IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 22, 2021, accepted July 10, 2021, date of publication July 20, 2021, date of current version August 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3098808

A Novel Auction-Based Optimization Algorithm and Its Application in Rough Set Feature Selection

NAJMEH SADAT JADDI [1,2] AND SALWANI ABDULLAH [3]

[1]Faculty of Electrical and Computer Engineering, Tarbiat Modares University, Tehran 14117-13116, Iran
[2]Department of Computer Engineering, Ooj Institute of Higher Education, Abyek 3441311151, Qazvin, Iran
[3]Center for Artificial Intelligence Technology, Data Mining and Optimization Research Group (DMO), Faculty of Information Science and Technology, National University of Malaysia, Selangor, Bangi 43600, Malaysia

Corresponding author: Najmeh Sadat Jaddi (najmehjaddi@gmail.com)

This work was supported by the National University of Malaysia under Grant DIP-2016-024.

ABSTRACT The selection of features from data, as one of the most important tasks in data mining, strongly affects the accuracy of classification. The removal of irrelevant and redundant features from data while simultaneously avoiding information loss is the main objective of feature selection. Feature selection is possible using rough set theory and meta-heuristic algorithms. In this paper, a novel auction-based optimization algorithm (ABOA) is proposed to contribute to generating an effective algorithm with a good trade-off between exploration and exploitation. This new algorithm simulates the auction sale process, where bidders offer higher/lower amounts to outbid each other. Auctions are categorized into ascending auctions and descending auctions and thus respectively represent maximization and minimization problems in ABOA. In the first step of the ABOA after initialization, a predefined number of bidders is selected and an auction is performed between them. The winner is selected and another auction is performed between the winner and a predefined number of the winner's neighbors. The winner of this round of auction is added to the winner list. This process is iterated until a predefined number of winners is found. Finally, one more auction is performed between all the winners on the winner list and the winner of that auction becomes the final winner. The algorithm with different parameter setting scenarios is tested on 25 benchmark test functions. The algorithm with the best results is then used to perform feature selection on 18 UCI datasets. The feature selection and classification accuracy results are compared with state-of-the-art results. The statistical analysis of the results proves the ability of the algorithm to solve optimization problems.

INDEX TERMS Artificial intelligence, auction-based optimization algorithm, feature selection, rough set theory.

I. INTRODUCTION

Finding a high-quality solution from all possible options is defined as an optimization problem. It has been proven that meta-heuristic algorithms can solve optimization problems effectively and they are now commonly used to solve such problems. Nature-inspired meta-heuristic algorithms, which are based on the behavior of physical or biological systems, have become particularly successful in many fields and optimization problems over the past years. Some applications of optimization problems include: timetabling and scheduling [1]–[4], industry [5]–[7], data mining [8]–[19], engineering [20]–[22], pattern recognition [23]–[26], and economics [27], [28]. In the field of data mining, classification is one of the key tasks and feature selection is an important pre-processing step for successful classification. Feature selection eliminates unnecessary features from data and results in an improvement in the ability of the classification algorithm [29]. One of the methods employed for feature selection is to search for a minimal subset (using meta-heuristic algorithms) within a full set of features that has the same level of discernibility as that of the full feature set. Many recent results for the feature selection methods using meta-heuristic algorithms are available in the literature [30]–[36]. This type of search for a solution can be achieved by the application of rough set theory, which

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

was proposed in [37]. Rough set is a device with a numerical establishment to manage loose information. It has been generally applied in artificial intelligence (AI), data mining, and knowledge discovery.

Although many existing algorithms show high performance to solve optimization problems, solving large-scale and complex optimization problems may generate a large search space, so maybe current optimization approaches are unable to present a high-quality solution. To achieve the goal of the superior performance of the algorithm, a strong balance of exploration and exploitation in the algorithm is required. Fig. 1 illustrates a schematic analysis of the exploration versus exploitation in some traditional algorithms. This figure shows the reason for interests and attempts to introduce perfect alternative algorithms for solving complex problems in recent researches [18], [19], [38]–[41].

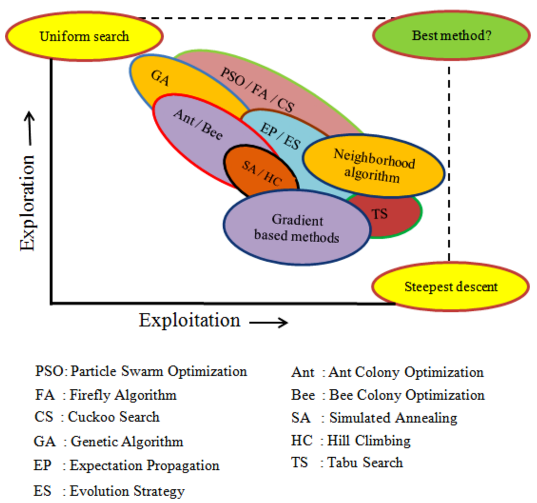

FIGURE 1. A schematic analysis of the exploration versus exploitation in some traditional algorithms.

In this paper, a new optimization algorithm that can be applied for rough set feature selection is proposed. The algorithm simulates the auction process in which bidders suggest increasing or decreasing (ascending or descending auction) amounts of recompense for an item or a service to outbid or undersell each other. The proposed algorithm is given the name Auction-Based Optimization Algorithm (ABOA) and starts by initializing the parameters and population. Next, the main loop of this algorithm starts with the selection of a predefined number of random bidders. Next, an auction is performed between the selected bidders and the winner of that auction is selected. The next step in this algorithm, in contrast to a real-world auction, is that the winner organizes another auction between itself and its internal and neighborhood bidders to try to find a better amount. Then, the winner of this additional auction is added to the winner list. This process is repeated until the predefined number of winners is reached. When that number has been reached, in the final step, an auction between all of the members already on the winner list, and all of the neighbors of the winners is performed and it is the winner of this final auction that is deemed the final winner. In ABOA, the selection of random bidders provides an exploration of the search space while the auction between neighbors plays the role of exploitation. Therefore, a trade-off between exploration and exploitation exists in this algorithm.

ABOA contributes to attempt a good balance between exploration and exploitation in this algorithm and therefore introduces a more productive and effective algorithm. In this algorithm, the selection of random bidders helps the algorithm to explore the search space for any alternative bidder with probably a better bid while the auction between neighbors plays the role of exploitation in the area around the winner. This provides the algorithm with a strong trade-off between exploration and exploitation.

The rest of this paper is organized as follows: Section 2 presents the works related to this work in the literature. The details of the proposed algorithm are presented in section 3. Section 4 presents a discussion and analysis of the experimental results when the algorithm is applied to some benchmark test functions and feature selection problems. Section 5 concludes the paper.

II. RELATED WORKS

Meta-heuristic algorithms proposed since long ago include the simulated annealing (SA) algorithm, which was designed by simulating the steel annealing procedure [42] and the great deluge (GD) algorithm [43], which imitates the process of someone in a heavy rainstorm climbing up a hill while trying to move to any track that does not get their feet wet with the ultimate aim of traveling in an upwards direction as the water level increases. The GD algorithm is similar to the SA algorithm and the hill-climbing algorithm (HC). On the other hand, the law of gravity forms the basis of the gravitational search algorithm (GSA) in which the searcher agents are a set of masses that interrelate with each other and whose actions are rooted in Newtonian gravity and the rule of movement [44]. Two notable algorithms based on biological systems are the genetic algorithm (GA), which is based on natural genetic variation [45], and ant colony optimization, which mimics the ability of real ants to find the shortest path between their nest and food sources [46]. Another path-seeking meta-heuristic of note is the intelligent water drops algorithm, which imitates the behavior of natural rivers that find the optimal pathway to their destination [47]. On the other hand, swarm behavior is the fundamental principle that underlies the particle swarm optimization (PSO) algorithm [48]. The fish [49] and bird [50] swarm optimization algorithms are also examples of this type of optimization algorithm. Then there is the bat algorithm, which is based on the behavior of bats as they try to find their prey through echolocation [51]. Similarly, the honey bee optimization algorithm mimics the food-foraging behavior of honey bee colonies [52]. Another approach inspired by the honey bee is the self-explanatory honey bee mating optimization algorithm [53]. The principle of attraction is also applied in the firefly algorithm, which imitates the flashing brightness emitted by fireflies [54]. In contrast, the cuckoo search algorithm was inspired by the behavior of the cuckoo, which lays its eggs in the nests of other types of birds [55]. Other methods of finding a solution include the use of the harmony search algorithm, which was inspired by discovering the harmony in music [56], and the black hole algorithm, which was designed based on the black hole phenomenon [57] where stars that are close to the black hole are pulled towards it and those that are too close are swallowed up by it and disappear forever. In the case of the imperialist competitive algorithm, the competition among empires provides the basis for its structure [58]. The kidney-inspired algorithm is rooted in the filtration process performed by the kidneys in the human body [59]. A trader algorithm was proposed in [60]. The cell separation algorithm (CSA) [18] imitates cell separation action by using a differential centrifugation process involving multiple centrifugation steps and increasing the rotor speed in each step.

