

Research Article

A Binary Superior Tracking Artificial Bee Colony with Dynamic Cauchy Mutation for Feature Selection

Xianghua Chu,^{1,2} Shuxiang Li,¹ Da Gao,¹ Wei Zhao,¹ Jianshuang Cui ,³
and Linya Huang ¹

¹College of Management, Shenzhen University, Shenzhen, China

²Institute of Big Data Intelligent Management and Decision, Shenzhen University, Shenzhen, China

³School of Economics and Management, University of Science and Technology Beijing, Beijing, China

Correspondence should be addressed to Jianshuang Cui; cuijs@manage.ustb.edu.cn and Linya Huang; huanglinya2019@email.szu.edu.cn

Received 7 August 2020; Revised 24 September 2020; Accepted 20 October 2020; Published 9 November 2020

Academic Editor: Zhile Yang

Copyright © 2020 Xianghua Chu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper aims to propose an improved learning algorithm for feature selection, termed as binary superior tracking artificial bee colony with dynamic Cauchy mutation (BSTABC-DCM). To enhance exploitation capacity, a binary learning strategy is proposed to enable each bee to learn from the superior individuals in each dimension. A dynamic Cauchy mutation is introduced to diversify the population distribution. Ten datasets from UCI repository are adopted as test problems, and the average results of cross-validation of BSTABC-DCM are compared with other seven popular swarm intelligence metaheuristics. Experimental results demonstrate that BSTABC-DCM could obtain the optimal classification accuracy and select the best representative features for the UCI problems.

1. Introduction

Feature selection is one of the cornerstones in machine learning [1] to select the proper combination of features to best describe the target problem. The useful features are retained while the redundant features and irrelevant features are removed by feature selection. As a result, an appropriate set of selected features can reduce computational complexity, improve knowledge discovery, and achieve satisfactory learning performance [2]. However, the number of possible combinations increases exponentially with the number of candidate features, which make efficient methods always in need for feature selection.

Existing feature selection methods can be classified as filter method, embedded method, and wrapper method [3]. The filter method is to evaluate the correlation between variables to reduce feature size of dataset, and the evaluation process does not involve specific learning algorithms [4]. The embedded method embeds feature selection with classifiers, where the commonly embedded methods include support vector machine (SVM), ID3, C4.5, and Lasso, a least-squares

regression method based on L1 regular terms [5]. The wrapper method uses a feature search component to produce feature subset and utilizes the specific classifier to evaluate the performance of different feature subsets until achieving termination conditions. In the wrapper method, swarm intelligence algorithms [6], such as particle swarm optimization [7], bacterial foraging optimization [8], grey wolf optimization [9], and brain storm optimization [10], have attracted great interests across various areas [5, 11].

Artificial bee colony (ABC) [12], as a recently proposed swarm intelligence metaheuristic [13], has been employed to address feature selection problems due to its promising efficiency and simple implementation. Keles and Kılıç [14] applied ABC to feature selection on SCADI dataset with 70 samples and 206 attributes. Seven features were finally selected from 206 features to classify the dataset with various classification methods. The classification accuracy was significantly improved. Kiliç and Keleş [15] proposed ABC to select features on z-Alizadeh Sani dataset with 303 samples and 56 attributes, and the classification accuracy is enhanced on the original data.

The promising applications encourage researchers to continuously improve the optimization performance of original ABC in feature selection. Shunmugapriya et al. [16] enhanced ABC by mining the global optimal solutions and previously abandoned solutions. Experimental results showed that the feature selection performance was improved without significant increase in computational cost. Özger et al. [17] implemented 8 variants of the binary ABC algorithm to solve feature selection problems, and the proposed algorithm using the bit-by-bit operator has a better global search capability. Wang et al. [18] diversified the initial food sources to make the initialization evenly distributed. Numerical results displayed that the method could get high classification accuracy and smaller feature subset. Liu [19] proposed an ABC variant based on knee points to accelerate the convergence speed on feature selection problems. Though the proposed strategies improve original ABC performance, however, in each iteration of the ABC variants, each food source is only updated in one dimension, and the searching strategy enables the bees to randomly learn from other bees. This results in slow convergence speed and inferior exploitation capability of ABC in the process of optimization.

In this work, a binary superior tracking artificial bee colony with dynamic Cauchy mutation (BSTABC-DCM) is proposed to further improve convergence speed and exploitation capability of ABC for feature selection. Two efficient search strategies, namely, superior tracking strategy [20] and dynamic Cauchy mutation, are integrated into the proposed algorithm. Compared with original ABC, superior tracking strategy enhances ABC' learning behaviors in two aspects: (1) instead of only updating one dimension in each iteration, bees learn from others in each dimension in each iteration; (2) instead of learning randomly, bees select individuals with better fitness to follow. A dynamic Cauchy mutation is integrated to diversify the population and improve global search ability. Ten datasets of different types of UCI are adopted to test BSTABC-DCM on feature selection problems. Seven popular swarm intelligence algorithms are included for comparison. Comprehensive experiments are conducted to evaluate the effectiveness of the proposed method for feature selection.

This paper is organized as follows: Section 2 introduces the background includes the principle of the original ABC algorithm and related work on feature selection. Section 3 presents the proposed BSTABC-DCM. Section 4 shows the process and results of the experiment. Section 5 makes conclusions.

2. Background

2.1. Artificial Bee Colony. The ABC framework is divided into employed bee stage, onlooker bee stage, and scout bee stage [21]. The employed bee first mines food source and shares the information of food source with the onlooker bee. The onlooker bee selects food source to be mined according to the information. If the better food source not be mined, the onlooker bee transforms into the scout bee and searches for the new food source [22, 23]. Firstly, initialize the food source. The initialization method is as follows:

$$x_{ij} = lb_j + \text{rand}(0, 1)(ub_j - lb_j), \quad (1)$$

in which $i = 1, 2, \dots, SN$, where SN refers to the number of food source, $j = 1, 2, \dots, D$, D refers to problem dimension, and ub_j and lb_j refer to the maximum and minimum values of the j dimension of source, respectively.

- (1) Employed bee stage: The amount of the employed bee is half of the initial food source, and it is attached to half of the better source. A new food source is mined near the attached food source. The mining method is shown in the following equation:

$$v_{ij} = x_{ij} + R_{ij}(x_{ij} - x_{kj}), \quad (2)$$

where v_{ij} refers to the newly mined food source, R_{ij} is a random number between $[-1, 1]$, and x_{ij} and x_{kj} refer to the j dimension of food source i, k , respectively.

Calculating and updating the objective function value of the new food source, and after the employed bee found a better food source, calculating the new fitness according to the following equation:

$$fit(x_i) = \begin{cases} \frac{1}{1 + f(x_i)}, & f(x_i) \geq 0, \\ 1 + |(x_i)|, & f(x_i) < 0, \end{cases} \quad (3)$$

where $f(x_i)$ refers to objective function value of food source x_i .

