Received 20 June 2022, accepted 3 July 2022, date of publication 7 July 2022, date of current version 18 July 2022.
Digital Object Identifier 10.1109/ACCESS.2022.3189476

RESEARCH ARTICLE

Multiobjective Harris Hawks Optimization With Associative Learning and Chaotic Local Search for Feature Selection

YOUHUA ZHANG[1,2], YUHE ZHANG[1,2], CUIJUN ZHANG[2], AND CHONG ZHOU[1,2]
[1]School of Information Engineering, Hebei GEO University, Shijiazhuang 050031, China
[2]Laboratory of Artificial Intelligence and Machine Learning, Hebei GEO University, Shijiazhuang 050031, China

Corresponding author: Chong Zhou (zhouchong@hgu.edu.cn)

This work was supported in part by the Natural Science Youth Foundation of Hebei Province under Grant F2019403207, in part by the PhD Research Startup Foundation of Hebei GEO University under Grant BQ2019055, in part by the Open Research Project of the Hubei Key Laboratory of Intelligent Geo-Information Processing under Grant KLIGIP-2021A06, in part by the Fundamental Research Funds for the Universities in Hebei Province under Grant QN202220, in part by the Guangxi Natural Science Fund General Project under Grant 2021GXNSFAA075029, and in part by the Guangxi Science and Technology Base and Talent Project under Grant Guike AD20159036.

ABSTRACT In the classification problem, datasets often have a large number of features, but not all features are useful for classification. A lot of irrelevant features may even reduce the performance. Feature selection is to remove irrelevant features by minimizing the number of the feature subset and minimizing the classification error rate.So it can be regarded as a multi-objective optimization problem. Because of its simple structure and easy implementation, Harris Hawks Optimization algorithm (HHO) is widely employed in optimization problems. In this paper, the multi-objective HHO is applied to address the feature selection problem. In order to improve the search ability of the algorithm, associative learning, grey wolf optimization and chaotic local search are introduced into it. An external repository is used to save non-dominant solution set. The results of feature selection on the sixteen University of California Irvine (UCI) datasets show that the proposed method can effectively remove redundant features and improve the classification performance of the algorithm.

INDEX TERMS Multi-objective Harris Hawks optimization, feature selection, associative learning, chaotic local search.

I. INTRODUCTION

Classification is an important task in machine learning and data mining. The goal of classification problem is to get a model based on the training set by learning and use this model to predict the unknown class in the test set. Many practical problems in real world are considered as classification problem, such as image analysis, medical care and statistical problems. There are a lot of features in the real data sets, including relevant, irrelevant and redundant features, which leads to a large search space, named ''the curse of dimensionality'' [1]. In order to reduce the feature dimension, irrelevant and redundant features need be eliminated. Feature selection can reduce the dimensionality of the data and improve the classification performance by reducing or eliminating redundant and irrelevant features [2].

Feature selection selects the most useful (relevant) features to create a better performance model. But it is not easy to select only relevant features in practical application, due to the complex interaction between features. Feature interaction existing frequently in many areas is two-way, three-way or complex multiway interactions among features [3]. The relevant features may improve classification performance, while irrelevant features may be redundant features and reduce classification performance. There will be 2^n possible feature subsets for an n-dimensional dataset. It is impossible to get all solutions for a large n. So feature selection is an NP-Hard problem [4], which cannot be solved by exhaustive approaches in most cases. A variety of search approaches are proposed to solve the problem of feature selection, such as sequential forward selection [5] (SFS)

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

and sequential backward selection [6] (SBS). However, all of these approaches have drawbacks, which may lead to premature convergence or high computational complexity. In order to better solve the problem of feature selection, an efficient global search technology is needed. Evolutionary computation (EC) technology is well known for their global search ability [7]. The most commonly used EC technologies in feature selection include Genetic Algorithm (GA) [8], Particle Swarm Optimization (PSO) [9]–[12], Grey Wolf Optimization (GWO) [13], [14], Artificial Bee Colony (ABC) [3], [15], Whale Optimization Algorithm (WOA) [16], [17] and Salp Swarm Algorithm (SSA) [18], [19].

There are two conflict objectives in feature selection, which are to minimize the size of feature subset and maximize the classification accuracy. Therefore, feature selection can be considered as multi-objective optimization problem. EC technologies particularly are good at dealing with multi-objective optimization problem, because their population-based search mechanism can produce multiple trade-off solutions in a single run. So some EC technologies are employed to handle multi-objective feature selection problem, such as GA, PSO, GWO and ABC in [8], [9], [13], [15]. No-Free-Lunch (NFL) theorem asserts that there is not a good enough optimization algorithm to solve all optimization problems, which indicates that the current feature selection approaches may have performance degradation on some problems. Harris Hawks optimization [20] (HHO) which simulates the foraging behavior of Harris Hawks is a new global search algorithm and effective on the numerical optimization problems and several real-world engineering problems. In addition, in the field of feature selection, some new researches about HHO have emerged. For example, In [21], a novel HHO, named IHHO, is proposed by embedding the salp swarm algorithm (SSA) into the original HHO to improve the search ability of the optimizer and expand the application fields. To enhance feature selection of Harris Hawks Optimization, the novel control factor and Brownian motion are employed in [22]. In [23], the paper introduces an improved HHO (IHHO) by utilizing elite opposite-based learning and proposing a new search mechanism. In [24], to overcome local optima and population diversity drawbacks, Chaotic HHO is proposed, in which the chaotic maps and simulated annealing algorithm are applied to enhance the population diversity and improve HHO exploitation, respectively. However, these researches are singe-objective based approaches and these algorithms easily converge to local optima. Furthermore, the results obtained by these algorithms are affected by the parameter a in the evaluation function, while in the multi-objective based approaches, parameter a is not used. So HHO is employed to solve feature selection in the paper including single-objective and multi-objective.

Based on the above-mentioned motivation, the main goal of this paper is to improve a feature selection approach based on HHO. Compared with using all features, the solution obtained by this approach should get smaller feature subset size and classification error rate. In order to achieve this goal, a multiobjective HHO algorithm with associative learning and chaotic local search (MOHHOAC) is proposed to solve feature selection problem. Search strategy of the GWO also is adopted to enhance performance of MOHHOAC. In addition, an external repository is used to save nondominated solution set. To maintain diversity of non-dominated solution set, grid strategy is introduced into MOHHOAC. Finally, a comprehensive experiment is designed to verify performance of HHO including singleobjective and multiobjective. Single-objective HHO is compared with two traditional approaches, three single objective approaches, and MOHHOAC is compared with four wellknown multi-objective feature selection approaches on the 16 benchmark datasets including various features, classes and instances. Multi-objective approaches also is compared with single-objective approaches with respect to classification error rate. Experiment results show that MOHHOAC can present promising performance to solve the feature selection problem.

Specifically, the research goals are as follows:

1) The performance of singleobjective HHO approach in reducing the size of feature subset and improving classification performance is analysised versus two traditional approaches and three singleobjective evolutionary algorithm on 16 UCI datasets.
2) A multi-objective HHO with associative learning and chaotic local search is proposed to solve feature selection problem.
3) The performance of the proposed MOHHOAC is evaluated on 16 UCI datasets to research its efficiency for the feature selection.

The organization of the rest of the paper is as follows. The knowledge of standard HHO algorithm, basic conception of multi-objective optimization, Harris hawks optimizer based approaches, the recent research on feature selection and multi-Objective Grey Wolf Opimization are introduced in Section II. Then, the proposed feature selection approach based on HHO algorithm is detailed description in Section III. The experimental design is offered in Section IV and the experimental results and discussion are presented in Section V. Finally, summarization and the future development trend are showed in Section VI.

II. LITERATURE REVIEW

In this section, the standard HHO algorithm is described, the definition of multi-objective optimization problem is given, and the literature about feature selection is briefly reviewed.

A. HHO ALGORITHM

HHO algorithm is a swarm intelligence algorithm. The main idea of HHO algorithm is inspired by the cooperative behavior of Harris Hawks when they hunt the escaped prey (mostly rabbits) [25]. HHO is a population-based and gradient-free optimization technique; Hence, it can be applied to any optimization problems subjecting to a proper formulation. HHO algorithm is divided into exploration and exploitation phase. The detailed introduction of each phase is listed as follows.

1) EXPLORATION PHASE

In the HHO algorithm, the Harris Hawks perch randomly on some locations and wait to find rabbit based on two strategies Eq.(1).

$$X(t+1)=\begin{cases}X_{\text{rand}}(t)-r_1\left|X_{\text{rand}}(t)-2r_2X(t)\right| & q\geq 0.5\\ (X_{\text{rabbit}}(t)-X_m(t)) & \\ \quad -r_3\left(LB+r_4(UB-LB)\right) & q<0.5\end{cases} \tag{1}$$

where $X(t+1)$ is the position vector of hawks in the next iteration t, $X_{rabbit}(t)$ is the position of rabbit, $X(t)$ is the current position vector of hawks, r_1, r_2, r_3, r_4 and q are random numbers inside (0, 1), which are updated in each iteration, LB and UB are the lower and upper bounds of variables, $X_{rand}(t)$ is a randomly selected hawks in the current population, and $X_m(t)$ is the average position of the current population of hawks. The average position of hawks is obtained by Eq. (2).

$$X_m(t)=\frac{1}{N}\sum_{i=1}^{N}X_i(t) \tag{2}$$

where $X_i(t)$ represents the location of each hawk in iteration t and N indicates the total number of hawks.

2) TRANSITION FROM EXPLORATION TO EXPLOITATION

The HHO algorithm can transfer from exploration to exploitation, and then change different exploitation behaviors according to the escaping energy of rabbit. In the process of rabbit escaping, the energy will be reduced, and the energy of rabbit is simulated by Eq. 3.

$$E=2E_0\left(1-\frac{t}{T}\right) \tag{3}$$

where E indicates the escaping energy of the rabbit, T is the maximum number of iterations, and $E0$ denotes the initial state of energy.