The use of rough set theory in machine learning and AI for feature selection particularly is for classification problems. This is done through searching a reduced set of features. A reduced set is a subset of all features which retains the classification accuracy of the full feature set. The application of meta-heuristic algorithms for rough set feature selection has been studied in-depth in the literature and has shown promising results. Some examples can be found in [10], [11], [17], [61]–[67]. The results of more recent research in this area can be found in [68], [69].

III. PROPOSED METHOD

In this section, a brief explanation of the rough set theory and the overall process of auctions in the market are given as preliminaries in the first subsection. In the second subsection, the details of the ABOA proposed are presented.

A. PRELIMINARIES

The basics of rough set theory and the auction process are provided in the next subsection. This offers the idea behind the ABOA as the proposed method.

1) ROUGH SET THEORY

The concepts of equivalence relation and dependency degree calculation was proposed in [37]. Based on these concepts: let (U, A) be an information system, where U is a non-empty set of objects and A is a non-empty set of features in which α: U $\rightarrow v_\alpha$ for each $\alpha \in A$. With any $P \subset A$ the equivalence relation *IND* (*P*) is calculated as shown in Eq. (1):

$$IND(P) = \{(x, y) \in U^2 | \forall \alpha \in P, \alpha(x) = \alpha(y)\} \tag{1}$$

The region of U, constructed by *IND* (*P*) is shown as $U/IND(P) = \otimes\{\alpha \in P : U/IND(\{\alpha\})\}$, where $A \otimes B = \{X \cap Y : \forall X \in A, \forall Y \in B, X \cap Y \neq \phi\}$. If $(x, y) \in IND(P)$, then x and y are indiscernible by features from P. $[x]_P$ is the representation of the equivalence classes of the P indiscernible relation.

Then, let $X \subseteq U$, where $\underline{P}X = \{x|[x]_P \subseteq X\}$ is the definition of the P-lower approximation $\underline{P}X$ of set x and $\overline{P}X = \{x|[x]_P \cap X \neq \phi\}$ is the definition of the P-upper approximation $\overline{P}X$. If we consider the P and Q equivalence relations over U, then the positive regions can be specified by Eq. (2):

$$POS_p(Q) = \bigcup_{X \in U/Q} \underline{P}X \tag{2}$$

One of the important tasks in data analysis is finding dependencies between features (genes). For P $\subset A$, Q depends on P in a degree k $(0 \leq k \leq 1)$, as shown in Eq. (3):

$$k = \gamma_P(Q) = \frac{|POS_P(Q)|}{|U|} \tag{3}$$

2) AUCTION OPERATION

An auction can be defined as a procedure of offering items or services that bidders bid on and the auction is completed by assigning the items or services to the bidder that bids/offers the best amount of recompense (Fig. 2(a)). An auction can be an ascending or descending price auction. The ascending price auction is the most popular form of the auction and is normally used for selling items, whereas the descending price auction is normally used by agents or companies offering services. The ascending price auctions are carried out interactively in real-time; with bidders present either physically or electronically. The seller gradually raises the price, bidders drop out until finally, only one bidder remains, and that bidder wins the object at this final price. In comparison, descending price auction is also an interactive auction format, in which the seller gradually lowers the price from some high initial value until the first moment when some bidder accepts and pays the current price. The actions of ascending price and descending price are shown in Fig.2(b) and Fig.2(c), respectively.

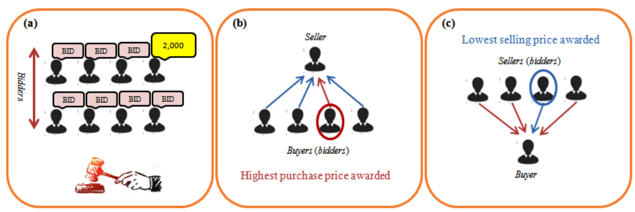

FIGURE 2. Auction process: (a) schematic overall auction procedure, (b) ascending price auction, (c) descending price auction.

B. AUCTION-BASED OPRIMIZATION ALFORITHM (ABOA)

The ABOA algorithm presented in this paper is rooted in auction behavior. In this algorithm, each candidate solution plays the role of the bidder and is a candidate to be a winner of the auction. The best solution among all candidate solutions in ABOA is the winner of the auction. The operation of finding the best solution in this algorithm is an imitation of the operation of finding a winner when an auction is running. The random solution involved in the ABOA algorithm is similar to the involvement of new bidders in auctions. In ABOA,

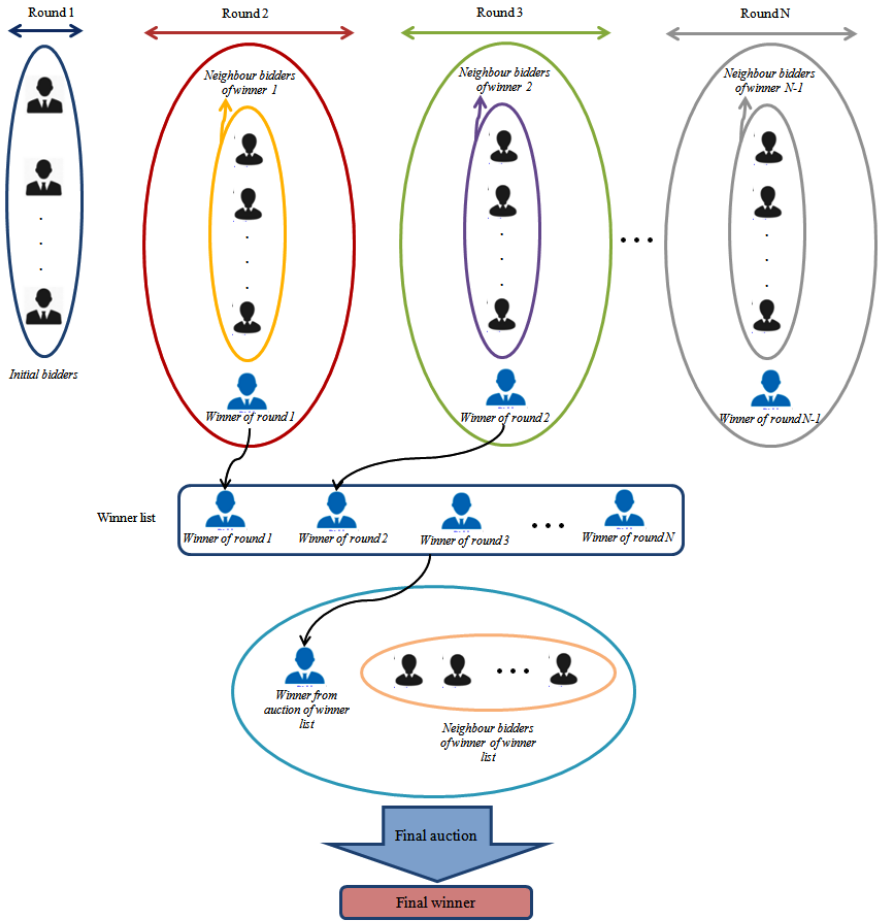

FIGURE 3. Iterative process in ABOA.

the best solution for each round is added to a list, similar to the listing of winners of each round in an auction. The analogy of ABOA with auction behavior is presented in Table 1.

TABLE 1. Analogy of ABOA with auction behavior.

Auction behavior	ABOA
Bidders	Solutions
Winner	Best solution
Performing an auction among bidders	Finding the best solution from among all the solutions
Involving new bidders in the auctions	Involving a new random solution in the global area or a neighbor solution in the local area
Listing the winners of each round	Adding the best solution from each round to the winner list

The ABOA procedure starts after the parameters and the population of the algorithm have been set in the initialization step. In the initialization step, the number of random candidate bidders and the number of neighbors for each round are set as the parameters of the algorithm. After initialization, the algorithm starts its iterative process. The details of this iterative process are shown in Fig. 3. First, the selection of an initialized number of random bidders is performed and an auction between them is completed (Round 1 in Fig. 3). In this step, an initialized number of neighbor bidders for the winner of Round 1 are generated.

Round 2, in Fig.3, is followed by an auction between the winner of Round 1 and its neighbors and the winner of this round is used for Round 3. In Round 3, this winner and its neighbors are used for the auction process. This process is repeated and the winner of each round is added to the winner list until the initialized number of winners (N in Fig. 3) is reached. After the completion of this iterative process, a list of winners with good bids is available. Next, another auction between all the winners in the winner list is carried out and one more auction between the winner of that auction and the initialized number of its neighbors is organized. The winner of this final auction is returned as the final winner.

The details of the components involved in the ABOA are explained in the following subsections.

1) INITIALIZATION AND FUNCTION EVALUATION

Same as most meta-heuristic algorithms, ABOA constructs an initial population of bidders to start the process of optimization. The initial concentrations are created based on the number of candidate bidders and the dimension of search space. The initial population of bidders is constructed with uniform random distribution. Random bidders are generated using Eq. 4:

$$Br_{random} = Br_{min} + Rand(Br_{max} - Br_{min}) \tag{4}$$

In this equation Br represents bidder. Br_{random} is a random bidder, and Br_{min} and Br_{max} denote the minimum and maximum values of bidders in the population of bidders, $Rand$ is a random value in the interval of [0,1]. Eq. 4 same as other meta-heuristic algorithms prepares the exploration ability of the algorithm by randomly initialization of the population.