- (2) Onlooker bee stage: Calculating the probability by (4) and then selecting food source, the better food sources are retained by the roulette method. Further exploring food source by the following equation:

$$P_i = \frac{fit(x_i)}{\sum_{i=1}^D fit(x_i)}, \quad (4)$$

$$v_{ij} = x_{kj} + R_{ij}(x_{kj} - x_{gj}), \quad (5)$$

where x_{kj} refers to the j dimension of food source selected by roulette and x_{gj} is the food source different from k .

- (3) Scout bee stage: If the food source is not replaced by a better one, then the scout bees start to generate new food sources randomly according to equation (1).

2.2. Related Work on Feature Selection. It is a great challenge to train the high-dimensional dataset. With the increase in dimensions, the demand of training samples increases exponentially causing "curse of dimensionality". In addition, many models are inapplicable to high-dimensional data [24]. Therefore, feature selection is an important way to improve learning performance and a key role in the data pre-processing step of machine learning.

Existing feature selection methods can be divided into three categories: filter method, embedded method, and

wrapper method [25]. The principles of each type of method and the corresponding algorithm are reviewed in this section.

- (1) Filter method: The filter method is to evaluate the correlation between variables according to statistical, information theory, and distance measurement [4]. Filter methods find out K variables with strong correlation based on the correlation parameters including correlation coefficient, chi-square test, Fisher score, and information gain [5]. It does not involve any specific learning algorithm in the evaluation process. Peng et al. [26] derived the minimal-redundancy-maximal-relevance criterion (mRMR) based on the principle of statistics and improved a two-stage feature selection algorithm, and experimental results show that mRMR performs well in classification accuracy. Meyer and Bontempi [27] proposed a filter method of feature selection based on the double input symmetrical relevance (DISR), a new information theoretic selection criterion. The proposed method is competitive with other filter method of feature selection. Almuallim and Dietterich [28] presented a filter method FOCUS, a quasipolynomial time algorithm, and FOCUS can achieve good performance on coverage, sample complexity, and generalization. ABB and relief are also belonging to the filter method [25].
- (2) Embedded method: The embedded method integrates the process of feature selection with the process of classifier learning, which is mainly to solve the problem of high computational costs caused by the reconstruction of the classification model when processing different datasets. The commonly embedded methods can be divided three types [5]. The first is the pruning method, which is eliminating features from all features, and support vector machine (SVM) eliminates features recursively. The second is the feature selection algorithms with a build-in mechanism such as ID3 and C4.5. The third is regularization models. This method minimizes fitting errors by objective functions and the features with low coefficients are eliminated, such as lasso regularization and bridge regularization [5]. Mohsenzadeh et al. [29] proposed relevance sample feature machine (RSFM), which is an embedded feature selection method based on sparse Bayesian approach and Gaussian priors. The results show that RSFM performs well in both eliminating redundant features and classification accuracy. Mirzaei et al. [30] proposed an embedded feature selection method based on a fully Bayesian framework and introduced a multistep algorithm with variational approximation to maximum the posteriori probability. The proposed method is successful in both regression and classification.
- (3) Wrapper method: Wrapper methods put the datasets into the algorithm for training until obtaining the

best combination of features within the number of iterations. The commonly used algorithms include greedy search and stochastic search [31]. Greedy search has two methods: forward selection and backward elimination; forward selection expands the subset of features from empty, and backward elimination gradually eliminates features from the complete feature set [25]. The swarm intelligence algorithm belongs to stochastic search, like genetic algorithm (GA), ant colony optimization (ACO), and particle swarm optimization (PSO). Yang et al. [32] proposed chaotic binary particle swarm optimization (CBPSO) for feature selection and used two classifiers to test the algorithm, and the classification accuracy obtained by CBPSO is higher than that of other methods from the literature. Jingwei Too et al. [4] put multiple inertia weight strategy into binary particle swarm optimization and proposed CBPSO-MIWS for feature selection. The results show that CBPSO-MIWS can achieve competitive performance in all five swarm intelligence algorithms. Xue et al. [33] studied two multiobjective feature selection algorithms based on PSO; the first algorithm applied the idea of nondominated sorting into PSO while the second algorithm put mutation and crowding strategy into PSO to search for the Pareto solutions. These two algorithms can evolve a set of nondominated solutions automatically. Compared with the three famous multiobjective algorithms, the second algorithm obtains the better results. Cheng and Lu [34] integrated the sampling survey method into the heuristic intelligent optimization algorithm and proposed a new feature selection method. The sampling survey method is used to build the feature-scoring system and reduced dimension length-scoring system. Results showed that the proposed algorithm can select features quickly and effectively.

We have studied the wrapper feature selection method in this paper. The flowchart of the wrapper feature selection method is given in Figure 1.

3. Proposed Method

3.1. Binary Superior Tracking Strategy. In BSTABC-DCM, a binary superior tracking strategy is proposed for feature selection. Specifically, compared with ABC's search strategy, the integrated superior tracking strategy has two main differences: (1) bees learn from others in each dimension instead of only learning one dimension in each iteration and (2) bees choose the better individuals to learn instead of random moving. The food sources are updated as the following equation:

$$v_i = x_i + r_i(x_i - SN_i), \quad (6)$$

where SN_i denotes the superior neighbor for guidance; it is a D -dimensional vector of which elements are constructed by itself (position of the food source) for i -th food source and other superior food sources with two probabilistic selection

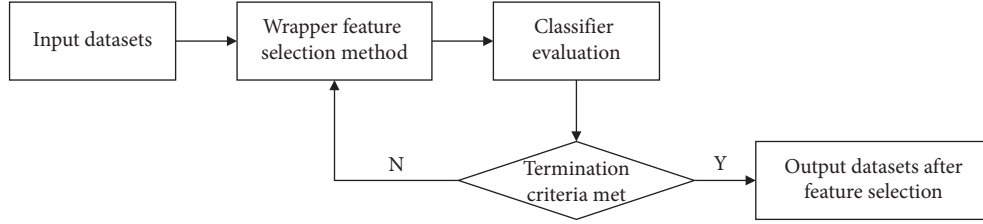


FIGURE 1: Flowchart of the wrapper feature selection method.

methods (roulette selection and tournament selection). The pseudocode for generating SN_i is presented in Algorithm 1.

Pr is the initialized probability threshold, which determines whether the current individual learns from his superior neighbor or itself. rand is a function used to produce an integer from a uniform discrete distribution, FS refers to the position of food source, and FV is the function value.