3) EXPLOITATION PHASE

During the phase, the Harris Hawks will pounce on rabbit found in the previous phase, but the rabbit will try to escape. Therefore, there will be different chasing styles in real life. According to the escaping behaviors of rabbit and chasing strategies of the Harris Hawks, four possible strategies are proposed in the HHO to model the attacking stage [20]. They are soft besiege, hard besiege, soft besiege with progressive rapid dives and hard besiege with progressive rapid dives. The following are four strategies.

a: SOFT BESIEGE

This behavior is modeled by Eq. (4) and Eq. (5).

$$X(t+1)=\Delta X(t)-E\left|JX_{\text{rabbit}}(t)-X(t)\right| \tag{4}$$
$$\Delta X(t)=X_{\text{rabbit}}(t)-X(t) \tag{5}$$

where $X(t)$ is the difference between the rabbit position vector and the current position in iteration t, $J=2(1-r5)$ indicates the random jump strength of the rabbit escaping process, r_5 is a random number inside (0, 1). The J value will randomly change in each iteration to simulate the nature of rabbit motions.

b: HARD BESIEGE

In this situation, the current positions are updated by Eq. (6).

$$X(t+1)=X_{\text{rabbit}}(t)-E|\Delta X(t)| \tag{6}$$

c: SOFT BESIEGE WITH PROGRESSIVE RAPID DIVES

In order to perform a soft besiege, we supposed that the hawks can decide their next action according to Eq. (7).

$$Y=X_{\text{rabbit}}(t)-E\left|JX_{\text{rabbit}}(t)-X(t)\right| \tag{7}$$

We supposed that hawks will dive based on the LF-based patterns by Eq. (8).

$$Z=Y+S*LF(D) \tag{8}$$

where D is the dimension of the problem, S denotes a random vector by size $1*D$ and LF is the levy flight function, which is calculated by Eq. (9).

$$LF(x)=0.01*\frac{u*\sigma}{|v|^{\frac{1}{\beta}}},\quad \sigma=\left(\frac{\Gamma(1+\beta)*\sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right)*\beta*2^{\left(\frac{\beta-1}{2}\right)}}\right)^{\frac{1}{\beta}} \tag{9}$$

where u, v are random values inside (0, 1), β is a default constant and is set to 1.5. Therefore, the final strategy for updating the positions of hawks during the soft besiege phase can be executed by Eq. (10).

$$X(t+1)=\begin{cases}Y & \text{if } F(Y)<F(X(t))\\ Z & \text{if } F(Z)<F(X(t))\end{cases} \tag{10}$$

where Y and Z are calculated by Eq. (7) and Eq. (8), respectively.

d: HARD BESIEGE WITH PROGRESSIVE RAPID DIVES

Eq. (11) is performed in hard besiege condition.

$$X(t+1)=\begin{cases}Y & \text{if } F(Y)<F(X(t))\\ Z & \text{if } F(Z)<F(X(t))\end{cases} \tag{11}$$

where Y and Z are calculated by Eq. (12) and Eq. (8).

$$Y=X_{\text{rabbit}}(t)-E\left|JX_{\text{rabbit}}(t)-X_m(t)\right| \tag{12}$$

where $X_m(t)$ is obtained by Eq. (2).

B. MULTI-OBJECTIVE OPTIMIZATION

Many problems involve two or more than two conflicting objectives, which are called multi-objective optimization problems. Without loss of generality, it can be formulated as follows [26].

$$\begin{aligned}&\text{MAX: } F(x)=f_1(x),f_2(x),\ldots f_o(x)\\ &\text{Subject to: } g_i(x)\geq 0,\quad i=1,2,\ldots,m\\ &\qquad h_i(x)=0,\quad i=1,2,\ldots,p\\ &\qquad L_i\leq x_i\leq U_i\quad i=1,2,\ldots,n\end{aligned} \tag{13}$$

where n is the number of variables, o indicates the number of objective functions, m denotes the number of inequalities constraint, p represents the number of equalities constraint. g_i means the ith inequality constraint, hi indicates the ith equality constraint. L_i, U_i are the boundaries of ith variable.

In the single-objective optimization problem, it is easy to obtain the optimal solution due to only one objective function. However, in the multi-objective optimization problems, the optimal solution can't be obtained by simple comparison due to the conflict between objectives and the phenomenon of incomparability. In this case, a solution is superior to another solution if and only if it shows better or equal values on all objective functions and has better values on at least one objective function. Therefore, the following concepts about multi-objective optimization are introduced.

1) Pareto Dominance: Supposed that there are two vectors such as: $x = (x_1, \ldots, x_k)$ and $y = (y_1, \ldots, y_k)$. Vector x dominates vector y(denote as $x \succ y$)

$$\begin{aligned} &\forall i \in \{1, \ldots, k\}, \quad f(x_i) \geq f(y_i) \\ &\wedge \exists i \in \{1, \ldots, k\} : f(x_i) > f(y_i) \end{aligned} \tag{14}$$

2) Pareto Optimality: A solution $x \succ y$ is called Pareto optimal, if and only if

$$\nexists y \in X \mid F(y) \succ F(x) \tag{15}$$

3) Pareto Optimality set: The set of Pareto optimal solutions is Pareto optimal set.

$$P_s = \{x \in X \mid x \text{ is Pareto optimal}\} \tag{16}$$

4) Pareto front: In Pareto optimality set, the set containing objective value corresponding to the Pareto optimal solution is called the Pareto front.

$$P_f = \{F(x) \mid x \in P_s\} \tag{17}$$

C. EXISTING FEATURE SELECTION APPROACHES

Feature selection problem is to select relevant features, eliminating irrelevant and redundant features, and reduce data dimension. Feature selection approaches are divided into three types: filter approach, wrapper approach and embedded approach. In filter approach, feature selection depends on data features, and does not use learning algorithm [27]. The filter approach has lower computational complexity; the wrapper approach uses classification algorithm and selects feature subsets according to the classification performance of the algorithm; The embedded approach obtains the optimal subsets in the learning process, which depends on the performance of the classifier. The computational complexity of embedded approach is lower than wrapper approach, but the embedded approach is more complex, and the model is not easy to modify [28]. The wrapper approach is widely used.

1) SINGLE OBJECTIVE EVOLUTIONARY COMPUTATION-BASED APPROACHES

In order to solve the defects of traditional feature selection approaches, researchers use evolutionary computation technology to solve the problem of feature selection. It includes GA [8], GP [29], PSO [9] and ABC [15]. The feature selection approach based on GA is proposed by Raymer *et al.* and is better than SFFS [30]. Oh *et al.* [31] implemented the hybridized GA (HGA) by embedding local search operation. The experimental results show that the performance of HGA is better than the standard GA. Chen *et al.* [32] proposed the feature selection approach based on multi-swarm PSO using the classification accuracy and the F-score in a weighted manner. HHO algorithm is very suitable for solving optimization problems and it is improved to solve feature selection. Zhang *et al.* [33] proposed an effective feature selection method based on firefly algorithm (FFA), where three aspects which are an indicator based on the return-cost, a Pareto dominance-based strategy and a binary movement operator based on the return-cost attractiveness are employed to enhance the capability of preventing premature convergence. Xue *et al.* [34] proposed a self-adaptive particle swarm optimization (SaPSO) algorithm for feature selection with large-scale features, where an encoding scheme and a typical self-adaptive mechanism are proposed. The experimental results show that the SaPSO algorithm is suitable for solving feature selection problems, particularly large-scale feature selection problems.

2) MULTI-OBJECTIVE EVOLUTIONARY COMPUTATION-BASED APPROACHES

There are two conflicting objectives in feature selection, which are regarded as multi-objective problems. The multi-objective optimization algorithm based on evolutionary computation is used to solve the problem of feature selection, and the classification error rate and the number of features is regarded as the two objectives. Hamdani *et al.* [35] proposed a genetic algorithm based on non-dominated sorting II (NSGA2). Waqas *et al.* [36] proposed a multi-objective GA based on wrapper approach, using decision tree as the classifier. Xue *et al.* [9] proposed a multi-objective PSO feature selection approach based on the wrapper approach, which was inspired by crowding distance, non-dominated sorting and mutation. The experimental results show that the performance of this approach is superior to the NSGA2. Although there are many feature selection approaches based on EC, most of them are single objective, which only take classification accuracy as objective. At present, there are few literatures regarding feature selection as multi-objective problem. Recently, it is found that the HHO algorithm is used in multi-objective feature selection. Dabba *et al.* [37] introduced a multi-objective binary Harris Hawks optimization (MOBHHO) for gene selection, where SVM with LOOCV classifier and KNN with K-fold classifier are used as two fitness functions to solve competing objectives. Dokeroglu *et al.* [38] proposed a new multi-objective HHO algorithm for binary classification problem and a new discrete exploration and exploitation operators for the hunting patterns of the hawks is developed. However, these researches are worked for binary classification problems or gene selection. In this paper, we solve feature selection problem with different feature numbers and class numbers.