2) AUCTION ROUND

Each auction round runs an auction between candidate bidders and finds the best bidder in which maximum bidder in maximization problem (ascending auction) and the minimum bidder in minimization problem (descending auction) among all candidate bidders is selected during an auction round. In ABOA, the first auction round is performed in the initial population of the random candidate bidders.

3) NEIGHBOR BIDDER

Neighbor bidders are the bidders around the winner bidder of the previous round of auction and are generated using Eq. 5:

$$Br_{neighbor} = win + rand(Br_{random} - win) \tag{5}$$

In Eq. 5, Br represents the bidder. The winner is represented by win. $Br_{neighbor}$ is the representation of the bidder in the neighborhood of the winner. Moreover, random bidder in the range of the dimension of search space is labeled by Br_{random}. A random number between 0 and 1 is represented by $rand$. Eq. 5 generates a new bidder while a slight balance of exploration (random bidder, Br_{random}) and exploitation (winner of the previous round, Win) is supplied.

4) WINNER LIST

A list of all winners of all auction rounds is maintained, from the first round to the last round, during the search process. Holding bidders with high quality provides the algorithm a chance of finding a much better bidder when in the last stage of the algorithm all winners involve in the final auction and finding final and probably the best bidder. This process increases the exploitation ability of the ABOA.

In ABOA, the selection of random bidders helps the algorithm to explore the search space for any alternative bidder with probably a better bid while the auction between neighbors plays the role of exploitation in the area around the winner (i.e., the best solution so far for, the current round). Therefore, a trade-off between exploration and exploitation exists in this algorithm. The flowchart and pseudocode of ABOA is presented in Fig. 4 and Fig. 5, respectively. The maximization version of this algorithm is a simulation of ascending auction and the minimization version of ABOA is an imitation of descending auction.

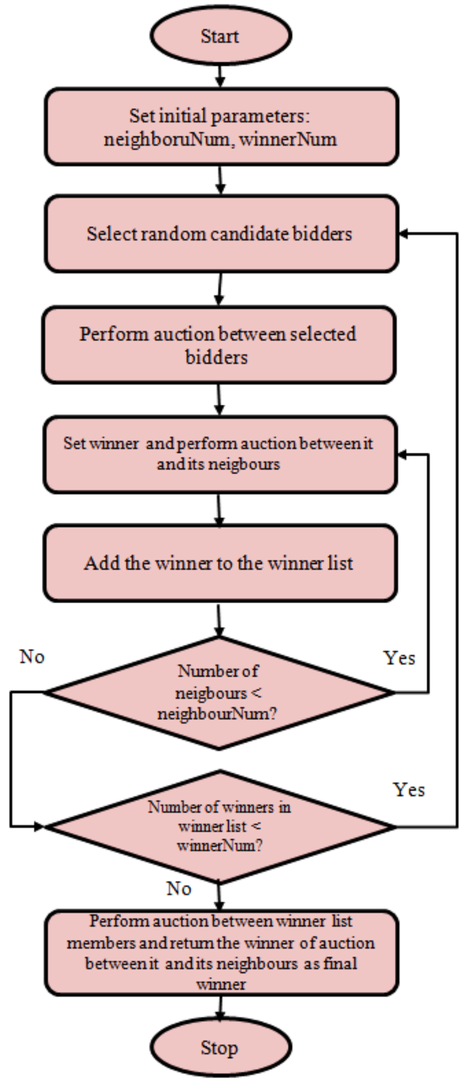

FIGURE 4. Flowchart of ABOA.

IV. EXPERIMENTAL RESULTS

Two experiments are performed to assess the capability of ABOA. In the first experiment, different ABOA scenarios are applied to 25 benchmark test functions and the results are compared with each other. In the second experiment, ABOA is applied to 18 benchmark feature selection problems and the results are compared with the results of some methods in the literature.

A. RESULTS OF TEST FUNCTIONS

In order to test the performance of ABOA, a set of 25 test functions known as the CEC 2005 benchmark functions is used. The test functions and their details are presented in Table 2, in which the test functions are classified into shifted (S1), separable (S2), scalable (S3), unimodal (U), multimodal (M), non-separable (N), and rotated (R).

In order to determine the parameter settings, nine ABOA scenarios are investigated. The scenarios and their specified

TABLE 2. Details of the CEC 2005 test functions.

Key	Test function	Range	Category	f(x)
f1	Shifted Sphere Function	[-100,100]	US1S2S3	-450
f2	Shifted Schwefels Problem 1.2	[-100,100]	UNS1S3	-450
f3	Shifted Rotated High Conditioned Elliptic Function	[-100,100]	URNS1S3	-450
f4	Shifted Schwefels Problem 1.2 with Noise in Fitness	[-100,100]	UNS1S3	-450
f5	Schwefels Problem 2.6 with Global Optimum on Bounds	[-100,100]	UNS3	-310
f6	Shifted Rosenbrocks Function	[-100,100]	MNS1S3	390
f7	Shifted Rotated Griewanks Function without Bounds	[0,600]	URNS1S3	-180
f8	Shifted Rotated Ackleys Function with Global Optimum on Bounds	[-32,32]	URNS1S3	-140
f9	Shifted Rastrigins Function	[-5,5]	MS1S2S3	-330
f10	Shifted Rotated Rastrigins Function	[-5,5]	MRNS1S3	-330
f11	Shifted Rotated Weierstrass Function	[-0.5,0.5]	MRNS1S3	90
f12	Schwefels Problem 2.13	[-π,π]	MNS1S3	-460
f13	Expanded Extended Griewanks plus Rosenbrocks Function (F8F2)	[-5,5]	MNS1S3	-130
f14	Shifted Rotated Expanded Scaffers F6	[-100,100]	MNS1S3	-300
f15	Hybrid Composition Function	[-5,5]	MS2S3	120
f16	Rotated Hybrid Composition Function	[-5,5]	MRNS3	120
f17	Rotated Hybrid Composition Function with Noise in Fitness	[-5,5]	MRNS3	120
f18	Rotated Hybrid Composition Function	[-5,5]	MRNS3	10
f19	Rotated Hybrid Composition Function with a Narrow Basin for the Global Optimum	[-5,5]	MNS3	10
f20	Rotated Hybrid Composition Function with the Global Optimum on the Bounds	[-5,5]	MNS3	10
f21	Rotated Hybrid Composition Function	[-5,5]	MRNS3	360
f22	Rotated Hybrid Composition Function with High Condition Number Matrix	[-5,5]	MNS3	360
f23	Non-Continuous Rotated Hybrid Composition Function	[-5,5]	MNS3	360
f24	Rotated Hybrid Composition Function	[-5,5]	MRNS3	260
f25	Rotated Hybrid Composition Function without Bounds	[2,5]	MRNS3	260

```
set number of neighbors, neighborNum
set number of winners, winnerNum
set list of winners, winnerList
set winnerCount ← 0
do while (winnerCount<winnerNum)
        select random bidders
        perform auction between selected bidders
        set win ← winner
        set neighborCount ← 0
        do while (neighborCount<neighborNum)
                perform auction between win and its neighbor
                bidder
                set win ← winner
        end while
        set winnerList ← win
end while
perform auction between winnerList members
set win ← winner
perform auction between winner and its neighbor bidders (same as inner
do while)
return win
```

FIGURE 5. Pseudocode of ABOA.

parameters are provided in Table 3. The algorithm is run 30 times in each scenario and the mean of the 30 results of each scenario is calculated. The results of this calculation are presented in Table 4.

TABLE 3. Parameter settings for ABOA scenarios.

Scenario	winnerNum	bidderNum	neighborNum
1	50	10	5
2	50	15	10
3	50	20	15
4	100	10	10
5	100	15	15
6	100	20	5
7	150	10	15
8	150	15	5
9	150	20	10

All the CEC 2005 test functions examined in this research are minimization problems. In order to determine which scenario achieves the best results, the rank of each scenario for each test function is calculated and the average of these ranks for each scenario is computed. The results of this computation are provided in Table 5. From Table 5, it is clear that scenario 4 (see Table 3), which has 100 winners, 10 bidders, and 10 neighbors, outperforms the other scenarios. Based on this outcome, the other experiments conducted in this study are performed using ABOA with the parameter settings of scenario 4.

TABLE 4. Means for ABOA scenarios.

