IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 1 February 2023, accepted 23 March 2023, date of publication 31 March 2023, date of current version 10 April 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3263584

RESEARCH ARTICLE

Threshold Binary Grey Wolf Optimizer Based on Multi-Elite Interaction for Feature Selection

HONGZHUO WU[1], SHIYU DU[2,3,4], YIMING ZHANG[2], QUAN ZHANG[1], KAI DUAN[1], AND YANRU LIN[1]

[1]Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315099, China
[2]Engineering Laboratory of Advanced Energy Materials, Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo 315201, China
[3]School of Materials Science and Engineering, China University of Petroleum (East China), Qingdao 266580, China
[4]Milky-Way Sustainable Energy Ltd., Zhuhai 519000, China

Corresponding author: Shiyu Du (dushiyu@nimte.ac.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 52250005, Grant 21875271, and Grant U20B2021; in part by the Entrepreneurship Program of Foshan National Hi-Tech Industrial Development Zone and Zhejiang Province Key Research and Development Program under Grant 2019C01060; in part by the ''Pioneer'' and ''Leading Goose'' Research and Development Program of Zhejiang under Grant 2022C01236; in part by the International Partnership Program of the Chinese Academy of Sciences under Grant 174433KYSB20190019; and in part by the Leading Innovative and Entrepreneur Team Introduction Program of Zhejiang under Grant 2019R01003.

ABSTRACT The traditional grey wolf algorithm is widely used for feature selection. However, within complex feature multi-dimensional problems, the grey wolf algorithm is prone to reach locally optimal solutions and premature convergence. In this paper, a threshold binary grey wolf optimizer based on multi-elite interaction for feature selection (MTBGWO) is proposed. Firstly, the multi-population topology is adopted to enhance the population's diversity for improving search space utilization. Secondly, an information interaction learning strategy is adopted for the update of sub-population elite wolf position (optimal position) via learning better position from other elite wolves; in order to improve the local exploitation ability of the sub-population. At the same time, the command of β and δ wolves (second and third best positions) for population position updates is removed. Finally, a threshold approach is employed to convert the continuous position of grey wolf individuals into binary one to apply in the feature selection problem. Further, The MTBGWO algorithm proposed in this paper is compared with the traditional binary grey wolf algorithm (BGWO), binary whale algorithm (BWOA), as well as some recently developed novel algorithms to exhibit its superiority and robustness. Totally 16 classification datasets, from the UCI Machine Learning Repository, are chosen for comparison. The Wilcoxon's rank-sum non-parametric statistical test is carried out at 5% significance level to evaluate whether the results of the proposed algorithms significantly differs from those of the other algorithms. In the experimental results for all datasets, the overall average accuracy of the MTBGWO algorithm is 94.7%, while the highest of the other algorithms is 92.8% and the selected feature subset is 25% of the total dataset. The MTBGWO algorithm selects much smaller subset of features than other algorithms. In terms of computational efficiency, the overall processing time of MTBGWO is 24.2 seconds, whereas HSGW is 44.1 seconds. The results reveal that the MTBGWO has shown its superiority in solving the feature selection problem.

INDEX TERMS Grey wolf optimizer, feature selection, binary optimizer, elite interaction strategy.

I. INTRODUCTION

With the development of information technology, the volume of data increases remarkably in recent years [1]. However, the growing archives inevitably contains redundant and irrelevant data. The massive useless information may retard valid data acquisition and thus reduce machine learning performance (classification or prediction). Therefore, feature selection is an essential data processing step [2]. In general, a subset of relevant features is selected from the original feature space

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

based on some evaluation criteria to eliminate the effects of redundant and cluttered data to improve the learning generalization ability. The existing feature selection techniques are mainly divided into filters and wrappers [3]. The filter method is based on the general representation of features, such as goal relevance, self-correlation, and divergence. The filtering method has the advantage of efficiency, but the disadvantage is that the filtered subset of features is not necessarily useful. In contrast to the filtering method, the wrapper method [4] filters features based on the effect of the training model, so the first step is to divide the training and test sets, and then search for an optimal subset of features that makes the model perform best on the test set in terms of metrics.

The main strategies of the wrapper approach for searching feature subsets are complete, random, and heuristic search. In a complete search, all possible combinations of feature subsets are traversed and the one with the best score is selected. No wonder, the complexity of this method is exponential (2^N, N is the number of features), which is computationally expensive and impractical in many cases. The random search method can be used as an alternative strategy, selecting multiple feature subsets and then selecting the feature subsets with high evaluation scores. However, the computational time required in random search for the optimal subset in a high-dimensional dataset is almost the same as the complexity of complete search. So heuristic search attracts attention. In heuristic search, heuristic information can guide the search and reduce the search space. Model score or feature weight can be used as heuristic information. In contrast to complete and random searches, the heuristic search can balance the efficiency and accuracy of feature selection.

Due to their strong adaptivity in exploration and exploitation, various metaheuristic algorithms have been suggested to solve feature selection problems [5], [6], [7]. These algorithms include the Differentiation Evolution algorithm (DE) [8], Artificial Bee Colony algorithm (ABC) [9], Genetic Optimization algorithm (GA) [10], Particle Swarm Optimization algorithm (PSO) [11], Harris Hawk Optimization (HHO) [12], simulated annealing algorithm (SA) [13] and Grey Wolf Optimization (GWO) [14], [15], [16] etc. Comparing with other heuristic algorithms, GWO algorithm has the advantages of fewer adjustable parameters, deprivation-free mechanism, and the ability to avoid the local optima. Therefore, it has been used in many research areas in the last years, such as network prediction [17], feature subset selection [18], [19], solving the dynamic economic load dispatch problem of the power system [20]. For the problem of feature selection, the solution can be represented as a vector of features with size n, which is the number of features and the vector items can be binary values with 1 (the feature is included) and 0 (the feature is not included). Hence, GWO starts with an initial random population of vectors holding randomly selected features. Then, using the exploration and exploitation capabilities, GWO can find the optimal subset of features.

However, GWO has some drawbacks in feature selection problems similar to other metaheuristic algorithms [21], [22]. The traditional GWO method is proposed to perform continuous optimization, but feature selection is a multi-objective combinatorial optimization problem, which means the grey wolf algorithm must be binarized. The binarization requires flipping of consecutive positions, leading to incomplete local exploitation, and thus may fall into local minima. Hence, poor performance in high-dimensional feature spaces is the main problem faced with feature selection. In this work, a threshold binary grey wolf optimizer based on multi-elite interaction is proposed to deal with these disadvantages in feature subset selection.

The main contributions of this article are summarized as follows:

1.The traditional S-shaped transfer function used by Grey Wolf optimizer for binarization is analyzed, and the threshold-based method is used to replace it.

2.A multi-subpopulation topology for global search is adopted, in which the elite wolf (best position) learns other excellent positions by swapping to promote information exchange among subpopulations and bring diversity into their subpopulations.

3.The guiding rights of β wolf (second-best position) and δ wolf (third-best position) are removed so that the population is allowed to explore the elite wolf position more deeply and improves the local exploitation ability.

The rest of the work in this paper is organized as follows. Section II reviews recent works on binary grey wolf optimizer (BGWO) for feature selection. Section III shows the basic background of the BGWO algorithm and the binarization approach. The MTBGWO algorithm is described in detail in Section IV. Section V discusses the MTBGWO algorithm's experimental results and the comparison of other algorithms. Lastly, conclusions and future work are stated in Section VI.

II. RELATED WORK

GWO has been applied for different disciplines such as face recognition, gene selection, electromyography classification, diagnoses of diseases, interference detection systems, and feature selection [23]. Table 1 shows the current excellent binary optimization algorithms. Recently, Purushothaman et al. [24] proposed a hybrid GWO with Grasshopper Optimization Algorithm to select the best global optimum from the local optimum. Furthermore, the selected optima were clustered using the Fuzzy c-means (FCM) clustering algorithm. This algorithm minimized the computational time in text feature selection and text clustering. Emary et al. [25] suggested two different binary methods. In the first approach, individual steps toward the first three best solutions are binarized and then stochastic crossover is performed among the three basic moves to find the updated binary grey wolf position. In the second approach, sigmoidal function is used to squash the continuous updated position, then stochastically threshold these values to find the updated binary grey wolf position.

Binary WOA algorithm has been introduced in [26] to select the subset of features for wrapper feature selection and classification. Authors in [27] and [28] proposed a binary hybrid GWO and they used the KNN classifier. They have assessed the performance of their method by using eighteen standard benchmark datasets from the repository of machine learning.

In [29], a multi-strategy ensemble GWO is proposed to boost the precision and efficiency of the original GWO. Dhal and Azad [30] proposed a binary version of the hybrid two-phase multi-objective FS approach, based on PSO and GWO. In the first stage, the PSO performs a global search. Then a modified PSO and GWO based on Newton's second law of motion starts the local search from the global search results in the second stage. ln [31], [32], [33], and [34], the search capability and convergence speed are improved in this method. Tu et al. [35] presented an improved algorithm called hierarchy strengthened GWO, in which the elites are strengthened to study the strategy from the superiority wolf, preventing the low-ranking wolf's misleading. Then, a differential evolution (DE) strategy is applied to the ω wolf to avoid falling into local optimization. Emary et al. [36] proposed a binary variant of the Antlion Optimizer (AID). Ant Lion Optimizer (ALO) balances exploration and exploitation using a single operator that adaptively searches the solution domain to find the best solution.

