IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 13, 2019, accepted May 9, 2019, date of publication May 17, 2019, date of current version July 3, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2917502

BIFFOA: A Novel Binary Improved Fruit Fly Algorithm for Feature Selection

YUN HOU[1,3], JINGYAO LI[1,2], HAIHONG YU[1,2], AND ZHANSHAN LI[1,2]
[1]Key Laboratory for Symbol Computation and Knowledge Engineering of National Education Ministry, Changchun 130012, China
[2]College of Computer Science and Technology, Jilin University, Changchun 130012, China
[3]College of Software, Jilin University, Changchun 130012, China

Corresponding author: Zhanshan Li (lizs@jlu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61672261, in part by the Jilin Province Natural Science Foundation under Grant 20180101043JC, and in part by the Jilin Province Development and Reform Commission under Grant 2019C053-9.

ABSTRACT Feature selection is an important method to reduce the number of attributes of high-dimensional data and an essential preprocess work in classification. It eliminates irrelevant, redundant, and noisy features improves the performance of the model and reduces the computational burden. Fruit fly optimization algorithm is a new algorithm proposed in recent years, which imitates the foraging behavior of fruit fly. To the best of our knowledge, it has not been systematically applied to feature selection. This paper uses the fruit fly optimization algorithm as a search strategy and designs a wrapper-based feature selection method, named binary improved fruit fly optimization algorithm (BIFFOA). Besides, four different strategies based on evolutionary population dynamics (EPD) and new mutation operators are employed to enhance the BIFFOA. The extensive experiments on 25 datasets (see Table 1) show that the performance of the BIFFOA is better than several state-of-the-art algorithms.

INDEX TERMS Classification, evolutionary population dynamics, feature selection, fruit fly optimization algorithm.

I. INTRODUCTION

In data mining, machine learning, pattern recognition, etc., datasets usually contain a large number of irrelevant, redundant, noisy features, which may reduce the efficiency of learning algorithms or lead to overfitting. Feature selection methods select a small subset that only contains relevant features, which help data mining and machine learning algorithms to work faster and more efficiently [1]–[3].

According to Liu and Motoda [4], feature selection algorithms can be classified based on two main criteria: the subset evaluation process and the search process. In terms of the former, feature selection is usually divided into two broad categories: filter and wrapper [5]. The filter-based method evaluates data features based on the information contained or statistical metrics. This kind of approach is very popular in high dimensional feature selection problems. The wrapper-based method is usually correlated to predetermined learning algorithms. The classification accuracy of the wrapper-based method is generally higher than that of the filtered-based method, which does not rely on any learning algorithm [6].

In recent years, meta-heuristic algorithms based on natural heuristics have taken the lead in dealing with complex real-world problems due to their powerful and efficient performance [7], [8]. Some of the most popular meta-heuristic algorithms are particle swarm optimization (PSO) [9], artificial bee colony (ABC) [10] and ant colony optimization (ACO) [11]. New optimization algorithms with specific global and local search strategies have emerged, such as grey wolf optimizer algorithm (GWOA) [12], the whale optimization algorithm (WOA) [13], grasshopper optimization algorithm (GOA) [14] and salp swarm algorithm (SSA) [15], etc.

Many meta-heuristic algorithms have been used to solve the problem of feature selection, such as bare bones particle swarm optimization algorithm (BPSO) [16], genetic algorithm feature selection (GAFS) [17], binary bat algorithm (BBA) [18], salp swarms algorithm (SSA) [19], binary gravitational search algorithm (BGSA) [20], binary grasshopper optimization algorithm (BGOA) [21], binary gray wolf optimization algorithm (BGWOA) [22] etc.

The associate editor coordinating the review of this manuscript and approving it for publication was Dongxiao Yu.

The no-free-lunch (NFL) theorem logically proves that no one can give an algorithm for solving all optimization problems. That theorem means that the success of the algorithm in solving a specific set of issues does not guarantee the solution of all optimization problems of different types and properties [23]. When considering all optimization problems, the average performance of all optimization techniques is the same, although they have superior performance on a subset of optimization problems. The NFL theorem encourages researchers to propose new optimization algorithms or to improve/modify existing algorithms to solve different issues [24].

Faris et al. proposed an efficient crossover scheme to improve the performance of BSSA for feature selection in [3]. The binary gravitational search algorithm (BGSA) was introduced in [25]. Majdi Mafarja et al. combined the Grasshopper optimization algorithm (GOA) with evolutionary population dynamics (EP) to find the optimal feature subset in the feature set [26].

Mirjalili and Lewis proposed a meta-heuristic algorithm called fruit fly optimization algorithm (FOA) [27]. As a new population-based meta-heuristic algorithm, compared with other optimization algorithms, FOA has the advantages of a simple parameter initialization process, simple structure, convenient implementation, and excellent performance [28]. FOA has been used to solve a variety of complex scheduling problems, including semiconductor final test scheduling problems [29], steel making casting problems [30], flow shop scheduling problems [31], and parallel machine green scheduling problems [32]. In addition, FOA also has excellent performance in other optimization fields, such as power load forecasting [33], set coverage problem [34], PID control [35], knapsack problem [28], [37], optimal gating system design of steel casting [13] and homogeneous fuzzy string parallels redundancy allocation problem [36].

However, as far as we know, there is no suitable binary fruit fly optimization algorithm for wrapper-based feature selection currently. Basic FOA generates food sources around the population located within a fixed radius of 1. The transition from the exploration stage to the exploitation stage of the algorithm is not smooth, and it usually takes several iterations to find the optimal solution. To overcome these drawbacks, an improved fruit fly optimization algorithm (IFFOA) [38] was proposed. IFFOA with dynamic search radius is the first fruit fly optimization algorithm to solve the high dimensional functions. In [38], IFFOA algorithm performs well when dealing with various optimization problems. The above reasons encourage us to choose IFFOA as the basis of our work.

This paper improves IFFOA algorithm with EPD and selection operator and obtains a novel algorithm called binary improved fruit fly optimization algorithm (BIFFOA) that can deal with feature selection tasks efficiently. We use EPD since it is a simple but effective operator for population-based techniques [26]. In this work, we have made the following three contributions:

- Binary improved fruit fly optimization algorithm (BIFFOA) is proposed.
- Combining with the evolutionary population dynamics mechanism (EPDM) makes BIFFOA more effective in dealing with the feature selection problem.
- A new mutation operator combined with four different EPDM is proposed.

II. RELATED WORK

A. BASIC FRUIT FLY OPTIMIZATION ALGORITHM (FOA)

Basic FOA, inspired by the foraging behavior of fruit flies in nature, is proposed by Pan [39]. The foraging behavior of fruit fly has divided into two stages: the olfactory search stage and visual search stage. During olfactory foraging, fruit fly searches and locates food sources around the population, and then evaluates the odor concentration corresponding to each possible food source. In the visual foraging phase, the best food source with the maximum smell concentration value is found, and then the fruit fly group flies towards it [38]. The procedure of the FOA is summarized as follows:

Step 1: Initialize parameters, including the maximum number of iterations and population size.

Step 2: Initialize the fruit fly swarm location.

Step 3 Olfactory foraging phase: generate several fruit flies randomly around the current fruit fly swarm location to construct a population:

Step 4: Evaluate the population to obtain the fitness value of each fruit fly.

Step 5 Visual foraging phase: find the fruit fly with the best fitness value, and then the fruit fly group flies towards the best one.

Step 6: If the maximum number of iterations is reached, the algorithm is terminated; otherwise, go back to Step 3.

B. IMPROVED FRUIT FLY OPTIMIZATION ALGORITHM (IFFOA)

Instead of generating new solutions by changing all the decision variables of the population location like the original FOA, IFFOA generates new solutions by randomly selected indexes to enhance the search in the development stage.

$$\lambda=\lambda_{\max} \cdot \exp(\log \frac{\lambda_{\min}}{\lambda_{\max}}) \cdot \frac{Iter}{Iter_{\max}} \tag{1}$$

In Eq.(1), λ represents the search radius of fruit flies in each iteration, $\lambda_{\max}$ is the maximum search radius, and $\lambda_{\min}$ is the minimum search radius. *Iter* represents the current iteration number, and *Max_Iter* represents the maximum iteration number.

$$x_{i,j} = \begin{cases} \delta_j \pm \lambda \cdot rand() & if\ j = d \\ \delta_j & otherwise, \quad j = 1, 2, \ldots n \end{cases} \tag{2}$$

$d \in \{1, 2, \ldots, n\}$ is an index randomly selected from uniformly distributed decision variables, n is the dimension of the solution, rand () is a random number within the range of [0,1], and the location of $x_{i,j}$ is updated by Eq.(2). δ_j is the value of the optimal solution in the j-th dimension.

C. EVOLUTIONARY POPULATION DYNAMICS (EPD)

Evolutionary algorithm (EA) is a random search mechanism. Some EAs apply crossover and mutation operators to alter the selected solution in order to evolve the best individuals. In contrast to EA, EPD is the process of relocating the worst solution in a population. Its purpose is to eliminate the bad solution in the population rather than develop the best individual [17]. EPD is based on self-organized criticality theory (SOC) [40]. In this theory, local changes in population may affect the entire population, providing a delicate balance without external forces.

The main reason for the success of EPD is to eliminate the worst individuals, thereby increasing the median of the population. Removing the worst individual is the first step when using EPD in a population-based algorithm. The next step is to mutate or relocate the removed individuals according to the best solution [41].

III. PROPOSED METHOD

A. EXPRESSION OF THE SOLUTION

Like most existing studies [18], [22], [26], feature selection is considered as a binary optimization problem. We use binary strings to represent the solution of the feature selection problem. The vector contains d elements, where d represents the number of features in the original data set. If we select the corresponding features then, set them to "1", otherwise, set them to "0". The decision variables of the problem are described as follows:

$$X = (x_1, x_2, \ldots, x_d), \quad x_i \in [0, 1], \; i = 1, 2, \ldots, d. \tag{3}$$

B. FITNESS FUNCTION

Feature selection is also considered as a multi-objective optimization problem. To maintain a balance between the number of features selected and the classification accuracy of the solution, the fitness function is designed as follows:

$$Fitness = \alpha\gamma_R(D) + \beta\frac{|R|}{|C|} \tag{4}$$

γ_R represents the classification error rate of a given classifier (where k-nearest neighbor classifier (KNN) is used in this paper). α represents the weight of classification accuracy, and β represents the weight of feature reduction. $|R|$ represents the number of features selected, $|C|$ represents the total number of features.

