After this phase, the three phases of the AEO algorithm are implemented based on a random number ($rand$) in the range 0–1, as follows:

$$If\ rand \geq \frac{2}{3},\ then\ Herbivore\ phase, \tag{16}$$

$$else\ if\ \frac{1}{3} \leq rand \leq \frac{2}{3}\ then,\ Carnivore\ phase, \tag{17}$$

$$else\ if\ rand \leq \frac{1}{3},\ then\ Omnivore\ phase, \tag{18}$$

The process repeats until all candidate solutions are processed. In the end, the Decomposition Phase Eq. (8) of the AEO algorithm is implemented, breaking down the dead candidate solutions as growth nutrients for producers. Finally, FV is calculated for all candidate solutions at the end of the iteration. The latter is updated if any candidate solution has an FV smaller than the global-best solution. In the next iteration, the entire process is repeated until the number of iterations has reached half the maximum ($T/2$).

The algorithm switches from AEO to DMOA after $T/2$ iterations. The $T/2$ th iteration starts by updating the AEO-optimized candidate solution using the Alpha group as in Eqs. 9, 10 and increments the counter C (which at first execution of DMOA is 0). If counter C is less than the babysitter exchange parameter L, then calculate the average value of the sleeping mound $AvgSum$ as in Eqs. 11, 12. If the average value of the sleeping mound of the previous iteration $AvgSum_{t-1}$ is less than the sleeping mound of the current iteration $AvgSum_t$ Then, the exploration phase of DMOA is implemented to update the candidate solutions, or the exploitation phase of DMOA is implemented. Finally, when maximum iterations are reached for DMOA, the optimum solutions' fitness FV is calculated, and the global-best solution is updated. The process continues until the counter C reaches babysitter exchange parameter L, when the babysitter group is updated, as in Eqs. 10–11, and counter C is initialized to 0. The process continues by updating the global-best solution. Figure 1 provides the process flow of the AEO-DMOA.and its pseudocode is reported in Algorithm 1.

The optimization stops when reaching the defined number of T. The global-best solution available at the end of all iterations is used as the optimum solution. As discussed earlier in Eq. 15, features with positions larger than 0.5 are added to the OFS. A classifier is trained for the OFS as input and desired class as the output. During the testing phase, OFS selects the salient features while others are rejected.

4 Experimental Results

The developed AEO-DMOA is applied to solve the problem of FS on seven UCI datasets, twenty-eight benchmarked test functions, eighteen CEC2017, and ten CEC2019. The results are compared with other MH methods and provided in this section.

4.1 Experimental Setup

The efficiency of the AEO-DMOA is investigated with other MH approaches PSO [10], MVO [11], WOA [12], SSA [13], AEO [16], and DMOA [17]. The parameter

Algorithm 1: Pseudocode of the developed AEO-DMOA.

```
Divide dataset into two subsets for training & testing data without repetition.
Training Phase
Load training data
Initialize hyper-parameters of both methods $t = 1, C = 1, N, and\ T$. Also, calculate upper $UB$ and lower $LB$ bounds for each feature dimension
Initialize $N$ candidate solutions using uniform random number generator, Eq. (1)
Calculate fitness value $FV$ for all candidate solutions using Eq. (14)
Initialize smallest $FV$ solutions as global-best solution
While $t < T$ do
    If $t < T/2$ then
        $i = 1$
        While $i < N$ do
            If $i = 1$ then
                Production phase of AEO, Eq. (1)–(3)
            Else
                Generate random number $rand$
                If $rand \geq \frac{2}{3}$ then
                    Herbivore Phase of AEO, Eq. (5)
                Else if $\frac{1}{3} \leq rand \leq \frac{2}{3}$ then
                    Carnivore Phase of AEO, Eq. (6)
                Else if $rand \leq \frac{1}{3}$ then
                    Omnivore Phase of AEO, Eq. (7)
            $i = i + 1$
    Decomposition Phase of AEO, Eq. (8)
    Else
        Alpha update of DMOA, Eq. (9)
        $C = C + 1$
        If $C > L$ then
            Babysitter update of DMOA, Eq. (10)–(11)
        Else
            If $AvgSum_t > AvgSum_{t-1}$ then
                Exploration Phase of DMOA, Eq. (13)
            Else
                Exploitation Phase of DMOA, Eq. (13)
    Calculate $FV$ of updated candidates and update global-best solution, Eq. (14)
Use threshold to calculate OFS from global-best solution, Eq. (15)
Testing Phase
Load testing data
Select significant features defined by OFS
Evaluate the classifier performance
```

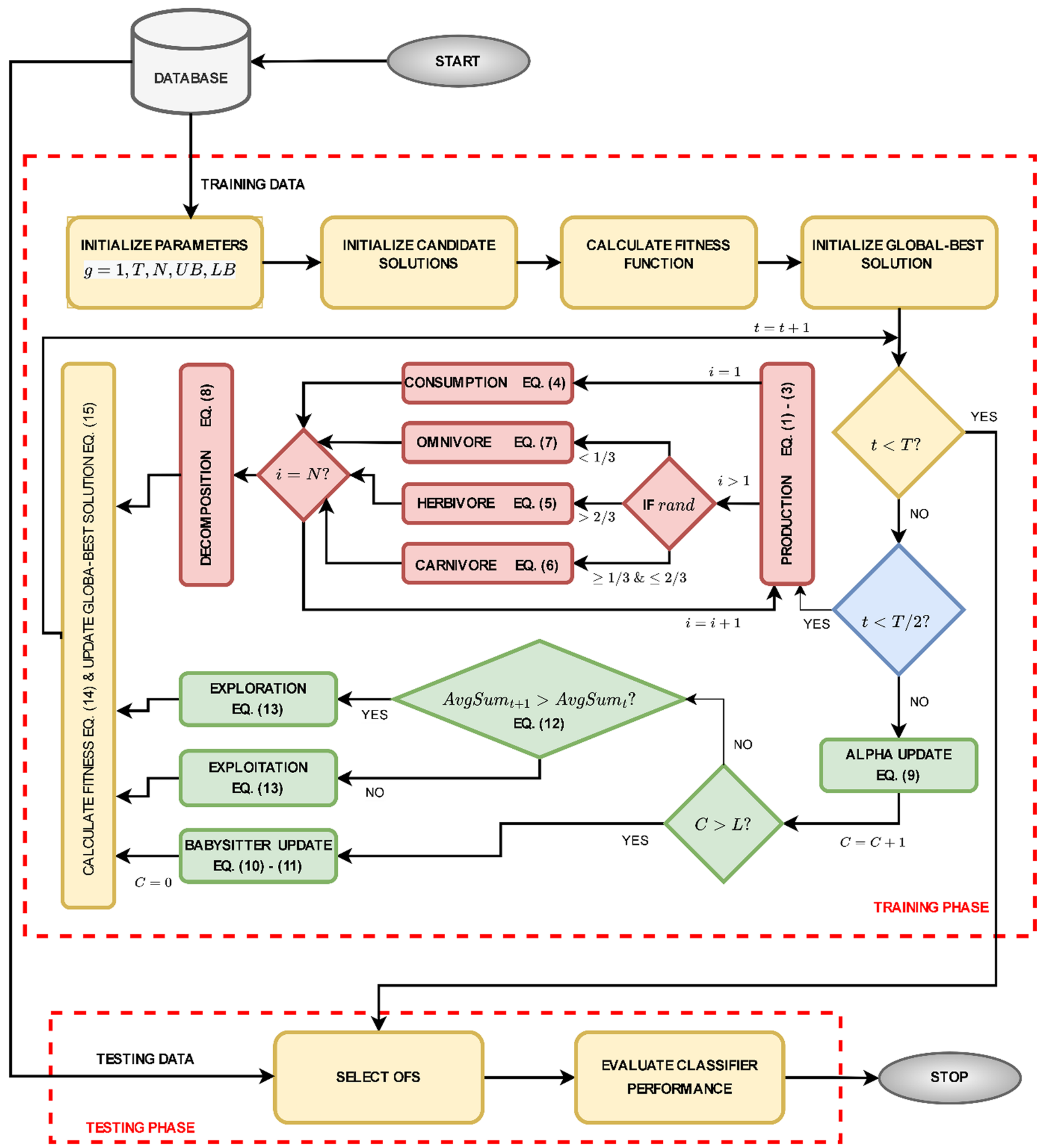

Fig. 1 Flowchart of the developed AEO-DMOA approach

settings for these methods are defined as how they are implemented in the original work, which is shown in Table 1. The standard parameters in this study are selected empirically and are set: Population size = 20, $T = 100$, and each method is indecently run 20 times. The experiments are performed on a 3.13 GHz Windows 10 machine with 32 GB RAM and implemented using Python Scikit-learn.

4.2 Datasets Descriptions

The efficiency of the AEO-DMOA is validated using seven datasets, and their characteristics are given in Table 2. Six of seven datasets are designed for binary classification problems, while the seventh is designed as a multiclass classification problem.

Table 1 Parameter settings for different MH algorithms

Algorithm	Parameters
PSO	$c_1 = c_2 = 2$, $w_{min} = 0.1$ and $w_{max} = 0.9$
MVO	$WEP_{max} = 1$, $WEP_{min} = 0.2$, $p = 6$, and α decreasing in the interval [-2, 0]
WOA	α reduces from 2 to 0 and α_2 reduces from − 1 to − 2
SSA	c_2 and c_2 are sampled randomly from a uniform distribution over [1,0]
AEO	r_1, r_2 and r_3 are uniformly distributed random numbers with values from 0–1
DMOA	*phi* is a uniformly distributed random number in the range 0–1
AEO-DMOA	Combined parameters of both AEO and DMOA

Table 2 The datasets description

Dataset	Instances	Features	classes	Domain
Breastcancer	699	9	2	Biology
Churn	3150	16	2	Telecom
IonosphereEW	351	34	2	Electromagnetic
KrvskpEW	3196	36	2	Game
SpectEW	267	22	2	Biology
Vote	300	16	2	Politics
Zoo	101	16	6	Artificial

4.3 Experimental Results and Discussion

In this section, the results of the developed AEO-DMOA are presented. Several evaluation metrics are employed, including accuracy, OFS, and the best, worst, Avg, and standard deviation (SD) of the fitness values to examine the effectiveness of the AEO-DMOA. To provide a fair comparison, the Friedman ranking test is utilized. The accuracy results of the AEO-DMOA are provided in Table 3. According to this table, the developed AEO-DMOA gained the best outcomes in almost all the datasets except for the Churn dataset, while DMOA got the first rank.

The number of the selected OFS by AEO-DMOA and other MH methods are compared in Table 4. According to this table, AEO-DMOA picked the least number of features in five out of seven datasets. For the Churn dataset, the AEO selected the least number of features, and for the KrvskpEW dataset, DMOA selected the least number of features, while in both datasets, the AEO-DMOA ranked second. This analysis shows the capability of the developed AEO-DMOA to select salient features in the datasets while reducing the search area.