As feature selection is a binary optimization problem, the continuous food source is converted to a binary one after updating the food source using equation (6). Two-step binarization technique [35] is used to make the binarization of the continuous solution. The first step corresponds to the transfer function; sigmoid function [4] is adopted to transform the position of food source into probability value according to equation (7). The second step is binarization technique; the probability value of food source is converted to the binary one by applying the following equation:

$$P(v_i) = \frac{1}{1 + \exp(-v_i)}, \quad (7)$$

$$V_i = \begin{cases} 1, & \text{rand} < P(v_i), \\ 0, & \text{rand} > P(v_i), \end{cases} \quad (8)$$

where v_i refers to the continuous food source, V_i refers to the binary food source, and rand is a random number uniformly distributed between 0 and 1.

3.2. Dynamic Cauchy Mutation Method. To enhance the global exploration of the proposed method, a dynamic Cauchy mutation is implemented to refine the global best solution in each iteration. The dynamic Cauchy mutation method in this study is defined [36] as the following equation:

$$v'_{g,d} = v_{g,d} + \alpha \cdot \text{cauchy}(\delta), \quad (9)$$

where d is a random number between $[1, D]$, δ denotes the Cauchy distribution scale, $v_{g,d}$ is the global best solution in each iteration, and cauchy refers to a random number generated by $\text{Cauchy}()$ distribution. The value of δ is set to 1 to balance the exploitation ability of the proposed algorithm, and α is a dynamic weight defined as follows:

$$\alpha = 1 - \text{rand}^{(\text{iter}/T)}, \quad (10)$$

where rand is a random number between $[0, 1]$, and iter means the current number of iteration.

3.3. Procedure of the Proposed Method. BSTABC-DCM includes two main searching components: binary superior tracking strategy (BST) and dynamic Cauchy mutation method (DCM). In BSTABC-DCM, the food sources are explored using equation (6) in the employed bee stage and onlooker bee stage, where each dimension of food source is updated by learning from the better food sources. This ensures the timely information exchange between food sources. After the onlooker bee stage, a dynamic Cauchy mutation method is implemented to increase population diversity, and the most fertile food sources are explored. The flowchart of BSTABC-DCM is given in Figure 2.

4. Experimental Study

In this section, a set of feature selection problems is employed to comprehensively verify the performance of BSTABC-DCM.

4.1. Experiment Settings. Three metrics including classification accuracy, average classification accuracy, and the number of selected features are adopted for performance assessment. Seven state-of-the-art swarm intelligence algorithms, i.e., artificial bee colony (ABC) algorithm [12], gbest-guided ABC (GABC) [37], quick ABC (qABC) [38], comprehensive learning particle swarm optimizer (CLPSO) [39], PSO [7], GWO [9], and BSO [10], are included for comparison.

Regarding the parameter setting of BSTABC-DCM, KNN has been widely used in various fields with relatively large sample size, such as pattern recognition, text categorization, and moving object recognition [40, 41]. As more than half of the test datasets are relatively large in size, KNN ($k = 5$) is selected as the classification algorithm in our study. In each experiment, we conduct 5-fold experiment by dividing 80% of the dataset as training set and the remaining 20% as testing set. Each algorithm runs 20 times independently in each dataset. Evaluation indicators include best accuracy, worst accuracy, mean accuracy, standard deviation (STD), and the number of redundant features removed. The final accuracy is the average accuracy of 5-fold cross-validation [42].

4.1.1. Benchmark Datasets. Considering the number of instances and dimensions of different datasets, ten datasets from the UCI are adopted as test problems [4]. The number of instances and dimensions of the datasets are shown in Table 1.

```

Input parameters
For  $d = 1 : D$ 
  If  $\text{rand} < Pr$ 
     $A_1 = FS(\text{randi}_1(ps))$ 
     $A_2 = FS(\text{randi}_2(ps))$ 
    If  $FV(A_1) < FV(A_2)$ 
       $SN_{i,d} = A_{1,d}$ 
    Else  $SN_{i,d} = A_{2,d}$ 
    End
  Else  $SN_{i,d} = A_{i,d}$ 
  End
End
Return  $SN_{i,d}$ 

```

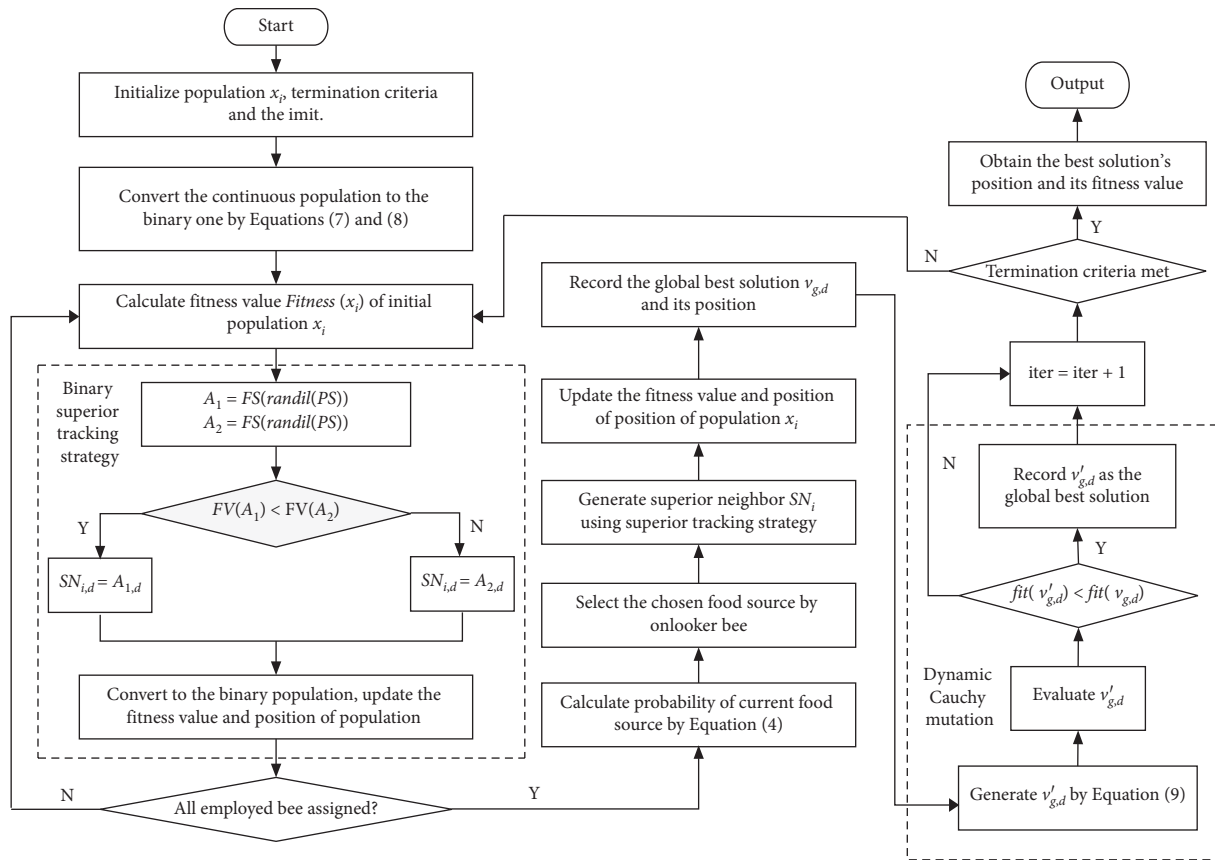
ALGORITHM 1: Pseudocode for generating SN_i .