Test function	scen1	scen2	scen3	scen4	scen5	scen6	scen7	scen8	scen9
f1	-437.977	-437.551	-438.155	-445.690	-438.718	-448.483	-438.879	-446.552	-439.724
f2	-441.011	-448.931	-441.397	-446.724	-444.842	-448.655	-445.216	-445.517	-449.138
f3	-443.278	-447.827	-442.063	-448.828	-442.313	-448.207	-446.198	-449.690	-439.807
f4	-440	-445.747	-444.851	-447.586	-443.25	-445.690	-445.647	-448	-442.325
f5	-307.517	-308.549	-308.368	-308.252	-307.347	-308.113	-307.542	-307.995	-308.012
f6	387.238	386.569	387.053	384.958	387.677	386.259	387.518	386.785	386.166
f7	-179.033	-105.995	-154.548	-142.904	-166.908	-155.257	-166.784	-167.534	-166.689
f8	-137.689	-138.413	-137.931	-138.655	-138.103	-138.828	-138.414	-138.759	-138.276
f9	-329.276	-329.414	-329.138	-329.448	-329.138	-329.379	-329.138	-329.517	-329.655
f10	-329.303	-329.534	-329.445	-329.583	-329.410	-329.534	-329.141	-329.400	-329.434
f11	90.724	90.551	90.655	90.586	90.620	90.448	90.482	90.620	90.344
f12	-459.414	-459.621	-459.552	-459.586	-459.724	-459.793	-459.586	-459.828	-459.759
f13	-129.655	-129.828	-129.793	-129.793	-129.759	-129.586	-129.828	-129.69	-129.724
f14	-298.769	-299.042	-298.780	-299.097	-298.769	-299.153	-298.927	-299.097	-299.162
f15	121.655	120.448	121.379	120.655	121.551	120.482	121.517	120.793	120.551
f16	120.103	120.241	120.241	120.206	120.275	120.275	120.310	120.172	120.137
f17	121.780	121.407	121.430	121.389	121.685	121.597	121.677	121.503	121.479
f18	10.896	10.413	10.793	10.137	10.620	10.379	10.689	10.137	10.241
f19	10.620	10.413	10.827	10.344	10.758	10.482	10.586	10.344	10.586
f20	10.758	10.551	10.655	10.482	10.827	10.517	10.689	10.448	10.482
f21	360.844	360.344	360.982	360.551	360.982	360.758	361.120	360.413	360.379
f22	361.103	360.413	360.810	360.224	360.810	360.396	360.862	360.655	361.051
f23	360.637	360.465	360.827	360.413	360.758	360.206	360.724	360.258	360.569
f24	260.586	260.310	260.724	260.448	260.655	260.241	260.413	260.379	260.172
f25	260.344	260.206	260.551	260.103	260.241	260.310	260.448	260.137	260.275

TABLE 5. Comparison of average of ranks of ABOA.

Scenario	Rank	Average of 25 ranks
scen1	9	7.32
scen2	4	3.68
scen3	7	6.6
scen4	1	3
scen5	8	6.76
scen6	3	3.6
scen7	6	6.16
scen8	2	3.24
scen9	5	3.88

Due to the symmetric 3D view provided by the eggcrate function, this function is used to visualize the step-by-step process performed by the algorithm. The whole area of the search space for the eggcrate function is shown in Fig. 6. The area of the eggcrate function discussed here is $-5 < x < 5$ and the location of the global minimum is at $x = (0, 0)$ and $f(x) = 0$.

The schematic in Fig. 7 shows the ABOA process during eggcrate minimization with contour plot background. The big red dots in the figure represent random bidders and the small purple dots are the bidders in the neighborhood of the winning bidder. In this experiment, the number of winners, bidders and neighbors was each set to 10 in order to simplify the visualization. Fig. 7(a), (b), (c), (d), (e), (f), (g), (h), (i), and (j) show the 10 steps of random bidder selection and performing auction between the winner and its neighbor bidders (the number of steps is the same as the number of winners parameter).

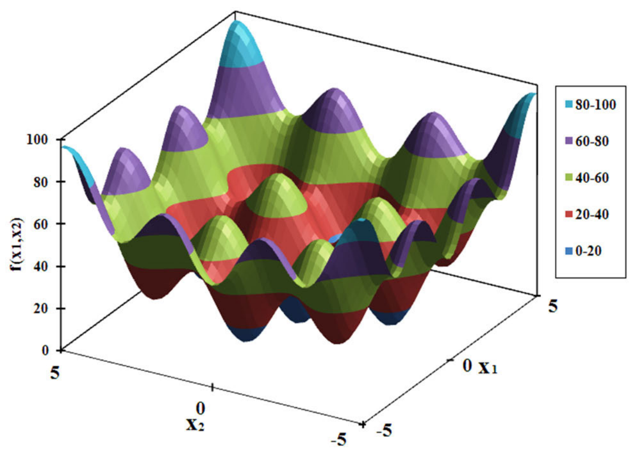

FIGURE 6. Search space for minimization of eggcrate test function.

In Fig. 6(k) all 10 winners from the previous 10 steps take part in an auction. The winner of this auction then takes part in an auction with its neighbor bidders, as shown in Fig. 7(l). The final winner of this final auction is the output of this search process. The exploration is provided by the random selection of the bidders for the auctions and the exploitation is offered by the auctions between the winning bidder in each round and its neighbor bidders.

B. RESULTS OF FEATURE SELECTION

In the next stage of the experiment, the feature selection performance of ABOA when applied to 18 UCI datasets [70]

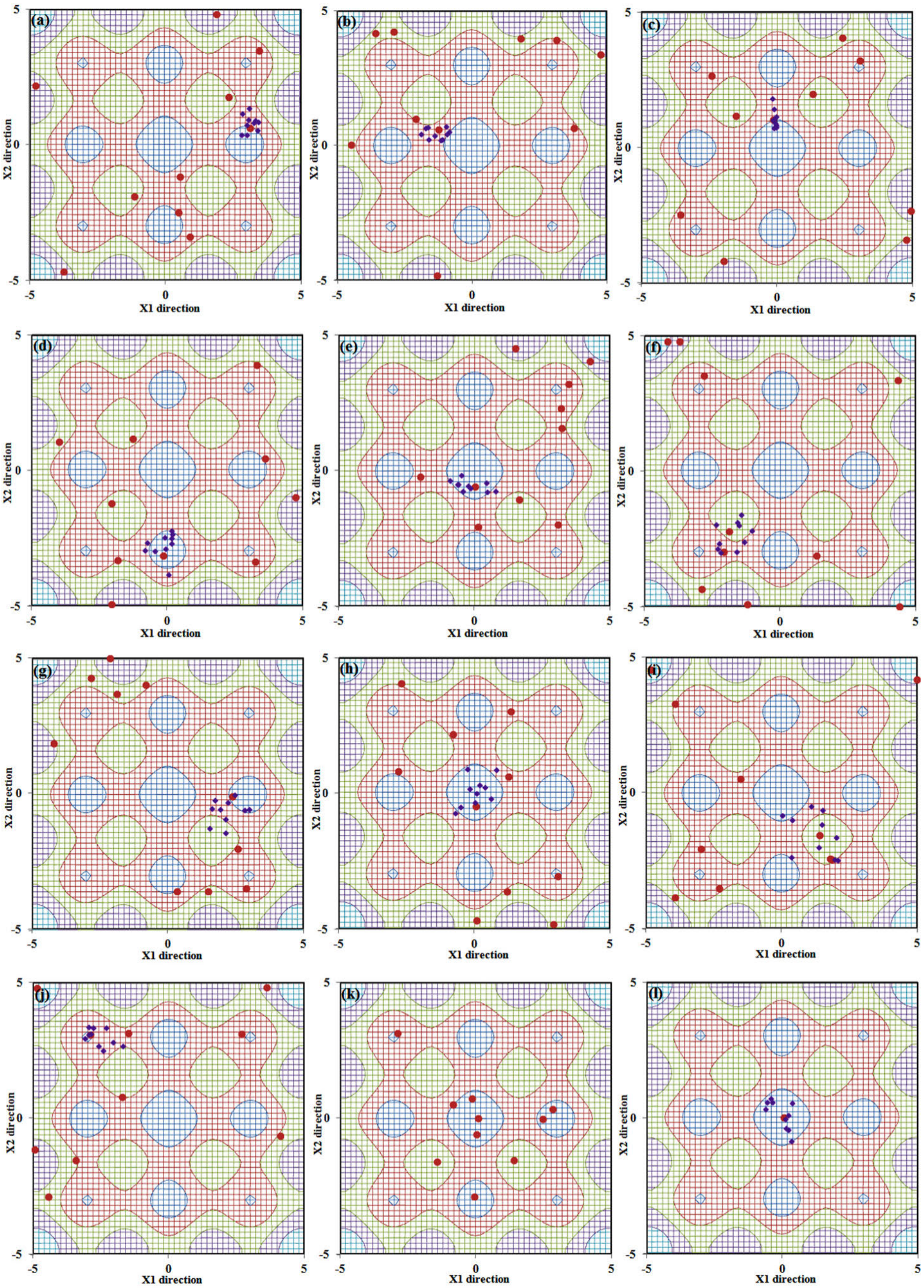

FIGURE 7. Example of ABOA process during eggcrate minimization with contour plot background.

is tested. Using these data provides an easy to comparison approach. Easy to comparison is because of that standard data easily is available and traditional available methods also have used the same data first for the examination in feature selection problem and therefore an easy and fair comparison is achieved. The details of the datasets are provided in Table 6.