Table 5 summarizes the comparative performance of AEO-DMOA and other MH methods in terms of Best, Worst, Avg, and SD of fitness values. For each dataset, ranks are assigned for each MH method based on the performance

Table 3 Comparative analysis using the accuracy of AEO-DMOA and other MH methods

Dataset	PSO	MVO	WOA	SSA	AEO	DMOA	AEO-DMOA
Breastcancer	99.1401	99.1473	99.1255	99.1620	99.1499	99.1626	99.2032
Churn	89.6476	92.9740	95.5873	96.3150	94.8934	95.2101	96.4569
IonosphereEW	93.1179	93.1345	92.3234	92.6408	92.9917	93.5513	92.5147
KrvskpEW	96.3220	96.7343	96.0780	96.4841	96.9089	96.9208	97.1065
SpectEW	87.0948	87.2818	86.0475	86.8869	87.0594	87.4325	87.5323
Vote	64.0552	64.3513	63.2006	63.9436	64.3290	63.6271	64.5523
Zoo	96.7110	97.7109	96.9901	96.4038	97. 6132	97.3910	97.8537

Table 4 Comparative analysis using average OFS of AEO-DMOA and the MH methods

Daraset	PSO	MVO	WOA	SSA	AEO	DMOA	AEO-DMOA
Breastcancer	9	9	9	9	8	8	8
Churn	14	13	9	11	8	11	9
IonosphereEW	4	5	4	4	4	5	4
KrvskpEW	31	29	31	29	27	22	25
SpectEW	11	11	14	11	12	10	9
Vote	6	7	5	7	5	6	5
Zoo	13	11	8	9	6	9	6

Table 5 Comparative performance analysis of the fitness values from different MH methods

Dataset	Metric	PSO	MVO	WOA	SSA	AEO	DMOA	AEO-DMOA
Breastcancer	Best	0.0160	0.0160	0.1605	0.0160	0.1605	0.0160	0.0158
	Worst	0.1095	0.0893	0.0955	0.0816	0.0695	0.0765	0.0589
	Avg	0.0185	0.0184	0.0187	0.0183	0.0181	0.0182	0.0178
	SD	0.0008	0.0010	0.0010	0.0011	0.0010	0.0011	0.0010
	Rank	6	5	7	4	2	3	1
Churn	Best	0.0418	0.0422	0.0403	0.0406	0.0393	0.0415	0.0393
	Worst	0.1346	0.1345	0.0817	0.0491	0.0896	0.0998	0.0484
	Avg	0.1112	0.0774	0.0495	0.0431	0.0541	0.0657	0.0417
	SD	0.0354	0.0353	0.0119	0.0025	0.0238	0.0216	0.0017
	Rank	7	6	3	2	4	5	1
IonosphereEW	Best	0.0904	0.0848	0.1045	0.0903	0.0819	0.0712	0.0862
	Worst	0.0692	0.0694	0.0773	0.0742	0.0726	0.0687	0.1051
	Avg	0.0692	0.0694	0.0773	0.0742	0.0683	0.0651	0.0757
	SD	0.0072	0.0075	0.0094	0.0080	0.0061	0.0035	0.0063
	Rank	3	4	7	5	2	1	6
KrvskpEW	Best	0.0500	0.0518	0.0519	0.0546	0.0426	0.0442	0.0493
	Worst	0.0451	0.0502	0.0404	0.0475	0.0427	0.0353	0.0450
	Avg	0.0264	0.0373	0.0236	0.0379	0.0380	0.0205	0.0312
	SD	0.0049	0.0077	0.0036	0.0099	0.0061	0.0079	0.0062
	Rank	4	5	2	7	6	1	3
SpectEW	Best	0.1227	0.1390	0.1338	0.1227	0.1263	0.1162	0.1032
	Worst	0.1328	0.1307	0.1444	0.1350	0.1328	0.1219	0.1216
	Avg	0.1328	0.1307	0.1444	0.1350	0.1271	0.1286	0.1266
	SD	0.0066	0.0099	0.0088	0.0086	0.0107	0.0076	0.0071
	Rank	5	4	7	6	3	2	1
Vote	Best	0.3756	0.3712	0.3824	0.3734	0.3748	0.3743	0.3703
	Worst	0.3597	0.3574	0.3674	0.3615	0.3568	0.3542	0.3413
	Avg	0.3597	0.3565	0.3674	0.3615	0.3574	0.3644	0.3546
	SD	0.0074	0.0058	0.0082	0.0074	0.091	0.0063	0.0051
	Rank	4	2	7	5	3	6	1
Zoo	Best	0.0731	0.0731	0.0620	0.0421	0.0403	0.0573	0.0413
	Worst	0.0406	0.0369	0.0406	0.0315	0.0383	0.0377	0.0292
	Avg	0.0406	0.0369	0.0406	0.0315	0.0327	0.0313	0.0296
	SD	0.0090	0.0135	0.0109	0.0066	0.0074	0.0065	0.0069
	Rank	7	5	6	3	4	2	1

level of the fitness values. The results are prioritized for ranking based on minimum average, SD, best, and worst fitness values. As per Table 5, the AEO-DMOA gained the first rank in five out of seven datasets. DMOA showed the best fitness value statistics on the IonosphereEW dataset, followed by the AEO. The developed AEO-DMOA is ranked five out of seven, indicating better performance due to switching between the algorithms. For the KrvskpEW dataset, DMOA showed the best fitness value performance, followed by AEO and AEO-DMOA, and the WOA attained the best Avg and SD. PSO method got the SD in both Breastcancer and SpectEW datasets. Overall, the results prove the proposed developed AEO-DMOA's ability to balance the exploration and exploitation phases.

An MH method that reaches a very low fitness value in the smallest number of iterations performs best. The characteristic average convergence curves for 100 iterations by the introduced AEO-DMOA and other MH methods are shown in Fig. 2. The number of iterations is plotted horizontally, while fitness values averaged over 20 independent runs are plotted vertically. From this figure, the AEO-DMOA converges faster speed compared to others on five out of seven datasets. For IonosphereEW and KrvskpEW datasets, the proposed AEO-DMOA performed slightly inferior to AEO and DMOA.

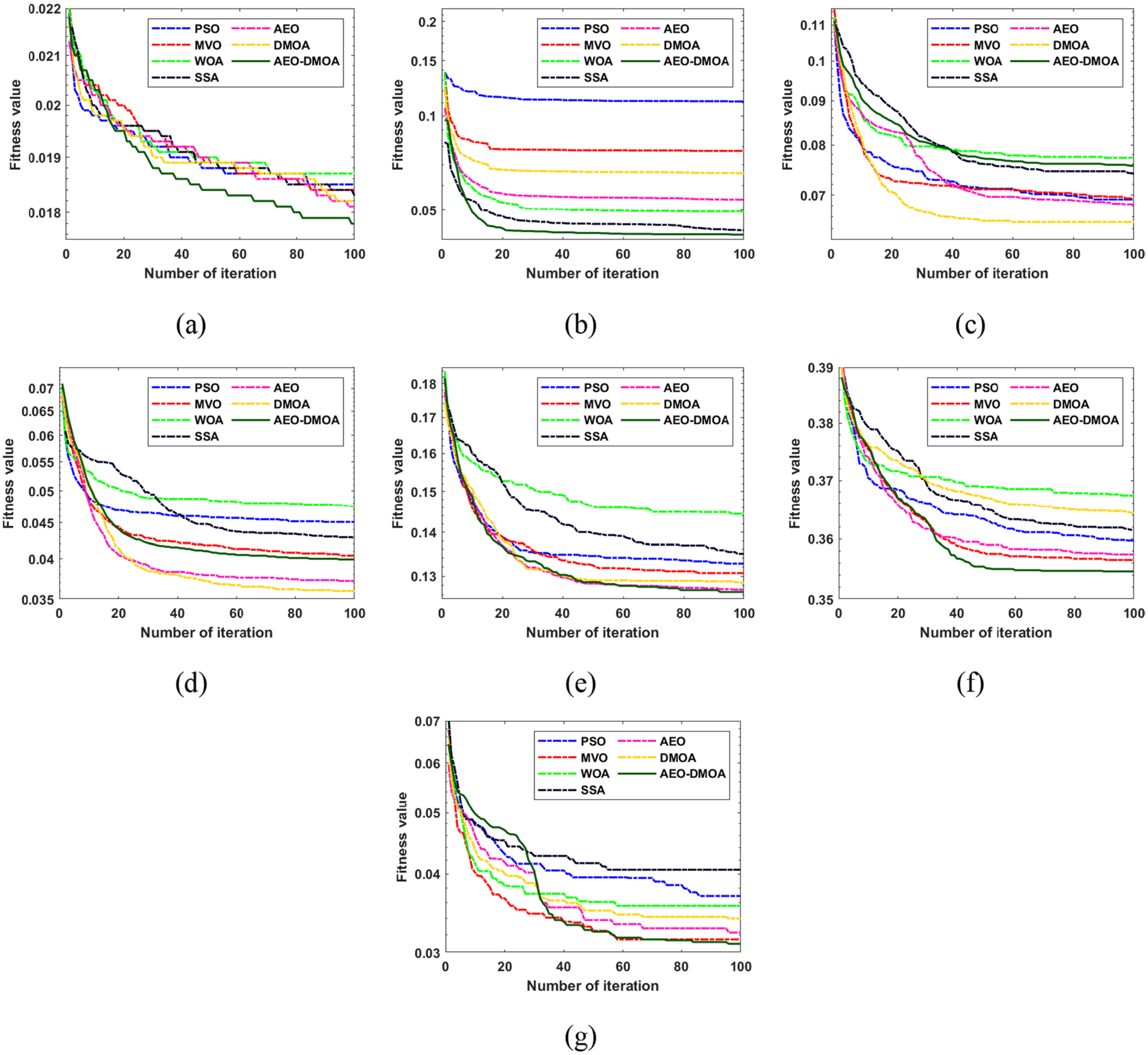

Fig. 2 Convergence analysis of the AEO-DMOA and other MH methods for **a** Breastcancer, **b** Churn, **c** IonosphereEW, **d** KrvskpEW, **e** SpectEW, **f** Vote, and **g** Zoo datasets

4.4 Benchmark Functions

The developed AEO-DMOA is also applied to solve common global optimization problems, using twenty-eight benchmark test functions, eighteen CEC2017 functions, and ten CEC2019 test functions. The AEO-DMOA is compared with several other methods, and the results are given in this section.