FIGURE 2: Flowchart of BSTABC-DCM.

4.1.2. Parameter Settings. For fair comparison, the number of population and the number of iterations of each algorithm are set to 10 and 100, and each algorithm runs 20 times independently. All the analyses are implemented in MATLAB 2017b with Intel Core i7 2.6 GHz and 16.0 GB RAM. The specific parameter settings of compared algorithms are shown in Table 2.

4.2. Experimental Results. For each experiment, we randomly divide the datasets into five parts, with 80% as training set and 20% as testing set. The average results

obtained by 20 times 5-fold cross-validation are obtained as the final results. Without feature selection, the classification accuracy of the KNN ($K = 5$) algorithm is shown in Table 3.

We implement the comparison algorithms on the ten datasets for feature selection. The average results obtained by cross-validation are regarded as the final experiment results. The experimental results of the proposed algorithm and other seven compared algorithms are shown in Tables 4 and 5. The best experimental results among the eight algorithms are shown in bold.

TABLE 1: The ten utilized benchmark datasets.

UCI dataset	Number of instances	Number of features	Number of classes
Breastcancerwisconsin	699	9	2
Diabetic retinopathy	1151	19	2
Glass identification	214	10	6
Ionosphere	351	34	2
Movementlibras	360	90	15
Musk_1	476	166	2
BreastCancerCombiadataR2	116	9	2
Lung cancer	32	55	3
Pdspeechfeatures	756	753	2
Seeds	210	7	3

TABLE 2: Specific parameter settings of comparison algorithms.

Algorithms	Parameter values
Artificial bee colony (ABC) [10]	Random number $R_{ij} \in [-1, 1]$
Binary superior tracking artificial bee colony with dynamic Cauchy mutation (BSTABC-DCM) [33]	Cauchy distribution scale $\delta = 1$
Gbest-guided ABC (GABC) [34]	Nonnegative constant $C = 1.5$, $\psi_{ij} = 1.5 \times \text{rand}$
Quick ABC (qABC) [35]	$\text{dis} \leftarrow (\text{sum}(\text{dis}) / (\text{FoodNumber} - 1))$
Comprehensive learning particle swarm optimizer (CLPSO) [36]	$w_0 = 0.9, w_1 = 0.4, c = 1.49445$
Particle swarm optimization (PSO) [11]	$w_0 = 0.9, w_1 = 0.4, c_1 = c_2 = 2$
Brain storm optimization (BSO) [9]	$\text{num_cluster} = 3, \text{prob_one_cluster} = 0.8$
Grey wolf optimization (GWO) [8]	$a = 2 - ((2 \cdot \text{current_num_iter}) / \text{num_iter})$

TABLE 3: Classification accuracy without feature selection.

UCI dataset	Classification accuracy without feature selection
Breastcancerwisconsin	97.36
Diabetic retinopathy	64.73
Glass identification	67.29
Ionosphere	84.62
Movementlibras	76.11
Musk_1	87.61
BreastCancerCombiadataR2	51.72
Lung cancer	56.25
Pdspeechfeatures	72.22
Seeds	88.10

TABLE 4: Experimental results of the eight algorithms.

Dataset	Algorithm	Best accuracy	Worst accuracy	Mean accuracy	STD
Breastcancerwisconsin	BSTABC-DCM	98.10	97.95	98.04	0.0008
	GABC	97.95	97.66	97.89	0.0013
	CLPSO	98.10	97.95	97.98	0.0006
	PSO	97.95	97.66	97.83	0.0012
	BSO	97.95	97.81	97.92	0.0006
	qABC	97.95	97.80	97.83	0.0007
	ABC	97.95	97.66	97.81	0.001
	GWO	98.10	97.80	97.92	0.0012
Diabetic retinopathy	BSTABC-DCM	72.03	69.07	70.69	0.0136
	GABC	70.98	69.94	70.29	0.0041
	CLPSO	70.98	70.11	70.6	0.0033
	PSO	70.72	68.12	69.73	0.0119
	BSO	71.15	69.51	70.25	0.0073
	qABC	71.41	69.59	70.51	0.0079
	ABC	70.81	70.2	70.53	0.0025
	GWO	71.85	69.06	71.05	0.0116

TABLE 4: Continued.

Dataset	Algorithm	Best accuracy	Worst accuracy	Mean accuracy	STD
Glass identification	BSTABC-DCM	73.87	72.88	73.29	0.0043
	GABC	73.03	70.99	72.17	0.0075
	CLPSO	72.99	72.40	72.63	0.003
	PSO	72.53	71.54	72	0.0046
	BSO	73.84	72.41	73.1	0.0065
	qABC	72	70.56	71.42	0.0052
	ABC	72.46	71.00	71.7	0.0071
	GWO	73.31	70.06	72.28	0.0133
Ionosphere	BSTABC-DCM	90.6	88.62	89.58	0.0087
	GABC	89.74	87.19	88.33	0.0094
	CLPSO	88.05	87.19	87.69	0.0037
	PSO	88.62	86.92	87.59	0.0062
	BSO	89.46	88.31	89.06	0.0043
	qABC	89.17	87.19	88.26	0.0073
	ABC	88.89	87.19	87.76	0.0066
	GWO	90.02	88.05	88.95	0.0081
Movementlibras	BSTABC-DCM	78.89	78.21	78.48	0.003
	GABC	78.03	77.23	77.61	0.0037
	CLPSO	78	77.55	77.77	0.0016
	PSO	78.57	76.93	77.54	0.0081
	BSO	78.38	77.18	77.8	0.0042
	qABC	78.12	77.59	77.8	0.0027
	ABC	77.79	77.28	77.65	0.0022
	GWO	79.16	77.28	78.22	0.008

TABLE 5: Experimental results of the eight algorithms.