Moreover, hybrid metaheuristic algorithms have received a lot of attention. Al-Wajih et al. [37] suggested exploration and exploitation of hybrid GWO-HHO balance will improve the performance of the search algorithm. In another study, Basak et al. [38] used PCA in combination with GWO to reduce calculation and ensure faster convergence. Hu et al. [21] introduction of covariance matrix adaptation evolution strategy and orthogonal learning strategy into GWO is adopted to accelerate the algorithm's convergence speed and to make the equilibrium characteristics more stable. Mafarja and Mirjalili [39] proposed two hybridization models for designing different feature selection techniques based on the Whale Optimization Algorithm (WOA). In the first model, the Simulated Annealing (SA) algorithm is embedded in the WOA algorithm, while in the second model, it is used to improve the best solution found after each iteration of the WOA algorithm.

With the range of values in the binary condition, Hu et al. [40] proposed a new update equation parameter to balance the ability of global search and local search. In [41], the authors established a time-varying transfer function after analyzing BPSO using existing transfer functions and identifying their drawbacks in the balance between exploration and exploitation. Nguyen et al. [42] used a dynamic parameter setting strategy to improve the searchability of BPSO and investigated whether and how the new momentum and velocity can help particles better explore the large and complex search space of feature selection and can produce smaller subsets of features with higher classification performance. The Khan et al. [43] proposed algorithm is based on modifying the salp swarm algorithm (SSA) using Levy Flight Distribution and spiral movement of particles to enhance the searching capabilities. Ebeed et al. [44] proposed Improved Lightning Attachment Procedure Optimization (ILAPO) to boost the search capability and avoid the stagnation of conventional LAPO. The algorithm is based on two improvements: i) Levy flight to enhance the exploration process, ii) Spiral movement of the particles to improve the exploitation process of the LAPO. Ma et al. [45] developed a multi-group optimization strategy in which decision variables are ranked according to their statistical significance determined during a limited number of initial PSO runs. This ranking divides the original set of high-dimensional decision variables into groups containing a finite number of decision variables. Chantar et al. constructed a binary grey wolf optimizer with elite-based crossover, which have a higher chance of jumping out of the local optimum. If their solution quality is low, they can achieve a high-quality solution instead of a low-quality average position as in the basic BGWO [46].

III. BASIC ALGORITHMS AND METHODS

A. GREY WOLF OPTIMIZER

Many intelligent optimization algorithms have been invented and applied to various optimization problems in recent years. GWO is a metaheuristic algorithm proposed by Faris et al. in 2014 [47]. It has attracted much attention in different optimization fields because the GWO requires few parameters to be adjusted and can achieve a good balance between global search and local exploration in a simple way to achieve effective convergence. The principle of GWO is as follows. A grey wolf pack has a strict hierarchy with four different levels of leadership: α, β, δ, and the rest called ω. The α wolf is the leading wolf, the β and δ wolf obey the α wolf and assist it in making decisions; the lowest level is the ω wolf, which generally needs to obey other grey wolves. GWO generally simulates three pack hunting behaviors in the algorithm: surrounding prey, chasing prey, and attacking prey.

1) SURROUNDING PREY

The grey wolf pack encircles the prey during the hunting process. Eq. 1 is utilized to mathematically model the distances between each search agent (wolf) and the prey. Grey wolf location is updated in Eq. 2.

$$\vec{D} = |\vec{C}\overrightarrow{Xp}(t) - \vec{X}(t)| \tag{1}$$

$$\vec{X}(t+1) = \overrightarrow{Xp}(t) - \vec{A} \cdot \vec{D} \tag{2}$$

where t represent the number of current iteration, $\vec{Xp}(t)$ and $\vec{X}(t)$ represent the position vector of the prey and α wolf, D represent the distance between the individual grey wolf and the prey, C and A are coefficient variables calculated as follows:

$$a = 2 - \frac{2t}{t_{\max}} \tag{3}$$

TABLE 1. The current excellent binary optimization algorithms.

Algorithm	Evaluation algorithm	Method	Application
GWO-GOA [24]	FCM	Wrapper	DS dataset
BGWO [25]	KNN	Wrapper	UCI dataset
BWOA [26]	KNN	Wrapper	UCI dataset
HSGW [27]	KNN	Wrapper	UCI dataset
BGWOPSO [28]	KNN	Wrapper	UCI dataset

$$\vec{C} = 2 \cdot r_2 \tag{4}$$
$$\vec{A} = 2\vec{a}r_1 - \vec{a} \tag{5}$$

where decays linearly from 2 to 0 during iteration, max is the maximum number of iterations of the algorithm, and r_1 and r_2 are random numbers between 0 and 1.

2) CHASING PREY

The grey wolf algorithm is guided by the three best solutions, α, β, and δ, for the population to pursue and besiege the prey. The α wolf position represents the optimal solution, and β and δ wolf are positioned closer to the prey than the rest grey wolves. Each search agent in the population updates its position around the prey randomly, as in Eq.6-12.

$$\vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}| \tag{6}$$
$$\vec{D}_\beta = |\vec{C}_2.\vec{X}_\beta - \vec{X}| \tag{7}$$
$$\vec{D}_\delta = |\vec{C}_3.\vec{X}_\delta - \vec{X}| \tag{8}$$
$$\overrightarrow{X_1} = \vec{X}_\alpha - \overrightarrow{A_1}.\vec{D}_\alpha \tag{9}$$
$$\overrightarrow{X_2} = \vec{X}_\beta - \overrightarrow{A_2} \cdot \vec{D}_\beta \tag{10}$$
$$\overrightarrow{X_3} = \vec{X}_\delta - \overrightarrow{A_3} \cdot \vec{D}_\delta \tag{11}$$
$$\vec{X}(t+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3} \tag{12}$$

where $\vec{X}_1$ $\vec{X}_2$, $\vec{X}_3$ represent the position updates of α, β, δ wolf and $\vec{X}(t+1)$ is the update of individual grey wolf position.

3) ATTACKING PREY

Wolves update their locations when attacking the prey between their current position and the prey's position so that $|A| < 1$. The α, β, δ wolves move in pursuit of the prey and unite when they strike the prey. $|A|$ that takes values above 1 or below -1 randomly to deviate the wolves from the prey. The diagram of grey wolf algorithm is shown in Fig.1.

Algorithm 1 presents continuous grey wolf optimization algorithm. It is worth mentioning that the updating of parameter a can assist in switching from exploration to exploitation trends.

B. BINARY GREY WOLF OPTIMIZATION ALGORITHM

The traditional GWO algorithm is used to solve continuous space problems. As the feature selection is a binary problem, a spatial mapping from real values to binary is required for

Algorithm 1 Continuous Grey Wolf Optimization Algorithm

Input: GWO population $\overrightarrow{X_i}$(i = 1, 2, . . ., n) with size n, maximum iterations number Max_iter, and fitness function F_n.

Output: $\overrightarrow{X_a}$: Optimal grey wolf position; F($\overrightarrow{X_a}$): Bestfitness value;

Find best, second best and third best individuals as $\overrightarrow{X_\alpha}$,$\overrightarrow{X_\beta}$,$\overrightarrow{X_\delta}$
while i<Max_iter **do**
  for each Wolf $\overrightarrow{X_i}$ in the population **do**
    Update current wolf's position according to
    Eq.12.
  end for
  Update $\vec{a}$, $\vec{A}$ and $\vec{C}$.
  Calculate the fitness function F_n for each $\overrightarrow{X_i}$
  Update $\overrightarrow{X_\alpha}$, $\overrightarrow{X_\beta}$, $\overrightarrow{X_\delta}$.
end while

GWO [48]. The grey wolf position dimension component is converted between 0 and 1, where 0 means the feature will not be selected and 1 means being selected. The sigmoid function is adopted whose role is to scale the continuous values to be 0 or 1.

The grey wolf position is constrained to the interval [0, 1] by the sigmoid function, and the transfer function converts the grey wolf position by Eq.13:

$$x_d^{t+1} = \begin{cases} 1, & \text{if sigmoid}\left(\dfrac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3}\right) \geq \text{rand} \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

x_d^{t+1} represents the binary update position in the d-dimension of t+1 iterations, $\overrightarrow{X_1}$, $\overrightarrow{X_2}$ and $\overrightarrow{X_3}$ represent the position updates of α, β, δ wolves, and the rand is the random number extracted between [0,1]. Eq.14 is the sigmoid function S(x):

$$sigmoid(x) = \frac{1}{1 + e^{(-10*(x-0.5))}} \tag{14}$$

IV. PROPOSED FEATURE SELECTION METHOD

This section will detail the threshold binary grey wolf optimizer based on multi-elite interaction for feature selection,

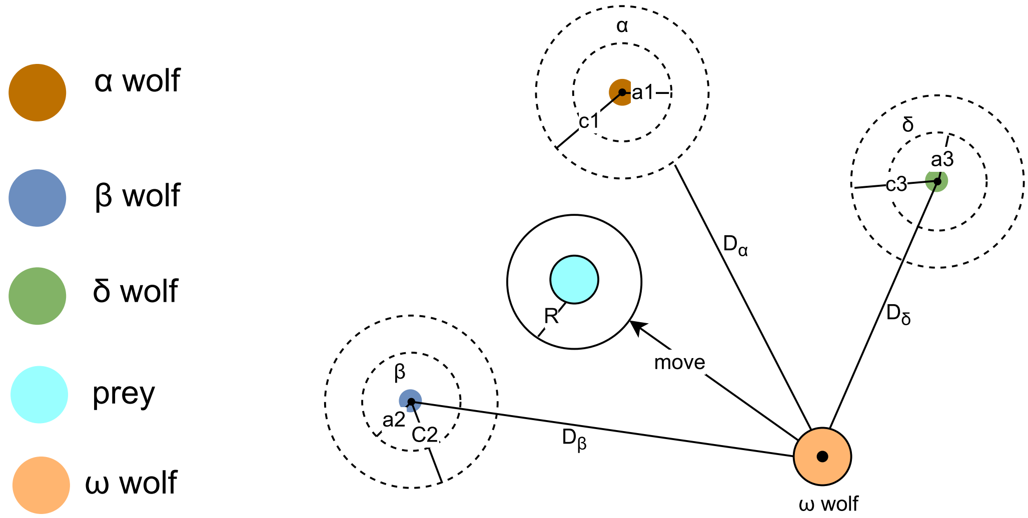


FIGURE 1. Position updating in the GWO algorithm.

referred to as MTBGWO. Also, the fitness function for measuring the solution quality of the GWO algorithm will be proposed.