C. A BINARY IMPROVED FRUIT FLY OPTIMIZATION ALGORITHM(BIFFOA) FOR FEATURE SELECTION

In the wrapper-based feature selection methods, the search for space is nonlinear, and there is a large number of local minima. Hence, an intelligent optimization method is required to reduce the number of evaluations. As reported in the literature [28], IFFOA algorithm shows good results when dealing with various optimization problems. The advantages of IFFOA prompted us to propose a binary version of the IFFOA optimization algorithm and use it as the core search engine to solve the feature selection problem. According to Mirjalili and Lewis [27], one of the easiest ways to convert an algorithm from continuous to the binary version without modifying its structure is to utilize transfer functions. Sigmoidal (S-shaped) function is a common transfer function (see Eq.(6)).

$$x_{i,j} = \begin{cases} 1 - \delta_j & if\ s(\Delta x_{i,j}) \geq rand() \\ \delta_j & otherwise, \quad j = 1, 2, \ldots n \end{cases} \tag{5}$$

$$s(\Delta x_{i,j}) = \frac{1}{1 + e^{-\Delta x_{i,j}}} \tag{6}$$

$$\Delta x_{i,j} = \begin{cases} \delta_j \pm r \cdot rand() & if\ j = d \\ \delta_j & otherwise,\ j = 1, 2, \ldots n \end{cases} \tag{7}$$

$$r = r_{\max} \cdot \exp(\log \frac{r_{\min}}{r_{\max}}) \cdot \frac{Iter}{Iter_{\max}} \tag{8}$$

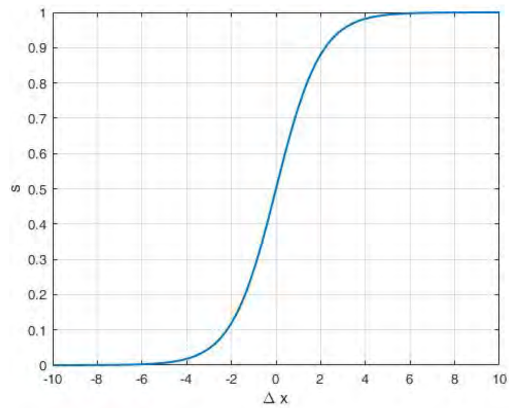


FIGURE 1. S-shaped transfer function.

Figure 1 shows the transfer functions. The position of the current fruit fly will be updated as in Eq.(5), where δ_j is the value of the optimal solution in the j-th dimension. Through the transfer function $\Delta x_{i,j}$ in IFFOA is converted into the probability $s(\Delta x_{i,j})$ of fruit fly updating its position. $\Delta x_{i,j}$ is calculated in Eq.(7), where rand(), r and d $\in \{1, 2, \} \ldots, n$ are a random number in the range of [0, 1], the search radius of fruit flies in each iteration and a randomly chosen index, respectively. The pseudo code of BIFFOA algorithm is given in Algorithm 1.

D. FRUIT FLY FEATURE SELECTION OPTIMIZATION ALGORITHM COMBINED WITH EPD

In [17], the EPD mechanism was proposed, and the grasshopper feature selection optimization algorithm was improved by this mechanism. Inspired by [17], this paper combines the new mutation operator with the fruit fly feature selection algorithm using four different EPD strategies in order to strengthen the exploration and development ability of BIFFOA. Next, the EPD mechanisms used are described in detail:

BIFFOA_EPD: An example of the BIFFOA_EPD is illustrated in Figure 2. In this method, the best three individuals

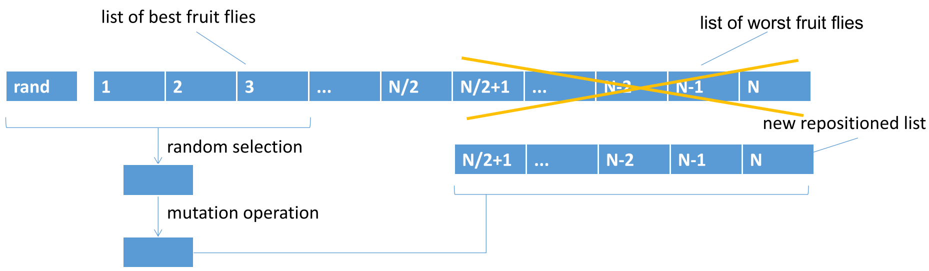


FIGURE 2. The mechanism of BIFFOA_EPD.

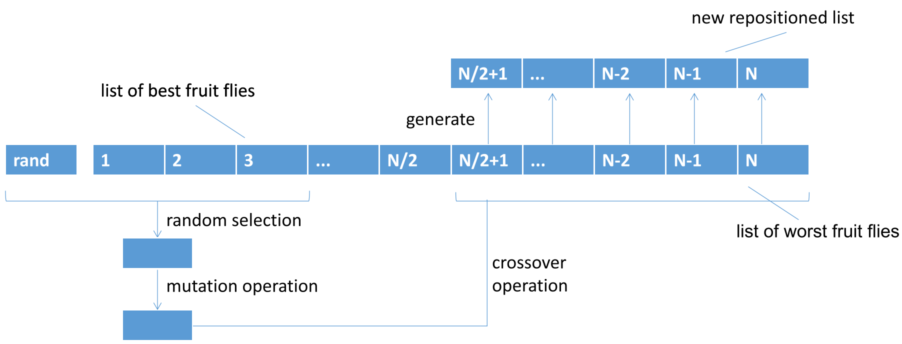


FIGURE 3. The mechanism of BIFFOA_EPD_CM.

Algorithm 1 The BIFFOA Algorithm

Input: PS, $\lambda_{\max}$, $\lambda_{\min}$, $Iter_{\max}$
Output: Solution X*
1//Initialize the BIFFOA parameter:
2: Set PS, $\lambda_{\max}$, $\lambda_{\min}$, $Iter_{\max}$
3: Calculate the fitness of all agents
4: Set the best solution as swarm location
5: $Iter = 0$
6: $X^* = \Delta$
7: **Repeat**
8: Calculate the search radius r using Eq.(8)
9: Calculate $\Delta x_{i,j}$ using Eq.(7)
10: //Osphres is foraging phase
11: **For** i=1,2,…,PS
12 Calculate $s(\Delta x_{i,j})$ using Eq.(6)
13: Using Eq.(5) to generate food source $x_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,n})$
14: **End For**
15: //Vision foraging phase
16: Calculate the fitness of all agents
17: Update swarm location if there is a better solution in population
18: **Until** $Iter = Iter_{max}$

are selected, and a fourth solution is randomly generated. The poorer flies in the latter half are repositioned with equal probability around any one of the four. Relocating around the best three flies each time might cause premature population convergence and fall into local optimization. Thus randomly generated fruit flies are added.

BIFFOA_EPD_CM: This version is similar to BIFFOA_EPD, except that it also uses a crossover (Eq.(9)) and mutation operators (Eq.(10)). In this method, a random number is generated, a solution similar to the first strategy is selected, and the selected solution is mutated. The mutated solution is then crossover with a weak solution. (see Figure 3)

BIFFOA_EPD_Tour: In this version, the Tournament Selection (TS) operator is used to select a solution from the first half of the group. In the TS operator, we randomly select t individuals from the whole population and then select the best individuals among the selected t individuals. Moreover, then the same crossover (Eq.(9)) and mutation operator (Eq.(10)) as BIFFOA_EPD_CM is applied to the obtained solution. The advantage of TS is that it provides an opportunity for all individuals to guide the different solutions, thus maintaining the diversity of fruit fly feature selection algorithm. In this paper, we set t to 3 [17]. (see Figure 4)

BIFFOA_EPD_RWS: as shown in Figure 5, this version is similar to the BIFFOA_EPD_Tour version, and the only difference is that it uses the Roulette Wheel Selection (RWS) operator instead of the TS operator. The probability of individuals selected by the RWS operator is based on their fitness values. Each in the group is designated as a small piece of roulette. The size of the block is in direct proportion to the individual's fitness value. The better the individual, the larger the area of the corresponding block in roulette. Rotate the

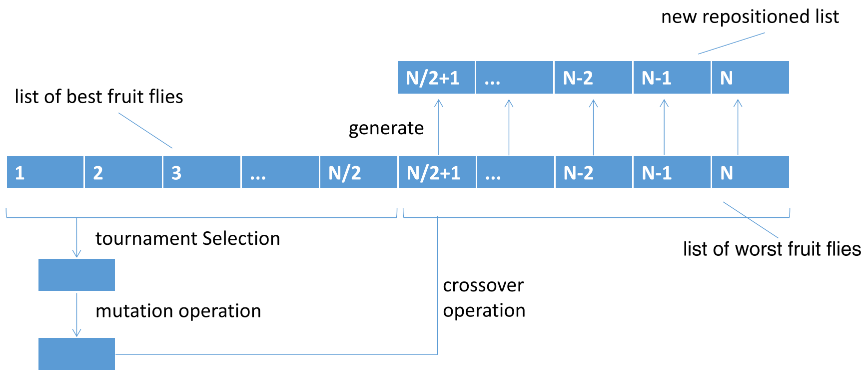


FIGURE 4. The mechanism of BIFFOA_EPD_Tour.

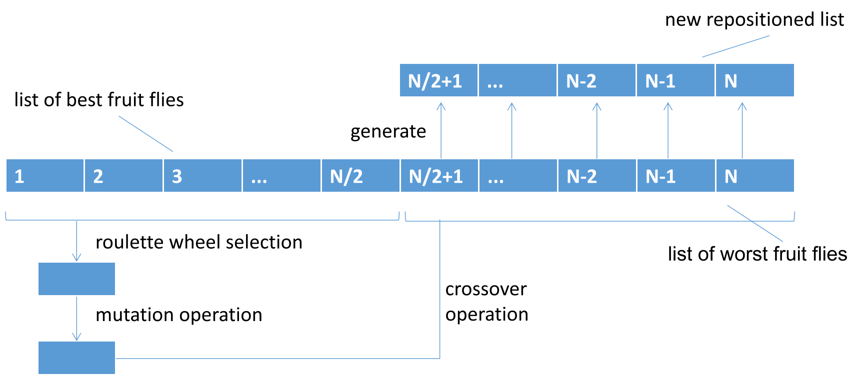


FIGURE 5. The mechanism of BIFFOA_EPD_RWS.

roulette, and when the roulette stops, select the individual on which the pointer stops. After selecting a solution with the RWS operator, we mutate it to explore more feature space regions. Then, the crossover (Eq.(9)) and mutation operator (Eq.(10)) are used to reposition the original differential solution [17]. The advantage of RWS is that it does not ignore any individual in the population, so it makes the population more diverse.

E. CROSSOVER OPERATOR AND MUTATION OPERATOR

1) MUTATION OPERATOR

In EPD, we use η to express the new mutation rate operator and it is calculated by Eq.(9). The mutation rate is iteratively refined by the new mutation operator.

$$\eta = 0.9 - \frac{0.89}{1 + e^{5 - \frac{10iter}{Max\_Iter}}} \tag{9}$$

$$x^d = \begin{cases} 1 - x^d & \eta \geq rand() \\ x^d & otherwise \end{cases} \tag{10}$$

η ranges from 0.9 to 0.01. *Iter* represents the number of current iterations. *Max_Iter* represents the maximum number of iterations. According to Eq.(10), the current solution is updated by inversion of different number.

2) CROSSOVER OPERATOR

The crossover operator used in the hybrid algorithm is represented in Eq.(11). [3]

$$x = \bowtie (x_a, x_b) \tag{11}$$

$$x^d = \begin{cases} x_a^d & rand() \leq 0.5 \\ x_b^d & otherwise \end{cases} \tag{12}$$

$\bowtie$ means cross operation. x_a and x_b represent two solutions that are to be crossed, d is the d-th dimension of the solution. $rand()$ represents a random number within the range of [0,1].

The pseudo code of BIFFOA with EPD is given in Algorithm 2.