Dataset	Algorithm	Best accuracy	Worst accuracy	Mean accuracy	STD
Musk_1	BSTABC-DCM	90.95	89.92	90.21	0.0043
	GABC	89.92	88.23	88.83	0.0065
	CLPSO	90.34	88.87	89.66	0.0054
	PSO	90.13	88.45	89.04	0.0069
	BSO	90.14	88.44	89.33	0.0071
	qABC	89.71	88.44	89.16	0.0059
	ABC	88.87	88.23	88.48	0.0023
	GWO	90.34	88.24	89.62	0.0087
BreastCancerCombiadataR2	BSTABC-DCM	86.99	85.43	86.08	0.0064
	GABC	86.30	84.41	85.08	0.0081
	CLPSO	86.20	84.44	85.02	0.0074
	PSO	85.43	77.57	82.11	0.0334
	BSO	86.11	81.8	84.31	0.0167
	qABC	86.19	83.54	85.08	0.0114
	ABC	86.23	80.74	83.94	0.0201
	GWO	86.30	83.69	84.92	0.0099
Lung cancer	BSTABC-DCM	82.38	76.67	80.48	0.0231
	GABC	78.38	72.86	75.81	0.021
	CLPSO	81.43	75.71	78.65	0.0238
	PSO	78.86	70.29	73.92	0.0327
	BSO	80.95	69.05	77.24	0.0468
	qABC	78.57	75.05	76.59	0.0164
	ABC	78.57	69.33	74.8	0.0401
	GWO	81.90	76.19	79.10	0.0235

TABLE 5: Continued.

Dataset	Algorithm	Best accuracy	Worst accuracy	Mean accuracy	STD
Pdspeechfeatures	BSTABC-DCM	76.19	75.27	75.77	0.0034
	GABC	75.79	75.4	75.58	0.0015
	CLPSO	75.79	75.52	75.66	0.0045
	PSO	75.66	75.26	75.42	0.0017
	BSO	75.79	75.26	75.53	0.0018
	qABC	76.06	75.27	75.64	0.0034
	ABC	75.67	75.27	75.51	0.0017
	GWO	75.80	75.39	75.63	0.0018
Seeds	BSTABC-DCM	97.14	96.19	96.67	0.0034
	GABC	96.67	96.19	96.57	0.0021
	CLPSO	96.67	96.19	96.48	0.0026
	PSO	96.67	93.33	95.52	0.0129
	BSO	96.67	95.24	96.1	0.0062
	qABC	96.67	95.71	96.19	0.0048
	ABC	96.67	95.24	96.19	0.0058
	GWO	96.67	96.19	96.48	0.0026

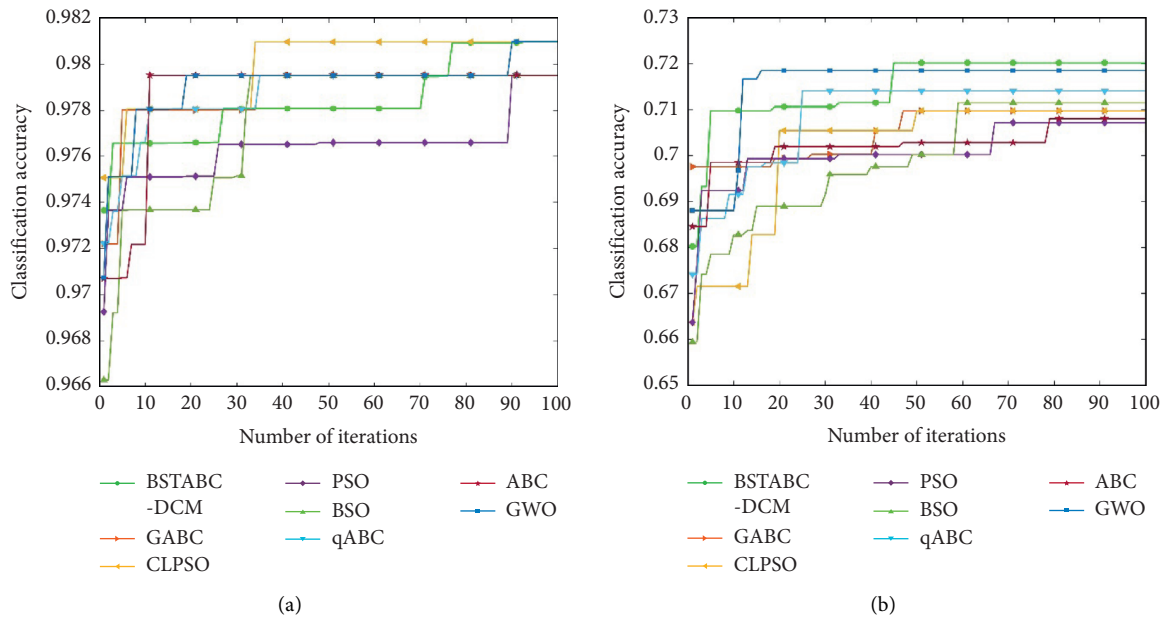
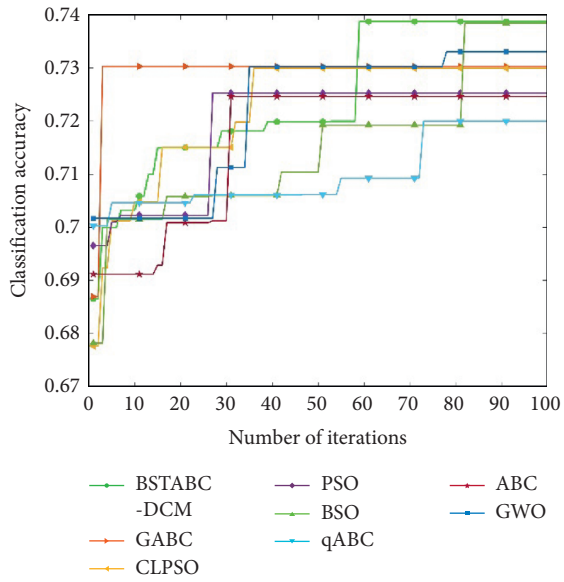
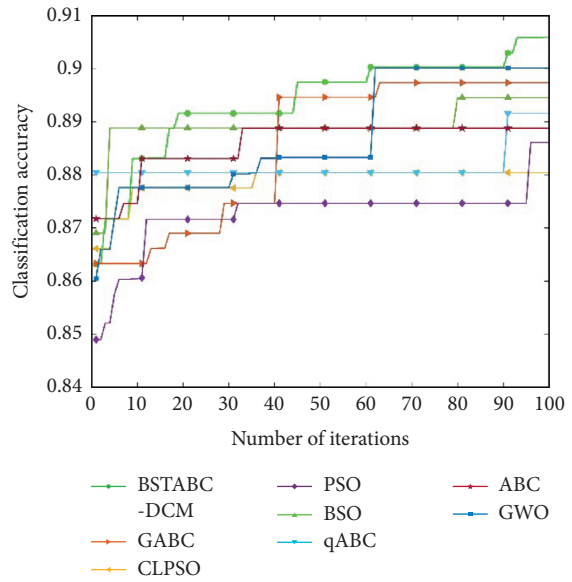