A. THRESHOLD-BASED BINARIZATION

In the standard BGWO, the coordinate is determined without considering the previous position. The new position is determined by summing the components of the weighted distance vector to determine the probability of the corresponding position taking the value 1. [49]. As shown in Eq.13, the position of the grey wolf changes from 1 to 0 or from 0 to 1 through the sigmoid function. Multiple overturn lead to ineffective continuous exploration and development of grey wolf position in certain area. The algorithm may fall into local optimum.

Many scholars have contributed to balance global and local searches. The first approach focuses on designing new rules to update individual positions. For example, Banka and Dara [50] proposed a hamming distance-based binary particle swarm optimization (HDBPSO) for feature selection, classification, and validation. The hamming distance is used as an approximation for updating particle velocity in binary PSO. In another improved BPSO [51], a new position update rule is used to enhance the performance of the original BPSO for gene selection from microarray data.

In contrast, the second approach focused on replacing the sigmoid transfer function with new ones to update each particle's position and encourage better search space exploration. The V-shaped and linear normalized transfer function is one typical scheme proposed in [49], [52], and [53]. These transfer functions can promote more exploration than traditional sigmoid transfer functions.

In order to keep the position change of the wolf in continuous space, the threshold method is adopted for the binary grey wolf algorithm. The principle is to continue the traditional GWO location update mechanism so that the individual positions of grey wolves are allowed to continuously move within the range of [0,1], and then the position values are separated into 0 or 1 based on the threshold. It preserves the structure and position motion mechanism of the GWO algorithm, and makes the dimension of the individual positions of the wolf represent the selection of features or not. This scheme replaces the sigmoid function of position mapping, which is helpful to improve the local utilization of the algorithm. The individual position transformation of the grey wolf is shown in Fig.3.

X_1	X_2	X_3	....	X_{N-1}	X_N
0	1	1	....	1	0

FIGURE 2. The individual position transformation of grey wolf.

Fig.2 shows an example of a demonstration solution for feature selection, where X_N represents the grey wolf individual in N-dimension. The real value of individual grey wolf is within [0,1], and its position is converted to 1 or 0 by Eq.15, representing selected and unselected features.

$$x_d^{t+1} = \begin{cases} 1, & x_d^t \geq 0.5 \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

x_d^{t+1} represents the binary update position in the d-dimension of t+1 iterations, and 0.5 is the threshold.

B. MULTI-ELITE INFORMATION INTERACTION GREY WOLF OPTIMIZATION ALGORITHM

Obtaining a high-quality feature subset requires balancing the global and local search capabilities of the grey wolf algorithm. This paper introduces a multi-elite interactive grey wolf algorithm. Using Multi-Subpopulation topology, the position of the elite wolf is shared for information analysis and transmission, thus improving global search capabilities.

1) MULTI-ELITE INTERACTION TOPOLOGY

In the process of evolution, the whole population is divided into several small sub-populations based on the total number of individuals [54]. The best position of each sub-population is called elite wolf. Elite wolves improve searchability by facilitating information interaction. The positions of elite wolves in each subpopulation is updated by sharing the best positions of other subpopulations, and the vicinity of other best positions is explored; the position updates in Eq. 16-17.

$$\mathrm{Alpha}_i = \mathrm{mean}\left(\sum_{j=1, j\neq i}^{m} Alpha_j\right) * (1.0 + N(0,1)) \tag{16}$$

$$X_{\alpha\_i} = \begin{cases} X_{\alpha\_i}, & f\left(X_{\alpha\_i}\right) < f\left(\mathrm{Alpha}_i\right) \\ \mathrm{Alpha}_i, & \text{otherwise} \end{cases} \tag{17}$$

Here m is the number of sub-populations and mean () is to take the average value. N (0,1) is normally distributed random number with mean 0 and variance 1. Alpha_i represents the position of the elite wolf after position transformation. X_{α_i} represents the best position in the ith population. Thus, elite wolves learn the best position of other subgroups through interactive strategies to find better positions. When position in other subpopulations are better than their own, updating their own position by learning other position information effectively avoids falling into local optimum.

2) POSITION UPDATE

GWO is an optimization algorithm based on hierarchical social relations, in which α, β and δ wolves have different degrees of accuracy in the distance and direction from their prey. But α, β and δ wolves have the same ability to guide the motion of ω wolves. Unreasonable guidance will cause the convergence speed of the algorithm to slow down, making it easier to reach local optimization, and ultimately making the optimal solution output by the algorithm not the optimal solution required in the actual situation. Therefore, to avoid the interference of β and δ wolves, only α wolves (the best position) is used to guide the updating of wolf group positions. At the same time, the best position of the subgroup is updated by using the interactive information of the elite wolves in multiple subgroups, which greatly improves the reliability and adaptability of the best position. The specific formula is as follows:

$$X(t+1)_i = X_{\alpha\_i} - \vec{A} \cdot \vec{D}_{\alpha\_i} \tag{18}$$

$\mathrm{X(t+1)}_i$ represents the position of the ith sub-population of individual, X_{α_i} is the position of the ith sub-population α wolf, and $\vec{A}$ and $\vec{D}$ are calculated by Eq.5-6.

So, updating the population position by α wolf after the information interaction endues the algorithm with a better optimum finding ability and faster convergence speed.

Algorithm 2 Threshold Binary Grey Wolf Algorithm

Input: GWO population $\overrightarrow{X_i}$(i = 1, 2, . . ., n) with size n, number of subgroups m, maximum iterations number Max_iter, and fitness function F_n.
Output: $\overrightarrow{X_a}$: Optimal grey wolf position; F($\overrightarrow{X_a}$): Bestfitness value;

Initialize an agent of n wolves positions ∈ [0,1].
Individual grey wolves are divided into m groups
Find best, second best and third best individuals as $\overrightarrow{X_\alpha}$, $\overrightarrow{X_\beta}$, $\overrightarrow{X_\delta}$
while i<Max_iter **do**
  for each Wolf $\overrightarrow{X_i}$ in the population **do**
    Update current wolf's position according to $Eq.15$.
  end for
  for each Wolf $\overrightarrow{X_i}$ in the population **do**
    Calculate the fitness function F_n for each subgroups
    Update the position of the best agent in the each subgroup $\overrightarrow{X_\alpha}$ by using $Eq.16-17$.
  end for
  Update $\overrightarrow{a}$, $\overrightarrow{A}$ and $\overrightarrow{C}$.
  update individual position based $Eq.18$.
end while

3) FITNESS FUNCTION

Feature selection can be considered a multi-objective optimization problem, where the evaluation of the generated feature subset depends on two criteria: maximizing the classification accuracy as the primary objective and minimizing the number of selected features as the secondary objective. Eq.19 implements the conversion of multi-objective optimization into a single-objective optimization problem. In the iterative process, having the smallest fitness value is considered as the best solution.

$$\text{Fitness} = \varphi * \text{error} + \mu * \frac{|\text{ num\_feat }|}{|\text{ max\_feat }|} \tag{19}$$

φ and μ are the parameters corresponding to the classification accuracy and the number of feature subsets, φ ranges from [0,1], μ=1-φ, and error is the error rate of the classifier. The fitness function maximizes the percentage of correct classification rate at φ=0.99; $|num\_feat|$ / $|max\_feat|$ represents the ratio of the number of selected features to the total number of features.

V. EXPERIMENTAL RESULTS

In this section, a series of performance evaluations and comparisons are made between the proposed MTBGWO algorithm and other algorithms. Firstly, the threshold-based binarization method of this paper is compared with the traditional binarization to verify its superiority. Then MTBGWO algorithm is compared with the excellent traditional optimization algorithms (BGWO [25], BWOA [26]) in recent years.

TABLE 2. List of used datasets in the experiment.

	Dataset	Attributes	Instances	Classes
1	Ionosphere	34	351	2
2	Wine	13	178	3
3	Breast_Cancer	9	699	2
4	lymphography	18	148	2
5	Sonar	60	208	2
6	Heart	13	270	2
7	KrVsKpEW	36	3196	2
8	Exactly	13	1000	2
9	BreastEW	30	569	2
10	PenglungEW	325	73	2
11	Vote	16	300	2
12	LSVT	310	126	2
13	Arrhythmia	279	452	16
14	Zoo	16	101	6
15	SPECT	22	267	2
16	Waveform	40	5000	3

Finally, the advanced methods (HSGW [27], BGWOPSO [28]) are compared with the current algorithms in this paper.

A. EXPERIMENTAL SETUP

To verify the efficiency of the proposed algorithm MTBGWO. This experiment selected 16 datasets from the UCI database [55]. Table 2 describes the properties of the datasets in terms of the number of feature attributes, the number of instances and the number of classes. Parameters of the comparison algorithm are taken from the literature to ensure fair comparison between algorithms. Parameters of the proposed algorithm MTBGWO are set either according to domain-specific knowledge as in the case of φ, μ parameters, based on trial and error on small simulations, or common in the literature such as the rest of the parameters. Also, to be fair for comparison and to be able to obtain reliable statistical analysis, each algorithm was set the same number of iterations and number of population sizes and executed 30 times independently. Table 3 lists parameters setting for all the algorithms used in this work. A KNN classifier (k=5) was taken to evaluate the best solution in this study. The dataset was divided into K segments by K-fold cross-validation, one segment was used as a test set and the rest was used for the training set. The results are analyzed and compared based on the classification accuracy, average fitness, best fitness, worst fitness, average selected features size and average calculating time. Moreover, a statistical Wilcoxon's test is assessed in order to verify the difference of the accuracy results between MTBGWO, BGWO, BWOA, HSGW and BGWOPSO.

