A Novel Hybrid Whale Optimization Algorithm with Flower Pollination Algorithm for Feature Selection: Case Study Email Spam Detection

Hekmat Mohmmadzadeh[1], Farhad Soleimanian Gharehchopogh[2,*]

Department of Computer Engineering, Urmia Branch, Islamic Azad University, Urmia, Iran[1,2]
Corresponding Author [*] {bonab.farhad@gmail.com}

Abstract

Feature Selection (FS) in data mining is one of the most challenging and most important activities in pattern recognition. The problem of choosing a feature is to find the most important subset of the main attributes in a specific domain, and its main purpose is removing additional or unrelated features, and ultimately improving the accuracy of the classification algorithms. As a result, the problem of FS can be considered as an optimization problem, and use metaheuristic algorithms to solve it. In this paper, a new hybrid model combining whale optimization algorithm (WOA) and flower pollination algorithm (FPA) is presented for the problem of FS based on the concept of Opposition based Learning (OBL) which name is HWOAFPA. In our proposed method, using natural processes of WOA and FPA, we tried to solve the problem of optimization of FS; and on the other hand, we used an OBL method to ensure the convergence rate and accuracy of the proposed algorithm. In fact, in the proposed method, WOA create solutions in their search space using the prey siege and encircling process, bubble invasion and search for prey methods, and try to improve the solutions for the FS problem; along with this algorithm, FPA improves the solution of the FS problem with two global and local search processes in an opposite space with the solutions of the WOA. In fact, we used all of the possible solutions to the FS problem from both the solution search space and the opposite of solution search space. To evaluate the performance of the proposed algorithm, experiments were carried out in two steps. In the first stage, the experiments were performed on 10 FS datasets from the UCI data repository. In the second step, we tried to test the performance of the proposed algorithm in terms of spam e-mails detection. The results obtained from the first step showed that the proposed algorithm, performed on 10 UCI datasets, was more successful in terms of the average size of selection and classification accuracy than other basic metaheuristic algorithms. Also, the results from the second step showed that the proposed algorithm which was run on the spam e-mail dataset, performed much more accurately than other similar algorithms in terms of accuracy of detecting spam e-mails.

 Keywords: Feature Selection, Hybrid Optimization, Whale Optimization Algorithm, Flower Pollination Algorithm, Classification, Opposition Based Learning, Emil Spam Detection.

1. Introduction

As mentioned in [1], extracting useful information from collected data has its own complexity given the very diverse applications of information systems. Due to the existence of the huge amounts of data and the necessity to transform such data to useful information, data mining has been one of the fastest growing research topics and investigation in the information industry during the past decade. In the knowledge discovery process, the preprocessing step such as classification has a strong effect on the performance of the data mining methods on the required running time to analyze the complete dataset and also the quality of the extracted patterns. In this regard, FS is also one of the essential preprocessing steps, because it gets rid of the redundant, irrelevant variables within a dataset. As mentioned in [2, 3], FS techniques are categorized as filters and wrappers. Given the following factors, FS is a fundamental processing in real-

world problems: (a) the huge variety of noise, (b) false or counterfeit information, and (c) redundant and irrelevant features in the original feature set. Therefore, FS has become an important and active research topic in a variety of fields such as data mining, pattern recognition, text categorization, and image mining [4]. Speculatively, an FS technique should seek through the subsets of features, and discover the best one among all possible subsets in accord with a certain evaluation criterion. In order to specify the best subset, optimal FS process should investigate 2^n feature subsets if n features exist in the feature set. Nevertheless, calculating this value is actually very difficult and costly. In this case, we can use the sub-optimal feature subset instead of using the optimal one, without significantly reducing the classification outcome. As shown in [5], the heuristic and random search techniques can be used to find these sub-optimal subsets. A number of FS approaches can be categorized into three classifications hybrid, wrapper, and filter. Obviously, the hybrid approach takes advantage of the complementary strengths of the filter and wrapper approaches [6, 7]. The statistical analysis of the feature set is required in the filter approach without utilizing any learning model, while in the wrapper approach, a predetermined learning model is assumed, wherein features are selected that justify the learning performance of the particular learning model. As shown in Fig. 1, how to find salient features is illustrated schematically by wrapper and filter approaches.
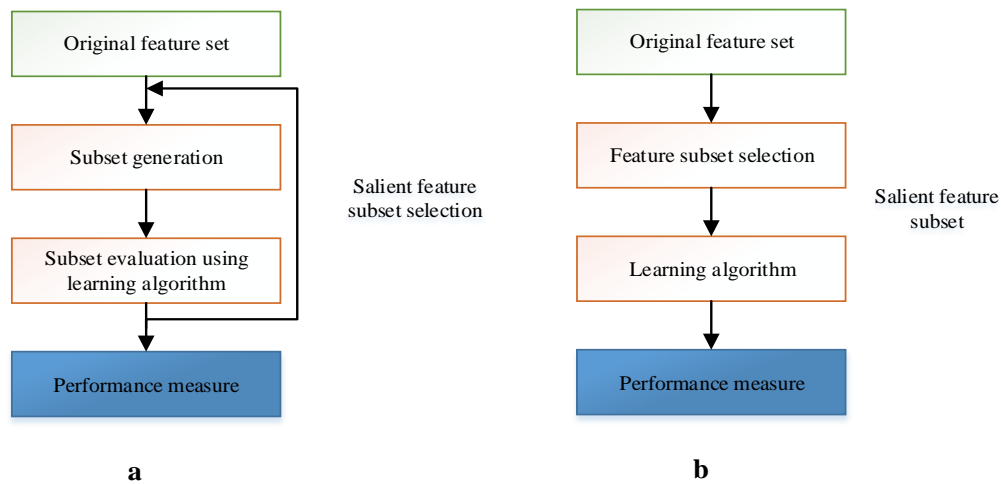
Fig. 1. *Schematic diagrams of (a) wrapper approach and (b) filter approach[7].*

As shown in [8], the classical optimization methods have some restrictions in solving the FS problems, thus, Evolutionary Computation algorithms are the alternative to resolve these restrictions and searching for the best solution [9]. It should be noted that the Evolutionary Computation algorithms are inspired by nature, social behavior, group dynamics, and biological interaction of species in a group. The binary versions of these algorithms enable us to study problems like FS and get to exceptional results. We have some benefits such as shorter training times, improved model interpretability, and improved generalization by reducing over fitting when constructing classification models utilizing the FS techniques. Since the FS can be examined as a search into a state space, a full search can be carried out in all the search spaces traversed. Nevertheless, if the number of features is too high, this approach is not achievable. Therefore, in order to evaluate, a heuristic search deliberates those features, which have not yet been selected at each iteration. As mentioned in [10], a random search produces random subsets within the search space which can be evaluated for the importance of classification performance. Optimization is a process of searching for the optimal solutions to a particular problem. It should be noted that for solving the optimization problems, nature-inspired metaheuristic algorithms are now among the most widely used algorithms and there are many natural inspired metaheuristic algorithms that are worked on the FS problem. In order to compose hybrid algorithms some of these algorithms are combined and in order to solve these problem others are applied alone. The meta-heuristics such as particle swarm optimization (PSO) [11], evolutionary algorithms

(EA) [12], ant lion optimizer (ALO) [13], Bat Algorithm (BAT) [14], Firefly Algorithm (FA) [15], WOA [16], Genetic Algorithm (GA) [17], FPA [18] and another metaheuristic algorithm are used for FS due to their randomized nature.

We organized the rest of this paper as follows: In Section 2 we present the related works. Section 3 presents the basics of the WOA algorithm and FPA. In Section 4 we describe the details of the proposed approach. Section 5 presents the experimental results and their analysis. Conclusions and future works are given in Section 6.

2. Related Works

During past decade there have been considerable researches in the field of optimization in which hybrid metaheuristic has been used by many researchers and many successful hybrid metaheuristic algorithms have been presented in FS domain [2, 7, 10, 19-21]. In [22], Oh et al. proposed the first hybrid technique for FS and in order to adjust the search process, they embedded local search operations that are sequential forward search, sequential forward floating search, and polynomial-time approximation into GA. It should be noted that the hybrid GA shows better convergence property with regards to standard GA in the experiments that are performed on various UCI Repository datasets including Wine, Glass, Wovel, Vehicle, Letter, WDBC, Segmentation, Sonar, Ionosphere, and Satellite. In another research, Khushaba et al. [23] combined Ant Colony Optimization (ACO) and Differential Evolution (DE) for FS. They applied DE to search for the optimal feature subset based on the solutions achieved by ACO. In [24], Olabiyisi et al. proposed a new hybrid algorithm which includes GA and SA metaheuristic to extract features for timetabling problem. In order to prevent from local optimum, they applied SA selection process instead of GA selection process. According to the experimental results of this study, SA performs better than GA, and the hybrid of GA-SA regarding runtime and optimality; nevertheless, runtimes of GA and SA are higher than that of the GA-SA technique. Compared to GA and SA, the GA-SA algorithm is more applicable due to the runtime performance. In order to form a single fitness function in a GA for FS to take the advantages of each measure, a wrapper measure (classification accuracy) and a filtered measure (Pearson correlation measure) were combined in [25].

In another survey, a hybrid filter and wrapper FS algorithm proposed by Akila [26] for classification problem using a combination of GA and local search method. In the hybrid technique, Local search is used at first by applying correlation-based filter techniques including transferring continuous functions into discrete counterparts, ranking, and redundancy removal with the symmetrical uncertainty measure for feature subsets and after that, the standard GA operators are used to these subsets. In this regard, the experimental analysis performed over the DNA Gene Analysis dataset which is achieved from the UCI Repository shows that the hybrid technique has the best performance. Babatunde et al. [27] presented a new hybrid algorithm for FS based on GA and ACO in which the selected feature subsets are evaluated by using the Support Vector Machine (SVM) classifier. This novel algorithm was compared with standard ACO and GA and tested on Face Detection dataset. The experimental results show that the new hybrid method exceeds in performance. In [28], Hasani et al. proposed a combination of Linear Genetic Programming and Bees Algorithm (BA) for FS on Intrusion Detection Systems (IDS) in which, LPG is applied to generate feature subset solutions. Then neighborhood search process of BAT is used to these solutions and eventually, SVM is applied to evaluate the feature subsets. The results show that the accuracy of classification increases by the hybrid LPGBA technique and it is more efficient than basic LPG and BA. Liao et al. [29] presented a new FS algorithm based on ACO, called Advanced Binary ACO (ABACO) which provides a suitable feature subset with good classification accuracy using a smaller feature set than competing FS techniques. In [30], Nekkaa and Boughaci presented a novel hybrid search technique by combining Harmony Search (HS) algorithm and Stochastic local search (SLS) for FS on classification task in which, a new probabilistic selection strategy is applied to use stochastic exploitation. This hybrid search technique is wrapped with SVM classifier. The results demonstrate that the hybrid HS-SLS technique is

better than HS and GA for FS. In [18], Sayed et al. investigated the effects of combining CSA with FPA to create a novel hybrid algorithm called Binary Clonal FPA (BCFA) to solve the FS problem. They used three public datasets from the UCI machine learning repository and in their experiments and defined the accuracy of OPF classifier as an objective function which should be maximized. The experimental results show that BCFA is able to obtain the best classification accuracy using the smallest number of selected features in less time. In [5], Khushaba et al. proposed a novel hybrid technique which combines artificial BCO method with DE algorithm for FS of classification tasks. The presented hybrid technique was evaluated by 15 common datasets from the UCI Repository. The results of the study demonstrate that the hybrid technique can select good features for classification tasks and improve runtime performance as well as the accuracy of the classifier. Mafarja et al. [2] presented two hybridization models to design different FS methods based on WOA. The performance of their methods was evaluated on eighteen standard benchmark datasets from the University of California, Irvine repository and compared with 3 common wrapper FS techniques in the literature. Compared to the other wrapper-based algorithms, the experimental results show that the efficiency of these two methods in improving the classification accuracy. In [21], Hasani et al. presented a new Swarm based hybrid algorithm called AC-ABC, which combines the characteristics of ACO and Artificial Bee Colony (ABC) algorithms to optimize FS. In order to evaluation of the presented algorithm, 13 UCI benchmark datasets have been used. The results demonstrate the promising behavior of the presented technique in increasing the classification accuracy and optimal selection of features. In order to improve classification accuracy, Liao et al. proposed 5 novel discrete Symbiotic Organisms Search (SOS) algorithms in another survey for simultaneous optimization of feature subset and neighborhood size of the KNN model. As described in [19], the performance of these five algorithms was evaluated by classification error and computational time based on stratified k-fold cross-validation with 11 datasets. In [1], Hasani et al. proposed evolutionary population dynamics and grasshopper optimization approach for FS problems. These approaches are employed to deal with various FS tasks and benchmarked on 22 UCI datasets. The experimental results show the superiority of these approaches when compared to other similar techniques in the literature. In order to solve FS problems, Khushaba et al. proposed a novel hybrid binary version of BAT and enhanced PSO algorithm [20]. This algorithm has compared with the original and other optimizers that have been applied for FS in the past. In this regard, a set of assessment indicators were applied to evaluate and compare the different optimizers over 20 standard datasets achieved from the UCI repository. The experimental results show the ability of the algorithm to search the feature space for optimal feature combinations. WOA and FPA are two types of metaheuristic algorithms that are combined in the proposed method to compose a hybrid algorithm for the FS based on OBL with a case study on the detection of spam emails. In the following, we will describe each of these algorithms.