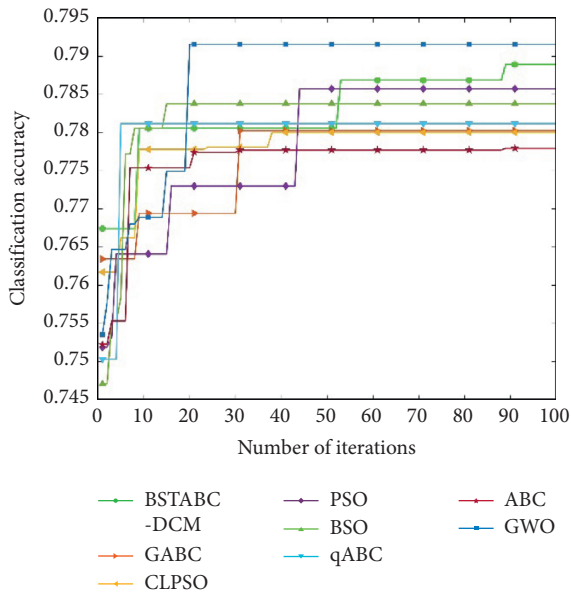
FIGURE 3: Continued.



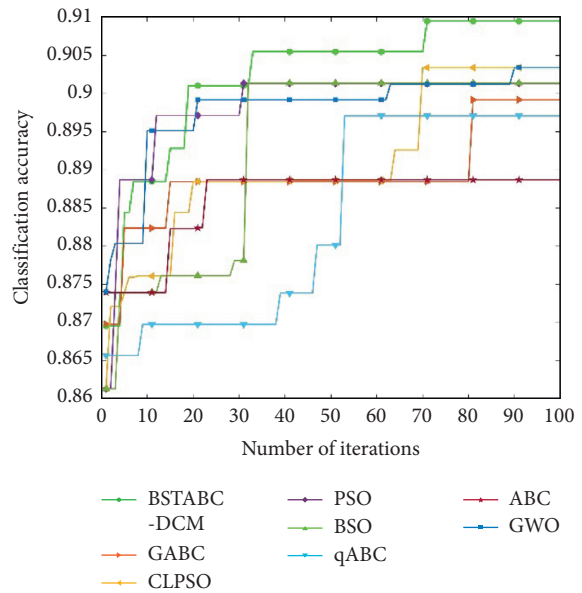
(c)



(d)



(e)



(f)

FIGURE 3: Continued.

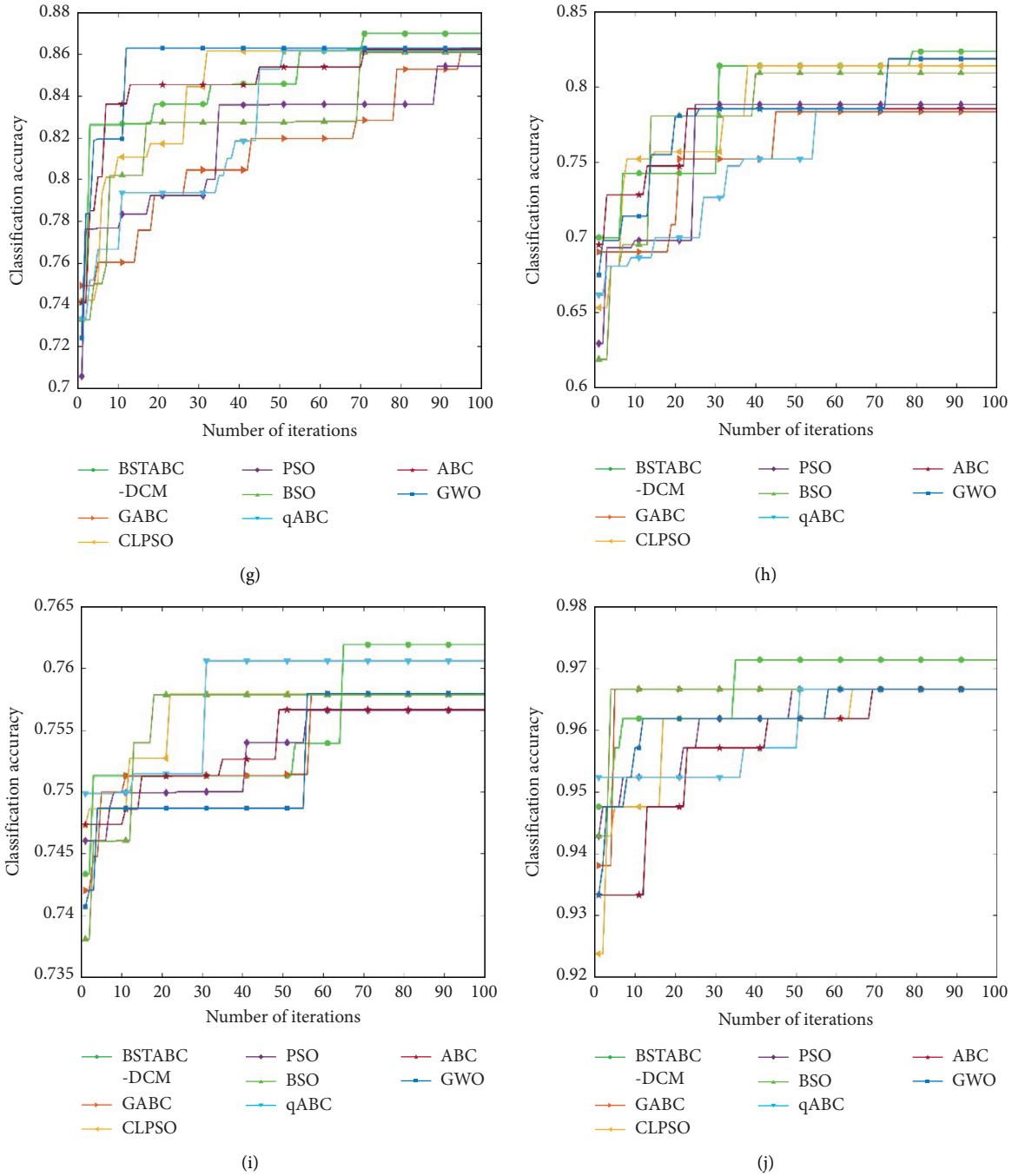


FIGURE 3: Convergence curves of eight feature selection algorithms. (a) Breastcancerwisconsin. (b) Diabetic retinopathy. (c) Glass identification. (d) Ionosphere. (e) Movementlibras. (f) Musk_1. (g) BreastCancerCombiadataR2. (h) Lung cancer. (i) Pdspeechfeatures. (j) Seeds.