F. COMPLEXITY ANALYSIS

In the proposed algorithm, O(1) essential operation is required to set the size of the group, the maximum number of iterations, the initial individual position, the calculation of individual fitness value, the setting of the group position and the check of termination conditions. Updating the swarm location needs $O(t \times d)$ essential operation. Generating food source for each particle needs $O(t \times d \times n)$ basic operations, where t indicates the number of iterations, d is the number of variables, and n shows the number of solutions. Binary operators do not change the computational complexity since they have been applied to the position. The computation

Algorithm 2 The BIFFOA_EPD Algorithm

Input: PS, $\lambda_{\max}$, $\lambda_{\min}$, $Iter_{\max}$
Output: Solution X*
1: //Initialize the BIFFOA parameter:
2: Set Parameters PS, $\lambda_{\max}$, $\lambda_{\min}$, $Iter_{\max}$
3: Calculate the fitness of all agents
4: Set the best solution as swarm location
5: $Iter = 0$
6: $X* = \Delta$
7: **Repeat**
8: Calculate the search radius r using Eq.(8)
9: Calculate$\Delta x_{i,j}$ using Eq.(7)
10: //Olfactory foraging phase
11: **For** i = 1,2,...,PS
12: Calculate $s(\Delta x_{i,j})$ using Eq.(6)
13: Using Eq.(5) to generate food source $x_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,n})$
14: **End For**
15: //Vision foraging phase
16: Calculate the fitness of all agents using Eq.(4)
17: **If** there is a better solution in population
18: Update swarm location Δ
19: **end If**
20: **For** i = (PS/2)+1 to PS
21: Update the population of i-th grasshopper using EPD approach
22: **End For**
23: **Until** Iter = Itermax

complexity of the proposed algorithm lies mainly in generating food source for each fruit fly. In the worst case, the computation complexity of the BIFFOA is simplified as O($t \times d \times n$).

Note that the computational complexity of the proposed BIFFOA_EPD is not significantly different from the BIFFOA. To re-initialize 50% of solutions, the additional complexity of O(n/2) is required, so the overall computational complexity of the proposed BIFFOA_EPD is O($t \times d \times n + n/2$).

However, when the proposed algorithm is applied to a real feature selection problem, it is hard to calculate its real run-time. Like other evolutionary algorithms, the proposed algorithm takes much time to calculate the fitness value of the individual. The time of getting a solution depends on the number of features, which is hard to predict. So, the run-time of the proposed algorithm depends on both the algorithm and the datasets [42].

IV. EXPERIMENT

A. EXPERIMENT DESIGN

In order to evaluate the performance of the proposed approaches, the experiments are performed on 25 datasets.

TABLE 1. Datasets.

No.	Dataset	No.of Features	No.of instance
1	breastercancer	9	699
2	BreastEW	30	596
3	Exactly	13	1000
4	Exactly2	13	1000
5	HeartEW	13	270
6	Lymphography	18	148
7	M-of-n	13	1000
8	penglungEW	325	73
9	SonarEW	60	208
10	SpectEW	22	267
11	CongressEW	16	435
12	IonosphereEW	34	351
13	KrvskpEW	36	3196
14	Tic-tac-toe	9	958
15	Vote	16	300
16	WaveformEW	40	5000
17	WineEW	13	178
18	Zoo	16	101
19	clean1	166	476
20	semeion	265	1593
21	Colon	2000	62
22	Leukemia	7129	72
23	Dermatology	34	366
24	Hepatitis	19	155
25	Lungcancer	56	32

Table 1 shows the data sets used, which are from the UCI data repository.

B. PARAMETER SETTINGS

We use KNN classifier based on Euclidean distance measurement (where k = 5). Different BIFFOA algorithms are adopted to find the optimal reduction with the minimum error. In each of the 30 runs, each dataset is randomly divided into two sets: 80% of the instances are used for training, and the remaining are used for testing. Therefore, the statistical measurements are collected based on the overall capabilities and final results throughout 30 independent runs. The dimensions of the tackled problems are equal to the number of features in the datasets. This partitioning was used in various previous works in the literature [46]–[48]. Note that we choose KNN because it is simple and cheap. Previous research [43] has shown that using a simple and relatively cheap classification algorithm in a wrapper approach can select a good feature subset for other complex learning/classification algorithms, which are computationally expensive but able to achieve better classification accuracy [44].

All experiments are carried out on a PC with Intel Core (TM) i5-5200uz CPU and 8.0GB RAM. All algorithms are tested using the MATLAB R2017a software. The maximum number of iterations is set to 100 and the number of search agents (N) is 24. The dimension of the algorithm is equal to the feature number of each data set. Then α and β parameters in the fitness function are set to 0.99 and 0.01, respectively.

TABLE 2. Average classification accuracy of proposed techniques.

Dataset	BIFFOA		BIFFOA_EPD		BIFFOA_EPD_CM		BIFFOA_EPD_RWS		BIFFOA_EPD_Tour	
	Acc	StdDev	Acc	StdDev	Acc	StdDev	Acc	StdDev	Acc	StdDev
breastercancer	0.9800	0.0080	0.9810	0.0090	0.9800	0.007	**0.9850**	0.0060	0.9820	0.0110
BreastEW	0.9763	0.0099	0.9763	0.0095	0.9763	0.0081	0.9768	0.0065	**0.9772**	0.0087
Exactly	**1.0000**	0.0000	**1.0000**	0.0000	0.9991	0.0039	0.9987	0.0059	0.9992	0.0078
Exactly2	0.7918	0.0169	0.7915	0.0177	0.7998	0.0508	0.8013	0.0497	**0.8020**	0.0498
HeartEW	0.8630	0.0303	0.8611	0.0303	0.8630	0.0326	0.8648	0.0324	**0.8648**	0.0307
Lymphography	0.9344	0.0339	0.9414	0.0322	0.9383	0.0270	0.9430	0.0306	**0.9430**	0.0287
M-of-n	**1.0000**	0.0000	**1.0000**	0.0000	0.9983	0.0075	**1.0000**	0.0000	0.9983	0.0075
penglungEW	0.9326	0.0530	0.9364	0.0480	0.9329	0.0530	0.9359	0.0459	**0.9395**	0.0551
SonarEW	0.9702	0.0327	0.9714	0.0333	**0.9726**	0.0292	**0.9726**	0.0292	0.9702	0.0298
SpectEW	0.8731	0.0308	0.8769	0.0331	0.8722	0.0334	0.8722	0.0288	0.8750	0.0344
CongressEW	0.9810	0.0107	0.9816	0.0108	**0.9816**	0.0101	0.9816	0.0120	0.9799	0.0111
IonosphereEW	0.9632	0.0195	0.9642	0.0196	0.9599	0.0150	0.9653	0.0212	**0.9653**	0.0169
KrvskpEW	0.9802	0.0050	0.9803	0.0048	0.9806	0.0043	0.9805	0.0046	**0.9811**	0.0048
Tic-tac-toe	0.8344	0.0157	0.8344	0.0157	0.8344	0.0157	0.8344	0.0157	**0.8425**	0.0395
Vote	0.9879	0.0087	0.9879	0.0087	**0.9891**	0.0079	0.9885	0.0091	0.9879	0.0087
WaveformEW	0.8067	0.0099	0.8072	0.0069	0.8062	0.0074	0.8077	0.0076	**0.8079**	0.0087
WineEW	**1.0000**	0.0000	**1.0000**	0.0000	**1.0000**	0.0000	**1.0000**	0.0000	**1.0000**	0.0000
Zoo	0.9976	0.0106	0.9976	0.0106	0.9976	0.0106	**1.0000**	0.0000	**1.0000**	0.0000
clean1	0.8510	0.0212	0.8510	0.0209	0.8526	0.0214	0.8510	0.0197	**0.8536**	0.0233
semeion	0.9904	0.0052	0.9900	0.0050	**0.9908**	0.0053	0.9906	0.0057	0.9904	0.0052
Colon	0.7769	0.1026	0.7846	0.0986	0.7885	0.1055	0.8807	0.0986	**0.8846**	0.0947
Leukemia	0.9833	0.0296	0.9867	0.0274	0.9833	0.0296	0.9833	0.0296	**0.9867**	0.0274
Dermatology	0.9950	0.0068	**0.9959**	0.0064	0.9953	0.0066	0.9953	0.0066	0.9953	0.0066
Hepatitis	0.9242	0.0409	0.9210	0.0398	0.9274	0.0417	0.9226	0.0383	**0.9274**	0.0390
Lungcancer	0.9694	0.0608	**0.9796**	0.0519	0.9694	0.0608	0.9694	0.0608	0.9694	0.0608

C. EVALUATION OF THE PROPOSED ALGORITHM

In this part, the classification accuracy (Acc), selected Attributes number (Atts), fitness value (Fitness) and CPU running time (Time) are the average results of 30 trials. The standard deviation of the running result (StdDev) also provides a metric for the algorithm. The five approaches are compared to evaluate the effect of using EPD (BIFFOA, BIFFOA_EPD, BIFFOA_EPD_CM, BIFFOA_EPD_RWS and BIFFOA_EPD_Tour). The experimental results are given in tables and the best results are represented in bold.

It can be seen in Table 2, the BIFFOA_EPD_Tour performs best among the five proposed algorithms for fifteen out of Twenty-five datasets. It outperforms BIFFOA over 18 datasets and the difference in classification accuracy between the BIFFOA and BIFFOA_EPD_Tour varies from 0.02% to 11%. Moreover, BIFFOA_EPD achieves superior Acc rates in tackling the Exactly, M-of-n, SpectEW, WineEW, Dermatology and Lungcancer especially in solving the Exactly and M-of-n datasets, the BIFFOA_EPD has attained the Acc of 100%. BIFFOA_EPD_CM and BIFFOA_EPD_RWS are superior to other methods on five data sets respectively in terms of the classification accuracy. In addition, BIFFOA and four hybrid algorithms are all 100% accurate in WineEW dataset. BIFFOA_EPD_RWS and BIFFOA_EPD_Tour are 100% accurate in Zoo dataset. This result proves that the EPD mechanism is helpful for the algorithm to find the optimal solution.

Compared with BIFFOA, the improvement of BIFFOA_EPD_Tour focuses on improving classification accuracy for optimization, so the improvement of Acc value is in line with the previous assumption. What is surprising is the improvement of the selected attributes value brought by BIFFOA_EPD_Tour. According to the selected attributes (Atts) in Table 3, it can be seen that the value of Atts of BIFFOA_EPD_Tour in 14 data sets is better than that of BIFFOA, and BIFFOA_EPD_Tour obtained the smallest number of feature subsets on 10 datasets. The reason for this is that the EPD mechanism makes the fitness function takes full account of the dimension reduction factor.

Inspecting the fitness value (Fitness) in Table 4, the best algorithm is the BIFFOA_EPD_Tour. It shows the lowest values for the objective function in tackling the 13 datasets. The BIFFOA_EPD_RWS has shown a relatively good performance in dealing with 5 datasets. The BIFFOA_EPD has provided a lower fitness for SpectEW, CongressEW, Dermatology, and Lungcancer. The hybrid BIFFOA algorithm

TABLE 3. Average selected attributes of proposed techniques.