D. HARRIS HAWKS OPTIMIZER BASED APPROACHES

Harris hawks optimizer (HHO) is a recently developed meta-heuristic algorithm that has previously shown excellent performance on function optimization and real world applications. However, HHO still has some drawbacks such as trapped in local optima and low solution precision in solving more complicated optimization problems. To overcome these drawbacks, some improved HHOs are proposed. Hao *et al.* [39] proposed an enhanced DE driven multi-population HHO (CMDHHO) algorithm, in which chaos strategy, topological multipopulation strategy, and differential evolution (DE) strategy are integrated to improve the performance of HHO. Long *et al.* [40] presented an improved HHO named LIL-HHO, where a modified escaping energy strategy based on sine function, the personal best position of each hawk, and lens-imaging learning (LIL) operator are introduced to achieve a good transition, modify the position search equation and maintain the population diversity respectively. Additionally, LIL-HHO also is employed to solve feature selection problem. Hu *et al.* [41] proposed an improved binary Harris hawk optimization (HHO) algorithm in combination with a kernel extreme learning machine to determine the factors that play a decisive role in the early recognition and discrimination of COVID-19 severity. This method uses specular reflection learning to improve the original HHO algorithm, called HHOSRL. The experimental results show that the proposed model provides an effective strategy for accurate early assessment of COVID-19 and distinguishing disease severity. Suresh *et al.* [42] proposed a Chaotic Multi Verse Harris Hawks Optimization algorithm based Deep Kernel Machine Learning Classifier (CMVHHO-DKMLC) method for medical diagnostics, by which the feature selection (FS) is done for finding ideal feature subset of medical documents. Li *et al.* [43] presented a called RLHHO, where two novel strategies are integrated into the original HHO to enhance exploration and exploitation capabilities. An exploration strategy based on logarithmic spiral and opposition-based learning and a local search technique for Rosenbrock Method (RM) are proposed to improve the exploration ability and enhance algorithm's local search capability of HHO. In [44], a hybrid optimization method, called SCHHO, is proposed for numerical optimization and feature selection, which fuses sine-cosine algorithm into the standard HHO to enhance exploitation by dynamically adjusting candidate solutions. SCHHO shows promising performance on the sixteen datasets with low and high-dimensions exceeding 15000 attributes. Dokeroglu *et al.* [45] proposed a new multiobjective HHO algorithm for the solution of the well-known binary classification problem in which a new discrete exploration (perching) and exploitation (besiege) operators for the hunting patterns of the hawks is developed. Moreover, it is applied to a real-world dataset, Coronavirus disease (COVID19) dataset. In [46], to automatically detect COVID-19 in radiological images, a two-stage pipeline composed of feature extraction followed by feature selection is proposed. HHO with Simulated Annealing and Chaotic initialization is used to eliminate the non-informative and redundant features. The experimental results show the proposed algorithm can decrease the number of features selected by around 75%, which is better than other algorithm. Ridha *et al.* [21] proposed a Boosted HHO (BHHO) to achieve a more stable model and effectively estimate the parameters of the single diode PV model. In the BHHO, random exploratory steps of evolution inspired by the flower pollination algorithm (FPA) and a powerful mutation scheme of the differential evolution (DE) with 2-Opt algorithms are combined to the standard HHO to accelerate the convergence rate but also assist it in exploring new regions. In addition, Kuolu *et al.* [47] proposed a Multi-Objective Harris Hawk Optimization (MOHHO) for multi-objective optimization problem, where an archive repository is added to the HHO algorithm to save and retrieve the Pareto optimal results.

E. MULTI-OBJECTIVE GREY WOLF OPIMIZATION

In the Multi-Objective Grey Wolf Optimizer (MOGWO) [26], a fixed-sized external archive is integrated to the GWO for saving and retrieving the Pareto optimal solutions. This archive has been employed to define the social hierarchy and simulate the hunting behavior of grey wolves in multi-objective search spaces. In order to perform multi-objective optimization by GWO, MOGWO integrates two new components. The first one is an archive, which is responsible for storing non-dominated Pareto optimal solutions obtained so far. The archive adopts grid mechanism to maintain diversity performance. In the method, the objective space is divided into several grids. These grids are adjusted according to the solution in the archive, over the course of the iteration. If a newly obtained solution lies outside the grid, all the grid locations should be recalculated to cover it. If a new solution lies within the grid, it becomes the portion of the grid with the lowest number of solutions. To enhance convergence performance, non-dominated solution obtained so far should be put into the archive, during the course of evolution. However, the archive has a maximum number of members. When a non-dominated solution obtained enters the archive, it needs to compare against the archive residents. If the new solution is dominated one or more members within the archive, the dominated members are removed from the archive and the new solution should be inserted into the archive. If the new solution is nondominated with solutions in the archive, the new solution should be added to the archive. If the archive is full, the grid mechanism should be called to redivide the grid of the objective space and find the most crowded grid to remove one of its solutions. Then, the new solution should be inserted to the least crowded segment in order to improve the diversity of the final approximated Pareto optimal front. If the new solution is dominated by at least one member of the archive, the solution should not be allowed to put into the archive.

The second component is a leader selection mechanism that assists to choose alpha, beta, and delta solutions as the leaders of the hunting process from the archive. In order to explore objective space freetly with the hope to find a solution close to the global optimum, three best solutions are commonly selected as leaders to guide the other agents. However, in the multi-objective space, the best solution cannot easily be compared according to Pareto dominance. In order to select best solution, a leader selection mechanism which is similar to that of the MOPSO is employed. The leader selection mechanism is done by a roulette-wheel method with the following probability for each grid:

$$P_i = \frac{c}{N_i} \tag{18}$$

where c is a constant number and N is the number of obtained Pareto optimal solutions in the i-th grid. From Eq.(18), it may be seen that less crowded grids have higher probability as new leader from the archive.

III. THE PROPOSED ALGORITHM

As mentioned in above sections, feature selection can be regarded as a multi-objective problem, which has two conflicting objectives: minimizing the size of feature subsets and maximizing classification accuracy. HHO algorithm is a new kind of swarm intelligence algorithm, which hasn't been widely used. In this paper, we try to improve the HHO algorithm and use it to solve the problem of multi-objective feature selection. The standard HHO algorithm is used to deal with single objective problems. The standard HHO algorithm is improved to solve multi-objective problems. In this paper, the concept of dominance relationship and external archive in MOPSO is introduced into HHO algorithm to get multi-objective HHO algorithm. To solve feature selection problem, solution representing feature subsets in the algorithm is within the range of 0 and 1. If a dimension in a solution is greater than 0.5, the corresponding feature is selected; otherwise, it is not selected. In order to express clearly, firstly, the whole framework of the algorithm is described (Algorithm 1) and the flowchart of the algorithm is showed in Figure 1, and then the multi-objective HHO algorithm with Grey wolf optimizer, associative learning and chaotic local search is described in detail.

A. GREY WOLF OPTIMIZER

Grey wolf optimizer [26] imitates the social hierarchy and the hunting technique of grey wolves. Hunting process is a strong search behavior because of considering information about the location of the optimum solution. Hunting process is guided by the three best solutions considered as α, β and δ. In order to improve exploration of HHO, hunting process is combinated with HHO. The mathematical model of hunting process is described as following:

$$\begin{aligned} D_\alpha &= \left|C_1 \cdot X_\alpha - X\right|, \\ D_\beta &= \left|C_2 \cdot X_\beta - X\right|, \\ D_\delta &= \left|C_3 \cdot X_\delta - X\right| \end{aligned} \tag{19}$$

$$\begin{aligned} X_1 &= X_\alpha - A_1 \cdot D_\alpha, \\ X_2 &= X_\beta - A_2 \cdot D_\beta, \\ X_3 &= X_\delta - A_3 \cdot D_\delta \end{aligned} \tag{20}$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3} \tag{21}$$

where α is the fittest solution. β and δ are the second and third best solution respectively.However, in the multi-objective optimization, these best solutions are selected from the external repository using leader selection mechanism [26]. C_1, C_2, C_3, A_1, A_2 and A_3 are coefficients updated by Eqs. (22, 23). r_1 and r_2 are two random vectors in [0, 1]. Components of a are linearly decreased from 2 to 0 over the course of iterations.

$$A_i = 2ar_1 - a \quad i = \{1, 2, 3\} \tag{22}$$

$$C_i = 2 \cdot r_2 \quad i = \{1, 2, 3\} \tag{23}$$

B. ASSOCIATIVE LEARNING

Associative learning is a new tactic which can improve the exploratory performance of algorithm [48]. In order to accelerate speed of the HHO, during the learning phase, associative learning is combined to the proposed HHO. Therefore, the new position of hawk is computed as follows:

$$\begin{aligned} X(t+1) = X(t) &+ 0.001\, G\left(X(t) - lb, ub - X(t)\right) \\ &+ S_1 r_1 \left(X_r(t) - X(t)\right) + S_2 r_2 \left(X_p(t) - X(t)\right) \end{aligned} \tag{24}$$

where r_1 and r_2 are two random numbers in (0, 1), X_r indicates a random solution selected from the previous populations, X_p is the best solution, which is selected from the external repository using leader selection mechanism. S_1 and S_2 denote adaptive cognitive and social factors, respectively. The values of S_1 and S_2 are updated in each iteration using:

$$\mathrm{S}_1 = (1 - \mathrm{t/T}) \tag{25}$$

$$\mathrm{S}_2 = 2(\mathrm{t/T}) \tag{26}$$

With iteration increasing, the value of S_1 reduces from 1 to 0, to decrease the impact of the random leader on the search pattern of hawks. However, the value of social factors S_2 raises to improve the impact of the random hawks. The design rule can enhance hawks to gradually approach to the high-quality solution.

C. CHAOTIC LOCAL SEARCH

In order to further refine the quality of solution, chaotic local search is embedded into the proposed algorithm. The local search is to find better solution near the current best solution. Due to randomicity and ergodicity of the chaotic sequence, Chaotic local search is very beneficial to further improve the quality of one solution by generating new solution around the best hawks. Hence, a new individual $X(t+1)$ is generated by chaotic local search presented in Eq.(27) at each

Algorithm 1: Framework of the Proposed Algorithm

Initial the population X, external repository REP and parameter;
Evaluate the initial population X;
Determine Pareto dominance of the population X;
Move nondominated solutions to REP;
Create grid for REP;
while $t < MaxIt$ **do**
  Select the leader of hawks by roulette wheel selection;
  Update parameter $a = 2 - t * (2/MaxIt)$;
  for *each hawks* X_i *do* **do**
    Update the initial energy E_0 by $E_0 = 2 * rand() - 1$ and Jump strength $J = 2 * (1 - rand())$;
    Update E using the Eq.(3);
    if $|E| >= 1$ **then**
      if $q < 0.5$ **then**
        Update the location vector using $Eq.$(21);
      end if
      else
        Update the location vector using Eq. (24);
      end if
    end if
    if $|E| < 1$ **then**
      if $r >= 0.5$ *and* $|E| >= 0.5$ **then**
        Update the location vector using Eq.(4);
      end if
      if $r >= 0.5$ *and* $|E| < 0.5$ **then**
        Update the location vector using Eq.(6);
      end if
      if $r < 0.5$ *and* $|E| >= 0.5$ **then**
        Update the location vector using Eq. (10);
      end if
      if $r < 0.5$ *and* $|E| < 0.5$ **then**
        Update the location vector using Eq.(27);
      end if
    end if
  end for
  Add new solutions to Population;;
  Determine dominance relationship to obtain nondominated solutions;
  Merge REP with the nondominated solutions and keep only non-dominated solutions in the REP;
  Update the grid;
  if REP is full, remove extra solutions from REP;
  $t \leftarrow t + 1$;
end while

generation.

$$z_{m+1} = 4 * z_m * (1 - z_m)$$
$$X(t+1) = \begin{cases} X_{\text{best}} + \text{rand}_1 \cdot (2 \cdot z_m - 1), & \text{if } \text{rand}_2 < 1-(t/\text{MaxIter}) \\ X_{\text{best}}, & \text{otherwise} \end{cases} \tag{27}$$

where m is the chaotic iteration index, which equals the current iteration t here. z_m is the value of mth chaotic iteration, and its initial value z_0 is randomly generated in [0, 1]. t is the current iteration and *MaxIter* is the maximal iteration times. $rand_1$ and $rand_2$ are two random numbers in the range [0, 1]. X_{best} is the current best leader, which is selected from the external repository using leader selection mechanism [26].