In order to set up and prepare the raw data for the model, pre-processing techniques such as removing noise, handling missing values and discretization were performed. A one-dimensional vector with n cells is the representation of each solution (individual) in this experiment, where the number of features in the full feature set is represented by n. Each cell contains one element. Each element is assigned the value of '1' if the selection of the corresponding feature is carried out; otherwise the element is assigned the value of '0'. Fig. 8 shows the details of an example for solution representation.

Based on the example given in Fig. 8, the corresponding features of numbers 2, 3, $n-2$, and n are included in selected

TABLE 6. Details of the UCI tested datasets.

Dataset	No. of features	No. of objects
Breastcancer	9	699
BreastEW	30	569
CongressEW	16	435
Exactly	13	1000
Exactly2	13	1000
HeartEW	13	270
IonosphereEW	34	351
KrvskpEW	36	3196
Lymphography	18	148
M-of-n	13	1000
PenglungEW	325	73
SonarEW	60	208
SpectEW	22	267
Tic-tac-toe	9	958
Vote	16	300
WaveformEW	40	5000
WineEW	13	178
Zoo	16	101

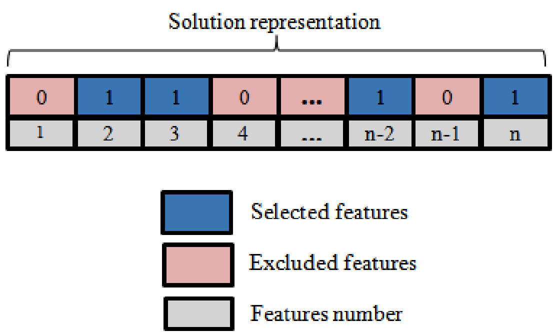

FIGURE 8. Example of solution representation for feature selection.

feature set whilst the features with numbers 1, 4, and n − 1 are not the selected for this example.

As a binary pattern was needed for each solution in feature selection and, in ABOA, the neighborhood of each solution in the search space is the direction of continuous values, in this experiment, a binary vector was presented using the sigmoid function, as in Eq. (6):

$$f(x) = \frac{1}{1+e^{-x}} \tag{6}$$

The input for this equation is the output of Eq. (1), which is the equation used for the generation of neighbor bidders in ABOA. Equation (5) generates a value between '0' and '1'. Values greater than a certain value in this range result in setting the cell in the solution vector to '1'; otherwise the cell is set to '0'. The setting of the solution vector in binary representation can be used for any binary problem.

In this experiment, the rough set approach was applied for feature selection. In order to perform rough set feature selection, the search process looks for a subset of the full feature set that has the highest dependency degree (in the ideal case a dependency degree equal to 1) and the lowest number of features. Therefore, the fitness function should deal with both the dependency degree and the number of features. To calculate fitness function, Eq. (7) is used:

$$F(R) = \gamma_R(D) * \frac{|C| - |R|}{|C|} \tag{7}$$

where R is a subset of the full set, C is the conditional feature set, and D is the decision feature.

The algorithm was run 20 times and the average of 20 selection ratios was calculated for each dataset. The selection ratio was calculated using the ratio of the number of features in the selected subset to the number of features in the full feature set. The average selection ratios and standard deviations of the 20 runs on each dataset are given in Table 7. The classification accuracy is also calculated in order to evaluate the feature selection performance of ABOA. Table 7 also provides the classification accuracy results. Moreover, a comparison of the fitness function of the methods is provided in this table. In this case, three measurements including selection ratio, classification accuracy, and fitness function were used to evaluate and compare the proposed method with other available methods. The best subsets found by ABOA were imported into Rosetta software and this software generates the rules. These rules are used to classify the data. In order to validate the classification results, a 10-fold cross-validation technique is carried out in which 30% of the data is used as the testing set and 70% as the training set. Cross-validation is a technique to evaluate predictive models by partitioning the original sample into a training set to train the model, and a test set to evaluate it. In 10-fold cross-validation, the original sample is randomly partitioned into 10 equal size subsamples. Of the 10 subsamples, a single subsample is retained as the validation data for testing the model, and the remaining 10-1 subsamples are used as training data. The cross-validation process is then repeated 10 times, with each of the 10 subsamples used exactly once as the validation data. The 10 results from the folds are then averaged to produce a single estimation. The whole process of feature selection and classification is illustrated in Fig. 9. The average selection ratio results of ABOA are compared with the results of GA, PSO, ant line optimization (ALO), and the whale optimization algorithm with crossover and mutation (WOA-CM), reported in [68]. These results are provided in Table 8. In this comparison, the two most common algorithms (GA and PSO) in the literature and the two most recently presented algorithms (ALO and WOA-CM) for feature selection were chosen.

In order to determine whether there are major differences between the results presented in Table 8, two statistical analyses are conducted. First, a Friedman test is performed. The result is 39.5333, which is higher than the critical value of 9.49. This indicates that there are differences between the results of the tested algorithms and, therefore, the null hypothesis can be rejected. Second, a Nemenyi post-hoc test is carried out to identify where these differences occur and whether they are significant. The results of the Nemenyi post-hoc test are presented in Table 9. The minimum significant difference (MSD) is equal to 0.08441,

TABLE 7. Results for average selection ratio for ABOA and accuracy of classification.

Dataset	Criteria	Selection ratio		Accuracy		Fitness	
		ABOA	BTLBO-V-ER	ABOA	BTLBO-V-ER	ABOA	BTLBO-V-ER
Breastcancer	Avg.	0.5128	0.6234	0.9684	0.9831	0.0342	0.0237
	Std. dev.	0.0892	0.8172	0.0269	0.0040	0.0028	0.0030
BreastEW	Avg.	0.5552	0.5938	0.9698	0.9971	0.0045	0.0061
	Std. dev.	0.0462	2.8945	0.0243	0.0048	0.0039	0.0043
CongressEW	Avg.	0.5515	0.5841	0.9692	0.9705	0.0248	0.0326
	Std. dev.	0.0646	1.8096	0.0194	0.0058	0.0031	0.0049
Exactly	Avg.	0.5570	0.5739	0.8702	1.0000	0.0045	0.0050
	Std. dev.	0.0448	0.0001	0.0154	0.0000	0.0004	0.0000
Exactly2	Avg.	0.4530	0.5134	0.7599	0.7627	0.1685	0.2383
	Std. dev.	0.0590	3.7277	0.0118	0.0177	0.0047	0.0199
HeartEW	Avg.	0.6008	0.6045	0.8433	0.8759	0.1133	0.1271
	Std. dev.	0.0282	0.9229	0.0159	0.0099	0.0042	0.0098
IonosphereEW	Avg.	0.5331	0.7942	0.9509	0.9869	0.0135	0.0154
	Std. dev.	0.0697	2.2273	0.0103	0.0082	0.0041	0.0081
KrvskpEW	Avg.	0.6570	0.6294	0.9311	0.9855	0.0238	0.0188
	Std. dev.	0.0223	4.0911	0.0320	0.0027	0.0038	0.0019
Lymphography	Avg.	0.4025	0.4932	0.8261	0.9764	0.0189	0.0263
	Std. dev.	0.0198	1.5071	0.0244	0.0251	0.0246	0.0252
M-of-n	Avg.	0.6436	0.7263	0.8841	1.0000	0.0034	0.0050
	Std. dev.	0.0550	0.0049	0.0339	0.0000	0.0023	0.0000
PenglungEW	Avg.	0.3010	0.4842	0.7537	1.0000	0.0032	0.0007
	Std. dev.	0.0424	0.5950	0.0151	0.0000	0.0044	0.0003
SonarEW	Avg.	0.6925	0.5457	0.9270	1.0000	0,0029	0.0023
	Std. dev.	0.0297	2.4731	0.0177	0.0000	0.0014	0.0004
SpectEW	Avg.	0.4127	0.3854	0.8533	0.8673	0.1138	0.1348
	Std. dev.	0.0725	1.7760	0.0277	0.0147	0.0032	0.0138
Tic-tac-toe	Avg.	0.6943	0.5739	0.8381	0.8312	0.2961	0.1758
	Std. dev.	0.0301	0.0021	0.0239	0.0054	0.0032	0.0053
Vote	Avg.	0.3878	0.3184	0.9527	0.9994	0.0242	0.0026
	Std. dev.	0.0267	0.2537	0.0164	0.0030	0.0045	0.0031
WaveformEW	Avg.	0.7362	0.7058	0.7980	0.7820	0.2485	0.2211
	Std. dev.	0.0290	3.1259	0.0253	0.0062	0.0037	0.0061
WineEW	Avg.	0.5839	0.5372	0.9507	1.0000	0.0013	0.0027
	Std. dev.	0.0396	0.5509	0.0250	0.0000	0.0003	0.0005
Zoo	Avg.	0.5969	0.5483	0.9550	0.9238	0.0624	0.0781
	Std. dev.	0.0360	1.0662	0.0221	0.0237	0.0218	0.0228

TABLE 8. Comparison of average selection ratio of ABOA and of other methods in the literature.