3. Material and Methods

WOA and FPA are two examples of meta-heuristic algorithms, which in proposed approach we use them in combination to FS based on OBL and case study has been used to detect spam emails. Now, in the discussion below, we discuss each of these algorithms.

3.1. Whale Optimization Algorithm

In recent years there has been growing interest in WOA which was proposed in [31]. This search and optimization algorithm is a mathematical simulation of the behavior and movement of humpback whales in their search for food and provisions. As shown in Fig 2, WOA has inspired by the Bubble-net attacking strategy, where the whales start targeting fish by creating spiral-shaped bubbles around their fish down to 12 meters deep from the surface, and then, they swim back up to trap and capture their targeted fish. Based on the relative positions of whales, in this algorithm, the exploration process is represented by the random search of food which can be mathematically translated by updating the old solutions instead of choosing the

best ones through randomly selecting other solutions. In addition to this interesting behavior, WOA is notably distinguished from other optimization algorithms, because it only needs to adjust two parameters. These parameters make it possible to transition smoothly between both the exploitation and exploration processes.
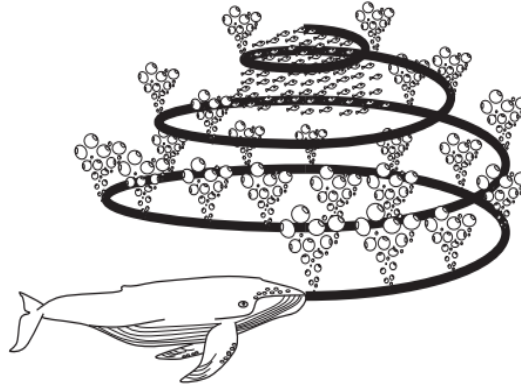
Fig.2. *Bubble-net feeding behavior of humpback whales [31].*

In the following section, we will describe the mathematical model of encircling prey, searching for prey, and spiral bubble-net foraging maneuver.

3.1.1. Encircling prey

By the increasing number of iterations from start to a maximum number of iterations, humpback whales encircle the prey and update their position in the direction of the best search agent. We can mathematically formulate this behavior as:

$$\vec{D}_i = \left| \vec{C}.\vec{X}^*(t) - \vec{X}(t) \right| \tag{1}$$

$$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A}.\vec{D} \tag{2}$$

Where '.' is an element-by-element multiplication, || indicates the absolute value, t indicates the current iteration, $\vec{A}$ and $\vec{C}$ are the coefficient vectors, $\vec{X}$ is the position vector, and X* is the position vector of the best solution achieved until this moment. The vectors $\vec{A}$ and $\vec{C}$ are computed as follows:

$$\vec{A} = 2\vec{a}.\vec{r} - \vec{a} \tag{3}$$

$$\vec{C} = 2.\vec{r} \tag{4}$$

3.1.2. Bubble-net attacking method

In order to mathematically model the bubble-net behavior of the humpback whales, two following approaches have designed:

1. Shrinking encircling mechanism: This behavior is obtained by reducing the value of 'a' from 2 to 0 in Eq. (3) over the course of iterations. We can define the new position of a search agent anywhere in between the position of the current best agent and the original position of the agent by selecting a random value $\vec{A}$ from the interval [-1, 1].

2. Spiral updating position: In this approach, the spiral equation between the position of the prey and the whale to imitate the helix-shaped movement of the humpback whales can be written as follows:

$$\vec{X}(t+1) = \vec{D}'. \, e^{bl}.\cos(2\pi l) + \vec{X}^*(t) \tag{5}$$

The pseudo code of the WOA algorithm is illustrated in Fig 3.

```
Initialize the whales population Xᵢ (i=1, 2,…,n)
Calculate the fitness of each search agent
X*= the best search agent
while (t < maximum number of iterations)
        for each search agent
        Update a, A, C, l and p
        if1 (p < 0.5)
        if2( │A│< 1)
        Update the position of the current search agent by the Eq. (1)
        else
        if2 ( │A│≥ 1)
        Select a random search agent (Xrand)
        Update the position of the current search agent by Eq. (8)
        end if2
        else if1 (p ≥ 0.5)
        Update the position of the current search by the Eq. (5)
        end if1
        end for
        Check if any search agent goes beyond the search space and amend it
        Calculate the fitness of each search agent
        Update X* if there is a better solution
         t=t+1
end while
return X*
```

Fig. 3. *Pseudo-code of the WOA.*

It should be noted that humpback whales swim along a spiral-shaped path and around the prey within a shrinking circle at the same time. Thus, for modeling this behavior, we choose one of the models (the shrinking encircling method or the spiral model) with the same probability of 50%. The mathematical model is described as follows:

$$\vec{X}(t + 1) = \{\vec{X}^*(t) - \vec{A}.\vec{D}\} \quad if \quad p \le 0.5 \tag{6}$$
$$\{\vec{D}'.\ e^{bl}.\cos(2\pi l) + \vec{X}^*(v)\} \quad if \quad p \ge 0.5$$

Where p represents a random number between [0, 1], l is a random number from interval [-1, 1], b is constant for defining the shape of the logarithmic spiral, and $D' = |\vec{X}^* - \vec{X}(t)|$ and shows the distance of the ith whale to the prey (the best solution achieved until this moment).

3.1.3. Search for prey (exploration phase)

In order to search for prey in the exploration phase, the variation of $\vec{A}$ vector can be used with the random values less than -1 or greater than 1 to force search agents to move away from a reference whale. The mathematical model for the exploration phase is as follows:

$$\vec{D} = |\vec{C}\vec{X}_{rand} - \vec{X}| \tag{7}$$
$$\vec{X}(t + 1) = \vec{X}_{rand} - \vec{A}.\vec{D} \tag{8}$$

where $\vec{X}_{rand}$ is a random position vector (i.e. a random whale) which is selected from the current population.

3.2. Flower Pollination Algorithm

Xin-She-Yang, a computer science specialist who invented various algorithms such as FA, BAT, and Cuckoo Algorithms, presented the FPA in 2012. Pollination is the transfer of pollen from a male part of a plant to a female part of a plant. Flowers must rely on vectors to move pollen. These vectors can include wind, water, birds, insects, bats, and other animals that visit flowers. We call animals or insects that transfer pollen from plant to plant "pollinators" [32]. Biotic and abiotic are two types of pollination. Biotic engages about 90% of FPA, while abiotic engages only about 10% and needs no pollinators. As described in [33], some insects incline to visit certain types of flowers and simultaneously, they bypass other species of flowers, and this phenomenon is called "flower constancy". It is worth noticing that all flowers which own the flower constancy property have the guarantee of reproduction maximization. This process is beneficial for both the pollinator and the flower. The flower maintains pollination whereas the pollinator gets sufficient amount of nectar and therefore increase the population of particular flower species. Pollination can also be defined as self and cross-pollination depend on the availability of pollinators. There is no reliable pollinator in self-pollination, while in cross-pollination, pollinators such as birds, bee, and bats fly over long distances and cause global pollination [34]. Applying the four following idealization rules, Yang [35] emulated the biological pollination process:

a) Global pollination represented in biotic and cross-pollination processes, as pollen-carrying pollinators fly following Lévy flights.
b) Local pollination represented in abiotic and self-pollination as the process does not require any pollinators.
c) Flower constancy which can be developed by insects can be considered as the reproduction probability that is proportional to the similarity of two flowers involved.
d) The interaction of local pollination and global pollination is controlled by a switching probability $p \in [0, 1]$. Local pollination can have a significant fraction p in the overall pollination activities due to the physical proximity and other factors, such as wind.

We should convert the aforementioned rules into updating equations in order to formulate updating formulas. For example, flower pollen gametes are transported by pollinators such as insects, and pollen can travel over a long range in the global pollination step because insects can often move and fly in a much longer distance. Thus, flower constancy and Rule 1 can be written as:

$$x_i^{(t+1)} = x_i^t + \gamma L(\lambda)(x_i^t - B) \tag{9}$$

Where B is the current best solution obtained among all solutions at the current generation/iteration and x_i^t is pollen *i* for solution vector *ix* at iteration *t*. Also, γ is a scaling factor in order to control the step size $L(\lambda)$ (the parameter that corresponds to the strength of the pollination). We can apply a Lévy flight to imitate this characteristic efficiently because insects may move over a long range with various distance steps. That means, we draw L > 0 from a Lévy distribution:

$$L \frac{\lambda.\Gamma(\lambda).Sin(\lambda)}{\pi} \frac{1}{s^{1+\lambda}} (S \geq\geq S_0 \geq 0) \tag{10}$$

In this equation, $\Gamma(\lambda)$ is the standard gamma function, and this distribution is valid for large steps *s* > 0. In order to model the local pollination, both Rule 2 and Rule 3 can be written as:

$$x_i^{(t+1)} = x_i^t + U(x_j^t - x_k^t) \tag{11}$$

where $x_k{}^t$ and $x_j{}^t$ are pollen from different flowers of the same plant species. This will imitate the flower constancy basically in a restricted neighborhood. From a mathematical aspect, if $x_j{}^t$ and $x_k{}^t$ selected from the same population or created from the same species, this correspondingly becomes a local random walk if U draws from a uniform distribution in [0, 1]. Although pollination of flowers can occur on any scale (local and global), adjacent flower patches or flowers close to each other are more likely to be pollinated by local flower pollen than those not close. To imitate this, we can effectively employ the proximity probability *p* or the switching probability like in Rule 4 to switch between common global pollination to intensive local pollination. We can use a trivial value of p = 0.5 to start, but as mentioned in [35], p = 0.8 may have a better result for most applications. Fig.4 presents the pseudo-code of FPA.

1: Objective min or max f(x), x = (x₁, . . . , xₙ);

2: Initialize a population of n flowers/pollen gametes with random solutions;

3: Find the best solution g∗ in the initial population;

4: Define a switch probability p ∈ [0, 1];

5: while (t < MaxGeneration) do

       6: for i = 1 : n (all n flowers in the population) do

       7: if rand < p, then

       8: Draw a (d-dimensional) step vector L which obeys a Levy flight distribution; ´ 9: Global pollination via Eq. (1);

       10: else

       11: Draw ε from a uniform distribution in [0,1]; 12: Randomly choose j and k among all the solutions;

       13: Do local pollination via Eq. (3);

       14: end if

       15: Evaluate new solutions;

       16: if new solutions are better, update them in the population;

       17: end for

19: end while

       18: Find the current best solution g ∗ ;

***Fig. 4**. Pseudo-Code FPA.*

In [36], another algorithm was proposed which is called Binary FPA (BFPA). In this algorithm, the search space is modeled as a d-dimensional Boolean lattice and the solutions are updated across the corners of a hypercube. As the FS problem is to select a specific feature or not, the solution can be represented as a binary vector. In this binary vector 1 indicates a feature will be selected to form the new dataset and 0 otherwise. It should be noted that the sigmoid function is applied to construct the mentioned binary vector as follows:

$$s\left(x_i^j(t)\right) = \frac{1}{1 + e^{x_i^j(t)}} \qquad (12)$$

Therefore, Eq. (9) and Eq. (11) are replaced by the following equation:

$$x_i^j(t) = \begin{cases} 1 & if \ s\left(x_i^j(t)\right) > \sigma \\ 0 & otherwise \end{cases} \tag{13}$$

where σ ~ U (0, 1) and $x_i^j(t)$ indicates the new pollen (solution) *i* with the *j*th feature vector, i = 1, 2, ... , m and j = 1, 2, ... , d, at the iteration *t*. Fig.5 shows the pseudo-code of BFPA.

Input: Training set Z_1 and evaluating set Z_2, α, number of flowers *m,* dimension *d* and iterations *T*

Output: Global best position $\hat{g}$.

Auxiliaries: Fitness vector *f* with size *m* and variables *acc, maxf it, global f it* ← −∞ and *maxindex.*

1. **for** *each flower i* (∀ *i* = 1 , . . . , *m*) **do for**
   each dimension j (∀ *j* = 1 , . . . , *d*) **do** *x j i*
   (0) ← Random { 0 , 1 } ; *f i* ← −∞ ;

2. **for** *each iteration t* (*t* = 1 , . . . , *T*) **do for**
   each flower i (∀ *i* = 1 , . . . , *m*) **do**

            Create *Z′* ₁ and *Z′* ₂ from *Z* ₁ and *Z*₂ , respectively, such that both contains only
            features such that $x_i^j = 0$, ∀ *j* = 1 , . . . , *d*;
            Train OPF over *Z′* ₁ , evaluate its over *Z′*₂ and stores the accuracy in *acc*;
            if (*acc* > *f i*) **then 10** *f i* ← *acc*;

                    for *each dimension j* (∀ *j* = 1 , . . . , *d*) **do**
             _ $x_i^j \leftarrow x_i^j$ (*t*   ) ;

       [*maxf it, maxindex*] ← *max* (*f*) ;
       if (*maxf it* > *gl obal f it*) **then**
            global f it ← *maxf it*; **for** *each*
            dimension j (∀ *j* = 1 , . . . , *d*) **do**
          $\widehat{g^j}$ ← *xjmaxindex* (*t*) ;

       for *each flower i* (∀ *i* = 1 , . . . , *m*) **do for** *each*
            dimension j (∀ *j* = 1 , . . . , *d*) **do**
            rand ← Random { 0 , 1 } ;

            $x_i^j(t) \leftarrow x_i^j(t-1) + α *$

            if *rand* < *p* **then**
                     Lévy (λ) ; **else**
            $x_i^j(t) \leftarrow x_i^j(t-1) + Ɛ(x_i^j(t-1) - x_i^k(t-1))$;
            if $(\sigma < \frac{1}{1+e^{x_i^j(t)}})$ **then**
            xij (*t*) ← 1 ; **else**
            $x_i^j(t) \leftarrow 0$;

Fig. 5. *Pseudo-code of Binary FPA* [36].