TABLE 3. Parameter settings of the algorithms used for comparison in the current study.

Algorithm	Paramter	Value
MTBGWO	a	[0,2]
	Maximum number of iterations	100
	Individual position range	[0,1]
	Binary threshold number	0.5
	The number of subgroups	3
	φ and μ in fitness function	0.99 ,0.01
BGWO	a	[0,2]
	Maximum number of iterations	100
BWOA	a	[0,2]
	b	1
	Maximum number of iterations	100
HSGW	Population size	20
	Dimension	Number of features
	Number of runs for each technique	30
	a in GWO and WOA	[0,2]
BGWOPSO	Search agents(n)	5
	R	[0,1]
	K-neighbors	5
	K-folds cross-validation	10

B. ASSESSMENT MEASURES

To evaluate the performance of the proposed method (MTBGWO), some measures are defined as follows:

1) THE MEAN ACCURACY

The algorithm accuracy is evaluated by the average accuracy of the subset of features selected as the most relevant in the classifier when the algorithm runs M times. It is calculated according to Eq.20.

$$\text{Average Accuracy} = \frac{1}{M}\sum_{k=1}^{M}\text{Accuracy}^{K} \tag{20}$$

2) THE MEAN FITNESS FUNCTION

The mean fitness function is the average of the fitness function obtained when the algorithm is run M times, which is calculated as follows:

$$\text{Mean fitness} = \frac{1}{M}\sum_{k=1}^{M} g_{*}^{k} \tag{21}$$

where g_{*}^{k} the value of the fitness function obtained at run k.

3) THE BEST FITNESS FUNCTION

The best fitness function is the minimum value of the fitness function obtained by M runs, calculated as follows.

$$\text{Best fitness} = \text{Min}_{K=1}^{M}\, g_{*}^{k} \tag{22}$$

where g_{*}^{k} the best value obtained at run k.

4) THE WORST FITNESS FUNCTION

The value of the worst fitness function refers to the maximum value of the fitness function obtained by running the algorithm M times and calculated as follows.

$$\text{Worst fitness} = \text{Max}_{K=1}^{M} g_*^k \tag{23}$$

where g_*^k the worst value obtained at run k.

5) AVERAGE SELECTED FEATURES SIZE

This metric refers to the average size of the selected features to the total number of features. When the algorithm runs M times, its average selection is calculated as in Eq.24:

$$\text{Average selection} = \frac{1}{M}\sum_{k=1}^{M}\frac{\text{Avg.size}^k}{T_f} \tag{24}$$

where Avg.size^k is the selected features at run k, and T_f shows the dataset's total number of features.

6) AVERAGE CALCULATING TIME

This indicator refers to the average calculation time in seconds. In the number of runs its average computation time is calculated as in Eq.25:

$$\text{Average time} = \sum_{k=1}^{M}\text{Avg.time}^k \tag{25}$$

where the average Calculating time spent when Avg.time^k is run k times.

7) WILCOXON's RANK SUM TEST

In order to better understand the importance of the technique under consideration, it is important to explain its impact from a statistical point of view. Therefore, the statistical Wilcoxon's ranksum test is usually used to verify the experimental results of meta-statistical computational methods. Wilcoxon's ranksum test is one of the non-parametric statistical tests used to statistically distinguish the performance of methods in competition [56]. The proposed MTBGWO was validated against each competing algorithm using this test. The Wilcoxon's rank sum test returned p-values, which helped to analyze the differences in the paired groups.

C. DISCUSSION

1) THE RESULT OF MTBGWO

Table 4 shows the average data of the proposed MTBGWO algorithm run 30 times on 16 datasets. The results include average accuracy, average fitness, the average number of selected features and computational time. The second highest accuracy rate in the Vote dataset is 99.1%. The average accuracy in the Breast_Cancer and KrVsKpEW datasets is also high at 98%. The selection of feature subsets also showed superiority in different datasets, of which 12 datasets have less than 10 feature subsets. Among them, Breast_Cancer, Heart, and Wine data have 3.3, 3.7, and 3.8 minimum feature subsets, respectively. In addition, the proposed feature selection method has less computation time (in seconds) on the following datasets: PenglungEW, Zoo, and LSVT with 2.0, 2.6 and 2.8 seconds, respectively. It can be concluded that the proposed model selects the small number of features within a reasonable time frame and has good results in terms of classification accuracy. Thus, the proposed MTBGWO algorithm can balance global search and local exploration during optimization iterations.

2) COMPARISON BETWEEN THRESHOLD BINARIZATION AND TRADITIONAL BINARIZATION

Table 5 compares the important performance metrics of the threshold binarization-based grey wolf algorithm (TBGWO) and the traditional binary grey wolf algorithm (BGWO). As seen from the table, TBGWO is significantly better than the BGWO algorithm for classification accuracy in 14 out of 16 datasets, which suggests that the threshold binarization approach positively affects algorithm optimization. In addition, TBGWO is significantly better than BGWO algorithm in feature selection for most datasets. BGWO is superior to TBGWO in only three small feature subsets. TBGWO is better than BGWO regarding classification accuracy and the number of selected features in medium and large datasets. The result shows that the threshold binarization approach benefits more for large and complex search spaces.

3) COMPARISON OF PROPOSED MTBGWO ALGORITHM WITH RELATED ALGORITHM

Tables 6 to 11 provide statistical comparisons of the proposed MTBGWO with other state-of-the-art algorithms in classification accuracy, average fitness, best fitness, worst fitness, selected feature size, and computation time. The best results are highlighted in bold. Table 6 shows that the accuracy of the proposed MTBGWO is higher than other methods in all datasets except Sonar, BreastEW and Vote. The best performance was achieved in Wine, Exactly, PenglungEW and Zoo datasets with 100% accuracy. According to the results in Table 10, the average number of feature subsets selected by MTBGWO and other methods in the dataset is shown. The proposed MTBGWO method selects fewer feature subsets than other algorithms in most of the datasets, and the number of feature subsets selected is equivalent to less than 25% of the total number of features. Examining the results in both Table 6 and Table 10, we can see that MTBGWO selects smaller subset of features while maintaining its good performance in classification. This proves that MTBGWO can search for optimized multi-objectives. Tables 7, 8, and 9 summarize the multiple statistical measures based on different algorithms running on different datasets. From these tables, the MTBGWO method has better average fitness than the BGWO, BWOA, HSGW and BGWOPSO algorithms on 13 datasets. The MTBGWO method outperforms other algorithms in terms of best fitness in 10 datasets and worse than other fitness in only three datasets. The results demonstrate the superiority of the proposed MTBGWO method showing that it can effectively balance global search and

TABLE 4. Results obtained by the proposed MTBGWO represented by Average (AVG) values and the corresponding standard deviation (SD).

	Dataset	Accuracy		Fitness		Feature select size		Computional time	
		AVG	SD	AVG	SD	AVG	SD	AVG	SD
1	Ionosphere	0.958	0.006	0.042	0.006	5.4	2.871	6.8	0.42
2	Wine	1.000	0	0.01	0.003	3.8	0.6	4.1	0.40
3	Breast_Cancer	0.980	0.005	0.027	0.005	3.3	1.187	9.6	0.65
4	lymphography	0.938	0.019	0.065	0.018	6.2	1.887	3.6	0.31
5	Sonar	0.962	0.011	0.039	0.01	9.3	2.002	4.8	0.20
6	Heart	0.909	0.011	0.093	0.012	3.7	1.269	4.7	0.06
7	KrVsKpEW	0.981	0.002	0.023	0.002	16.1	3.015	98.1	5.9
8	Exactly	1.000	0	0.005	0	6.1	0.3	16.3	1.4
9	BreastEW	0.968	0.007	0.032	0.006	4.1	1.375	9.0	0.45
10	PenglungEW	1.000	0	0	0	19.1	3.936	2.0	0.05
11	Vote	0.991	0.004	0.011	0.003	5.1	2.022	5.5	0.18
12	LSVT	0.900	0.016	0.099	0.016	4.5	3.748	2.8	0.06
13	Arrhythmia	0.753	0.007	0.245	0.007	35.7	15.107	8.8	0.37
14	Zoo	1.000	0	0.003	0	5.8	1.4	2.6	0.06
15	SPECT	0.968	0.007	0.035	0.006	6.5	4.006	3.9	0.20
16	Waveform	0.843	0.004	0.16	0.004	18.9	3.3	204.5	7.3

TABLE 5. Comparison of the average accuracy, average fitness function and average number of features selected for proposed TBGWO and BGWO.