4.4.1 CEC2017

To assess the effectiveness of the AEO-DMOA approach, three groups of test functions with various features are used. Many authors widely use these functions in the literature to test the effectiveness of different optimization methods [41]. The functions f_1 to f_7, are called unimodal functions, which have a single extreme point in the search domain, as given in Table 6. Functions f_8 to f_{13}, are called multimodal functions, which have more than an extreme solution and f_{14} to

Table 6 List of CEC2017 benchmark functions

Fun type	Mathematical description	Dimension	Range	Fi^*
Unimodal	$F_1(x) = \sum_{i=1}^{n} x_i^2$	30	[− 100, 100]	100
	$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	[− 10, 10]	200
	$F_3(x) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j \right)^2$	30	[− 100, 100]	300
	$F_4(x) = max_i \left\{ \lvert x_i \rvert ; 1 \le i \le n \right\}$	30	[− 100, 100]	400
	$F_5(x) = \sum_{i=1}^{n-1} \left[100 \left(x_{i+1} - x_i^2 \right)^2 \left(x_i - 1 \right)^2 \right]$	30	[− 30, 30]	500
	$F_6(x) = \sum_{i=1}^{n} \left(\left[x_i + 0.5 \right] \right)^2$	30	[− 100, 100]	600
	$F_7(x) = \sum_{i=1}^{n} i x_i^4 + random[0, 1\rangle$	30	[− 1.28, 1.28]	700
Multimodal	$F_8(x) = \sum_{i=1}^{n} - x_i sin \left(\sqrt{\lvert x_i \rvert} \right)$	30	[− 500, 500]	800
	$F_9(x) = \sum_{i=1}^{n} \left[x_i^2 - 10 coscos \left(2\pi x_i \right) + 10 \right]$	30	[− 5.12, 5.12]	900
	$F_{10}(x) = -20 exp \left(-0.2 \sqrt{ \frac{1}{n} \sum_{i=1}^{n} x_i^2 } \right) - exp \left(\frac{1}{n} \sum_{i=1}^{n} coscos \left(2\pi x_i \right) \right) + 20 + e$	30	[− 32, 32]	1000
	$F_{11}(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} coscos \left(\frac{x_i}{\sqrt{i}} \right) + 1$	30	[− 600, 600]	1100
	$F_{12}(x) = \frac{\pi}{n} \left\{ 10 sinsin \left(\pi y_1 \right) + \sum_{i=1}^{n-1} \left(y_i - 1 \right)^2 \left[1 + 10 sin^2 \left(\pi y_{i+1} \right) \right] + \left(y_n - 1 \right)^2 \right\} + \sum_{i=1}^{n} u \left(x_i, 10, 100, 4 \right),$ $where; y_i = 1 + \frac{x_i + 1}{4},$ $u \left(x_i, a, k, m \right) = \{ k \left(x_i - a \right)^m x_i > a 0 - a < x_i < a k \left(-x_i - a \right)^m x_i < -a$	30	[− 50, 50]	1200
	$F_{13}(x) = 0.1 sin^2 \left(3\pi x_1 \right) + \sum_{i=1}^{n} (x_i - 1)^2 \left[1 + sin^2 \left(3\pi x_1 + 1 \right) \right] + (x_n - 1)^2 1 + sin^2 \left(2\pi x_{n)} \right) + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	30	[− 50, 50]	1300
Multimodal (fixed dim)	$F_{14}(x) = \frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2} \left(x_i - a_{ji} \right)}$	2	[− 65.536, 65.536]	1400
	$F_{15}(x) = \sum_{i=1}^{11} \lfloor a_i - \frac{x_i \left(b_1^2 + b_i x_2 \right)}{b_1^2 + b_i x_3 + x_4} \rfloor^2$	4	[− 5, 5]	1500
	$F_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3} x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4$	2	[− 5, 5]	1600
	$F_{17}(x) = \left(x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6 \right)^2 + 10 \left(1 - \frac{1}{8\pi} \right) coscos x_1 + 10$	2	×1: [− 5, 10] ×2: [0, 15]	1700
	$F_{18}(x) = \lfloor 1 + \left(x_1 + x_2 + 1 \right)^2 \left(19 - 14x_1 + 3x_1^2 - 14x_2 + 16x_1 x_2 + 3x_2^2 \right) \rfloor \times \lfloor 30 + \left(2x_1 - 3x_2 \right)^2 \times \left(18 - 32x_i + 12x_1^2 + 48x_2 - 36x_1 x_2 + 27x_2^2 \right) \rfloor$	2	[− 2, 2]	1800

f_{18} are multimodal functions with fixed dimensions. These functions, along with their details, are given in Table 6. Figures 3, 4, and 5 show the search spaces for unimodal, multimodal, and multimodal with fixed dimensions.

The statistical results from the eighteen functions are provided in Table 7. Each MH algorithm is ranked based on the minimum average fitness value followed by the minimum SD of the fitness values. The results show that AEO-DMOA acquires the best results against its competitors in five out of seven unimodal test functions, four out of six multimodal test functions, and four out of five multimodal test functions with fixed dimensions. WOA is the second best method after AEO-DMOA, followed by AEO, PSO, DMOA, MVO, and SSA.

The AEO-DMOA ranked first in twelve out of all the tested functions using the Friedman rank test. WOA got the first rank in F_2, F_6, F_9, and F_{10}, while PSO is first in F_{16} and F_{17}. The superior performance of the AEO-DMOA over the other used MH methods validates its priority as FS over others. The results also indicate that AEO-DMOA processes high exploitation and has a high capability in exploration. Moreover, statistical rank tests prove that AEO-DMOA is statistically significant improvement compared to other methods.

Figure 6 depicts the convergence behavior of the AEO-DMOA method and other MH algorithms. From these figures, the developed method has a smaller fitness value

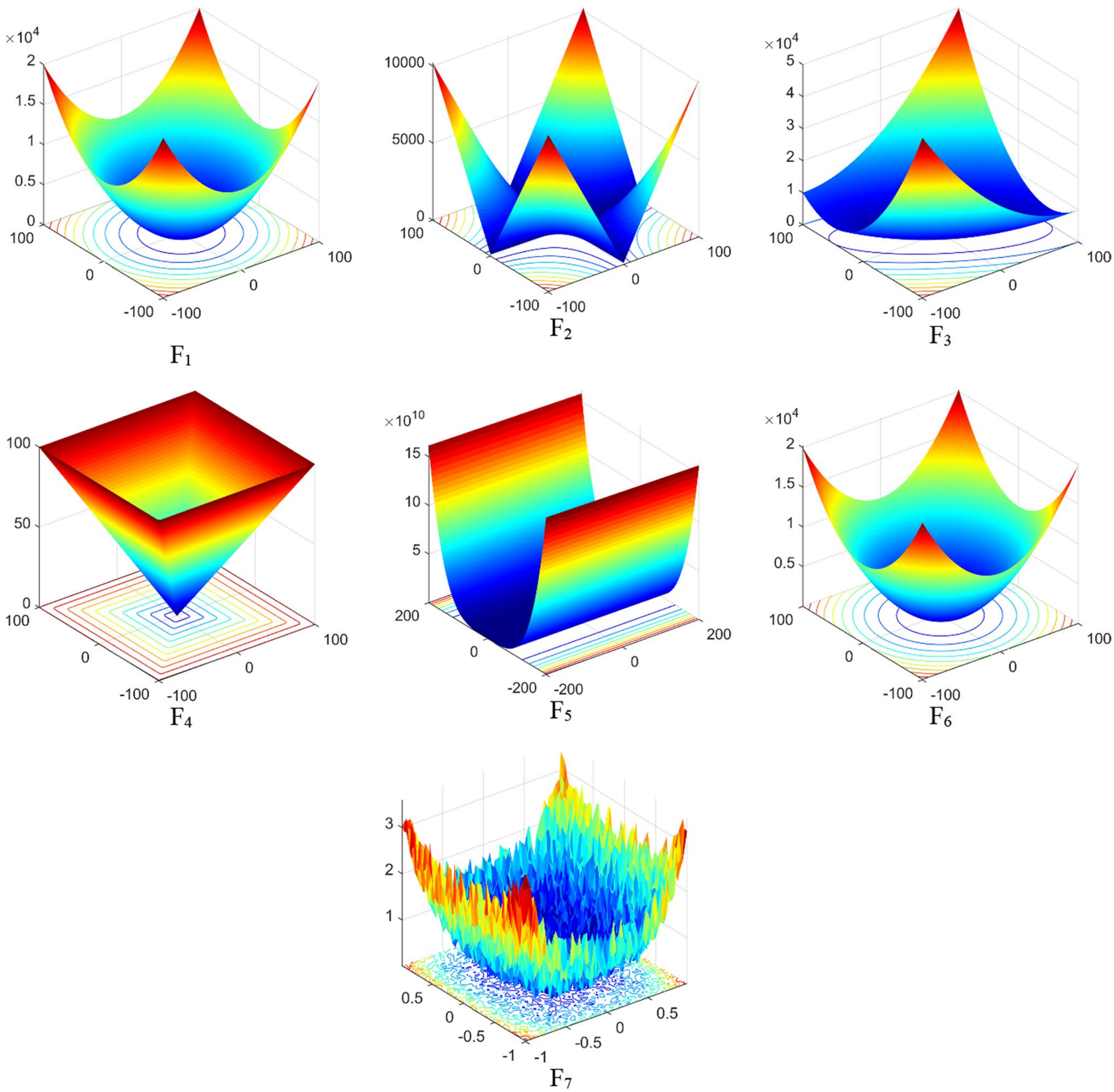

Fig. 3 Search landscape of the CEC2017 unimodal functions

than the other methods for ten functions from CEC2017 comprising five functions from unimodal functions (F_1, F_3, F_4, F_5, and F_7), two multimodal functions (F_8 and F_{11}), and three multimodal functions with fixed dimension (F_{14}, F_{16}, and F_{17}). Considering all the tested functions, the developed method shows the best performance, followed by SSA, WOA, and DMOA and PSO is the worst. For F2, F6, F12, F13, and F15 functions, the developed AEO-DMOA gained the second-best solution.

Table 8. The results of different feature selection methods on CEC2017 functions (dimension = 50).