3.3. Opposition Based Learning

For the first time in the ancient Chinese philosophy [37], the primary opposition concept was expressed in the Yin-Yang symbol as shown in Fig. 6. This symbol shows the duality concept in which black and white

are Yin (receptive, feminine, dark, passive force) and Yang (creative, masculine, bright, active force), respectively. In other words, they are descriptions of complementary opposites rather than absolutes. Also, Fig. 7 shows the Greek classical elements of nature patterns. These elements describe the opposition concepts such as water (cold and wet) vs. fire (hot and dry), air (hot and wet) vs. earth (cold and dry). As described in [37], wet, dry, cold, and hot demonstrate nature entities and their opposite entities.
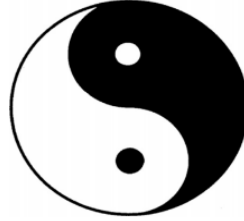
Fig. 6. Early opposite concept was mentioned in the Yin-Yang symbol [37]

One of the effective concept to enhance various optimization approaches is OBL[38]. The central idea behind this concept is the concurrent consideration of corresponding opposite estimate as a second set of candidate solutions in order to obtain a better approximation for the current candidate solution. It demonstrated that an opposition candidate solution will increase the chance of getting closer to the global optimum solution compare to a randomly chosen candidate solution. In a conceptual manner, the opposite numbers can be specified as:

Def 1: (OBL by Rahnamayan et al. [39] transforms existing search space to a new one): If $x \in [L, U]$ be a solution in existing search space, then, new solution x^* in the transformed space similar to OBL model is determined by:

$$x* = L + U - x \tag{14}$$
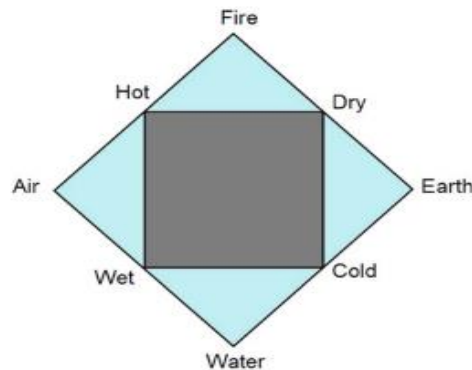
Fig. 7. The Greek classical elements to explain patterns in the nature[37]

4. Proposed Approach

In this section, we describe the proposed method, which includes a combination of WOA and FPA. In our proposed method for the FS problem, we used binary version of WOA [2] and FPA [36]. In the proposed method, we also used the concept of OBL to search for opposite search space and create better and new solutions to improve the hybrid algorithm. In this section, we first describe the target function of FS for the proposed algorithm and other metaheuristic algorithms in this paper. FS can be considered as a multi-objective optimization problem in which two opposite objectives exist: the minimum number of selected features and higher classification accuracy. Therefore, to define the target function of FS, we need a classification algorithm, and since most of the authors [1, 2, 10, 16, and 19] used the simplest classifier, i.e.

the KNN classifier [40], we also use this classifier in the proposed method to define the target function of the FS problem.

K-nearest neighbor (KNN) [40] is a widely-used simple method for classification problems. It is a supervised learning algorithm for classification of unknown sample instances based on the majority vote of its KNNs. Here, a wrapper approach to the FS is applied which exploits the KNN classifier as a guide for the same purpose. Classifiers do not rely on any model for KNNs and are determined solely based on the minimum distance from the current query instance to neighboring training samples.

Hence, in our proposed method, the KNN classifier is used for more accurate evaluation of the features selected by the proposed algorithm and other algorithms. Each solution is evaluated based on the proposed multi-objective function that depends on the KNN classifier. In the proposed multi-objective function, in order to balance the number of selected features in each solution (minimum) and the accuracy of the classification (maximum), the fitness function in equation (15) is used to evaluate a solution in any metaheuristic algorithm.

$$Fitness = \alpha \gamma_R(D) + \beta \frac{|R|}{|N|} \tag{15}$$

Where $\alpha \gamma_R (D)$ represents the classification error rate of a classifier; $|R|$ is multilinearity of the selected subset; $|N|$ is the total number of features in the dataset; the parameter α is the importance of the quality of classification and the parameter β is the length of the subset. The values of these two parameters are considered to be $\alpha \in [0, 1]$ and $\beta = (1 - \alpha)$ according to [13].
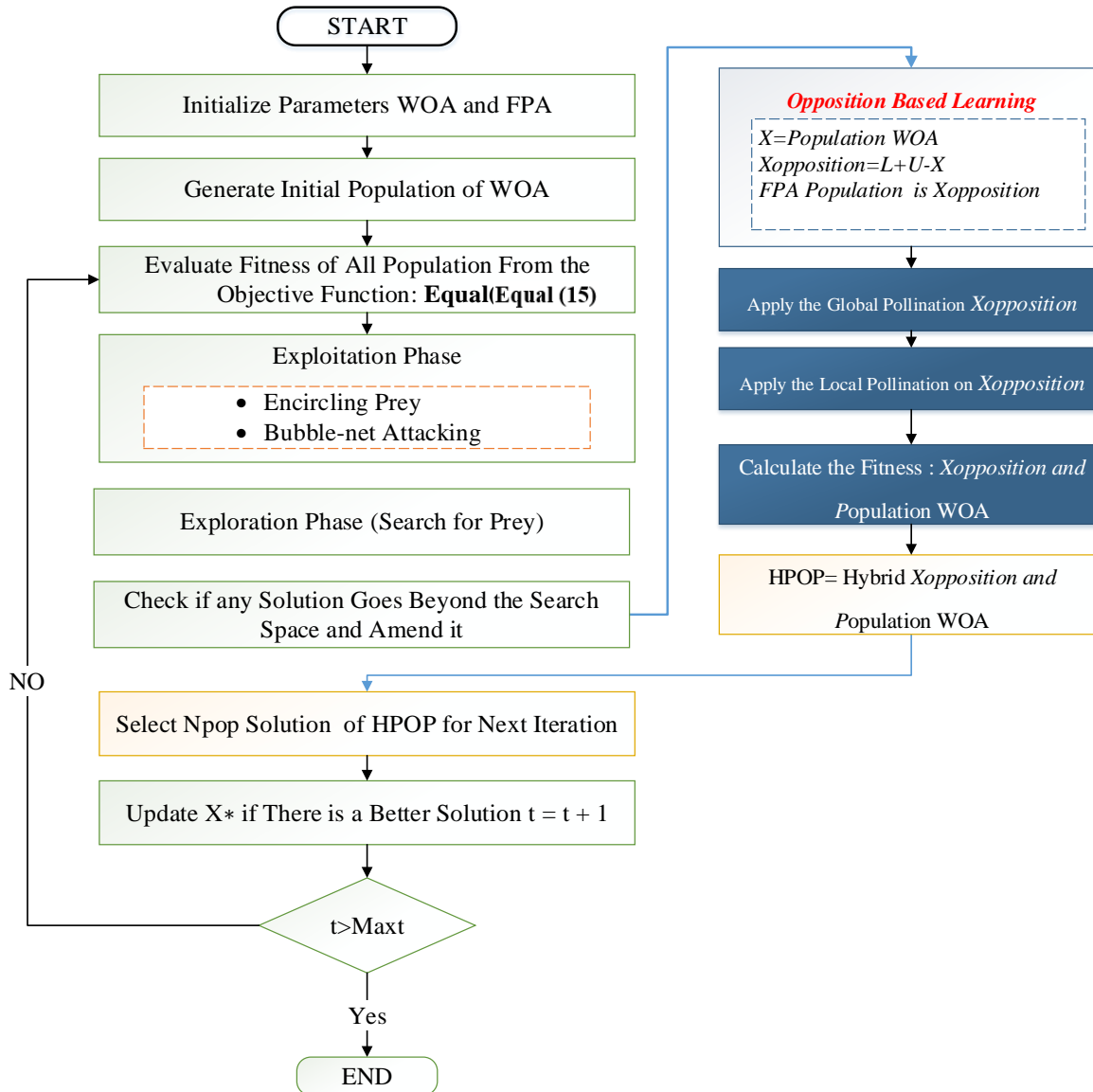
Fig. 8. *OBL Hybrid WOA with FPA (Proposed Approach).*

After defining the target function, we first proceeded with the hybrid method of WOA and FPA, and in the next step, we used the concept of OBL to improve the hybrid algorithm, to which we refer as the proposed method in this paper. The main goal of combining metaheuristic algorithms is to use the natural process of two different algorithms to solve a variety of difficult optimization problems. When combining metaheuristic algorithms, the performance and accuracy of the hybrid algorithm will be improved by maintaining the balance between exploration and productivity. The WOA is a recently introduced optimization algorithm that has shown great results in solving many optimization problems. However, the exploration in the standard WOA algorithm (Equation 8) depends on changing the position of each search agent based on a randomly chosen solution; we believe that the use of the exploration process of FPA will improve exploration in the standard WOA algorithm. Accordingly, we will use the utilization process of FPA to improve the WOA algorithm in the FS problem.

Considering that all metaheuristic algorithms, such as WOA and FPA, use random Initialization or uniform distribution to produce primary population, and sometimes production of inappropriate primary population cause metaheuristic algorithms to less converge towards the optimal target, and also many of the measured variables, such as the computation time, the use of storage memory and complexity, are related to the distance of this initialization from the global optimal solution, then, If we were able to simultaneously examine a solution and its opposite, in fact, we would have a great potential to accelerate convergence and improve the accuracy of metaheuristic algorithms using a population and its opposite. As a result, by adding the concept of OBL to our own hybrid algorithm, we guaranteed the convergence rate and the accuracy of the proposed algorithm. Figure 8 shows how the WOA and FPA are combined based on the OBL in the proposed method.

Therefore, it may be said that in our proposed method, we tried to solve the problem of optimization of FS using natural processes of WOA and FPA, and on the other hand, we used the OBL method to ensure the convergence rate and accuracy of the proposed algorithm. In the proposed method, whaling optimization algorithms, using the prey siege and encircling process, bubble invasion and search for prey methods, create solutions in their search space and try to improve them. Moreover, the FPA creating the opposite of the solutions of the WOA acts in accordance with the principle of OBL method. In fact, the FPA improves the solutions produced in the opposite space of the solutions of the WOA through two global and local search processes. Also, at the end of each generation of the proposed algorithm, the population of the WOA is combined with its opposite population, i.e. the population of the pollination algorithm; and only the solutions are chosen to be used in the next generation that have a more optimal value in terms of the target function. Thus, by combining the two optimization algorithms and using the opposite search space, we were able to present a powerful algorithm with a high convergence rate and high accuracy in this paper. The simulation results in Section (5) confirm the convergence rate and the accuracy of the proposed algorithm based on various experiments.

5. Experimental Results

The experiments were performed using a PC Intel (R) Core (TM) i5-2430M CPU 2.40 GHz with 6GB RAM, Windows 10 operating system. The proposed algorithm is implemented using Matlab. To evaluate the performance of the proposed algorithm, experiments are conducted in this section in two steps. In both steps, the binary version of the comparative algorithms was used. In the first stage, the experiments were performed on 10 FS datasets from the UCI data repository [41-48]. In subsection (5.1), we presented the first step experiments in details: the name and feature of the UCI dataset, the initial settings, and the results of the proposed algorithm application on the UCI dataset.

In the second step, we tested the performance of the proposed algorithm in terms of spam e-mails detection. In subsection 5.2, we presented the second step experiments in details: the name and feature of the spam e-mails dataset, the initial settings and the results of the proposed algorithm application on the spam e-mails dataset. The results obtained from the first step showed that the proposed algorithm, performed on 10 UCI datasets, was more successful in terms of the average number of FS and classification accuracy than other algorithms. Also, the results from the second step showed that the proposed algorithm which was run on the spam e-mail dataset, performed much more accurately than other similar algorithms in terms of accuracy of detecting spam e-mails.