Dataset	BIFFOA		BIFFOA_EPD		BIFFOA_EPD_CM		BIFFOA_EPD_RWS		BIFFOA_EPD_Tour	
	Atts	StdDev	Atts	StdDev	Atts	StdDev	Atts	StdDev	Atts	StdDev
breastercancer	**4.33**	1.184	4.83	1.116	4.8	1.105	4.5	1.214	4.86	1.15
BreastEW	**14.45**	2.01	15.65	1.98	15.70	3.10	14.75	2.43	15.40	2.56
Exactly	**6.10**	0.31	6.20	0.41	6.60	1.79	6.90	1.97	6.65	2.23
Exactly2	**8.05**	2.67	8.60	1.88	7.95	2.76	8.20	2.44	8.30	1.78
HeartEW	**5.00**	1.34	**5.00**	1.56	5.10	1.52	5.40	1.39	5.50	1.36
Lymphography	9.00	2.32	9.20	1.77	9.25	1.59	8.95	1.82	**8.70**	1.45
M-of-n	**6.10**	0.31	6.25	0.44	6.45	1.36	6.35	1.35	6.30	1.13
penglungEW	167.45	19.50	166.40	19.34	157.15	39.55	**154.75**	38.54	156.70	41.70
SonarEW	31.80	3.24	**31.00**	3.32	32.40	2.72	31.70	2.79	32.90	3.19
SpectEW	12.40	1.85	12.20	1.74	12.20	2.26	12.60	1.82	**11.40**	2.33
CongressEW	7.10	1.83	**6.30**	1.81	7.00	1.38	6.35	1.69	6.70	1.45
IonosphereEW	16.77	2.20	16.23	2.59	**15.08**	1.93	17.69	2.63	16.69	1.80
KrvskpEW	23.55	2.01	23.25	2.31	22.50	3.41	22.55	2.87	**21.95**	3.22
Tic-tac-toe	6.65	0.59	6.55	0.69	6.55	0.69	6.65	0.59	**6.45**	0.76
Vote	6.65	1.76	6.80	1.64	7.45	2.28	7.05	1.70	**6.50**	1.85
WaveformEW	16.15	2.13	14.80	1.64	14.65	1.69	15.80	1.61	15.50	2.28
WineEW	**4.30**	0.92	4.50	0.83	4.55	1.00	4.35	0.88	4.45	0.94
Zoo	4.85	1.42	4.90	1.59	4.60	1.14	4.65	1.73	**4.55**	1.28
clean1	98.40	4.82	99.35	9.56	99.40	5.57	97.50	5.00	**96.95**	5.31
semeion	158.85	9.02	159.95	6.81	160.40	8.65	160.95	7.88	**157.60**	10.97
Colon	1035.50	95.59	1042.50	103.49	1027.35	105.85	951.65	98.27	**954.45**	107.89
Leukemia	3665.60	303.45	3726.00	346.04	3611.30	247.85	3619.10	269.09	**3603.60**	247.68
Dermatology	17.35	2.72	**17.25**	2.83	17.35	1.87	17.70	2.92	17.85	2.81
Hepatitis	7.85	1.87	**7.65**	1.50	7.80	1.64	7.80	2.19	8.40	2.30
Lungcancer	23.79	2.01	24.86	3.39	**23.07**	2.27	23.50	2.47	24.00	2.66

TABLE 4. Average fitness results of proposed techniques.

Dataset	BIFFOA		BIFFOA_EPD		BIFFOA_EPD_CM		BIFFOA_EPD_RWS		BIFFOA_EPD_Tour	
	Fitness	StdDev	Fitness	StdDev	Fitness	StdDev	Fitness	StdDev	Fitness	StdDev
breastercancer	0.023	0.009	0.024	0.009	0.023	0.007	**0.0198**	0.006	0.0222	0.0111
BreastEW	0.0283	0.0097	0.0287	0.0093	0.0287	0.0079	0.0279	0.0063	**0.0277**	0.0087
Exactly	**0.0047**	0.0002	0.0048	0.0003	0.0056	0.0039	0.0063	0.0058	0.0065	0.0079
Exactly2	0.2124	0.0168	0.2130	0.0177	0.2044	0.0503	0.2031	0.0495	**0.2023**	0.0496
HeartEW	0.1395	0.0303	0.1413	0.0301	0.1396	0.0324	**0.1380**	0.0322	0.1381	0.0304
Lymphography	0.0696	0.0333	0.0631	0.0312	0.0662	0.0268	0.0630	0.0279	**0.0613**	0.0302
M-of-n	**0.0047**	0.0002	0.0048	0.0003	0.0065	0.0078	0.0048	0.0005	0.0064	0.0077
penglungEW	0.0719	0.0526	0.0680	0.0477	0.0715	0.0526	0.0684	0.0456	**0.0650**	0.0544
SonarEW	0.0348	0.0324	0.0335	0.0329	0.0325	0.0288	**0.0324**	0.0289	0.0349	0.0297
SpectEW	0.1312	0.0303	**0.1275**	0.0327	0.1320	0.0329	0.1322	0.0287	0.1289	0.0338
CongressEW	0.0232	0.0106	**0.0221**	0.0106	0.0226	0.0099	0.0222	0.0117	0.0241	0.0107
IonosphereEW	0.0414	0.0193	0.0402	0.0191	0.0441	0.0149	0.0395	0.0209	**0.0392**	0.0168
KrvskpEW	0.0258	0.0049	0.0259	0.0048	0.0254	0.0045	0.0256	0.0042	**0.0253**	0.0045
Tic-tac-toe	0.1714	0.0155	0.1712	0.0154	0.1712	0.0154	0.1714	0.0155	**0.1630**	0.0395
Vote	0.0161	0.0088	0.0162	0.0087	**0.0155**	0.0084	0.0158	0.0092	0.0160	0.0089
WaveformEW	0.1991	0.0096	0.1980	0.0071	0.1988	0.0072	0.1980	0.0074	**0.1976**	0.0084
WineEW	**0.0033**	0.0007	0.0035	0.0006	0.0035	0.0008	**0.0033**	0.0007	0.0034	0.0007
Zoo	0.0054	0.0109	0.0054	0.0109	0.0052	0.0108	0.0029	0.0011	**0.0028**	0.0008
clean1	0.1528	0.0209	0.1544	0.0203	0.1544	0.0211	0.1543	0.0196	**0.1517**	0.0229
semeion	0.0155	0.0052	0.0160	0.0050	**0.0152**	0.0052	0.0154	0.0055	0.0154	0.0053
Colon	0.2260	0.1016	0.2184	0.0976	0.2146	0.1044	0.1227	0.0977	**0.1190**	0.0937
Leukemia	0.0216	0.0294	0.0184	0.0271	0.0216	0.0294	0.0216	0.0294	**0.0183**	0.0270
Dermatology	0.0105	0.0066	**0.0091**	0.0064	0.0098	0.0065	0.0099	0.0065	0.0099	0.0064
Hepatitis	0.0792	0.0405	0.0823	0.0392	0.0760	0.0412	0.0808	0.0379	**0.0763**	0.0388
Lungcancer	0.0346	0.0602	**0.0246**	0.0513	0.0344	0.0602	0.0345	0.0601	0.0346	0.0601

TABLE 5. Average time (seconds) of proposed techniques.

Dataset	BIFFOA		BIFFOA_EPD		BIFFOA_EPD_CM		BIFFOA_EPD_RWS		BIFFOA_EPD_Tour	
	Time	StdDev	Time	StdDev	Time	StdDev	Time	StdDev	Time	StdDev
breastercancer	**22.852**	1.078	25.317	1.701	24.317	1.701	25.521	0.996	27.209	1.304
BreastEW	**19.925**	0.895	31.314	1.127	47.225	2.634	47.294	2.398	49.152	2.448
Exactly	**29.785**	0.590	29.798	0.525	30.458	0.718	30.615	0.698	30.765	0.717
Exactly2	**29.838**	0.576	30.145	0.639	30.781	0.583	30.770	0.470	30.673	0.497
HeartEW	**19.875**	0.762	21.925	0.839	23.142	1.319	23.234	3.652	26.404	0.828
Lymphography	**19.775**	0.702	22.048	0.815	24.326	8.924	25.205	0.777	25.148	3.157
M-of-n	23.463	1.102	**23.214**	0.475	23.678	0.980	23.701	0.779	23.927	1.065
penglungEW	27.326	0.398	27.364	0.379	**27.131**	1.387	27.194	1.307	27.152	1.306
SonarEW	**20.423**	0.799	24.782	2.326	24.787	1.817	25.535	0.904	26.045	4.599
SpectEW	**20.619**	0.881	29.891	2.892	22.976	1.367	26.465	0.454	23.964	1.209
CongressEW	**40.612**	1.815	43.900	3.609	50.646	1.320	46.302	1.265	47.839	4.897
IonosphereEW	**25.984**	1.492	34.306	1.803	26.093	2.648	26.261	0.989	30.320	1.781
KrvskpEW	34.754	0.356	34.516	0.332	33.945	2.584	33.877	2.805	33.858	2.834
Tic-tac-toe	23.152	1.302	41.540	0.866	50.807	0.886	27.125	1.223	25.240	0.860
Vote	21.867	0.568	22.610	0.646	49.775	0.696	27.660	1.599	**33.213**	1.191
WaveformEW	**38.256**	0.579	40.726	0.832	42.346	1.086	47.889	1.401	43.213	1.908
WineEW	**22.890**	0.678	43.065	0.325	32.014	1.311	27.552	1.201	24.900	1.228
Zoo	**18.086**	1.277	20.704	0.167	24.858	0.707	26.618	2.184	24.898	0.818
clean1	**28.300**	0.601	31.763	3.504	37.220	1.639	37.716	1.376	39.881	2.076
semeion	**43.262**	1.039	46.532	1.556	54.135	1.346	49.383	0.814	46.182	0.480
Colon	**44.263**	1.025	60.977	0.432	64.133	0.571	68.086	9.392	64.875	0.885
Leukemia	**148.220**	3.820	151.224	2.602	168.081	18.587	193.971	14.415	155.384	12.167
Dermatology	29.065	1.171	28.878	1.203	**28.809**	1.701	29.125	0.976	29.405	1.755
Hepatitis	**25.538**	1.553	25.731	0.953	26.040	1.521	25.951	0.949	26.063	1.542
Lungcancer	23.509	3.229	23.738	3.568	23.772	3.490	**23.379**	3.169	23.570	3.575

has some improvements over the original BIFFOA algorithm in terms of the fitness value, which shows that the hybrid BIFFOA algorithm is superior to the original BIFFOA algorithm in the optimization.

Table 5 records the running time of the five algorithms. It can be seen that the original BIFFOA algorithm is the fastest. The difference between the BIFFOA hybrid algorithm and the BIFFOA is only that the n/2 solutions need to be re-initialized, and therefore the extra time overhead is expected to come from a function evaluation of n/2 solutions.

The convergence curves for the proposed algorithms on all 25 datasets are demonstrated in Figure 6. As can be seen from Figure 5, BIFFOA_EPD_Tour has exposed the best curves in tackling in 13 datasets. Compared with the other three algorithms, BIFFOA_EPD_Tour can rapidly converge to the optimal solution on most datasets. The BIFFOA_EPD_RWS shows a faster tendency to converge than others in processing in treating Breast cancer, HeartEW, SonarEW, and WineEW datasets. It appears that the versions of TS and RWS can better rearrange the latter half of fruit flies compared to BIFFOA_EPD and BIFFOA_EPD_CM. BIFFOA_EPD and BIFFOA_EPD_RWS are also superior to basic BIFFOA in optimization. It supports that the EPD schemes have balanced the exploration and exploitation traits.

Based on the above experimental results, it can be seen that the BIFFOA_EPD_Tour algorithm with TS operator can improve the quality of the solution, which promotes us to keep the proposed method BIFFOA_EPD_Tour.

D. COMPARISON WITH OTHER NATURAL HEURISTIC ALGORITHMS

In this section, the hybrid algorithm BIFFOA_EPD_Tour is compared with the natural heuristic algorithm bGWO1, BGSA, BBA. In order to ensure the accuracy of the experimental results, some experimental results published in literature [3], [17] are used. The specific information of other comparison algorithms are given in Table 6.

