IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 14 May 2023, accepted 5 June 2023, date of publication 14 June 2023, date of current version 20 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3285815

RESEARCH ARTICLE

Improved Binary Gray Wolf Optimizer Based on Adaptive β-Hill Climbing for Feature Selection

TAMARA AMJAD AL-QABLAN [1], MOHD HALIM MOHD NOOR [1], MOHAMMED AZMI AL-BETAR [2,3], (Member, IEEE), AND AHAMAD TAJUDIN KHADER [1]

[1]School of Computer Sciences, Universiti Sains Malaysia, Pulau Pinang 11800, Malaysia
[2]Artificial Intelligence Research Center (AIRC), College of Engineering and Information Technology, Ajman University, Ajman, United Arab Emirates
[3]Department of Information Technology, Al-Huson University College, Al-Balqa Applied University, Irbid 19117, Jordan

Corresponding authors: Tamara Amjad Al-Qablan (tqablan@student.usm.my) and Mohd Halim Mohd Noor (halimnoor@usm.my)

ABSTRACT According to the literature reviews, the Gray Wolf Optimization (GWO) algorithm has been applied to various optimization problems, including feature selection. It is important to consider two opposing ideas while using the metaheuristic technique, exploring the search field, and exploiting the best possible solutions. Despite the increased performance of the GWO, stagnation in local optima areas could still be a concern. This paper proposes a hybridized version of Binary GWO (BGWO) and another recent metaheuristic algorithm, namely adaptive β-hill climbing (AβCH), to enhance the performance of a wrapper-based feature selection approach. The sigmoid transfer function is used to transfer the continuous search space into a binary version to meet the feature selection nature requirement. The K-Nearest Neighbor (KNN) classifier is used to evaluate the goodness of the selected features. To validate the performance of the proposed hybrid approach, 18 standard feature selection UCI benchmark datasets were used. The performance of the proposed hybrid approach was also compared with the Binary hybrid Gray Wolf Optimization Particle Swarm Optimization (BGWOPSO), BGWO (bGWO1,bGWO2), Binary Particle Swarm Optimization (BPSO), Binary Genetic Algorithm (BGA), Whale Optimization Algorithm with Simulated Annealing (WOASAT-2), AβHC with Binary Sailfish (AβBSF), Binary β-Hill Climbing (βHC), Binary JAYA with Adaptive Mutation (BJAM), and Binary Horse herd Optimization Algorithm(BHOA). The findings revealed that the proposed hybrid algorithm was effective in improving the performance of the normal BGWO algorithm, also the proposed hybrid approach outperforms the two approaches of the BGWO algorithm in terms of accuracy and selected feature size. Similarly, compared with BGWOPSO, BPSO, BGA, WOASAT-2, AβBSF, βHC, BJAM, and BHOA feature selection approaches, the proposed approach surpassed them and yielded better accuracy and smaller size of feature selection.

INDEX TERMS Binary Grey wolf Optimizer, adaptive β-hill climbing, local search, feature selection, optimization.

I. INTRODUCTION

The growth in data volume and variety made the evaluating and extracting process of relevant information from big data more difficult, which necessarily requires the development of novel data management, processing, and analysis techniques. The high dimensional of data is considered a major challenge in analyzing and extracting insights from big data, due to the fact that when the dimensionality increases, the computing costs increase significantly [1], making it difficult to solve, particularly for datasets with many features.

Feature selection is a technique concerned with minimizing the data size and selecting only the most relevant features and removing irrelevant or redundant ones to improve the performance and lower the computing costs. There are four approaches for feature selection: filter-based feature selection, wrapper-based feature selection, embedding-based feature selection and hybrid feature selection. [2], [3], where the filter-based and the wrapper-based approaches can be considered the most common approaches [4]. In the filter-based

The associate editor coordinating the review of this manuscript and approving it for publication was Mostafa M. Fouda.

feature selection approach, features are evaluated using statistical metrics, ignoring the inter-dependencies between them. The highest-ranking features are selected, increasing the likelihood of discovering irrelevant and redundant features that effectively reduce the set size; however, their accuracy is low [5]. Chi-square, Information Gain and Document frequency are some of the common filter-based feature selection approaches [6]. The wrapper-based feature selection approach is a combination of a learning algorithm and a feature subset search strategy where it evaluates a candidate features subset depending on the effectiveness of the applied learning algorithm [7], [8]. The feature subset is formed according to the opted search strategy [9]. Since the feature selection problem falls into the category of NP-hard problems [10], it is not possible to find the optimal solution in a polynomial time. Therefore, the optimization techniques are necessary to obtain a near-optimal solution. In recent years, there has been growing interest in using metaheuristic algorithms to solve optimization problems, particularly in cases where traditional optimization techniques such as gradient descent, are ineffective. Metaheuristics refer to the algorithms that are designed to explore the solution space intelligently and efficiently to find a suitable solution within a reasonable amount of time.

Metaheuristic algorithms have proven successful at coming up with good solutions for complex issues [11]. A variety of metaheuristic optimization techniques are employed in wrapper approaches to handle the feature selection problem, like Genetic Algorithm (GA) [12], Ant Colony Optimization (ACO) algorithm [13], Firefly Algorithm (FA) [5], and Whale Optimization Algorithm (WOA) [14].

The GWO is a novel metaheuristic swarm-based optimization algorithm inspired by the gray wolf's natural hunting mechanism and the leadership hierarchy [15]. It has attractive characteristics that make it very popular, such as it is easy and simple to adapt to any optimization problem, it is quick to converge, derivative-free and parameter-free, and it can harmonize the wide exploration and nearby exploitation of the problem search space through its intelligent operators. Therefore, a wide range of research domains have used the GWO to tackle their optimization problems, as reported in the GWO survey [16].

Since the solutions in the feature selection optimization problem are limited to a binary format, where each feature has a binary value that indicates whether it is selected or not, a binary version of GWO called BGWO will be utilized in this study. In the BGWO algorithm, each solution is updated based on the positions of the top three solutions, alpha, beta, and delta, which moves the algorithm's emphasis from exploration to exploitation. As a result, the algorithm could fall in the local optimal area [17], [18]. To tackle this issue, we suggest hybridizing the BGWO algorithm with the AβCH algorithms to enhance the performance of the BGWO algorithm in solving the feature selection problem.

One of the advantages of local search is its ability to find good solutions quickly. By starting with an initial subset of features and exploring the nearby feature subsets, the local search can often find a good solution much faster than the exhaustive search, which considers all possible subsets of features. In addition, local search can be combined with other optimization algorithms to further improve the quality of exploitation capabilities [19].

AβCH is a recently proposed local search algorithm that is an adaptable variation of the β-hill climbing algorithm [20]. The AβCH algorithm starts with an initial solution and iteratively enhances it by investigating the area around the currently selected solution. The algorithm systematically adds or removes features from the solution while measuring the fitness of each modified solution.

Typically, two alternative models for hybridizing metaheuristic algorithms were used. The first one is low-level hybridization. In this model, one component of a metaheuristic algorithm is replaced with another metaheuristic algorithm to create a new hybrid algorithm. The second one is high-level hybridization, where candidate metaheuristics are executed in sequence, and the outcome from one metaheuristic algorithm is used as input for the next metaheuristic algorithm in the sequence.

In this work, a high-level fashion to enhance the feature selection classification result is proposed. We propose a hybrid wrapper feature selection approach based on the BGWO algorithm with AβCH algorithm (Aβ-BGWO) to find the optimal feature subset for enhancing the prediction results.

For the proposed hybrid approach, the K-Nearest Neighbors (KNN) classifier is used as the evaluator. 18 standard feature selection UCI datasets are utilized to assess the performance of the proposed hybrid approach. The extensive results and comparisons demonstrate the superiority of the proposed hybrid approach in decreasing the number of selected features and improving the classification in terms of accuracy.

To the best of our knowledge, this is the first time the BGWO algorithm is hybridized with the AβCH algorithm and applied to solve the feature selection problem. Therefore, the main contributions of this research are as follows:

- The BGWO algorithm is hybridized with another recently proposed metaheuristic algorithm called adaptive β-hill climbing (AβCH).
- The proposed hybrid approach is evaluated on 18 standard feature selection UCI datasets using the KNN classifier.
- The proposed approach is compared with 6 state-of-the-art metaheuristic based feature selection approaches.

The rest of the paper is structured as follows: Section II presents the previous related works. Section III provides a detailed description of our approach. While the setup of experiments, the results, and the analysis are given in Section IV. Finally, Section V presents the conclusion of the work and the future scope.

II. RELATED WORK

Different swarm-based metaheuristic algorithms were implemented to solve the feature selection problem for different applications to obtain the best possible feature subset, such as particle swarm optimization (PSO) [21], artificial bee colony (ABC) [22], ant colony optimization (ACO) [23], JAYA algorithm [24], β-hill climbing (βHC) [25], and binary horse optimization algorithm (BHOA) [26]. These algorithms formulated the feature selection problem as an NP-hard optimization problem; this demonstrates how these algorithms are appropriate to tackle the problem [27], [28]. In the following lines, we will go through recent works that have used the BGWO as a feature selection algorithm. Furthermore, a summary of previous state-of-the-art BGWO-based feature selection approaches is shown in Table 1.