5.1. UCI Datasets

Experiments are performed on 10 FS benchmark datasets from the UCI data repository [42] to evaluate the performance of the proposed approaches. This paper adopts the low-dimensional and large sample and high-dimensional and small sample datasets to analyze the performance of the proposed algorithm in terms of features and samples, respectively. A wrapper approach based on the KNN classifier (where K = 5 [13]) is used to generate the best reduction. In the proposed approach, each dataset is divided into cross

validation in the same way as in [43] for evaluation. In the K-fold cross validation, K-1 folds are used for training and validation, and the remaining folds are used for testing. This process is repeated M times. The training and validation sample sets are of the same size. The details of the used datasets, such as the number of attributes and instances in each dataset, which includes BreastEW, Wine, Glass, Contraceptive Method Choice (CMC), Stat log (Heart), Ionosphere, Lymphography, Spect, Blood, Zoo, are presented in Table 1.

Table 1: List of Datasets Used in the Experiments.

NO	Dataset	No. of attributes	No. of objects
1.	Boold	4	748
2.	BreastEW	30	569
3.	CMC	9	1473
4.	Glass	10	214
5.	Heart	13	270
6.	Ionosphere	34	350
7.	lymphography	18	148
8.	Spect	22	267
9.	Wine	13	178
10.	Zoo	16	101

5.1.1. Comparison of WOA, FPA and HWOAFPA

The performance of WOA, FPA, and HWOAFPA is discussed in terms of the average FS and classification accuracy criteria in this subsection. In the first experiment in this subsection, the parameters are set as follows. The maximum number of replicates is 25 and the population size is 20. The results of WOA, FPA and HWOAFPA are shown in Table 2.

Table 2: Classification accuracy and average selected attributes obtained from in test1 with 25 iteration

Dataset	Accuracy			Attributes		
	WOA	FPA	HWOAFPA	WOA	FPA	HWOAFPA
Boold	**0.7086**	0.6845	**0.7086**	2.00	2.40	**2.00**
BreastEW	0.9439	0.9263	**0.9474**	13.7	19.2	**13.5**
CMC	0.4898	0.4681	**0.4912**	4.95	5.80	5.00
Glass	**0.9907**	0.9907	0.9907	2.00	6.30	**1.85**
Heart	0.8148	0.6222	**0.8222**	2.95	8.30	4.10
Ionosphere	**0.9261**	0.8523	0.9205	**4.90**	22.1	12.7
lymphography	0.8784	0.6757	**0.9189**	11.0	11.1	**7.00**
Spect	**0.6791**	0.5896	0.6716	**2.00**	14.9	6.80
Wine	0.9326	0.5843	**0.9387**	8.00	7.85	**6.20**
Zoo	0.9216	0.8824	**0.9804**	6.00	10.4	9.00

In a review of Table 2, it is clear that the proposed algorithm outperforms the WOA and FPA in terms of achieving the two main goals of classification accuracy and the number of selected features. The proposed algorithm in this experiment has a better performance on 7 out of 10 UCI datasets in terms of classification accuracy than WOA and FPA, and also the proposed algorithm has a better performance on 6 out of 10 UCI datasets in terms of the selected features than WOA and FPA. Based on the results reported in this

experiment, according to Table 2, it can be concluded that the hybrid model using the OBL method has a significantly superior performance compared to WOA and FPA. Since the OBL method is embedded in the proposed method between the replications of the WOA and FPA, better results can be obtained by increasing the number of repetitions in the proposed algorithm. This motivates us to examine the proposed approach with a maximum number of repetitions of 50 and a population size of 10. The results of implementing the proposed algorithm with more repetitions are shown in Table 3

Table 3: Classification accuracy and average selected attributes obtained from in test2 with 50 iteration

Dataset	Accuracy			Attributes		
	WOA	FPA	HWOAFPA	WOA	FPA	HWOAFPA
Boold	0.6604	0.6016	**0.7086**	3.00	2.60	**2.00**
BreastEW	0.9579	0.9404	**0.9614**	18.0	20.4	**16.0**
CMC	0.5020	0.4695	**0.5115**	3.20	5.10	6.40
Glass	**0.9907**	0.9813	**0.9907**	4.00	6.40	**2.20**
Heart	0.7704	0.5037	**0.8593**	**2.00**	8.90	4.00
Ionosphere	0.8977	0.8693	**0.9148**	7.00	22.3	13.0
lymphography	0.8378	0.7432	**0.8649**	9.00	11.5	**9.00**
Spect	0.7015	0.4627	**0.7761**	10.1	13.7	**3.00**
Wine	0.9438	0.7079	**0.9551**	10.0	8.80	**3.00**
Zoo	0.8824	0.6667	**0.9608**	9.00	10.8	12.0

In a review of Table 3, it is obvious that with increasing the number of repetitions, the proposed algorithm outperforms the WOA and FPA in terms of achieving the two main goals of classification accuracy and the number of selected features. This experiment showed that the proposed algorithm has a better performance on all of 10 UCI datasets in terms of classification accuracy than WOA and FPA; moreover, the proposed algorithm has a better performance on 7 out of 10 UCI datasets in terms of the number of selected feature compared to the WOA and FPA. Based on the results reported in this experiment, according to Table 3, it can be concluded that the hybrid model, using the OBL method, has achieved a significantly higher convergence rate and accuracy compared to the performance of WOA and FPA. Since the opposite-based learning method has been embedded in the proposed method between the replications of WOA and FPA, it can be seen that by increasing the number of repetitions, better results are obtained by the proposed algorithm.

According to the results of these two experiments, it can be concluded that the proposed algorithm successfully solves the problem of FS optimization; and also according to the later experiment, considering the algorithm's success in all datasets, the convergence rate and accuracy of the proposed algorithm is guaranteed by using the OBL method. In the next subsection, we will compare the proposed algorithm with other basic metaheuristic algorithms for further evaluation.

5.1.2. Comparison with the State-of-the-Art Approaches

In subsection 5.1.1, analyzing the results of the proposed algorithm implementation on 10 valid UCI datasets, we found that the proposed method using the OBL method outperformed the WOA and FPA in all datasets in terms of classification accuracy, and it is competitive to WOA and FPA in terms of the number of selected features. Here, we will compare the performance of the proposed method with other powerful metaheuristic methods such as GA, PSO, and BATs in addition to the WOA and FPA to see how well the proposed method performs in comparison with other powerful methods. In this test, the parameters in all the comparison algorithms are set as follows: maximum number of replicates= 80 and population size=10.

Table 4 shows the results of the comparison of the proposed method with WOA, GA, PSO, BAT, and FPA in terms of classification accuracy.

Table 4: Comparison between the proposed approaches and the state-of-the-art approaches in terms of classification accuracy.

Dataset	BAT	GA	PSO	WOA	FPA	HWOAFPA
Boold	0.6658	0.7406	0.7406	0.7406	0.6791	**0.7406**
BreastEW	0.9439	0.9719	0.9684	0.9614	0.9509	0.9649
CMC	0.3948	0.4858	0.4980	0.4980	0.4668	**0.5061**
Glass	0.9720	0.9813	0.9813	0.9813	0.9720	**0.9813**
Heart	0.6667	0.8074	0.7852	0.7852	0.5704	0.7852
Ionosphere	0.8466	0.9261	0.9261	0.9091	0.8523	**0.9318**
lymphography	0.5946	0.8378	0.8378	0.8108	0.6351	**0.8378**
Spect	0.6045	0.7761	0.6866	0.7687	0.5970	0.7463
Wine	0.8090	0.9438	0.9551	0.9663	0.6742	**0.9488**
Zoo	0.8627	0.9117	0.9412	0.9216	0.8235	**0.9412**

As shown in Table 4, the proposed algorithm has a high performance in terms of classification accuracy. In addition, the proposed algorithm performs better than all other methods on the entire dataset, except for the three datasets, in which GA and other methods are slightly better than the proposed method. Table 5 shows the results of comparing the proposed method with WOA, GA, PSO, BAT, and FPA in terms of the average of selected features.

Table 5: Comparison between the proposed approaches and the state-of-the-art approaches in terms of average number of selected attributes.

Dataset	BAT	GA	PSO	WOA	FPA	HWOAFPA
Boold	2.30	1.00	1.71	1.00	1.90	**1.00**
BreastEW	11.7	12.5	16.6	18.0	20.0	12.1
CMC	4.00	3.00	3.61	4.00	6.00	**4.00**
Glass	3.30	2.00	5.69	3.00	6.10	**2.00**
Heart	5.00	5.00	6.27	5.00	8.30	**4.30**
Ionosphere	13.6	15.0	15.6	8.00	21.9	**4.90**
lymphography	5.40	8.90	8.46	3.00	12.1	11.00
Spect	8.10	10.0	12.0	4.00	14.6	**4.00**
Wine	4.00	5.00	6.90	7.00	8.20	5.00
Zoo	4.80	6.00	8.49	5.00	9.90	6.00

As shown in Table 5, the proposed algorithm has a good performance in terms of the average of selected features compared to other advanced metaheuristic algorithms. In addition, the proposed algorithm performs better than all other methods in most datasets, except for few datasets in which other methods are slightly better than the proposed method.

In this subsection, the convergence rate of each of the WOA, FPA, GA, PSO, BATs, and finally that of our proposed method on 10 valid UCI datasets was compared according to the convergence rate of the target function defined in Equation 15 to see that how well the proposed method performs in comparison with other methods in terms of the convergence rate of the target function. In this test, the parameters of all the comparison algorithms are set as follow: maximum number of repetitions = 20 and population size = 10.

The results of this experiment are shown in Figures 9 to 18, each of which represents the convergence rate of algorithms on 10 datasets, respectively. In Figure 9, the results of the comparison of the proposed method with WOA, GA, PSO, BAT, and FPA are shown in terms of the convergence rate of the target function on the Boold dataset.
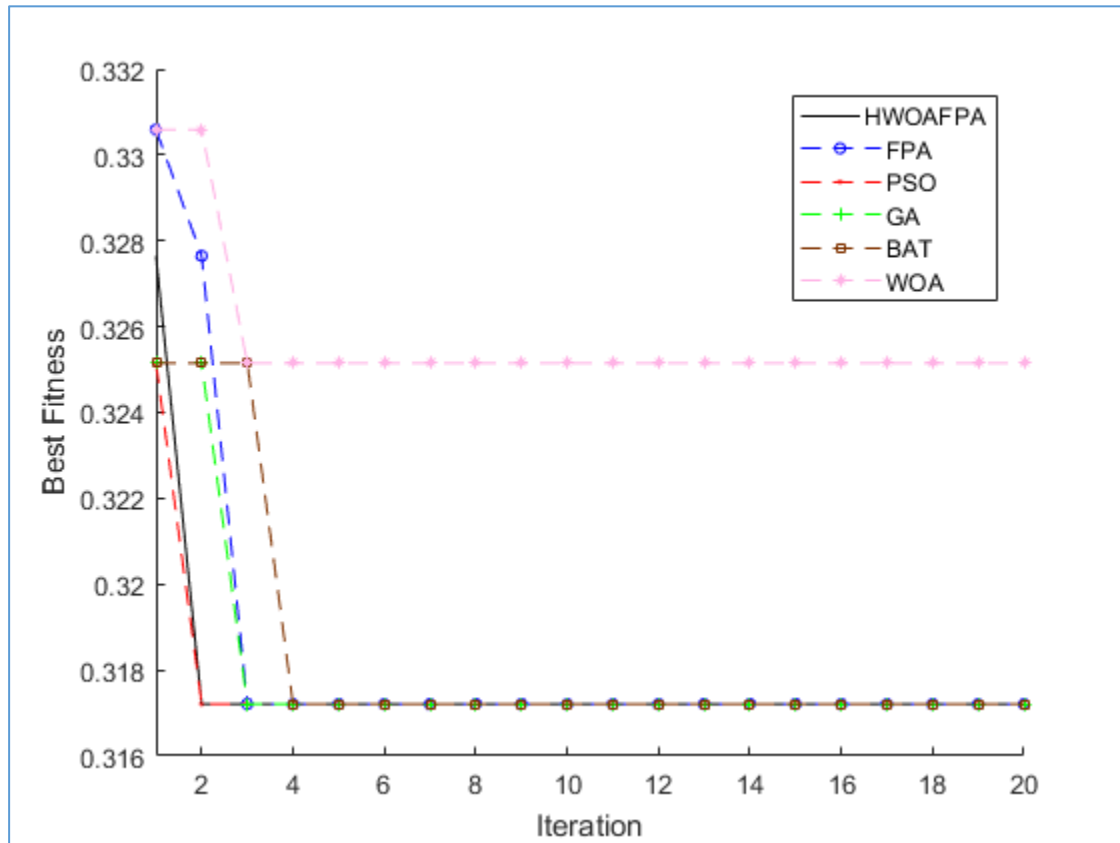
Fig. 9. *Comparison between the proposed approaches and the state-of-the-art approaches in terms of convergence-Dataset Boold*