Dataset	ABOA	WOA-CM	ALO	PSO	GA
Breastcancer	0.512	0.478	0.698	0.636	0.566
BreastEW	0.555	0.527	0.536	0.552	0.545
CongressEW	0.551	0.403	0.436	0.427	0.414
Exactly	0.557	0.465	0.509	0.750	0.832
Exactly2	0.453	0.404	0.823	0.475	0.475
HeartEW	0.600	0.535	0.793	0.611	0.730
IonosphereEW	0.533	0.424	0.277	0.564	0.509
KrvskpEW	0.657	0.515	0.686	0.578	0.623
Lymphography	0.402	0.456	0.614	0.499	0.614
M-of-n	0.643	0.462	0.852	0.695	0.525
PenglungEW	0.301	0.394	0.505	0.550	0.545
SonarEW	0.692	0.594	0.632	0.520	0.555
SpectEW	0.412	0.366	0.734	0.568	0.534
Tic-tac-toe	0.694	0.767	0.777	0.734	0.761
Vote	0.387	0.463	0.595	0.550	0.414
WaveformEW	0.736	0.635	0.893	0.568	0.632
WineEW	0.583	0.523	0.823	0.643	0.664
Zoo	0.596	0.375	0.873	0.609	0.632

TABLE 9. Nemenyi test results for feature selection.

Algorithm		ABOA	WOA-CM	ALO	PSO	GA
	Mean	0.548483	0.488111	0.669778	0.584944	0.587222
ABOA	0.548483	-	0.060371	**0.121295**	0.036462	0.03874
WOA-CM	0.488111	-	-	**0.181667**	**0.096833**	**0.099111**
ALO	0.669778	-	-	-	**0.084833**	0.082556
PSO	0.584944	-	-	-	-	0.002278
GA	0.587222	-	-	-	-	-

so values higher than MSD denote significant differences. The values higher than MSD are shown in bold in the table.

In addition, the classification accuracy derived from the features selected by ABOA is compared with that of WOA-CM, ALO, PSO, GA and the full feature set.

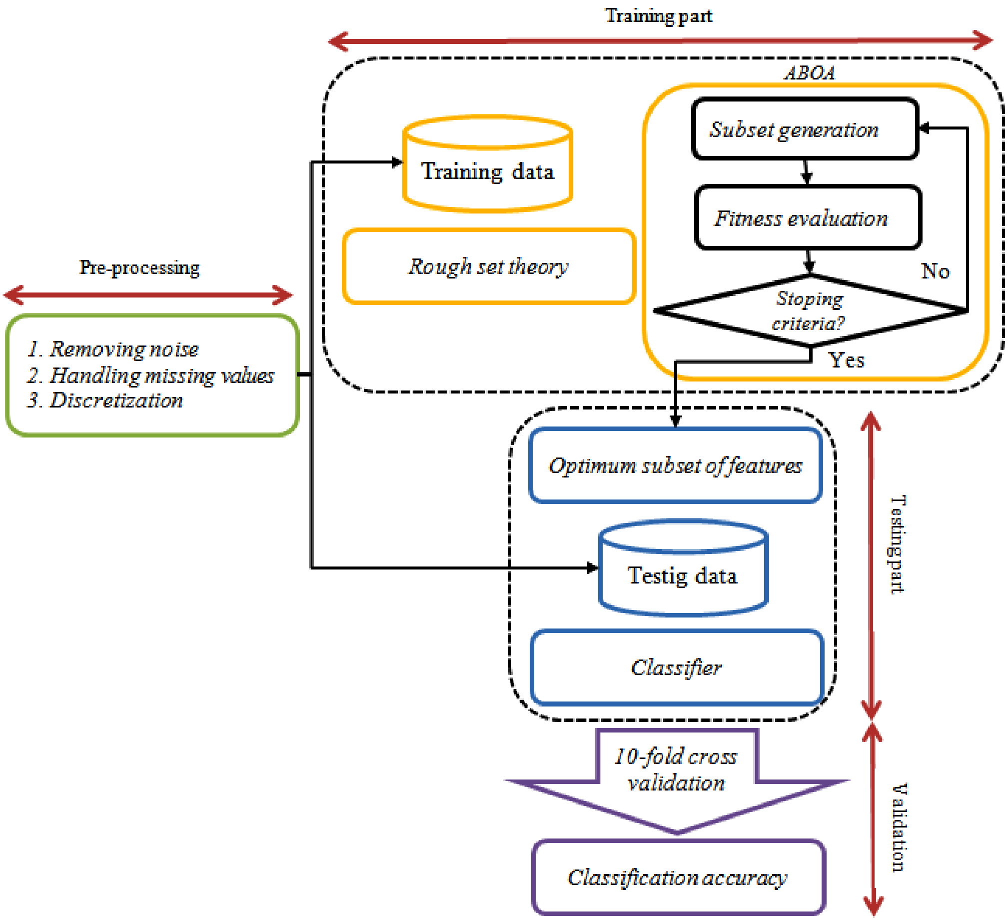

FIGURE 9. Overall process of feature selection by ABOA and classification.

TABLE 10. Comparison of classification accuracy of ABOA and of other methods in the literature.

Dataset	ABOA	WOA-CM	ALO	PSO	GA	Full
Breastcancer	0.968	0.968	0.961	0.954	0.955	0.944
BreastEW	0.969	0.971	0.930	0.941	0.938	0.963
CongressEW	0.969	0.956	0.929	0.937	0.938	0.917
Exactly	0.870	1.000	0.660	0.684	0.666	0.673
Exactly2	0.759	0.742	0.745	0.746	0.757	0.743
HeartEW	0.843	0.807	0.826	0.784	0.822	0.815
IonosphereEW	0.950	0.926	0.866	0.843	0.834	0.866
KrvskpEW	0.931	0.972	0.956	0.942	0.923	0.915
Lymphography	0.826	0.852	0.787	0.692	0.708	0.683
M-of-n	0.884	0.991	0.864	0.864	0.927	0.849
PenglungEW	0.753	0.792	0.627	0.720	0.696	0.951
SonarEW	0.927	0.919	0.738	0.740	0.726	0.620
SpectEW	0.853	0.866	0.801	0.769	0.775	0.831
Tic-tac-toe	0.838	0.785	0.725	0.728	0.713	0.715
Vote	0.952	0.939	0.917	0.894	0.894	0.877
WaveformEW	0.798	0.753	0.773	0.761	0.767	0.768
WineEW	0.950	0.959	0.911	0.950	0.933	0.932
Zoo	0.955	0.980	0.909	0.834	0.884	0.792

The results of this comparison are given in Table 10. It can be seen that the results of ABOA are comparable with those of the above mentioned methods, which confirms the ability of the proposed algorithm with respect to feature selection and classification accuracy.

The Friedman test is also performed to identify whether there are any major differences in the classification accuracy results presented in Table 10. The result is equal to 34.5873 which is greater than the critical value of 10.57. Therefore, there are significant differences between the

TABLE 11. Nemenyi test results for classification accuracy.

Algorithm		ABOA	WOA-CM	ALO	PSO	GA	Full
	Mean	0.889029	0.898778	0.829167	0.821278	0.825333	0.825222
ABOA	0.889029	-	0.009749	0.059863	**0.067751**	**0.063696**	**0.063807**
WOA-CM	0.898778	-	-	**0.069611**	**0.0775**	**0.073444**	**0.073556**
ALO	0.829167	-	-	-	0.007889	0.003833	0.003944
PSO	0.821278	-	-	-	-	0.004056	0.003944
GA	0.825333	-	-	-	-	-	**0.821389**
Full	0.825222	-	-	-	-	-	-

compared methods and, therefore, the null hypothesis can be rejected. The results of the subsequent Nemenyi post-hoc test are presented in Table 11. In this test, the value of MSD is equal to 0.062466. The values greater than MSD are shown in bold in Table 11.

It can be derived from the above that, overall, the ability of ABOA is comparable with that of other methods in the literature. This is due to the algorithm being designed in such a way so as to achieve a balance between exploration and exploitation. In ABOA, exploration is attained by selecting random bidders within the search space, while exploitation is achieved by generating neighbor bidders. The trade-off between exploration and exploitation ensures that ABOA performs well when applied to the feature selection procedure. The good results indicate that this algorithm can be tested on other problems as well.

V. CONCLUSION

This paper proposed a new meta-heuristic algorithm named the auction-based optimization algorithm (ABOA), which is based on the behavior seen at auctions. In an auction, bidders make bids in progressively higher or lower amounts in order to outbid each other. This process can be used for maximization or minimization problems. In the first step of ABOA after initialization, a predefined number of bidders are selected and an auction between them is performed. The winner of the auction is selected and another auction is performed between the winner and a predefined number of its neighbors. The winner of this auction is added to the winner list. This process is iterated until the predefined number of winners is reached. Finally, one more auction is performed between all the winners in the winner list and its neighbors and the winner is the final winner. The algorithm with nine different parameter settings (scenarios) was tested on 25 benchmark test functions. The algorithm with the best results was then applied to 18 UCI benchmark feature selection problems. The feature selection capability and the classification accuracy of the proposed algorithm were compared with that of state-of-the-art results by performing some statistical analyses. The results of the statistical analyses showed that the performance of ABOA was comparable with that of the compared methods. The good performance of ABOA was due to achieving a balance between exploration and exploitation. The results indicate that the proposed algorithm may be suitable for a number of applications and will therefore be tested on real-world optimization problems in future work.

