IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 14, 2020, accepted July 25, 2020, date of publication July 29, 2020, date of current version September 4, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3012838

A Hybrid Improved Dragonfly Algorithm for Feature Selection

XUETING CUI[1,2], YING LI[1,2], JIAHAO FAN[1,2], TAN WANG[3], AND YUEFENG ZHENG[4]

[1]College of Computer Science and Technology, Jilin University, Changchun 130012, China
[2]Key Laboratory of Symbolic Computation and Knowledge Engineering, Ministry of Education, Jilin University, Changchun 130012, China
[3]Space Technology (Jilin) Company Ltd., Changchun 130000, China
[4]Computer College, Jilin Normal University, Siping 136000, China

Corresponding author: Jiahao Fan (jihanfan@hotmail.com)

This work was supported in part by the Development and Reform Commission of Jilin Province under Grant 2019C053-13, and in part by the Department of Science and Technology of Jilin Province under Grant 20190303135SF.

ABSTRACT Feature selection, which eliminates irrelevant and redundant features, is one of the most efficient classification methods. However, searching for an optimal subset from the original set is still a challenging problem. This paper proposes a novel feature selection algorithm named hybrid improved dragonfly algorithm (HIDA) which combines the advantages of both mRMR and improved dragonfly algorithm (IDA) in order to generate promising candidate subset and achieve higher classification accuracy rate. Firstly, to generate promising subset, features with small weight have chance to be selected into candidate subset with a small probability in mRMR. Secondly, to balance the exploitation and exploration capabilities of IDA, dynamic swarming factors are proposed to balance global and local capability. Lastly, to enhance the exploitation capability of IDA, quantum local optimum and global optimum are introduced in the position updating mechanism. The performance of HIDA is investigated on ten gene expression datasets and eight UCI data sets from the UCI Machine Learning Data Repository. Results show that the performance of HIDA is superior to BBA, BDA, CDA, LBPSO, MPMDWOA and MSMCCS.

INDEX TERMS Feature selection, hybrid optimization, dragonfly algorithm, mRMR, classification.

I. INTRODUCTION

Real-life datasets are often characterized by a large number of irrelevant or redundant features. The combination of features requires testing different subsets to find the optimal one [1]. This approach is highly time or space consuming. Mining large data requires the help of machine learning algorithms because of the increasing trend of high-dimensional data collection [2]. Feature selection is able to find a subset of features to decrease the number of features without degrading the prediction accuracy of the classifier [3], [4].

Feature selection algorithms are divided into three categories, filter methods [5], wrapper methods [6], and hybrid methods [7]. Filter methods analyze the internal characteristics of data regardless of machine learning. The advantages of filter methods are low computational cost and suitable for high-dimensional data. The famous Maximum Relevance Minimum Redundancy (mRMR) algorithm [8] and ReliefF [9] are representative filter methods. Wrapper methods utilize machine learning to get higher classification accuracy with computational overhead.

The meta-heuristic algorithms are very successful machine learning algorithms to solve various optimization problems [10]. From Fig.1, the meta-heuristic algorithms are divided into two families which are called individual-based algorithms and population-based algorithms. The well-known individual-based algorithms are hill climbing [11], iterated local search [12], and simulated annealing [13]. Population-based algorithms can be classified into three main categories including evolutionary algorithms, swarm-based algorithms, and physical phenomena algorithms. Some of the recently proposed evolutionary algorithms are biogeography-based optimization algorithms [14], evolutionary membrane algorithms [15], and human evolutionary model algorithms [16]. Some of the well-known swarm-based algorithms include Particle Swarm Optimization (PSO) [17], Genetic Algorithm (GA) [18], Ant Colony Optimization (ACO) [19], Bacterial Foraging Optimization (BFO) [20], Artificial Bee Colony (ABC) [21], Cuckoo Search (CS) [22], Whale Optimization Algorithm [23], [24]

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

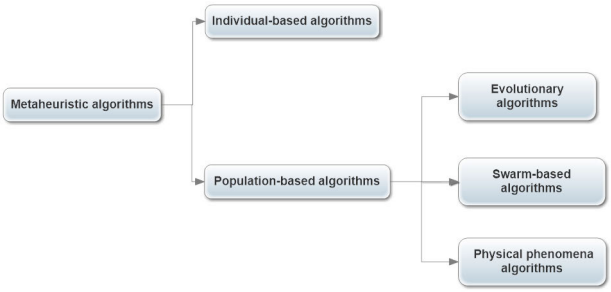

FIGURE 1. The classification of meta-heuristic algorithms.

and Dragonfly Algorithm (DA) [25]. The DA is regarded as a successful algorithm that outperforms other well-known optimizers due to its simplicity and efficiency [25]. The most recent physical phenomena algorithms in this category are Henry Gas Solubility Optimization [26]–[28], Gravitational Search Algorithm [29], Artificial Chemical Reaction Optimization. Algorithm [30], Ray Optimization [31], Central Force Optimization [32], Kinetic Gas Molecules, and Gases Brownian Motion Optimization [33].

Among these meta-heuristic algorithms, PSO and GA are well-known optimizers. Compared to PSO, high exploitation assists DA algorithm to rapidly converge towards the global optimum. Compared to GA, DA algorithm appropriately balances exploration and exploitation to handle difficulty in challenging search space. Therefore, in this paper, DA is selected as a wrapper method for feature selection.

There are some efficient evolutionary feature selection approaches, such as variable-size cooperative coevolutionary particle swarm optimization algorithm (VS-CCPSO) [34], binary differential evolution with self-learning (MOFS-BDE) [35], two-archive multi-objective artificial bee colony algorithm (TMABC-FS) [36], and return-cost-based binary FFA (Rc-BBFA) [37]. However, VS-CCPSO, MOFS-BDE, TMABC-FS and Rc-BBFA are not suitable for high-dimensional datasets. Memory based Hybrid Dragonfly Algorithm (MHDA) introduces internal memory to overcome the premature convergence of DA [38]. However, balancing local and global search capability of DA is not taken into consideration in the MHDA. To accelerate the convergence rate of DA, chaotic maps are employed to adjust five behaviors of dragonfly for high-dimensional data in Chaotic Dragonfly Algorithm (CDA) [39]. However, the chaotic weights of five behaviors are not able to balance exploration and exploitation ability in CDA. To overcome the drawback of premature convergence, a novel hybrid dragonfly algorithm combined with differential evolution (Hybrid DA-DE) for solving global optimization problems is proposed [41]. However, Hybrid DA-DE is not suitable for high-dimensional data.

The Hybrid feature selection algorithm has the advantages of both filter and wrapper algorithm [53]. The hybrid feature selection algorithm has two-stage. In the first stage, the filter algorithm is used to reduce the number of features and remove some irrelevant features. Forming candidate feature subsets is beneficial to improve computational efficiency in the next stage. In the second stage, the candidate feature subsets obtained in the first stage are used as inputs, and the wrapper is used to obtain the desired number of features and higher classification rate. To find the optimal feature sets, some hybrid feature selection algorithms are proposed. Zheng *et al.* propose a hybrid feature selection algorithm which uses improved mRMR and modified whale optimization algorithm [42]. Zheng *et al.* present a hybrid feature selection algorithm based on a new filter algorithm and the improved cuckoo search algorithm [43]. Unler *et al.* come up with a hybrid feature selection algorithm based on particle swarm optimization (PSO) and mutual information [44]. Akadi *et al.* put up forward a two-stage selection algorithm by MRMR and Genetic Algorithm (GA) [45]. A two-stage feature selection algorithm combining IG (Information Gain) approach and Binary Particle Swarm Optimization (BPSO) is proposed [46]. However, the subsets obtained from the above filter algorithms are always fixed combination and the diversity of population initialization is not rich enough. Thus, it is prone to premature convergence at local optimum in the above algorithms.