As shown in Figure 9, the proposed algorithm and other comparative algorithms have almost the same results. The reason for the proximity of the results in this figure is that the Boold dataset has 4 features, and each algorithm will obtain the optimal solution with several runs. Accordingly, the results of the implementation of the proposed method, WOA, GA, PSO, BAT, and FPA on the Boold dataset showed that all the compared algorithms give similar solutions due to the small size of the dataset. Figure 10 shows the results of comparing the proposed method with WOA, GA, PSO, BAT, and FPA in terms of the convergence rate of the target function on the BreastEW dataset.
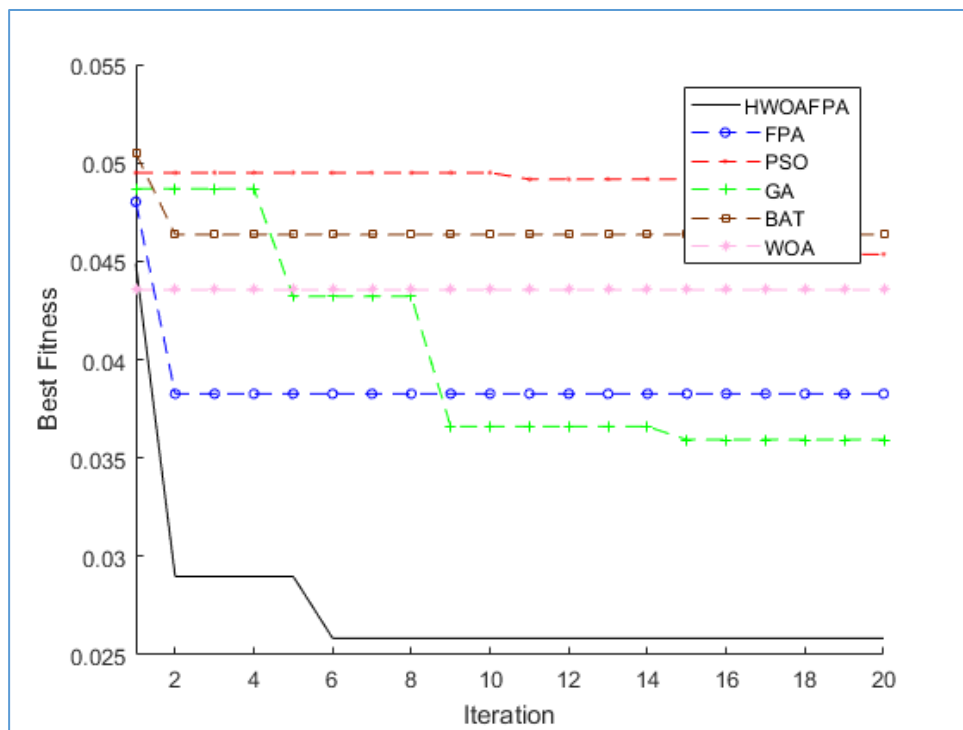
Fig. 10. *Comparison between the proposed approaches and the state-of-the-art approaches in terms of convergence-Dataset BreastEW*

As shown Figure 10, the proposed algorithm compared to other algorithms has a high performance in terms of the convergence of the target function, because the BreastEW dataset has more features. In each run, the proposed algorithm, using the opposite space of the solutions, has been able to outperform other algorithms. Accordingly, the results of the implementation of the proposed method, WOA, GA, PSO, BAT and FPA on the BreastEW dataset showed that all the other algorithms have lower efficiency due to the large size of the dataset; however, the proposed method obtained much better results due to the exploitation of the opposite space of the solutions in this article. Figure 11 shows the results of comparing the proposed method with WOA, GA, PSO, BAT, and FPA in terms of the convergence rate of the target function on the CMC dataset.
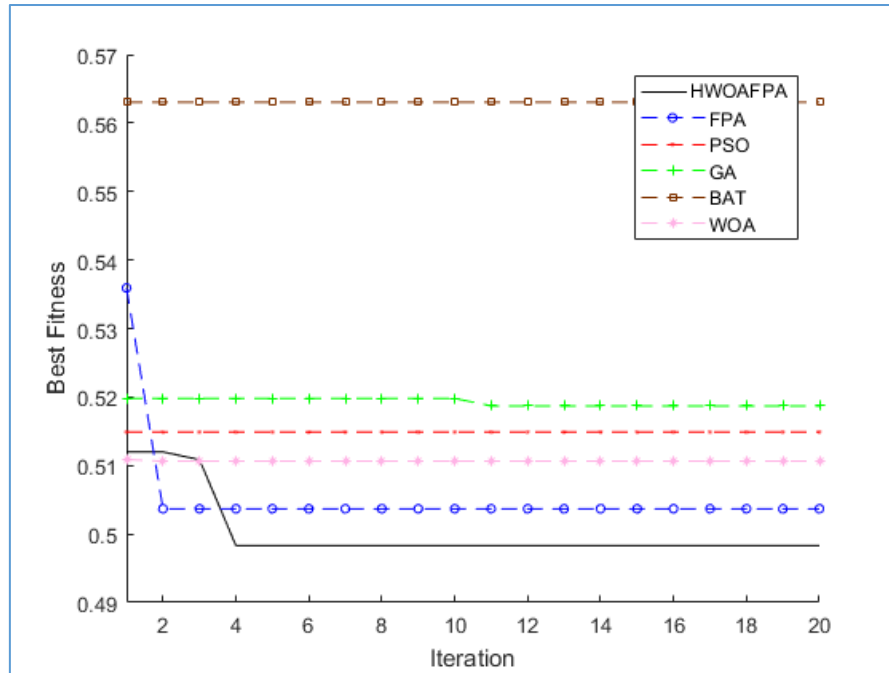
Fig. 11. *Comparison between the proposed approaches and the state-of-the-art approaches in terms of convergence-Dataset CMC*

As the results of (11) is shown, the proposed algorithm has better performance than other comparison algorithms have been on the convergence objective. In this experiment, the proposed algorithm had better performance compared to the GA, PSO, and BAT algorithms at the beginning; and with increasing the repetitions, it generated better results using the solutions and their opposite space at the final repetitions compared to all other algorithms. Figure 12 shows the results of comparing the proposed method with WOA, GA, PSO, BAT, and FPA in terms of the convergence rate of the target function on the Glass dataset.
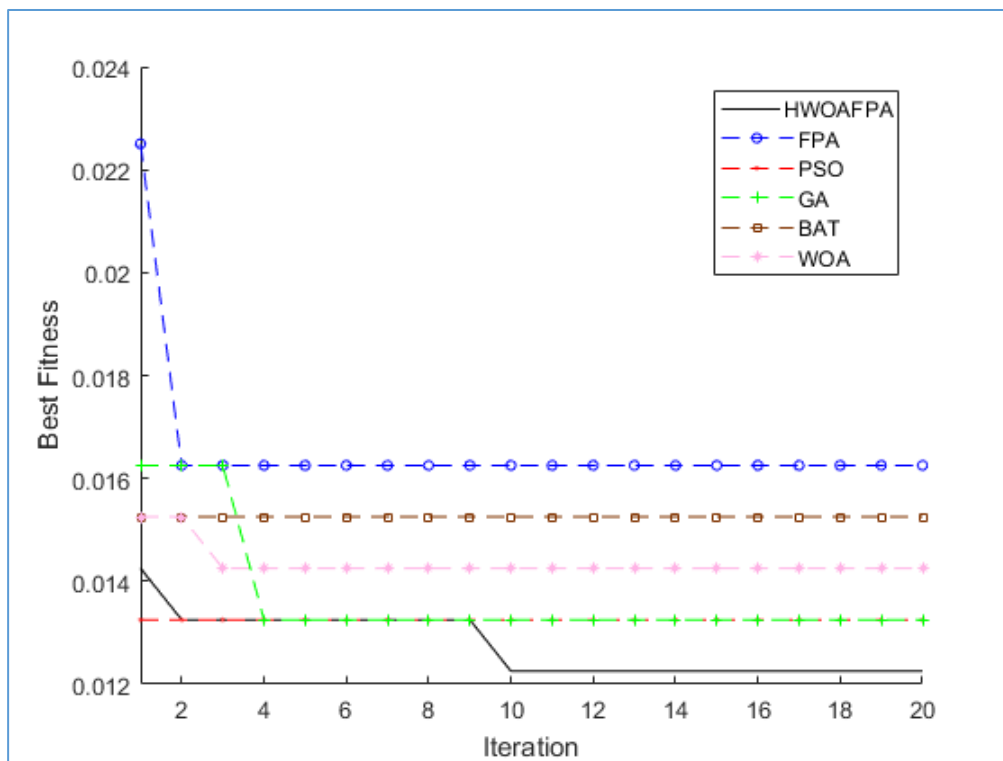
Fig. 12. *Comparison between the proposed approaches and the state-of-the-art approaches in terms of convergence-Dataset Glass*

As shown in the results of Fig. 12, the proposed algorithm still had a relatively better performance in the convergence of the objective function than other comparative algorithms. In this experiment, the proposed algorithm had an acceptable performance compared to the WOA and FPA at the beginning; and with increasing the repetitions, it generated better results using the solutions and their opposite space at the final repetitions compared to all other algorithms. Figure 13 shows the results of comparing the proposed method with WOA, GA, PSO, BAT, and FPA in terms of the convergence rate of the target function on the Heart dataset.
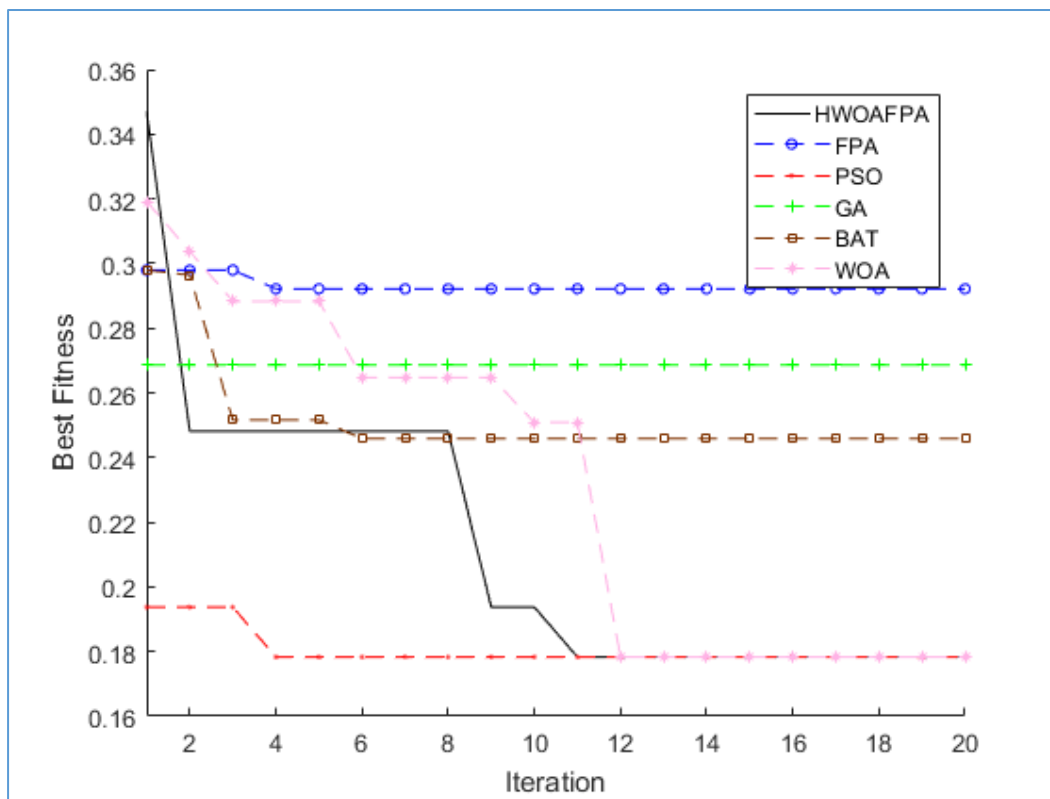
Fig. 13. *Comparison between the proposed approaches and the state-of-the-art approaches in terms of convergence-Dataset Heart*