TABLE 6. The parameter settings.

Algorithm	Parameter	Value
BGSA	G0	100
	α	20
BA	Qmin Frequency minimum	0
	Qmax Frequency maximum	2
GWO	α	[2 0]
BGOA_EPD_Tour	cMax	1
	cMin	0.00001
	MaxIter	100
	α	0.99
	β	0.01

The experimental results are given in Table 7, Table 8 and Table 9. The classification accuracy (Acc), the number of selected Attributes (Atts), the value of fitness function (Fitness) and the corresponding mean standard deviation (StdDev) of several algorithms are recorded respectively.

It is not difficult to see from Table 7. The hybrid algorithm BIFFOA_EPD_Tour obtains the best classification accuracy in breastercancer, BreastEW, HeartEW

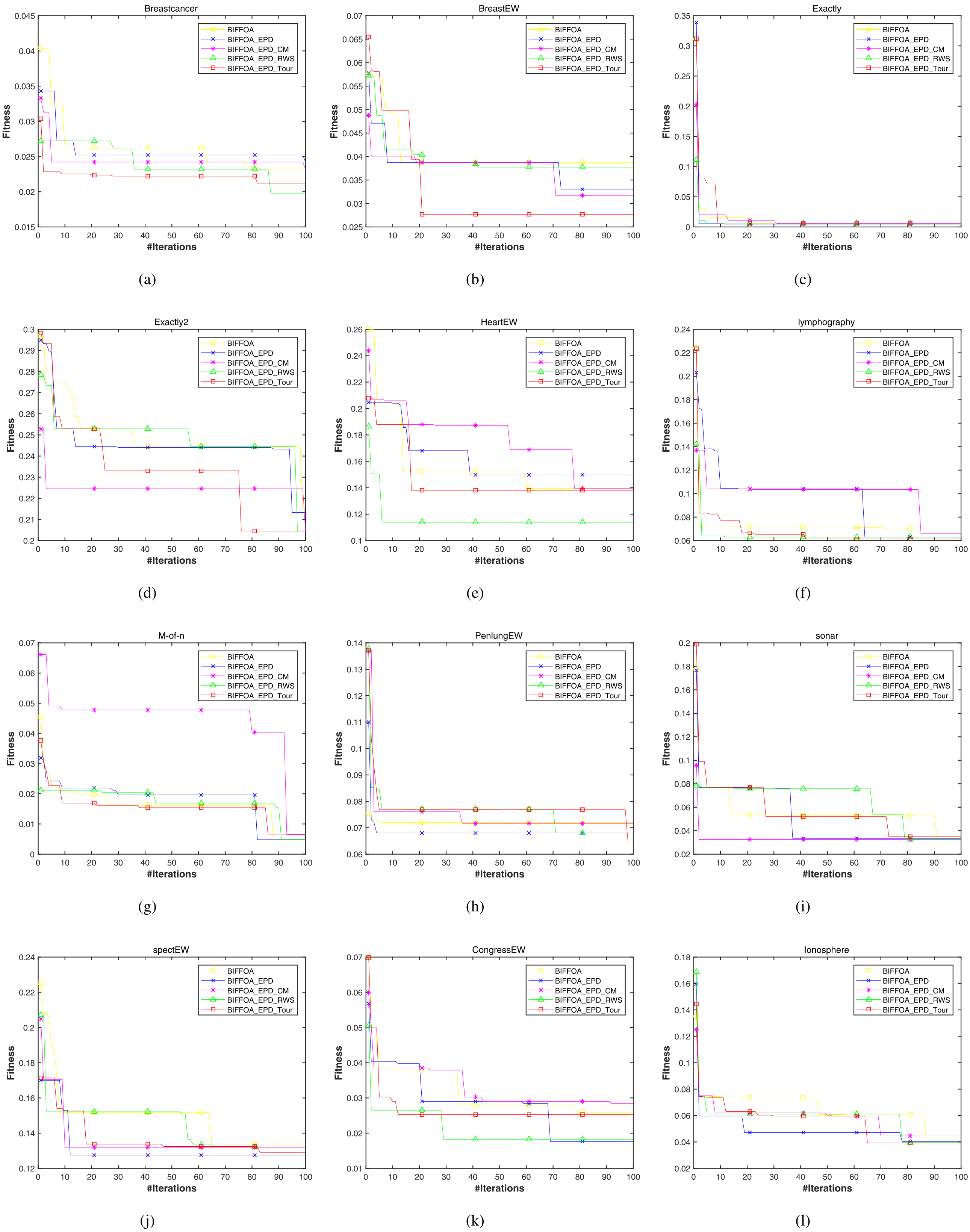


FIGURE 6. Typical Convergence curves of the proposed approaches.

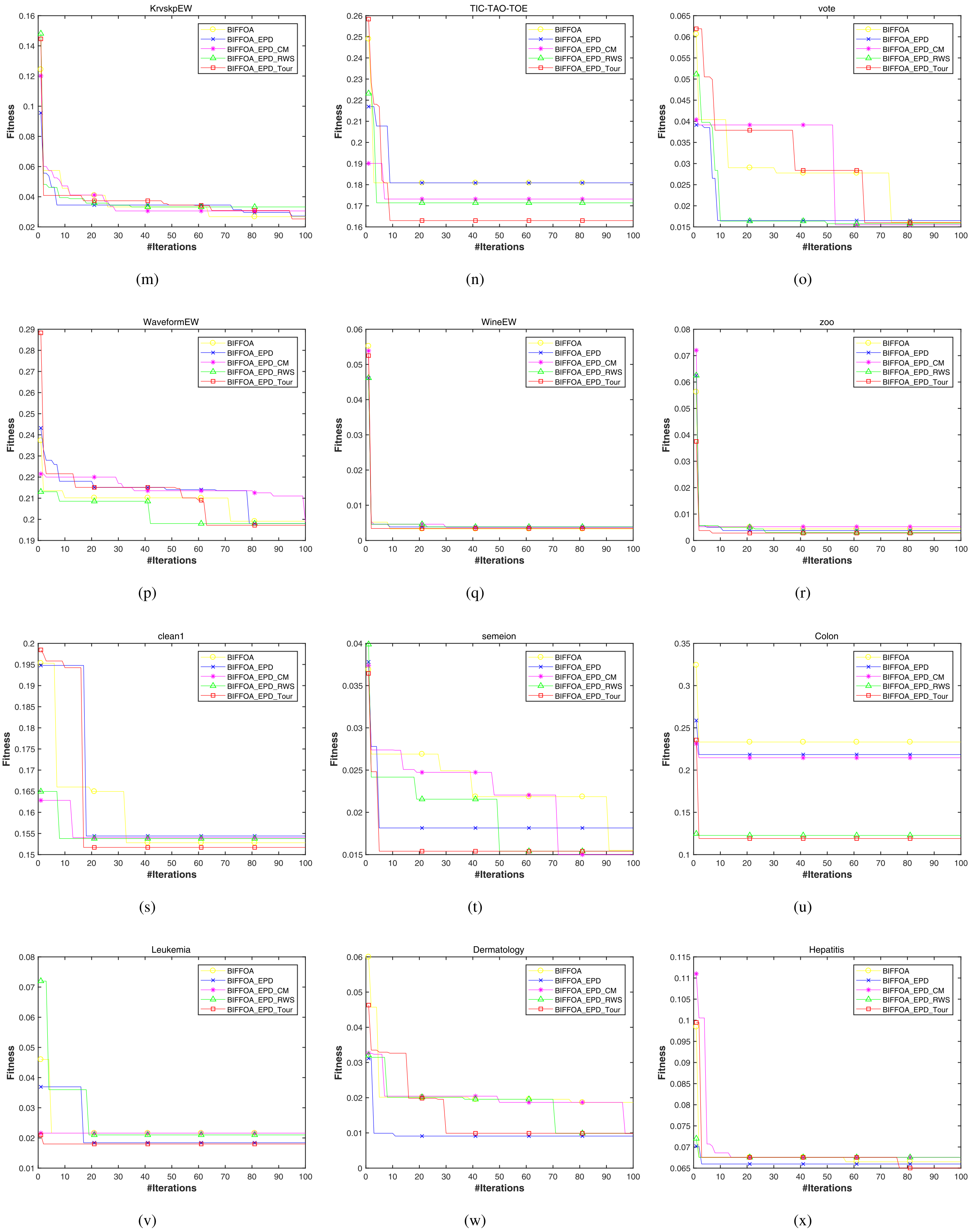


FIGURE 6. ***(Continued.)*** **Typical Convergence curves of the proposed approaches.**

TABLE 7. Average classification accuracy results obtained by different algorithms.

Dataset	BIFFOA_EPD_Tour		bGWO [26]		BGSA [18]		BBA [22]	
	Acc	StdDev	Acc	StdDev	Acc	StdDev	Acc	StdDev
Breastcancer	**0.982**	0.011	0.968	0.002	0.957	0.004	0.937	0.031
BreastEW	**0.977**	0.009	0.954	0.007	0.942	0.006	0.931	0.014
Exactly	**0.998**	0.008	0.809	0.076	0.697	0.06	0.61	0.065
Exactly2	**0.802**	0.050	0.743	0.017	0.706	0.023	0.628	0.057
HeartEW	**0.865**	0.031	0.792	0.017	0.777	0.022	0.754	0.033
Lymphography	**0.943**	0.029	0.813	0.028	0.781	0.022	0.701	0.069
M-of-n	**0.998**	0.008	0.894	0.041	0.835	0.063	0.722	0.08
penglungEW	**0.940**	0.055	0.85	0.014	0.919	0	0.795	0.029
SonarEW	**0.970**	0.030	0.836	0.016	0.888	0.015	0.844	0.036
SpectEW	**0.875**	0.034	0.81	0.014	0.783	0.024	0.8	0.027
CongressEW	**0.980**	0.011	0.948	0.011	0.951	0.008	0.872	0.075
IonosphereEW	**0.965**	0.017	0.885	0.009	0.881	0.01	0.877	0.019
KrvskpEW	**0.981**	0.005	0.934	0.015	0.908	0.048	0.816	0.081
Tic-tac-toe	**0.843**	0.040	0.754	0.032	0.753	0.024	0.665	0.063
Vote	**0.988**	0.009	0.944	0.01	0.931	0.011	0.851	0.096
WaveformEW	**0.808**	0.009	0.723	0.007	0.695	0.014	0.669	0.033
WineEW	**1.000**	0.000	0.96	0.012	0.951	0.015	0.919	0.052
Zoo	**1.000**	0.000	0.975	0.009	0.939	0.008	0.874	0.095
clean1	0.854	0.023	**0.908**	0.006	0.898	0.011	0.826	0.021
semeion	**0.990**	0.005	0.972	0.003	0.971	0.002	0.962	0.006
Colon	**0.885**	0.095	0.661	0.022	0.766	0.015	0.682	0.038
Leukemia	**0.987**	0.027	0.884	0.016	0.844	0.014	0.877	0.029
Dermatology	**0.995**	0.007	0.9688	0.0138	0.9905	0.0091	0.8608	0.1150
Hepatitis	0.927	0.039	0.8246	0.0288	**0.9387**	0.0355	0.8000	0.0787
Lungcancer	0.969	0.061	0.8312	0.0452	**0.9982**	0.0032	0.7500	0.1368

TABLE 8. Average number of selected attributes obtained by different algorithms.