To sum up, three shortcomings are not considered in the studies mentioned above. Firstly, the subsets are fixed combination, because low-ranking features that associate with the best fitness have no chance to be randomly selected into candidate subset in mRMR. Hence, mRMR is easy to fall into a local optimum. Secondly, DA utilizes fixed weights to adjust swarming factors, thus the importance of local and global search capabilities is equal in the whole search process. The exploration capability in the early stage and exploitation capability in the later stage are not guaranteed. Therefore, fixed weights are not able to balance local and global optimum capabilities. Lastly, the best fitness of the current dragonfly and the best fitness of dragonflies are not taken into consideration in the position update of DA, so DA fails to guide dragonflies to potential candidate solutions that are associated with the best fitness. Therefore, the exploitation ability of DA is weak.

To address the above problems, we propose a novel feature selection algorithm named hybrid improved dragonfly algorithm (HIDA). Firstly, to avoid falling into a local optimum, features with small weight are randomly selected into candidate subset with a small probability which is decreased with iterations in mRMR. Therefore, the diversity of population initialization of IDA is rich. Secondly, to balance local and global optimum capabilities, dynamic weights which is consist of the separation weight, the alignment weight, the cohesion weight and the enemy weight are decreased with iterations. Four weights adjust the exploration ability of DA. The exploration capability in the early stage and exploitation capability in the later stage are guaranteed by dynamic weights. Lastly, to enhance the exploitation ability of DA, the concept of quantum optimal solution of current dragonfly and optimal solution of the current population are presented. Therefore, potential candidate solutions that are associated with the best fitness have a greater probability to be stored for further exploitation stage.

To sum up, three corresponding improvements are presented in HIDA as follows.

1. The diversity of candidate subset is enriched to avoid falling into a local optimum.

2. To balance global and local capability of HIDA, dynamic weights of swarming factors are used to adjust weights of four behaviors.

3. To enhance the exploitation capability of HIDA, the concept of quantum pbest and gbest are presented.

The rest of our paper is organized as follows: In section 2, we introduce the basic theories of our research. In section 3, we come up with a hybrid improved dragonfly algorithm called HIDA. In section 4. We evaluate the performance obtained with three improvements. In section 5, a conclusion is illustrated in detail.

II. THEORETICAL BACKGROUNDS

In this section, we introduce a filter algorithm and a wrapper algorithm. A filter algorithm is mRMR and a wrapper algorithm is DA.

A. DRAGONFLY ALGORITHM

Dragonfly algorithm (DA) is inspired by the hunting and migration mechanisms of idealized dragonflies. The hunting mechanism is known as static swarm behavior in which the dragonflies search for food sources over a small area by the formation of a small group of dragonflies. The migration mechanism is known as dynamic swarm behavior which is characterized by a massive number of dragonflies flying in one direction over long distances. The static and dynamic swarming behaviors of dragonflies represent the exploration and exploitation phases of meta-heuristic optimization, respectively.

The behaviors of dragonflies follow the principles of separation, alignment, cohesion, attraction towards the food, and distraction from the enemies. To model the swarming behaviors of dragonflies, the mathematical position of DA can be explained as follows.

$$X_i = (x_i^1, x_i^d, \ldots, x_i^D) \tag{1}$$

where $i = 1, 2, 3 \ldots, N$, x_i^d corresponds to the position of the ith dragonfly in the dth dimension of the search space, D is the number of dimensions and N is the number of search agents.

Separation (S_i) represents the static collision avoidance of individuals from other neighboring search individuals, this behavior is mathematically modeled as Eq. (2).

$$S_i = -\sum_{j=1}^{N} X - X_i \tag{2}$$

Alignment (A_i) corresponds to the velocity matching of individuals to other neighboring search individuals, this behavior is mathematically modeled as Eq. (3).

$$A_i = \frac{\sum_{i=1}^{N} V_i}{N} \tag{3}$$

Cohesion (C_i) indicates to the individuals towards the neighboring center of the mass, this behavior is mathematically modeled as Eq. (4).

$$C_i = \frac{\sum_{i=1}^{N} X_i}{N} - X \tag{4}$$

where X_i and V_i correspond to the position and velocity of the ith individual, respectively. X refers to the position of the current individual and N denotes the number of neighboring individuals.

In the DA, the fitness and position of food source are supposed to be updated using the best candidate(attraction). In addition, the fitness and position of the enemy should be updated using the worst candidate(distraction). The attraction of the ith individual F_i towards the food source is mathematically modeled as Eq. (5). The distraction E_i of the ith individual from enemies is mathematically modeled as Eq. (6):

$$F_i = F_+ - X \tag{5}$$

$$E_i = E_+ + X \tag{6}$$

X is the position of the current individual, F_+ represents the position of the food source and E_+ represents the enemy's position.

DA uses two vectors to update the position of a dragonfly, the step vector (ΔX) and the position vector (X). The step vector represents the movement direction of dragonflies, which is similar to the velocity vector in PSO. The step vector is modeled as Eq. (7):

$$\Delta X_i^{t+1} = (sS_i^t + aA_i^t + cC_i^t + fF_i^t + eE_i^t) + wX_i^t \tag{7}$$

where t is the iteration counter, s is the separation weight, S_i^t is the separation of the ith individual, a is the alignment weight, A_i^t is the alignment of ith individual, c indicates the cohesion weight, C_i^t is the cohesion of the ith individual, f is the food factor, F_i^t is the food source of the ith individual, e is the enemy factor, E_i^t is the position of enemy of the ith individual and w is the inertia weight.

If dragonfly has at least one dragonfly in the neighborhood, the position of the dragonfly is updated as Eq. (8). If there is no dragonfly in the neighborhood radius, the position of the dragonfly is updated using Levy Flight equation as given in Eq. (9).

$$X_{t+1} = X_t + \Delta X_{t+1} \tag{8}$$

$$X_{t+1} = X_t + levy(d)X_t \tag{9}$$

where d is the dimension of position vectors.

The Levy flight is calculated in Eq. (10).

$$levy(x) = 0.01 \times \frac{r_1 \times \sigma}{|r_2|} \tag{10}$$

where r_1 and r_2 are two random numbers in [0,1], β is a constant (equal to 1.5 in this work), and σ is calculated in

Eq. (11).

$$\sigma = \left[\frac{\Gamma(1+\beta) \times \sin(\frac{\pi\beta}{2})}{\Gamma(\frac{1+\beta}{2}) \times \beta \times 2^{(\frac{\beta-1}{2})}} \right]^{1/\beta} \tag{11}$$

where $\Gamma(x) = (x-1)!$

B. mRMR

Peng proposed Maximum relevance minimum redundancy (mRMR) filter method in 2015 [47]. The mRMR method uses Mutual information (MI) which is used to calculate feature correlation and redundancy in two phases. In the first phase, the MI is applied to calculate the relevancy between the label and the feature. In the second phase, the redundancy between each of the two features is calculated. The mathematical position of mRMR can be explained as follows.

$$X = (x_1, x_i, \ldots, x_D) \tag{12}$$

where x_i is a feature of feature set X, and D is the number of features.

$$S = (s_1, s_2 \ldots, s_M) \tag{13}$$

where s_j is a selected feature of X, and M is the number of selected features.

$$R = (r_1, r_2, \ldots, r_W) \tag{14}$$

where r_m is a remained feature of X, W is the number of remained features and $D = M + W$.

First, the mRMR method calculates the relatancy (Rl) between the label and the feature, with the largest mutual information value of feature being selected.

$$Rl = \frac{1}{M} \left(\sum_{j=1}^{M} I\left(S_j, C\right) + I(r_m, C) \right) \tag{15}$$

where $I(s_i, C)$ and $I(r_m, C)$ are the mutual information value between label (C) and feature.

$$C = (C_1, C_n, \ldots, C_H) \tag{16}$$

where C_n is the label of X, and H is the number of label.

Then, the mRMR method measures the redundancy (Rd) between every two features.

$$Rd = \frac{1}{M^2} \sum_{j=1}^{M} I(S_j, r_m) \tag{17}$$

where $I(S_j, r_m)$ is the mutual information value between every two features.