According to literature reviews, many studies have used the BGWO for feature selection. For example, a competitive BGWO (CBGWO) algorithm to tackle the feature selection problem in EMG signals classification was proposed [42]. The proposed method permits the wolves to participate in pairs, where the losers can improve their positions by learning from the victors. The method was implemented on 10 distinct subjects, each having 120 features, and it reduced the number of features by up to 31%. While another study implemented the BGWO algorithm to enhance the result of their Facial Emotion Recognition (FER) system [41]. In their approach, after the data pre-processing stages were performed, the BGWO algorithm was used to choose the best features subset. Following that, depending on the features picked, a GWO-based neural network (NN) was utilized to classify the emotions. A Modified GWO (MGWO) algorithm was proposed for feature selection to aid in the early diagnosis of Parkinson's disease symptoms [40]. The authors used KNN, RF, and DT classifiers to classify the selected features and compared their proposed algorithm to the Optimized Cuttlefish Algorithm. They reported that the MGWO achieved better outcomes. The GWO was strengthened by chaotic theory in [39] for diagnosing paraquat-poisoned patients, the authors incorporated the chaos theory with the GWO to establish a better balance between exploitation and exploration. Their Enhanced GWO (EGWO) algorithm reduced the number of features from 119 to approximately 56-73, with a reduced rate of 38%-53%, after that they used the Extreme Learning Machine (ELM) classifier to perform the prediction depending on the optimal subset that had been chosen. While the authors in [38] suggested a Multi-Strategy Ensemble GWO (MEGWO) algorithm that improved the global search ability of the GWO by incorporating the enhanced global-best lead strategy, adaptive cooperative strategy and disperse foraging strategy. The suggested approach was tested on 12 standard UCI benchmark datasets using the KNN classifier, with total features ranging from 9 to 60. According to the findings, the reduction rate ranged from 30% to 68%. Despite that the MEGWO outperformed the GWO's performance, it includes additional parameters. Thus researchers who wish to use it must first train the parameters to have the best possible setting.

In addition, a binary version of the GWO algorithm and the PSO algorithm termed as BGWOPSO was presented in [37], 18 UCI benchmark datasets with total number of features ranging from 9 to 325 were used. The results showed a decrease in the number of features from 45% to 87.5% with improvement in the accuracy and the computational time compared with the BGWO, the Binary PSO, the binary GA and the WOA. Similarly, [32] merged the PSO and GWO algorithms, resulting in a hybrid algorithm known as GWOPSO. The authors employed 17 datasets from the UCI ML to test the proposed algorithm and the KNN classifier to assess the performance of the chosen set of features. To improve the performance of intrusion detection systems, [36] proposed a modified feature selection algorithm based on BGWO termed (MBGWO). The NSL-KDD network intrusion benchmark was used in this study to evaluate the suggested approach, and the SVM was used to classify the datasets. The outcomes showed that the proposed algorithm improved intrusion detection accuracy by 99.22% and lowered the number of features by up to 65%.

Another study [34] suggested an enhanced BGWO (EBGWO) algorithm for feature selection in anomaly detection. Also, the NLS-KDD dataset was used to evaluate the proposed approach, and classification was done using the SVM. The EBGWO decreased the number of features by up to 54% with an accuracy of 87.46%, according to the testing data. Reference [35] suggested an improved GWO-based algorithm for feature selection to increase the performance of the electronic nose. The presented algorithm was tested on three electronic nose datasets, and the fitness value of the proposed algorithm was calculated using the KNN classifier. While the authors in [33] proposed a framework for financial crisis prediction using improved GWO (IGWO) and fuzzy neural classifier (FNC) to find the optimal set of features. The IGWO-based feature selection method was employed, and the FNC was used for the classification. Two datasets were used to test the proposed technique: Australian Credit and German Credit. The proposed framework reduced the number of features in the Australian Credit dataset by 50% and the German dataset by 63%, with an average accuracy of 98.85%.

Another approach was introduced by [31] to deal with feature selection issue using the GWO algorithm termed Two Phase Crossover Grey Wolf Optimization (TCGWO). The authors improved the algorithm by using a two-phase crossover operator; the approach was trained using the KNN classifier on 10 standard UCI benchmark datasets with a total number of features ranging from 4 to 19. According to the findings, the proposed algorithm outperformed the FA, WA, MVO, and PSO algorithms, and the reduction rate of features ranged from 37.5% to 70%. Furthermore, the GWO as a feature selection algorithm [30] was implemented to build a plant leaf classification system, where three ML were

TABLE 1. GWO-based feature selection summary.

Author	Algorithm	Feature Selection Method	Transfer Function	Dataset	Classifier	Year
[29]	MOBGWO	Wrapper	Sigmoidal	UCI ML	RF, KNN, DT, ET	2021
[30]	GWO	Wrapper	Sigmoidal	Flavia	SVM, RF, NB	2021
[31]	TCGWO	Wrapper	Sigmoidal, Tanh	–	KNN	2021
[32]	GWOPSO	Wrapper	Sigmoidal	UCI ML	KNN	2020
[33]	IGWO	Wrapper	—	AUS Credit, GER	FNC	2020
[34]	EBGWO	Wrapper	Sigmoidal	NLS-KDD	SVM	2020
[35]	IGWO	Wrapper	Sigmoidal	Electronic Nose	KNN, SVM, RF	2020
[36]	MBGWO	Wrapper	Sigmoidal	NSL-KDD	SVM	2019
[37]	BGWOPSO	Wrapper	Sigmoidal	UCI benchmark	KNN	2019
[38]	MEGWO	Wrapper	Sigmoidal	UCI benchmark	KNN	2019
[39]	EGWO	Wrapper	Sigmoidal	PQ-poisoned patients	ELM	2019
[40]	MGWO	Wrapper	Sigmoidal	Hand, Speech,Voice PD	RF, KNN, DT	2019
[41]	GWO	Wrapper	—	Facial motion recognition	KNN	2018
[42]	CBGWO	Wrapper	Sigmoidal	EMG Signals	KNN	2018

utilized to classify the leaves: NB, SVM and RF. The system was evaluated on the Flavia dataset; the results showed that the GWO outperformed the PSO in the number of selected features, accuracy, precision, recall, and F-measure. Also, an approach based on multi-objective BGWO (MOBGWO) to cope with the feature selection problem in energy consumption of appliances [29]. In this study, four ML methods were implemented: RF, Extra trees (ET), DT, and KNN, the result showed that the existing algorithms such as the GA and PSO algorithms performed worse than the suggested algorithm.

Based on the findings of prior studies, the GWO as an optimization algorithm achieved good results in seeking the optimal solutions, where flexibility and simplicity are characteristics of it [43]. On the other hand, the GWO algorithm obviously showed that each solution is updated based on the positions of the top three solutions, alpha, beta, and delta in the population, which moves the algorithm's emphasis from exploration to exploitation. As a result, the algorithm has early convergence and gets stuck in the local minima area [17], [18]. Even though the aforementioned algorithms offer significant benefits, the no free lunch (NFL) theorem [44] proves that no optimization algorithm can completely solve all optimization problems. Each optimization algorithm has its own pros and cons, leaving open the possibility of improving the BGWO's performance in terms of exploitation. This motivates us to propose a hybrid version of BGWO with the well-known local search algorithm AβCH to improve the performance of BGWO in terms of local search ability and convergence rate by finding the local optimal solution in each search iteration to enhance feature selection solutions.

III. PROPOSED METHOD

A high-dimensional dataset contains redundant, nosy, and irrelevant features. Furthermore, in such datasets, the search space expands, making determining the best set of features difficult and time-consuming. Feature selection methods seek to minimize the data's dimensionality and only choose the most important features to improve the classification performance while lowering the computing costs [45]. To tackle this issue, a wrapper feature selection approach combining the Aβ-BGWO algorithm with the KNN classifier is proposed. In this section, we begin by providing an overview of the KNN classifier, the adaptive β-hill Climbing algorithm, the BGWO algorithm, the S-shape curve, followed by a detailed description of the Aβ-BGWO algorithm.

A. KNN CLASSIFIER

In wrapper feature selection approaches, learning algorithms are used to determine the performance of the selected features [32]. KNN classifier is a simple supervised ML algorithm for solving classification issues; it assumes that new testing data and existing data are similar, so it categorizes the new data into the category closest to the existing categories. Some factors, such as the K value, impact the performance of a KNN classifier. Suppose the K value is set too low; the performance may suffer as a result of noisy data, whereas if the K value is set too high, the performance may suffer due to a lack of ability to anticipate training data [46]. Figure 1

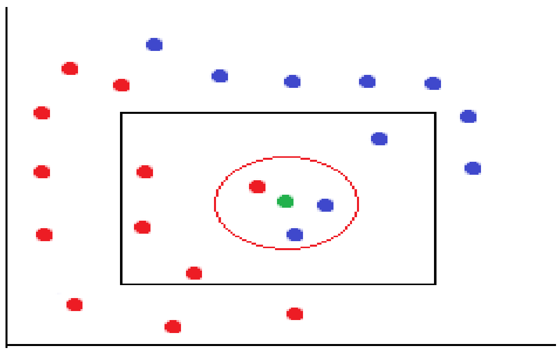


FIGURE 1. KNN classifier when k=3, k=7, and classes = 2.

shows an example of KNN classification. The test object (green dot) should be classified as either a blue class or a red class. If k = 3, the test object is assigned to the blue dots because there are 2 blue dots and only 1 red dot inside the circle area. If k = 7 it is assigned to the red dots where there are 4 red dots vs 3 blue dots inside the square area. Many research have utilized the KNN as a learning algorithm to determine the performance of the selected features because of its simplicity, as shown in Table 1. This encouraged us to utilize this ML algorithm in our study.