REFERENCES

[1] M. Alzaqebah and S. Abdullah, "Hybrid bee colony optimization for examination timetabling problems," *Comput. Oper. Res.*, vol. 54, pp. 142–154, Feb. 2015.
[2] S. Abdullah and M. Abdolrazzagh-Nezhad, "Fuzzy job-shop scheduling problems: A review," *Inf. Sci.*, vol. 278, pp. 380–407, Sep. 2014.
[3] M. Alzaqebah and S. Abdullah, "An adaptive artificial bee colony and late-acceptance hill-climbing algorithm for examination timetabling," *J. Scheduling*, vol. 17, no. 3, pp. 249–262, Jun. 2014.
[4] S. L. Goh, G. Kendall, N. R. Sabar, and S. Abdullah, "An effective hybrid local search approach for the post enrolment course timetabling problem," *Opsearch*, vol. 57, no. 4, pp. 1131–1163, Dec. 2020.
[5] B. Fox, W. Xiang, and H. Lee, "Industrial applications of the ant colony optimization algorithm," *Int. J. Adv. Manuf. Technol.*, vol. 31, pp. 805–814, Jan. 2007.
[6] B. V. Patil and P. S. V. Nataraj, "An improved bernstein global optimization algorithm for MINLP problems with application in process industry," *Math. Comput. Sci.*, vol. 8, nos. 3–4, pp. 357–377, Sep. 2014.
[7] G. Besiashvili, M. Khachidze, and D. Chokhonelidze, "Application of adaptive genetic algorithm in mining industry," in *Proc. IV Int. Conf. Problems Cybern. Inform. (PCI)*, Sep. 2012, pp. 1–3.
[8] N. S. Jaddi, S. Abdullah, and A. R. Hamdan, "Multi-population cooperative bat algorithm-based optimization of artificial neural network model," *Inf. Sci.*, vol. 294, pp. 628–644, Feb. 2015.
[9] N. S. Jaddi, S. Abdullah, and A. R. Hamdan, "Taguchi-based parameter designing of genetic algorithm for artificial neural network training," in *Proc. Int. Conf. Informat. Creative Multimedia*, Sep. 2013, pp. 278–281.
[10] N. S. Jaddi and S. Abdullah, "Nonlinear great deluge algorithm for rough set attribute reduction," *J. Inf. Sci. Eng.*, vol. 29, no. 1, pp. 49–62, 2013.
[11] N. S. Jaddi and S. Abdullah, "Hybrid of genetic algorithm and great deluge algorithm for rough set attribute reduction," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 21, no. 6, pp. 1737–1750, 2013.
[12] S. S. Shreem, S. Abdullah, and M. Z. A. Nazri, "Hybrid feature selection algorithm using symmetrical uncertainty and a harmony search algorithm," *Int. J. Syst. Sci.*, vol. 47, no. 6, pp. 1312–1329, 2014.
[13] M. Mafarja and S. Abdullah, "A fuzzy record-to-record travel algorithm for solving rough set attribute reduction," *Int. J. Syst. Sci.*, vol. 46, no. 3, pp. 503–512, Feb. 2015.
[14] S. Abdullah, N. R. Sabar, M. Z. A. Nazri, and M. Ayob, "An exponential monte-carlo algorithm for feature selection problems," *Comput. Ind. Eng.*, vol. 67, pp. 160–167, Jan. 2014.
[15] N. S. Jaddi and S. Abdullah, "Optimization of neural network using kidney-inspired algorithm with control of filtration rate and chaotic map for real-world rainfall forecasting," *Eng. Appl. Artif. Intell.*, vol. 67, pp. 246–259, Jan. 2018.
[16] N. S. Jaddi, S. Abdullah, and M. A. Malek, "Master-leader-slave cuckoo search with parameter control for ANN optimization and its real-world application to water quality prediction," *PLoS ONE*, vol. 12, no. 1, Jan. 2017, Art. no. e0170372.
[17] M. Abdolrazzagh-Nezhad and S. Izadpanah, "A modified electromagnetic-like mechanism for rough set attribute reduction," in *Proc. Inf. Softw. Technol.*, Cham, Switzerland, 2016, pp. 356–368.
[18] N. S. Jaddi and M. S. Abadeh, "Gene selection of non-small cell lung cancer data for adjuvant chemotherapy decision using cell separation algorithm," *Int. J. Speech Technol.*, vol. 50, no. 11, pp. 3822–3836, Nov. 2020.
[19] N. S. Jaddi and M. S. Abadeh, "DNA methylation-based age prediction using cell separation algorithm," *Comput. Biol. Med.*, vol. 121, Jun. 2020, Art. no. 103747.

[20] X.-S. Yang, ''1—Optimization and metaheuristic algorithms in engineering,'' in *Metaheuristics in Water, Geotechnical and Transport Engineering*, X.-S. Yang, A. H. Gandomi, S. Talatahari, and A. H. Alavi, Eds. Oxford, U.K.: Elsevier, 2013, pp. 1–23.

[21] K. S. Lee and Z. W. Geem, ''A new meta-heuristic algorithm for continuous engineering optimization: Harmony search theory and practice,'' *Comput. Methods Appl. Mech. Eng.*, vol. 194, nos. 36–38, pp. 3902–3933, Sep. 2005.

[22] J. Alvankarian, N. S. Jaddi, M. Vakilian, F. Barantalab, R. Jamal, and B. Y. Majlis, ''Consideration of nonuniformity in elongation of microstructures in a mechanically tunable microfluidic device for size-based isolation of microparticles,'' *J. Microelectromech. Syst.*, vol. 24, no. 2, pp. 309–318, Apr. 2015.

[23] J.-F. Connolly, E. Granger, and R. Sabourin, ''An adaptive classification system for video-based face recognition,'' *Inf. Sci.*, vol. 192, pp. 50–70, Jun. 2012.

[24] E. Cuevas, D. Oliva, D. Zaldivar, M. Pérez, and R. Rojas, ''Circle detection algorithm based on electromagnetism-like optimization,'' in *Handbook of Optimization*, vol. 38, I. Zelinka, V. Snášel, and A. Abraham, Eds. Berlin, Germany: Springer, 2013, pp. 907–934.

[25] E. Cuevas, D. Oliva, M. Diaz, D. Zaldivar, M. Perez-Cisneros, and G. Pajares, ''White blood cell segmentation by circle detection using electromagnetism-like optimization,'' *Comput. Math. Methods Med.*, vol. 2013, p. 15, Dec. 2013.

[26] D. Picard, A. Revel, and M. Cord, ''An application of swarm intelligence to distributed image retrieval,'' *Inf. Sci.*, vol. 192, pp. 71–81, Jun. 2012.

[27] Y.-C. Liang and J. R. C. Juarez, ''A normalization method for solving the combined economic and emission dispatch problem with meta-heuristic algorithms,'' *Int. J. Electr. Power Energy Syst.*, vol. 54, pp. 163–186, Jan. 2014.

[28] M. Chamba and O. Ano, ''Economic dispatch of energy and reserve in competitive markets using meta-heuristic algorithms,'' *IEEE Latin Amer. Trans.*, vol. 11, no. 1, pp. 473–478, Feb. 2013.

[29] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. San Mateo, CA, USA: Morgan Kaufmann, 2011.

[30] C. He, Y. Zhang, and D. Gong, ''A pseudo-label guided artificial bee colony algorithm for hyperspectral band selection,'' *Remote Sens.*, vol. 12, no. 20, p. 3456, Oct. 2020.

[31] X.-H. Wang, Y. Zhang, X.-Y. Sun, Y.-L. Wang, and C.-H. Du, ''Multi-objective feature selection based on artificial bee colony: An acceleration approach with variable sample size,'' *Appl. Soft Comput.*, vol. 88, Mar. 2020, Art. no. 106041.

[32] Z. Yong, H. Chun-Lin, S. Xian-Fang, and S. Xiao-Yan, ''A multi-strategy integrated multi-objective artificial bee colony for unsupervised band selection of hyperspectral images,'' *Swarm Evol. Comput.*, vol. 60, Feb. 2021, Art. no. 100806.

[33] L. Abualigah, D. Yousri, M. A. Elaziz, A. A. Ewees, M. A. A. Al-Qaness, and A. H. Gandomi, ''Aquila optimizer: A novel meta-heuristic optimization algorithm,'' *Comput. Ind. Eng.*, vol. 157, Jul. 2021, Art. no. 107250.

[34] L. M. Abualigah, A. T. Khader, and E. S. Hanandeh, ''A new feature selection method to improve the document clustering using particle swarm optimization algorithm,'' *J. Comput. Sci.*, vol. 25, pp. 456–466, Mar. 2018.