Finally, the mRMR method selects the feature of maximum relevancy minimum redundancy (mRMR) from the remaining feature sets.

$$mRMR = Rl - Rd = \frac{1}{M}(\left[\sum_{j=1}^{M} I(S_j, C) + I(r_m, C) \right] - \frac{1}{M} \sum_{j=1}^{M} I(S_j, r_m)) \tag{18}$$

Because $\sum_{j=1}^{M} I(S_j, C)$ and $\frac{1}{M}$ is constant, the formula simplified into formula(19).

$$mRMR = I(r_m, C) - \frac{1}{M} \sum_{s_j \in S, r_m \in R} I(s_j, r_m) \tag{19}$$

III. THE PROPOSED ALGORITHM

A. IMPROVED DRAGONFLY ALGORITHM (IDA)

1) QUANTUM PBEST AND GBEST

In DA, all the positions of dragonflies are updated according to Eq. (8) and Eq. (9). This position updating mechanism leads to premature convergence because the dragonflies are not allowed to keep track of previously obtained potential solutions. Therefore, in order to overcome the aforementioned limitation of the DA, pbest and gbest are introduced in IDA. pbest is the best fitness value obtained so far by a dragonfly while gbest is the best fitness value obtained so far by all dragonflies in the neighborhood. Besides, IDA integrates quantum physics to update the position of the dragonflies. During each iteration, the fitness value of a dragonfly is compared with the pbest value in current population. Better fitness value is saved to pbest. The best fitness value obtained so far by all dragonflies is saved to gbest value. A dragonfly updates its position as shown in the following formula (20).

$$\begin{aligned} \Delta X_i^{t+1} = & (sS_i^t + aA_i^t + cC_i^t + fF_i^t + eE_i^t) \\ & + wX_i^t + C_1 r_1 (MP_i^t - X_i^t) \\ & * \ln(1/u) + C_2(1 - r_1)(P_G - X_i^t) \end{aligned} \tag{20}$$

$$MP_i^t = \frac{1}{N} \sum_{i=1}^{N} p_i^t \tag{21}$$

$$r_1, u \sim (0, 1) \tag{22}$$

where C_1 and C_2 represents the cognitive and social parameters respectively, C_1 and C_2 are set to 2 by default, P_i^t and P_G represents the best fitness of the i th dragonfly and the best fitness of the swarm upto t th iteration respectively. N is the number of instances.

2) ADAPTIVE PARAMETERS

DA utilizes fixed parameters to adjust swarming factors. Therefore, DA can't balance local and global search capabilities. In order to overcome this shortage, dynamic curves are proposed to tune parameters of swarming factors. In the early optimization stage, IDA explores a huge search space to avoid premature convergence. In the later optimization stage, IDA efficiently exploits small regions to refine the final solutions. Because too many parameters of IDA need to be adjusted, the factors with the same change trend are designed to the same curve distribution. The parameters (the separation weight s, the alignment weight a, the cohesion weight c and the enemy factor e are mathematically modeled in eq. (23). Fig. 2 demonstrates the values of weights vary with iterations. First, the weights are set at a relatively high value and declines slowly to prevent blinding follow-up and falling into local

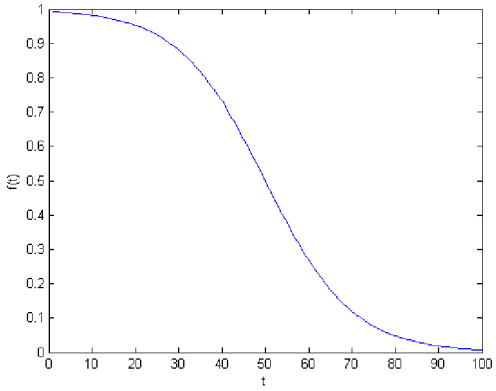

FIGURE 2. Adaptive parameters.

optimum. Then the curve rapidly goes down to slow position in order to balance exploration and exploitation ability. Last, the parameters decrease slowly to enhance exploitation ability.

$$f(t) = Init(1 - \frac{1}{1 + e^{-0.1(t-50)}}) \tag{23}$$

where t is the number of iterations and *Init* is the initialization of four parameters.

Fig. 3 illustrates the process of IDA. t is the number of iterations and T is the max number of iterations. First, we calculate classification accuracy of position vectors using SVM and update food source, enemy, pbest and gbest. Second, the separation weight s, the alignment weight a, the cohesion weight c and the enemy factor e are decreased adaptively using formula (23). Third, Δx is updated using all parameters in formula (20). Finally, position vectors are updated based on their classification accuracy rate.

B. HYBRID IMPROVED DRAGONFLY ALGORITHM (HIDA)

The integration of the filter algorithm (mRMR) and the wrapper algorithm (IDA) leads to a Hybrid Improved Dragonfly Algorithm (HIDA). mRMR selects features with maximum relevancy minimum redundancy, the subsets are always fixed combinations. Low-ranking features have no chance to be selected into candidate subset. It has been recognized that the combinations of individually good features do not necessarily lead to good classification performance. Thus, a dynamic number of features are selected for HIDA. From Fig.4, the x-coordinate is the order of features from mRMR, the y-coordinate is the probability of the feature which will be selected. If the probability is greater than ε, the feature is selected to initialization of IDA. The top-ranking feature has higher probability of being selected. Each dragonfly in the population initialization of IDA is the subset which is obtained from mRMR. The diversity of population ensures that at least one subset in the population is able to contain all interacting features which are associated with label. The

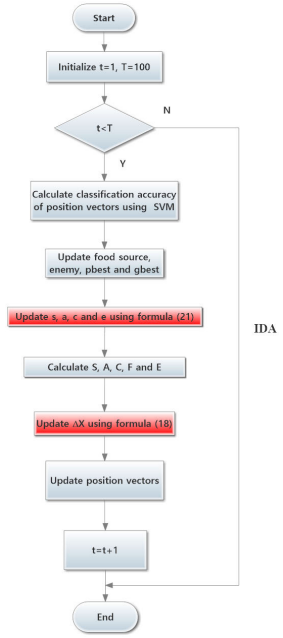

FIGURE 3. The flowchart of IDA.

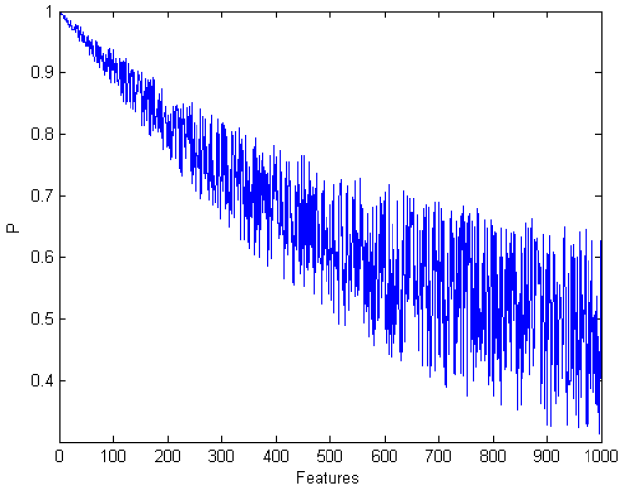

FIGURE 4. Adaptive selected probability *p*.

mathematical formulas are formulated in eq. (24) and (25).

$$p = -(1 + rand) * \frac{1}{\pi} \arctan(2 * x/D) + 1 \tag{24}$$

where p is the selected probability, x is the feature set which is obtained by mRMR and D is the number of features.

$$\begin{cases} if\ p > \varepsilon, & x(i) = 1 \\ if\ p > \varepsilon, & x(i) = 0 \end{cases} \tag{25}$$

The larger dimensionality of dataset is, the larger ε is.

In Fig.5, Every solution is characterized as a single dimensional vector, and the length of the vector is the number of features in the dataset. Every cell of the vector contains two values (1 or 0). Value 1 depicts that the corresponding feature

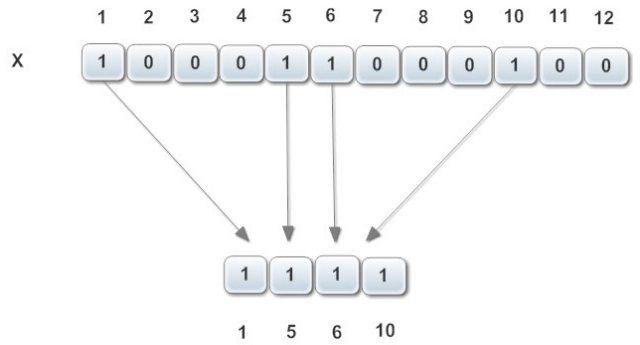

FIGURE 5. Adaptive selected probability.