B. ADAPTIVE β-HILL CLIMBING

Recently, a new variation of the β-hill climbing algorithm known as the adaptive β-hill climbing (AβCH) algorithm was proposed by [20]. The comparative outcomes demonstrate the applicability of the suggested adaptive algorithm, where it has achieved competitive results for the IEEE-CEC2005 datasets [47]. In the β-hill Climbing algorithm, the parameters $\mathcal{N}$ and β are set in advance and remain fixed during the search. The AβCH algorithm adjusted these parameters based on the algorithm's performance. Where the $\mathcal{N}$ parameter is updated during the search and starts with a value close to one, this value is reduced as the iteration number increases, allowing the algorithm to dynamically adjust the value of $\mathcal{N}$ in response to the results of the search. The method for updating the $\mathcal{N}$ parameter during the search is introduced in [48] as follows:

$$\mathcal{N}_t = 1 - \mathcal{C}_t \tag{1}$$

$$\mathcal{C}_t = \frac{t^{\frac{1}{k}}}{Max\beta_{iter}^{\frac{1}{k}}} \tag{2}$$

where $\mathcal{N}_t$ is the value of $\mathcal{N}$ at time t. The value of $\mathcal{N}$ is gradually reduced from a starting value close to one to a value close to zero over the journey of the search, with the rate of reduction controlled by a fixed number K, and $Max\beta_{iter}$ is the AβHC algorithm's maximum number of iterations.

The value of the β parameter is deterministically adapted within a specific range of $[\beta_{min}, \beta_{max}]$. The specific range of $[\beta_{min}, \beta_{max}]$ is determined as introduced in [49] as follows:

$$\beta_t = \beta_{min} + t * (\beta_{max} - \beta_{min})/Max\beta_{iter} \tag{3}$$

where, the value of β_t is the value of the β rate at time t. β_{min} is the minimum value of the β rate, β_{max} is the maximum value of the β rate, and t is the current time. Algorithm 1 shows the pseudocode of the AβHC algorithm.

Algorithm 1 Pseudocode for the AβHC

Initialize AβCH parameters: β_{min}, β_{max}, and K
Initialize solution of x

$$x_i = L\beta_i + (U\beta_i - L\beta_i) * U(0, 1)\ \forall i = 1 \ldots N$$

Calculate(f(x))
t=0
While $t \leq Max\beta_{iter}$
    x'=x
    Calculate $\mathcal{C}_t$ using Eq. (2)
    Calculate $\mathcal{N}_t$ using Eq. (1)
    Select random position $R - pos \in$ (1,N)
    $x'_{R-pos} = x'_{R-pos} + \mathcal{N}_t$
    x''=x'
    Update β_t using Eq. (3)
    For $i = 1 \cdots N$ do
        Select $r_a \in [0, 1]$
        If ($r_a \leq \beta_t$) then
            $x''_i = x_k$, where $x_k \in x_i$
        End If
    End For
    If $f(x'') \leq f(x)$ then
        x=x''
        $f(x) = f(x'')$
    End If
    $t = t + 1$
End While

C. BGWO ALGORITHM

The BGWO algorithm and how it works will be covered in this subsection.

The GWO algorithm was developed by Mirjalili et al [15]. As the name implies, it is a metaheuristic inspired by nature and based on wolves' herd behavior. This algorithm explores the search space for an optimal solution, similar to other nature-inspired metaheuristics such as the GA and the ACO. It is considered one of the most recent swarm algorithms [16], where the algorithm follows these idealized steps:

1) Encircling prey
The first action of gray wolves hunting is encircling their prey. The encircling behavior can be expressed mathematically as:

$$X(t + 1) = X_p(t) - A * D \tag{4}$$

where X is the position vector of a grey wolf, Xp is the position vector of the prey, t is the current iteration, $t+$

1 is the next iteration, A and D are coefficient vectors expressed as

$$D = \|C * X_p(t) - X_t\| \tag{5}$$

where

$$C = 2r_2 \tag{6}$$

and

$$A = 2a * r_1 - a \tag{7}$$

where $r1$ and $r2$ are randomly generated vectors in the range of [0-1], while a is a vector that decreases linearly from 2 to 0 throughout iterations changes; it can be illustrated in the following way:

$$a = 2 - t\frac{2}{Max_{iter}} \tag{8}$$

where t denotes the current iteration and Max_{iter} denotes the maximum number of optimization iterations.

2) Hunting Prey

The social hierarchy of wolves consists of four distinct levels: alpha, beta, delta, and omega. In most cases, the alpha leads the hunt, also hunting may be led by the beta and delta occasionally. Because we don't know what is the optimal solution, we can consider alpha, beta, and delta to be the best three options, and we can let the remaining wolves (omega) start looking for new solutions. To simulate this, the best three solutions can be denoted by the following formulas.

$$\begin{aligned} X_1 &= X_\alpha(t) - A_1 * D_\alpha \\ X_2 &= X_\beta(t) - A_2 * D_\beta \\ X_3 &= X_\delta(t) - A_3 * D_\delta \end{aligned} \tag{9}$$

where

$$\begin{aligned} D_\alpha &= \|C_1 * X_\alpha - X\| \\ D_\beta &= \|C_2 * X_\beta - X\| \\ D_\delta &= \|C_3 * X_\delta - X\| \end{aligned} \tag{10}$$

As a result, other wolves (omega) should be required to update their positions in the following manner:

$$X_{(t+1)} = \frac{X_1 + X_2 + X_3}{3} \tag{11}$$

A binary version of GWO presented by [50] was used to tackle the binary limitation of the feature selection problem. BGWO algorithm is designed to solve optimization problems in which the variables can take on only two possible values, 0 and 1. The BGWO algorithm follows the basic steps of the Gray Wolf Optimization (GWO) algorithm, which involves the following phases: Initialize the positions of the search agents (wolves) randomly in the search space, then evaluate the fitness value of each search agent by applying the objective function. After that, select the alpha, beta, and delta wolves as the three best search agents based on their fitness values; then, the position of each search agent is updated by following a specific formula based on the position of the leader wolves. The next step is Applying a boundary handling method to ensure the search agents remain within the search space boundaries, then checking if the termination condition has been met. If yes, the algorithm returns the best solution value (alpha), else returns back to the fitness evaluation step. Algorithm 2 shows the pseudocode of the BGWO.

Algorithm 2 Pseudocode for the BGWO Algorithm for Feature Selection

Initialize BGWO parameters
$t = 1$
Randomly initialize BGWO population w_i, $\forall i = 1, 2, \ldots, N$
Calculate the fitness value f of each search agent in the population according to Eq. (16)
Select the best three solutions X_α, X_β and X_δ
While $t \leq Max_{iter}$
  For $i = 1$ to N
    Update the position of the current search agent W_i by Eqs. (9-11)
  End For
  Update the parameters C, A and a according to Eqs. (6-8)
  Calculate the fitness value f of all search agents in the population by Eq. (16)
  Update X_α, X_β and X_δ
  Save the best solutions X_α, X_β and X_δ
  $t = t + 1$
End While
Return $X\alpha$

D. THE S-SHAPE CURVE TRANSFER FUNCTION

The transfer function is used to transform the input features into a more useful representation format by converting the values from a real form to a binary form, which are then used to represent the candidate solutions in the binary search space. As the feature selection problem's solutions only have 0 and 1 values. We have employed a binary version of GWO presented by [50] to tackle this limitation. The authors in [50] proposed two approaches for manipulating the binary representation; in the second approach, the sigmoid function was used to convert a vector of data from continuous to binary form. The sigmoid function or the so-called sigmoid curve is a mathematical function that has an ''S''-shaped curve. We chose the second approach in our search since it outperformed the other one in terms of performance. In this approach, the main updating equation, shown in Equation 12, was employed to force the updated gray wolf position vector to be binary and to allow the wolves to move in a binary search

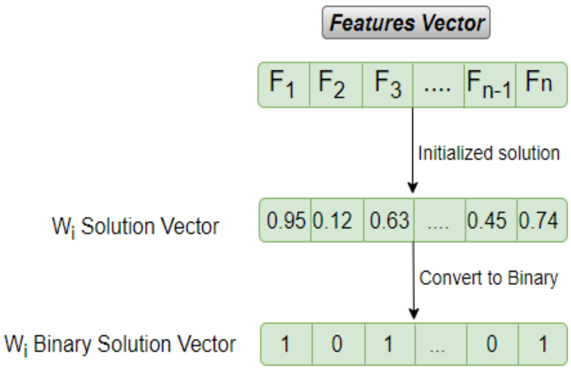


FIGURE 2. BGWO Solution Representation.

space.

$$X_{t+1} = \begin{cases} 1 & \text{if } sigmoid(X_{t+1}) > rand() \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

where X_{t+1} is the continuous solution vector at iteration $t+1$, $rand()$ is a random number $\in [0, 1]$.

E. IMPROVED BGWO BASED ON AβCH

Implementing the Aβ-BGWO algorithm for determining the best subset of features with the KNN classifier is described in depth in this subsection, including the BGWO universe representation, the applied fitness measurement and the Aβ-BGWO architecture.

1) SOLUTIONS' REPRESENTATION AND INITIALIZATION

The solutions generated by the BGWO algorithm can be mathematically represented as an array of randomly generated real numbers with upper and lower boundaries. All upper and lower bounds are set to 1 and 0, respectively. Each solution has the same number of features obtained, and it is initialized using Equation 13.

$$w_{ij} = lb + (ub - lb) * rand() \quad (13)$$

where w_{ij} is a representation of the value of feature j for the wolf i, lb is the lower bound and ub is the upper bound. Then after applying Equation 13 the solutions vectors for every wolf are converted to the binary format by applying Equation 14

$$w_{ij} = \begin{cases} 1 & \text{if } w_{ij} > 0.5 \\ 0 & otherwise \end{cases} \quad (14)$$

Figure 2 shows a representative graphic for initializing a solution by Equations [13, 14]. When the feature value is 1, it indicates that the feature has been picked; if it is 0, it indicates that it has not been selected. For instance, w_i has a 11010 solution vector with five features, which indicates that the first, second and fourth features were chosen, while the third and the fifth features were not.