D. EXTERNAL REPOSITORY

The main goal of the external repository is to keep a historical record of the nondominated solutions obtained in the search process. The external repository consists of two parts: the repository controller and the grid. First, the process of the controller is introduced. The repository controller is to decide whether a solution should be included in the external repository or not. The nondominated solutions obtained in each iteration will be compared with the solutions in the external repository. At the beginning of the search, the external repository is empty, and the current population nondominated solutions are accepted by the external repository. If a new solution is dominated by an solution in the external repository, the solution is automatically called. If a solution in the repository is dominated by a new solution, the solution is removed from the repository. If the external repository reaches its maximum capacity, removing solution process is called. The basic idea of the external repository is to use an external repository to store solutions, which are not dominated by the repository content. In order to maintain diversity, the grid is employed in the external repository [49].

E. THE COMPUTATIONAL COMPLEXITY

The complexity of the proposed algorithm (Algorithm 1) depends on mainly the determining the Pareto dominance in line 23 and line 247 and producing new solution from line 29 to line 245. The computational complexity of the determining Pareto dominance is $O(N * N * m)$, where N is the size of population and m is the number of objectives. The computational complexity of the producing new solution is $O(N * dim)$. the other steps in the Algorithm 1 is the create grid and remove extra solutions. Their computational complexity are $O(m * div)$ and $O(|REP| - N)$, where div is the number of grids per dimension and $|REP|$ is the size of the external repository. Therefore, the whole computational complexity of the Algorithm 1 is $O(N * N * m)$.

IV. EXPERIMENTAL SIMULATIONS

In this experiment, 16 datasets in UCI machine learning library [50] are selected, which include different feature numbers (from 4 to 617), class numbers (from 2 to 26) and sample numbers (from 32 to 7797), as shown in Table 1. *KNN* classifier is selected as the classifier in the algorithm, where K is set to 5. In the experiments, the samples in each dataset are randomly divided into two sets: 70% as the training set and 30% as the test set.

The proposed multi-objective HHO feature selection algorithm is compared with a variety of algorithms, including two traditional approaches, three single objective approaches and four multi-objective approaches. Two traditional approaches are linear forward selection (LFS) [51] based on SFS and greedy stepwise backward selection (GSBS) [52] based on SBS. Their performance is better than SFS and SBS and they are implemented in Waikato Environment for Knowledge Analysis (WEKA) platform [53]. The experiment of LFS and GSBS are completed via WEKA platform and 5*NN* is used to evaluate feature subset obtained by the approaches on test datasets.

The single objective feature selection approaches involves the standard GA [31], PSO [32] and HHO [20] algorithm, and the classification error rate and number of features (Eq. 28) are employed to evaluate the performance of the algorithm with the weight way determined by parameter a.

$$\text{fitness} = a\frac{\text{SubsetSize}}{\text{AllsetSize}} + (1-a)\text{ErrorRate} \tag{28}$$

where a is a constant within [0, 1], *SubsetSize* is the size of the feature subset, *AllsetSize* denotes the number of all features for the dataset and *ErrorRate* indicates the classification error rate of the feature subset. For the experiments, the parameter values of used algorithms are set as follows: the population sizes in each algorithms are set to 30; the maximum number of iterations take the value of 100. Crossover rate and Mutation rate used in GA are defined as 0.7 and 0.1, respectively. The parameters of PSO are Inertia Weight, Inertia Weight Damping Ratio, Personal Learning Coefficient and Global Learning Coefficient set to $w1 = 1$, $wdamp = 0.99$, $c1 = 2$ and $c2 = 2$. There are no extra parameters used in HHO.

The multi-objective approaches include: MOGWO [26], MOPSO [49], NSPSOFS [9] and NSGA2 [35]. These algorithms are improved to solve feature selection problems. *ErrprRate* and number of selected features are considered as two objective functions to evaluate performance of the multi-objective algorithms. For the experiments of multi-objective algorithms, the defined parameter values are as follows: the population size is empirically set to 30; the maximum number of iterations is empirically defined as 100; the parameters of MOPSO are employed as in [9] where $c1 = 1, c2 = 2$ and inertial weight = 0.5; the external repository size and the number of grids per dimension the are set to 100 and 7, respectively; the parameters of NSGA2 are selected according to [35] where crossover rate and mutation rate are set to 0.7 and 0.1; In the MOHHOAC and MOGWO, an external repository and grid technology are used where the external repository size and the number of grids per dimension are also set to 100 and 7. Taking the number of features and classification

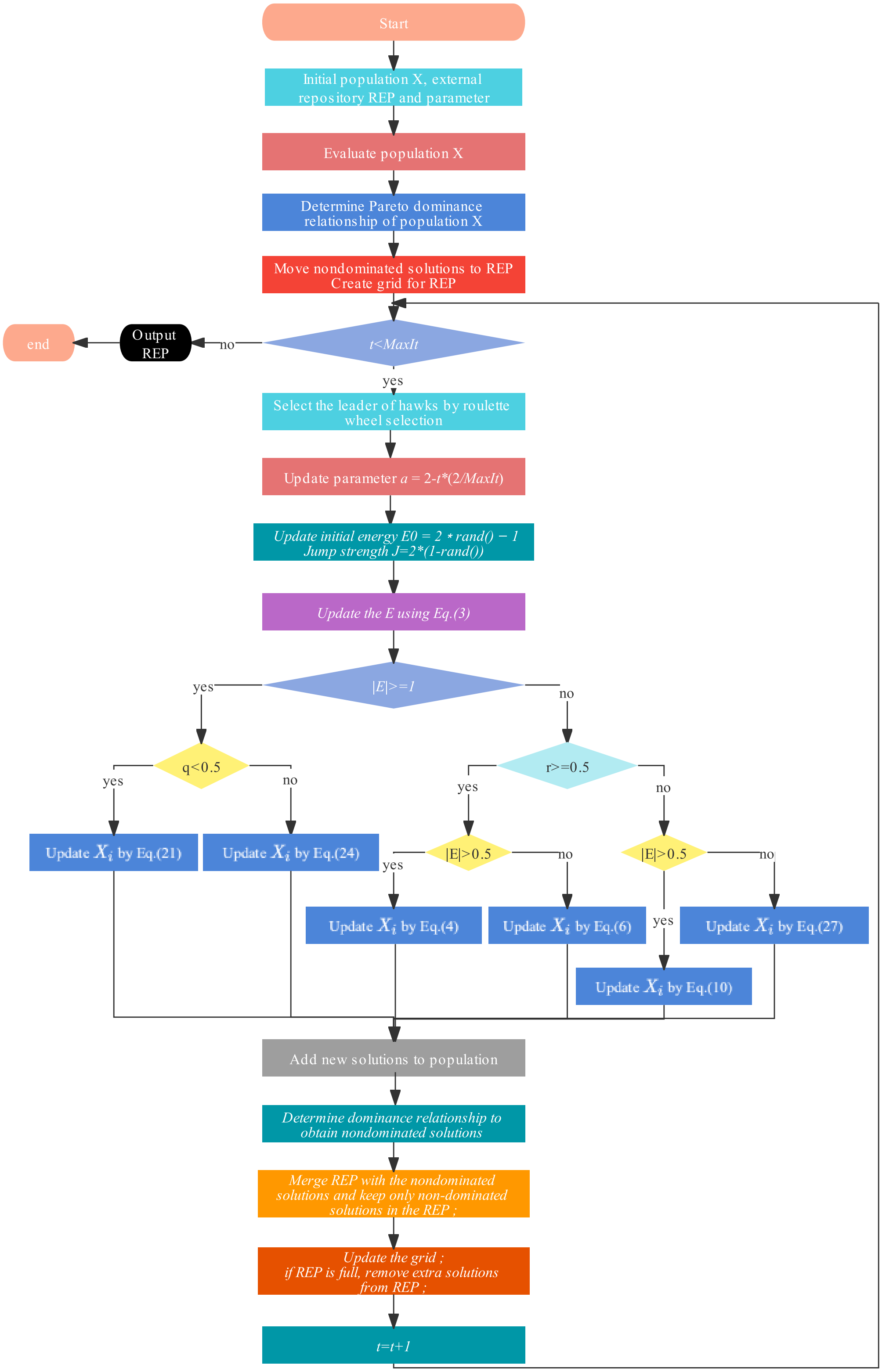

FIGURE 1. The flowchart of the MOHHOAC.

error rate as the optimization objective, the Pareto front is obtained, and the performance of the algorithm is evaluated.

Due to unknown of the true Pareto front, Hypervolume (HV) indicator is employed to measure the performance of the

TABLE 1. Description of datasets.