	Dataset	TBGWO			BGWO		
		Accuracy	Fitness	feature_size	Accuracy	Fitness	feature_size
1	Ionosphere	**0.96**	0.041	7	0.955	0.046	**5.1**
2	Wine	**0.998**	0.005	3.9	0.993	0.01	**3.2**
3	Breast_Cancer	0.963	0.038	**3.6**	**0.967**	0.034	3.8
4	lymphography	**0.931**	0.064	**5.9**	0.927	0.076	6.8
5	Sonar	**0.962**	0.037	**12.9**	0.96	0.037	14.1
6	Heart	0.893	0.111	5.2	**0.894**	0.108	**4.2**
7	KrVsKpEW	**0.983**	0.02	**12.9**	0.981	0.023	13.2
8	Exactly	**0.98**	0.02	**5.5**	0.845	0.161	10.8
9	BreastEW	0.966	0.035	**3.6**	0.957	0.048	15.6
10	PenglungEW	**0.958**	0.043	**29.8**	0.941	0.064	180.2
11	Vote	**0.988**	0.014	**4.4**	0.978	0.028	11
12	LSVT	**0.839**	0.159	**17.3**	0.747	0.255	174.3
13	Arrhythmia	**0.723**	0.276	**48.1**	0.702	0.301	190
14	Zoo	**1**	0.003	**6**	0.977	0.029	9.9
15	SPECT	**0.965**	0.038	**6.2**	0.961	0.046	15.2
16	Waveform	**0.840**	0.16	**16**	0.839	0.167	36

local exploitation capabilities during optimization iterations. Another key aspect is the running time, and Table 11 shows the average running time (in seconds) for all algorithms in the dataset. The results show that the average running time is the least for the WOA and GWO methods and the longest for HSGW. This is because the hybrid methods embed more

TABLE 6. Comparison between average accuracy and corresponding standard deviation (SD) of proposed MTBGWO and related work methods.

	Dataset	MTBGWO		BGWO		BWOA		HSGW		BGWOPSO	
		AVG	SD	AVG	SD	AVG	SD	AVG	SD	AVG	SD
1	Ionosphere	**0.958**	0.006	0.955	0.015	0.922	0.022	0.943	0.011	0.940	0.006
2	Wine	1.000	0.000	0.993	0.021	0.961	0.023	1.000	0.006	1.000	0.009
3	Breast_Cancer	**0.980**	0.005	0.967	0.014	0.953	0.014	0.968	0.005	0.970	0.006
4	lymphography	**0.938**	0.019	0.927	0.014	0.869	0.036	0.907	0.009	0.910	0.030
5	Sonar	0.962	0.011	0.96	0.012	0.875	0.043	0.925	0.014	**0.981**	0.014
6	Heart	**0.909**	0.011	0.894	0.016	0.83	0.039	0.878	0.014	0.877	0.016
7	KrVsKpEW	**0.981**	0.002	**0.981**	0.004	0.966	0.006	0.973	0.004	0.97	0.003
8	Exactly	**1**	0	0.845	0.069	0.794	0.11	0.934	0.053	**1**	0
9	BreastEW	0.968	0.007	0.957	0.007	0.943	0.016	0.968	0.005	**0.971**	0.006
10	PenglungEW	**1**	0	0.941	0.029	0.932	0.047	0.982	0.022	0.991	0.018
11	Vote	0.991	0.004	0.978	0.012	0.958	0.016	0.997	0.008	**0.998**	0.003
12	LSVT	**0.9**	0.016	0.747	0.029	0.8	0.039	0.868	0.024	0.871	0.018
13	Arrhythmia	**0.753**	0.007	0.702	0.011	0.683	0.017	0.706	0.008	0.721	0.011
14	Zoo	**1**	0	0.977	0.015	0.945	0.025	**1**	0.015	0.991	0.016
15	SPECT	**0.968**	0.007	0.961	0.013	0.933	0.023	0.967	0.005	0.963	0.015
16	Waveform	**0.843**	0.004	0.839	0.004	0.829	0.007	0.839	0.003	0.83	0.008

TABLE 7. Comparison between mean fitness function of the proposed MTBGWO and related work method.

	Dataset	MTBGWO	BGWO	BWOA	HSGW	BGWOPSO
1	Ionosphere	**0.042**	0.046	0.079	0.057	0.061
2	Wine	0.000	0.010	0.043	0.000	0.001
3	Breast_Cancer	**0.020**	0.034	0.049	0.033	0.031
4	lymphography	**0.065**	0.076	0.134	0.097	0.100
5	Sonar	0.039	0.037	0.128	0.078	**0.018**
6	Heart	**0.093**	0.108	0.172	0.125	0.126
7	KrVsKpEW	**0.023**	**0.023**	0.041	0.034	0.033
8	Exactly	**0.005**	0.161	0.211	0.071	**0.005**
9	BreastEW	0.032	0.048	0.059	0.035	**0.031**
10	PenglungEW	**0.000**	0.064	0.068	0.019	0.010
11	Vote	0.011	0.028	0.006	0.004	**0.003**
12	LSVT	**0.099**	0.255	0.198	0.131	0.128
13	Arrhythmia	**0.245**	0.301	0.317	0.294	0.278
14	Zoo	**0.003**	0.029	0.058	0.004	0.005
15	SPECT	**0.035**	0.046	0.070	0.040	0.040
16	Waveform	**0.160**	0.167	0.177	0.167	0.172

operators to the extent that they add additional computational overhead. The proposed MTBGWO method is close to the runtime of the BGWO method, but it is desirable considering the trade-off between accuracy and runtime. It achieves high classification accuracy, and the running time is within a reasonable range.

TABLE 8. Best fitness function comparison between the proposed MTBGWO and related work methods.

	Dataset	MTBGWO	BGWO	BWOA	HSGW	BGWOPSO
1	Ionosphere	**0.029**	0.039	0.040	0.039	0.048
2	Wine	**0.000**	0.002	0.020	**0.000**	0.002
3	Breast_Cancer	**0.013**	0.090	0.030	0.020	0.018
4	lymphography	**0.027**	0.048	0.092	0.091	0.090
5	Sonar	0.032	0.033	0.065	0.049	**0.017**
6	Heart	**0.063**	0.076	0.112	0.104	0.090
7	KrVsKpEW	0.018	**0.017**	0.027	0.023	0.029
8	Exactly	0.005	0.038	0.005	0.005	0.005
9	BreastEW	0.024	0.028	0.028	0.024	**0.020**
10	PenglungEW	0.000	0.005	0.001	0.000	0.001
11	Vote	0.003	0.006	0.002	**0.001**	**0.001**
12	LSVT	**0.052**	0.187	0.105	0.078	0.104
13	Arrhythmia	**0.228**	0.274	0.285	0.279	0.252
14	Zoo	**0.000**	0.006	0.004	**0.000**	0.004
15	SPECT	**0.020**	0.026	0.039	0.027	**0.020**
16	Waveform	**0.151**	0.161	0.165	0.161	0.157

TABLE 9. Worst fitness function comparison between the proposed MTBGWO and related work methods.

	Dataset	MTBGWO	BGWO	BWOA	HSGW	BGWOPSO
1	Ionosphere	0.105	**0.096**	0.123	0.123	0.135
2	Wine	**0.076**	0.096	0.151	0.149	0.131
3	Breast_Cancer	**0.076**	**0.076**	0.085	0.077	0.083
4	lymphography	**0.179**	0.180	0.244	0.224	0.204
5	Sonar	**0.112**	0.128	0.193	0.191	0.205
6	Heart	0.222	**0.186**	0.283	0.257	0.235
7	KrVsKpEW	**0.035**	0.048	0.060	0.056	0.156
8	Exactly	**0.273**	0.306	0.311	0.311	0.311
9	BreastEW	**0.071**	0.082	0.089	0.082	0.082
10	PenglungEW	**0.136**	0.186	0.180	0.138	0.182
11	Vote	**0.079**	0.104	0.096	0.084	0.089
12	LSVT	0.443	0.500	0.313	**0.287**	0.394
13	Arrhythmia	**0.324**	0.351	0.358	0.351	0.347
14	Zoo	**0.068**	0.106	0.130	0.101	0.163
15	SPECT	**0.087**	0.090	**0.087**	**0.087**	0.089
16	Waveform	**0.177**	0.208	0.197	0.194	0.214

Fig.3 show boxplots of the accuracies for all datasets achieved by the optimizers. The proposed MTBGWO, BGWO, BWOA, HSGW and BGWOPSO. From each boxplot we can determine the first quartile (Q1), the third quartile (Q3), the lower and the higher values. In addition, the red line in the box indicates the median value.

TABLE 10. Comparison between average selected feature size of proposed MTBGWO and related work methods.

	Dataset	MTBGWO	BGWO	BWOA	HSGW	BGWOPSO
1	Ionosphere	5.400	5.100	4.500	**4.300**	5.900
2	Wine	3.800	**3.200**	5.500	6.100	4.400
3	Breast_Cancer	**3.300**	3.800	7.400	7.100	5.100
4	lymphography	6.200	6.800	7.800	7.500	**4.600**
5	Sonar	**9.300**	14.100	21.800	22.100	11.500
6	Heart	**3.700**	4.200	4.200	4.900	4.000
7	KrVsKpEW	16.100	**13.200**	26.500	27.400	**14.500**
8	Exactly	**6.100**	10.800	8.200	8.000	7.000
9	BreastEW	**4.100**	15.600	7.600	6.400	7.100
10	PenglungEW	**19.100**	180.200	31.700	48.900	35.500
11	Vote	5.100	11.000	**3.200**	3.800	4.300
12	LSVT	4.500	174.300	3.600	**3.400**	18.100
13	Arrhythmia	35.700	190.000	89.000	101.900	**25.500**
14	Zoo	**5.800**	9.900	6.400	6.600	6.000
15	SPECT	**6.500**	15.200	7.700	7.800	7.100
16	Waveform	18.900	36.000	30.500	30.800	**15.800**

TABLE 11. The average computational time (seconds) comparison between the proposed model and related work methods.