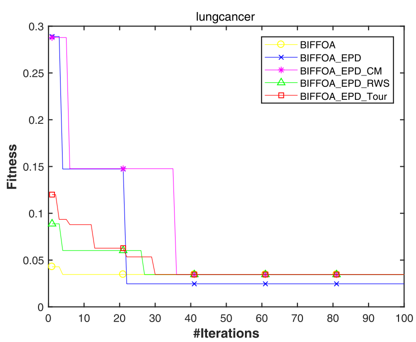
Dataset	BIFFOA_EPD_Tour		bGWO [26]		BGSA [18]		BBA [22]	
	Att	StdDev	Att	StdDev	Att	StdDev	Att	StdDev
Breastcancer	4.86	1.15	7.10	1.45	6.07	1.14	**3.67**	1.37
BreastEW	15.40	2.56	19.00	4.31	16.57	2.98	**12.40**	2.76
Exactly	6.65	2.23	10.23	1.65	8.73	1.05	**5.73**	1.89
Exactly2	8.30	1.78	7.33	4.16	**5.10**	2.11	6.07	2.33
HeartEW	**5.50**	1.36	8.17	2.00	6.83	1.32	5.90	1.65
Lymphography	**8.70**	1.45	11.10	1.97	9.17	1.90	7.80	2.20
M-of-n	6.30	1.13	9.63	0.96	8.47	1.43	**6.17**	2.09
penglungEW	156.70	41.70	166.33	28.23	157.17	7.73	**126.17**	15.60
SonarEW	32.90	3.19	36.23	8.61	30.03	3.70	**24.70**	5.38
SpectEW	11.40	2.33	12.63	2.44	9.53	2.30	**7.97**	2.28
CongressEW	6.70	1.45	7.30	2.14	6.77	2.40	**6.23**	2.06
IonosphereEW	16.69	1.80	19.23	5.02	15.40	2.51	**13.40**	2.59
KrvskpEW	21.95	3.22	27.37	3.39	19.97	2.13	**15.00**	2.85
Tic-tac-toe	6.45	0.76	6.70	1.34	5.87	1.14	**4.70**	1.49
Vote	6.50	1.85	7.40	2.22	8.17	1.82	**6.13**	2.18
WaveformEW	**15.50**	2.28	31.97	4.61	19.90	2.92	16.67	3.30
WineEW	**4.45**	0.94	8.60	1.75	7.37	1.10	6.07	1.74
Zoo	**4.55**	1.28	10.37	2.48	8.17	1.18	6.57	2.50
clean1	96.95	5.31	121.27	20.69	83.70	5.42	**64.77**	10.02
semeion	157.60	10.97	200.10	31.02	133.53	7.42	**107.03**	10.95
Colon	954.45	107.89	1042.10	126.72	995.83	20.02	**827.50**	55.37
Leukemia	3603.60	247.68	3663.77	294.87	3555.13	39.71	**2860.00**	247.64
Dermatology	17.85	2.81	23.45	2.28	12.80	1.23	**11.50**	2.92
Hepatitis	8.40	2.30	7.70	2.49	**5.20**	2.10	6.10	1.66
Lungcancer	24.00	2.66	**8.00**	2.91	13.00	2.94	17.50	2.08

and other 22 data sets. We can remark that the performance of the BIFFOA_EPD_Tour overcomes the obtained results for bGWO1, BGSA, and BBA, which proves its future performance on the unseen data, and hence it can be used as a candidate for feature selection.

In terms of the number of selected features in Table 8, Atts, BIFFOA_EPD_Tour obtains the smallest value in HeartEW,

TABLE 9. Average Fitness results obtained by different algorithms.

Dataset	BIFFOA_EPD_Tour		bGWO [26]		BGSA [18]		BBA [22]	
	Fitness	StdDev	Fitness	StdDev	Fitness	StdDev	Fitness	StdDev
Breastcancer	**0.023**	0.009	0.039	0.003	0.049	0.003	0.044	0.005
BreastEW	**0.028**	0.010	0.051	0.007	0.063	0.006	0.056	0.006
Exactly	**0.005**	0.000	0.197	0.077	0.307	0.059	0.323	0.074
Exactly2	**0.212**	0.017	0.26	0.019	0.295	0.024	0.326	0.017
HeartEW	**0.140**	0.030	0.213	0.017	0.226	0.021	0.208	0.015
Lymphography	**0.070**	0.033	0.191	0.028	0.222	0.022	0.226	0.024
M-of-n	**0.005**	0.000	0.112	0.041	0.17	0.063	0.171	0.056
penglungEW	**0.072**	0.053	0.154	0.013	0.085	0	0.168	0.017
SonarEW	**0.035**	0.032	0.169	0.016	0.116	0.015	0.11	0.021
SpectEW	**0.131**	0.030	0.194	0.014	0.22	0.024	0.172	0.012
CongressEW	**0.023**	0.011	0.056	0.011	0.053	0.008	0.064	0.015
IonosphereEW	**0.041**	0.019	0.12	0.009	0.122	0.01	0.108	0.012
KrvskpEW	**0.026**	0.005	0.073	0.015	0.097	0.047	0.117	0.047
Tic-tac-toe	**0.171**	0.016	0.251	0.032	0.251	0.024	0.257	0.024
Vote	**0.016**	0.009	0.06	0.01	0.073	0.011	0.071	0.013
WaveformEW	**0.199**	0.010	0.283	0.007	0.307	0.014	0.304	0.014
WineEW	**0.003**	0.001	0.047	0.012	0.054	0.015	0.036	0.013
Zoo	**0.005**	0.011	0.032	0.009	0.065	0.008	0.042	0.015
clean1	0.153	0.021	**0.099**	0.006	0.106	0.01	0.156	0.013
semeion	**0.016**	0.005	0.036	0.003	0.034	0.002	0.033	0.003
Colon	**0.226**	0.102	0.341	0.022	0.237	0.014	0.279	0.035
Leukemia	**0.022**	0.029	0.12	0.016	0.16	0.013	0.085	0.023
Dermatology	**0.011**	0.007	0.038	0.013	0.013	0.009	0.046	0.017
Hepatitis	0.079	0.041	0.178	0.028	**0.063**	0.035	0.115	0.044
Lungcancer	0.035	0.060	0.171	0.044	**0.002**	0.001	0.038	0.071



(y)

FIGURE 6. *(Continued.)* Typical Convergence curves of the proposed approaches.

Lymphography, WaveformEW, WineEW and Zoo. For the rest datasets, Although the Atts values of other comparison algorithms are the smallest, they have a great degraded on the classification accuracy. The number of attributes selected depends on the dataset being processed and the algorithm itself. For the 25 datasets used in the experiment, BIFFOA_EPD_Tour achieved a maximum compression ratio of 72% on the dataset Zoo. The minimum compression rate on the Tic-tac-toe dataset is 29%, because the Tic-tac-toe attribute represents nine positions in the chessboard, and too little information on the chessboard cannot determine the final win or loss. To ensure classification accuracy, datasets can no longer be compressed.

The results in Table 9 record the fitness values of four algorithms. The fitness values comprehensively consider the classification accuracy and the number of selected features. It can be seen that the proposed BIFFOA_EPD_Tour has the best performance in the obtained fitness value in Table 9, which proves that the BIFFOA_EPD_Tour is better than other methods in the ability of adaptive search feature space. Figure 7 shows the convergence curves of several algorithms. For the feature selection problem, the premature convergence problem cannot be ignored. Although the BIFFOA_EPD_Tour curve also has a certain degree of premature convergence problem, it is still superior to the curve of other competitors. The BIFFOA_EPD_Tour algorithm has obtained the optimal curve on 22 data sets. Despite it has not obtained the optimal solution on the clean1, Hepatitis and lungcancer datasets, it shows a good trend of exploration and development.

E. COMPARISON WITH OTHER ALGORITHMS REPORTED IN PREVIOUS LITERATURE

In this part, the classification accuracy of the proposed BIFFOA_EPD_Tour is compared to the reported results for 25 datasets. Table 10 reveals the comparative classification rates of different approaches. The average classification rates of the BIFFOA_EPD_Tour is compared here to the reported

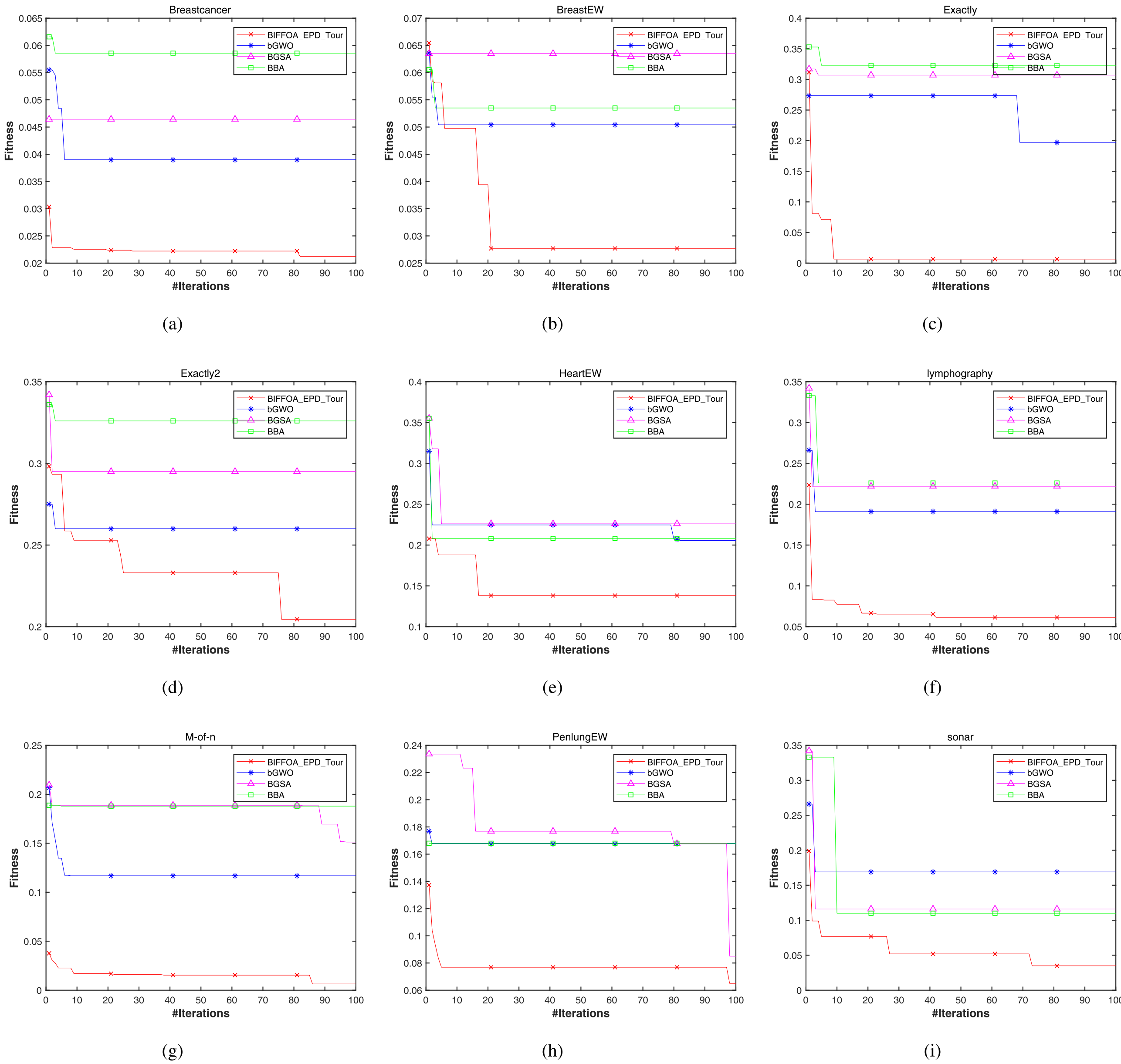


FIGURE 7. **Typical Convergence curves of BIFFOA_EPD_Tour and other state-of-art methods.**

performances of the GA, Spectrum and PSO algorithms in [34]. In addition, the results of the BIFFOA_EPD_Tour approach is also compared to the results of the bGWO1, bGWO2 techniques reported in [26]. Note that the results of the rest methods are from [3] and [17].