As shown in Figure 13, the proposed algorithm in comparison with other algorithms has similar performance to that of the WOA and PSO algorithms. Since the Heart dataset is a modest dataset in terms of the features, some algorithms such as WOA and PSO algorithms had a high performance in terms of the convergence rate of the target function. Figure 14 shows the results of comparing the proposed method with WOA, GA, PSO, BAT, and FPA in terms of the convergence rate of the target function on the Ionosphere dataset.
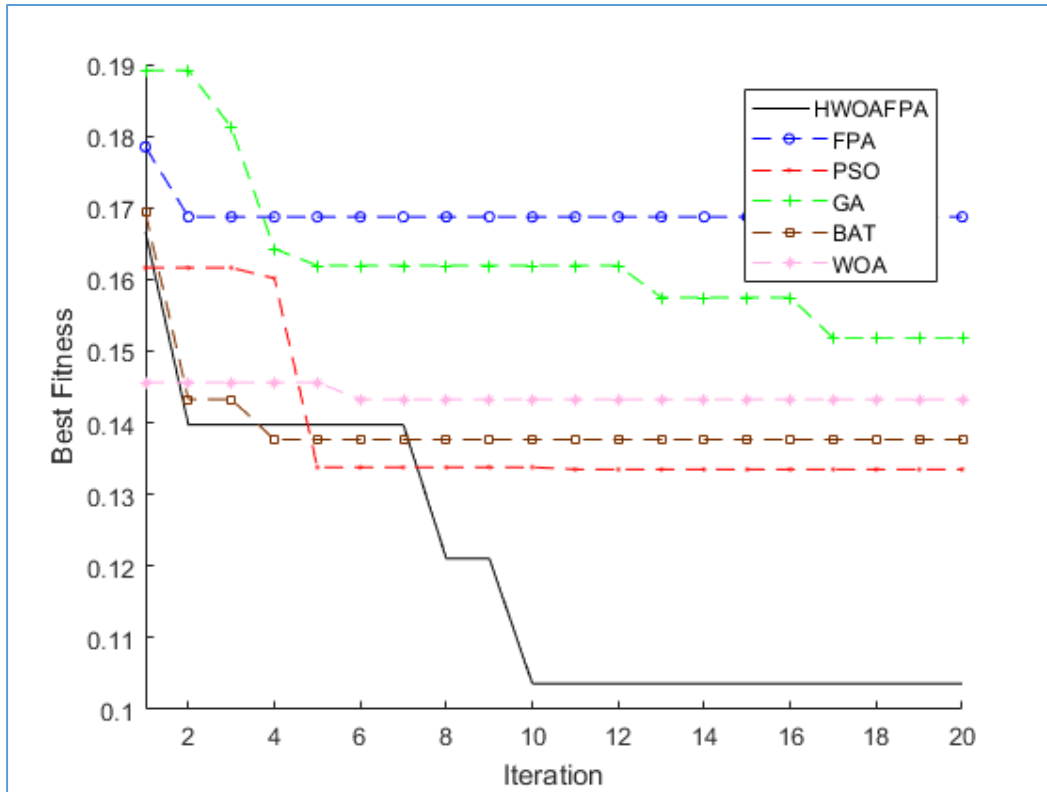
Fig. 14. *Comparison between the proposed approaches and the state-of-the-art approaches in terms of convergence-Dataset Ionosphere*

As the results of (14) is shown, the proposed algorithm has better performance than other comparison algorithms have been on the convergence objective. That is, the high performance of the proposed algorithm is because the Ionosphere dataset has high features. In each run, the proposed algorithm, using the opposite space of the solutions, has been able to outperform other algorithms. Accordingly, the results of the implementation of the proposed method, WOA, GA, PSO, BAT and FPA on the Ionosphere dataset showed that all the other algorithms have lower efficiency due to the large size of the dataset; however, the proposed method obtained much better results due to the exploitation of the opposite space of the solutions in this article. Figure 15 shows the results of comparing the proposed method with WOA, GA, PSO, BAT, and FPA in terms of the convergence rate of the target function on the Lymphography dataset.
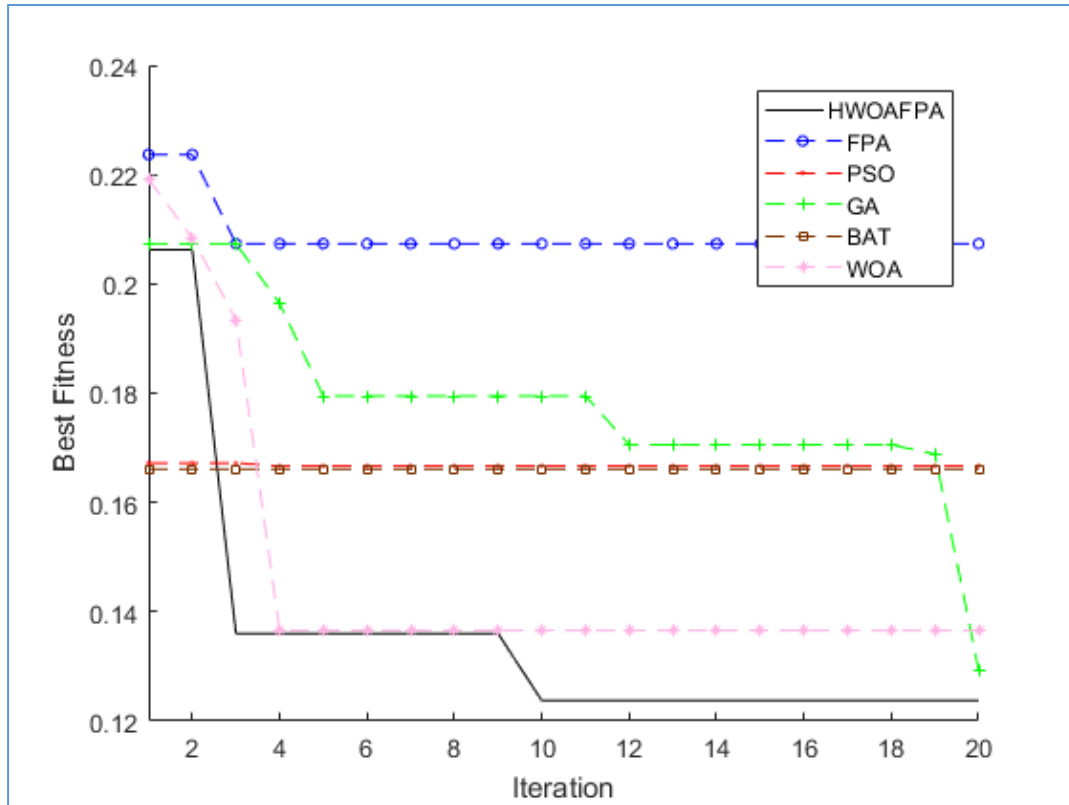
Fig. 15*. Comparison between the proposed approaches and the state-of-the-art approaches in terms of convergence-Dataset Lymphography.*

As the results of (15) is shown, the proposed algorithm has better performance than other comparison algorithms have been on the convergence objective. That is, the high performance of the proposed algorithm is due to the fact that the Lymphography dataset has medium to high features; and consequently, some algorithms, such as the FPA and BAT algorithms, lost their efficiency as a result of feature increase, and, even with the increase of repetitions, these algorithms could not increase the convergence rate of the target function. Finally, the proposed algorithm has been able to outperform other algorithms using the opposite space of the solutions. Figure 16 shows the results of comparing the proposed method with WOA, GA, PSO, BAT and FPA in terms of the convergence rate of the target function on the Spect dataset.
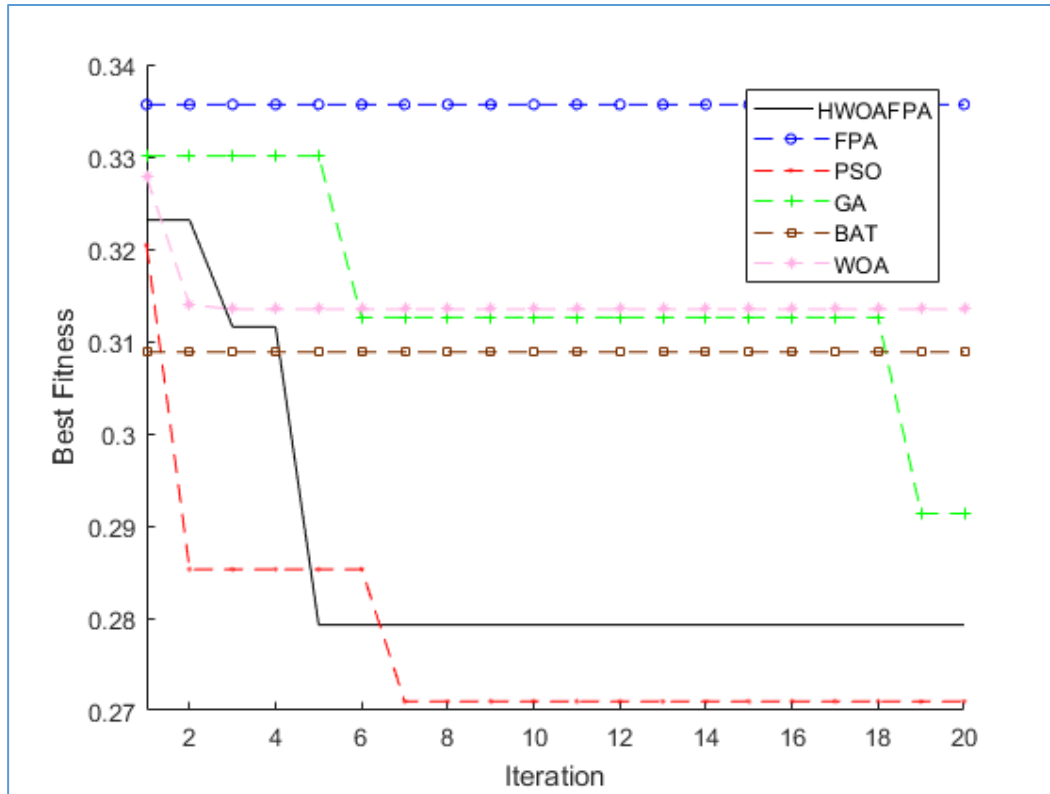
Fig. 16. *Comparison between the proposed approaches and the state-of-the-art approaches in terms of convergence-Dataset Spect.*

As shown in Figure 16, the proposed algorithm in comparison with other algorithms has similar performance to that of the PSO algorithm. Since the Spect dataset is a relatively modest dataset in terms of the features, some algorithms such as WOA and the proposed algorithms had a high performance in terms of the convergence rate of the target function. Figure 17 shows the results of comparing the proposed method with WOA, GA, PSO, BAT, and FPA in terms of the convergence rate of the target function on the Wine dataset.
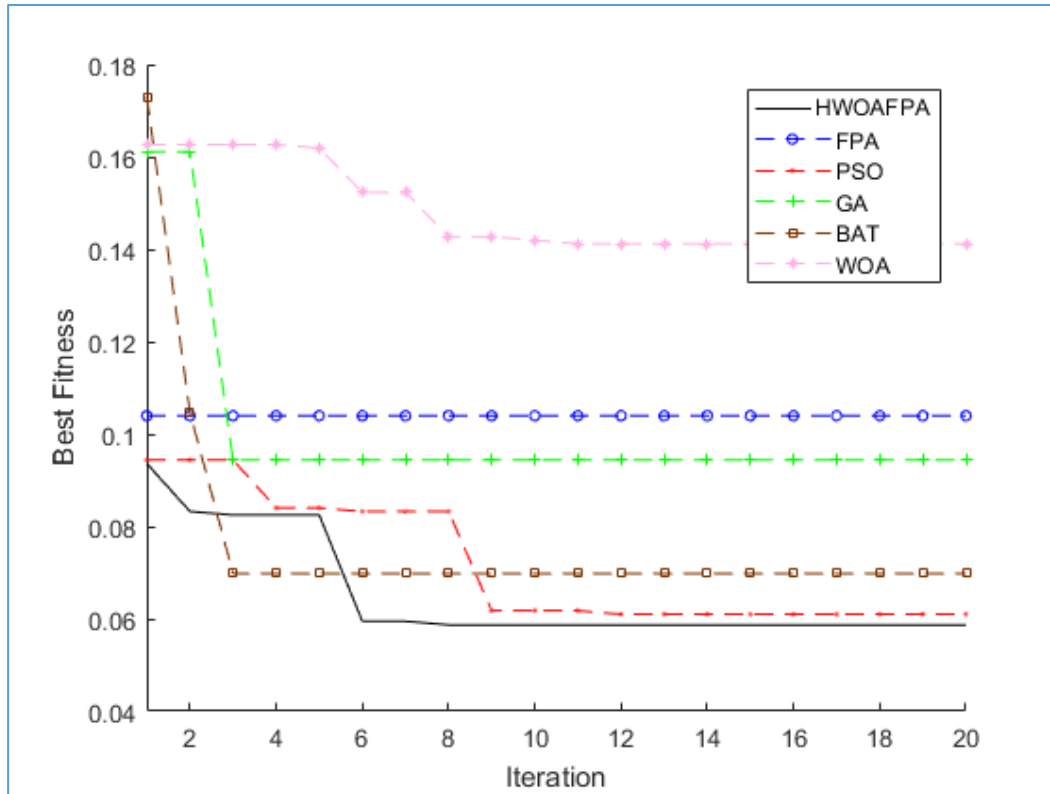
Fig. 17. *Comparison between the proposed approaches and the state-of-the-art approaches in terms of convergence-Dataset Wine.*