From the results, it can be observed that the optimal classification accuracy obtained by BSTABC-DCM is higher than that of the other seven algorithms on 8 out of 10 datasets. The average classification accuracy obtained by BSTABC-DCM is higher on 9 out of 10 datasets. The classification accuracy of BSTABC-DCM increases by 16.67% on average, while the classification accuracy increases by 15.27% on average by other seven algorithms. Specifically, for classification accuracy, BSTABC-DCM

increases more than 10% on 4 datasets, the next is GWO, which increases more than 10% on 3 datasets, and other six algorithms increase only on 2 datasets. The highest increased classification accuracy obtained by BSTABC-DCM is in BreastCancerCombiadataR2 dataset, which is 68.19%.

Figure 3 demonstrates the convergence process of the comparison algorithms, where the proposed BSTABC-DCM is shown in green. It is observed that BSTABC-DCM

TABLE 6: The number of redundant features removed.

	BSTABC-DCM	GABC	CLPSO	PSO	BSO	qABC	ABC	GWO
Breastcancerwisconsin	1.4	2.0	1.0	3.0	2.8	2.0	2.2	2.0
Diabetic retinopathy	11.2	9.8	9.0	10.8	10.6	10.2	10.6	11
Glass identification	1.8	1.2	2.2	3	3	2.4	2.6	1.0
Ionosphere	22.6	21.4	17	18.6	21.2	21.4	19.8	20.4
Movementlibras	43	43.2	33.2	41.2	41.4	43.6	43.8	44.6
Musk_1	81.2	83.8	81.2	75.2	78.8	84.2	81.4	81.5
BreastCancerCombiadataR2	3.2	3.6	3.4	3.8	3.8	3.8	3.4	3.8
Lung cancer	24.6	25.6	18.6	24.2	24	24.6	25.8	26.2
Pdspeechfeatures	392.1	372.4	373.7	368.6	382.0	374.2	377.8	365
Seeds	3.2	4.0	3.6	3.2	4.2	4.2	4.0	4.0
Optimal number	3/584.3	0/567	0/542.9	3/551.6	3/571.8	3/570.6	2/571.4	3/559.5

converges faster than other algorithms on 8 out of 10 datasets. The classification results and the convergence curves indicate BSTABC-DCM enhances ABC's exploitation capability.

In addition to classification accuracy, another indicator to measure the performance of algorithms is the number of redundant features removed. The number of redundant features removed is the average number obtained by 20 runs. Table 6 shows the number of redundant features removed by the eight algorithms. The best value of the eight methods is shown in bold.

The result in the optimal number row represents the number of data sets that each algorithm performs best/the total number of features removed. As can be seen from the above table, the proposed algorithm BSTABC-DCM removed the most redundant features in three datasets, which are diabetic retinopathy dataset, ionosphere dataset, and Pdspeechfeatures dataset. Although BSO and qABC also performed well in removing redundant features, judging from the percentage of redundant functions removed in the original dataset, BSTABC-DCM is undoubtedly the most promising, reaching 43.37%. In other words, compared with the original dataset, BSTABC-DCM can greatly reduce the number of features and ensure higher classification accuracy after feature selection. In conclusion, BSTABC-DCM has promising performance on the test problems.

5. Discussion

In summary, the above experimental results show that the proposed method has improved the best classification accuracy on 8 out of 10 datasets. For the number of removed redundant features, BSTABC-DCM is outperforming with the highest percentage of redundant features removed. This reveals that our proposed strategies have significantly enhanced the original ABC algorithm in terms of global exploration and exploitation capabilities. In addition, it can be seen from Figure 2 that the convergence speed of BSTABC-DCM is comparable to the other comparison algorithms. Though convergence speed is comparable to the other algorithms, it is promising to combine the advantages of different methods, e.g., random forest and Naive Bayes, with the proposed algorithm to further refine the optimization process.

6. Conclusion

In this study, a binary superior tracking artificial bee colony with dynamic Cauchy mutation (BSTABC-DCM) is proposed for feature selection. Specifically, a binary superior tracking strategy is integrated to improve the learning behavior of bees by boosting the efficiency of information sharing in population. In each iteration, the bees can learn from the superior bees in each dimension. A dynamic Cauchy mutation is introduced to diversify population and enhance global search ability. We select ten datasets from UCI to verify the performance of BSTABC-DCM, and seven state-of-the-art swarm intelligence algorithms are included for comparison. Experimental results indicate that BSTABC-DCM achieves the best results on classification accuracy while removing nearly half of the redundant features. The convergence speed of BSTABC-DCM is comparable to the comparison algorithms.

While promising, there are still margins for further investigation. Firstly, different binary conversion methods could be considered to improve optimization efficiency. Secondly, both the classification accuracy and feature size will be combined with fitness function to build a multi-objective model. Thirdly, adaptive search strategies can be considered to enhance the adaptability of the proposed method. Applying BSTABC-DCM to more practical problems such as capacitated location problem of distribution center, vehicle routing problems, and scheduling problems is also one of our research directions.

Data Availability

The data used to support the findings are available from UCI Machine Learning Repository.

Disclosure

An earlier version of this work was previously presented in the International Conference on Neural Computing for Advanced Application.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (grant nos. 71971142, 71501132, and 71701079).