TABLE 2. Benchmark datasets.

No.	Dataset	No.Instanses	No.Features	No.Classes
1	BreastCancer	699	9	2
2	BreastEW	569	30	2
3	CongressEW	435	16	2
4	Exactly	1000	13	2
5	Exactly2	1000	13	2
6	HeartEW	270	13	2
7	IonosphereEW	351	34	2
8	KrvskpEW	3196	36	2
9	Lymphography	148	18	2
10	M-of-n	1000	13	2
11	PenglungEW	73	325	2
12	SonarEW	208	60	2
13	SpectEW	267	22	2
14	Tic-tac-toe	958	9	2
15	Vote	300	16	2
16	WaveformEW	5000	40	3
17	WineEW	178	13	3
18	Zoo	101	16	6

TABLE 3. Aβ-BGWO Parameters.

Parameters	value
Number of iterations	100
No of search agents	10
Problem dimension	Number of features for the documents
No. Repetitions of runs	20
KNN	5

2) OBJECTIVE FUNCTION

Feature selection problem is considered a multi-objective problem with two main objectives, which are reducing the number of selected features and maximizing the classification accuracy simultaneously. To deal with these two opposed objectives, we need a fitness function as feedback to assess the solutions provided by the Aβ-BGWO algorithm and aid the approach in identifying a desirable minimum set of features while also maximizing classification accuracy. The fitness function shown in Equation (16) is utilized in this study to deal with these two objectives and balance between them. The first step is to implement the KNN classifier to acquire classification accuracy, which is used to find the error rate value using Equation (15). After that. The fitness function f is calculated. The Aβ-BGWO algorithm has been updated to minimize the value of fitness function f.

$$\gamma = 1 - acc \quad (15)$$

$$f = \alpha \times \gamma + \beta \frac{|R|}{|M|} \quad (16)$$

where f denotes the fitness function value, γ is the error rate for the KNN classifier, $|R|$ is the number of selected features and $|M|$ is the total number of features. $\alpha \in [0, 1]$ and β= (1- α) are two parameters to control the importance of classification quality and the subset length [50].

3) UPDATE POPULATION

Like all other swarm algorithms, the Aβ-BGWO algorithm repeats a series of stages and updates the outcomes using a set of equations each time.

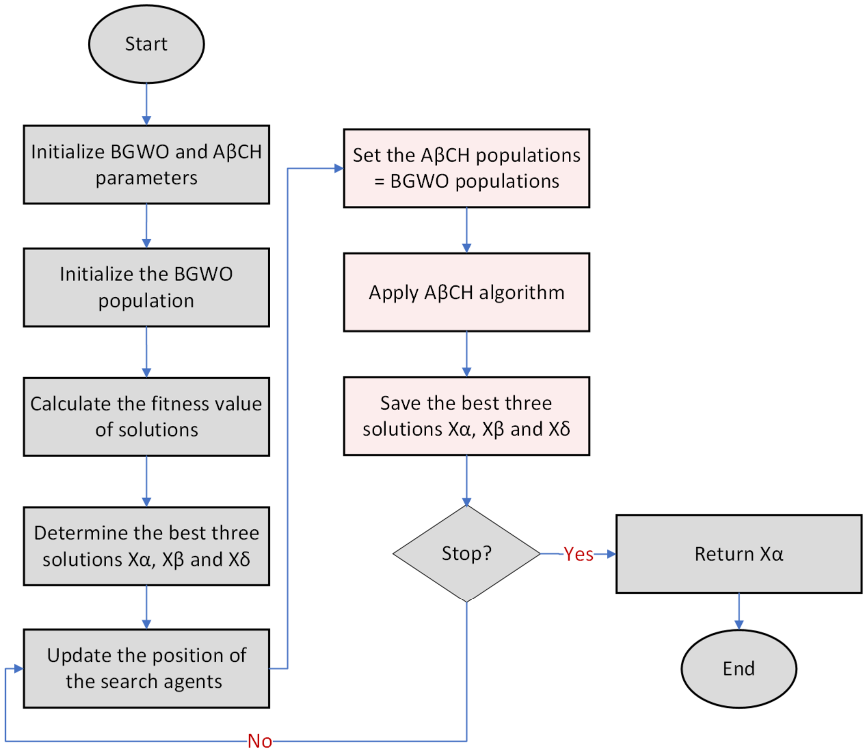
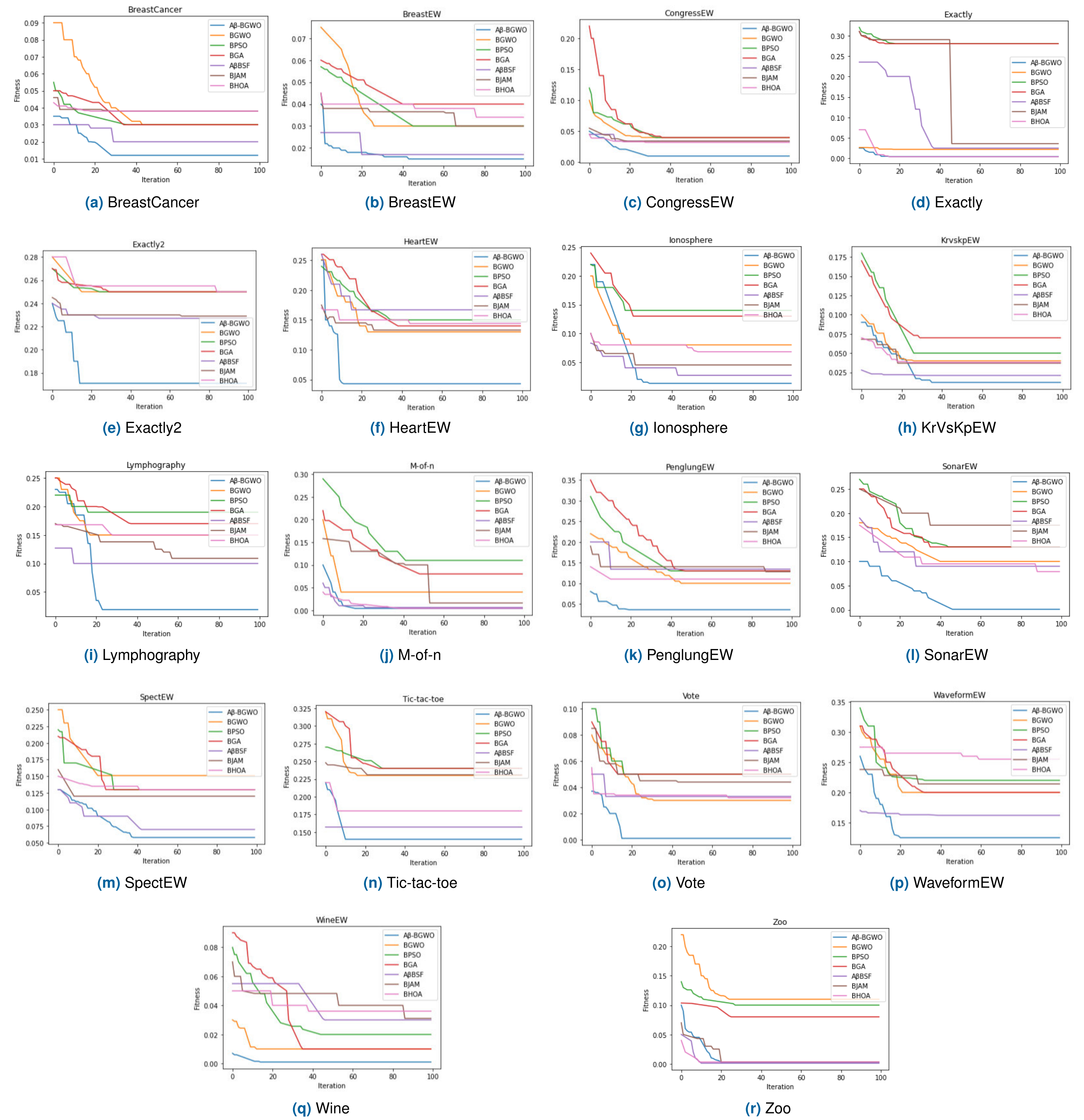


FIGURE 3. **Aβ-BGWO Flowchart.**

TABLE 4. **A comparison between the proposed Aβ-BGWO and the original dataset using the KNN classifier in terms of accuracy and No. of selected features.**

No.	Dataset	Original		Aβ-BGWO	
		Accuracy	All Features	Accuracy	Selected Features
1	BreastCancer	96	9	99	4
2	BreastWE	93	30	98	2
3	CongressEW	92	16	99	4
4	Exactly	72	13	100	6
5	Exactly2	73	13	83	7
6	HeartEW	68	13	96	4
7	IonosphereEW	83	34	99	5
8	KrvskpEW	96	36	99	16
9	Lymphography	81	18	99	6
10	M-of-n	87	13	100	6
11	PenglungEW	81	325	96	13
12	SonarEW	81	60	100	12
13	SpectEW	82	22	94	4
14	Tic-tac-toe	81	9	87	9
15	Vote	92	16	100	3
16	WaveformEW	81	40	87	13
17	WineEW	67	13	100	3
18	Zoo	87	16	100	5