As the results of (17) is shown, the proposed algorithm has better performance than other comparison algorithms have been on the convergence objective. In this experiment, the proposed algorithm had an acceptable performance compared to the WOA and FPA at the beginning; and with increasing the repetitions, it generated better results using the solutions and their opposite space at the final repetitions compared to all other algorithms. Figure 18 shows the results of comparing the proposed method with WOA, GA, PSO, BAT, and FPA in terms of the convergence rate of the target function on the Zoo dataset.
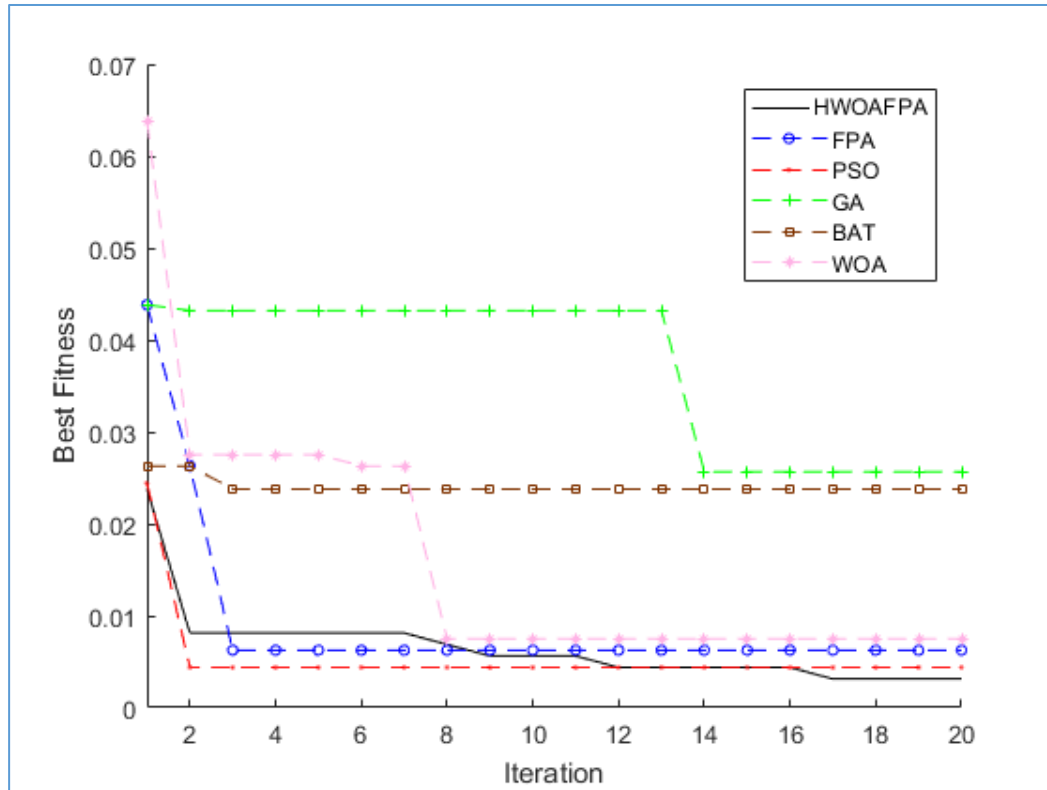
Fig. 18. *Comparison between the proposed approaches and the state-of-the-art approaches in terms of convergence-Dataset Zoo.*

As the results of (18) is shown, the proposed algorithm has better performance than other comparison algorithms have been on the convergence objective. In this test, the performance of the proposed algorithm and other algorithms is roughly the same because of the small size of the Zoo dataset. So that the WOA, PSO, FPA and the proposed algorithms performs similarly with increasing the repetition of generations. Of course, the proposed algorithm, with increasing the repetitions, generated better results using the solutions and their opposite space at the final repetitions compared to all other algorithms.

Considering the results reported in subsections 5.1.1 and 5.1.2, it seems that the proposed algorithm has a significant advantage in terms of choosing fewer features, the higher classification accuracy and convergence rate of the target function compared to other metaheuristic algorithms. We have succeeded to present a powerful algorithm with high convergence rate and high accuracy for solving the FS optimization problem through a combination of WOA and FPA and using an OBL method.

5.2. Email Spam Detection

In subsection 5.1, we implemented our proposed algorithm on 10 valid UCI datasets; the results showed that the proposed algorithm had a significantly better performance compared to other metaheuristic algorithms in terms of fewer features selection and classification accuracy. So, given the strong results of the proposed algorithm on the valid UCI datasets, we were encouraged to implement our proposed algorithm on the spam emails datasets in order to make sure that the proposed algorithm still has a better performance. To do this, we use the valid spam email dataset describe in subsection 5.2.1. We divided all data into the test and training datasets using classification sampling, and tried to include both spam and non-spam emails in both test and training datasets to better evaluate the performance and efficiency of the

proposed algorithm and other algorithms. We also divide spam dataset into two groups of the training and test datasets as follows.

• 70% of the entire dataset was used for training and construction of the proposed implementation model;
• 30% of the remaining dataset was used for testing and validation of the model.

5.2.1. Spam base dataset analysis

The spam base dataset is an acquisition from email spam message and we can achieve the corpus benchmark from it. In [44], the total number of acquired messages in the dataset were 4601 and 39 percent of them (i.e. 1813 messages) were marked to be spam messages and the remaining 61 percent (i.e. 2788 messages) were identified as non-spam. The non-spam message was acquired from a single mailbox and gave by Forman. It should be noted that, unlike most corpora which come in their raw form, acquisition of this corpus was preprocessed. The features or instances in the survey were represented as 58-dimensional vectors. 48 features of the corpus of 58 features were enlisted as most unbalanced words for the class spam and they were represented by words produced from the original messages without stop list or stemming. The remaining 10 features included: 3 features were a representation of different measure of manifestation of capital letters that existed in the texts, 6 features were the percentage of manifestation of the special characters "#", ";", "(", "$", "[", "!" and the final feature was the class label in the corpus which gave the condition of an instance to be spam '1' or non-spam '0'. As described in [45] , spam base dataset is considered among one of the best test bed that carries out good during evaluation and learning methods.

5.2.2. Experimental Results and Discussion

In this subsection, we will show the results of the implementation of the proposed algorithm and other comparative algorithms on the spam emails dataset; and then, we will discuss the results. We divided the spam e-mails dataset into the test (30%) and training (70%) datasets; both test and training datasets contain spam and non-spam emails to better evaluate the performance and efficiency of the proposed algorithm and other comparative algorithms in this subsection. We also considered the initial population of all algorithms to be 10 and the number of repetitions to be 1-100. Now in this subsection, we first compare the proposed algorithm with WOA and FPA algorithms in terms of classification accuracy and average accuracy of classification with different repetitions; and then in the next step, the proposed algorithm and other comparative algorithms will be tested on the spam e-mail dataset in terms of the accuracy of the classification with different repetitions.

In the first test of this subsection, the performance of WOA, FPA and HWOAFPA in terms of accurately detecting spam emails is shown in Figure 9. Parameters are set to a maximum number of repetitions of 1-100 and a population size of 10.
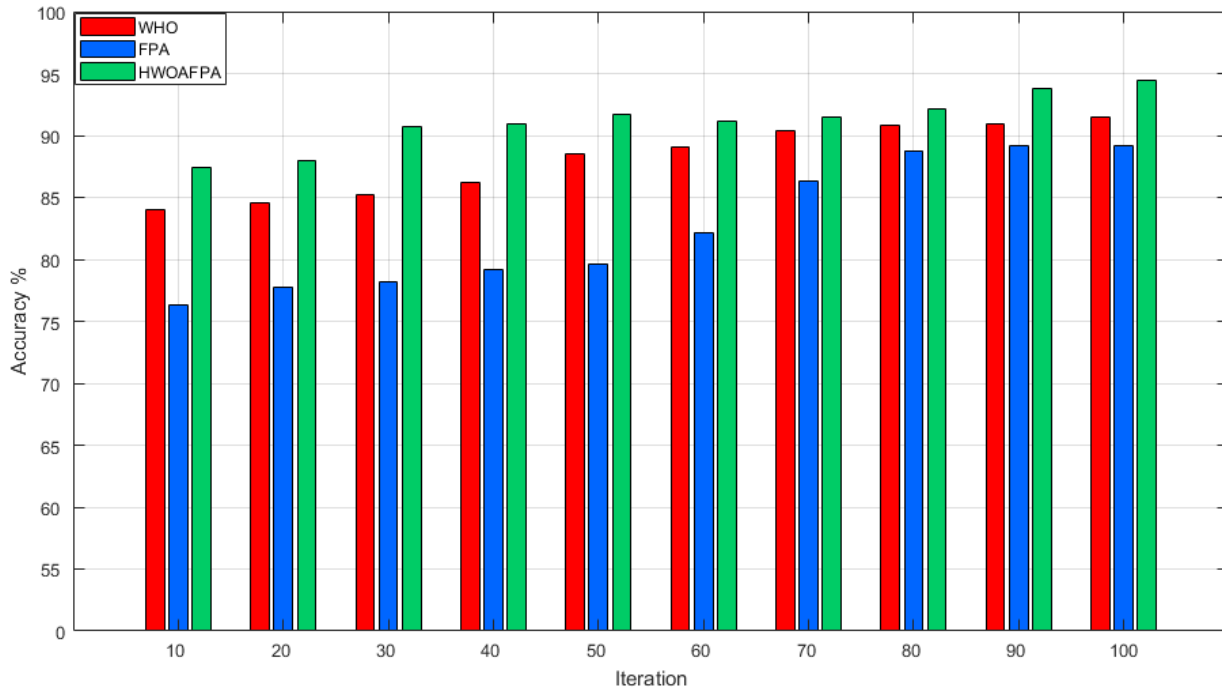
Fig. 9. Best accuracy for WOA ، FPA and HWOAFPA .

The results of the first experiment in this subsection (see Fig. 9) show that the higher the number of repetitions, the better is the accuracy of results obtained by the proposed algorithm compared to that of the WOA and the FPA in the detection of spam emails. The proposed algorithm has performed similar to other algorithms in first repetitions; however, with increasing the number of repetitions, as shown in Figure 9, it can be seen that due to more use of the OBL method to create solutions in the opposite search space, the proposed method outperformed WOA and FPA.

For further experiments in this subsection, in addition to testing classification accuracy, we examined the proposed algorithm in terms of the average accuracy of spam emails detection, including the total population of proposed algorithm, WOA and FPA. The average accuracy of WOA, FPA, and HWOAFPA algorithms in spam emails detection is shown in Figure 10. Parameters are set to a maximum number of repetitions of 1-100 and a population size of 10.
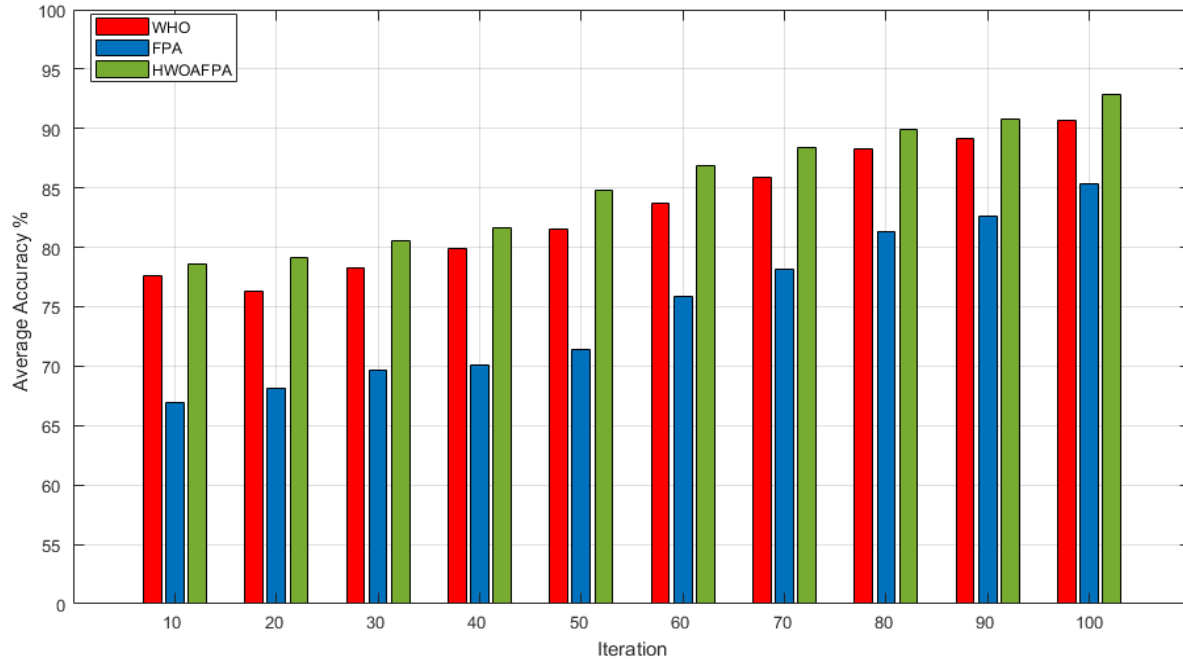
Fig. 10. Average accuracy of each number of iteration for WOA ، FPA and HWOAFPA .