Datasets	Features	Classes	Samples
Iris	4	3	150
page_blocks	10	5	5473
Wine	13	2	178
Zoo	17	7	101
Audit_risk	18	14	776
Vehicle	18	4	846
German	24	2	1000
Ionosphere	34	2	351
Spect	23	2	267
Lungcancer	56	3	32
Movement_libras	90	15	360
Hillvalley_with_noise	100	2	1212
Hillvalley_without_noise	100	2	1212
Urban_land_cover	148	9	675
Musk1	166	2	476
Isolet5	617	26	7797

proposed algorithm. Its definition is listed as follows:

$$HV = \text{volume}\left(\mathrm{U}_{i=1}^{|P|} v_i\right) \tag{29}$$

Hypervolume indicator gives the volume of hypercube covered by the members of Pareto-solutions. $|P|$ is the size of Pareto-soultion set. v_i is the volume of ith Pareto solution and reference point. Each algorithm runs 30 times separately. Finally, the classification accuracy and feature subset size are obtained. The experimental results are presented in the fifth sections. The traditional approaches LFS and GSBS can get a unique feature subset on each data set; The single objective approaches get an optimal result after executing 30 times on each data set; The multi-objective algorithms get a set of feature subsets after executing on each data set. All the experimental results are analyzed and compared.

V. EXPERIMENTAL RESULTS

The experimental results are mainly divided into three parts: 1) the comparisons between single objective and traditional approaches; 2) the comparisons between multi-objective approaches; 3) the comparisons between multi-objective and single objective approaches. To have statistically sound conclusions, the Wilcoxon's rank-sum test for independent sample at a 0.05 significance level, which is a nonparametric statistical test method, is conducted to judge the significance of the results between two algorithms in terms of HV value in Table 4. Signs $+$, $-$ and $\sim$ indicate that the corresponding comparative algorithm is worse than, better than, and similar to MOHHOAC, respectively. Moreover, the Friedman test with Bonferroni-Dunn's procedure is employed to achieve the final ranking of different algorithms on the different type datasets, according to classification error rate in Table 5.

A. THE COMPARISONS BETWEEN SINGLE OBJECTIVE AND TRADITIONAL APPROACHES

The experimental results of this part are shown in Table 2, where the best results are highlighted in red block and the second good results are highlighted in blue block, including five single objective approaches PSO, GA, WOA-SA, GWO and HHO, and two traditional approaches LFS and GSBS. The classification error rate and the number of features are taken as the evaluation index and KNN algorithm is used as classification model. In single objective approaches, Eq. (28) is also used as the evaluation index.

From Table 2, single objective evolutionary algorithm can obtain better result than traditional approaches on the most datasets, which indicates that single objective evolutionary algorithms have stronger search performance. The traditional approaches only obtain the lowest error rate on the four datasets, Iris, Page_blocks, Musk1 and Vehicle, while on the number of selected features, they do not obtain fewer selected features. However, single objective evolutionary algorithm can obtain smaller number of features, but error rate is not the lowest. For 12 out of 16 datasets, single objective evolutionary algorithm can achieve the lowest classification error rate and number of features. The reason may be that single objective evolutionary algorithm considers both classification error rate and number of features. For single objective evolutionary algorithm, GWO can perform much better than HHO and PSO in terms of the classification error rate. According to classification error rate, HHO can obtain the best result on three datasets and second best result on two datasets, ranked thrid in the all algorithms, which HHO has stronger search ability than traditional approaches.

B. COMPARISONS BETWEEN MULTI-OBJECTIVE APPROACHES

Due to poor ability of HHO, an improved version of MOHHO, multi-objective harris hawks optimization with associative learning and chaotic local search, named MOHHOAC, is proposed. In order to verify ability of MOHHOAC, a comprehensive experiment is designed. Firstly, MOHHOAC is compared with the original MOHHO to prove the effectiveness of three strategies adopted in the proposed algorithm. Figure 2 and Table 3 show the results of non-dominated Pareto fronts and the average classification error rate obtained by MOHHOAC and MOHHO respectively. Then, MOGWO, NSGA2, MOPSO and NSPSOFS are employed to compare with MOHHOAC. Table 4 presents the overall results of using the HV indicator for performance comparisons between the multi-objective algorithms. Figure 3 shows the results of non-dominated Pareto fronts obtained by these algorithms.

1) COMPARISONS BETWEEN MOHHO AND THE PROPOSED MOHHOAC

The results obtained by MOHHO and MOHHOAC are shown in Figure 2 for comparison from two dimensions, including classification error rate and selected feature number. It is noted that the number of selected features and classification error rate are treated as two dimensions, respectively corresponding to x axis and y axis. From Figure 2, it is seem that the proposed algorithm can obtain better performance than the original MOHHO on the most datasets. According to classification error rate, Table 3 presents average results obtained

TABLE 2. Results obtained by traditional approaches and single-objective approaches.

Datasets	LFS		GSBS		GA			PSO			HHO			WOA-SA			GWO		
	NOF	ErrRate	NOF	ErrRate	NOF	ErrorRate	Fit	NOF	ErrorRate	Fit	NOF	ErrorRate	Fit	NOF	ErrorRate	Fit	NOF	ErrorRate	Fit
Iris	3	0.04	3	0.04	3	0.12	0.246	2	0.1067	0.1853	3	0.0933	0.2247	1	0.1867	0.1993	1	0.16	0.178
page_blocks	7	0.02887	7	0.02887	7	0.0464	0.1644	5	0.046	0.1277	3	0.0585	0.1013	4	0.0555	0.1172	1	0.0822	0.0839
Wine	13	0.99438	13	0.99438	6	0.8571	0.7714	4	0.9	0.7771	5	0.9063	0.7964	4	0.9375	0.8071	3	0.931	0.7877
Zoo	11	0.06931	11	0.06931	6	0.0196	0.1216	6	0.0392	0.102	5	0.1373	0.1686	7	0.0196	0.098	6	0.0196	0.0862
Audit_risk	6	0.04897	6	0.04639	10	0.0077	0.0803	12	0.0052	0.093	11	0.0052	0.0856	10	0.0232	0.0926	1	0.0025	0.0094
Vehicle	12	0.32979	12	0.32506	7	0.7844	0.7053	9	0.8061	0.7449	7	0.8345	0.7454	7	0.8203	0.734	4	0.7778	0.6667
German	2	0.3	2	0.3	1	0.3	0.265	2	0.308	0.2964	1	0.286	0.2538	2	0.378	0.3524	1	0.314	0.2762
Ionosphere	15	0.10256	18	0.09117	14	0.0398	0.1118	17	0.1193	0.1926	17	0.0909	0.1699	13	0.1136	0.1652	6	0.0568	0.0797
Spect	4	0.26217	4	0.26217	7	0.2164	0.234	11	0.2388	0.2867	9	0.2985	0.3171	14	0.291	0.3546	4	0.2836	0.2616
Lungcancer	9	0.25	9	0.25	21	0.0625	0.1237	28	0.0625	0.1482	29	0.0625	0.1518	35	0.125	0.2228	8	0.0625	0.0781
Movement_libras	11	0.81944	12	0.81944	36	0.2111	0.248	36	0.1833	0.2258	39	0.1722	0.2235	42	0.1944	0.2479	13	0.1444	0.1441
Hillvalley_with_noise	4	0.5033	4	0.5033	46	0.3828	0.3974	45	0.4158	0.4218	5	0.4092	0.3372	45	0.4208	0.4257	8	0.3729	0.3142
Hillvalley_without_noise	2	0.49505	2	0.49505	46	0.3251	0.3512	38	0.3416	0.3485	5	0.3448	0.2858	47	0.3927	0.4073	9	0.3317	0.2832
Urban_land_cover	22	0.89185	38	0.89778	71	0.7887	0.7276	64	0.7468	0.6845	2	0.6489	0.5218	70	0.8478	0.7735	9	0.8033	0.6549
Musk1	2	0.0042	2	0.0042	77	0.0798	0.1561	71	0.0588	0.1321	19	0.105	0.1067	74	0.1555	0.213	28	0.0462	0.0705
Isolet5	13	0.88647	26	0.85375	311	0.1385	0.2114	278	0.1436	0.2048	75	0.1935	0.1791	279	0.2218	0.2677	79	0.1103	0.1138

TABLE 3. Average classification error rates obtained by MOHHO and MOHHOAC.

Dataset	Algorithms	
	MOHHOAC	MOHHO
Iris	0.103124	0.026667
Page-blocks	0.042476	0.038363
Wine	0.833333	0.05618
Zoo	0.007003	0.019608
Audit_risk	0	0.005155
Vehicle	0.7641	0.286052
German	0.309484	0.228
Ionosphere	0.04639	0.0625
Spect	0.227146	0.246269
Lungcancer	0	0.0625
Movement_libras	0.083333	0.377778
Hill_Valley_with_noise	0.363036	0.427393
Hill_Valley_without_noise	0.311832	0.429043
isolet5	0.139744	0.188462
Musk1	0.084034	0.10084
Urban_Land_Cover	0.647554	0.257396

by MOHHO and MOHHOAC. From Table 3, it is seen that MOHHOAC shows better classification performance than the original MOHHO. Therefore, the results in the Figure 2 and Table 3 demonstrate that the three strategies used in the MOHHOAC, Associative learning, grey wolf optimization, and chaotic local search, can improve the performance of the original MOHHO and enhance search ability of the proposed algorithm.