The algorithm (3) provides a pseudocode for the proposed approach. The First step is to set the Aβ-BGWO parameters, which include the number of wolves in the population, the coefficient vectors A, C, a, the maximum number of iterations required to complete the termination operation, B_{max}, B_{min}, and K. Following that, each wolf is initially assigned

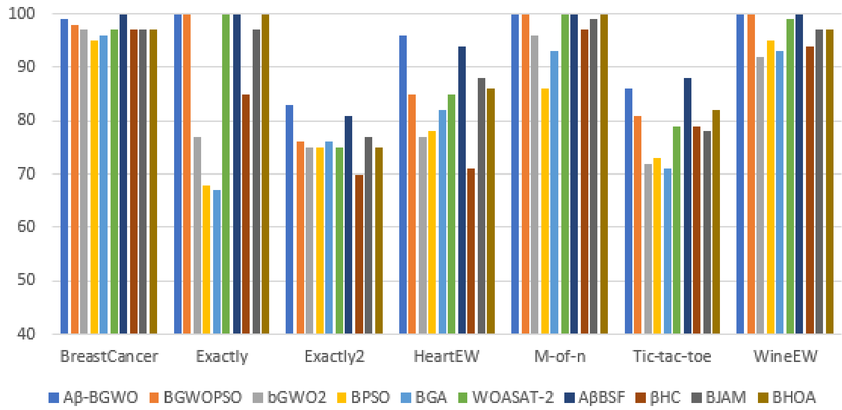


(a) BreastCancer (b) BreastEW (c) CongressEW (d) Exactly (e) Exactly2 (f) HeartEW (g) Ionosphere (h) KrVsKpEW (i) Lymphography (j) M-of-n (k) PenglungEW (l) SonarEW (m) SpectEW (n) Tic-tac-toe (o) Vote (p) WaveformEW (q) Wine (r) Zoo

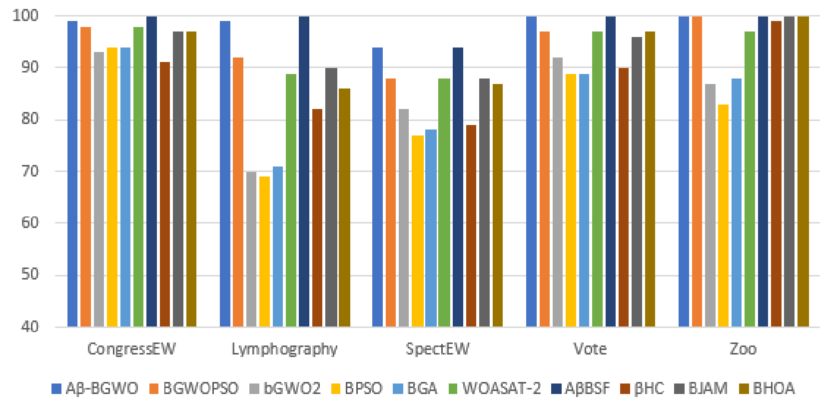
FIGURE 4. **The convergence curves of BGWO and Aβ-BGWO algorithms for all datasets.**

to a random position solution vector $[W_i = f_{i1}, f_{i2}, \ldots f_{im}]$, where m denotes the total number of features and i denotes the ith candidate solution. In the third step, the fitness value f for each search agent W_i in the population will be calculated using Equation 16, followed by identifying the best three solutions $X\alpha$, $X\beta$ and $X\delta$ to update the solutions of the wolves using to Equations 9-11. The coefficient vectors parameters A, C, and a will then be updated using Equations 6-8, and again the fitness value f for each search agent wi in the population will be calculated according to Equation 16, then keeping track of the best three solutions $X\alpha$, $X\beta$ and $X\delta$, after that the AβCH algorithm will apply for the new search agents (offspring solutions) to improve them locally. Thereafter, the $X\alpha$, $X\beta$ and $X\delta$ solutions are selected. Finally, if the termination requirements are met, the proposed method will end the search and retain the best solution $X\alpha$; otherwise, it will repeat the third step.

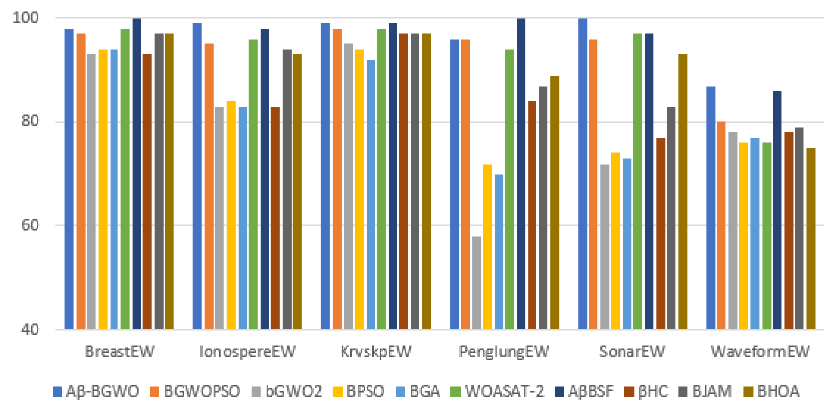
To identify the optimum feature subset, the feature selection algorithm must locate the global optima, which necessitates rigorous search space exploration and exploitation.



(a) Datasets in range of 1-15 features



(b) Datasets in range of 16 - 29 features



(c) Datasets in range of 30 - 350 features

FIGURE 5. Classification accuracy obtained by Aβ-BGWO with state-of-the-art feature selection approaches for 18 UCI datasets.

As a result, we have enhanced BGWO's exploitation powers through AβHC as shown in Algorithm 3.

According to Aβ-BGWO analysis, the worst time complexity is

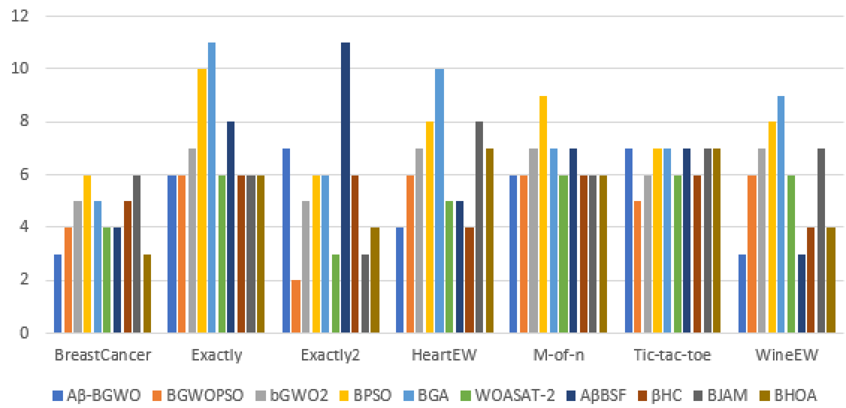
$$O(Max_{iter} * (Agent_{num} * Fittnes_{time} + D)) \tag{17}$$

where Max_{iter} is the maximum number of iterations, $Agent_{num}$ is the number of wolves, $Fittnes_{time}$ is the amount of time needed to calculate a specific agent's fitness value using the KNN classifier, and D is the total number of features.

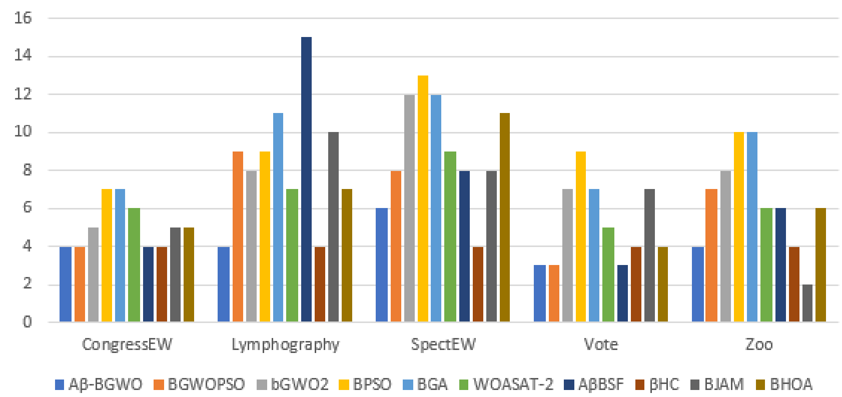
Figure 3 illustrates the flowchart of proposed Aβ-BGWO for feature selection

IV. RESULT AND DISCUSSION

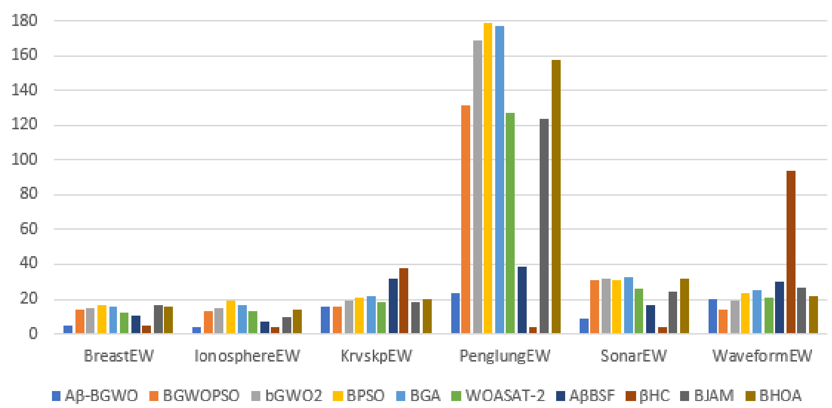
This section will discuss the outcomes of the proposed hybrid approach for the datasets listed in Table 2.