A similar performance analysis is carried out for all MH algorithms by increasing the dimension of CEC2017 functions to 50. The comparative analysis supports the

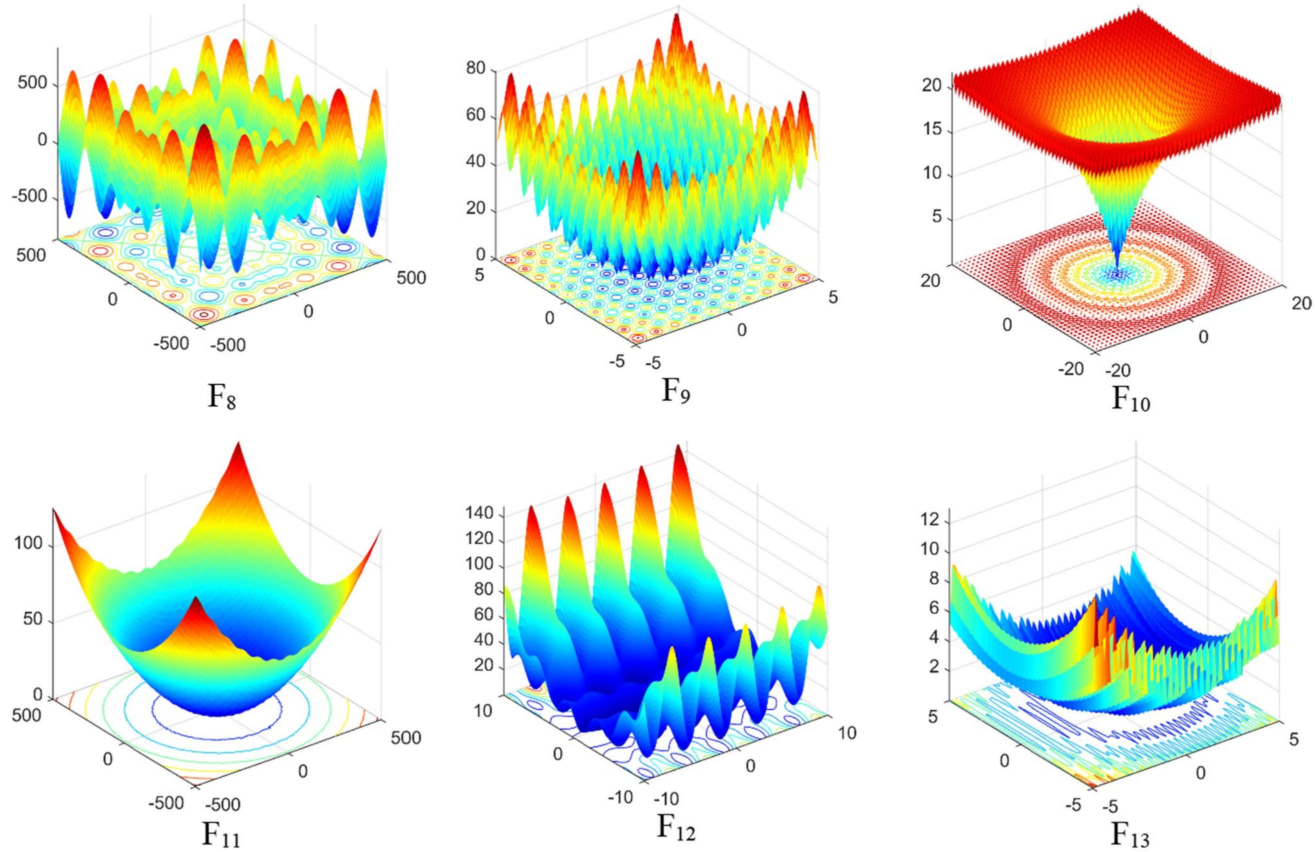

Fig. 4 Search space of the CEC2017 multimodal functions

earlier claim that the developed AEO-DMOA is the best feature selection among all. AEO shows the second-best performance, followed by WOA, PSO, SSA, MVO, and DMOA. It must be noted that increasing the dimensionality of the CEC functions has certainly decreased the DMOA performance, but the hybridization with AEO does not allow the performance to decrease for the developed AEO-DMOA. This proved the cooperative relationship between the two algorithms. Figure 7 shows the convergence plots for CE2017 with higher dimensions for all MH algorithms. CEC2017 function F_{16} showed the same average performance for all MH algorithms, as shown in the convergence plot, and hence ranking is based on SD.

4.4.2 CEC2019

The robustness of the developed AEO-DMOA is verified using the CEC2019 test function [42]. CEC2019 has ten functions, each with dimensions and search range, as shown in Table 9.

The comparison between the AEO-DMOA and the other methods of CEC2019 functions is provided in Table 10. From this table, the AEO-DMOA got more performance than other MH methods on five out of ten tested functions, demonstrating its worthy performance. Like earlier CEC function evaluations, WOA and DMOA are second best, followed by MVO, AEO, SSA, and PSO. The developed AEO-DMOA comes in the first rank in F_2, F_3, F_4, $F_{7,}$ and F_8, WOA in the second rank in F5 F10, and PSO only first in F1, respectively SSA F_6 and DMOA in F_9., indicating that AEO-DMOA is significantly better than all other competing methods.

To further test the performance of the developed AEO-DMOA method to find high-quality solutions using CEC2019 functions, convergence behavior is plotted, as shown in Fig. 8. It can be seen that the AEO-DOMA has

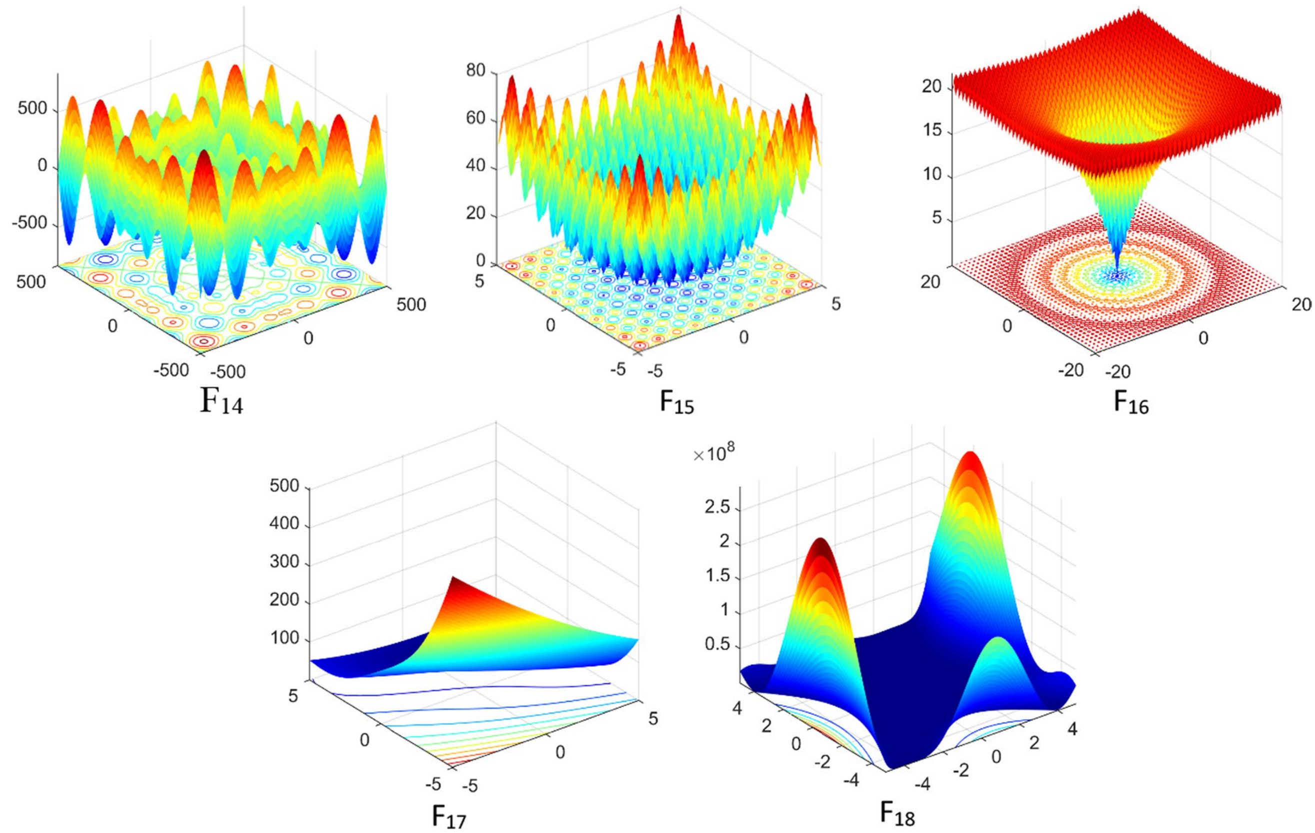

Fig. 5 Search space of the CEC2017 multimodal functions with fixed-dimension

the smallest fitness value of the other methods on seven functions, comprising F_1, F_2, F_3, F_4, F_8, $F_{9,}$ and F_{10} functions. The developed method shows the worst performance for functions F_5, F_6, and F_7. However, the overall ranking shows that the developed AEO-DMOA has the best performance, followed by AEO, WOA, and DMOA, while SSA shows the last performance on the CEC2019.

5 Discussion

One of the main goals of an efficient FS method is to specify the optimal number of required features for the machine learning task and prevent the selection of too many or too few features during the feature selection process. For instance, when too many features are selected in a feature selection method, the probability of selecting redundant and irrelevant features will increase; therefore, the prediction accuracy will decrease. On the other hand,

Table 7 The results of different feature selection methods on CEC2017 functions (dimension = 30)