	Dataset	MTBGWO	BGWO	BWOA	HSGW	BGWOPSO
1	ionosphere	6.8	6.6	**3.4**	17.2	6.4
2	wine	4.1	3.4	**1.9**	16.3	5.6
3	Breast_Cancer	9.6	9.9	**4.7**	21.2	5.9
4	lymphography	3.6	3.2	**1.9**	16.1	5.3
5	sonar	4.8	4.4	**2.9**	16.6	5.4
6	heart	4.7	4.7	**2.6**	17.2	5.2
7	KrVsKpEW	98.1	97.3	**57.5**	164.3	65.6
8	Exactly	16.3	17.9	**9.3**	25.0	12.6
9	BreastEW	9.0	9.4	**4.4**	19.4	5.6
10	PenglungEW	**2.0**	2.1	2.4	20.4	4.9
11	Vote	5.5	5.3	**2.6**	16.8	5.7
12	LSVT	2.8	3.0	**2.4**	10.8	2.6
13	arrhythmia	8.8	12.2	**6.0**	6.4	5.9
14	zoo	2.6	2.3	**1.4**	11.5	5.2
15	SPECT	3.9	3.6	**1.8**	15.9	5.7
16	waveform	204.5	200.3	**117.3**	309.8	127.9

Note that the boxplots are drawn after running each algorithm 30 times, and they reflect the classification accuracy. From Fig.3 we can conclude that the MTBGWO has higher accuracy for all datasets except sonar, KrVsKpEW, BreastEW and Vote in comparison to other algorithms. Also, the median of the proposed algorithm has greater value

TABLE 12. P-values of the Wilcoxon rank sum test (p⩾ 0.05 are shown in bold face).

	MTBGWO vs.	BGWO	BWOA	HSGW	BGWOPSO
1	ionosphere	**6.45E-02**	9.77E-03	1.95E-03	1.95E-03
2	wine	**6.25E-01**	**1.22E-01**	1.95E-03	1.95E-03
3	Breast_Cancer	7.63E-03	**4.92E-01**	1.95E-03	1.95E-03
4	lymphography	3.91E-03	1.95E-03	1.95E-03	1.95E-03
5	sonar	**3.75E-01**	1.95E-03	1.95E-03	1.95E-03
6	heart	1.95E-03	**2.08E-02**	1.95E-03	1.95E-03
7	KrVsKpEW	1.95E-03	1.95E-03	1.95E-03	1.95E-03
8	Exactly	1.95E-03	1.95E-03	1.95E-03	1.95E-03
9	BreastEW	3.91E-03	**2.32E-01**	1.95E-03	**1.60E-01**
10	PenglungEW	1.95E-03	7.69E-03	1.95E-03	1.95E-03
11	Vote	1.95E-03	**3.22E-01**	1.95E-03	1.95E-03
12	LSVT	1.95E-03	1.95E-03	**1.39E-01**	1.95E-02
13	arrhythmia	1.95E-03	1.95E-03	1.95E-03	1.95E-03
14	zoo	1.95E-03	9.77E-03	1.95E-03	1.95E-03
15	SPECT	9.77E-03	3.89E-03	4.68E-03	**6.60E-01**
16	waveform	1.95E-03	1.95E-03	1.95E-03	1.95E-03

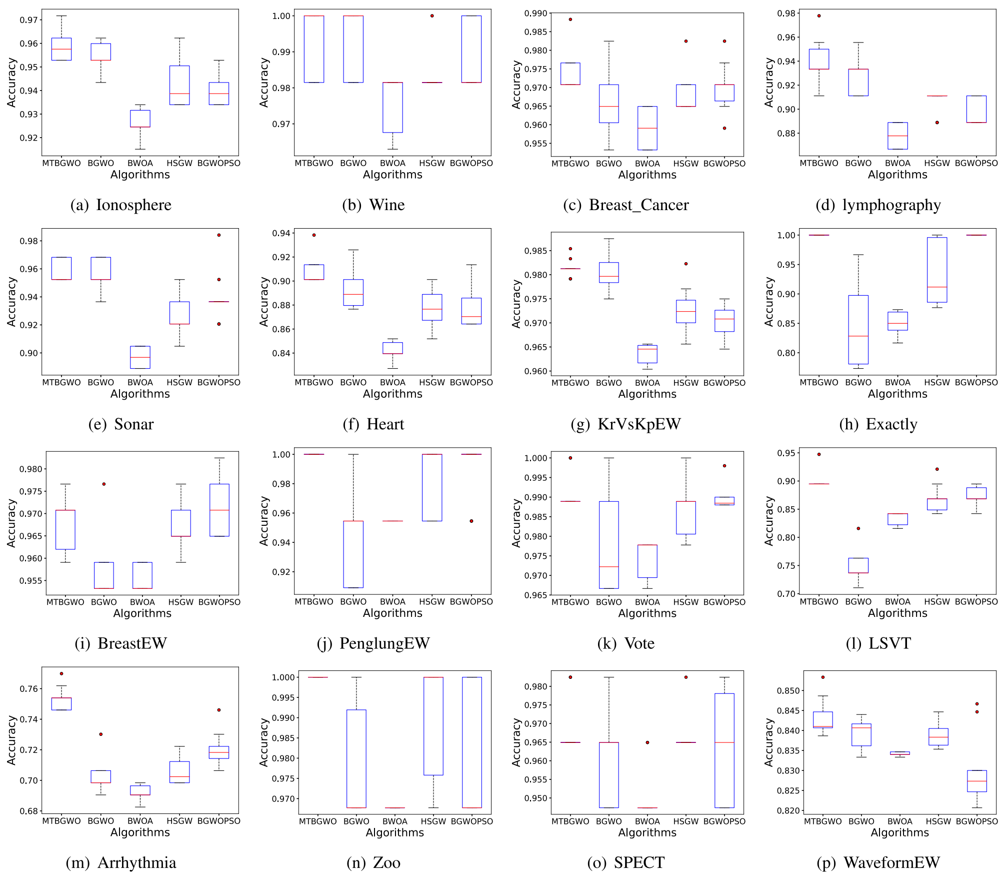
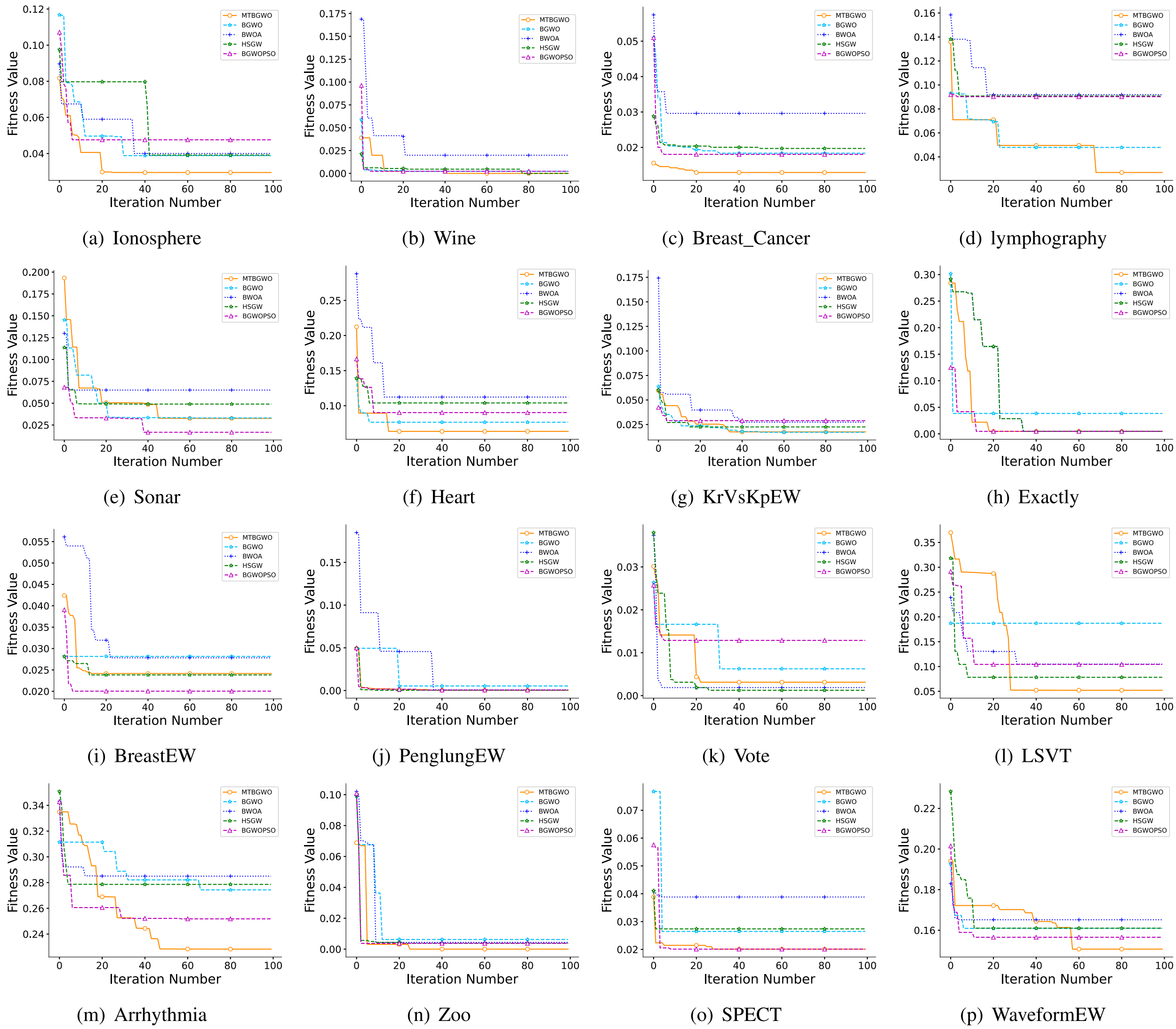


FIGURE 3. Boxplot of MTBGWO and other algorithms over all datasets.

compared to the remaining algorithms. Overall, these boxplots allow us to observe that MTBGWO is competitive and often superior on the majority of datasets. To verify these algorithms' convergence curves of their fitness functions



(a) Ionosphere (b) Wine (c) Breast_Cancer (d) lymphography
(e) Sonar (f) Heart (g) KrVsKpEW (h) Exactly
(i) BreastEW (j) PenglungEW (k) Vote (l) LSVT
(m) Arrhythmia (n) Zoo (o) SPECT (p) WaveformEW

FIGURE 4. Convergence curves for all compared approaches.

are recorded in Fig.4. MTBGWO performs optimally on most datasets, which especially evolves better subset of features with large dimensional features while having smaller convergence degree values. The convergence speed of the MTBGWO method is lower than that of other algorithms in a small number of datasets. Because the method uses multiple elite interactions for local development search, which increases the computational cost by multiple information swapping. However, at the same time, MTBGWO evolves better solutions than other methods in large and complex search spaces.