(a) Datasets in range of 1-15 features



(b) Datasets in range of 16 - 29 features



(c) Datasets in range of 30 - 350 features

FIGURE 6. Number of selected features obtained by Aβ-BGWO with state-of-the-art feature selection approaches for 18 UCI datasets.

A. EXPERIMENTAL SETUP

In this study, the performance and effectiveness of the proposed approach were evaluated using 18 benchmark datasets from the UCI ML Repository (see Table 2). Our experiments were carried out on a personal computer with a Core i7 processor having 2.6 GHz frequency, 16GB of RAM, and 1TB of hard disk space. The framework is written in Python programming language.

The Aβ-BGWO algorithm's parameters are specified as mentioned in Table 3; these parameters were applied in [37] and [50]. For all experiments, we used 10-fold cross-validation during the classification process, which involved splitting the dataset into 10 equal parts and using one of those parts for testing and the other 9 parts for training, this technique was adopted from [37], [51], and [52]. The classification process is repeated 20 times for each dataset, then the

TABLE 5. Classification accuracy comparison between the proposed Aβ-BGWO, the two approaches of BGWO (bGWO1,bGWO2), and with original Dataset using KNN classifier.

No.	Dataset	Original	bGWO1	bGWO2	Aβ-BGWO
1	BreastCancer	96	98	98	**99**
2	BreastWE	93	92	94	**98**
3	CongressEW	92	94	94	**99**
4	Exactly	72	71	78	**100**
5	Exactly2	73	75	75	**83**
6	HeartEW	68	78	78	**96**
7	IonosphereEW	83	81	83	**99**
8	KrvskpEW	96	94	96	**99**
9	Lymphograpgy	81	74	70	**99**
10	M-of-n	87	91	96	**100**
11	PenglungEW	81	60	58	**96**
12	SonarEW	81	73	73	**100**
13	SpectEW	82	82	82	**94**
14	Tic-tac-toe	81	73	73	**87**
15	Vote	92	91	92	**100**
16	WaveformEW	81	79	79	**87**
17	WineEW	67	93	92	**100**
18	Zoo	87	88	88	**100**

Algorithm 3 Pseudocode of the Aβ-BGWO Algorithm for Feature Selection

1: Initialize BGWO and AβCH parameters
2: $t = 1$
3: Randomly initialize BGWO population w_i, $\forall i = 1, 2, \ldots, N$
4: Calculate the fitness value f of each search agent in the population according to Eqs. (15, 16)
5: Select the best three solutions X_α, X_β and X_δ
6: While $t \leq Max_{iter}$
    For $i = 1$ to N
        Update the position of the current search agentW_i by Eqs. (9-12)
    Update the parameters C, A and a according to Eqs. (6-8)
    Calculate the fitness value f of all search agents in the population according to Eqs. (15, 16)
    Update X_α, X_β and X_δ
    Apply AβCH algorithm
    Save the best solutions X_α, X_β and X_δ
    $t = t + 1$
  End while
7: Return $X\alpha$

average accuracy, average number of selected features, and average fitness across 20 runs are reported. Moreover, The literature clarifies that they accomplished an iterative calculation to determine the proper value of K. Their experiments show that the k = 5 value is able to produce the best overall results across all datasets. This value of k=5 was adopted in many state-of-the-art methods targeting similar problem [25], [26], [37], [50], [51], [52], [53].

B. THE RESULTS OF Aβ-BGWO

In this section, we have analyzed and evaluated the outcomes of the suggested approach.

As illustrated in Table 4, the Aβ-BGWO achieved >=99% accuracy rate for 11 datasets named: BreastCancer, CongressEW, Exactly, IonosphereEW, KrvskpEW, Lymphography, M-of-N, SonorEW, Vote, WineEW, and Zoo. It is worth highlighting that Aβ-BGWO demonstrated a remarkable achievement of a 100% accuracy rate in six of these datasets. Moreover, Table 5 presents a comparison of classification accuracies among the proposed approach, the two approaches of BGWO (bGWO1, bGWO2), and the original dataset using the KNN classifier. we can observe that the results of the Aβ-BGWO outperformed the others in all datasets in terms of accuracy. This indicates the algorithm's ability to efficiently select relevant features, which is an important aspect of feature selection problems.

Based on the information in Tables 4 and 5, it can be concluded that the use of AβCH significantly improves the performance of the BGWO algorithm in terms of exploring the search space and obtaining better solutions. This is evident from the improvement in classification accuracy by more than 15% on several datasets such as Exactly, HeartEW, IonospereEW, Lymphograpgy, PenglungEW, SonarEW, and WineEW. Furthermore, in Figure 4, the convergence behavior of the proposed AβCH algorithm is displayed, allowing for a comparison with the convergence behavior of other algorithms. The x-axis represents 100 iterations, while the y-axis represents the corresponding fitness values, providing a visual representation of the convergence process. According to the data, it seems that Aβ-BGWO performed better than BGWO since the curve for Aβ-BGWO consistently displayed lower values (which can be interpreted as higher performance) than BGWO across all datasets. This shows that in the given search space, Aβ-BGWO was more effectively able to converge to a better solution than BGWO.

C. COMPARISON WITH THE STATE-OF-ART

The proposed algorithm Aβ-BGWO was compared with 9 state-of-the-art algorithms for feature selection: BGWOPSO, bGWO2, BPSO, BGA, WOASAT-2, AβBSF, βHC, BJAM, and BHOA. Reading the results in Table 6 and Figure 5, as a whole, illustrate that the performance of the Aβ-BGWO algorithm with the proposed approach outperformed the BGWOPSO, bGWO2, BPSO, BGA, WOASAT-2, βHC, BJAM, and BHOA algorithms in twelve datasets, and better than AβBSF in five datasets, while it can provide very competitive results for the remaining datasets in term of accuracy.

Table 7 and Figure 6 show how the proposed Aβ-BGWO performs better in most datasets in term of selecting less number of features, where the average number of selected features

TABLE 6. Comparison of classification accuracy obtained by Aβ-BGWO with some state-of-the-art feature selection approaches for 18 UCI datasets.

No.	Dataset	Aβ-BGWO	BGWOPSO	bGWO2	BPSO	BGA	WOASAT-2	AβBSF	βHC	BJAM	BHOA
1	BreastCancer	99	98	97	95	96	97	100	97	97	97
2	BreastEW	98	97	93	94	94	98	100	93	97	97
3	CongressEW	99	98	93	94	94	98	100	91	97	97
4	Exactly	**100**	100	77	68	67	100	100	85	97	100
5	Exactly2	**83**	76	75	75	76	75	81	70	77	75
6	HeartEW	**96**	85	77	78	82	85	94	71	88	86
7	IonosphereEW	**99**	95	83	84	83	96	98	83	94	93
8	KrvskpEW	**99**	98	95	94	92	98	99	97	97	97
9	Lymphography	99	92	70	69	71	89	100	82	90	86
10	M-of-n	**100**	100	96	86	93	100	100	97	99	100
11	PenglungEW	96	96	58	72	70	94	100	84	87	89
12	SonarEW	**100**	96	72	74	73	97	97	77	83	93
13	SpectEW	**94**	88	82	77	78	88	94	79	88	87
14	Tic-tac-toe	87	81	72	73	71	79	88	79	78	82
15	Vote	**100**	97	92	89	89	97	100	90	96	97
16	WaveformEW	**87**	80	78	76	77	76	86	78	79	75
17	WineEW	**100**	100	92	95	93	99	100	94	97	97
18	Zoo	**100**	100	87	83	88	97	100	99	100	100

TABLE 7. Comparison of the number of selected features obtained by Aβ-BGWO with some state-of-the-art feature selection approaches for 18 UCI datasets.

No.	Dataset	Aβ-BGWO	BGWOPSO	bGWO2	BPSO	BGA	WOASAT-2	AβBSF	βHC	BJAM	BHOA
1	BreastCancer	**3**	4	5	6	5	4	4	5	6	3
2	BreastEW	**5**	14	15	17	16	12	11	5	17	16
3	CongressEW	**4**	4	5	7	7	6	4	4	5	5
4	Exactly	**6**	6	7	10	11	6	8	6	6	6
5	Exactly2	7	2	5	6	6	3	11	6	3	4
6	HeartEW	**4**	6	7	8	10	5	5	4	8	7
7	IonosphereEW	**4**	13	15	19	17	13	7	4	10	14
8	KrvskpEW	**16**	16	19	21	22	18	32	38	18	20
9	Lymphography	**4**	9	8	9	11	7	15	4	10	7
10	M-of-n	**6**	6	7	9	7	6	7	6	6	6
11	PenglungEW	23	131	169	179	177	127	39	4	124	158
12	SonarEW	9	31	32	31	33	26	17	4	24	32
13	SpectEW	6	8	12	13	12	9	8	4	8	11
14	Tic-tac-toe	7	5	6	7	7	6	7	6	7	7
15	Vote	**3**	3	7	9	7	5	3	4	7	4
16	WaveformEW	20	14	19	23	25	21	30	94	27	22
17	WineEW	**3**	6	7	8	9	6	3	4	7	4
18	Zoo	4	7	8	10	10	6	6	4	2	6
	Average	**7.4**	15.88	19.7	21.7	21.8	15.99	12.1	11.2	16.4	18.4

TABLE 8. Comparison results in terms of the average fitness values obtained by Aβ-BGWO with some state-of-the-art feature selection approaches for 18 UCI datasets.