By comparing the results in Table 10, it can be seen that the accuracies of the BIFFOA_EPD_Tour proposed in this study is superior to those obtained from the past works on 84% of the datasets. It shows a substantial advantage over the BSSA, BGOA and FCBF algorithms on the 24 datasets. The results of the BIFFOA_EPD_Tour are better than those of GA, PSO, bGWO1, CFS, IG, Spectrum and F-Score for all datasets used in the experiment. The BIFFOA_EPD_Tour technique can realize enhanced classification rates compared to the bGWO2 on around 88% of the datasets.

The performance of the algorithm is better than the most advanced methods in most selected data sets. The main reason for the good performance of the algorithm is the integration of operators in the algorithm. On the one hand, the IFFOA algorithm itself has good performance and can effectively map continuous values to binary values by using the sigmod function. Note that this does not mean that the proposed BIFFOA algorithm is and will be the best choice to deal with all problems. On the other hand, EPD mechanisms can effectively promote the movement of fruit flies to promising areas.

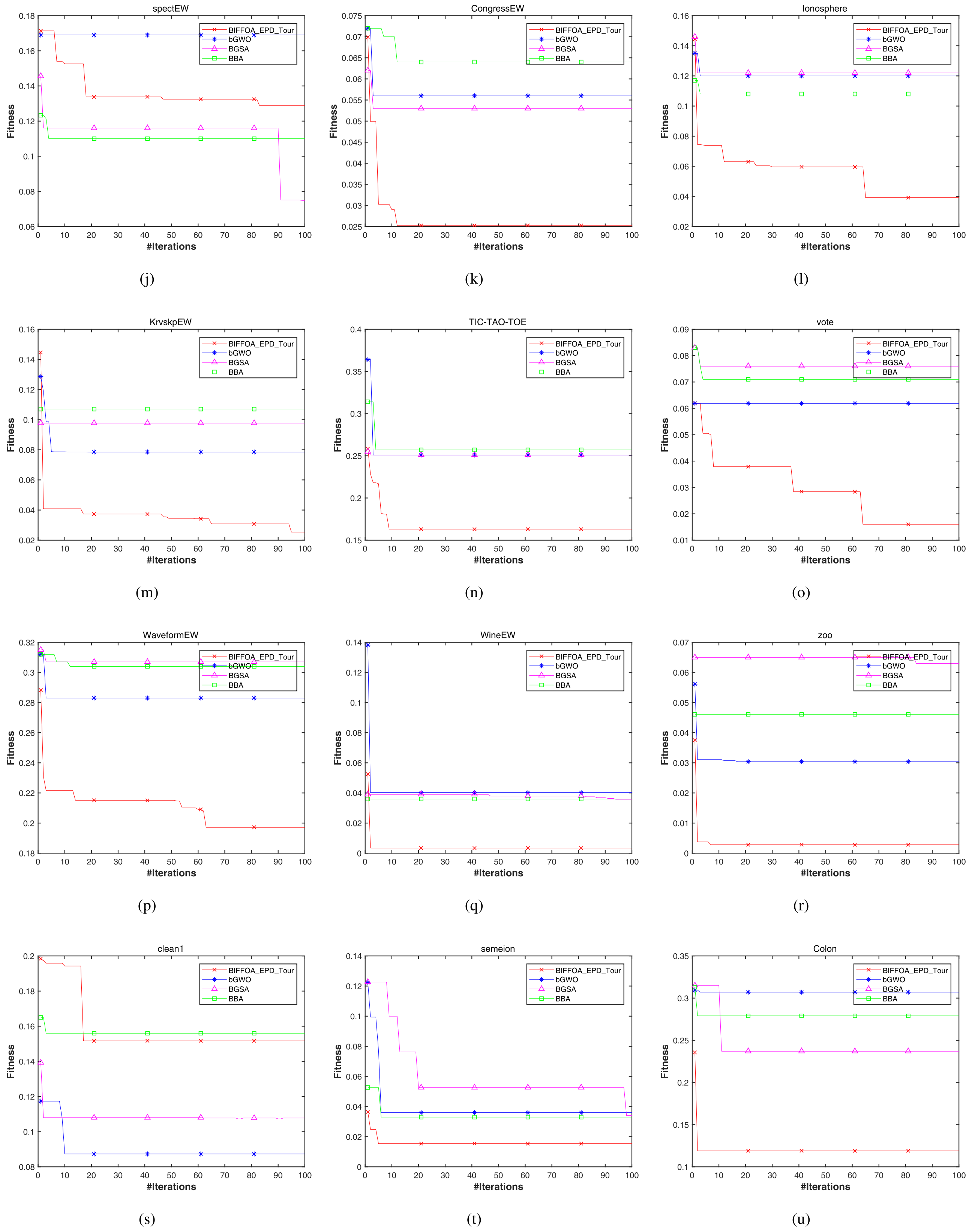
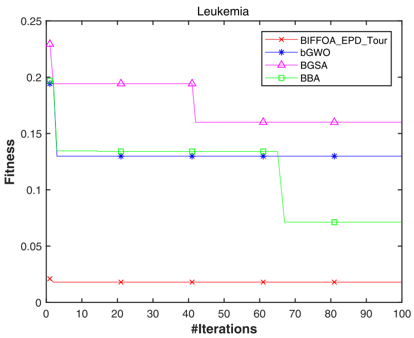
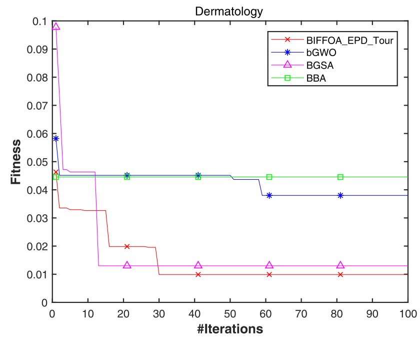


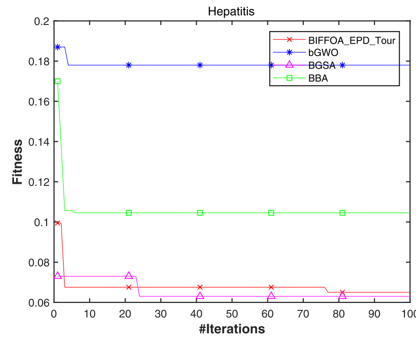
FIGURE 7. ***(Continued.)*** **Typical Convergence curves of BIFFOA_EPD_Tour and other state-of-art methods.**



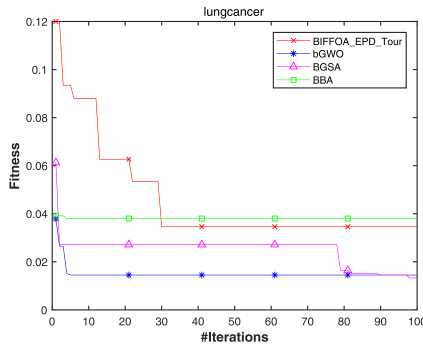
(v) h



(w)



(x)



(y)

FIGURE 7. ***(Continued.)*** **Typical Convergence curves of BIFFOA_EPD_Tour and other state-of-art methods.**

TABLE 10. **Classification accuracies of the BIFFOA_EPD_Tour versus other meta-heuristics.**

Dataset	BIFFOA_EPD_Tour	BSSA	BGOA	GA	PSO	bGWO1	bGWO2	CFS	FCBF	F-Score	IG	Spectrum
Breastcancer	0.982	0.977	0.980	0.957	0.949	0.976	0.975	0.957	**0.986**	0.979	0.957	0.957
BreastEW	**0.977**	0.948	0.947	0.923	0.933	0.924	0.935	0.825	0.798	0.930	0.930	0.772
Exactly	**0.999**	0.980	**0.999**	0.822	0.973	0.708	0.776	0.670	0.440	0.600	0.615	0.575
Exactly2	**0.802**	0.758	0.780	0.677	0.666	0.745	0.750	0.705	0.545	0.680	0.620	0.660
HeartEW	**0.865**	0.861	0.833	0.732	0.745	0.776	0.776	0.648	0.648	0.759	0.759	0.796
Lymphography	**0.943**	0.890	0.868	0.758	0.759	0.744	0.700	0.500	0.567	0.667	0.667	0.767
M-of-n	**0.998**	0.992	1.000	0.916	0.996	0.908	0.963	0.785	0.815	0.815	0.815	0.580
penglungEW	**0.940**	0.878	0.927	0.672	0.879	0.600	0.584	0.600	0.667	0.800	0.667	0.400
SonarEW	**0.970**	0.937	0.912	0.833	0.804	0.731	0.729	0.310	0.214	0.048	0.191	0.048
SpectEW	0.875	0.836	0.826	0.756	0.738	0.820	0.822	0.736	0.774	0.793	0.793	0.736
CongressEW	**0.980**	0.963	0.964	0.898	0.937	0.935	0.938	0.793	0.793	0.908	0.828	0.828
IonosphereEW	**0.965**	0.918	0.899	0.863	0.876	0.807	0.834	0.857	0.857	0.729	0.800	0.829
KrvskpEW	**0.981**	0.964	0.968	0.940	0.949	0.944	0.956	0.768	0.934	0.959	0.934	0.377
Tic-tac-toe	**0.843**	0.821	0.808	0.764	0.750	0.728	0.727	0.000	0.000	0.010	0.010	0.167
Vote	**0.988**	0.951	0.966	0.808	0.888	0.912	0.920	0.950	0.950	0.933	0.967	0.850
WaveformEW	**0.808**	0.734	0.737	0.712	0.732	0.786	0.789	0.620	0.710	0.662	0.662	0.292
WineEW	**1.000**	0.993	0.989	0.947	0.937	0.930	0.920	0.778	0.889	0.861	0.889	0.889
Zoo	**1.000**	1.000	0.993	0.946	0.963	0.879	0.879	0.800	0.900	0.650	0.850	0.600
clean1	0.854	0.880	0.863	0.862	0.845	0.949	**0.975**	0.716	0.642	0.632	0.547	0.611
semeion	**0.990**	0.980	0.976	0.963	0.967	0.984	**0.990**	0.875	0.875	0.875	0.868	0.875
Colon	**0.885**	0.686	0.870	0.682	0.624	0.850	0.880	0.750	0.667	0.667	0.667	0.500
Leukemia	0.987	0.989	0.931	0.705	0.862	0.983	**0.993**	0.929	0.857	0.980	0.980	0.357
Dermatology	**0.995**	0.992	0.935	0.863	0.876	0.994	0.990	-	-	-	-	-
Hepatitis	**0.927**	0.920	0.902	0.876	0.923	0.897	0.904	-	-	-	-	-
Lungcancer	**0.969**	0.962	0.957	0.923	0.935	0.886	0.968	-	-	-	-	-

V. CONCLUSION

In this paper, based on the improved fruit fly optimization algorithm, a binary IFFOA (BIFFOA) algorithm is proposed to deal with feature selection problems. Four mechanisms based on EPD with new mutation operator were used to improve the exploration and development capabilities of the BIFFOA, and the diversity of the population was improved. Although, in contrast to the feature selection algorithms that have been proposed in recent years, the improved hybrid algorithm, BIFFOA_EPD_Tour has a specific advantage in dealing with feature selection problems. However, the algorithm still has some defects and deficiencies in dimension reduction, so how to improve the search strategy of fruit fly algorithm so that it can guarantee a high classification accuracy and reduce the number of selected features at the same time is the focus of our further research. In future work, BIFFOA_EPD_Tour will be used with more classifiers like SVM, Artificial Neural Networks (ANN) to verify and extend this approach.