4) WILCOXON's RANKSUM TEST

According to the results of Table 12, MTBGWO algorithm is statistically significant fifteen datasets compared to HSGW because the p-value is less than 5%. In addition, for BGWOPSO, MTBGWO still statistically significative over fourteen datasets. On the other hand, BWOA, BGWO, and MTBGWO showed the same performance for over most dataset. At the meanwhile, it can be easily observed that in most comparisons, p-values obtained using the rank sum test were less than 0.05, which proves the superiority of MTBGWO algorithm to be statistically significant.

5) DISCUSSION

Summarizing the results of the seven experiments, the proposed model MTBGWO achieves better results in terms of accuracy, convergence adaptation value, and the number of features selected. The MTBGWO method has an average accuracy of 94.7% for all datasets, compared to the highest average accuracy of 92.8% for the total dataset of the comparison algorithms. Finally, in terms of the total number of features, the MTBGWO method selects subset of features about 25% of this dataset, compared to the GWO and WOA algorithms, which select almost twice as many features as the MTBGWO algorithm. According to the results, the number of total feature subsets selected by the MTBGWO method is smaller than other algorithms and the accuracy of the classification task is guaranteed, confirming the method's reliability and generality. In terms of computational time, the average time for MTBGWO on all datasets is 24.2 s, compared to 13.9 s for BWOA, 44.1 s for the other algorithms HSGW, and 17.2 s for BGWOPSO. Therefore, MTBGWO is more time-consuming than some conventional algorithms. Since the method takes multiple elite interactions to increase the local exploration capability, it leads to more computational resources in the search area. However, better convergence is obtained simultaneously, so the MTBGWO method provides a good compromise between effectiveness and efficiency.

VI. CONCLUSION AND FUTURE DIRECTION

This paper proposes threshold binary grey wolf optimizer based on multi-elite interaction for feature selection problems. To verify the effectiveness and efficiency of the proposed method, 16 standard UCI datasets are adopted for experiments and the algorithm performance is evaluated using five criteria. The results show that the proposed method effectively improves the performance of the BGWO algorithm compared with the traditional algorithm (BGWO, BWOA) and the advanced algorithm (BHSGWO, BGWOPSO). The new method (MTBGWO) outperforms the other algorithms in terms of accuracy and the number of selected feature subsets in most datasets. The method has advantages and disadvantages like other algorithms. The MTBGWO method uses multi-elite information interaction method, which improves the local exploitation capability but sometimes decreases the convergence speed. There is potential for further enhancement of the MTBGWO method. Additionally, the results of Wilcoxon's test also indicate that the improvement of the proposed algorithm is statistically significant compared to the other metaheuristics.

Furthermore, the proposed algorithm can be applied to more practical problems in real-world scenarios. For example, as a filter feature selection method, seeking to evaluate the generality of the selected features would be a valuable contribution. It can also solve weighted parameter optimization problems for neural networks, constraint engineering problems, etc. In the future work, we have a few ideas that can be investigated in addition to the work presented here:

1. Use enhanced initialization method that starts the optimization with solution closer to optimal.

2. Extend the proposed algorithm to work on parallel distributed mode to enhance convergence time.

3. Test the methodology on other big datasets besides those from UCI datasets.

REFERENCES

[1] M.-S. Chen, J. Han, and P. S. Yu, "Data mining: An overview from a database perspective," *IEEE Trans. Knowl. Data Eng.*, vol. 8, no. 6, pp. 866–883, Dec. 1996.
[2] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Comput. Elect. Eng.*, vol. 40, no. 1, pp. 16–28, Jan. 2014.
[3] U. M. Khaire and R. Dhanalakshmi, "Stability of feature selection algorithm: A review," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 4, pp. 1060–1073, 2019.
[4] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 273–324, 1997.
[5] A. Got, A. Moussaoui, and D. Zouache, "Hybrid filter-wrapper feature selection using whale optimization algorithm: A multi-objective approach," *Expert Syst. Appl.*, vol. 183, Nov. 2021, Art. no. 115312.
[6] A. P. Engelbrecht, J. Grobler, and J. Langeveld, "Set based particle swarm optimization for the feature selection problem," *Eng. Appl. Artif. Intell.*, vol. 85, pp. 324–336, Oct. 2019.
[7] H. Jia, Z. Jiang, Y. Li, and K. Sun, "Simultaneous feature selection optimization based on improved spotted hyena optimizer algorithm," *J. Comput. Appl.*, vol. 41, no. 5, p. 1290, 2021.
[8] A. Ghosh, S. Das, A. Chowdhury, and R. Giri, "An improved differential evolution algorithm with fitness-based adaptation of the control parameters," *Inf. Sci.*, vol. 181, no. 18, pp. 3749–3765, Sep. 2011.
[9] D. Karaboga and B. Gorkemli, "A quick artificial bee colony (qABC) algorithm and its performance on optimization problems," *Appl. Soft Comput.*, vol. 23, no. 1, pp. 227–238, Oct. 2014.
[10] D. Whitley, "A genetic algorithm tutorial," *Statist. Comput.*, vol. 4, no. 2, pp. 65–85, Jun. 1994.
[11] B. Wei, X. Xia, F. Yu, Y. Zhang, X. Xu, H. Wu, L. Gui, and G. He, "Multiple adaptive strategies based particle swarm optimization algorithm," *Swarm Evol. Comput.*, vol. 57, Sep. 2020, Art. no. 100731.
[12] S. Gupta, K. Deep, A. A. Heidari, H. Moayedi, and M. Wang, "Opposition-based learning Harris hawks optimization with advanced transition rules: Principles and analysis," *Expert Syst. Appl.*, vol. 158, Nov. 2020, Art. no. 113510. [Online]. Available: https://aliasgharheidari.com/
[13] J. Lee and D. Perkins, "A simulated annealing algorithm with a dual perturbation method for clustering," *Pattern Recognit.*, vol. 112, Apr. 2021, Art. no. 107713.
[14] X. Zhang, Q. Lin, W. Mao, S. Liu, Z. Dou, and G. Liu, "Hybrid particle swarm and grey wolf optimizer and its application to clustering optimization," *Appl. Soft Comput.*, vol. 101, Mar. 2021, Art. no. 107061.
[15] S. Gupta and K. Deep, "A novel random walk grey wolf optimizer," *Swarm Evol. Comput.*, vol. 44, pp. 101–112, Feb. 2019.
[16] M. Banaie-Dezfouli, M. H. Nadimi-Shahraki, and Z. Beheshti, "R-GWO: Representative-based grey wolf optimizer for solving engineering problems," *Appl. Soft Comput.*, vol. 106, Jul. 2021, Art. no. 107328.
[17] Y. Xing, J. Yue, C. Chen, Y. Xiang, Y. Chen, and M. Shi, "A deep belief network combined with modified grey wolf optimization algorithm for $PM_{2.5}$ concentration prediction," *Appl. Sci.*, vol. 9, no. 18, p. 3765, Sep. 2019.
[18] S. Arora, H. Singh, M. Sharma, S. Sharma, and P. Anand, "A new hybrid algorithm based on grey wolf optimization and crow search algorithm for unconstrained function optimization and feature selection," *IEEE Access*, vol. 7, pp. 26343–26361, 2019.
[19] J. Wang, D. Lin, Y. Zhang, and S. Huang, "An adaptively balanced grey wolf optimization algorithm for feature selection on high-dimensional classification," *Eng. Appl. Artif. Intell.*, vol. 114, Sep. 2022, Art. no. 105088.
[20] V. K. Kamboj, S. K. Bath, and J. S. Dhillon, "Solution of non-convex economic load dispatch problem using grey wolf optimizer," *Neural Comput. Appl.*, vol. 27, no. 5, pp. 1301–1316, Jul. 2016.
[21] J. Hu, H. Chen, A. A. Heidari, M. Wang, X. Zhang, Y. Chen, and Z. Pan, "Orthogonal learning covariance matrix for defects of grey wolf optimizer: Insights, balance, diversity, and feature selection," *Knowl.-Based Syst.*, vol. 213, Feb. 2021, Art. no. 106684.
[22] J. Hu, A. A. Heidari, L. Zhang, X. Xue, W. Gui, H. Chen, and Z. Pan, "Chaotic diffusion-limited aggregation enhanced grey wolf optimizer: Insights, analysis, binarization, and feature selection," *Int. J. Intell. Syst.*, vol. 37, no. 8, pp. 4864–4927, Aug. 2022.
[23] M. H. Nadimi-Shahraki, S. Taghian, and S. Mirjalili, "An improved grey wolf optimizer for solving engineering problems," *Expert Syst. Appl.*, vol. 166, Mar. 2021, Art. no. 113917.
[24] R. Purushothaman, S. P. Rajagopalan, and G. Dhandapani, "Hybridizing gray wolf optimization (GWO) with grasshopper optimization algorithm (GOA) for text feature selection and clustering," *Appl. Soft Comput.*, vol. 96, Nov. 2020, Art. no. 106651.
[25] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary grey wolf optimization approaches for feature selection," *Neurocomputing*, vol. 172, pp. 371–381, Jan. 2016.