Fun	Stat	PSO	MVO	WOA	SSA	AEO	DMOA	AEO-DMOA
F_1	Mean	1.57E+02	3.24E+01	3.16E−11	8.91E+02	1.17E−02	6.42E−01	2.20E−14
	SD	7.82E+01	8.55E+00	1.36E−10	4.54E+02	1.28E−02	9.42E−02	3.75E−15
	Rank	7	5	2	6	3	4	1
F_2	Mean	6.08E+00	6.63E+01	3.23E−09	1.77E+01	2.31E−02	1.13E+00	1.03E−05
	SD	4.48E+00	4.35E+01	4.08E−09	3.77E+00	1.01E−02	5.98E−02	2.59E−06
	Rank	4	7	1	6	3	5	2
F_3	Mean	8.91E+03	6.12E+03	8.56E+04	7.01E+03	3.55E+02	1.45E+01	2.29E−08
	SD	2.63E+03	1.78E+03	2.60E+04	4.73E+03	2.69E+02	7.38E+00	4.51E−09
	Rank	7	4	6	5	3	2	1
F_4	Mean	1.72E+01	2.11E+01	6.89E+01	2.03E+01	1.20E+00	6.14E−01	2.54E−06
	SD	2.31E+00	9.30E+00	2.34E+01	4.60E+00	4.39E−01	5.94E−02	2.38E−06
	Rank	4	6	7	5	3	2	1
F_5	Mean	2.26E+04	3.85E+03	2.87E+01	9.52E+04	3.43E+01	1.12E+02	2.13E+01
	SD	2.95E+04	4.27E+03	2.27E−01	8.18E+04	2.01E+01	1.46E+01	8.25E+00
	Rank	2	6	3	7	4	5	1
F_6	Mean	1.69E+02	2.85E+01	1.98E+00	9.39E+02	3.00E+00	9.27E+00	5.85E+00
	SD	6.59E+01	7.01E+00	5.35E−01	3.20E+02	7.12E−01	1.73E−01	2.06E+00
	Rank	6	5	1	7	2	4	3
F_7	Mean	1.79E−01	1.36E−01	1.76E−02	5.58E−01	1.32E−02	8.00E+00	1.01E−03
	SD	6.43E−02	4.54E−02	1.74E−02	1.65E−01	5.96E−03	1.84E+01	6.68E−04
	Rank	5	4	3	6	2	7	1
F_8	Mean	− 7.31E+03	− 7.40E+03	− 9.66E+03	− 6.54E+03	− 5.58E+03	− 4.46E+03	− 1.15E+04
	SD	6.92E+02	7.40E+02	1.51E+03	6.34E+02	9.79E+02	7.35E+02	1.24E+03
	Rank	4	3	2	5	6	7	1
F_9	Mean	9.97E+01	1.67E+02	1.18E−01	8.98E+01	2.75E+01	1.99E+00	3.83E+01
	SD	2.08E+01	2.42E+01	5.28E−01	2.37E+01	9.77E+00	1.15E−03	2.89E+01
	Rank	6	7	1	5	3	2	4
F_{10}	Mean	4.74E+00	3.59E+00	3.07E−07	9.06E+00	2.34E−02	9.05E−01	5.83E−06
	SD	4.39E−01	6.09E−01	5.96E−07	1.30E+00	8.23E−03	6.28E−02	2.34E−06
	Rank	6	5	1	7	3	4	2
F_{11}	Mean	2.49E+00	1.28E+00	7.88E−02	9.33E+00	5.37E−02	4.16E−02	5.09E−12
	SD	7.13E−01	8.75E−02	2.46E−01	3.00E+00	5.00E−02	3.15E−02	1.15E−11
	Rank	6	5	4	7	3	2	1
F_{12}	Mean	9.36E+00	8.25E+00	3.49E−01	2.16E+01	3.55E−01	1.83E+00	3.31E−01
	SD	4.92E+00	3.77E+00	7.00E−02	1.47E+01	2.84E−01	2.25E−02	4.52E−01
	Rank	6	5	2	7	3	4	1
F_{13}	Mean	1.38E+02	1.55E+01	1.22E+00	2.68E+04	2.07E+00	2.80E+00	1.08E+00
	SD	1.40E+02	1.17E+01	3.29E−01	4.05E+04	4.89E−01	3.71E−03	1.29E+00
	Rank	6	5	2	7	3	4	1
F_{14}	Mean	1.78E+00	3.10E+00	4.47E+00	4.09E+00	5.79E+00	1.57E+02	1.61E+00
	SD	2.16E+00	4.04E+00	4.55E+00	3.62E+00	4.47E+00	1.42E+02	2.18E+00
	Rank	2	3	5	4	6	7	1
F_{15}	Mean	1.81E−03	5.46E−03	6.78E−03	8.25E−03	3.66E−03	1.27E−01	1.17E−03
	SD	4.38E−03	8.07E−03	3.49E−04	8.55E−03	7.20E−03	9.28E−02	1.30E−03
	Rank	2	4	5	6	3	7	1
F_{16}	Mean	− 1.03E+00	− 1.03E+00	− 1.03E+00	− 1.03E+00	− 1.03E+00	5.55E−01	− 1.03E+00
	SD	8.74E−15	9.95E−06	4.18E−07	2.16E−13	5.12E−07	1.68E+00	1.44E−11
	Rank	1	6	4	2	5	7	3
F_{17}	Mean	3.98E−01	3.98E−01	4.00E−01	3.98E−01	3.98E−01	5.93E−01	3.98E−01
	SD	5.70E−14	1.50E−05	3.13E−03	2.47E−03	9.51E−04	3.13E−01	2.28E−07
	Rank	1	3	6	5	4	7	2
F_{18}	Mean	3.00E+00	3.00E+00	4.36E+00	3.00E+00	3.00E+00	1.05E+02	2.35E+00
	SD	1.43E−13	9.60E−05	6.08E+00	1.43E−12	1.13E−03	1.33E+02	6.04E+00
	Rank	2	4	7	3	5	6	1

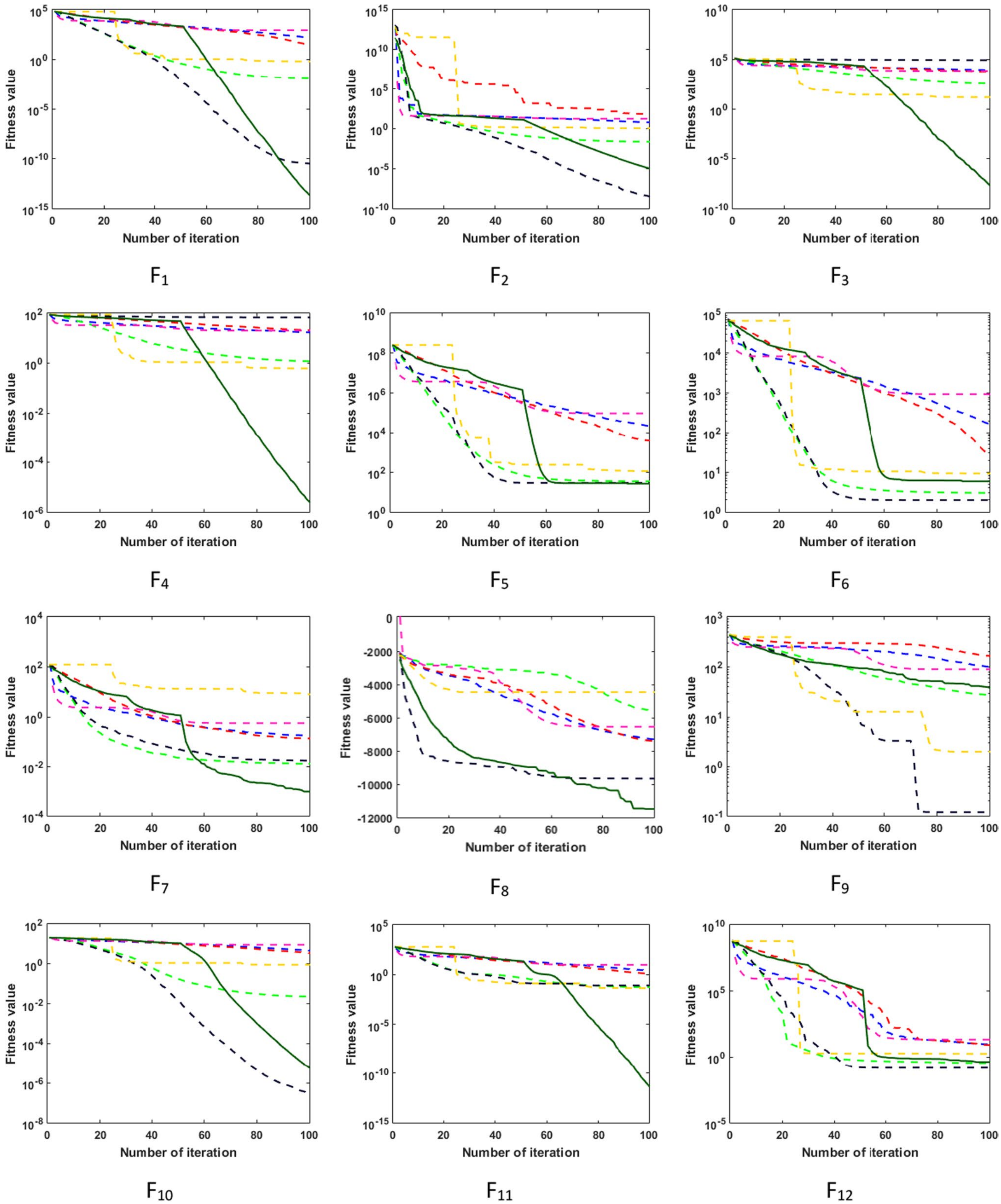

Fig. 6 Convergence curves of the AEO-DMOA and other methods using the tested CEC2017 functions (dimension = 30)

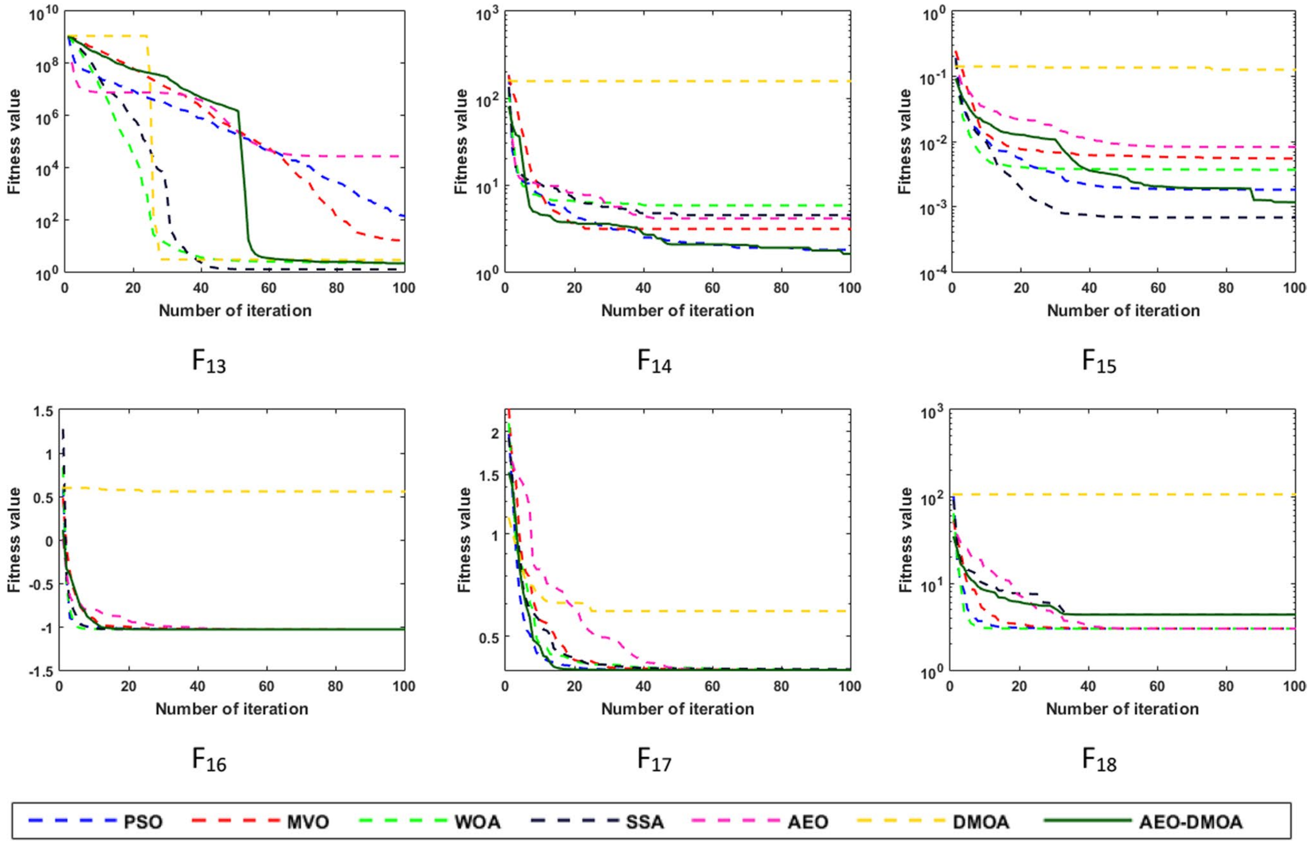

Fig. 6 (continued)