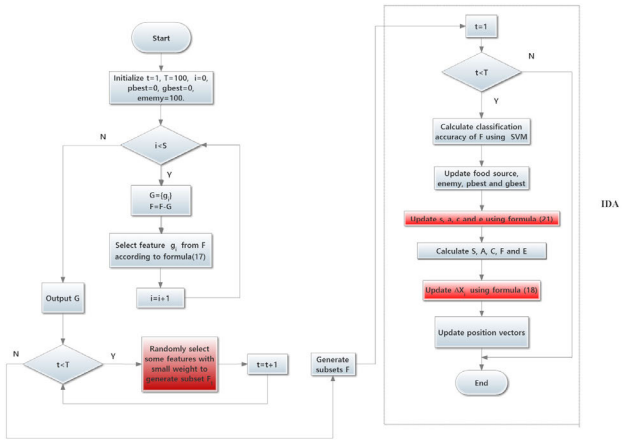

FIGURE 6. The flowchart of HIDA.

is chosen while value 0 represents that the feature is not selected.

Fig. 6 illustrates the process of IDA. First, all variables are initialized. t is the number of iterations, T is the max number of iterations, S is the number of selected features, G is the feature set of selected features, F is the feature set of remained features. Second, we use mRMR to construct a feature subset which has maximum relevancy minimum redundancy. Thirdly, Low-ranking features are randomly selected to a feature subset with small probability using formula (23). Finally, given a subset, IDA is applied to generate a promising candidate subset. The first judgment of "t < T" is used to generate candidate subsets for mRMR, and second judgment of "t < T" is used to control the number of iterations for IDA.

IV. EXPERIMENT RESULTS

In this section, we perform comprehensive experiments to compare the HIDA algorithm with BBA [48], BDA [25], CDA [39], LBPSO [40], MPMDWOA [42] and MSMCCS [43] methods on ten different gene expression datasets and eight UCI datasets.

A. DATASETS DESCRIPTION AND PREPROCESSING

To evaluate the usefulness of the HIDA approach, we carried out experiments on five gene expression datasets [49] and eight UCI datasets from the UCI Machine Learning Data Repository [50], [51]. The characteristics of datasets are summarized in Table 1.

TABLE 1. Benchmark data.

Dataset	N	Num	#C	Abbr
Arrhythmia	452	279	13	Arrhythmia
Musk	6598	168	2	Musk
Molecular Biology	3183	61	3	Molecular
Zoo	101	17	7	Zoo
Ozone Level Detection	2536	73	2	Ozone
Secom	1567	590	2	Secom
Segment	2310	19	7	Segment
Sonar	208	60	2	Sonar
LUNG_Cancer	203	12600	5	LUNG
Leukemia1	72	7129	2	Leukemia1
Leukemia2	72	7129	3	Leukemia2
Lymphoma	45	4026	9	Lymphoma
DLBCL	77	7129	2	DLBCL
Colon	62	2000	2	Colon
9-Tumors	60	5726	26	9-Tumors
11-Tumors	174	12533	11	11-Tumors
14-Tumors	308	15009	9	14-Tumors
Prostate_Tumors	102	10509	2	Prostate

N—number of instances, Num—number of features, #C—number of classes, Abbr-abbreviation of datasets

TABLE 2. Parameter settings.

Algorithm	Parameter	Value	Reference
BDA	Separation weight,	0.1	[25]
	Alignment weight	0.1	
	Cohesion weight	0.7	
	Food factor	1	
	Enemy factor	1	
BBA	Loudness	1.5	[48]
	Pulse rate	0.5	
	The maximum value of frequency	1	
	The minimum value of frequency	0	
MPMDIWOA	Maximum of wormhole existence probability	1	[42]
	Minimum of wormhole existence probability	0.2	
CDA	A constant β	1	[39]
LBPSO	c_1, c_2	[2.2, 2.2]	[40]
	c_r	0.2	
	M_u	0.2	
MSMCCS	Discovery rate of alien eggs/solution	0.25	[43]
	Levy exponent and coefficient	1.5	
	Step size	0.01	
HIDA	initialization of s	0.1	
	initialization of a	0.1	
	initialization of c	0.7	
	initialization of f	1	
	initialization of e	1	
	initialization of w	0.9	

B. PARAMETER SETTINGS

The parameter setting is introduced in Table 2 according to the literatures. The number of particles for five algorithms is set to 50, and the maximum number of iterations is set to 100.

The radial base function (RBF) is used as the kernel function of the SVM model [52]. In the following experiments, cross validation is made up of 10 folds. 9 folds are used for training and the last one is used for testing. All of the codes are implemented in MATLAB R2018b and run on a Window 10 PC with Intel Core i5-3470K 3.20 GHz CPU and 16 GB RAM.

According to the initialization of six parameters of HIDA, four parameters (s, a, c and e are decreased with the iteration number in formula (23) while two parameters (w and f) are increased with iteration number in formula (26). Furthermore, we present a threshold ε. ε is adjusted depending on the dimensionality of dataset in formula (27). For low-dimension dataset, setting a small value to enrich the diversity of candidate subset. For high-dimension dataset, setting a large value to save running time.

$$f(t) = -Init(1 - \frac{1}{1 + e^{-0.1(t-50)}}) \tag{26}$$

where t is the number of iterations and $Init$ is the initialization of two parameters (w and f).

$$\varepsilon = 1 - 0.5 * \text{rand} * \frac{1}{D} \tag{27}$$

where D is the number of features.

C. EXPERIMENT RESULTS

The experiments are tested through implementing BBA, DBA, CDA, LBPSO, MPMDWOA, MSMCCS, and HIDA using MATLAB. Due to the efficiency of SVM algorithm, we adopt SVM in HIDA. Four criteria are used including Acc, average selected features, AUC and F indicator. For balanced dataset, we often use the Acc as the criterion. But Acc is not enough for imbalanced problems. AUC is an indicator used to measure the classification performance in imbalanced data. F indicator is a comprehensive measure for balancing recall and precision. The Acc, AUC and F indicator are computed by the following four formulas.

$$Acc = \frac{TP + TN}{TP + FN + TN + FP} \tag{28}$$

$$AUC = \frac{1 + TP_{rate} - FP_{rate}}{2} \tag{29}$$

$$F_\beta = \frac{(1 + \beta^2) * TP_{rate} * PP_{rate}}{\beta^2 * PP_{rate} + TP_{rate}} \tag{30}$$

$$TP_{rate} = \frac{TP}{TP + FN} \tag{31}$$

$$PP_{rate} = \frac{TP}{TP + FP} \tag{32}$$

$$FP_{rate} = \frac{FP}{TN + FP} \tag{33}$$

where TP is the number of positive instances which are classified correctly and FP is the number of negative instances which are misclassified. β is a coefficient which is used to adjust the important degree of recall compared with precision. β is usually set to 1, and it means that recall and precision are important equally.

TABLE 3. The average classification accuracy rate (%) obtained from three algorithms.