2) COMPARISONS BETWEEN THE PROPOSED ALGORITHM AND OTHER MULTI-OBJECTIVE APPROACHES

From Table 4, it is noted that the proposed algorithm MOHHOAC can obtain the best performance on the 9 datasets, accounting for 56.25%, with respect to HV indicator. The MOGWO can attain the best performance on the Page_blocks, Audit_risk, Hillvalley_with_noise and Lungcancer datasets. Winner is 4 out of 16. MOGWO ranks the second place in the all algorithms. However, MOGWO also doesnt́ obtain HV value on the Wine dataset. On the Audit_risk dataset, MOGWO can obtain the same performance with the proposed MOHHOAC. NSGA2 and NSPSOFS can achieve the best HV value on the only one dataset, Audit_risk, and they show the same performance with MOHHOAC. NSPSOFS and MOGWO on the Audit_risk dataset. MOHHO can obtain the best HV value on the 4 datasets. However, MOPSO cannot obtain the best HV value on the all datasets, which presents MOPSO cannot search the better Pareto front. On the wine dataset, MOHHOAC, MOGWO, NSGA2 and NSPSOFS doesn't search the Pareto front. In addition, according to the comparison from Table 4, MOHHOAC is significantly better than others. Individually compared with MOHHOAC, these numbers are (9, 4, 3) with respect to MOGWO, (13, 2, 1) with NSGA2, (14, 2, 0) with MOPSO, (12, 0, 4) with NSPSOFS, and (12, 4, 0) with MOHHO, where three numbers denote significantly better, worse and equivalent, respectively. In brief, from Table 4, MOHHOAC shows the best performance in the all algorithms on 9 out of 16 datasets. The main reasons maybe that in the proposed algorithm, associative learning and chaotic local search are introduced into MOHHO, to enhance the exploration and exploitation ability of MOHHO.

In Figure 3, the results obtained by these algorithms are shown for comparison from two dimensions, including classification error rate and selected feature number. Average results running 30 times by each algorithm are presented in the figure 3. From Figure 3, it is noted that number of features and classification error rate are treated as two dimensions, respectively corresponding to x axis and y axis. On the iris dataset, the Pareto front obtained by MOHHOAC cannot converge the Pareto optimum. MOPSO can obtain the classification error rate, while number of features obtained by it is the largest. On the Page_blocks dataset, MOHHOAC can obtain nearly the same Pareto Front, where NSGA2 can get the best error rate on the seven features. On the Wine dataset, MOHHOAC can obtain similar shape of Pareto front, but MOPSO can achieve the best performance. For the datasets, zoo, Audit_risk, Vehicle, Ionosphere, Lungcancer, Hill_ Valley_without_noise, Musk1, Urban_Land_Cover and Movement_libras, MOHHOAC can converge the Pareto optimum in terms of the Pareto Front obtained by it. On the dataset, German, MOHHOAC not only obtains the minimum classification error rate, but also achieves the minimum feature subset. For the remaining one dataset, MOHHOAC can obtain the better classification error rate when the number of features is less than 5, while MOPSO can

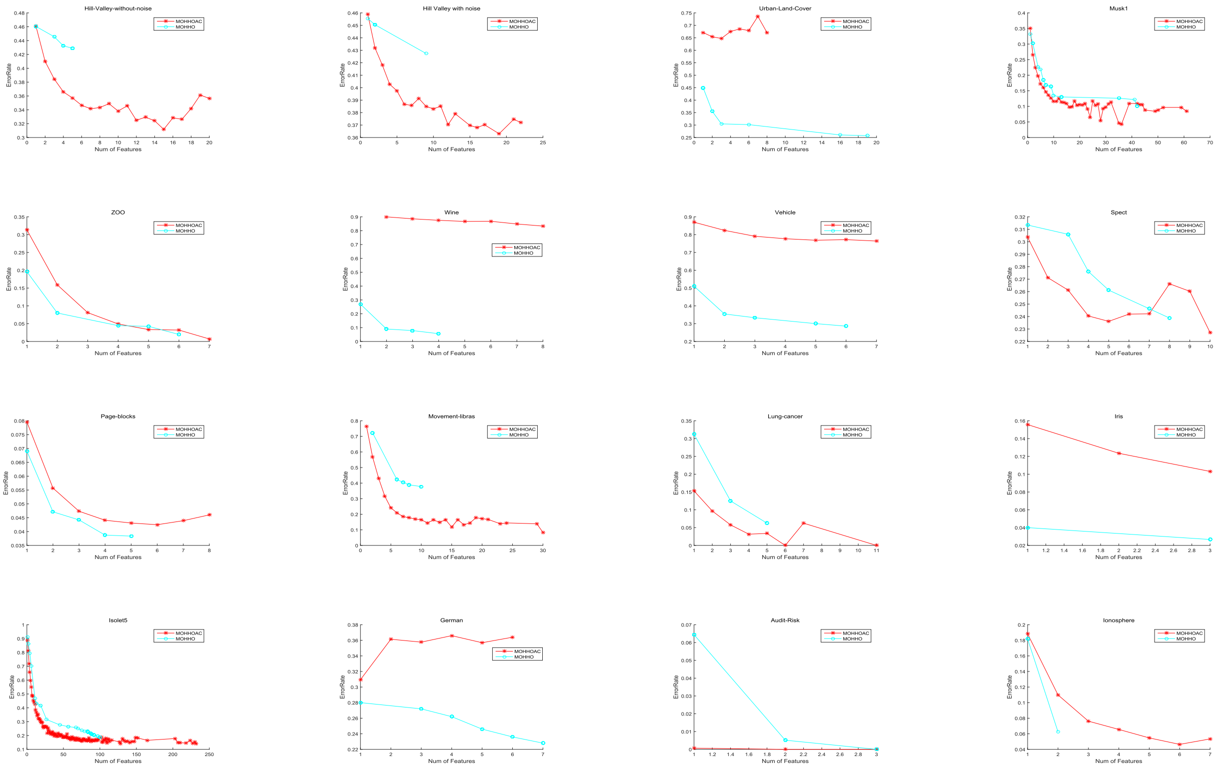

FIGURE 2. Pareto fronts obtained by MOHHO and MOHHOAC.

TABLE 4. HV values obtained by multi-objective evolutionary approaches.

Datasets	MOHHOAC	MOGWO	NSGA2	MOPSO	NSPSOFS	MOHHO
Iris	0.8717	0.8716∼	0.8649-	0.8696-	0.8696-	0.9607+
page_blocks	1.0684	1.0707+	1.0675-	1.0703+	1.0688∼	1.0671-
Wine	NaN	NaN∼	NaN∼	0.3952+	NaN∼	1.0544+
Zoo	1.1453	1.1452-	1.1229-	1.1245-	1.1452-	1.1046-
Audit_risk	1.1693	1.1693∼	1.1693∼	1.1667-	1.1693∼	1.1667-
Vehicle	0.3623	0.3687+	0.3566-	0.3579-	0.3483-	0.8314+
German	0.8093	0.8073-	0.7995-	0.8073-	0.7956-	0.8078-
Ionosphere	1.1533	1.1299-	1.1447-	1.0517-	1.1193-	1.1082-
Spect	0.9865	0.9556-	0.9597-	0.9695-	0.9247-	0.8966-
Lungcancer	1.1885	1.1907+	1.1896+	1.0843-	1.1885∼	1.1121-
Movement_libras	1.0845	1.0589-	1.0524-	0.8853-	1.0004-	0.7624-
Hillvalley_with_noise	0.8014	0.8223+	0.7941-	0.6756-	0.7731-	0.7313-
Hillvalley_without_noise	0.8594	0.8544-	0.8165-	0.6939-	0.82-	0.7306-
Urban_land_cover	0.6182	0.6006-	0.4752-	0.3283-	0.5289-	0.9152+
Musk1	1.1477	1.1375-	1.0439-	0.8638-	1.0801-	1.0806-
Isolet5	1.0499	1.0482-	0.8697-	0.6981-	0.9846-	0.9769-
+/-/∼		4/9/3	1/13/2	2/14/0	0/12/4	4/12/0

+, - and ∼ indicate that the result is significantly better,significantly worse and statistically similar to that obtained by MOHHOAC, respectively.

obtain the best classification error rate when the number of features is 11.

C. COMPARISONS BETWEEN MULTI-OBJECTIVE AND SINGLE-OBJECTIVE APPROACHES

In the section, single-objective evolutionary algorithms compare with multi-objective optimization algorithm on the feature selection problem, with respect to classification error rate. Classification error rate is the key goal for the feature selection problem. From Table 5, it is noted that MOHHOAC can obtain the minimum classification error rate on the seven datasets. The performance of MOHHOAC is superior to other algorithms including multi-objective evolutionary algorithms and single-objective evolutionary algorithms. It is seen that

TABLE 5. Classification error rates obtained by multi-objective approaches and single-objective approaches.

Datasets	Algorithms									
	MOHHOAC	MOGWO	MOPSO	NSGA2	NSPSOFS	GWO	WOA-SA	GA	PSO	HHO
audit_risk	0	0.001804	0.002577	0.001718	0.001546	0.0025	0.0232	0.0077	0.0052	0.0052
german	0.309484	0.320373	0.32291	0.320488	0.318669	0.314	0.378	0.3	0.308	0.286
Hill_Valley_with_noise	0.363036	0.341584	0.40099	0.359736	0.40099	0.3729	0.4208	0.3828	0.4158	0.4274
Hill_Valley_without_noise	0.311832	0.325083	0.362376	0.321881	0.343234	0.3317	0.3927	0.3251	0.3416	0.3944
Ionosphere	0.04639	0.039773	0.073864	0.046667	0.056818	0.0568	0.1136	0.0398	0.1193	0.0909
iris	0.103124	0.11251	0.085664	0.098889	0.103659	0.16	0.1867	0.12	0.1067	0.0933
isolet5	0.139744	0.148718	0.164103	0.112965	0.198077	0.1103	0.2218	0.1385	0.1436	0.2064
lung-cancer	0	0.027664	0.044643	0.0625	0.03125	0.0625	0.125	0.0625	0.0625	0.0625
movement_libras	0.083333	0.122222	0.149382	0.146296	0.155556	0.1444	0.1944	0.2111	0.1833	0.1722
Musk1	0.084034	0.071429	0.088235	0.065826	0.12605	0.0462	0.1555	0.0798	0.0588	0.1303
page-blocks	0.042476	0.042181	0.043661	0.040205	0.040697	0.0822	0.0555	0.0464	0.046	0.0585
spect	0.227146	0.22877	0.171642	0.18797	0.246269	0.2836	0.291	0.2164	0.2388	0.2985
Urban_Land_Cover	0.647554	0.672131	0.759494	0.711864	0.703125	0.8033	0.8478	0.7887	0.7468	0.8806
vehicle	0.7641	0.770686	0.789598	0.761229	0.775414	0.7778	0.8203	0.7844	0.8061	0.8345
wine	0.833333	0.860292	0.814134	0.831035	0.85	0.931	0.9375	0.8571	0.9	0.9063
zoo	0.007003	0.019608	0.019608	0.02	0.010221	0.0196	0.0196	0.0196	0.0392	0.1373