REFERENCES

[1] G. I. Sayed, G. Khoriba, and M. H. Haggag, "A novel chaotic salp swarm algorithm for global optimization and feature selection," *Appl. Intell.*, vol. 48, no. 10, pp. 3462–3481, Oct. 2018.

[2] C. Wang, Q. Hu, X. Wang, D. Chen, Y. Qian, and Z. Dong, "Feature selection based on neighborhood discrimination index," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 7, pp. 2986–2999, Jul. 2018.

[3] H. Faris *et al.*, "An efficient binary Salp swarm algorithm with crossover scheme for feature selection problems," *Knowl.-Based Syst.*, vol. 154, pp. 43–67, Aug. 2018.

[4] H. Liu and H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*. Boston, MA, USA: Kluwer, 1998.

[5] Z. Zhu, Y.-S. Ong, and M. Dash, "Wrapper–filter feature selection algorithm using a memetic framework," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 1, pp. 70–76, Feb. 2007.

[6] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Jan. 2003.

[7] S. Arora and P. Anand, "Binary butterfly optimization approaches for feature selection," *Expert Syst. Appl.*, vol. 116, pp. 147–160, Feb. 2018.

[8] Y. Xinshe, *Nature-Inspired Metaheuristic Algorithms*. Cambridge, U.K.: Univ. Cambridge, 2008.

[9] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 4, Nov./Dec. 2002, pp. 1942–1948.

[10] K. Dervis and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *J. Global Optim.*, vol. 39, no. 3, pp. 459–471, Nov. 2007.

[11] A. Colorni, M. Dorigo, and M. Maniezzo, "Distributed optimization by ant colonies," in *Proc. 1st Eur. Conf. Artif. Life*, 1991, pp. 134–176.

[12] M. Mafarja and S. Mirjalili, "Hybrid whale optimization algorithm with simulated annealing for feature selection," *Neurocomputing*, vol. 260, pp. 302–312, Oct. 2017.

[13] J.-Q. Li, Q.-K. Pan, K. Mao, and P. N. Suganthan, "Solving the steelmaking casting problem using an effective fruit fly optimisation algorithm," *Knowl.-Based Syst.*, vol. 72, pp. 28–36, Dec. 2014.

[14] H. Faris, I. Aljarah, M. A. Al-Betar, and S. Mirjalili, "Grey wolf optimizer: A review of recent variants and applications," *Neural Comput. Appl.*, vol. 30, no. 2, pp. 413–435, 2018.

[15] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2016.

[16] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimization algorithm: Theory and application," *Adv. Eng. Softw.*, vol. 105, pp. 30–47, Mar. 2017.

[17] M. Mafarja *et al.*, "Evolutionary population dynamics and grasshopper optimization approaches for feature selection problems," *Knowl.-Based Syst.*, vol. 145, pp. 25–45, Apr. 2018.

[18] E. Rashedi, H. Nezamabadi-Pourand, and S. Saryazdi, "BGSA: Binary gravitational search algorithm," *Natural Comput.*, vol. 9, no. 3, pp. 727–745, Sep. 2010.

[19] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp swarm algorithm: A bio-inspired optimizer for engineering design problems," *Adv. Eng. Softw.*, vol. 114, pp. 163–191, Dec. 2017.

[20] Y. Zhang, D. Gong, Y. Hu, and W. Zhang, "Feature selection algorithm based on bare bones particle swarm optimization," *Neurocomputing*, vol. 148, pp. 150–157, Jan. 2015.

[21] C.-H. Lin, H.-Y. Chen, and Y.-S. Wu, "Study of image retrieval and classification based on adaptive features using genetic algorithm feature selection," *Expert Syst. Appl.*, vol. 41, no. 15, pp. 6611–6621, Nov. 2014.

[22] S. Mirjalili, S. M. Mirjalili, and X.-S. Yang, "Binary bat algorithm," *Neural Comput. Appl.*, vol. 25, no. 3, pp. 663–681, Sep. 2014.

[23] A. E. Hegazy and M. A. Makhlouf, and G. S. El-Tawel, "Improved salp swarm algorithm for feature selection," *J. King Saud Univ.-Comput. Inf. Sci.*, to be published.

[24] J. P. Papa *et al.*, "Feature selection through gravitational search algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 2052–2055.

[25] E. Rashedi and H. Nezamabadi-Pour, "Feature subset selection using improved binary gravitational search algorithm," *J. Intell. Fuzzy Syst.*, vol. 26, no. 3, pp. 1211–1221, May 2014. doi: 10.3233/IFS-130807.

[26] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary grey wolf optimization approaches for feature selection," *Neurocomputing*, vol. 172, pp. 371–381, Jan. 2016.

[27] S. Mirjalili and A. Lewis, "S-shaped versus V-shaped transfer functions for binary particle swarm optimization," *Swarm Evol. Comput.*, vol. 9, pp. 1–14, Apr. 2013.

[28] L. Wang, X.-L. Zheng, and S.-Y. Wang, "A novel binary fruit fly optimization algorithm for solving the multidimensional knapsack problem," *Knowl.-Based Syst.*, vol. 48, no. 2, pp. 17–23, Aug. 2013.

[29] H. Qian, Q. Zhang, D. Lei, and Z. Pan, "A cooperated fruit fly optimization algorithm for Knapsack problem," in *Proc. Chin. Automat. Congr. (CAC)*, Oct. 2017, pp. 591–595.

[30] X.-L. Zheng, L. Wang, and S.-Y. Wang, "A novel fruit fly optimization algorithm for the semiconductor final testing scheduling problem," *Knowl.-Based Syst.*, vol. 57, pp. 95–103, Feb. 2014.

[31] L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proc. 20th Int. Conf. Mach. Learn. (ICML)*, 2003, pp. 856–863.

[32] Q.-K. Pan, L. Gao, X.-Y. Li, and K.-Z. Gao, "Effective metaheuristics for scheduling a hybrid flowshop with sequence-dependent setup times," *Appl. Math. Comput.*, vol. 303, pp. 89–112, Jun. 2017.

[33] X.-L. Zheng and L. Wang, "A collaborative multiobjective fruit fly optimization algorithm for the resource constrained unrelated parallel machine green scheduling problem," *IEEE. Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 5, pp. 790–800, May 2018.

[34] N. Dongxiao, M. Tiannan, and L. Bingyi, "Power load forecasting by wavelet least squares support vector machine with improved fruit fly optimization algorithm," *J. Combinat. Optim.*, vol. 33, no. 3, pp. 1122–1143, Apr. 2017.

[35] B. Crawford *et al.*, "A binary fruit fly optimization algorithm to solve the set covering problem," in *Proc. Int. Conf. Comput. Sci. Appl.* Cham, Switzerland: Springer, 2015.

[36] S. Juan, "A hybrid fly fruit algorithm for PID control parameters optimization," *Transducer Microsyst. Technol.*, 2015. doi: 10.13873/J.1000-9787(2015)06-0137-04.

[37] S. M. Mousavi, N. Alikar, and S. T. A. Niaki, "An improved fruit fly optimization algorithm to solve the homogeneous fuzzy series–parallel redundancy allocation problem under discount strategies," *Soft Comput.*, vol. 20, no. 6, pp. 2281–2307, Jun. 2016.

[38] Q.-K. Pan, H.-Y. Sang, J.-H. Duan, and L. Gao, "An improved fruit fly optimization algorithm for continuous function optimization problems," *Knowl.-Based Syst.*, vol. 62, pp. 69–83, May 2014.

[39] W.-T. Pan, "A new fruit fly optimization algorithm: Taking the financial distress model as an example," *Knowl.-Based Syst.*, vol. 26, pp. 69–74, Feb. 2012.

[40] P. Bak, C. Tang, and K. Wiesenfeld, "Self-organized criticality: An explanation of the 1/f noise," *Phys. Rev. Lett.*, vol. 59, no. 4, pp. 381–384, Jul. 1987.

[41] S. Saremi, S. Z. Mirjalili, and S. M. Mirjalili, "Evolutionary population dynamics and grey Wolf optimizer," *Neural Comput. Appl.*, vol. 26, no. 5, pp. 1257–1263, 2015.

[42] Z. Yong, D. W. Gong, X. Y. Sun, and Y. N. Guo, "A PSO-based multi-objective multilabel feature selection method in classifcation," *Sci. Rep.*, vol. 7., no. 1, p. 376, Mar. 2017.

[43] B. Xue, M. Zhang, and W. N. Browne, "A comprehensive comparison on evolutionary feature selection approaches to classification," *Int. J. Comput. Intell. Appl.*, vol. 14, no. 2, Jun. 2015, Art. no. 1550008.

[44] E. Hancer, B. Xue, M. Zhang, D. Karaboga, and B. Akay, "Pareto front feature selection based on artificial bee colony optimization," *Inf. Sci.*, vol. 422, pp. 462–479, 2017.

[45] S. Kashef and H. Nezamabadi-Pour, "An advanced ACO algorithm for feature subset selection," *Neurocomputing*, vol. 147, pp. 271–279, Jan. 2015.

[46] L. Ping *et al.*, "Hashing algorithms for large-scale learning," in *Proc. NIPS*, 2011, pp. 2672–2680.

[47] A. Areej and E. Ashraf, "A minimal subset of features using feature selection for handwritten digit recognition," *J. Intell. Learn. Syst. Appl.*, vol. 9, no. 4, p. 422, 2017.

[48] M. Wieland and M. Pittore, "Performance evaluation of machine learning algorithms for urban pattern recognition from multi-spectral satellite images," *Remote Sens.*, vol. 6., no. 4, pp. 2912–2939, 2014.

YUN HOU received the B.E. degree in software engineering from Northeast Electric Power University, Jilin, China, in 2017. She is pursuing the M.Sc. degree in software engineering with Jilin University, China. Her research interests include intelligence optimization and feature selection.

JINGYAO LI received the B.S. degree in computer science and technology from Jilin University, Changchun, China, in 2016, where she is currently pursuing the Ph.D. degree in software engineering. Her research interests mainly include transfer learning and feature selection.

HAIHONG YU was born in 1975. He received the Ph.D. degree from Jilin University, where he is a Professor. His main research interests include decision support systems and machine learning. He is a member of CCF.

ZHANSHAN LI is currently a Professor of computer science with Jilin University. His main research interests include constraint reasoning and machine learning.