Table 8 The results of different feature selection methods on CEC2017 functions (dimension = 50)

Fun	Stat	PSO	MVO	WOA	SSA	AEO	DMOA	AEO-DMOA
F_1	Mean	1.41E+03	2.89E+02	6.89E−01	7.15E−01	7.64E−12	4.67E+03	3.42E−11
	SD	3.93E+02	8.19E+01	2.52E−01	1.02E−01	1.64E−11	1.42E+03	5.21E−11
	Rank	6	5	3	4	1	7	2
F_2	Mean	3.01E+01	3.00E+11	2.66E−01	1.19E+00	1.13E−08	3.89E+01	5.36E−05
	SD	9.15E+00	1.34E+12	7.44E−02	1.10E−01	2.96E−08	5.76E+00	9.85E−06
	Rank	5	7	3	4	1	6	2
F_3	Mean	2.90E+04	3.25E+04	2.95E+03	3.88E+01	2.57E+05	2.34E+04	4.76E−06
	SD	8.18E+03	5.90E+03	1.34E+03	1.71E+01	5.43E+04	1.99E+04	1.96E−05
	Rank	5	6	3	2	7	4	1
F_4	Mean	3.09E+01	5.07E+01	5.85E+00	6.50E−01	6.67E+01	2.47E+01	8.34E−06
	SD	2.74E+00	8.41E+00	1.73E+00	7.71E−02	2.78E+01	3.92E+00	7.02E−06
	Rank	5	6	3	2	7	4	1
F_5	Mean	3.10E+05	2.81E+04	1.60E+02	1.38E+02	4.87E+01	8.87E+05	4.65E+01
	SD	1.33E+05	2.11E+04	1.90E+02	1.89E+01	4.70E−02	4.62E+05	1.09E+01
	Rank	6	5	4	3	2	7	1
F_6	Mean	1.33E+03	2.81E+02	8.05E+00	1.44E+01	3.65E+00	4.50E+03	1.04E+01
	SD	3.21E+02	6.08E+01	1.35E+00	2.45E−01	9.51E−01	1.82E+03	3.51E+00
	Rank	6	5	2	4	1	7	3
F_7	Mean	7.10E−01	5.90E−01	4.14E−02	8.56E+00	2.22E−02	1.84E+00	1.28E−03
	SD	2.15E−01	1.69E−01	1.60E−02	8.29E+00	2.66E−02	4.70E−01	1.03E−03
	Rank	5	4	3	7	2	6	1
F_8	Mean	− 1.04E+04	− 1.14E+04	− 8.21E+03	− 6.95E+03	− 1.53E+04	− 9.02E+03	− 1.85E+04
	SD	1.18E+03	8.92E+02	1.87E+03	8.65E+02	2.29E+03	1.24E+03	2.60E+03
	Rank	4	3	6	7	2	5	1
F_9	Mean	2.67E+02	3.32E+02	7.09E+01	2.00E+00	1.80E+01	2.26E+02	4.18E+01
	SD	5.26E+01	4.70E+01	3.10E+01	2.14E−02	8.05E+01	3.19E+01	6.56E+01
	Rank	6	7	4	1	2	5	3
F_{10}	Mean	7.52E+00	7.16E+00	2.99E−01	6.41E−01	1.29E−07	1.15E+01	1.60E−05
	SD	5.30E−01	4.53E+00	3.40E−01	1.93E−02	1.48E−07	9.07E−01	5.27E−06
	Rank	6	5	3	4	1	7	2
F_{11}	Mean	1.85E+01	3.56E+00	4.80E−01	2.35E−02	1.23E−01	4.09E+01	1.23E−10
	SD	2.03E+01	6.09E−01	1.38E−01	1.63E−02	3.02E−01	8.05E+00	2.60E−10
	Rank	6	5	4	2	3	7	1
F_{12}	Mean	2.65E+03	3.30E+01	1.15E+00	1.58E+00	1.89E−01	2.26E+03	6.71E−01
	SD	4.84E+03	3.94E+01	3.75E−01	2.35E−02	8.48E−02	3.83E+03	4.82E−01
	Rank	7	5	3	4	1	6	2
F_{13}	Mean	1.85E+05	4.25E+02	6.44E+00	4.80E+00	2.33E+00	8.46E+05	3.63E+00
	SD	1.76E+05	7.32E+02	1.70E+00	2.41E−02	6.95E−01	7.71E+05	2.07E+00
	Rank	6	5	4	3	1	7	2
F_{14}	Mean	1.45E+00	1.93E+00	6.67E+00	1.03E+02	3.65E+00	2.92E+00	1.64E+00
	SD	6.01E−01	2.25E+00	4.24E+00	1.45E+02	3.75E+00	2.30E+00	1.67E+00
	Rank	1	3	6	7	5	4	2
F_{15}	Mean	1.85E−03	6.58E−03	3.61E−03	1.01E−01	7.70E−04	5.46E−03	2.92E−03
	SD	4.37E−03	8.75E−03	7.22E−03	6.94E−02	4.63E−04	7.58E−03	6.27E−03
	Rank	2	6	4	7	1	5	3
F_{16}	Mean	− 1.03E+00	− 1.03E+00	− 1.03E+00	1.70E−01	− 1.03E+00	− 1.03E+00	− 1.03E+00
	SD	1.14E−13	8.62E−06	4.82E−07	8.34E−01	1.49E−07	1.78E−13	1.54E−11
	Rank	1	2	3	7	4	5	6
F_{17}	Mean	3.98E−01	3.98E−01	3.98E−01	7.73E−01	4.00E−01	3.98E−01	3.98E−01
	SD	4.85E−15	1.36E−05	6.28E−05	7.97E−01	3.13E−03	3.24E−13	5.74E−12
	Rank	1	2	3	7	6	4	5
F_{18}	Mean	3.00E+00	3.00E+00	3.00E+00	8.96E+01	4.35E+00	3.00E+00	4.93E+00
	SD	4.17E−14	6.79E−05	1.04E−03	1.06E+02	6.04E+00	2.09E−12	8.61E+00
	Rank	1	2	3	7	5	4	6

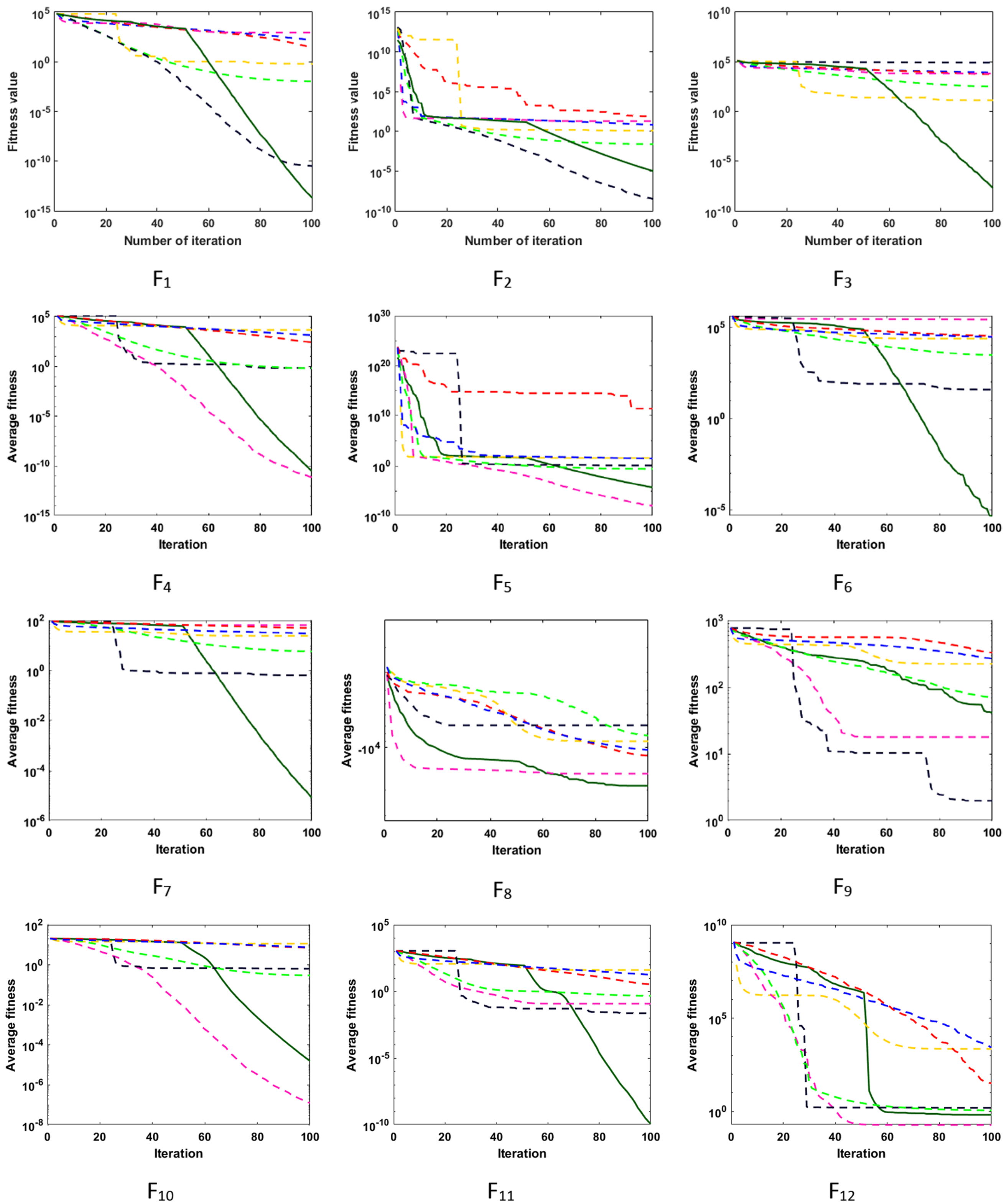

Fig. 7 Convergence curves of the AEO-DMOA and other methods using the tested CEC2017 functions (dimension = 50)