Dataset	BDA		IDA		HIDA	
	Acc	NumF	Acc	NumF	Acc	NumF
Arrhythmia	72.98	187.6	74.17	194.8	**74.77**	**169.4**
Musk	98.94	91.0	99.12	87.4	**99.77**	**16.3**
Moleculor	99.85	12.5	99.85	12.3	**99.91**	**4.0**
Zoo	91.78	7.7	93.42	6.8	**98.00**	**5.3**
Ozone	97.15	57.1	**97.16**	48.4	97.12	**5.4**
Secom	93.42	454.7	93.43	387.4	**93.43**	**19.2**
Segment	97.97	15.2	98.18	13.7	**98.25**	**12.3**
Sonar	91.83	51.8	92.70	47.4	**92.81**	**45.3**
LUNG	98.04	257.0	98.04	293.7	**98.26**	**13.4**
Leukemia2	99.12	56.1	99.35	117.4	**100**	**19.0**
Leukemia3	87.45	500.0	95.44	397.8	**99.71**	**25.3**
Lymphoma	98.80	214.4	98.80	214.4	**98.89**	**15.5**
DLBCL	95.54	246.3	96.67	179.6	**100**	**16.0**
Colon	87.26	76.8	93.54	77.6	**98.83**	**10.2**
9-Tumors	60.24	342.1	65.67	267.8	**91.54**	**24.2**
11-Tumors	90.23	658.3	91.78	734.5	**95.30**	**167.4**
14-Tumors	47.74	497.0	70.27	395.3	**76.62**	**56.7**
Prostate	92.00	672.3	94.27	394.5	**99.21**	**20.5**

NumF—Number of selected features

TABLE 4. The average classification accuracy rate (%) obtained from seven algorithms.

Dataset	BBA	BDA	CDA	LBPSO	MPMDWOA	MSMCCS	HIDA
Arrhythmia	72.24	72.98	73.45	73.52	74.17	72.36	**74.77**
Musk	95.45	98.94	97.71	96.85	95.58	99.63	**99.77**
Moleculor	99.85	99.85	99.88	99.85	99.88	99.89	**99.91**
Zoo	90.04	91.78	93.37.	92.85	93.42	93.18	**98.00**
Ozone	97.15	97.15	97.12	97.12	97.15	**97.16**	97.12
Secom	93.42	93.42	93.42	93.42	93.42	93.42	**93.43**
Segment	97.81	97.97	98.03	98.18	98.19	98.03	**98.25**
Sonar	92.70	91.83	92.23	92.64	92.79	92.31	**92.81**
LUNG	97.57	98.04	97.73	98.05	95.33	84.57	**98.26**
Leukemia2	98.66	99.12	98.85	99.03	99.25	87.32	**100**
Leukemia3	88.24	87.45	95.44	96.36	98.75	77.15	**99.71**
Lymphoma	98.00	98.80	95.78	96.88	95.83	83.22	**98.89**
DLBCL	93.57	95.54	94.64	96.63	97.50	93.57	**100**
Colon	87.26	90.27	93.73	93.55	95.47	96.56	**98.83**
9-Tumors	60.24	58.57	62.14	65.34	65.51	78.23	**91.54**
11-Tumors	90.23	90.47	90.56	91.67	94.52	94.32	**95.30**
14-Tumors	47.74	69.34	70.25	70.47	74.27	71.37	**76.62**
Prostate	92.00	92.00	94.16	92.15	94.12	94.53	**99.21**

From Table 3, it is clear that HIDA shows the best performance among all the methods on all datasets except Ozone. HIDA has higher accuracy rate compared to the other methods. Similarly, the average selected features of HIDA is lower than other methods. Thus, the proposed subset generation improves HIDA a lot. Besides, Acc and NumF of IDA outperform BDA on most of the datasets. Take 9-Tumors for instance, Acc and NumF of IDA are 65.67 and 267.8 respectively, while Acc and NumF of BDA are 60.24 and 342.1 respectively.

Table 4 shows the Acc of seven algorithms in 18 datasets. Table 5 and Table 6 show and the average number of selected features and max number of selected features respectively.

TABLE 5. The average selected features obtained from seven algorithms.

Dataset	BBA	BDA	CDA	LBP SO	MPMD WOA	MSMC CS	HIDA
Arrhythmia	209.8	187.6	195.4	213.4	206.2	**150.4**	169.4
Musk	121.0	91.0	86.5	76.4	78.0	**3.2**	16.3
Moleculor	28.0	12.5	21.5	17.9	8.8	29.6	**4.0**
Zoo	7.2	7.7	6.9	5.8	6.5	8.2	**5.3**
Ozone	66.0	57.1	48.5	52.7	55.3	41.0	**5.4**
Secom	411.5	454.7	377.5	364.3	434.5	297.7	**19.2**
Segment	15.6	15.2	12.6	13.5	**9.1**	14.4	12.3
Sonar	45.8	51.8	56.7	49.5	48.5	54.2	**45.3**
LUNG	378.2	257.0	289.4	237.4	134.3	71.5	**13.4**
Leukemia2	251.0	56.1	127.8	115.6	135.6	33.4	**19.0**
Leukemia3	557.0	500.0	358.2	378.1	160.0	49.5	**25.3**
Lyphoma	299.6	214.4	193.4	175.3	158.1	53.2	**15.5**
DLBCL	246.1	246.3	168.2	179.1	153.3	57.5	**16.0**
Colon	79.4	76.8	84.3	70.4	23.7	20.3	**10.2**
9-Tumors	527.8	342.1	187.2	234.2	40.25	34.6	**24.2**
11-tumors	746.2	658.3	486.3	396.2	268.2	245.1	**167.4**
14-Tumors	486.8	497.0	289.3	276.2	59.2	58.2	**56.7**
Prostate	612.6	672.3	412.5	378.4	83.9	87.3	**20.5**

TABLE 6. The maximum selected features obtained from seven algorithms.

Dataset	BBA	BDA	CDA	LBP SO	MPMD WOA	MSM CCS	HIDA
Arrhythmia	231	217	228	232	228	**172**	184
Musk	153	108	93	82	85	**5**	21
Moleculor	34	23	27	22	11	34	**4**
Zoo	9	9	9	7	8	9	**6**
Ozone	70	60	51	56	58	47	**7**
Secom	432	472	396	387	453	325	**23**
Segment	17	16	14	14	**10**	15	13
Sonar	58	54	58	52	53	55	**47**
LUNG	432	314	327	285	176	82	**15**
Leukemia2	308	68	156	135	164	50	**19**
Leukemia3	623	523	386	402	194	54	**28**
Lyphoma	341	253	218	196	184	62	**17**
DLBCL	309	253	184	192	163	65	**16**
Colon	86	81	92	75	27	23	**12**
9-Tumors	594	387	214	278	48	39	**26**
11-Tumors	924	678	523	427	289	296	**185**
14-Tumors	583	505	306	294	63	65	**63**
Prostate	765	684	435	412	94	91	**23**

The performance of HIDA is better than the other six methods. In Table 4, it can be observed that *Acc* of HIDA is superior to other algorithms in all datasets except Ozone Level Detection. The of Ozone Level Detection is lower than the best result while the number of average selected features is superior to MSMCCS which gets the best performance among five algorithms. In Table 5, we can find that MPMDWOA, MSMCCS and HIDA get one, two and fifteen best average selected features separately. In DLBCL dataset, HIDA provides 100% accuracy by using 19 attributes only, at the same time MPMDWOA show 97.5% accuracy and 153.3 attributes. The result shows that HIDA outperforms other algorithms in both classification accuracy and selected attributes for the same dataset.

In addition to *Acc* and average selected features, Tables 7 and 8 show F_1 and *AUC* of seven algorithms in 18 datasets. HIDA outperforms all algorithms in 15 and 16 datasets in both F_1 and *AUC* respectively. Inspecting Table 5, It can be analyzed that HIDA achieves superior F_1 results in fifteen out of eighteen datasets. In Table 8, BDA does not outperform HIDA over most datasets in terms of *AUC*,

TABLE 7. The average F_1 result obtained from seven algorithms.