No.	Dataset	Aβ-BGWO	BGWOPSO	bGWO2	BPSO	BGA	WOASAT-2	AβBSF	βHC	BJAM	BHOA
1	BreastCancer	**0.012**	0.030	0.030	0.030	0.030	0.040	0.020	0.020	0.038	0.037
2	BreastEW	**0.015**	0.040	0.030	0.030	0.040	0.030	0.017	0.050	0.031	0.034
3	CongressEW	**0.010**	0.030	0.040	0.040	0.040	0.030	0.034	0.026	0.033	0.032
4	Exactly	**0.004**	0.004	0.220	0.280	0.280	0.010	0.025	0.042	0.036	0.004
5	Exactly2	**0.171**	0.240	0.250	0.250	0.250	0.250	0.227	0.245	0.229	0.250
6	HeartEW	**0.043**	0.150	0.130	0.150	0.140	0.160	0.167	0.215	0.133	0.144
7	IonosphereEW	**0.013**	0.050	0.080	0.140	0.130	0.040	0.027	0.038	0.045	0.068
8	KrvskpEW	**0.013**	0.020	0.040	0.050	0.070	0.020	0.021	0.039	0.037	0.038
9	Lymphography	**0.019**	0.080	0.150	0.190	0.170	0.110	0.100	0.088	0.109	0.15
10	M-of-n	**0.004**	0.004	0.040	0.110	0.080	0.010	0.006	0.028	0.016	0.004
11	PenglungEW	0.036	0.050	0.100	0.130	0.130	0.030	0.134	0.147	0.128	0.111
12	SonarEW	**0.001**	0.050	0.100	0.130	0.130	0.030	0.090	0.053	0.175	0.079
13	SpectEW	**0.059**	0.120	0.150	0.130	0.140	0.130	0.070	0.184	0.121	0.130
14	Tic-tac-toe	0.142	0.190	0.230	0.240	0.240	0.210	0.157	0.131	0.231	0.181
15	Vote	**0.001**	0.030	0.030	0.050	0.050	0.040	0.033	0.052	0.044	0.032
16	WaveformEW	**0.125**	0.210	0.200	0.220	0.200	0.250	0.162	0.253	0.214	0.255
17	WineEW	**0.001**	0.004	0.010	0.020	0.010	0.010	0.030	0.005	0.031	0.036
18	Zoo	0.002	0.004	0.110	0.100	0.080	0.040	0.001	0.003	0.003	0.003

for the proposed algorithm is 7.4, which is fewer than the average number of selected features for the nearest algorithm βHC, which averages 11.2. The other algorithms (AβBSF, BGWOPSO, WOASAT-2, BJAM, BHOA, bGWO2, BPSO, and BGA) have a higher average number of selected features (12.1, 15.88, 15.99, 16.4, 18.4, 19.69, 21.66, and 21.77, respectively). Furthermore, a comparison in terms of average fitness values obtained by Aβ-BGWO with other algorithms was presented in Table 8, according to the presented results, the Aβ-BGWO algorithm outperforms the other algorithms in fifteen datasets, while it can provide very competitive results compared to the other algorithms for the remaining datasets.Moreover, based on the convergence curve depicted in Figure 4, it is evident that Aβ-BGWO outperformed other state-of-the-art algorithms. This observation indicates that within the provided search space, Aβ-BGWO demonstrated a more effective ability to converge towards optimal solutions when compared to its counterparts.

The results in Tables 6, 7, and 8 indicate that the Aβ-BGWO can achieve good classification results while reducing the number of features used, leading to a more efficient and interpretable model, which can be helpful in various real-world applications. Overall, the proposed Aβ-BGWO still demonstrates good results on all 18 datasets, which indicates its ability to find optimal solutions effectively and demonstrates the ability to balance exploitation and exploration during optimization.

V. CONCLUSION

An improved feature selection approach is proposed by hybridizing the BGWO algorithm with AβCH local search algorithm. Through hybridizing the BGWO with the AβCH the proposed approach improves the accuracy and convergence speed of the solution. The AβCH algorithm is known for its ability to find the local optimum efficiently, and by incorporating this algorithm into the BGWO algorithm, the Aβ-BGWO algorithm can improve the quality of the solution obtained by the BGWO algorithm.

To validate the performance of the proposed approach, 18 standard feature selection UCI benchmark datasets were used. The performance of the proposed hybrid approach was compared with nine feature selection approaches called BGWOPSO, bGWO2, BPSO, BGA, WOASAT-2, AβBSF, βHC, BJAM, and BHOA. The study results showed that the proposed approach outperformed a wide range of approaches in terms of accuracy, number of features selected and the fitness value. The comparison of the fitness also showed the superior performance of the proposed approach compared to other state-of-the-art approaches. This indicates the ability of the Aβ-BGWO algorithm to control the trade-off between exploratory and exploitative behaviors during optimization iterations, which is an important aspect of optimization algorithms. The high accuracy and efficient feature selection provided by the Aβ-BGWO algorithm can be valuable in many ML and data analysis applications.

For future works, the Aβ-BGWO algorithm has a wide range of potential applications, and there is room for further exploration and experimentation. Applying the Aβ-BGWO algorithm to different real-world problems, such as facial emotion recognition, engineering optimization problems, handwriting recognition, and sentiment recognition, can provide valuable insights into the algorithm's capabilities and potential for solving complex optimization problems. Additionally, hybridizing the Aβ-BGWO algorithm with other meta-heuristic algorithms can further enhance its performance and broaden its range of applications. This can also provide opportunities to explore different combinations of algorithms and identify the best combination for a particular optimization problem. Finally, experimenting with the Aβ-BGWO algorithm using other popular classifiers, such as SVM and Naive Bayes, can provide a comprehensive evaluation of the algorithm's performance and its ability to solve complex optimization problems in different domains. In conclusion, there is ample scope for future work on the Aβ-BGWO algorithm, and its performance can be further enhanced and validated through additional experiments and applications in various domains.

REFERENCES

[1] Z. M. Hira and D. F. Gillies, "A review of feature selection and feature extraction methods applied on microarray data," *Adv. Bioinf.*, vol. 2015, pp. 1–13, Jun. 2015.

[2] N. Hoque, H. A. Ahmed, D. K. Bhattacharyya, and J. K. Kalita, "A fuzzy mutual information-based feature selection method for classification," *Fuzzy Inf. Eng.*, vol. 8, no. 3, pp. 355–384, Sep. 2016.

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

[4] M. A. Hassonah, R. Al-Sayyed, A. Rodan, A. M. Al-Zoubi, I. Aljarah, and H. Faris, "An efficient hybrid filter and evolutionary wrapper approach for sentiment analysis of various topics on Twitter," *Knowl.-Based Syst.*, vol. 192, Mar. 2020, Art. no. 105353.

[5] A. Kumar and R. Khorwal, "Firefly algorithm for feature selection in sentiment analysis," in *Computational Intelligence in Data Mining: Proceedings of the International Conference on CIDM, 10–11 December 2016*. Springer, 2017, pp. 693–703.

[6] M. Labani, P. Moradi, F. Ahmadizar, and M. Jalili, "A novel multivariate filter method for feature selection in text classification problems," *Eng. Appl. Artif. Intell.*, vol. 70, pp. 25–37, Apr. 2018.

[7] M. Alweshah, S. Alkhalaileh, M. A. Al-Betar, and A. A. Bakar, "Coronavirus herd immunity optimizer with greedy crossover for feature selection in medical diagnosis," *Knowl.-Based Syst.*, vol. 235, Jan. 2022, Art. no. 107629.

[8] H. Liu, M. Zhou, and Q. Liu, "An embedded feature selection method for imbalanced data classification," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 3, pp. 703–715, May 2019.

[9] R. J. Urbanowicz, M. Meeker, W. L. Cava, R. S. Olson, and J. H. Moore, "Relief-based feature selection: Introduction and review," *J. Biomed. Informat.*, vol. 85, pp. 189–203, Sep. 2018.

[10] L. Brezočnik, I. Fister Jr., and V. Podgorelec, "Swarm intelligence algorithms for feature selection: A review," *Appl. Sci.*, vol. 8, no. 9, p. 1521, Sep. 2018.

[11] C. M. Fonseca and P. J. Fleming, "An overview of evolutionary algorithms in multiobjective optimization," *Evol. Comput.*, vol. 3, no. 1, pp. 1–16, Mar. 1995.

[12] F. Iqbal, J. M. Hashmi, B. C. M. Fung, R. Batool, A. M. Khattak, S. Aleem, and P. C. K. Hung, "A hybrid framework for sentiment analysis using genetic algorithm based feature reduction," *IEEE Access*, vol. 7, pp. 14637–14652, 2019.

[13] S. Aggarwal and B. Chhabra, "Sentimental analysis of tweets using ant colony optimizations," in *Proc. Int. Conf. Intell. Sustain. Syst. (ICISS)*, Dec. 2017, pp. 1219–1223.

[14] M. Tubishat, M. A. M. Abushariah, N. Idris, and I. Aljarah, "Improved whale optimization algorithm for feature selection in Arabic sentiment analysis," *Appl. Intell.*, vol. 49, no. 5, pp. 1688–1707, May 2019.

[15] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

[16] 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, Jul. 2018.

[17] M. Abdel-Basset, K. M. Sallam, R. Mohamed, I. Elgendi, K. Munasinghe, and O. M. Elkomy, "An improved binary grey-wolf optimizer with simulated annealing for feature selection," *IEEE Access*, vol. 9, pp. 139792–139822, 2021.