[35] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, and A. H. Gandomi, ''The arithmetic optimization algorithm,'' *Comput. Methods Appl. Mech. Eng.*, vol. 376, Apr. 2021, Art. no. 113609.

[36] L. M. Q. Abualigah, *Feature Selection and Enhanced Krill Herd Algorithm for Text Document Clustering*. Berlin, Germany: Springer, 2019.

[37] Z. Pawlak, ''Rough sets,'' *Int. J. Comput. Inf. Sci.*, vol. 11, no. 5, pp. 341–356, Oct. 1982.

[38] F. A. Hashim, K. Hussain, E. H. Houssein, M. S. Mabrouk, and W. Al-Atabany, ''Archimedes optimization algorithm: A new Metaheuristic algorithm for solving optimization problems,'' *Int. J. Speech Technol.*, vol. 51, no. 3, pp. 1531–1551, Mar. 2021.

[39] I. Ahmadianfar, O. Bozorg-Haddad, and X. Chu, ''Gradient-based optimizer: A new metaheuristic optimization algorithm,'' *Inf. Sci.*, vol. 540, pp. 131–159, Nov. 2020.

[40] H. N. Ghafil and K. Jármai, ''Dynamic differential annealed optimization: New metaheuristic optimization algorithm for engineering applications,'' *Appl. Soft Comput.*, vol. 93, Aug. 2020, Art. no. 106392.

[41] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili, ''Equilibrium optimizer: A novel optimization algorithm,'' *Knowl.-Based Syst.*, vol. 191, Mar. 2020, Art. no. 105190.

[42] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon, ''Optimization by simulated annealing: An experimental evaluation; part I, graph partitioning,'' *Oper. Res.*, vol. 37, no. 6, pp. 865–892, Dec. 1989.

[43] G. Dueck, ''New optimization heuristics: The great deluge algorithm and the record-to-record travel,'' *J. Comput. Phys.*, vol. 104, no. 1, pp. 86–92, Jan. 1993.

[44] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, ''GSA: A gravitational search algorithm,'' *Inf. Sci.*, vol. 179, no. 13, pp. 2232–2248, Jun. 2009.

[45] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Cambridge, MA, USA: MIT Press, 1992.

[46] L. Bianchi, L. M. Gambardella, and M. Dorigo, ''An ant colony optimization approach to the probabilistic traveling salesman problem,'' presented at the 7th Int. Conf. Parallel Problem Solving Nature, 2002.

[47] H. S. Hosseini, ''Problem solving by intelligent water drops,'' in *Proc. IEEE Congr. Evol. Comput.*, Sep. 2007, pp. 3226–3231.

[48] J. Kennedy and R. Eberhart, ''Particle swarm optimization,'' in *Proc. IEEE Int. Conf. Neural Netw.*, vol. 4, Nov. 1995, pp. 1942–1948.

[49] X.-L. Li and J.-X. Qian, ''Studies on artificial fish swarm optimization algorithm based on decomposition and coordination techniques,'' *Chin. J. Circuits Syst.*, vol. 1, pp. 1–6, Jan. 2003.

[50] A. Askarzadeh, ''Bird mating optimizer: An optimization algorithm inspired by bird mating strategies,'' *Commun. Nonlinear Sci. Numer. Simul.*, vol. 19, no. 4, pp. 1213–1228, Apr. 2014.

[51] X.-S. Yang, ''A new metaheuristic bat-inspired algorithm,'' in *Nature Inspired Cooperative Strategies for Optimization (NICSO)*, vol. 284, J. González, D. Pelta, C. Cruz, G. Terrazas, and N. Krasnogor, Eds. Berlin, Germany: Springer, 2010, pp. 65–74.

[52] D. T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi, ''The bees algorithm,'' Manuf. Eng. Centre, Cardiff Univ., Cardiff, U.K., Tech. Rep. ICINCO2008, 2005.

[53] M. Fathian, B. Amiri, and A. Maroosi, ''Application of honey-bee mating optimization algorithm on clustering,'' *Appl. Math. Comput.*, vol. 190, no. 2, pp. 1502–1513, Jul. 2007.

[54] X.-S. Yang, ''Firefly algorithm, Lévy flights and global optimization,'' in *Research and Development in Intelligent Systems XXVI*, M. Bramer, R. Ellis, and M. Petridis, Eds. London, U.K.: Springer, 2010, pp. 209–218.

[55] X.-S. Yang and S. Deb, ''Cuckoo search via Lévy flights,'' in *Proc. World Congr. Nature Biologically Inspired Comput. (NaBIC)*, 2009, pp. 210–214.

[56] Z. W. Geem, J. H. Kim, and G. V. Loganathan, ''A new heuristic optimization algorithm: Harmony search,'' *Simulation*, vol. 76, no. 2, pp. 60–68, Feb. 2001.

[57] A. Hatamlou, ''Black hole: A new heuristic optimization approach for data clustering,'' *Inf. Sci.*, vol. 222, pp. 175–184, Feb. 2013.

[58] E. Atashpaz-Gargari and C. Lucas, ''Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition,'' in *Proc. IEEE Congr. Evol. Comput.*, Sep. 2007, pp. 4661–4667.

[59] N. S. Jaddi, J. Alvankarian, and S. Abdullah, ''Kidney-inspired algorithm for optimization problems,'' *Commun. Nonlinear Sci. Numer. Simul.*, vol. 42, pp. 358–369, Jan. 2017.

[60] Y. Masoudi-Sobhanzadeh, Y. Omidi, M. Amanlou, and A. Masoudi-Nejad, ''Trader as a new optimization algorithm predicts drug-target interactions efficiently,'' *Sci. Rep.*, vol. 9, no. 1, pp. 1–14, Dec. 2019.

[61] Y. Chen, D. Miao, and R. Wang, ''A rough set approach to feature selection based on ant colony optimization,'' *Pattern Recognit. Lett.*, vol. 31, no. 3, pp. 226–233, Feb. 2010.

[62] Q. Hu, X. Li, and D. Yu, ''Analysis on classification performance of rough set based reducts,'' in *Proc. PRICAI Trends Artif. Intell.*, 2006, pp. 423–433.

[63] Y.-C. Hu, ''Rough sets for pattern classification using pairwise-comparison-based tables,'' *Appl. Math. Model.*, vol. 37, nos. 12–13, pp. 7330–7337, Jul. 2013.

[64] N. S. Jaddi and S. Abdullah, ''An interactive rough set attribute reduction using great deluge algorithm,'' in *Advances in Visual Informatics*. Cham, Switzerland: Springer, 2013, pp. 285–299.

[65] J. Liang, F. Wang, C. Dang, and Y. Qian, ''A group incremental approach to feature selection applying rough set technique,'' *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 2, pp. 294–308, Feb. 2014.

[66] M. Majdi, S. Abdullah, and N. S. Jaddi, ''Fuzzy population-based metaheuristic approaches for attribute reduction in rough set theory,'' *World Acad. Sci., Eng. Technol., Int. J. Comput., Elect., Automat., Control Inf. Eng.*, vol. 9, no. 1, pp. 2065–2073, 2015.

[67] A. Zeng, T. Li, D. Liu, J. Zhang, and H. Chen, ''A fuzzy rough set approach for incremental feature selection on hybrid information systems,'' *Fuzzy Sets Syst.*, vol. 258, pp. 39–60, Jan. 2015.

[68] M. Mafarja and S. Mirjalili, ''Whale optimization approaches for wrapper feature selection,'' *Appl. Soft Comput.*, vol. 62, pp. 441–453, Jan. 2018.
[69] M. M. Mafarja and S. Mirjalili, ''Hybrid whale optimization algorithm with simulated annealing for feature selection,'' *Neurocomputing*, vol. 260, pp. 302–312, Oct. 2017.
[70] C. L. Blake and C. J. Merz. (1998). *UCI Repository of Machine Learning Databases*. [Online]. Available: http://www.ics.uci.edu/~mlearn/

NAJMEH SADAT JADDI received the B.S. degree in software engineering from Staffordshire University, Stoke-on-Trent, U.K., (Malaysia Campus), in 2008. She joined the Faculty of Information Science and Technology, National University of Malaysia, in 2008, where she received the M.S. and Ph.D. degrees in information technology in 2011 and 2014, respectively. Her current research interests include data mining, artificial intelligence, machine learning, artificial neural network, deep learning, optimization algorithms and applications, and mathematical modeling.

SALWANI ABDULLAH received the B.Sc. degree in computer science from the University of Technology Malaysia and the master's degree specializing in computer science from The National University of Malaysia. She received the Ph.D. degree in computer science from the University of Nottingham, U.K. She is currently a Professor in computational optimization with the Faculty of Information Science and Technology, The National University of Malaysia. Her research interest falls under artificial intelligence and operation research, particularly computational optimization algorithms (heuristic and meta-heuristic, evolutionary algorithms, and local search) that involve different real-world applications for single- and multi-objective continuous and combinatorial optimization problems, such as timetabling, scheduling, routing, nurse rostering, dynamic optimization, data mining problems (feature selection, clustering, classification, and time-series prediction), intrusion detection, and search-based software testing.