Dataset	BBA	BDA	CDA	LBP SO	MPMD WOA	MSMC CS	HIDA
Arrhythmia	0.83	0.85	0.84	0.84	**0.89**	0.83	0.84
Musk	0.86	0.92	0.91	0.90	0.90	0.98	**0.99**
Moleculor	0.97	0.98	**0.99**	**0.99**	0.97	0.98	**0.99**
Zoo	0.94	0.89	0.90	0.89	0.91	0.90	**0.96**
Ozone	0.22	0.21	0.24	0.25	0.33	0.33	**0.40**
Secom	0.16	**0.24**	0.18	0.18	0.18	0.18	0.18
Segment	0.98	0.90	0.98	0.98	0.84	0.94	**0.99**
Sonar	0.81	0.89	0.87	0.88	0.83	**0.91**	**0.91**
LUNG	0.74	0.78	0.74	0.78	**0.94**	0.61	0.61
Leukemia2	0.98	0.98	0.98	0.98	0.99	0.81	**1**
Leukemia3	0.96	0.97	0.97	0.97	0.97	0.59	**0.98**
Lymphoma	0.95	0.90	0.90	0.92	0.90	0.50	**0.98**
DLBCL	0.83	0.92	0.90	0.94	0.83	0.85	**1**
Colon	0.84	0.89	0.94	0.94	0.94	**0.98**	**0.98**
9-Tumors	0.52	0.46	0.51	0.54	0.54	0.58	**0.85**
11-Tumors	0.83	0.83	0.85	0.85	0.88	0.88	**0.91**
14-Tumors	0.34	0.38	0.45	0.48	0.68	0.71	**0.73**
Prostate	0.76	0.82	0.80	0.79	0.86	0.86	**0.91**

TABLE 8. The average *AUC* result obtained from seven algorithms.

Dataset.	BBA	BDA	CDA	LBP SO	MPMD WOA	MSMC CS	HIDA
Arrhythmia	0.51	0.52	0.54	0.53	0.49	0.50	**0.58**
Musk	0.97	0.92	0.91	0.91	0.90	0.98	**0.99**
Moleculor	0.97	0.98	0.98	0.97	0.97	0.98	**0.99**
Zoo	0.60	0.89	0.91	0.90	0.91	0.91	**0.97**
Ozone	0.56	0.56	0.55	0.55	0.60	0.60	**0.63**
Secom	0.55	0.57	0.58	**0.59**	0.55	0.55	0.55
Segment	0.95	0.90	0.91	0.92	0.85	0.92	**0.99**
Sonar	**0.97**	0.95	0.95	0.96	0.95	0.91	0.91
LUNG	0.80	0.96	0.85	0.92	0.95	0.65	**0.98**
Leukemia2	0.97	0.98	0.97	0.98	0.99	0.81	**1**
Leukemia3	0.95	0.97	0.97	0.97	0.97	0.68	**0.98**
Lymphoma	0.98	0.98	0.92	0.94	0.98	0.86	**0.99**
DLBCL	0.50	0.80	0.76	0.82	0.67	0.79	**1**
Colon	0.76	0.84	0.89	0.90	0.92	0.94	**0.98**
9-Tumors	0.51	0.47	0.52	0.54	0.57	0.65	**0.87**
11-Tumors	0.87	0.87	0.88	0.90	0.92	0.92	**0.94**
14-Tumors	0.41	0.45	0.58	0.58	0.63	0.63	**0.65**
Prostate	0.88	0.88	0.90	0.88	0.90	0.90	**0.99**

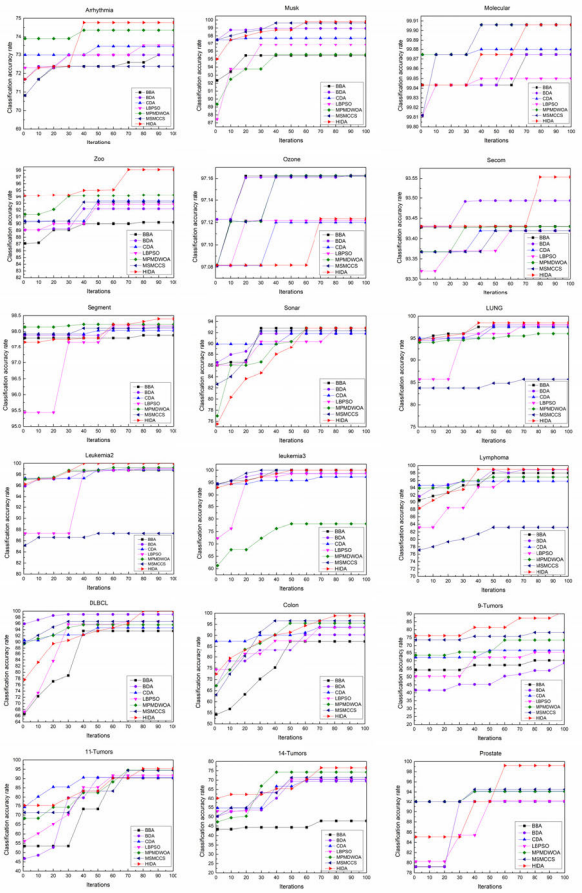

FIGURE 7. Typical convergence curves of the Seven algorithms on 18 datasets.

whereas it performs better than others on Secom datasets with an insignificant difference. Moreover, it is worth noting that HIDA outperforms other algorithms over fifteen datasets.

The computational complexity of HIDA is $O(T^*(D^2 + N^*D + N^*S))$. T represents the maximum number of iterations, N represents the number of populations, D represents the number of features in the data set, and S represents the time taken to execute the SVM classifier. In the table, N^*D represents the computational complexity of the wrapper algorithm position update in one iteration, N^*S represents the computational complexity of the SVM classifier in one iteration, and $O(D^2)$ represents the computational complexity of the filter algorithm in one iteration. The computational complexity of MPMDWOA and MSMCCS are the same as HIDA. BBA, BDA, CDA, LBPSO are wrapper algorithms, the computational complexity of four algorithms is $O(T^*(N^*D + N^*S))$. Therefore, the computational complexity of the hybrid algorithm is higher than the wrapper algorithm.

Additionally, in order to better validate the superior convergence behavior of HIDA on test datasets, Fig.7 is provided to demonstrate the convergence behavior of five algorithms on 18 datasets. By directly drawing curves of classification accuracy rate with the iterative number, we can see the classification accuracy rate is increased monotonously

TABLE 9. The comparison based on Wilcoxon signed-rank test on Ozone data set.

ALGORITHM1	HIDA	HIDA	HIDA	HIDA	HIDA	HIDA
ALGORITHM2	BBA	BDA	CDA	LBPSO	MWOA	MCS
Z	-1.345	-2.345	-2.675	-1.078	-1.267	-2.803
P	0.042	0.037	0.002	0.013	0.021	0.005

MWOA: MPMDWOA MCS: MSMCCS

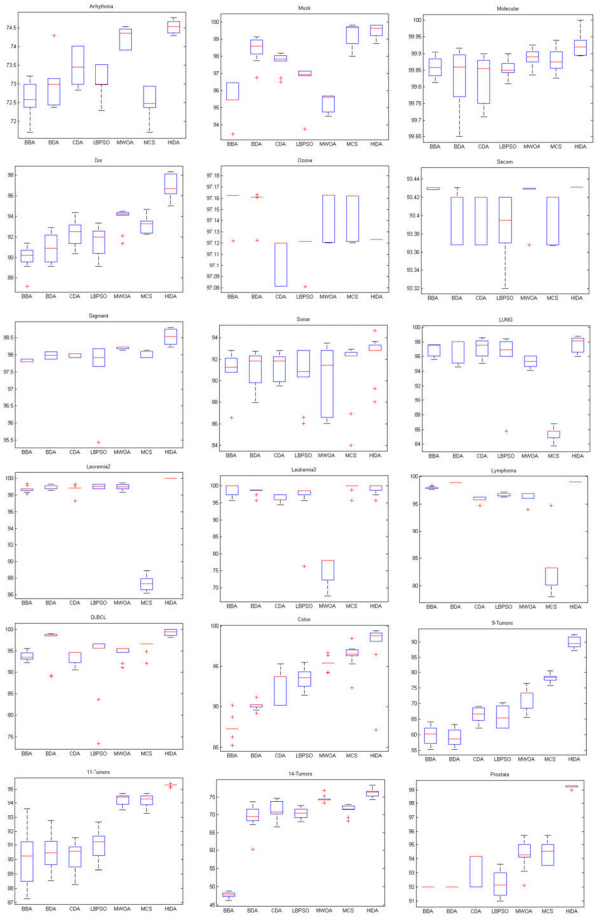

FIGURE 8. Box plots for classification accuracies of the Seven algorithms on 18 datasets.

in each iteration until level off. As it can be observed from these figures that HIDA converges fast compared to other algorithms and achieves best classification accuracy rate in most datasets. The results demonstrate the strong exploitation ability of HIDA in the later stage of optimization.