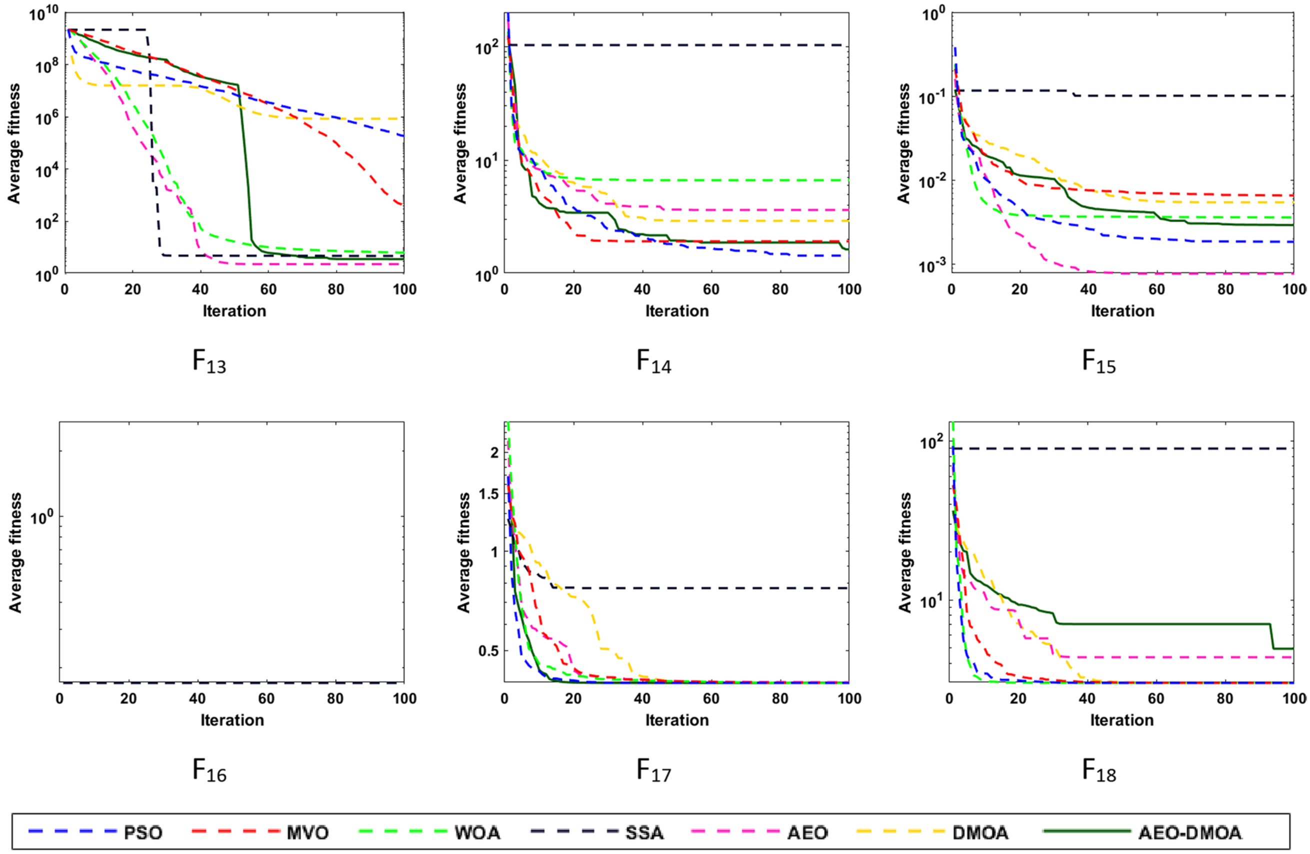

Fig. 7 (continued)

Table 9 CEC2019 benchmark functions

No	Function	$F_I^* = F_I(x^*)$	Dimension	Search range
1	Storn's Chebyshev Polynomial Fitting Problem	1	9	[− 8192, 8192]
2	Inverse Hibert Matrix Problem	1	16	[− 16384, 16384]
3	Lernard-Jones minimum energy cluster	1	18	[− 4, 4]
4	Rastrigin's function	1	10	[− 100, 100]
5	Griewangk's function	1	10	[− 100, 100]
6	Weierstrass function	1	10	[− 100, 100]
7	Modified Schwefel's function	1	10	[− 100, 100]
8	Expanded Schaffer's F6 function	1	10	[− 100, 100]
9	Happy Cat function	1	10	[− 100, 100]
10	Ackley function	1	10	[− 100, 100]

Table 10 Results of the AEO-DMOA using CEC2019 functions

Fun	Stat	PSO	MVO	WOA	SSA	AEO	DMOA	AEO-DMOA
F_1	Mean	5.67E+05	4.02E+08	3.43E+10	1.49E+11	1.32E+09	2.17E+10	9.79E+09
	SD	5.74E+01	5.39E+02	5.34E+03	1.07E+04	1.50E+04	1.74E+03	1.07E+03
	Rank	1	2	6	7	3	5	4
F_2	Mean	1.99E+01	1.75E+01	2.03E+01	1.76E+01	1.74E+01	4.09E+01	1.74E+01
	SD	2.92E−01	1.79E−01	1.29E+01	1.70E−01	7.24E−02	1.14E+01	3.29E−02
	Rank	5	3	6	4	2	7	1
F_3	Mean	1.27E+01	1.27E+01	1.27E+01	1.27E+01	1.27E+01	1.27E+01	1.27E+01
	SD	1.73E−03	3.44E−03	8.80E−04	9.95E−03	2.45E−03	1.16E−03	3.40E−04
	Rank	4	6	2	7	5	3	1
F_4	Mean	1.69E+04	1.94E+02	1.32E+02	2.19E+03	4.27E+02	6.48E+01	6.19E+01
	SD	4.69E+03	1.08E+02	1.17E+02	1.31E+03	7.32E+02	1.60E+01	8.22E+00
	Rank	7	4	3	5	6	2	1
F_5	Mean	5.51E+00	1.56E+00	1.16E+00	2.39E+00	1.50E+00	1.33E+00	1.30E+00
	SD	4.07E−01	2.00E−01	1.22E−01	3.15E−01	2.74E−01	8.90E−02	2.20E−01
	Rank	7	5	1	6	4	3	2
F_6	Mean	1.24E+01	1.16E+01	7.29E+00	1.08E+01	1.20E+01	1.08E+01	1.09E+01
	SD	7.27E−01	9.46E−01	1.82E+00	1.40E+00	7.85E−01	9.89E−01	1.54E+00
	Rank	6	4	7	1	5	2	3
F_7	Mean	1.37E+03	6.55E+02	3.17E+02	8.63E+02	8.10E+02	4.91E+02	2.39E+02
	SD	2.26E+02	2.79E+02	2.51E+02	2.88E+02	3.65E+02	2.97E+02	2.27E+02
	Rank	7	4	2	6	5	3	1
F_8	Mean	6.87E+00	6.00E+00	5.74E+00	6.37E+00	6.03E+00	5.91E+00	5.63E+00
	SD	3.32E−01	6.16E−01	5.81E−01	4.50E−01	6.55E−01	6.86E−01	9.56E−01
	Rank	7	4	2	6	5	3	1
F_9	Mean	3.96E+03	4.46E+00	3.92E+00	1.58E+02	5.09E+00	2.92E+00	3.03E+00
	SD	1.17E+03	7.37E−01	6.39E−01	1.08E+02	1.68E+00	2.94E−01	3.11E−01
	Rank	7	4	3	6	5	1	2
F_{10}	Mean	2.07E+01	2.06E+01	2.00E+01	2.05E+01	2.02E+01	2.05E+01	2.05E+01
	SD	1.08E−01	1.34E−01	7.76E−02	1.09E−01	1.90E+00	1.19E−01	1.33E−01
	Rank	7	6	1	3	2	4	5

when too few features are selected, they cannot represent all the information of original features [40]. However, the developed AEO-DMOA considered that the number of OFS in their fitness function showed better performance, and the number of final selected features by these methods were fewer.

Exploration of the search space and exploitation of the best solutions found are two conflicting objectives that must be taken into account when using MH methods. From the results provided above, AEO-DMOA demonstrated a better performance in balancing the factors of exploration and exploitation and better convergence speed.

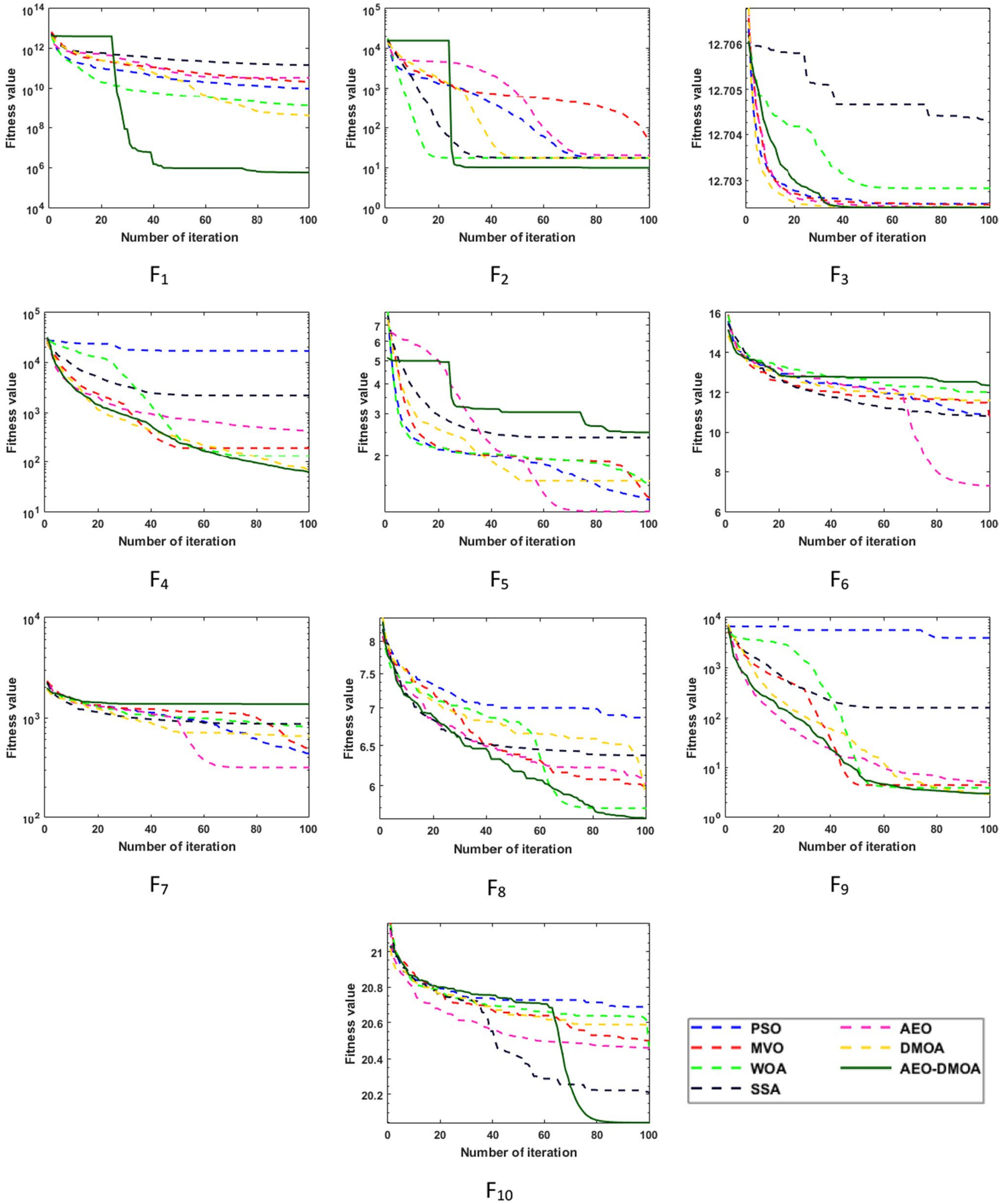

Fig. 8 Convergence behaviour for CEC2019 functions

Based on the previous results and discussion, the developed AEO-DMOA has a high ability to explore the feasible region which contains the optimal solution. However, the time complexity of AEO-DMOA still needs more improvements, when applied to handle high-dimensional data.