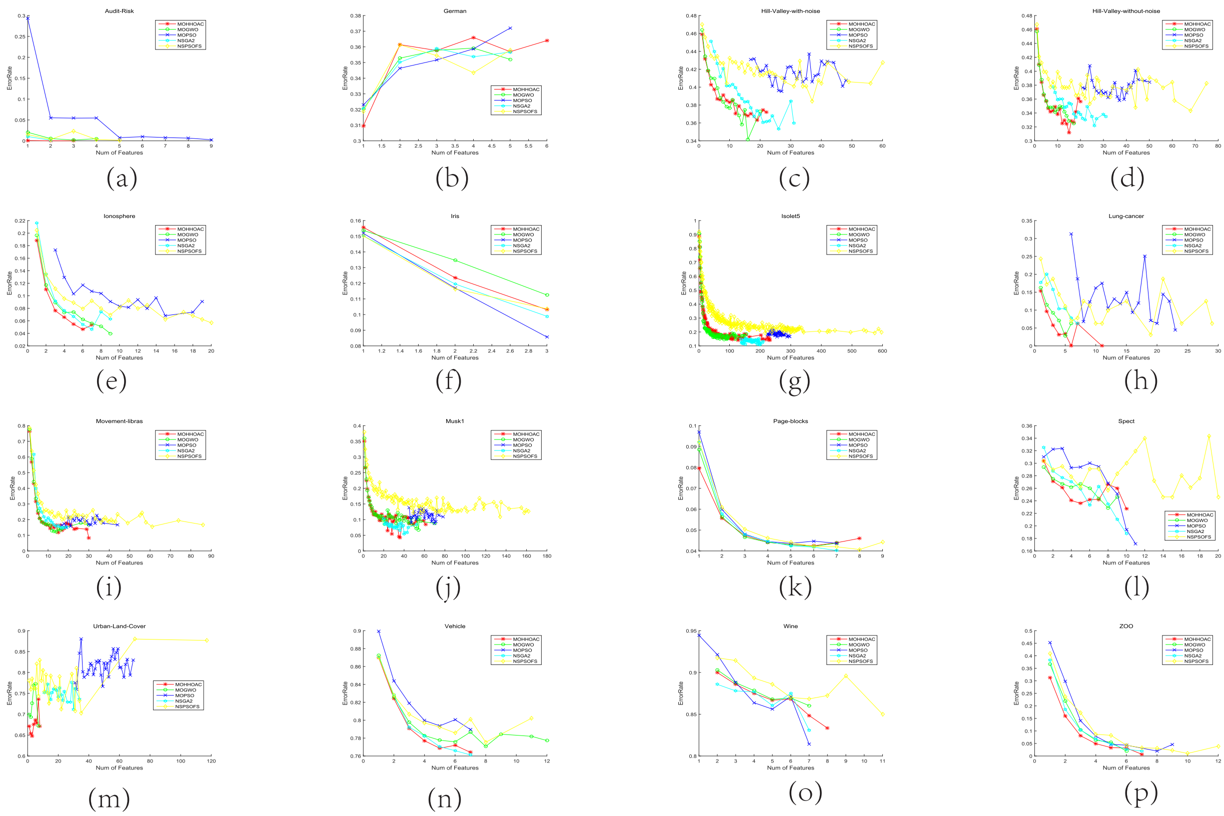

FIGURE 3. Pareto fronts obtained by MOHHOAC, MOGWO, MOPSO, NSGA2, NSPSOFS on the (a) Audit_risk, (b) German, (c) Hill_Valley_with_noise, (d) Hill_Valley_without_noise, (e) Ionosphere, (f) Iris, (g) Isolet5, (h) Lung-cancer, (i) Movement_libras, (j) Musk1, (k) Page-blocks, (l) Spect, (m) Urban_Land_Cover, (n) Vehicle, (o) Wine, (p) Zoo datasets.

single-objective evolutionary algorithm can obtain the best classification error rate on the three datasets, German, isolet5 and Musk1. For the remaining datasets, multi-objective evolutionary algorithms can show better performance. The reason maybe that in the multi-objective evolutionary algorithm, feature selection problem is considered as a bi-objective optimization problem, which can optimize two objectives simultaneously, classification error rate and feature subset size, and multi-objective evolutionary algorithm can obtain a non-dominated solution set which can cover various feature subset size in a run. However, in single-objective evolutionary algorithm, classification error rate and feature subset size both are considered as a fitness function by a coefficient. So results obtained by single-objective evolutionary algorithm vary with the coefficient. In addition, the Friedman test with Bonferroni-Dunn's procedure is implemented based on

TABLE 6. Average rankings of the algorithms (Friedman).

Algorithm	Ranking
MOHHOAC	2.6875
MOGWO	3.9062
MOPSO	5.4375
NSGA2	3.5
NSPSOFS	4.9688
GWO	5.5625
WOA-SA	9.0625
GA	5.4375
PSO	6.4688
HHO	7.9688

TABLE 7. Post Hoc comparison Table for $\alpha = 0.05$ (FRIEDMAN).

i	algorithm	$z = (R_0 - R_i)/SE$	p
9	WOA-SA	5.955517	0
8	HHO	4.933737	0.000001
7	PSO	3.532439	0.000412
6	GWO	2.685821	0.007235
5	MOPSO	2.569047	0.010198
4	GA	2.569047	0.010198
3	NSPSOFS	2.131141	0.033078
2	MOGWO	1.138555	0.254889
1	NSGA2	0.759036	0.447831

KEEL software, and the results are reported in Table 6 and 7. As shown in Table 6, MOHHOAC obtains the best place. In Table 7, p-values obtained by applying post hoc methods over the results of Friedman procedure. From Table 7, MOHHOAC can obtain significantly better performance than all algorithms, except for MOGWO and NSGA2. Therefore, MOHHOAC shows superior performance in the all algorithms.

VI. CONCLUSION

In the paper, a multi-objective Harris Hawks optimization with associative learning and chaotic local search algorithm, named MOHHOAC, is proposed to solve feature selection problem. In the proposed algorithm, to improve performance of Harris Hawks optimization, associative learning and chaotic local search are introduced into it. A comprehensive experiment is designed to demonstrate effectiveness and efficiency of the MOHHOAC. In the experiment, traditional algorithms (LFS and GSBS), single objective algorithms (GA, PSO, HHO, WOA-SA, GWO) and multi-objective algorithms (MOGWO, MOHHO, MOPSO, NSGA2, NSPSOFS) are compared on 16 benchmark datasets. The experiment results show that MOHHOAC outperform other compared algorithms in terms of classification error rate and feature subset size by fair comparison.

Despite the good performance, there are also some drawbacks with MOHHOAC, for instance it is computationally expensive, and their scalability to datasets with thousands of features is still unknown. In the future, the performance of multi-objective evolutionary algorithm on the imbalanced dataset and large-scale dataset with a large of features and samples should be investigated to enhance their scalability.

REFERENCES

[1] I. A. Gheyas and L. S. Smith, *Feature Subset Selection in Large Dimensionality Domains*. Amsterdam, The Netherlands: Elsevier, 2010.
[2] M. Dash and H. Liu, "Feature selection for classification," *Intell. Data Anal.*, vol. 1, no. 1, pp. 131–156, 1997.
[3] E. Hancer, B. Xue, M. Zhang, D. Karaboga, and B. Akay, "Pareto front feature selection based on artificial bee colony optimization," *Inf. Sci.*, vol. 422, pp. 462–479, Jan. 2018.
[4] A. Unler, A. Murat, and R. B. Chinnam, "mr2PSO: A maximum relevance minimum redundancy feature selection method based on swarm intelligence for support vector machine classification," *Inf. Sci.*, vol. 181, no. 20, pp. 4625–4641, 2011.
[5] A. W. Whitney, "A direct method of nonparametric measurement selection," *IEEE Trans. Comput.*, vol. C-20, no. 9, pp. 1100–1103, Sep. 1971.
[6] T. Marill and D. Green, "On the effectiveness of receptors in recognition systems," *IEEE Trans. Inf. Theory*, vol. IT-9, no. 1, pp. 11–17, Jan. 1963.
[7] B. Xue, M. Zhang, W. N. Browne, and X. Yao, "A survey on evolutionary computation approaches to feature selection," *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 606–626, Aug. 2016.
[8] M. L. Raymer, W. F. Punch, E. D. Goodman, L. A. Kuhn, and A. K. Jain, "Dimensionality reduction using genetic algorithms," *IEEE Trans. Evol. Comput.*, vol. 4, no. 2, pp. 164–171, Jul. 2000.
[9] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimization for feature selection in classification: A multi-objective approach," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1656–1671, Dec. 2013.
[10] X.-M. Hu, S.-R. Zhang, M. Li, and J. D. Deng, "Multimodal particle swarm optimization for feature selection," *Appl. Soft Comput.*, vol. 113, Dec. 2021, Art. no. 107887.
[11] F. Kılıç, Y. Kaya, and S. Yildirim, "A novel multi population based particle swarm optimization for feature selection," *Knowl.-Based Syst.*, vol. 219, May 2021, Art. no. 106894.
[12] 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.
[13] O. S. Qasim and N. M. Noori, "A new hybrid algorithm based on binary gray wolf optimization and firefly algorithm for features selection," *Asian-Eur. J. Math.*, vol. 14, no. 10, Oct. 2021, Art. no. 2150172.
[14] 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.
[15] E. Zorarpacı and S. A. Özel, "A hybrid approach of differential evolution and artificial bee colony for feature selection," *Expert Syst. Appl.*, vol. 62, pp. 91–103, Nov. 2016.
[16] 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.
[17] J. Wang, M. Ye, F. Xiong, and Y. Qian, "Cross-scene hyperspectral feature selection via hybrid whale optimization algorithm with simulated annealing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 2473–2483, 2021.
[18] I. Aljarah, M. Habib, H. Faris, N. Al-Madi, A. A. Heidari, M. Mafarja, M. A. Elaziz, and S. Mirjalili, "A dynamic locality multi-objective salp swarm algorithm for feature selection," *Comput. Ind. Eng.*, vol. 147, Sep. 2020, Art. no. 106628.
[19] M. Tubishat, S. Ja'afar, M. Alswaitti, S. Mirjalili, N. Idris, M. A. Ismail, and M. S. Omar, "Dynamic salp swarm algorithm for feature selection," *Expert Syst. Appl.*, vol. 164, Feb. 2021, Art. no. 113873.
[20] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019.
[21] H. M. Ridha, A. A. Heidari, M. Wang, and H. Chen, "Boosted mutation-based Harris hawks optimizer for parameters identification of single-diode solar cell models," *Energy Convers. Manage.*, vol. 209, Apr. 2020, Art. no. 112660.
[22] K. Balakrishnan, R. Dhanalakshmi, and U. M. Khaire, "A novel control factor and Brownian motion-based improved Harris Hawks Optimization for feature selection," *J. Ambient Intell. Hum. Comput.*, pp. 1–23, 2022.
[23] R. Sihwail, K. Omar, K. A. Z. Ariffin, and M. Tubishat, "Improved Harris hawks optimization using elite opposition-based learning and novel search mechanism for feature selection," *IEEE Access*, vol. 8, pp. 121127–121145, 2020.
[24] Z. M. Elgamal, N. B. M. Yasin, M. Tubishat, M. Alswaitti, and S. Mirjalili, "An improved Harris hawks optimization algorithm with simulated annealing for feature selection in the medical field," *IEEE Access*, vol. 8, pp. 186638–186652, 2020.