To further assess HIDA, boxplots for classification accuracies are obtained in comparison with the other methods. These plots are shown in Fig.8. The plots show that HIDA is better than the rest of the methods considered in most of the cases.

Table 9 shows that six pairs of Wilcoxon signed-rank tests are made on Ozone data set. With the significant level 0.05, it can be detected the performance of HIDA is statistically significant compared to BBA, BDA, CDA, LBPSO, MPMDWOA and MSMCCS.

D. DISCUSSION

According to the above observations, it can be concluded that the performance of HIDA is superior to the other algorithms for feature selection. The main reason for the good performance of HIDA is the integration of mRMR and IDA. On the one hand, IDA is able to enhance the exploitation and exploration capabilities. One the other hand, the proposed subset generation enriches population diversity and helps HIDA jump out of local optimum.

Firstly, Acc and NumF of IDA outperform BDA on most of the datasets in Table 3. IDA has higher accuracy rate compared to the other methods, because adaptive quantum optimum enhances the exploitation capability of IDA. Secondly, due to high accuracy and minimized feature size selected of HIDA in Table 4 and Table 5, we can conclude that HIDA significantly enhances the exploration of DA. Lastly, by analyzing the results of F_1 and AUC in Table 7 and Table 8, HIDA outperforms all algorithms in 15 and 16 datasets in both F_1 and AUC respectively. We can say that random selection plays a complementary role in enhancing exploration in the HIDA algorithm besides adaptive quantum optimum which enhances exploitation. Moreover, F_1 and AUC indicators are appropriate to measure imbalanced classification results, thus HIDA has an excellent performance in resolving imbalanced classification problems.

Despite efficient results generated by HIDA, yet it maintains certain limitations because of the nature of the hybrid algorithm whose computational complexity is higher than the wrapper algorithm. Moreover, mRMR sequentially constructs the feature subset by including one feature at a time. As a result, the mutual information of a feature is calculated and used to weigh the feature based on the feature subset available. Therefore, there is a dependence on the feature subset construction sequence in mRMR, and mRMR is not able to achieve high classification accuracy rate.

V. CONCLUSION

Due to the low classification accuracy rate of DA, a novel HIDA optimization algorithm which is combined with mRMR is proposed for feature selection. Firstly, a novel subset generation with random selection improves the classification accuracy rate of HIDA. Secondly, each dragonfly is designed to keep track of its coordinates with the addition of internal memory in quantum space. The novel position updating mechanism is able to balance local and global search capabilities of HIDA. Lastly, to enhance the exploitation capability of DA, HIDA utilizes dynamic parameters to adjust swarming factors.

To demonstrate the effectiveness of HIDA, experiments have been conducted on 8 UCI and 10 microarray datasets. Compared with the other six algorithms, HIDA is able to escape from local optima. Although HIDA doesn't have the best performance in all datasets, the five criteria of HIDA are better than other algorithms in all microarray datasets. HIDA is more suitable for the high dimensional dataset. The results verify that the integration of IDA and mRMR improves HIDA significantly.

REFERENCES

[1] A. A. Raweh, M. Nassef, and A. Badr, "A hybridized feature selection and extraction approach for enhancing cancer prediction based on DNA methylation," *IEEE Access*, vol. 6, pp. 15212–15223, 2018.
[2] S. F. Crone, S. Lessmann, and R. Stahlbock, "The impact of preprocessing on data mining: An evaluation of classifier sensitivity in direct marketing," *Eur. J. Oper. Res.*, vol. 173, no. 3, pp. 781–800, Sep. 2006.
[3] H. Peng, C. Ying, S. Tan, B. Hu, and Z. Sun, "An improved feature selection algorithm based on ant colony optimization," *IEEE Access*, vol. 6, pp. 69203–69209, 2018.
[4] 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, Jan. 2018.
[5] W. Ke, C. Wu, Y. Wu, and N. N. Xiong, "A new filter feature selection based on criteria fusion for gene microarray data," *IEEE Access*, vol. 6, pp. 61065–61076, 2018.
[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] E. H. Houssein, M. E. Hosney, D. Oliva, W. M. Mohamed, and M. Hassaballah, "A novel hybrid harris hawks optimization and support vector machines for drug design and discovery," *Comput. Chem. Eng.*, vol. 133, Feb. 2020, Art. no. 106656.
[8] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.
[9] O. Reyes, C. Morell, and S. Ventura, "Scalable extensions of the ReliefF algorithm for weighting and selecting features on the multi-label learning context," *Neurocomputing*, vol. 161, pp. 168–182, Aug. 2015.
[10] E. H. Houssein, Y. Mina, and E. H. Aboul, "Nature-inspired algorithms: A comprehensive review," in *Hybrid Computational Intelligence: Research and Applications*. Boca Raton, FL, USA: CRC Press, 2019, p. 1.
[11] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, "The max-min hill-climbing Bayesian network structure learning algorithm," *Mach. Learn.*, vol. 65, no. 1, pp. 31–78, Oct. 2006.
[12] T. Stützle, "Iterated local search for the quadratic assignment problem," *Eur. J. Oper. Res.*, vol. 174, no. 3, pp. 1519–1539, Nov. 2006.
[13] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simmulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, May 1983.
[14] D. Simon, "Biogeography-based optimization," *IEEE Trans. Evol. Comput.*, vol. 12, no. 6, pp. 702–713, Dec. 2008.
[15] J. Xiao, X. Zhang, and J. Xu, "A membrane evolutionary algorithm for DNA sequence design in DNA computing," *Chin. Sci. Bull.*, vol. 57, no. 6, pp. 698–706, Feb. 2012.
[16] O. Montiel, O. Castillo, P. Melin, A. R. Díaz, and R. Sepúlveda, "Human evolutionary model: A new approach to optimization," *Inf. Sci.*, vol. 177, no. 10, pp. 2075–2098, May 2007.
[17] Y. Liu, G. Wang, H. Chen, H. Dong, X. Zhu, and S. Wang, "An improved particle swarm optimization for feature selection," *J. Bionic Eng.*, vol. 8, no. 2, pp. 191–200, Jun. 2011.
[18] A. Nazemi and M. Dehghan, "A neural network method for solving support vector classification problems," *Neurocomputing*, vol. 152, pp. 369–376, Mar. 2015.
[19] S. Kashef and H. Nezamabadi-pour, "An advanced ACO algorithm for feature subset selection," *Neurocomputing*, vol. 147, pp. 271–279, Jan. 2015.
[20] Y.-P. Chen, Y. Li, G. Wang, Y.-F. Zheng, Q. Xu, J.-H. Fan, and X.-T. Cui, "A novel bacterial foraging optimization algorithm for feature selection," *Expert Syst. Appl.*, vol. 83, pp. 1–17, Oct. 2017.
[21] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, "A comprehensive survey: Artificial bee colony (ABC) algorithm and applications," *Artif. Intell. Rev.*, vol. 42, no. 1, pp. 21–57, Jun. 2014.
[22] P. Mohapatra, S. Chakravarty, and P. K. Dash, "An improved cuckoo search based extreme learning machine for medical data classification," *Swarm Evol. Comput.*, vol. 24, pp. 25–49, Oct. 2015.
[23] A. G. Hussien, E. H. Aboul, H. H. Essam, B. Siddhartha, and A. Mohamed, "S-shaped binary whale optimization algorithm for feature selection," in *Recent Trends in Signal and Image Processing*. Singapore: Springer, 2019, pp. 79–87.
[24] A. G. Hussien, E. H. Houssein, and A. E. Hassanien, "A binary whale optimization algorithm with hyperbolic tangent fitness function for feature selection," in *Proc. 8th Int. Conf. Intell. Comput. Inf. Syst. (ICICIS)*, Dec. 2017, pp. 166–172.