References

- [1] Y. Cui, S. Xu, L. Zhang, R. E. Welsh, and B. K. P. Horn, "The key techniques and future vision of feature selection in machine learning," *Journal of Beijing University of Posts and Telecommunications*, vol. 41, no. 1, pp. 1–12, 2018.
- [2] D. Manoranjan and H. Liu, "Feature selection for classification," *Intelligent Data Analysis*, vol. 1, no. 1–4, pp. 131–156, 1997.
- [3] C. Girish and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.
- [4] J. Too, A. R. Abdullah, N. Mohd Saad, and N. M. Saad, "A new co-evolution binary particle swarm optimization with multiple inertia weight strategy for feature selection," *Informatics*, vol. 6, no. 2, p. 21, 2019.
- [5] J. L. Tang, S. Alelyani, and S. H. Liu, "Feature selection for classification: A review," *Data Classification*, vol. 37, pp. 29–60, 2014.
- [6] X. Chu, F. Cai, C. Cui, M. Hu, L. Li, and Q. Qin, "Adaptive recommendation model using meta-learning for population-based algorithms," *Information Sciences*, vol. 476, pp. 192–210, 2019.
- [7] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, Perth, Western Australia, November 1995.
- [8] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 52–67, 2002.
- [9] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary grey wolf optimization approaches for feature selection," *Neurocomputing*, vol. 172, pp. 371–381, 2016.
- [10] Y. Shi, "An optimization algorithm based on brainstorming process," *International Journal of Swarm Intelligence Research*, vol. 2, no. 4, pp. 35–62, 2011.
- [11] Y. Wang, "Research on Feature Selection Methods Based on Swarm Intelligent Algorithm," Jilin University, Changchun, China, 2019.
- [12] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.
- [13] X. Chu, T. Wu, J. D. Weir, Y. Shi, B. Niu, and L. Li, "Learning–interaction–diversification framework for swarm intelligence optimizers: a unified perspective," *Neural Computing and Applications*, vol. 32, no. 6, pp. 1–21, 2018.
- [14] M. K. Keleş and Ü. Kılıç, "Artificial bee colony algorithm for feature selection on scadi dataset," in *Proceedings of the 2018 3rd International Conference On Computer Science and Engineering (UBMK)*, pp. 463–466, Bosnia and Herzegovina, Sarajevo, September 2018.
- [15] Ü. Kiliç and M. K. Keleş, "Feature selection with artificial bee colony algorithm on Z-Alizadeh Sani dataset," in *Proceedings of the 2018 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pp. 1–3, Adana, Turkey, October 2018.
- [16] P. Shunmugapriya, S. Kanmani, R. Supraja, and K. Saranya, "Feature selection optimization through enhanced artificial bee colony algorithm," in *Proceedings of the International Conference on Recent Trends in Information Technology (ICRTIT)*, pp. 56–61, Piscataway, NJ, USA, July 2013.
- [17] Z. B. Özger, B. Bolat, and B. Diri, "A comparative study on binary artificial bee colony optimization methods for feature selection," in *Proceedings of the International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*, pp. 1–4, Sinaia, Romania, August 2016.
- [18] H. Wang, H. Yu, Q. Zhang et al., "Parameters optimization of classifier and feature selection based on improved artificial bee colony algorithm," in *Proceedings of the International Conference On Advanced Mechatronic Systems (ICAMEchs)*, pp. 242–247, Melbourne, Australia, November 2016.
- [19] M. Liu, "Improved Multi-Objective Artificial Bee Colony Algorithm and its Application in Feature Selection," Anhui University, Hefei, China, 2018.
- [20] X. Chu, G. Hu, B. Niu, L. Li, and Z. Chu, "An superior tracking artificial bee colony for global optimization problems," in *Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC)*, pp. 2712–2717, Vancouver, BC, Canada, July 2016.
- [21] X. Chao and W. Li, "Feature selection method optimized by artificial bee colony algorithm," *Journal of Frontiers of Computer Science and Technology*, vol. 13, no. 2, pp. 300–309, 2019.
- [22] Y. Li, "Research and Application of Multi-Objective Artificial Bee Colony algorithm"[D], Northeastern University, Boston, MA, USA, 2012.
- [23] Q. Qin, S. Cheng, L. Li, and Y. Shi, "A review of artificial bee colony algorithm," *CAAI Transactions on Intelligent Systems*, vol. 9, no. 2, pp. 127–135, 2014.
- [24] Z. Li, J. Du, B. Nie et al., "Summary of feature selection methods," *Computer Engineering and Applications*, vol. 55, no. 24, pp. 10–19, 2019.
- [25] V. Kumar and S. Minz, "Feature selection: a literature review," *The Smart Computing Review*, vol. 4, no. 3, pp. 211–229, 2014.
- [26] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.
- [27] P. E. Meyer and G. Bontempi, "On the use of variable complementarity for feature selection in cancer classification," in *Proceedings of the Workshops On Applications of Evolutionary Computation*, pp. 91–102, Budapest, Hungary, April 2006.
- [28] H. Almuallim and T. G. Dietterich, "Learning with many irrelevant features," *Artificial Intelligence*, vol. 91, pp. 547–552, 1991.
- [29] Y. Mohsenzadeh, H. Sheikhzadeh, A. M. Reza, N. Bathaee, and M. M. Kalayeh, "The relevance sample-feature machine: a sparse Bayesian learning approach to joint feature-sample selection," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 2241–2254, 2013.
- [30] A. Mirzaei, Y. Mohsenzadeh, and H. Sheikhzadeh, "Variational relevant sample-feature machine: a fully Bayesian approach for embedded feature selection," *Neurocomputing*, vol. 241, pp. 181–190, 2017.
- [31] B. Venkatesh and J. Anuradha, "A review of feature selection and its methods," *Cybernetics and Information Technologies*, vol. 19, no. 1, pp. 3–26, 2019.
- [32] C.-S. Yang, L.-Y. Chuang, J.-C. Li, and C.-H. Yang, "Chaotic maps in binary particle swarm optimization for feature

- selection,” in *Proceedings of the 2008 IEEE Conference On Soft Computing in Industrial Applications*, pp. 107–112, 2008.
- [33] B. Xue, M. Zhang, and W. N. Browne, “Particle swarm optimization for feature selection in classification: a multi-objective approach,” *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1656–1671, 2012.
- [34] Z. Cheng and Z. Lu, “A novel efficient feature dimensionality reduction method and its application in engineering,” *Complexity*, vol. 2018, Article ID 2879640, 14 pages, 2018.
- [35] B. Crawford, R. Soto, G. Astorga, J. García, C. Castro, and F. Paredes, “Putting continuous metaheuristics to work in binary search spaces,” *Complexity*, vol. 2017, Article ID 8404231, 19 pages, 2017.
- [36] X. Chu, D. Gao, S. Cheng et al., “Worker assignment with learning-forgetting effect in cellular manufacturing system using adaptive memetic differential search algorithm,” *Computers & Industrial Engineering*, vol. 136, pp. 381–396, 2019.
- [37] G. Zhu and S. Kwong, “Gbest-guided artificial bee colony algorithm for numerical function optimization,” *Applied Mathematics and Computation*, vol. 217, no. 7, pp. 3166–3173, 2010.
- [38] D. Karaboga and B. Gorkemli, “A quick artificial bee colony (qABC) algorithm and its performance on optimization problems,” *Applied Soft Computing*, vol. 23, pp. 227–238, 2014.
- [39] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, “Comprehensive learning particle swarm optimizer for global optimization of multimodal functions,” *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 3, pp. 281–295, 2006.
- [40] Z. Deng, X. Zhu, D. Cheng, M. Zong, and S. Zhang, “Efficient k NN classification algorithm for big data,” *Neurocomputing*, vol. 195, pp. 143–148, 2016.
- [41] S. Dhanabal and S. Chandramath, “A review of various k -nearest neighbor query processing techniques,” *International Journal of Computer Applications*, vol. 31, no. 7, pp. 14–22, 2011.
- [42] C. Alippi and M. Roveri, “Virtual k -fold cross validation: An effective method for accuracy assessment,” in *Proceedings of the International Joint Conference On Neural Networks (IJCNN)*, pp. 1–6, Barcelona, Spain, July 2010.