[25] J. C. Bednarz, "Cooperative hunting Harris' hawks (parabuteo unicinctus)," *Science*, vol. 239, no. 4847, pp. 1525–1527, Mar. 1988.
[26] S. Mirjalili, S. Saremi, S. M. Mirjalili, and L. D. S. Coelho, "Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization," *Expert Syst. Appl.*, vol. 47, pp. 106–119, Apr. 2016.
[27] S. Y. Wang, H. Liu, and H. Motoda, Eds., "Computational methods of feature selection," *Pattern Anal. Appl.*, vol. 13, pp. 247–249, 2010.
[28] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 4, pp. 491–502, Apr. 2005, doi: 10.1109/TKDE.2005.66.
[29] B. Tran, B. Xue, and M. Zhang, "Genetic programming for feature construction and selection in classification on high-dimensional data," *Memetic Comput.*, vol. 8, no. 1, pp. 3–15, Mar. 2016.
[30] P. Pudil, J. Novovičová, and J. Kittler, "Floating search methods in feature selection," *Pattern Recognit. Lett.*, vol. 15, no. 11, pp. 1119–1125, Nov. 1994.
[31] I.-S. Oh, J.-S. Lee, and B.-R. Moon, "Hybrid genetic algorithms for feature selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1424–1437, Nov. 2004.
[32] L.-F. Chen, C.-T. Su, and K.-H. Chen, "An improved particle swarm optimization for feature selection," *Intell. Data Anal.*, vol. 16, no. 2, pp. 167–182, Mar. 2012.
[33] Y. Zhang, X.-F. Song, and D.-W. Gong, "A return-cost-based binary firefly algorithm for feature selection," *Inf. Sci.*, vols. 418–419, pp. 561–574, Dec. 2017.
[34] Y. Xue, B. Xue, and M. Zhang, "Self-adaptive particle swarm optimization for large-scale feature selection in classification," *ACM Trans. Knowl. Discovery From Data*, vol. 13, no. 5, pp. 1–27, 2019.
[35] T. M. Hamdani, J. M. Won, A. M. Alimi, and F. Karray, "Multi-objective feature selection with NSGA II, adaptive and natural computing algorithms," in *Proc. 8th Int. Conf., (ICANNGA)*. Warsaw, Poland, Apr. 2007, pp. 240–247.
[36] K. Waqas, R. Baig, and S. Ali, "Feature subset selection using multi-objective genetic algorithms," in *Proc. IEEE 13th Int. Multitopic Conf.*, Dec. 2009, pp. 1–6.
[37] A. Dabba, A. Tari, and S. Meftali, "A new multi-objective binary Harris Hawks optimization for gene selection in microarray data," *J. Ambient Intell. Hum. Comput.*, pp. 1–20, 2021.
[38] T. Dokeroglu, A. Deniz, and H. E. Kiziloz, "A robust multiobjective Harris' hawks optimization algorithm for the binary classification problem," *Knowl.-Based Syst.*, vol. 227, Sep. 2021, Art. no. 107219.
[39] H. Chen, A. A. Heidari, H. Chen, M. Wang, Z. Pan, and A. H. Gandomi, "Multi-population differential evolution-assisted Harris hawks optimization: Framework and case studies," *Future Gener. Comput. Syst.*, vol. 111, pp. 175–198, Oct. 2020.
[40] W. Long, J. Jiao, M. Xu, M. Tang, T. Wu, and S. Cai, "Lens-imaging learning Harris hawks optimizer for global optimization and its application to feature selection," *Expert Syst. Appl.*, vol. 202, Sep. 2022, Art. no. 117255.
[41] J. Hu, Z. Han, A. A. Heidari, Y. Shou, H. Ye, L. Wang, X. Huang, H. Chen, Y. Chen, and P. Wu, "Detection of COVID-19 severity using blood gas analysis parameters and Harris hawks optimized extreme learning machine," *Comput. Biol. Med.*, vol. 142, Mar. 2022, Art. no. 105166.
[42] T. Suresh, Z. Brijet, and T. B. Sheeba, "CMVHHO-DKMLC: A chaotic multi verse Harris hawks optimization (CMV-HHO) algorithm based deep kernel optimized machine learning classifier for medical diagnosis," *Biomed. Signal Process. Control*, vol. 70, Sep. 2021, Art. no. 103034.
[43] C. Li, J. Li, H. Chen, M. Jin, and H. Ren, "Enhanced Harris hawks optimization with multi-strategy for global optimization tasks," *Expert Syst. Appl.*, vol. 185, Dec. 2021, Art. no. 115499.
[44] K. Hussain, N. Neggaz, W. Zhu, and E. H. Houssein, "An efficient hybrid sine-cosine Harris hawks optimization for low and high-dimensional feature selection," *Expert Syst. Appl.*, vol. 176, Aug. 2021, Art. no. 114778.
[45] T. Dokeroglu, A. Deniz, and H. E. Kiziloz, "A robust multiobjective Harris' hawks optimization algorithm for the binary classification problem," *Knowl.-Based Syst.*, vol. 227, Sep. 2021, Art. no. 107219.
[46] R. Bandyopadhyay, A. Basu, E. Cuevas, and R. Sarkar, "Harris hawks optimisation with simulated annealing as a deep feature selection method for screening of COVID-19 CT-scans," *Appl. Soft Comput.*, vol. 111, Nov. 2021, Art. no. 107698.
[47] U. Yüzgeç and M. Kusoglu, "Multi-objective Harris hawks optimizer for multiobjective optimization problems," *BSEU J. Eng. Res. Technol.*, vol. 1, pp. 31–41, Dec. 2020.
[48] A. A. Heidari, I. Aljarah, H. Faris, H. Chen, J. Luo, and S. Mirjalili, "An enhanced associative learning-based exploratory whale optimizer for global optimization," *Neural Comput. Appl.*, vol. 32, no. 9, pp. 5185–5211, May 2020.
[49] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 256–279, Jun. 2004.
[50] (2007). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml/index.php
[51] M. Gutlein, E. Frank, M. Hall, and A. Karwath, "Large-scale attribute selection using wrappers," in *Proc. IEEE Symp. Comput. Intell. Data Mining*, Mar. 2009, pp. 332–339.
[52] R. Caruana and D. Freitag, "Greedy attribute selection," in *Machine Learning Proceedings 1994*, W. W. Cohen and H. Hirsh, Eds. San Francisco, CA, USA: Morgan Kaufmann, 1994, pp. 28–36.
[53] B. Pfahringer, P. Reutemann, I. H. Witten, M. Hall, E. Frank, and G. Holmes, "The WEKA data mining software: An update," *ACM SIGKDD Explor. Newslett.*, vol. 11, pp. 10–18, Nov. 2009.

YOUHUA ZHANG received the B.S. degree in computer science from the University Geosciences of China, Wuhan, in 1996. He is currently an Associate Professor with Hebei GEO University. He is the author of one book and more than ten articles. His research interests include intelligence computation, feature selection, and machine learning.

YUHE ZHANG was born in Liaocheng, Shandong, China, in 1994. He received the B.S. and M.S. degrees in computer science from Hebei GEO University, Shijiazhuang, in 2018 and 2021, respectively. He is the author of more than two articles. His research interests include intelligence computation, evolutionary algorithm, and machine learning.

CUIJUN ZHANG received the B.S. degree in computer science from the Huazhong University of Science and Technology, Wuhan, Hubei, in 1994, and the M.S. degree in computer science from the Ordnance Engineering College. She is currently a Professor with Hebei GEO University. She is the author of more than 20 articles and two books. Her research interests include intelligence computation, evolutionary algorithm, and machine learning.

CHONG ZHOU received the M.S. degree in computer science from Hebei GEO University, in 2015, and the Ph.D. degree in information engineering of geosciences from the China University of Geosciences, Wuhan, in 2018. He is currently a Lecturer with Hebei GEO University. He is the author of more than 20 articles, including *Information Sciences* and *Knowledge-Based Systems*. His research interests include intelligence computation, evolutionary algorithm, multi/many-objective optimization, and machine learning.