[26] M. Sharawi, H. M. Zawbaa, E. Emary, H. M. Zawbaa, and E. Emary, ''Feature selection approach based on whale optimization algorithm,'' in *Proc. 9th Int. Conf. Adv. Comput. Intell. (ICACI)*, Feb. 2017, pp. 163–168.

[27] M. Mafarja, A. Qasem, A. A. Heidari, I. Aljarah, H. Faris, and S. Mirjalili, ''Efficient hybrid nature-inspired binary optimizers for feature selection,'' *Cognit. Comput.*, vol. 12, no. 1, pp. 150–175, Jan. 2020.

[28] Q. Al-Tashi, S. J. A. Kadir, H. M. Rais, S. Mirjalili, and H. Alhussian, ''Binary optimization using hybrid grey wolf optimization for feature selection,'' *IEEE Access*, vol. 7, pp. 39496–39508, 2019.

[29] Q. Tu, X. Chen, and X. Liu, ''Multi-strategy ensemble grey wolf optimizer and its application to feature selection,'' *Appl. Soft Comput.*, vol. 76, pp. 16–30, Mar. 2019.

[30] P. Dhal and C. Azad, ''A multi-objective feature selection method using Newton's law based PSO with GWO,'' *Appl. Soft Comput.*, vol. 107, Aug. 2021, Art. no. 107394.

[31] I. M. El-Hasnony, S. I. Barakat, M. Elhoseny, and R. R. Mostafa, ''Improved feature selection model for big data analytics,'' *IEEE Access*, vol. 8, pp. 66989–67004, 2020.

[32] N. Singh and S. B. Singh, ''Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance,'' *J. Appl. Math.*, vol. 2017, pp. 1–15, Nov. 2017.

[33] S. Sundaramurthy and P. Jayavel, ''A hybrid grey wolf optimization and particle swarm optimization with C4.5 approach for prediction of rheumatoid arthritis,'' *Appl. Soft Comput.*, vol. 94, Sep. 2020, Art. no. 106500.

[34] T. Niu, J. Wang, H. Lu, W. Yang, and P. Du, ''Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting,'' *Expert Syst. Appl.*, vol. 148, Jun. 2020, Art. no. 113237.

[35] Q. Tu, X. Chen, and X. Liu, ''Hierarchy strengthened grey wolf optimizer for numerical optimization and feature selection,'' *IEEE Access*, vol. 7, pp. 78012–78028, 2019.

[36] E. Emary, H. M. Zawbaa, and A. E. Hassanien, ''Binary ant lion approaches for feature selection,'' *Neurocomputing*, vol. 213, pp. 54–65, Nov. 2016.

[37] R. Al-Wajih, S. J. Abdulkadir, N. Aziz, Q. Al-Tashi, and N. Talpur, ''Hybrid binary grey wolf with Harris hawks optimizer for feature selection,'' *IEEE Access*, vol. 9, pp. 31662–31677, 2021.

[38] H. Basak, R. Kundu, S. Chakraborty, and N. Das, ''Cervical cytology classification using PCA and GWO enhanced deep features selection,'' *Social Netw. Comput. Sci.*, vol. 2, no. 5, pp. 1–17, Sep. 2021.

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

[40] P. Hu, J.-S. Pan, and S.-C. Chu, ''Improved binary grey wolf optimizer and its application for feature selection,'' *Knowl.-Based Syst.*, vol. 195, May 2020, Art. no. 105746.

[41] M. J. Islam, X. Li, and Y. Mei, ''A time-varying transfer function for balancing the exploration and exploitation ability of a binary PSO,'' *Appl. Soft Comput.*, vol. 59, pp. 182–196, Oct. 2017.

[42] B. H. Nguyen, B. Xue, P. Andreae, and M. Zhang, ''A new binary particle swarm optimization approach: Momentum and dynamic balance between exploration and exploitation,'' *IEEE Trans. Cybern.*, vol. 51, no. 2, pp. 589–603, Feb. 2021.

[43] N. H. Khan, Y. Wang, D. Tian, R. Jamal, S. Kamel, and M. Ebeed, ''Optimal siting and sizing of SSSC using modified salp swarm algorithm considering optimal reactive power dispatch problem,'' *IEEE Access*, vol. 9, pp. 49249–49266, 2021.

[44] M. Ebeed, A. Ali, M. I. Mosaad, and S. Kamel, ''An improved lightning attachment procedure optimizer for optimal reactive power dispatch with uncertainty in renewable energy resources,'' *IEEE Access*, vol. 8, pp. 168721–168731, 2020.

[45] Z. Ma, O. Volkov, and L. J. Durlofsky, ''Multigroup strategy for well control optimization,'' *J. Petroleum Sci. Eng.*, vol. 214, Jul. 2022, Art. no. 110448.

[46] H. Chantar, M. Mafarja, H. Alsawalqah, A. A. Heidari, I. Aljarah, and H. Faris, ''Feature selection using binary grey wolf optimizer with elite-based crossover for Arabic text classification,'' *Neural Comput. Appl.*, vol. 32, pp. 12201–12220, Jul. 2019.

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

[48] I. Gölcük and F. B. Ozsoydan, ''Evolutionary and adaptive inheritance enhanced grey wolf optimization algorithm for binary domains,'' *Knowl.-Based Syst.*, vol. 194, Apr. 2020, Art. no. 105586.

[49] J. C. Bansal and K. Deep, ''A modified binary particle swarm optimization for knapsack problems,'' *Appl. Math. Comput.*, vol. 218, no. 22, pp. 11042–11061, Jul. 2012.

[50] H. Banka and S. Dara, ''A Hamming distance based binary particle swarm optimization (HDBPSO) algorithm for high dimensional feature selection, classification and validation,'' *Pattern Recognit. Lett.*, vol. 52, pp. 94–100, Jan. 2015.

[51] F. Han, C. Yang, Y.-Q. Wu, J.-S. Zhu, Q.-H. Ling, Y.-Q. Song, and D.-S. Huang, ''A gene selection method for microarray data based on binary PSO encoding gene-to-class sensitivity information,'' *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 14, no. 1, pp. 85–96, Jan. 2017.

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

[53] J. H. Liu, R. H. Yang, and S. H. Sun, ''The analysis of binary particle swarm optimization,'' *J. Nanjing Univ., Natural Sci.*, vol. 47, no. 5, pp. 504–514, Sep. 2011.

[54] X. Zhang, W. Chen, C. Dai, and W. Cai, ''Dynamic multi-group self-adaptive differential evolution algorithm for reactive power optimization,'' *Int. J. Electr. Power Energy Syst.*, vol. 32, no. 5, pp. 351–357, Jun. 2010.

[55] K. Bache and M. Lichman, ''UCI machine learning repository,'' vol. 7, no. 1, pp. 1–17, 2013.

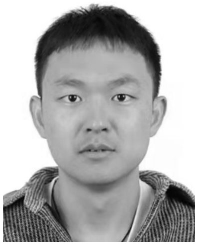
[56] J. Derrac, S. García, D. Molina, and F. Herrera, ''A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms,'' *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, Mar. 2011.

HONGZHUO WU received the bachelor's degree in electronic information engineering from the Neusoft Institute, in 2020. He is currently pursuing the master's degree in computer technology with Ningbo University, where he is focusing on optimization algorithms and neural networks.

SHIYU DU received the bachelor's degree from the Department of Chemistry, University of Science and Technology of China, and the Ph.D. degree from the Department of Chemistry, Purdue University, West Lafayette, IN, USA. From July 2009 to December 2013, he was a Postdoctoral Researcher and a Visiting Scientist with the Los Alamos National Laboratory, Los Alamos, NM, USA. He was a Ph.D. Supervisor. He is currently a Researcher with the Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences. He is also the Chief Scientist of the National Key Research and Development Program. During his stay at the Los Alamos National Laboratory, he mainly undertook the theoretical calculation of key physical properties of fuels and materials. During this period, the anisotropy of the thermal conductivity of uranium dioxide was found by theoretical calculation and was verified by experiments. He returned to China, in January 2014. He established a team of ''theoretical calculation and simulation of energy materials'' with the Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences. He has been working in *Nature Communications*, *Proceedings of the National Academy of Sciences*, *Journal of the American Chemical Society*, *Angewandte Chemie* (International Edition), *ACS Nano*, *Nanoscale*, *The Journal of Physical Chemistry Letters*, and other international scientific research journals. He has published more than 100 academic articles.

YIMING ZHANG received the master's (Hons.) and Ph.D. degrees in engineering from the Queen Mary University of London, in 2006 and 2010, respectively. From 2001 to 2003, he studied mechanical and electronic engineering with the Zhejiang University of Technology. He is currently an Associate Researcher. He is mainly engaged in the application of material informatics, intelligent optimization algorithms, system design methods in the research and development of new ternary layered materials, and extreme environmental energy materials. He has published more than 40 articles, including *Acta Materialia*, *Materialia*, *Philosophical Magazine*, *Journal of Power Sources*, and other magazines, with more than 400 citations, six patents, and five software copyrights. He is a member of the British Society of Minerals, Mining and Materials (IOM3).

QUAN ZHANG received the bachelor's degree, in 2020. He is currently pursuing the master's degree in computer technology with Ningbo University, where he is focusing on optimization algorithms.



KAI DUAN received the bachelor's degree in computer science and technology from Zhejiang Ocean University, in 2020. He is currently pursuing the master's degree in computer technology with Ningbo University, where he is focusing on recommended algorithms and knowledge graphs.

YANRU LIN received the bachelor's degree in network engineering from Shangluo College, in 2020. She is currently pursuing the master's degree in computer technology with Ningbo University, where she is focusing on recommended algorithms and knowledge graphs.