The results of the second experiment in this subsection show that, with increasing number of repetitions, the results obtained by the proposed algorithm in terms of the average accuracy of spam emails detection are much better than that of WOA and FPA. This test proves that the proposed algorithm changes its total population in higher numbers of repetitions and improves all the solutions in the search space. Now, after comparing the proposed algorithm with WOA and FPA, and showing the superiority of the proposed algorithm in terms of classification accuracy and average classification accuracy, in order to further evaluate the proposed algorithm, we compared the proposed algorithm with other metaheuristic algorithms such as GA, PSO and BAT algorithm in terms of classification accuracy with different repetitions.
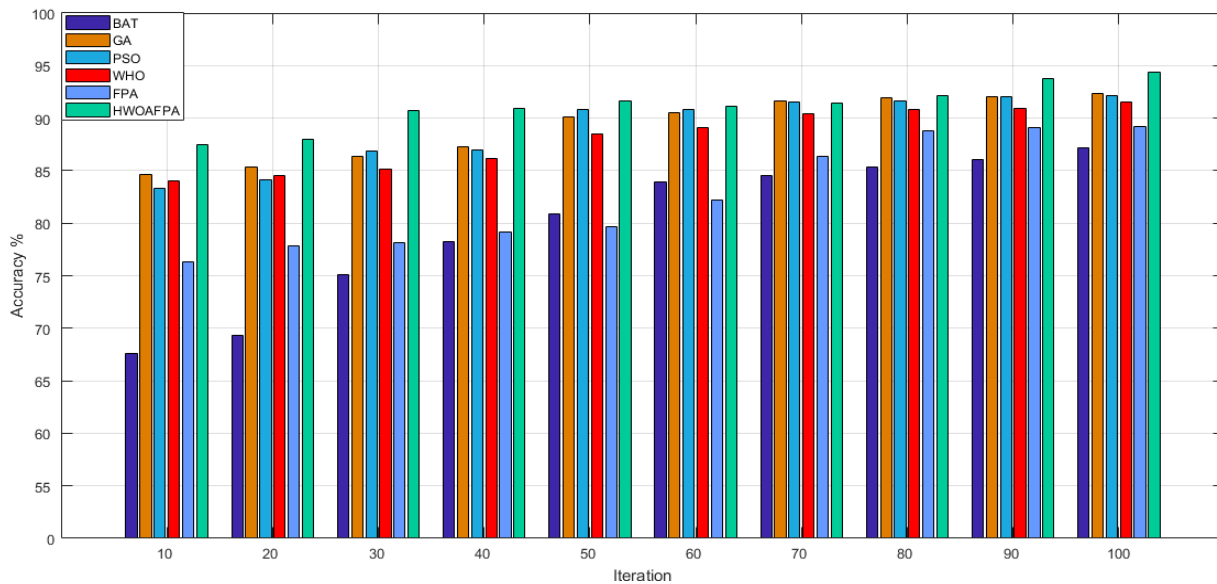
Fig. 11 . Evaluation Best accuracy of Proposed Algorithm (HWOAFPA) and Its Comparison with other Algorithms

The results of the third experiment in this subsection, according to Fig. 11, show that the proposed algorithm yielded stronger results than comparative algorithms such as GA, PSO, BAT, whale optimizer and FPA in terms of the accuracy of spam emails detection. The results of these three experiments regarding spam e-mails detection revealed that the proposed method was well-combined and we succeeded in presenting a powerful algorithm for detecting spam emails using a hybrid OBL method.

6. Conclusion

A FS method should search the best possible subset of the features among all possible subsets in the desired dataset and extract these features based on a specific evaluation criterion. Applying the FS in the dataset before implementing the learning process is essential in improving the efficiency of the classification process. Finding this optimal subset can be considered as an optimization issue, and random, heuristic and metaheuristic search methods may be used to find these optimal or semi-optimal subsets. The important point is that selecting the subset of features through traditional and heuristic optimization methods is not effective and efficient when the search space is of high dimensions; the metaheuristic algorithms have been presented as the most suitable option for solving this problem by many researchers. Also, metaheuristic algorithms can be combined to increase the convergence rate and accuracy of the generated solution. In this paper, we used the natural processes of WOA and FPA to solve the problem of optimization of FS, and on the other hand, we used an OBL method to ensure the convergence rate and accuracy of the proposed algorithm. In fact, in the proposed method, WOA create solutions in their search space using the prey siege and encircling process, bubble invasion and search for prey methods, and try to improve the solutions for the FS problem; along with this algorithm, FPA improves the solution of the FS problem with two global and local search processes in an opposite space with the solutions of the WOA. In fact, we used all of the possible solutions to the FS problem from both the solution search space and the opposite of solution search space. To evaluate the performance of the proposed algorithm, experiments were carried out in two steps. In the first stage, the experiments were performed on 10 FS datasets from the UCI data repository. In the second step, we tried to test the performance of the proposed algorithm in terms of spam e-mails detection. The results obtained from the first step showed that the proposed algorithm, performed on 10 UCI datasets, was more successful in terms of the average size of selection and classification accuracy than other basic metaheuristic algorithms. Also, the results from the second step showed that the proposed algorithm which was run on the spam e-mail dataset, performed much more accurately than other modern metaheuristic algorithms in terms of accuracy of detecting spam e-mails.

7. References

1.   Mafarja, M., et al., *Evolutionary Population Dynamics and Grasshopper Optimization Approaches for Feature Selection Problems.* Knowledge-Based Systems, 2017.
2.   Mafarja, M.M. and S. Mirjalili, *Hybrid Whale Optimization Algorithm with simulated annealing for feature selection.* Neurocomputing, 2017. **260**: p. 302-312.
3.   Han, J., J. Pei, and M. Kamber, *Data mining: concepts and techniques*. 2011: Elsevier.
4.   Aghdam, M.H., N. Ghasem-Aghaee, and M.E. Basiri, *Text feature selection using ant colony optimization.* Expert systems with applications, 2009. **36**(3): p. 6843-6853.
5.   Zorarpacı, E. and S.A. Özel, *A hybrid approach of differential evolution and artificial bee colony for feature selection.* Expert Systems with Applications, 2016. **62**: p. 91-103.
6.   Huang, J., Y. Cai, and X. Xu, *A hybrid genetic algorithm for feature selection wrapper based on mutual information.* Pattern Recognition Letters, 2007. **28**(13): p. 1825-1844.

7.   Kabir, M.M., M. Shahjahan, and K. Murase, *A new hybrid ant colony optimization algorithm for feature selection.* Expert Systems with Applications, 2012. **39**(3): p. 3747-3763.

8.   Khalid, S., T. Khalil, and S. Nasreen. *A survey of feature selection and feature extraction techniques in machine learning.* in *Science and Information Conference (SAI), 2014.* 2014. IEEE.

9.   Xue, B., et al., *A survey on evolutionary computation approaches to feature selection.* IEEE Transactions on Evolutionary Computation, 2016. **20**(4): p. 606-626.

10.   Rajamohana, S. and K. Umamaheswari, *Hybrid approach of improved binary particle swarm optimization and shuffled frog leaping for feature selection.* Computers & Electrical Engineering, 2018.

11.   Xue, B., M. Zhang, and W.N. Browne, *Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms.* Applied Soft Computing, 2014. **18**: p. 261-276.

12.   Wang, Z., M. Li, and J. Li, *A multi-objective evolutionary algorithm for feature selection based on mutual information with a new redundancy measure.* Information Sciences, 2015. **307**: p. 73-88.

13.   Emary, E., H.M. Zawbaa, and A.E. Hassanien, *Binary ant lion approaches for feature selection.* Neurocomputing, 2016. **213**: p. 54-65.

14.   Nakamura, R.Y.M., et al., *Binary bat algorithm for feature selection*, in *Swarm Intelligence and Bio-Inspired Computation*. 2013, Elsevier. p. 225-237.

15.   Zhang, L., et al., *Feature selection using firefly optimization for classification and regression models.* Decision Support Systems, 2017.

16.   Mafarja, M. and S. Mirjalili, *Whale optimization approaches for wrapper feature selection.* Applied Soft Computing, 2018. **62**: p. 441-453.

17.   Kabir, M.M., M. Shahjahan, and K. Murase, *A new local search based hybrid genetic algorithm for feature selection.* Neurocomputing, 2011. **74**(17): p. 2914-2928.

18.   Sayed, S.A.-F., E. Nabil, and A. Badr, *A binary clonal flower pollination algorithm for feature selection.* Pattern Recognition Letters, 2016. **77**: p. 21-27.

19.   Liao, T. and R. Kuo, *Five Discrete Symbiotic Organisms Search Algorithms for Simultaneous Optimization of Feature Subset and Neighborhood Size of KNN Classification Models.* Applied Soft Computing, 2017.

20.   Tawhid, M.A. and K.B. Dsouza, *Hybrid Binary Bat Enhanced Particle Swarm Optimization Algorithm for solving feature selection problems.* Applied Computing and Informatics, 2018.

21.   Shunmugapriya, P. and S. Kanmani, *A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC Hybrid).* Swarm and Evolutionary Computation, 2017. **36**: p. 27-36.

22.   Oh, I.-S., J.-S. Lee, and B.-R. Moon, *Hybrid genetic algorithms for feature selection.* IEEE Transactions on pattern analysis and machine intelligence, 2004. **26**(11): p. 1424-1437.

23.   Khushaba, R.N., et al. *A combined ant colony and differential evolution feature selection algorithm.* in *International Conference on Ant Colony Optimization and Swarm Intelligence*. 2008. Springer.

24.   Olabiyisi, S., et al., *Hybrid metaheuristic feature extraction technique for solving timetabling problem.* Int. J. Sci. Eng. Res, 2012. **3**(8): p. 1-6.

25.   Canuto, A.M. and D.S. Nascimento. *A genetic-based approach to features selection for ensembles using a hybrid and adaptive fitness function*. in *Neural Networks (IJCNN), The 2012 International Joint Conference on*. 2012. IEEE.

26.   Akila, M.M., *Hybrid local feature selection in DNA analysis based cancer classification.* Indian Journal of Computer Science and Engineering, 2012. **3**(3): p. 470-475.

27.   Babatunde, R., S. Olabiyisi, and E. Omidiora, *Feature dimensionality reduction using a dual level metaheuristic algorithm.* optimization, 2014. **7**(1).

28.   Hasani, S.R., Z.A. Othman, and S.M.M. Kahaki, *Hybrid feature selection algorithm for intrusion detection system.* Journal of Computer Science, 2014. **10**(6): p. 1015.

29.   Kashef, S. and H. Nezamabadi-pour, *An advanced ACO algorithm for feature subset selection.* Neurocomputing, 2015. **147**: p. 271-279.

30.   Nekkaa, M. and D. Boughaci, *Hybrid harmony search combined with stochastic local search for feature selection.* Neural Processing Letters, 2016. **44**(1): p. 199-220.

31.   Mirjalili, S. and A. Lewis, *The whale optimization algorithm.* Advances in Engineering Software, 2016. **95**: p. 51-67.

32.   Glover, B., *Understanding flowers and flowering second edition*. 2014: OUP Oxford.

33.   Waser, N.M., *Flower constancy: definition, cause, and measurement.* The American Naturalist, 1986. **127**(5): p. 593-603.

34.   Pavlyukevich, I., *Lévy flights, non-local search and simulated annealing.* Journal of Computational Physics, 2007. **226**(2): p. 1830-1844.

35.   Yang, X.-S. *Flower pollination algorithm for global optimization*. in *International conference on unconventional computing and natural computation*. 2012. Springer.

36.   Rodrigues, D., et al., *Binary flower pollination algorithm and its application to feature selection*, in *Recent advances in swarm intelligence and evolutionary computation*. 2015, Springer. p. 85-100.

37.   Rahnamayan, S., H.R. Tizhoosh, and M.M. Salama, *Opposition-based differential evolution.* IEEE Transactions on Evolutionary computation, 2008. **12**(1): p. 64-79.

38.   Shaw, B., V. Mukherjee, and S. Ghoshal, *A novel opposition-based gravitational search algorithm for combined economic and emission dispatch problems of power systems.* International Journal of Electrical Power & Energy Systems, 2012. **35**(1): p. 21-33.

39.   Rahnamayan, S., H.R. Tizhoosh, and M.M. Salama, *Opposition versus randomness in soft computing techniques.* Applied Soft Computing, 2008. **8**(2): p. 906-918.

40.   Altman, N.S., *An introduction to kernel and nearest-neighbor nonparametric regression.* The American Statistician, 1992. **46**(3): p. 175-185.

41.   Mabodi, K., Yusefi, M., Zandiyan, S., Irankhah, L., & Fotohi, R. (2020). Multi-level trust-based intelligence schema for securing of internet of things (IoT) against security threats using cryptographic authentication. The Journal of Supercomputing, 1-25.

42.   Seyedi, B., & Fotohi, R. (2020). NIASHPT: a novel intelligent agent-based strategy using hello packet table (HPT) function for trust Internet of Things. The Journal of Supercomputing, 1-24.

43.   Fotohi, R. (2020). Securing of Unmanned Aerial Systems (UAS) against security threats using human immune system. Reliability Engineering & System Safety, 193, 106675.

44.   Fotohi, R., Firoozi Bari, S., & Yusefi, M. (2020). Securing Wireless Sensor Networks Against Denial-of-Sleep Attacks Using RSA Cryptography Algorithm and Interlock Protocol. International Journal of Communication Systems, 33(4), e4234.

44.   Asuncion, A. and D. Newman, *UCI machine learning repository*. 2007.

45.   Merz, C.J. and P.M. Murphy, *{UCI} Repository of machine learning databases.* 1998.

46.   Friedman, J., T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Vol. 1. 2001: Springer series in statistics New York.

47.   Mark Hopkins, E.R., George Forman, Jaap Suermondt *UCI Machine Learning Repository: Spambase Data Set*, in *Hewlett-Packard Labs*. 1999.

48.   Koprinska, I., et al., *Learning to classify e-mail.* Information Sciences, 2007. **177**(10): p. 2167-2187.