[25] S. Mirjalili, "Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Comput. Appl.*, vol. 27, no. 4, pp. 1053–1073, May 2016.
[26] F. A. Hashim, E. H. Houssein, M. S. Mabrouk, W. Al-Atabany, and S. Mirjalili, "Henry gas solubility optimization: A novel physics-based algorithm," *Future Gener. Comput. Syst.*, vol. 101, pp. 646–667, Dec. 2019.
[27] F. A. Hashim, H. H. Essam, H. Kashif, S. M. Mai, and A. Walid, "A modified Henry gas solubility optimization for solving motif discovery problem," *Neural Comput. Appl.*, vol. 32, pp. 10759–10771, Nov. 2019.
[28] N. Neggaz, E. H. Houssein, and K. Hussain, "An efficient henry gas solubility optimization for feature selection," *Expert Syst. Appl.*, vol. 152, Aug. 2020, Art. no. 113364.
[29] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: A gravitational search algorithm," *Inf. Sci.*, vol. 179, no. 13, pp. 2232–2248, Jun. 2009.
[30] B. Alatas, "A novel chemistry based Metaheuristic optimization method for mining of classification rules," *Expert Syst. Appl.*, vol. 39, no. 12, pp. 11080–11088, Sep. 2012.
[31] A. Kaveh and M. Khayatazad, "A new meta-heuristic method: Ray optimization," *Comput. Struct.*, vols. 112–113, pp. 283–294, Dec. 2012.
[32] R. A. Formato, "Central force optimization: A new nature inspired computational framework for multidimensional search and optimization," in *Nature Inspired Cooperative Strategies for Optimization (NICSO)*. Berlin, Germany: Springer, 2008, pp. 221–238.
[33] M. Abdechiri, M. R. Meybodi, and H. Bahrami, "Gases brownian motion optimization: An algorithm for optimization (GBMO)," *Appl. Soft Comput.*, vol. 13, no. 5, pp. 2932–2946, May 2013.
[34] X.-F. Song, Y. Zhang, Y.-N. Guo, X.-Y. Sun, and Y.-L. Wang, "Variable-size cooperative coevolutionary particle swarm optimization for feature selection on high-dimensional data," *IEEE Trans. Evol. Comput.*, early access, Jan. 22, 2020, doi: 10.1109/TEVC.2020.2968743.
[35] Y. Zhang, D.-W. Gong, X.-Z. Gao, T. Tian, and X.-Y. Sun, "Binary differential evolution with self-learning for multi-objective feature selection," *Inf. Sci.*, vol. 507, pp. 67–85, Jan. 2020.
[36] Y. Zhang, S. Cheng, Y. Shi, D.-W. Gong, and X. Zhao, "Cost-sensitive feature selection using two-archive multi-objective artificial bee colony algorithm," *Expert Syst. Appl.*, vol. 137, pp. 46–58, Dec. 2019.
[37] Y. Zhang, X.-F. Song, and D.-W. Gong, "A return-cost-based binary firefly algorithm for feature selection," *Inf. Sci.*, vols. 418–419, pp. 561–574, Dec. 2017.
[38] S. R. K. S. and S. Murugan, "Memory based hybrid dragonfly algorithm for numerical optimization problems," *Expert Syst. Appl.*, vol. 83, pp. 63–78, Oct. 2017.
[39] G. I. Sayed, A. Tharwat, and A. E. Hassanien, "Chaotic dragonfly algorithm: An improved Metaheuristic algorithm for feature selection," *Int. J. Speech Technol.*, vol. 49, no. 1, pp. 188–205, Jan. 2019.
[40] N. Kushwaha and M. Pant, "Link based BPSO for feature selection in big data text clustering," *Future Gener. Comput. Syst.*, vol. 82, pp. 190–199, May 2018.
[41] M. Duan, H. Yang, B. Yang, X. Wu, and H. Liang, "Hybridizing dragonfly algorithm with differential evolution for global optimization," *IEICE Trans. Inf. Syst.*, vol. 102, no. 10, pp. 1891–1901, Oct. 2019.
[42] Y. Zheng, Y. Li, G. Wang, Y. Chen, Q. Xu, J. Fan, and X. Cui, "A novel hybrid algorithm for feature selection based on whale optimization algorithm," *IEEE Access*, vol. 7, pp. 14908–14923, 2019.
[43] Y. Zheng, Y. Li, G. Wang, Y. Chen, Q. Xu, J. Fan, and X. Cui, "A novel hybrid algorithm for feature selection," *Pers. Ubiquitous Comput.*, vol. 22, nos. 5–6, pp. 971–985, Oct. 2018.
[44] A. Unler, A. Murat, and R. B. Chinnam, "mr^2PSO: A maximum relevance minimum redundancy feature selection method based on swarm intelligence for support vector machine classification," *Inf. Sci.*, vol. 181, no. 20, pp. 4625–4641, Oct. 2011.
[45] A. El Akadi, A. Amine, A. El Ouardighi, and D. Aboutajdine, "A two-stage gene selection scheme utilizing MRMR filter and GA wrapper," *Knowl. Inf. Syst.*, vol. 26, no. 3, pp. 487–500, Mar. 2011.
[46] X. Zhao, D. Li, B. Yang, H. Chen, X. Yang, C. Yu, and S. Liu, "A two-stage feature selection method with its application," *Comput. Electr. Eng.*, vol. 47, pp. 114–125, Oct. 2015.
[47] S. Ramírez-Gallego, I. Lastra, and D. Martínez-Rego, "Fast-mRMR: Fast minimum redundancy maximum relevance algorithm for high-dimensional big data," *Int. J. Intell. Syst.*, vol. 32, no. 2, pp. 134–152, Feb. 2017.
[48] S. Mirjalili, S. M. Mirjalili, and X.-S. Yang, "Binary bat algorithm," *Neural Comput. Appl.*, vol. 25, nos. 3–4, pp. 663–681, Sep. 2014.
[49] A. Statnikov, C. F. Aliferis, I. Tsamardinos, D. Hardin, and S. Levy, "A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis," *Bioinformatics*, vol. 21, no. 5, pp. 631–643, Mar. 2005.
[50] C. L. Blake, C.J. Merz. (1998). *UCI Repository of Machine Learning Databases*. Accessed: Jun. 1, 2016. [Online]. Available: http://www.ics.uci.edu/~mlearn/
[51] S. D. Bay. (1999). *The UCI KDD Archive*. [Online]. Available: http://kdd.ics.uci.edu
[52] S. Paul, M. Magdon-Ismail, and P. Drineas, "Feature selection for linear SVM with provable guarantees," *Pattern Recognit.*, vol. 60, pp. 205–214, Dec. 2016.
[53] O. A. Alomari, A. T. Khader, M. A. Al-Betar, and Z. A. Alkareem Alyasseri, "A hybrid filter-wrapper gene selection method for cancer classification," in *Proc. 2nd Int. Conf. BioSignal Anal., Process. Syst. (ICBAPS)*, Jul. 2018, pp. 113–118.

XUETING CUI received the B.S. degree in software engineering from Jilin University, China, where she is currently pursuing the Ph.D. degree with the College of Computer Science and Technology. Her research interests include object detection and feature selection.

YING LI received the B.S., M.S., and Ph.D. degrees from Jilin University. From 2000 to 2006, she was an Associate Professor with the Department of Space Information Processing, Jilin University, where she has been a Professor in computer application technology, since 2006. She has published over 60 papers in journals and international conferences. Her research interests include big data, 3D visual modeling, 3D image processing, machine vision, and machine learning. She is a Fellow of the China Computer Federation.

JIAHAO FAN received the B.S. and master's degrees from the Computer Science and Technology College, Jilin University, Changchun, China, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree. His research interests include swarm intelligence algorithm, machine learning, image processing, data mining, and 3D data processing.

TAN WANG received the B.S. degree in journalism and communication from the Jilin University of Finance and Economics, Changchun, China, in 2017, and the M.S. degree from Northeast Normal University, in 2020. She holds a Part-Time with Space Technology (Jilin) Company Ltd. Her research interests include media technology ethics, network ecological, and feature selection.

YUEFENG ZHENG received the Ph.D. degree in computer application from the Jilin University, in 2019. He is currently with Jilin Normal University. His research interests include feature selection and object detection.