6 Conclusion and Future Work

The paper introduces an FS-based approach, named AEO-DMOA, based on a hybridization of AEO and DMOA methods to improve the capabilities in the exploration and exploitation phases. The optimization starts by dividing the defined number of iterations into two parts; in the first half, AEO while DMOA is employed on the remaining number of iterations. The efficiency of the AEO-DMOA is investigated using seven datasets collected from the UCI repository, and an extensive study is performed on twenty-eight test functions, eighteen CEC2017, and ten CEC2019. The simulation and statistical results show that AEO-DMOA is competitive with other well-known MH methods in terms of accuracy, the number of selected OFS, and fitness values. In addition, the AEO-DMOA provides reliable performance on high-dimensional functions. The developed AEO-DMOA method can be used in other applications such as renewable energy, signal processing, and big data. Also, it can be adopted for solving other complex optimization problems such as vehicle routing, timetabling, and engineering design.

Author Contributions All authors contributed equally to this paper.

Funding Open access funding provided by Linköping University. Authors have received no funding.

Data Availability Data availability is available based on reasonable request.

Declarations

Conflicts of interest The authors declare no conflict of interest.

References

1. Zelinka, I.: A survey on evolutionary algorithms dynamics and its complexity–Mutual relations, past, present and future. Swarm Evol. Comput. **25**, 2–14 (2015)
2. Hussien, A.G., Oliva, D., Houssein, E.H., Juan, A.A., Yu, X.: Binary whale optimization algorithm for dimensionality reduction. Mathematics **8**(10), 1821 (2020)
3. Chhabra, A., Hussien, A.G., Hashim, F.A.: Improved bald eagle search algorithm for global optimization and feature selection. Alex. Eng. J. **68**, 141–180 (2023)
4. Büyüksaatçı, S., Baray, A.: A brief review of metaheuristics for document or text clustering. In: Intelligent techniques for data analysis in diverse settings, pp. 252–264. IGI-Global (2016)
5. Dif, N., Elberrichi, Z.: Gene selection for microarray data classification using hybrid meta-heuristics. In: International symposium on modelling and implementation of complex systems, pp. 119–132. Springer, Cham (2018)
6. Razmjooy, N., Estrela, V.V., Loschi, H.J.: A study on metaheuristic-based neural networks for image segmentation purposes. In: Data science, pp. 25–49. CRC Press (2019)
7. Akbar, H., Dewi, S., Rozali, Y.A., Lunanta, L.P., Anwar, N., Anwar, D.: Exploiting facial action unit in video for recognizing depression using metaheuristic and neural networks. In: 2021 1st International conference on computer science and artificial intelligence (ICCSAI), vol. 1, pp. 438–443. IEEE (2021)
8. Khalid, R., Javaid, N.: A survey on hyperparameters optimization algorithms of forecasting models in smart grid. Sustain. Cities Soc. **61**, 102275 (2020)
9. Abualigah, L., Diabat, A.: A comprehensive survey of the Grasshopper optimization algorithm: results, variants, and applications. Neural Comput. Appl. **32**(19), 15533–15556 (2020)
10. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, vol. 4, pp. 1942–1948. IEEE (1995)
11. Mirjalili, S., Mirjalili, S.M., Hatamlou, A.: Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Comput. Appl. **27**(2), 495–513 (2016)
12. Mirjalili, S., Lewis, A.: The whale optimization algorithm. Adv. Eng. Softw. **95**, 51–67 (2016)
13. Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H., Mirjalili, S.M.: Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv. Eng. Softw. **114**, 163–191 (2017)
14. Holland, J.H.: Genetic algorithms. Sci. Am. **267**(1), 66–73 (1992)
15. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. Adv. Eng. Softw. **69**, 46–61 (2014)
16. Zhao, W., Wang, L., Zhang, Z.: Artificial ecosystem-based optimization: a novel nature-inspired meta-heuristic algorithm. Neural Comput. Appl. **32**(13), 9383–9425 (2020)
17. Agushaka, J.O., Ezugwu, A.E., Abualigah, L.: Dwarf mongoose optimization algorithm. Comput. Methods Appl. Mech. Eng. **391**, 114570 (2022)
18. Hashim, F.A., Hussien, A.G.: Snake Optimizer: A novel meta-heuristic optimization algorithm. Knowl. Based Syst. **242**, 108320 (2022)
19. Hashim, F.A., Mostafa, R.R., Hussien, A.G., Mirjalili, S., Sallam, K.M.: Fick's law algorithm: a physical law-based algorithm for numerical optimization. Knowl.-Based Syst. **260**, 110146 (2023)
20. Hu, G., Wang, J., Li, M., Hussien, A.G., Abbas, M.: EJS: Multi-strategy enhanced jellyfish search algorithm for engineering applications. Mathematics **11**(4), 851 (2023)
21. Abdel-Basset, M., Ding, W., El-Shahat, D.: A hybrid Harris Hawks optimization algorithm with simulated annealing for feature selection. Artif. Intell. Rev. **54**(1), 593–637 (2021)

22. Hans, R., Kaur, H.: Hybrid binary Sine Cosine Algorithm and Ant Lion Optimization (SCALO) approaches for feature selection problem. Int. J. Comput. Mater. Sci. Eng. **9**(01), 1950021 (2020)
23. Kareem, S.S., Mostafa, R.R., Hashim, F.A., El-Bakry, H.M.: An effective feature selection model using hybrid metaheuristic algorithms for iot intrusion detection. Sensors **22**(4), 1396 (2022)
24. Al-Shourbaji, I., Helian, N., Sun, Y., Alshathri, S., Abd Elaziz, M.: Boosting ant colony optimization with reptile search algorithm for churn prediction. Mathematics **10**(7), 1031 (2022)
25. Al-Shourbaji, I., Kachare, P.H., Alshathri, S., Duraibi, S., Elnaim, B., Elaziz, M.A.: An efficient parallel reptile search algorithm and snake optimizer approach for feature selection. Mathematics **10**(13), 1–19 (2022)
26. Cuevas, E., Gálvez, J., Toski, M., Avila, K.: Evolutionary-Mean shift algorithm for dynamic multimodal function optimization. Appl. Soft Comput. **113**, 107880 (2021)
27. Rodríguez, A., Camarena, O., Cuevas, E., Aranguren, I., Valdivia-G, A., Morales-Castañeda, B., Pérez-Cisneros, M.: Group-based synchronous-asynchronous grey wolf optimizer. Appl. Math. Model. **93**, 226–243 (2021)
28. Yu, H., Jia, H., Zhou, J., Hussien, A.: Enhanced Aquila optimizer algorithm for global optimization and constrained engineering problems. Math. Biosci. Eng. **19**(12), 14173–14211 (2022)
29. Hussien, A.G., Hashim, F.A., Qaddoura, R., Abualigah, L., Pop, A.: An enhanced evaporation rate water-cycle algorithm for global optimization. Processes **10**(11), 2254 (2022)
30. Hashim, F.A., Khurma, R.A., Albashish, D., Amin, M., Hussien, A.G.: Novel hybrid of AOA-BSA with double adaptive and random spare for global optimization and engineering problems. Alex. Eng. J. **73**, 543–577 (2023)
31. Azadifar, S., Rostami, M., Berahmand, K., Moradi, P., Oussalah, M.: Graph-based relevancy-redundancy gene selection method for cancer diagnosis. Comput. Biol. Med. **147**, 105766 (2022)
32. Rostami, M., Berahmand, K., Nasiri, E., Forouzandeh, S.: Review of swarm intelligence-based feature selection methods. Eng. Appl. Artif. Intell. **100**, 104210 (2021)
33. Zheng, R., Hussien, A.G., Qaddoura, R., Jia, H., Abualigah, L., Wang, S., Saber, A.: A multi-strategy enhanced African vultures optimization algorithm for global optimization problems. J. Comput. Design Eng. **10**(1), 329–356 (2023)
34. Hussien, A., Liang, G., Chen, H., Lin, H.: A double adaptive random spare reinforced sine cosine algorithm. Comput. Model. Eng. Sci. **136**(3), 2267–2289 (2023)
35. Singh, S., Singh, H., Mittal, N., Hussien, A.G., Sroubek, F.: A feature level image fusion for Night-Vision context enhancement using Arithmetic optimization algorithm based image segmentation. Expert Syst. Appl. **209**, 118272 (2022)
36. El-Dabah, M.A., El-Sehiemy, R.A., Becherif, M., Ebrahim, M.A.: Parameter estimation of triple diode photovoltaic model using an artificial ecosystem-based optimizer. Int. Trans. Electr. Energy Syst. **31**(11), e13043 (2021)
37. Ewees, A.A., Abualigah, L., Yousri, D., Sahlol, A.T., Al-qaness, M.A., Alshathri, S., Elaziz, M.A.: Modified artificial ecosystem-based optimization for multilevel thresholding image segmentation. Mathematics **9**(19), 2363 (2021)
38. Hassan, M.H., Kamel, S., Salih, S.Q., Khurshaid, T., Ebeed, M.: Developing chaotic artificial ecosystem-based optimization algorithm for combined economic emission dispatch. IEEE Access **9**, 51146–51165 (2021)
39. Kamal Kumar, U., Janamala, V.: Artificial ecosystem-based optimization for optimal location and sizing of solar photovoltaic distribution generation in agriculture feeders. In: Congress on intelligent systems, pp. 743–757. Springer, Singapore (2022)
40. Mostafa, R.R., Ewees, A.A., Ghoniem, R.M., Abualigah, L., Hashim, F.A.: Boosting chameleon swarm algorithm with consumption AEO operator for global optimization and feature selection. Knowl.-Based Syst. **246**, 108743 (2022)
41. Wu, G., Mallipeddi, R., Suganthan, P. N.: Problem definitions and evaluation criteria for the CEC 2017 competition on constrained real-parameter optimization. National University of Defense Technology, Changsha, Hunan, PR China and Kyungpook National University, Daegu, South Korea and Nanyang Technological University, Singapore, Technical Report. (2017)
42. Price, K. V., Awad, N. H., Ali, M. Z., Suganthan, P. N.: Problem definitions and evaluation criteria for the 100-digit challenge special session and competition on single objective numerical optimization. In Technical Report. Singapore: Nanyang Technological University. (2018)