[18] R. R. Rajammal, S. Mirjalili, G. Ekambaram, and N. Palanisamy, "Binary grey wolf optimizer with mutation and adaptive K-nearest neighbour for feature selection in Parkinson's disease diagnosis," *Knowl.-Based Syst.*, vol. 246, Jun. 2022, Art. no. 108701.

[19] Z. M. Elgamal, N. M. Yasin, A. Q. M. Sabri, R. Sihwail, M. Tubishat, and H. Jarrah, "Improved equilibrium optimization algorithm using elite opposition-based learning and new local search strategy for feature selection in medical datasets," *Computation*, vol. 9, no. 6, p. 68, Jun. 2021.

[20] M. A. Al-Betar, I. Aljarah, M. A. Awadallah, H. Faris, and S. Mirjalili, "Adaptive β–hill climbing for optimization," *Soft Comput.*, vol. 23, no. 24, pp. 13489–13512, 2019.

[21] 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.

[22] H. Wang, H. Yu, Q. Zhang, S. Cang, W. Liao, and F. Zhu, "Parameters optimization of classifier and feature selection based on improved artificial bee colony algorithm," in *Proc. Int. Conf. Adv. Mech. Syst. (ICAMechS)*, Nov. 2016, pp. 242–247.

[23] S. R. Ahmad, A. A. Bakar, and M. R. Yaakub, "Ant colony optimization for text feature selection in sentiment analysis," *Intell. Data Anal.*, vol. 23, no. 1, pp. 133–158, 2019.

[24] M. A. Awadallah, M. A. Al-Betar, A. I. Hammouri, and O. A. Alomari, "Binary Jaya algorithm with adaptive mutation for feature selection," *Arabian J. Sci. Eng.*, vol. 45, no. 12, pp. 10875–10890, Dec. 2020.

[25] M. A. Al-Betar, A. I. Hammouri, M. A. Awadallah, and I. A. Doush, "Binary β-hill climbing optimizer with s-shape transfer function for feature selection," *J. Ambient Intell. Hum. Comput.*, vol. 12, no. 7, pp. 7637–7665, 2021.

[26] M. A. Awadallah, A. I. Hammouri, M. A. Al-Betar, M. S. Braik, and M. A. Elaziz, "Binary horse herd optimization algorithm with crossover operators for feature selection," *Comput. Biol. Med.*, vol. 141, Feb. 2022, Art. no. 105152.

[27] S. R. Ahmad, M. Zakwan, N. Syafira, N. Moziyana, and S. Ismail, "A review of feature selection and sentiment analysis technique in issues of propaganda," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 11, pp. 1–6, 2019.

[28] H. Zhao, Z. Liu, X. Yao, and Q. Yang, "A machine learning-based sentiment analysis of online product reviews with a novel term weighting and feature selection approach," *Inf. Process. Manage.*, vol. 58, no. 5, Sep. 2021, Art. no. 102656.

[29] D. Moldovan and A. Slowik, "Energy consumption prediction of appliances using machine learning and multi-objective binary grey wolf optimization for feature selection," *Appl. Soft Comput.*, vol. 111, Nov. 2021, Art. no. 107745.

[30] B. P. Dudi and V. Rajesh, "The plant leaf classification system using an optimum feature selection by grey wolf optimization," *Turkish J. Physiotherapy Rehabil.*, vol. 32, pp. 138–142, 2021.

[31] M. Nimbiwal and J. Vashishtha, "A novel hybrid grey wolf optimization algorithm using two-phase crossover approach for feature selection and classification," *Computación Sistemas*, vol. 25, no. 4, pp. 793–801, Dec. 2021.

[32] E.-S. M. El-Kenawy and M. Eid, "Hybrid gray wolf and particle swarm optimization for feature selection," *Int. J. Innov. Comput. Inf. Control*, vol. 16, no. 3, pp. 831–844, 2020.

[33] S. Sankhwar, D. Gupta, K. C. Ramya, S. S. Rani, K. Shankar, and S. K. Lakshmanaprabu, "Improved grey wolf optimization-based feature subset selection with fuzzy neural classifier for financial crisis prediction," *Soft Comput.*, vol. 24, no. 1, pp. 101–110, Jan. 2020.

[34] H. Almazini and K. Ku-Mahamud, "Grey wolf optimization parameter control for feature selection in anomaly detection," *Int. J. Intell. Eng. Syst.*, vol. 14, no. 2, pp. 474–483, Apr. 2021.

[35] C. Zhang, W. Wang, and Y. Pan, "Enhancing electronic nose performance by feature selection using an improved grey wolf optimization based algorithm," *Sensors*, vol. 20, no. 15, p. 4065, Jul. 2020.

[36] Q. M. Alzubi, M. Anbar, Z. N. M. Alqattan, M. A. Al-Betar, and R. Abdullah, "Intrusion detection system based on a modified binary grey wolf optimisation," *Neural Comput. Appl.*, vol. 32, no. 10, pp. 6125–6137, May 2020.

[37] 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.

[38] 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.

[39] X. Zhao, X. Zhang, Z. Cai, X. Tian, X. Wang, Y. Huang, H. Chen, and L. Hu, "Chaos enhanced grey wolf optimization wrapped ELM for diagnosis of paraquat-poisoned patients," *Comput. Biol. Chem.*, vol. 78, pp. 481–490, Feb. 2019.

[40] P. Sharma, S. Sundaram, M. Sharma, A. Sharma, and D. Gupta, "Diagnosis of Parkinson's disease using modified grey wolf optimization," *Cogn. Syst. Res.*, vol. 54, pp. 100–115, May 2019.

[41] N. P. Nirmala Sreedharan, B. Ganesan, R. Raveendran, P. Sarala, B. Dennis, and R. Boothalingam, "Grey wolf optimisation-based feature selection and classification for facial emotion recognition," *IET Biometrics*, vol. 7, no. 5, pp. 490–499, Sep. 2018.

[42] J. Too, A. Abdullah, N. M. Saad, N. M. Ali, and W. Tee, "A new competitive binary grey wolf optimizer to solve the feature selection problem in EMG signals classification," *Computers*, vol. 7, no. 4, p. 58, Nov. 2018.

[43] J.-S. Wang and S.-X. Li, "An improved grey wolf optimizer based on differential evolution and elimination mechanism," *Sci. Rep.*, vol. 9, no. 1, pp. 1–21, May 2019.

[44] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, Apr. 1997.

[45] Q. Al-Tashi, H. Md Rais, S. J. Abdulkadir, S. Mirjalili, and H. Alhussian, "A review of grey wolf optimizer-base feature selection methods for classification," in *Evolutionary Machine Learning Techniques: Algorithms and Applications*. 2020, pp. 273–286.

[46] S. B. Imandoust and M. Bolandraftar, "Application of k-nearest neighbor (KNN) approach for predicting economic events: Theoretical background," *Int. J. Eng. Res. Appl.*, vol. 3, pp. 605–610, Sep. 2013.

[47] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y.-P. Chen, A. Auger, and S. Tiwari, "Problem definitions and evaluation criteria for the CEC 2005 special session on realparameter optimization," KanGAL, Tech. Rep. 2005005, 2005, no. 2005, p. 2005.

[48] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: A nature-inspired algorithm for global optimization," *Neural Comput. Appl.*, vol. 27, no. 2, pp. 495–513, Feb. 2016.

[49] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems," *Appl. Math. Comput.*, vol. 188, no. 2, pp. 1567–1579, May 2007.

[50] 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.

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

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

[53] K. K. Ghosh, S. Ahmed, P. K. Singh, Z. W. Geem, and R. Sarkar, "Improved binary sailfish optimizer based on adaptive β-hill climbing for feature selection," *IEEE Access*, vol. 8, pp. 83548–83560, 2020.

TAMARA AMJAD AL-QABLAN received the B.Sc. degree in computer science from Yarmouk University, in 2001, and the M.Sc. degree in computer science from Al Al-Bayt University, in 2004. She is currently pursuing the Ph.D. degree in the field of artificial intelligence with the School of Computer Sciences, Universiti Sains Malaysia. Her research interests include optimization, data mining, machine learning, and deep learning.

MOHD HALIM MOHD NOOR received the B.Eng. degree (Hons.), in 2004, the M.Sc. degree, in 2009, and the Ph.D. degree in computer systems engineering from The University of Auckland, New Zealand, in 2017. He is currently a Senior Lecturer with the School of Computer Sciences, Universiti Sains Malaysia. His research interests include machine learning, deep learning, computer vision, and pervasive computing.

MOHAMMED AZMI AL-BETAR (Member, IEEE) received the Ph.D. degree in artificial intelligence from the University of Science, Malaysia (USM), in 2010. He was a Postdoctoral Research Fellow with USM for three years and was invited as a Visiting Researcher two times. He has been a full-time Faculty Member with the Master of Artificial Intelligence (MSAI) Program, Ajman University, since 2020. He is currently the Head of the Artificial Intelligence Research Center (AIRC), Ajman University. He is also the Head of the Evolutionary Computation Research Group (ECRG). He has more than 15 years of teaching experience in higher education institutions. He has taught several courses in computer science and artificial intelligence fields. He has published more than 175 scientific publications in high-quality and well-reputed journals and conferences. His main research interests include scheduling and optimization. He ranked in Stanford's study of the world's top 2% scientists.

AHAMAD TAJUDIN KHADER received the B.Sc. and M.Sc. degrees in mathematics from Ohio University, USA, in 1982 and 1983, respectively, and the Ph.D. degree in computer science from the University of Strathclyde, U.K., in 1993. He is currently a Honorary Professor with the School of Computer Sciences, Universiti Sains Malaysia, where he is also the Dean. His research interests include optimization and scheduling.